

November Project

Lance Merrick

Western Wheat Quality Lab Data

Questions:

My main question is to what extent can we use early generation traits to predict the Milling and product traits.

What are the correlations among the traits?

What are the heritabilities?

How does the variation due to environment differ among the traits?

Are there any pedigrees that perform particularly well or particularly bad? Any clusters?

What should I select? A good club wheat has high cake volume, low gluten strength, high test weight, high break flour and high flour yield. Our stakeholders have asked me to increase kernel size and test weight and reduce LDOPA as well. Can you make any suggestions for crosses that I might want to make?

Data Exploration

Structure

```
str(qdata)
```

```
## 'data.frame': 258 obs. of 35 variables:
## $ NURSCO : int 95 95 95 95 95 95 95 111 95 111 ...
## $ NURNAME : Factor w/ 9 levels "2ND CHANCE PRELIMINA",...: 7 7 7 7 7 7 7 7 7 7 ...
## $ SAMPLE : int 19 20 21 22 23 27 28 2 29 3 ...
## $ entry : int 19 20 21 22 23 27 28 2 29 3 ...
## $ CLASS : Factor w/ 2 levels "CLUB","SWW": 1 1 1 1 1 1 1 1 1 1 ...
## $ NAMET : Factor w/ 138 levels "08X035-0-0-2*CAW",...: 1 2 3 4 5 6 7 7 8 8 ...
## $ pedigree: Factor w/ 61 levels "", "3J030235-1//Bruehl -1 -4",...: 50 51 46 58 59 14 18 18 19 19 ...
## $ LOCATION: Factor w/ 5 levels "FARMINGTON","PENDLETON",...: 2 2 2 2 2 2 2 2 2 2 ...
## $ loccode : Factor w/ 2 levels "ADV","PUL": 1 1 1 1 1 1 1 1 1 1 ...
## $ year1 : int 2018 2018 2018 2018 2018 2018 2018 2018 2019 2018 2019 ...
## $ year0 : int 2017 2017 2017 2017 2017 2017 2017 2017 2018 2017 2018 ...
## $ mmbran : num 3.36 3.62 3.66 3.43 4.08 4.27 3.66 4.18 4.19 4.67 ...
## $ mmflour : num 4.2 4.05 3.52 3.87 3.99 3.92 3.82 6.25 3.93 5.96 ...
## $ mmmids : num 8.98 12.1 12.4 11.9 11.1 12.6 11 11.3 12.1 10.6 ...
## $ microsd : num 2 1.4 2.6 1.3 1.9 1.1 1.1 -9 0.5 -9 ...
## $ FSIDS : num 6.4 7 7.9 5 7.3 4.4 4.7 5.3 4.1 6.1 ...
## $ FSR_C : num 79 78.6 79.4 74 83.8 79.1 76.4 NA 80.6 NA ...
## $ FSR_L : num 105.1 102.3 116.2 92.5 106.6 ...
## $ FSR_W : num 47.2 47.1 48 45.9 49.5 49.6 49.3 53.2 47.8 50.4 ...
## $ FSR_S : num 98.8 101.2 72.2 68.3 73.6 ...
## $ LDOPA : num 1.579 1.406 0.748 0.825 0.681 ...
## $ TWT : num 59.1 59.6 62.4 64.5 64.1 63.4 64.3 63.6 64.4 63 ...
## $ SKHRD : num 31 33.7 38.5 42 46.9 ...
## $ SKHRDSD : num 16.4 16.2 16.9 14.7 18.4 17.6 18.3 16.4 15.3 15.3 ...
## $ SKWT : num 29.4 31 35.1 34.6 32.7 35.4 34.9 33.7 32.8 32.7 ...
```

```
## $ SKWTS : num 8.07 8.94 9.94 8.67 10.4 9.53 10.7 8.52 7.81 7.62 ...
## $ SKSIZE : num 2.35 2.4 2.59 2.69 2.65 2.67 2.62 2.53 2.67 2.61 ...
## $ SKSIZESD : num 0.34 0.33 0.38 0.36 0.35 0.39 0.35 0.37 0.34 0.35 ...
## $ FPROT : num 9 8.2 8.6 8.9 10 8.8 9 7.8 7.7 7.9 ...
## $ FASH : num 0.41 0.45 0.38 0.33 0.3 0.35 0.32 0.33 0.39 0.32 ...
## $ FYELD : num 72.2 72.4 69.1 71.2 69.6 69.9 70.5 69.8 72.5 70.3 ...
## $ MSCOR : num 86.9 84.6 84.9 90.8 90.6 87.8 90.5 89 88.6 90.3 ...
## $ BKFYELD : num 52.9 52.4 47.8 52.5 47.9 47.2 51.9 53.5 51.2 53 ...
## $ CODI : num 9.41 9.48 9.51 9.65 9.18 9.13 9.26 9.31 9.39 9.44 ...
## $ CAVOL : int 1360 1275 1205 1280 1205 1155 1230 1260 1250 1295 ...
```

#We need to convert most of the description variables to factors

```
qdata$NURSCO=as.factor(qdata$NURSCO)
qdata$SAMPLE=as.factor(qdata$SAMPLE)
qdata$entry=as.factor(qdata$entry)
qdata$year1=as.factor(qdata$year1)
qdata$year0=as.factor(qdata$year0)
```

Summary

Summary of the Description Variables

```
summary(qdata[,1:11])
```

```
##      NURSCO      NURNAME      SAMPLE      entry
## 95      :53  SOFT ELITE      :202  7      : 7  53      : 8
## 111     :53  CARA/XERPHA DOUBLE H: 23  8      : 7  54      : 8
## 92      :49  2ND.CHANCE.PRELIMINA: 7  12      : 7  52      : 7
## 104     :25  UNREPLICATED      : 7  15      : 7  10      : 6
## 94      :23  CLUB WHEAT SAMPLES T: 6  16      : 7  50      : 6
## 106     :18  RHT2 UNREPLICATED   : 5  18      : 7  51      : 6
## (Other):37  (Other)              : 8  (Other):216  (Other):217
##  CLASS      NAMET      pedigree
## CLUB:187  ARS-CRESCENT: 13  CARA/XERPHA      : 75
## SWW : 71  PRITCHETT   : 12
##      BRUEHL      : 9  Dusty//MDN sib/Dusty//WA7665/RULO: 13
##      JASPER      : 9  3J030235-1//Bruehl -1 -4      : 12
##      BOBTAIL     : 8  J950081-0001/Brundage 96 - 2      : 9
##      CASTELLA    : 5  UNA(NS1971)/5/Oasis/4/Luke//Itana/: 9
##      (Other)     :202  (Other)              :109
##  LOCATION  loccode  year1  year0
## FARMINGTON : 49  ADV:183  2016:75  2015:75
## PENDLETON  :130  PUL: 75  2017:62  2016:62
## PULLMAN    : 38      2018:55  2017:55
## WALLA WALLA: 23      2019:66  2018:66
## WALLA.WALLA: 18
##
##
```

Summary of Variables

#The describe function returns typical statistical description measurements including skew and kurtosis

```
describe(qdata[,12:35])
```

```
##      vars  n  mean  sd  median trimmed  mad  min  max
```

## mbran	1	258	4.54	0.68	4.64	4.55	0.80	3.08	6.03
## mmflour	2	258	4.25	1.39	3.78	4.12	1.04	2.15	7.99
## mmmids	3	258	12.01	1.11	12.00	12.02	1.19	8.86	15.00
## microsd	4	183	-2.48	4.93	0.60	-2.25	1.48	-9.00	4.00
## FSDS	5	258	7.02	1.96	7.00	6.91	2.08	2.60	13.10
## FSR_C	6	149	67.07	6.61	65.20	66.46	5.34	57.10	84.80
## FSR_L	7	149	82.47	14.25	81.80	81.53	16.61	57.90	120.90
## FSR_W	8	258	52.59	3.22	52.90	52.65	2.82	45.80	60.50
## FSR_S	9	149	88.18	9.10	89.30	88.75	7.56	66.70	108.50
## LDOPA	10	201	1.02	0.34	0.96	1.00	0.38	0.37	2.13
## TWT	11	258	61.57	2.01	61.70	61.66	2.08	55.40	65.70
## SKHRD	12	258	28.28	10.64	28.60	28.50	10.24	-1.20	52.20
## SKHRDSD	13	258	16.16	1.79	16.10	16.16	2.08	11.40	21.00
## SKWT	14	258	34.52	4.60	34.40	34.45	4.15	22.10	48.60
## SKWTSD	15	258	8.85	1.30	8.79	8.81	1.42	6.30	12.90
## SKSIZE	16	258	2.60	0.14	2.60	2.60	0.15	2.20	3.10
## SKSIZESD	17	258	0.34	0.04	0.34	0.34	0.06	0.25	0.43
## FPROT	18	258	8.94	1.38	8.90	8.89	1.33	5.90	12.40
## FASH	19	258	0.36	0.05	0.36	0.36	0.04	0.27	0.54
## FYELD	20	258	70.22	1.79	70.30	70.31	1.78	64.10	75.40
## MSCOR	21	258	87.43	3.40	88.00	87.70	3.11	73.20	96.10
## BKFYELD	22	258	48.58	3.79	48.30	48.57	4.60	39.60	56.90
## CODI	23	258	9.41	0.24	9.43	9.42	0.24	8.74	9.99
## CAVOL	24	155	1239.77	57.53	1245.00	1242.56	51.89	1020.00	1380.00
##			range	skew	kurtosis	se			
## mbran			2.95	-0.10	-0.92	0.04			
## mmflour			5.84	0.83	-0.48	0.09			
## mmmids			6.14	-0.08	-0.36	0.07			
## microsd			13.00	-0.54	-1.67	0.36			
## FSDS			10.50	0.43	-0.17	0.12			
## FSR_C			27.70	0.83	-0.21	0.54			
## FSR_L			63.00	0.51	-0.31	1.17			
## FSR_W			14.70	-0.17	-0.30	0.20			
## FSR_S			41.80	-0.55	0.09	0.75			
## LDOPA			1.75	0.58	-0.17	0.02			
## TWT			10.30	-0.45	0.13	0.13			
## SKHRD			53.40	-0.17	-0.24	0.66			
## SKHRDSD			9.60	0.06	-0.38	0.11			
## SKWT			26.50	0.17	0.02	0.29			
## SKWTSD			6.60	0.34	-0.28	0.08			
## SKSIZE			0.90	0.05	0.14	0.01			
## SKSIZESD			0.18	0.48	-0.77	0.00			
## FPROT			6.50	0.31	-0.38	0.09			
## FASH			0.27	0.62	0.91	0.00			
## FYELD			11.30	-0.40	0.14	0.11			
## MSCOR			22.90	-0.92	1.58	0.21			
## BKFYELD			17.30	0.05	-1.05	0.24			
## CODI			1.25	-0.19	-0.22	0.01			
## CAVOL			360.00	-0.58	1.02	4.62			

Change WALLA WALLA

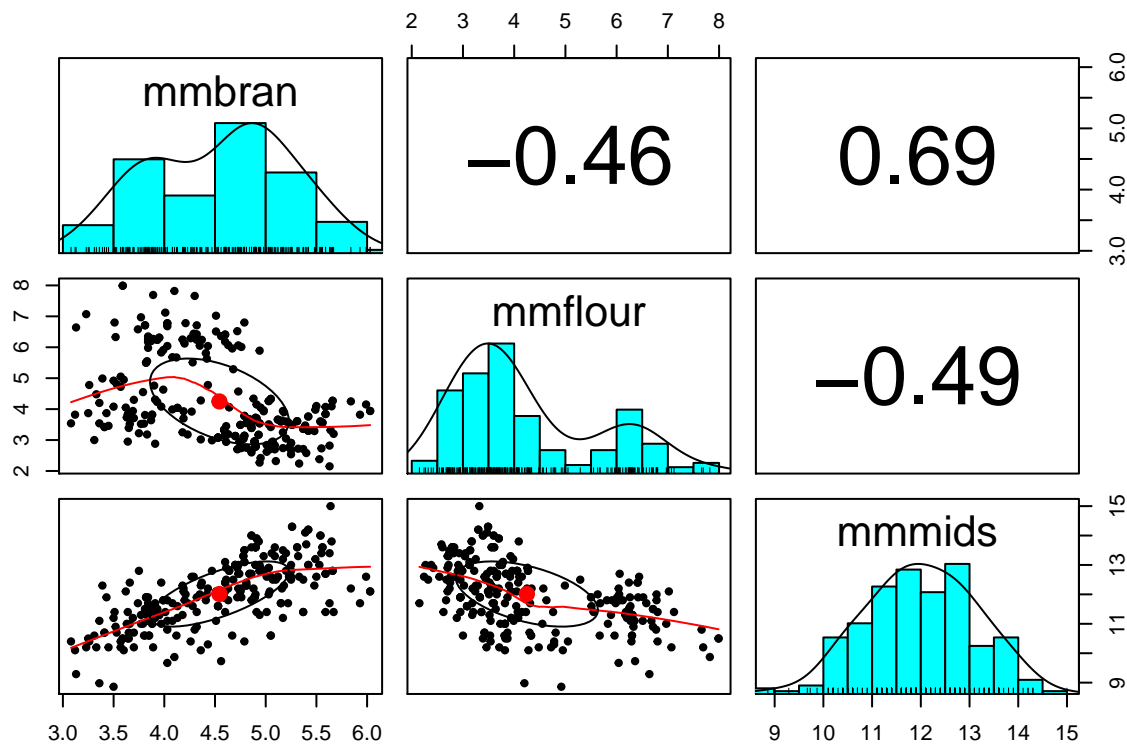
```
#The variable for the WALLA WALLA location is not constant.
levels(qdata$LOCATION)
```

```
## [1] "FARMINGTON" "PENDLETON" "PULLMAN" "WALLA WALLA" "WALLA.WALLA"
levels(qdata$LOCATION)[5] <-"WALLA WALLA" #You can simply rename the levels in a dataset. This renames
levels(qdata$LOCATION)
```

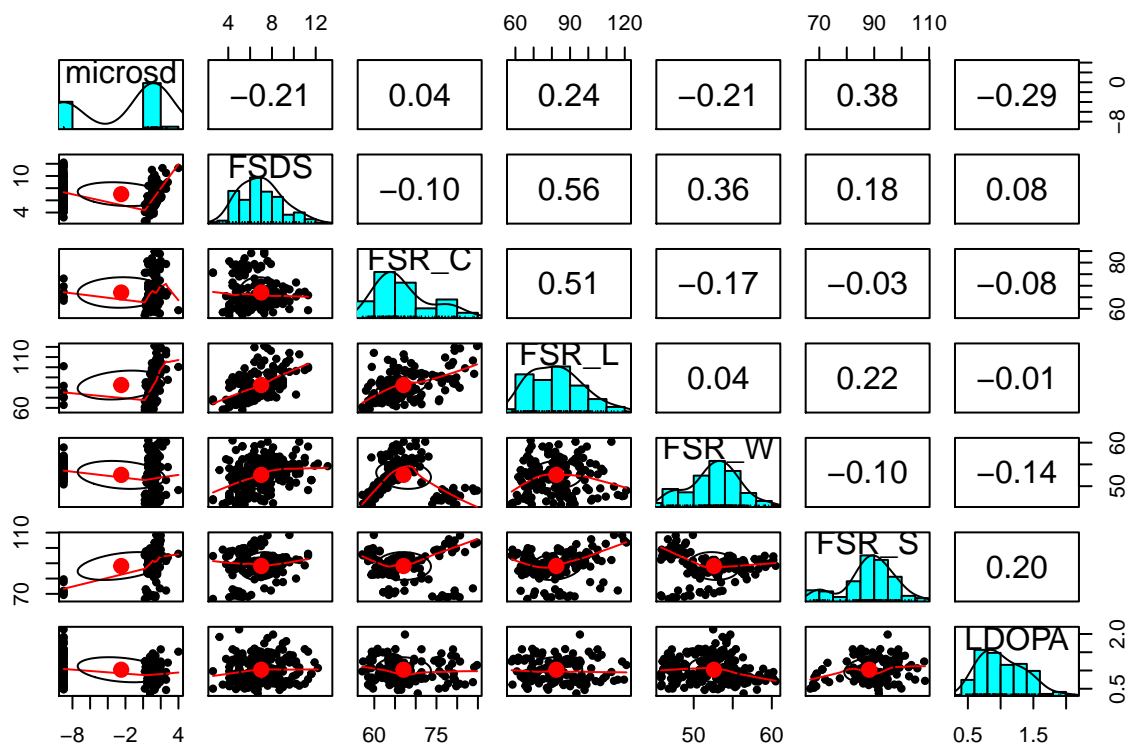
```
## [1] "FARMINGTON" "PENDLETON" "PULLMAN" "WALLA WALLA"
```

Histograms and Distribution

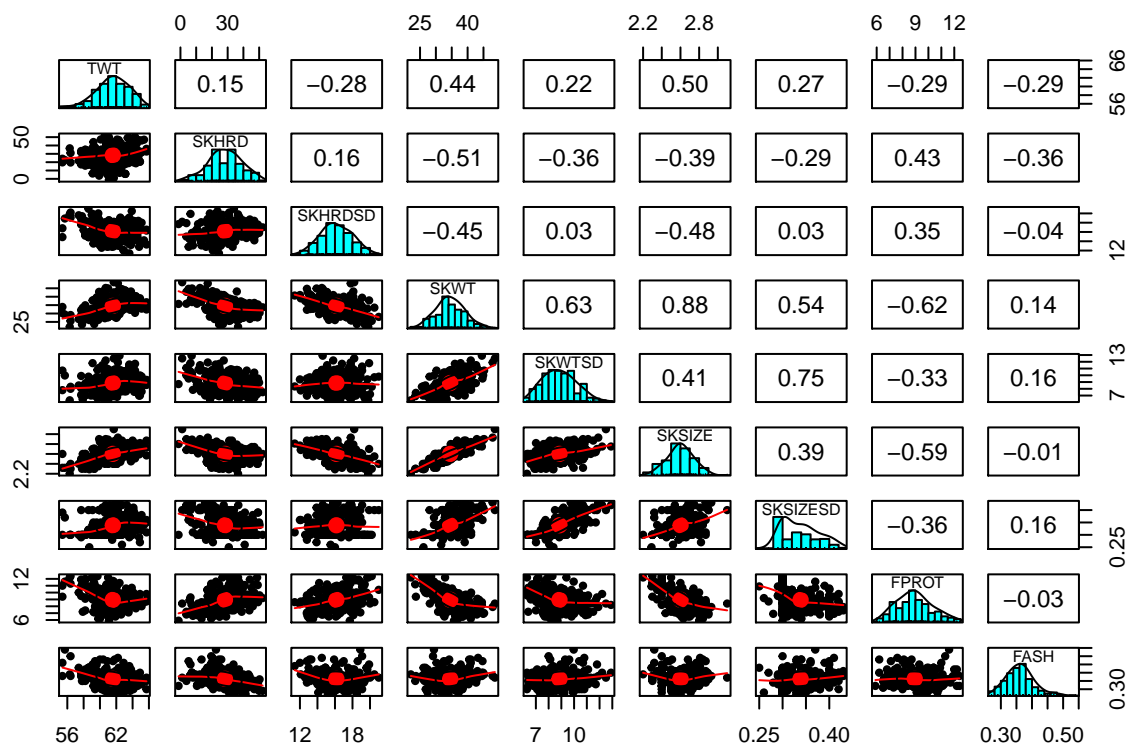
```
#I like this function from the package psych, because it displays histograms, scatterplots and correlat
pairs.panels(qdata[,12:14])
```



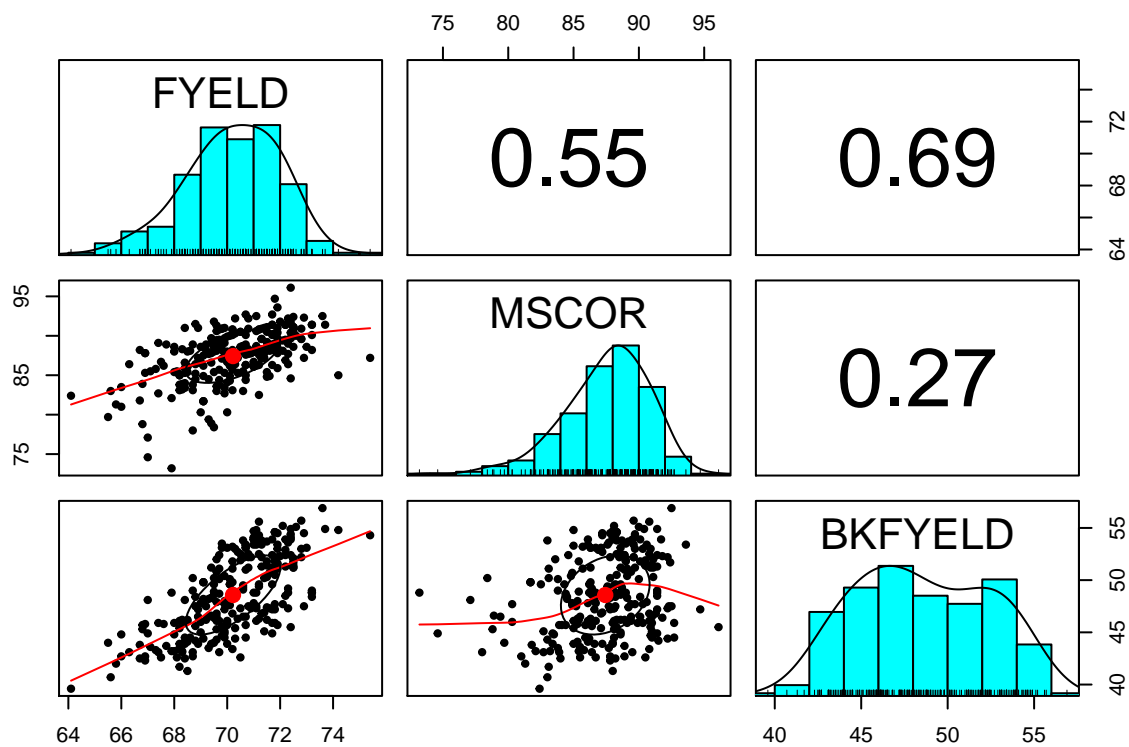
```
pairs.panels(qdata[,15:21])
```



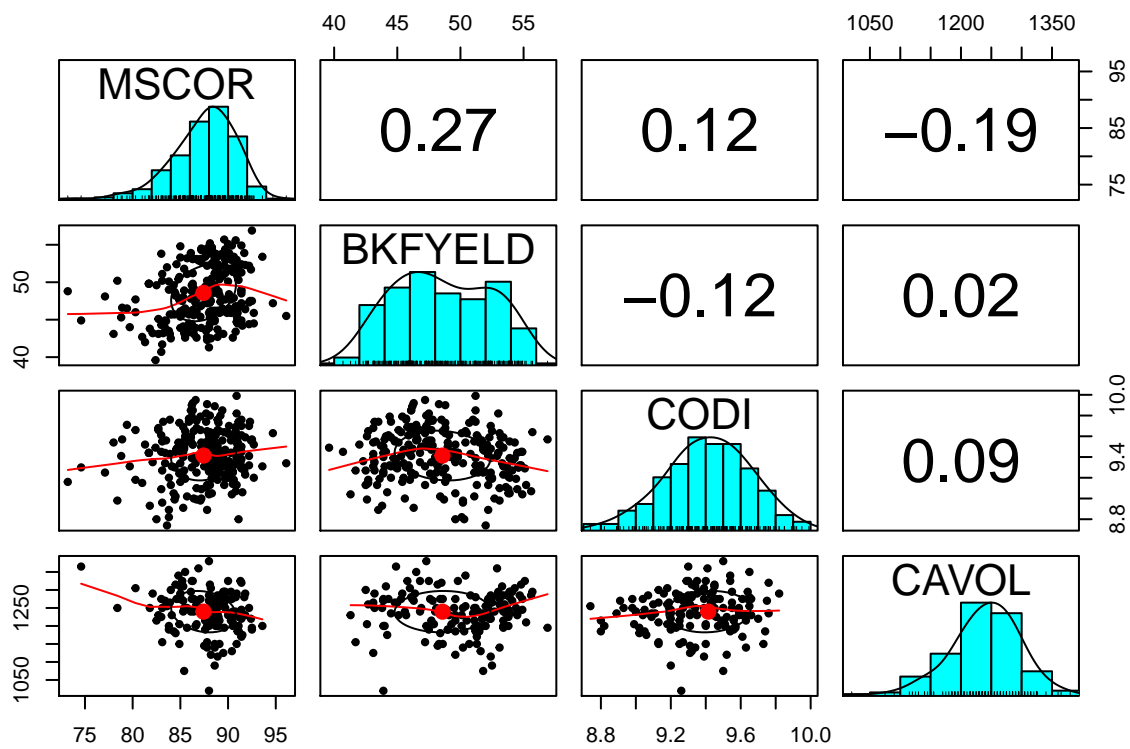
```
pairs.panels(qdata[,22:30])
```



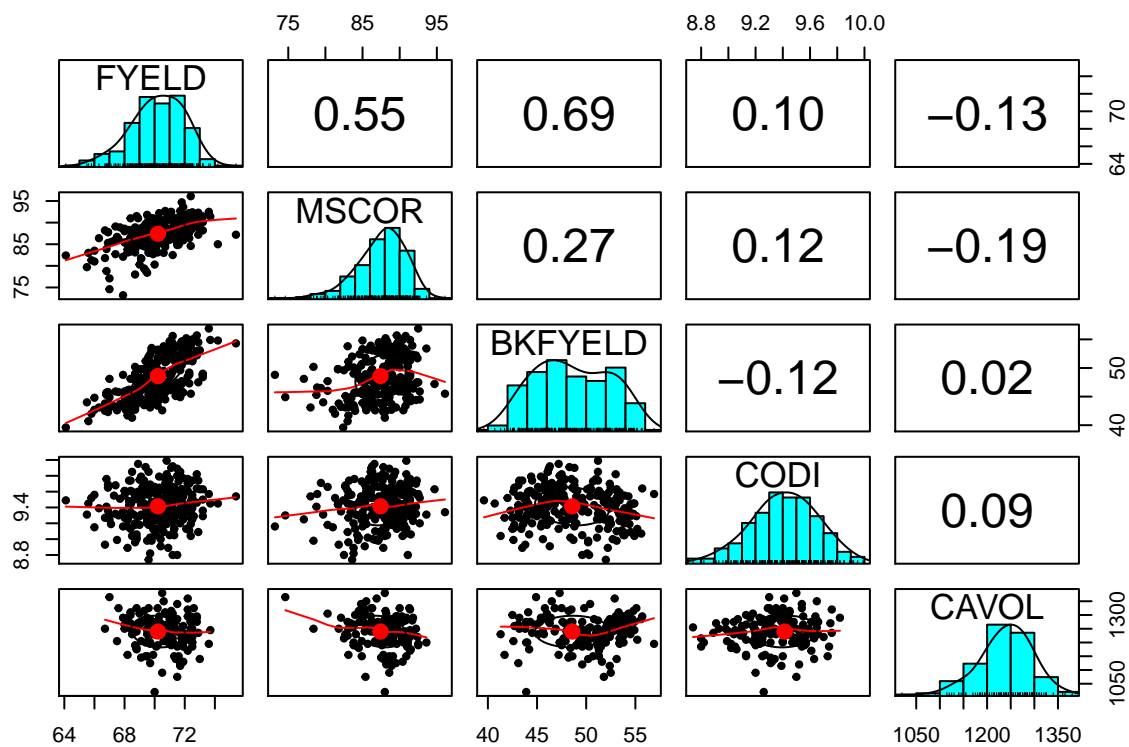
```
pairs.panels(qdata[,31:33])
```



```
pairs.panels(qdata[,32:35])
```



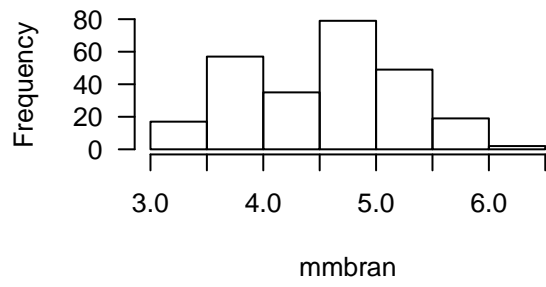
```
pairs.panels(qdata[,31:35])
```

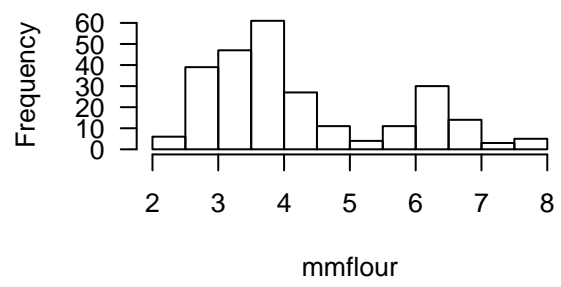
Histograms

```
#Loop function "lapply" to output histograms for every trait
#lapply apply a function to a list and is an easy option that reduces the memory requirement as compared to for loop
par(mfrow=c(2,2))
outs=lapply(names(qdata[,12:35]), function(x) hist(qdata[[x]],data=qdata,main="Histogram of Quality Trait",xlab=x))
```

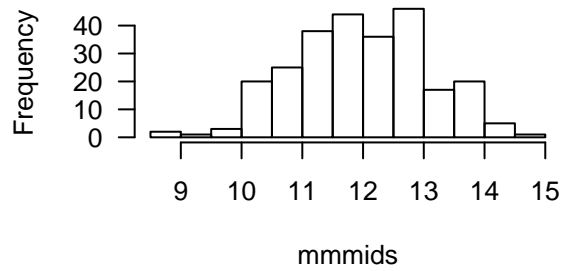
Histogram of Quality Trait



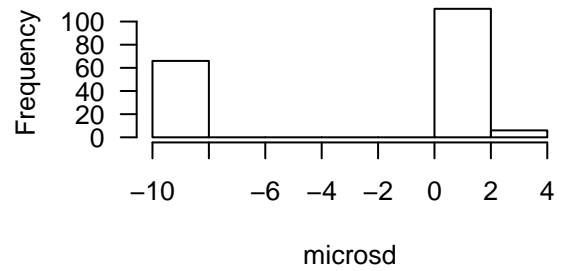
Histogram of Quality Trait



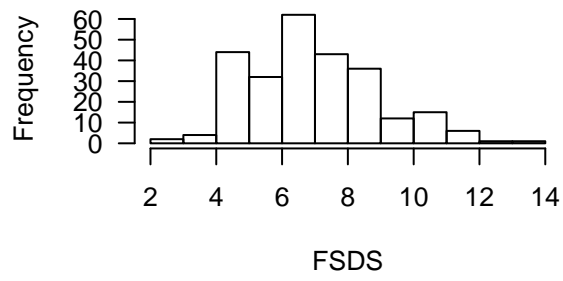
Histogram of Quality Trait



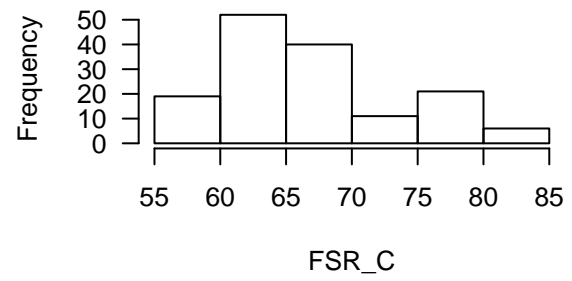
Histogram of Quality Trait



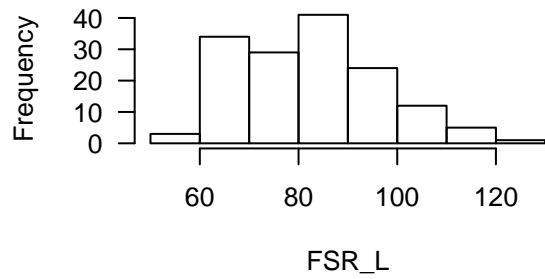
Histogram of Quality Trait



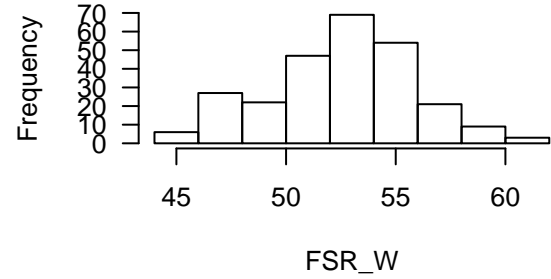
Histogram of Quality Trait



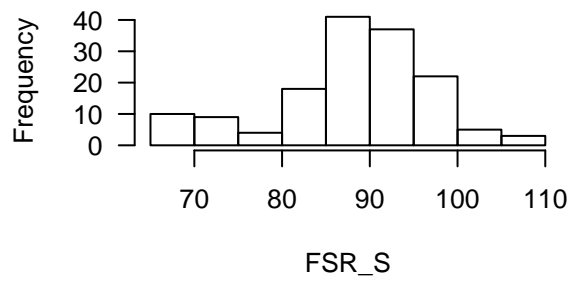
Histogram of Quality Trait



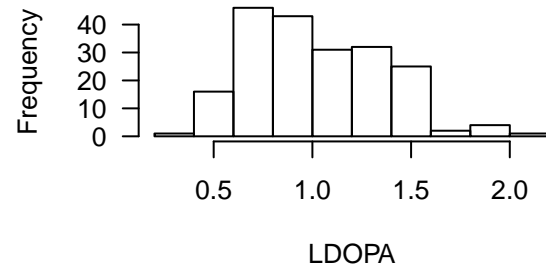
Histogram of Quality Trait



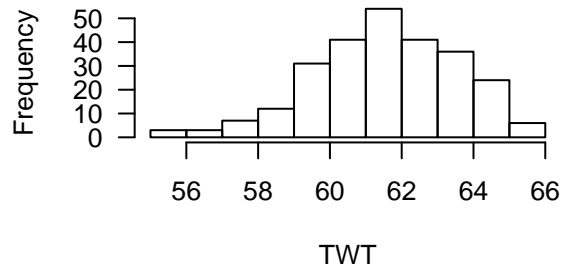
Histogram of Quality Trait



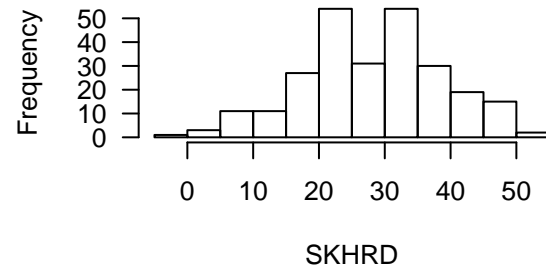
Histogram of Quality Trait



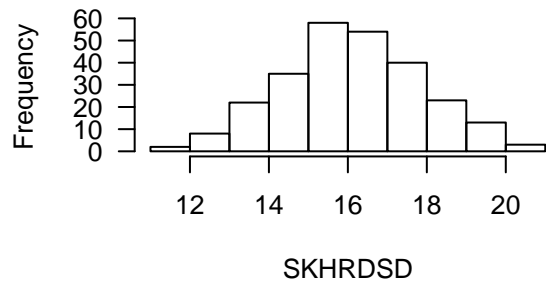
Histogram of Quality Trait



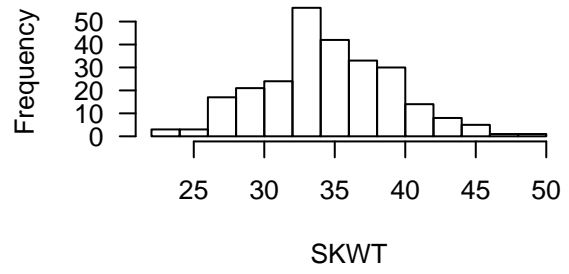
Histogram of Quality Trait



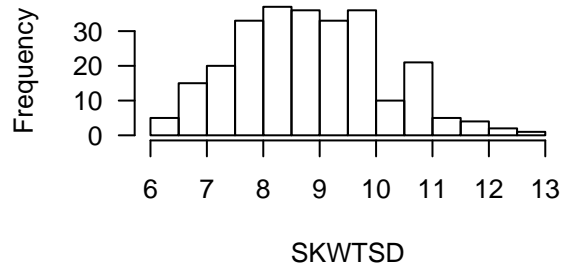
Histogram of Quality Trait



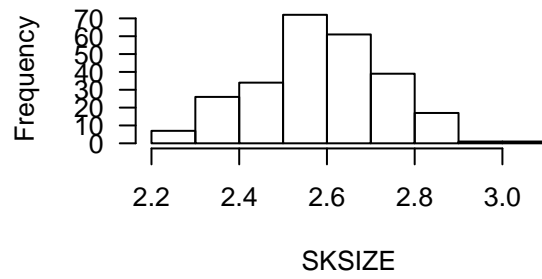
Histogram of Quality Trait



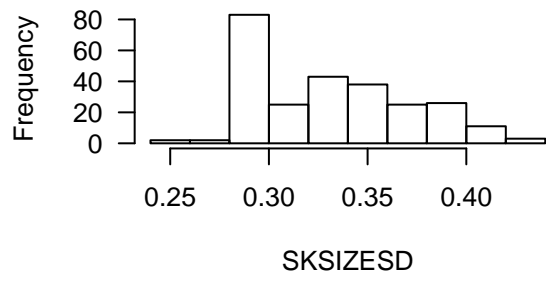
Histogram of Quality Trait



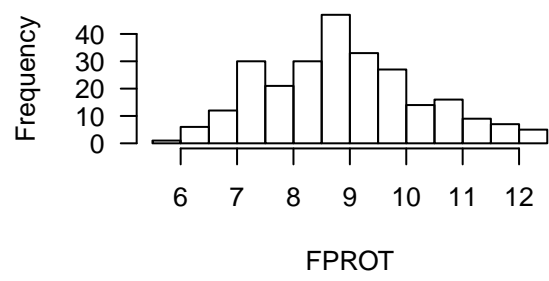
Histogram of Quality Trait



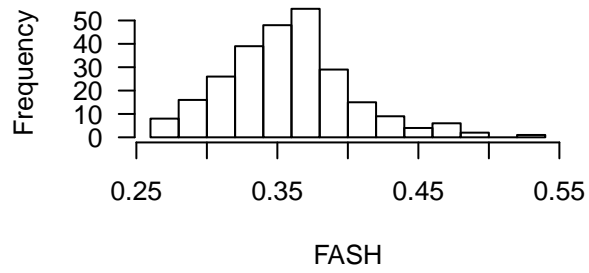
Histogram of Quality Trait



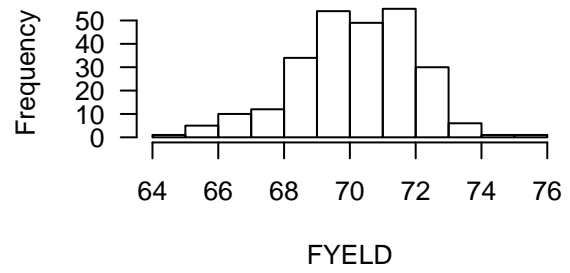
Histogram of Quality Trait

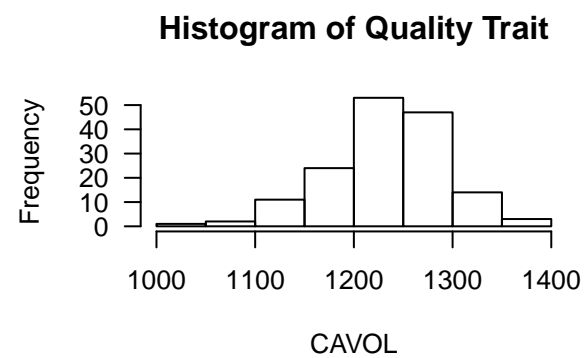
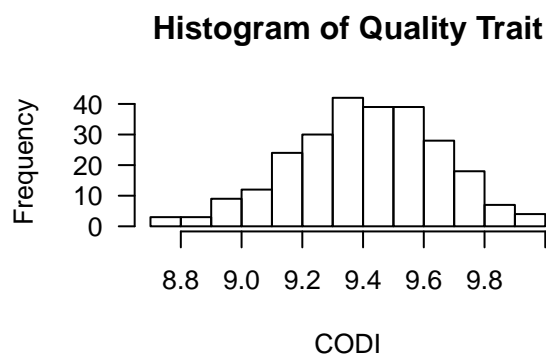
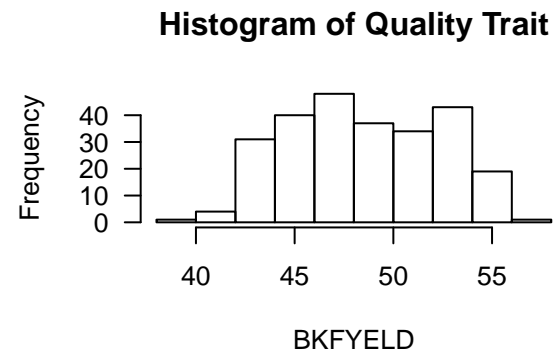
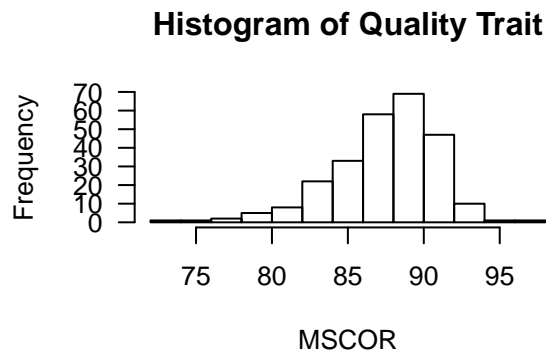


Histogram of Quality Trait



Histogram of Quality Trait



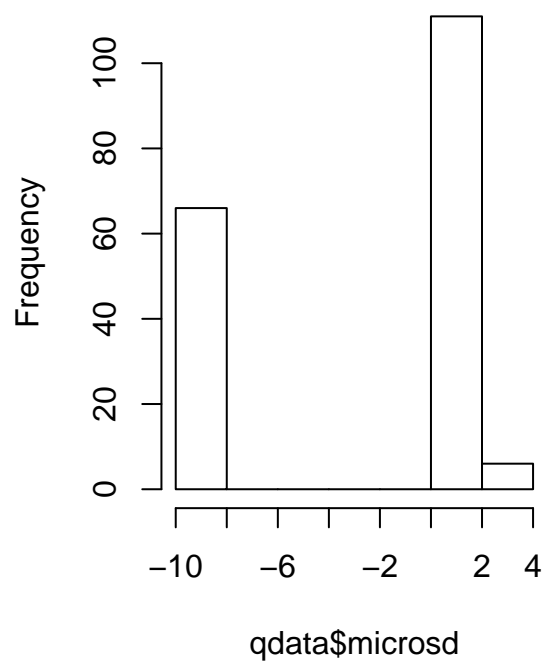


Transformations examples

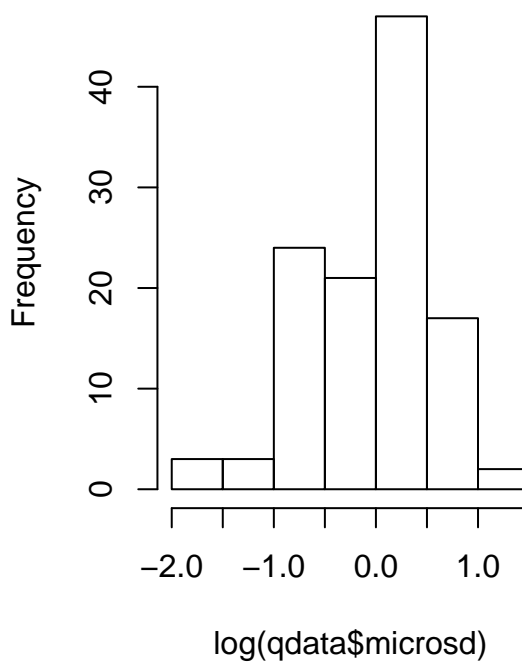
#It is important to explore the distributions of the data you are exploring. The distribution is vital

```
par(mfrow=c(1,2))
hist(qdata$microsd)
hist(log(qdata$microsd))
```

Histogram of qdata\$microsd

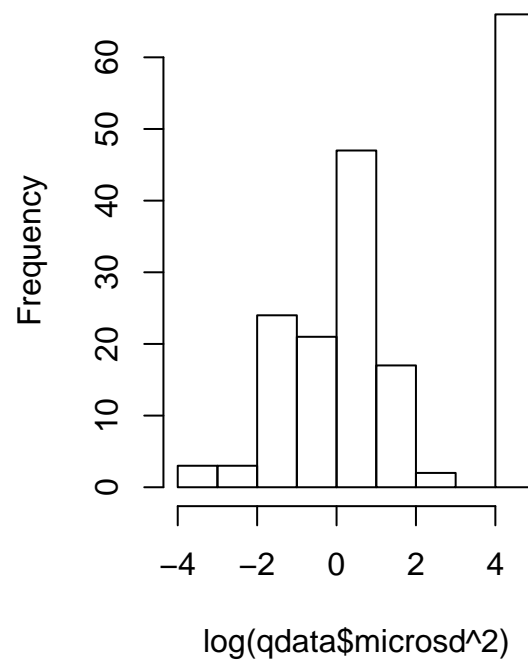
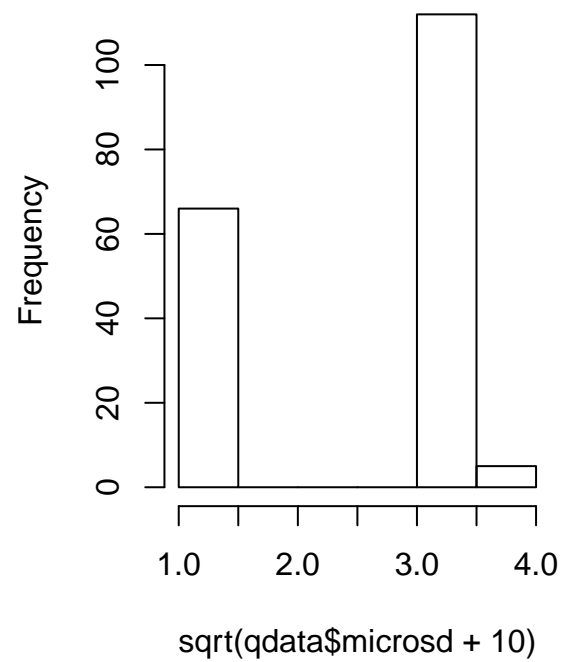


Histogram of log(qdata\$microsd)



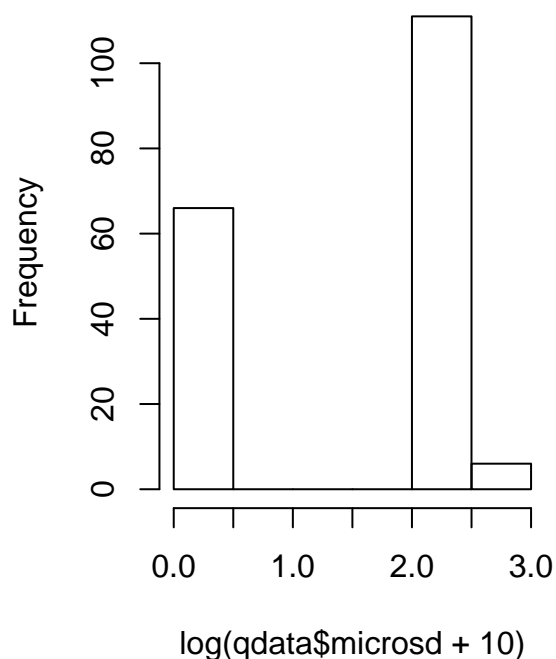
```
hist(sqrt(qdata$microsd+10))  
hist(log(qdata$microsd^2))
```


Histogram of $\sqrt{\text{qdata\$microsd} + 10}$ + Histogram of $\log(\text{qdata\$microsd}^2)$



```
hist(log(qdata$microsd+10))
```

Histogram of log(qdata\$microsd +



Problem is with data

#The problem with this dataset was that the NA's are actually displayed as -9.0. Dr. Campbell replaced
`qdata$microsd`

```
## [1] 2.0 1.4 2.6 1.3 1.9 1.1 1.1 -9.0 0.5 -9.0 0.6 1.4 1.4 -9.0
## [15] 0.2 -9.0 1.2 0.8 1.3 -9.0 0.8 1.2 -9.0 1.4 1.2 2.6 1.9 1.4
## [29] 1.0 1.0 -9.0 1.1 0.2 1.1 -9.0 2.8 1.7 -9.0 -9.0 -9.0 -9.0 -9.0
## [43] -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0
## [57] -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 NA 0.5 NA
## [71] 1.6 2.0 NA NA 0.6 0.6 0.6 0.6 0.6 0.3 0.3 0.3 -9.0 -9.0
## [85] -9.0 NA NA NA NA 0.5 1.0 -9.0 -9.0 NA 0.6 NA 1.6 1.9
## [99] NA NA 1.1 NA NA 1.5 1.1 0.8 1.0 1.7 -9.0 -9.0 1.5 1.3
## [113] -9.0 -9.0 NA NA 1.0 1.0 1.0 1.0 4.0 -9.0 NA NA 1.3 1.3
## [127] 1.3 1.3 1.6 1.5 1.5 NA 1.5 1.7 -9.0 -9.0 NA NA NA NA
## [141] NA NA 1.4 NA NA 1.2 NA NA NA NA 1.5 1.1 -9.0 1.6
## [155] 1.6 1.6 NA NA 1.6 1.2 1.6 1.1 NA NA 1.3 1.7 1.0 NA
## [169] NA NA NA 2.2 1.9 1.0 1.0 2.6 0.5 1.9 NA NA 1.0 NA
## [183] NA 0.6 0.4 NA NA 2.0 NA NA 1.0 1.5 NA NA NA NA
## [197] NA NA 0.6 NA NA NA NA NA 0.6 NA 0.8 1.0 1.4 2.0
## [211] 1.3 NA 1.0 1.0 1.0 1.6 -9.0 -9.0 -9.0 -9.0 -9.0 -9.0 NA -9.0
## [225] NA 0.6 0.6 0.6 0.2 0.5 0.4 0.5 0.5 -9.0 -9.0 -9.0 -9.0 -9.0
## [239] -9.0 -9.0 -9.0 NA NA NA NA NA 1.4 NA NA 0.5 NA NA
## [253] NA 0.6 NA 1.5 NA NA
```

```
qdata <- qdata %>% mutate(microsd = replace(microsd, microsd == -9.0, NA))
qdata$microsd
```

```
## [1] 2.0 1.4 2.6 1.3 1.9 1.1 1.1 NA 0.5 NA 0.6 1.4 1.4 NA 0.2 NA 1.2
## [18] 0.8 1.3 NA 0.8 1.2 NA 1.4 1.2 2.6 1.9 1.4 1.0 1.0 NA 1.1 0.2 1.1
## [35] NA 2.8 1.7 NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [52] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
## [69] 0.5 NA 1.6 2.0 NA NA 0.6 0.6 0.6 0.6 0.6 0.3 0.3 0.3 NA NA NA
## [86] NA NA NA NA 0.5 1.0 NA NA NA 0.6 NA 1.6 1.9 NA NA 1.1 NA
## [103] NA 1.5 1.1 0.8 1.0 1.7 NA NA 1.5 1.3 NA NA NA NA 1.0 1.0 1.0
## [120] 1.0 4.0 NA NA NA 1.3 1.3 1.3 1.3 1.6 1.5 1.5 NA 1.5 1.7 NA NA
## [137] NA NA NA NA NA NA 1.4 NA NA 1.2 NA NA NA NA 1.5 1.1 NA
## [154] 1.6 1.6 1.6 NA NA 1.6 1.2 1.6 1.1 NA NA 1.3 1.7 1.0 NA NA NA
## [171] NA 2.2 1.9 1.0 1.0 2.6 0.5 1.9 NA NA 1.0 NA NA 0.6 0.4 NA NA
## [188] 2.0 NA NA 1.0 1.5 NA NA NA NA NA NA NA 0.6 NA NA NA NA NA
## [205] 0.6 NA 0.8 1.0 1.4 2.0 1.3 NA 1.0 1.0 1.0 1.6 NA NA NA NA NA
## [222] NA NA NA NA 0.6 0.6 0.6 0.2 0.5 0.4 0.5 0.5 NA NA NA NA NA
## [239] NA NA NA NA NA NA NA NA 1.4 NA NA 0.5 NA NA NA 0.6 NA
## [256] 1.5 NA NA
```

```
hist(qdata$microsd)
```



Frequency Tables

```
#I explored the frequencies of the tvarious location year combinations. It revealed that most of the li
tbl <- xtabs(~LOCATION+year0, qdata)
```

```
fable(tbl)
```

```
##          year0 2015 2016 2017 2018
## LOCATION
## FARMINGTON      49    0    0    0
## PENDLETON       0   30   47   53
## PULLMAN         3   14    8   13
## WALLA WALLA     23   18    0    0
```

```
tbl1 <- xtabs(~LOCATION+year1, qdata)
fable(tbl1)
```

```
##          year1 2016 2017 2018 2019
## LOCATION
## FARMINGTON      49    0    0    0
## PENDLETON       0   30   47   53
## PULLMAN         3   14    8   13
## WALLA WALLA     23   18    0    0
```

```
tbl2 <- xtabs(~NURNAME+LOCATION, qdata)
fable(tbl2)
```

```
##          LOCATION FARMINGTON PENDLETON PULLMAN WALLA WALLA
## NURNAME
## 2ND CHANCE PRELIMINA              0          0          3          0
## 2ND.CHANCE.PRELIMINA              0          5          2          0
## CARA/XERPHA DOUBLE H              0          0          0         23
## CLUB WHEAT SAMPLES T              0          0          6          0
## RHT1 UNREPLICATED                 0          0          3          0
## RHT2 UNREPLICATED                 0          0          5          0
## SOFT ELITE                        49         125        10         18
## UNREPLICATED                      0          0          7          0
## UNREPLICATED.SOFT                 0          0          2          0
```

```
tbl3 <- xtabs(~NURNAME+year0, qdata)
fable(tbl3)
```

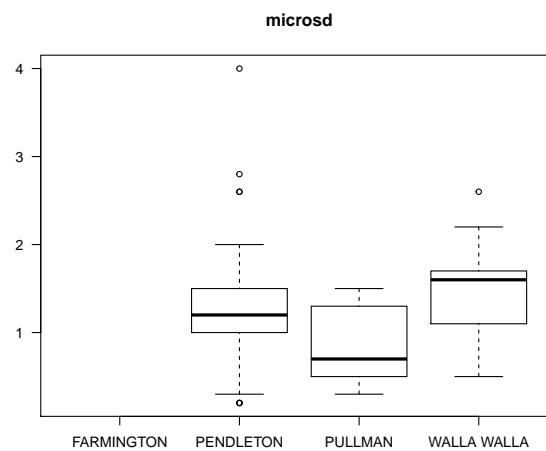
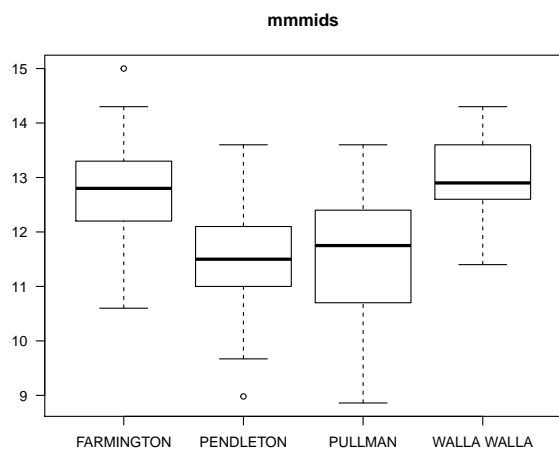
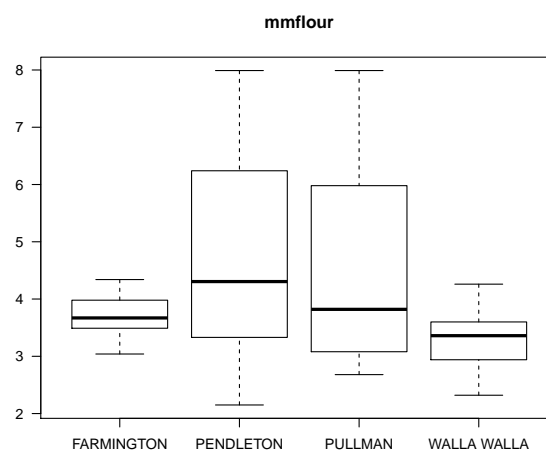
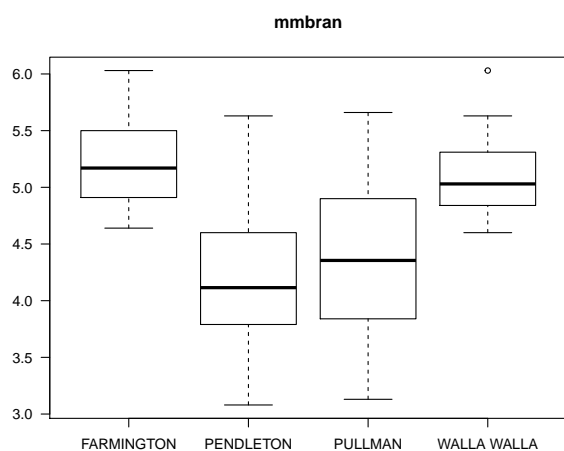
```
##          year0 2015 2016 2017 2018
## NURNAME
## 2ND CHANCE PRELIMINA              3    0    0    0
## 2ND.CHANCE.PRELIMINA              0    7    0    0
## CARA/XERPHA DOUBLE H             23    0    0    0
## CLUB WHEAT SAMPLES T              0    0    0    6
## RHT1 UNREPLICATED                 0    0    3    0
## RHT2 UNREPLICATED                 0    0    5    0
## SOFT ELITE                        49   53   47   53
## UNREPLICATED                      0    0    0    7
## UNREPLICATED.SOFT                 0    2    0    0
```

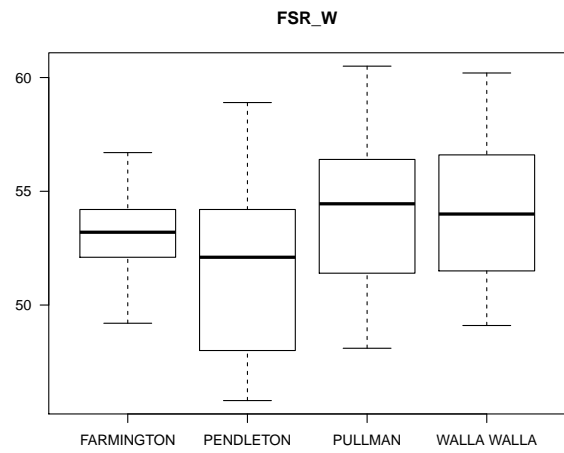
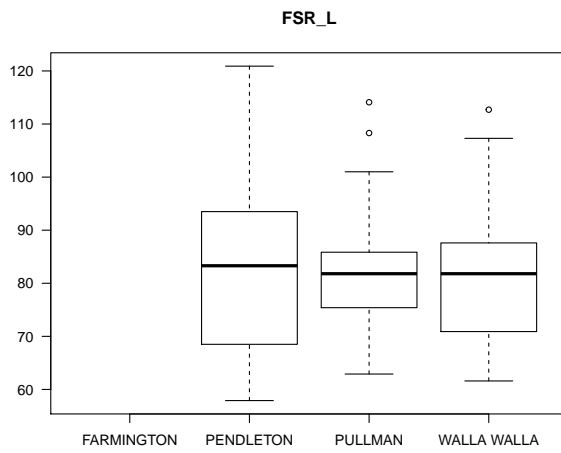
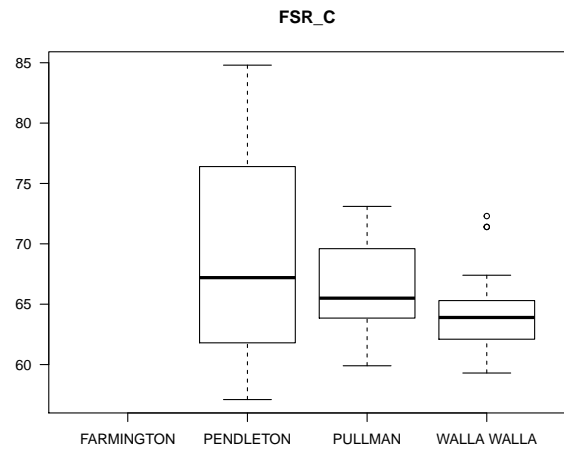
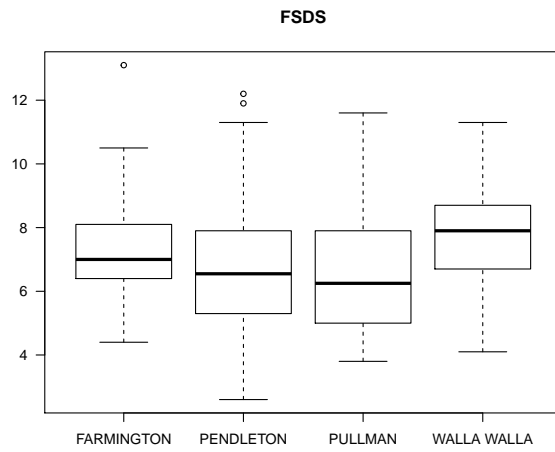
Boxplots

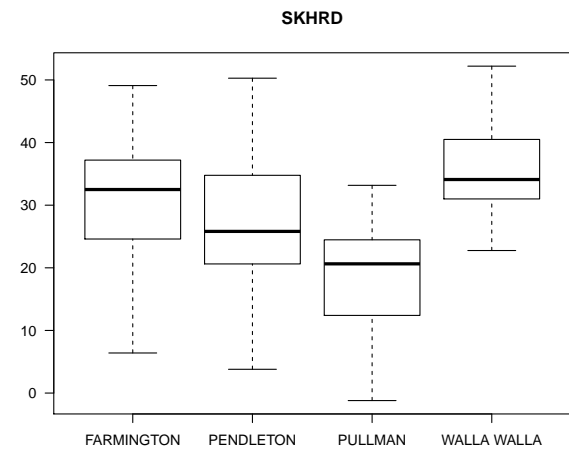
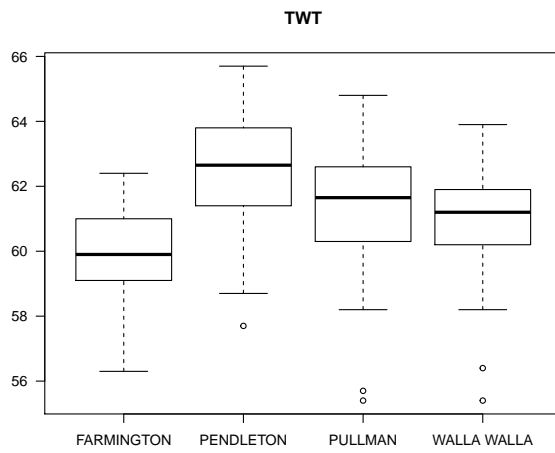
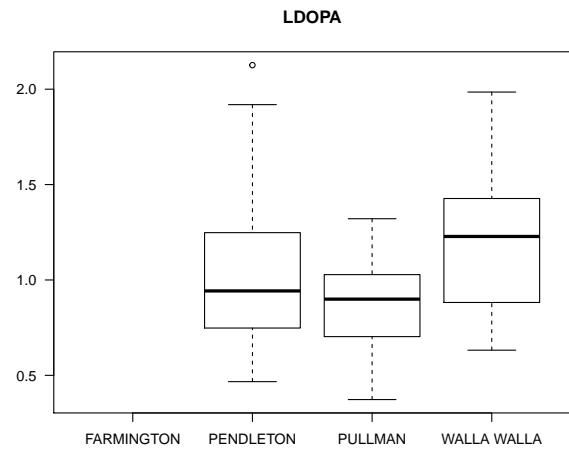
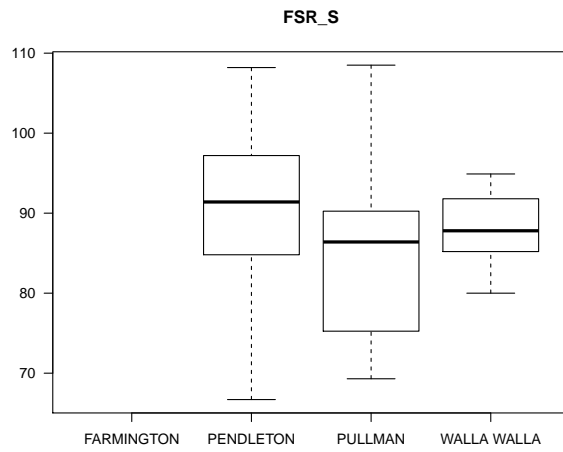
Distribution for traits in different locations

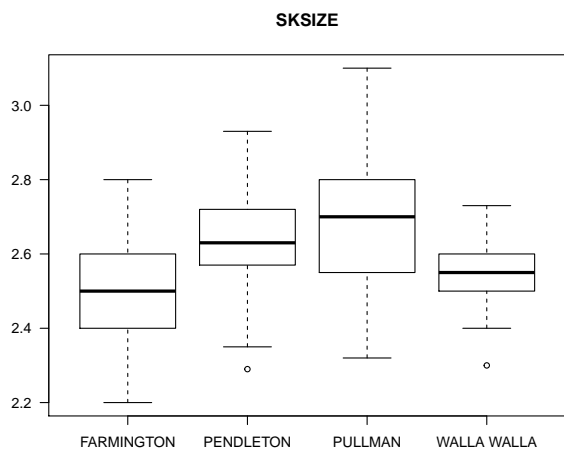
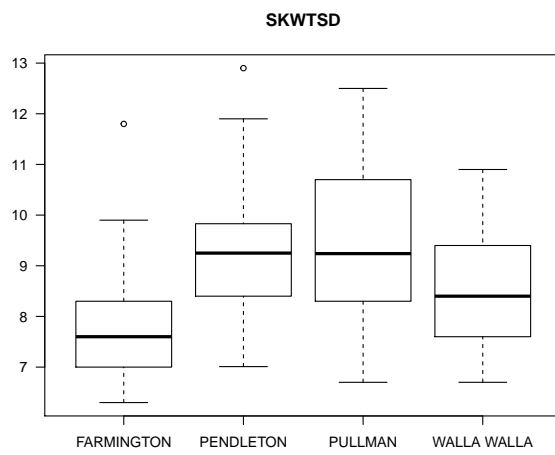
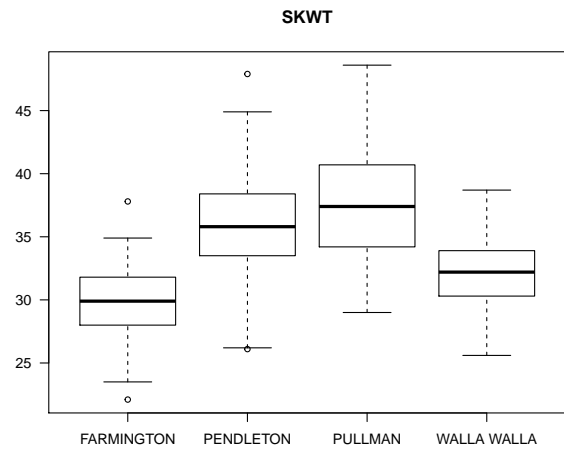
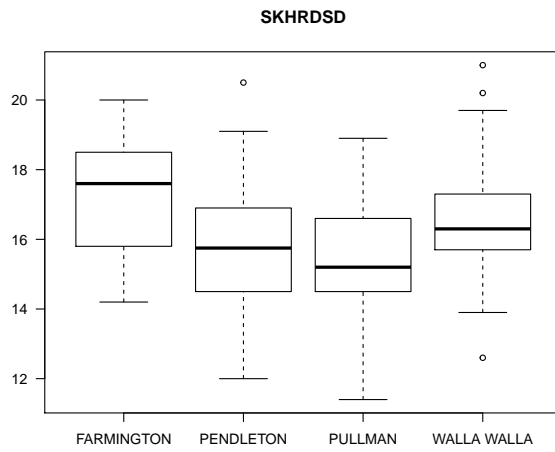
```
#I did the same thing here as I did for the histograms, but had it divide out per location
par(mfrow=c(2,2))
```

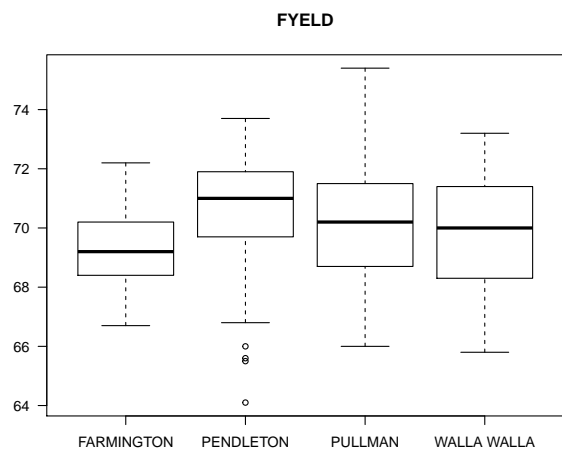
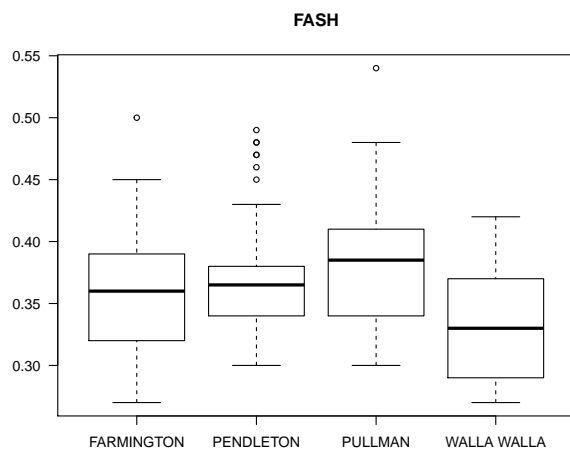
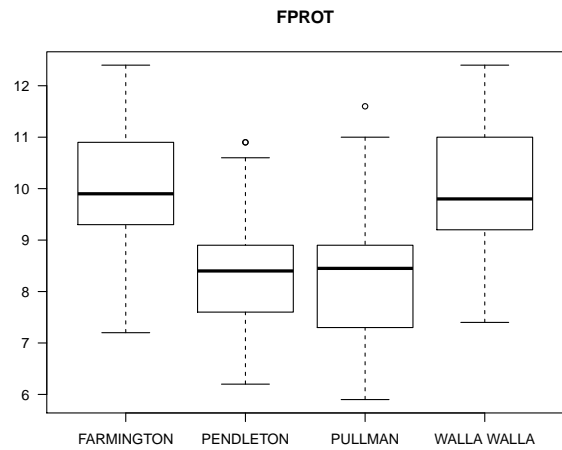
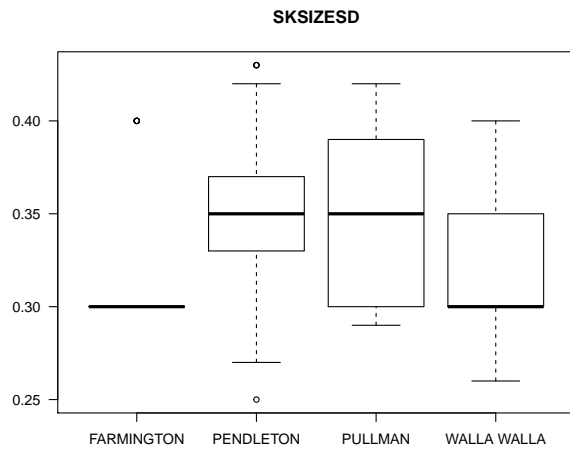
```
#par(mar = c(2, 2, 2, 2) + 0.1)
outs=lapply(names(qdata[,12:35]), function(x) boxplot(formula(paste0(x,"~LOCATION")),data=qdata,main=as
```

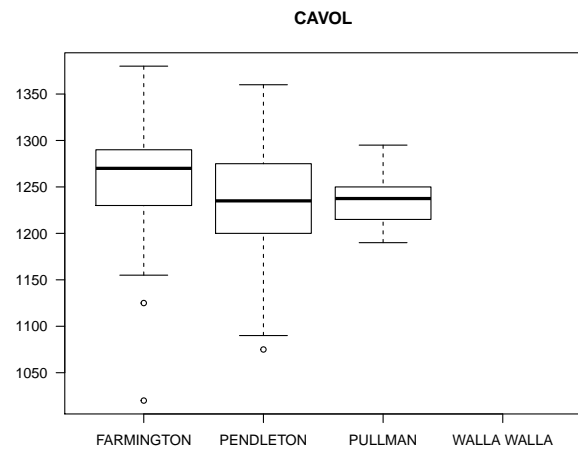
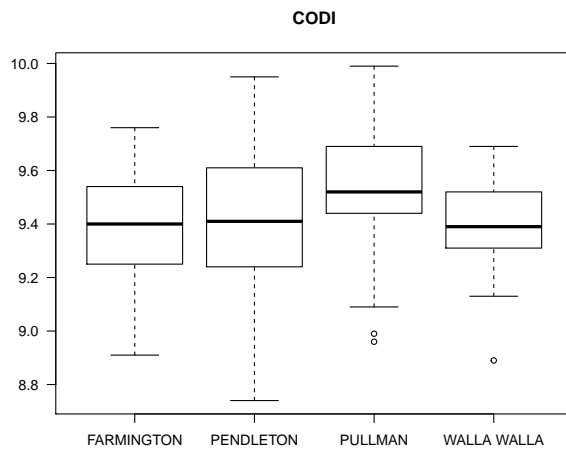
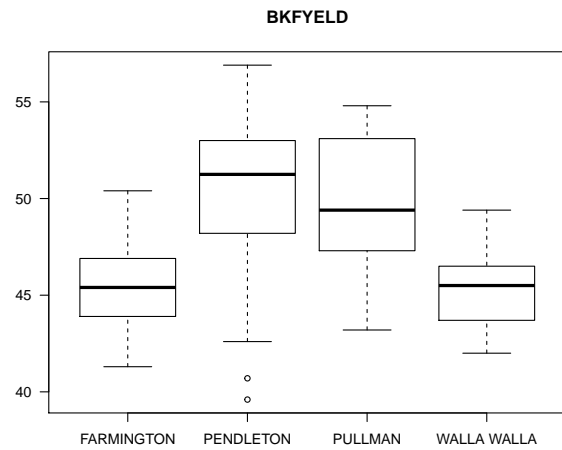
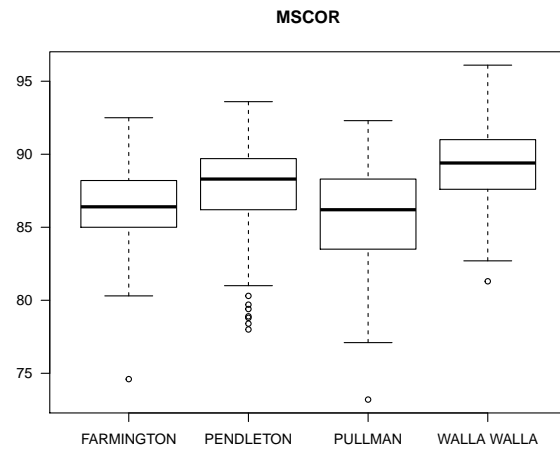










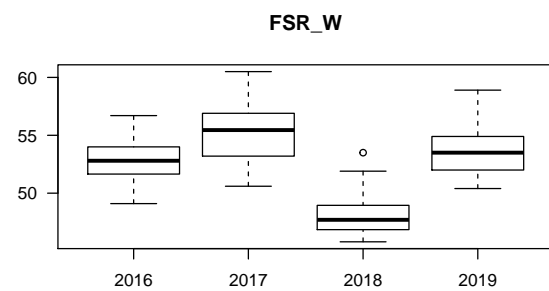
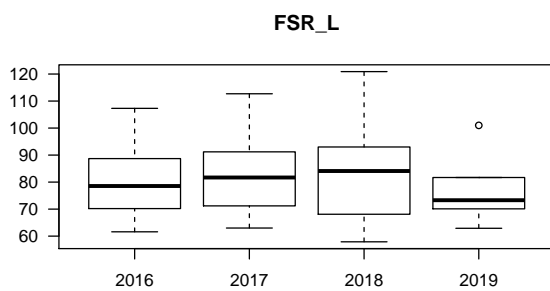
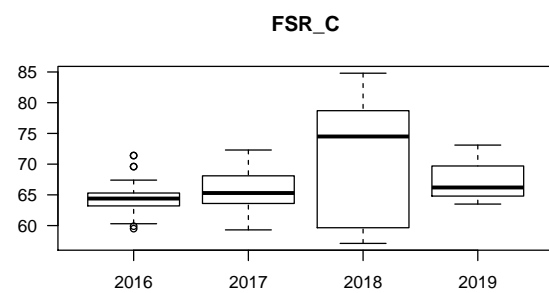
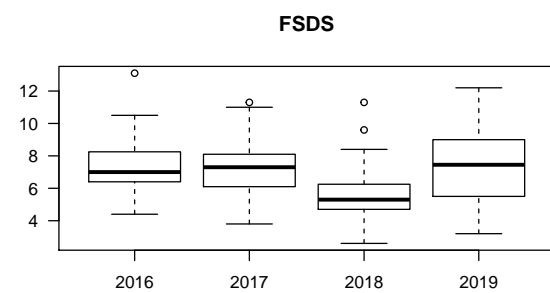
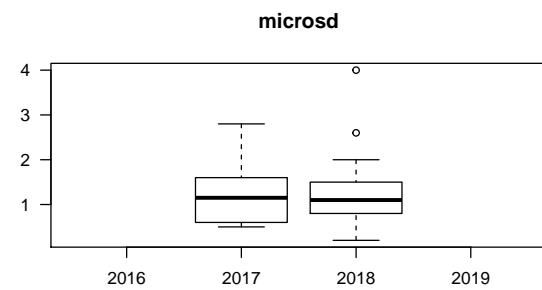
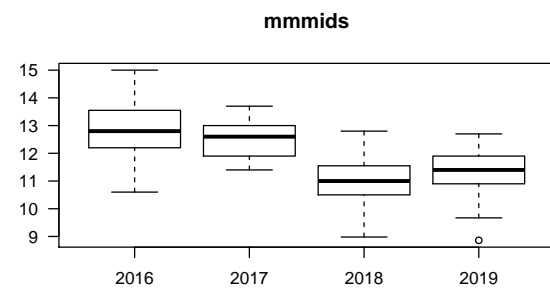
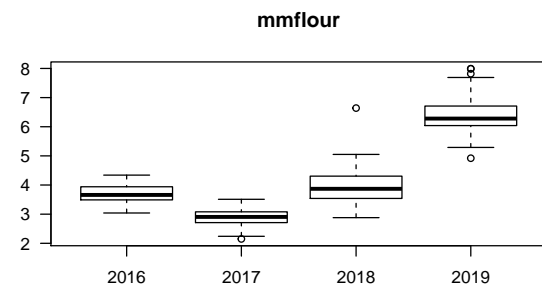
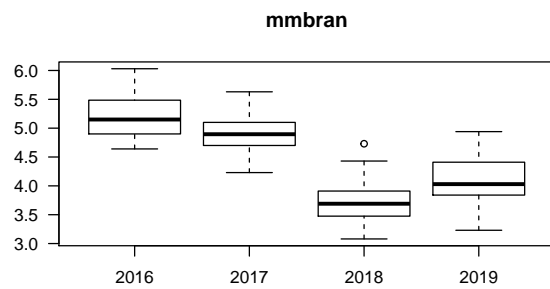


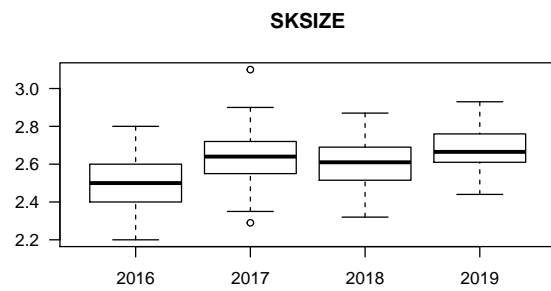
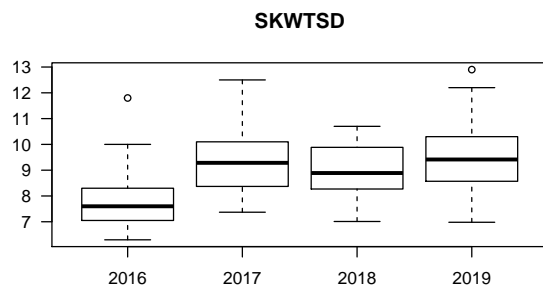
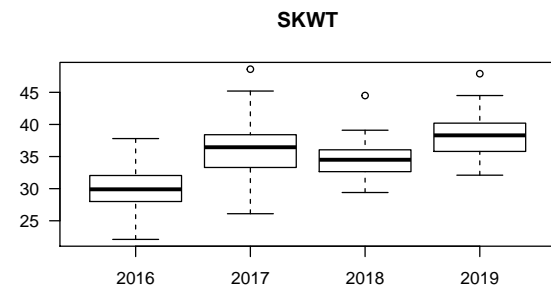
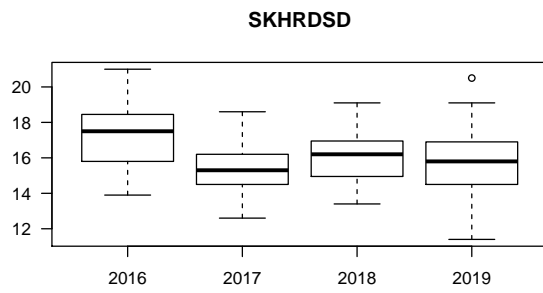
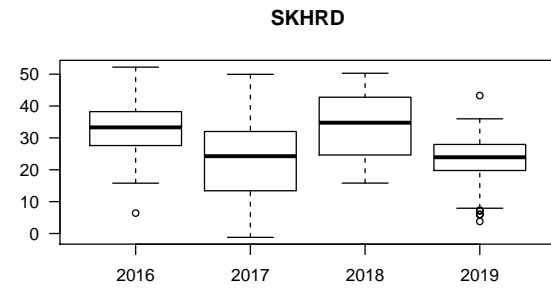
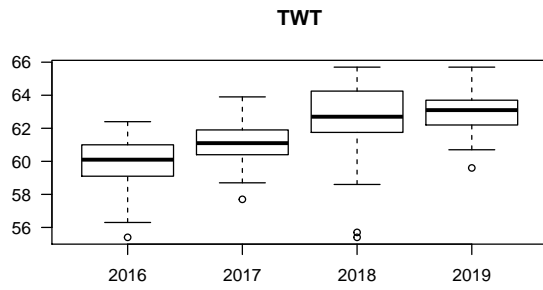
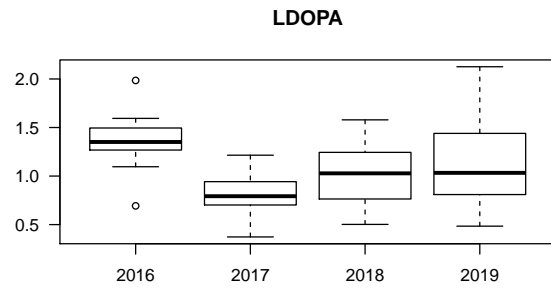
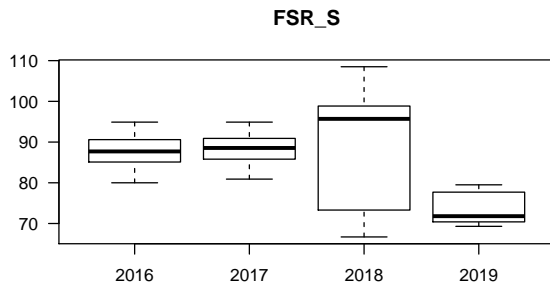
Distribution for traits in different years

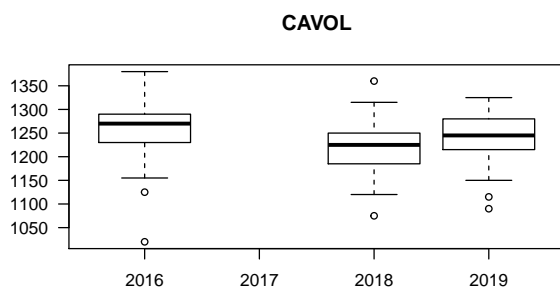
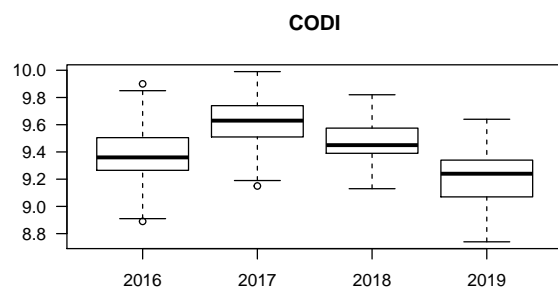
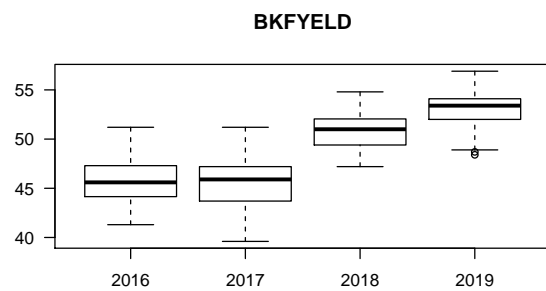
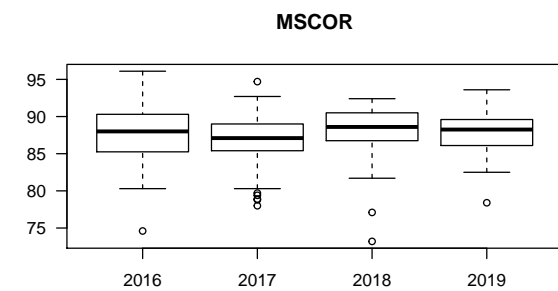
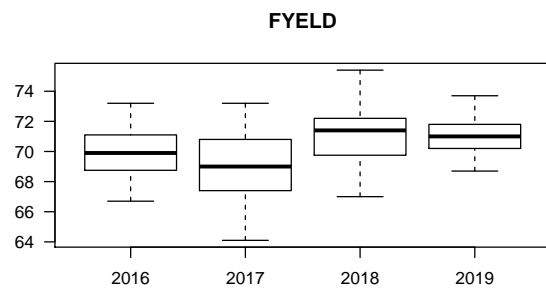
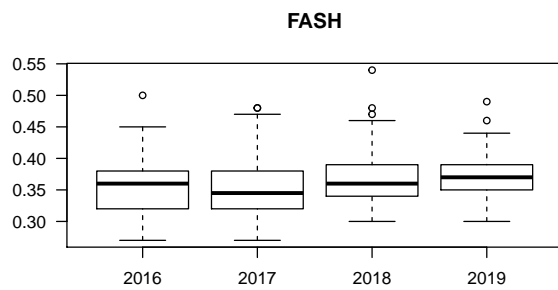
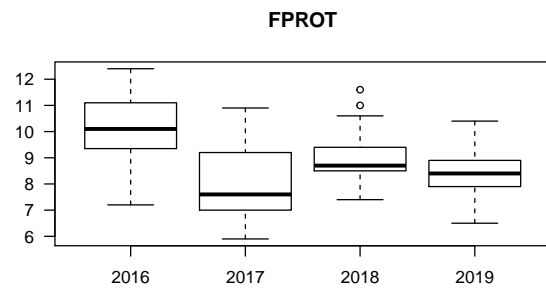
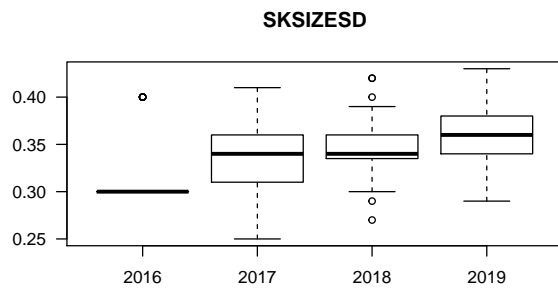
#This is the same as above but just for years.

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

```
outs=lapply(names(qdata[,12:35]), function(x) boxplot(formula(paste0(x,"~year1")),data=qdata,main=as.ch
```







Statistical Analyses and Interpretation

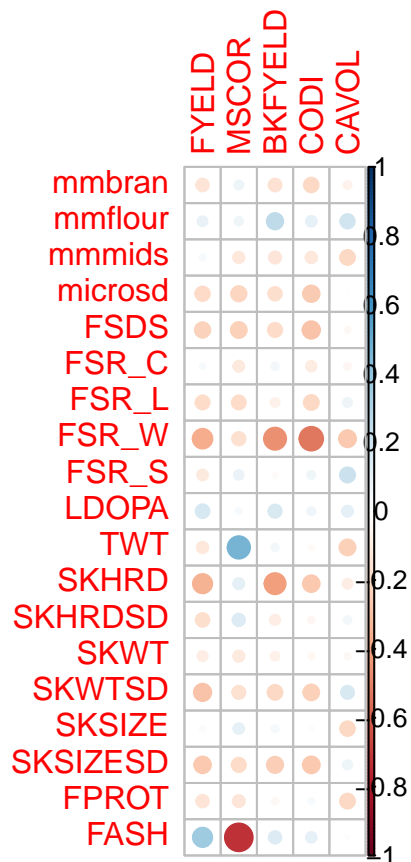
What are the correlations among the traits?

Corrplot and correlations

```
#Overall  
#This correlation function can not run with NAs and therefore, "na.or.complete" must be specified.  
qdoverall=cor(qdata[,12:35],use="na.or.complete")  
#I subsetted the correlation output to focus on the early vs late trait correlations  
qdoverall[1:19,20:24]
```

##	FYELD	MSCOR	BKFYELD	CODI	CAVOL
## mmbran	-0.14906432	0.07587264	-0.15472639	-0.20016759	-0.061871970
## mmflour	0.09235415	0.06749649	0.25158722	0.10885541	0.194784881
## mmmids	0.03788323	-0.12261905	-0.13898319	-0.12471533	-0.207523754
## microsd	-0.19548556	-0.21874229	-0.17010970	-0.25565618	-0.004454738
## FSDS	-0.22328439	-0.23853010	-0.18958059	-0.28397751	-0.033597130
## FSR_C	0.03317348	-0.11471225	0.05277805	-0.10226216	-0.046386839
## FSR_L	-0.18601233	-0.18156843	-0.07579306	-0.20935883	0.078569708
## FSR_W	-0.36374918	-0.16834182	-0.45826044	-0.52481854	-0.269822077
## FSR_S	-0.11238702	0.08772750	-0.02149174	0.06050568	0.214189375
## LDOPA	0.17753333	0.03702356	0.16516590	0.06274665	0.114220811
## TWT	-0.12284480	0.46949118	0.05026827	-0.02479151	-0.238847679
## SKHRD	-0.34774637	0.11644127	-0.41578712	-0.26103418	-0.091572106
## SKHRDSD	-0.17654281	0.14507154	-0.09935174	-0.04291933	0.057540242
## SKWT	-0.09594006	-0.11984415	-0.07289176	-0.03487383	-0.035024454
## SKWTSD	-0.28105572	-0.16093171	-0.20541028	-0.23392482	0.160438270
## SKSIZE	0.02892247	0.10892582	0.04712609	-0.01722734	-0.200739667
## SKSIZESD	-0.26042113	-0.18949411	-0.24656185	-0.26672147	0.079879208
## FPROT	-0.14838805	-0.13673002	-0.03715635	0.03878310	-0.207793832
## FASH	0.35068470	-0.72126792	0.14636891	0.10683401	0.011078367

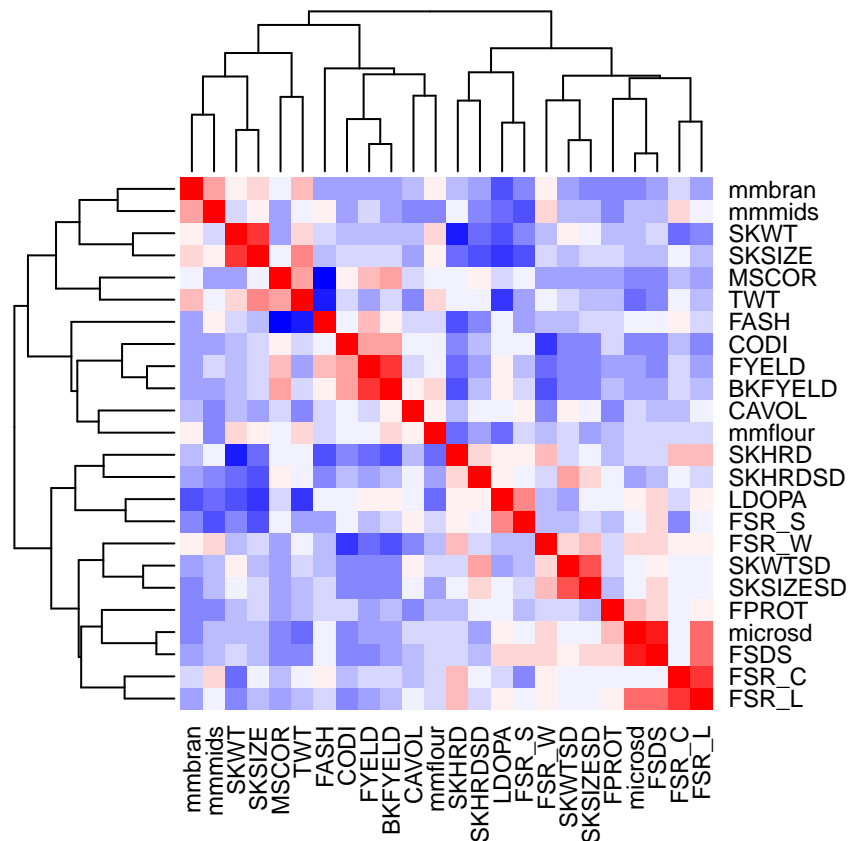
```
corrplot(qdooverall[1:19,20:24])
```



Correlations primarily between the early generation to later generation

Heatmap for correlations

```
#tThis is a heat map that also overlays a hierarchical clustering tree
col<- colorRampPalette(c("blue", "white", "red"))(20)
heatmap(x = qdoverall, col = col, symm = TRUE)
```



Subsetting for Soft Elites in Pendleton

#I subsetting the pendleton location for just the elite soft trial to take a different look at the model.

```
summary(qdata$NURNAME)
```

```
## 2ND CHANCE PRELIMINA 2ND.CHANCE.PRELIMINA CARA/XERPHA DOUBLE H
##                3                7                23
## CLUB WHEAT SAMPLES T      RHT1 UNREPLICATED      RHT2 UNREPLICATED
##                6                3                5
##          SOFT ELITE          UNREPLICATED      UNREPLICATED.SOFT
##                202                7                2
```

```
summary(qdata$LOCATION)
```

```
## FARMINGTON  PENDLETON  PULLMAN WALLA WALLA
##          49          130          38          41
```

qdse<-subset(qdata,qdata\$NURNAME==levels(qdata\$NURNAME)[7]) #This is subsetting for the 7th level which

```
summary(qdse$NURNAME)
```

```
## 2ND CHANCE PRELIMINA 2ND.CHANCE.PRELIMINA CARA/XERPHA DOUBLE H
##                0                0                0
## CLUB WHEAT SAMPLES T      RHT1 UNREPLICATED      RHT2 UNREPLICATED
##                0                0                0
##          SOFT ELITE          UNREPLICATED      UNREPLICATED.SOFT
##                202                0                0
```



```
summary(qdse$LOCATION)
```

```
## FARMINGTON    PENDLETON    PULLMAN WALLA WALLA
##           49           125           10           18
```

```
tbl1 <- xtabs(~LOCATION+year0, qdse)
ftable(tbl1)
```

```
##           year0 2015 2016 2017 2018
## LOCATION
## FARMINGTON           49    0    0    0
## PENDLETON            0   25   47   53
## PULLMAN              0   10    0    0
## WALLA WALLA          0   18    0    0
```

```
qdsep<-subset(qdse,qdse$LOCATION==levels(qdse$LOCATION)[2]) #This further subsets the data to just the
qdsep$NURSCO=as.factor(qdsep$NURSCO)
qdsep$SAMPLE=as.factor(qdsep$SAMPLE)
qdsep$entry=as.factor(qdsep$entry)
qdsep$year1=as.factor(qdsep$year1)
qdsep$year0=as.factor(qdsep$year0)
```

Extra validation will be done for the lines present in the Soft elite lines in Pendleton due to the vast majority of lines being present in this location and nursery.

What are the heritabilities?

$$p(X) = \frac{V_g}{V_g + \frac{V_{g \times year}}{N_{year}} + \frac{V_{g \times loc}}{N_{loc}} + \frac{V_e}{(N_{loc})(N_{year})}}$$

Different version of mixed models will be used to determine best model for BLUPs and most accurate heritabilities.

Lme4

Mixed Model for Line, Location, Year and with no interactions.

```
#I used the same lapply function to create a list of models
#this model takes the variables for line, location, and year with no interaction effects
models4 <- lapply(names(qdata[,12:35]), function(x) lmer(formula(paste0(x, "~(1|NAMET)*(1|LOCATION)*(1|",
names(models4)<-names(qdata[,12:35])
```

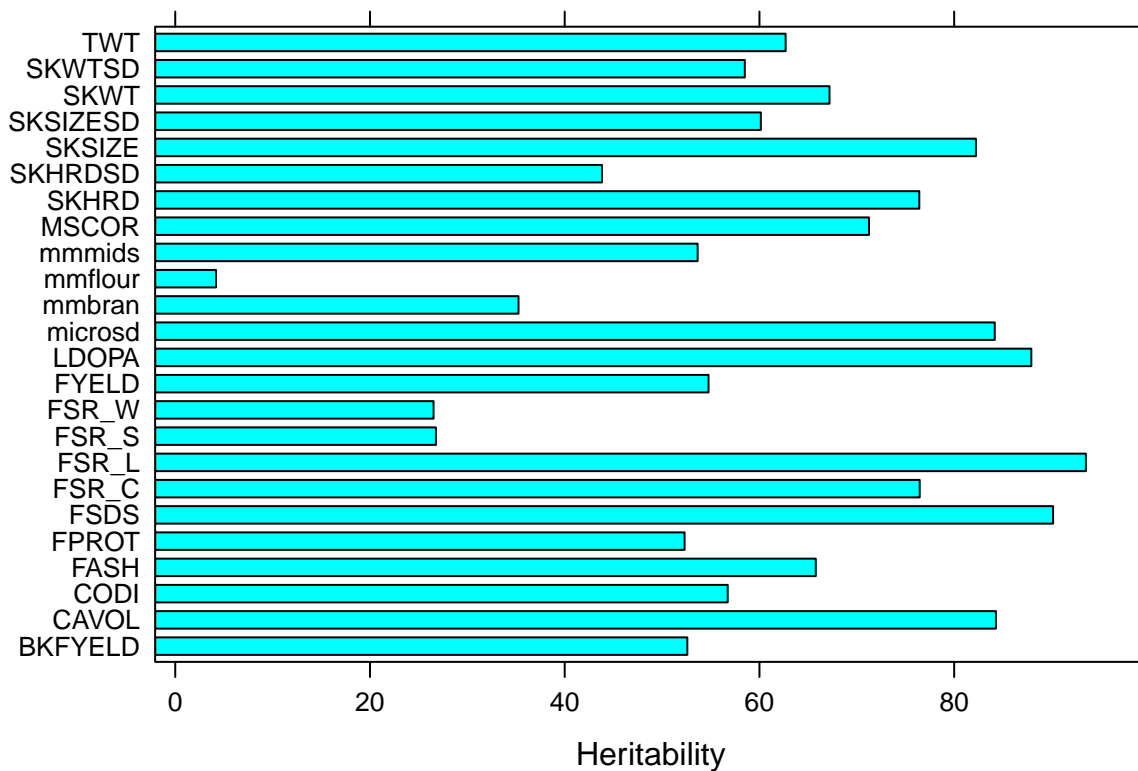
```
#I wrote a loop using the VarCorr function to subset the variance components from each model
#This heritability using the variance components for line, location, and year and divides the year, loc
hert1=c()
```

```
for(i in 1:length(models4)){
  x=models4[[i]]
  x=(as.data.frame(VarCorr(x))[c(1,4)][1,2]/ #it goes through and creates a dataframe for the name and
  (as.data.frame(VarCorr(x))[c(1,4)][1,2]+
    (as.data.frame(VarCorr(x))[c(1,4)][2,2]/4)+
    (as.data.frame(VarCorr(x))[c(1,4)][3,2]/4)+
    (as.data.frame(VarCorr(x))[c(1,4)][4,2]/16)))*100
  x=as.numeric(x)
  hert1=c(hert1,x)
}
```

```

hert1=cbind(names(models4),round(hert1,5))
hert1=as.data.frame(hert1)
names(hert1)<-c("Trait","Heritability")
hert1$Heritability=as.numeric(as.character(hert1$Heritability))
barchart(Trait~Heritability,data=hert1)

```



hert1

```

##      Trait Heritability
## 1  mmbran    35.25424
## 2  mmflour     4.18815
## 3  mmmids    53.65209
## 4  microsd    84.16729
## 5    FSDS    90.15717
## 6   FSR_C    76.45715
## 7   FSR_L    93.52582
## 8   FSR_W    26.53077
## 9   FSR_S    26.77419
## 10  LDOPA    87.92276
## 11   TWT    62.68798
## 12  SKHRD    76.41711
## 13 SKHRDSD    43.82516
## 14   SKWT    67.19506
## 15  SKWTSD    58.49594
## 16  SKSIZE    82.23043

```

```
## 17 SKSIZESD      60.14340
## 18   FPROT       52.30967
## 19    FASH       65.78485
## 20   FYELD       54.76967
## 21   MSCOR       71.24918
## 22  BKFYELD      52.59335
## 23    CODI       56.73987
## 24   CAVOL       84.28723
```

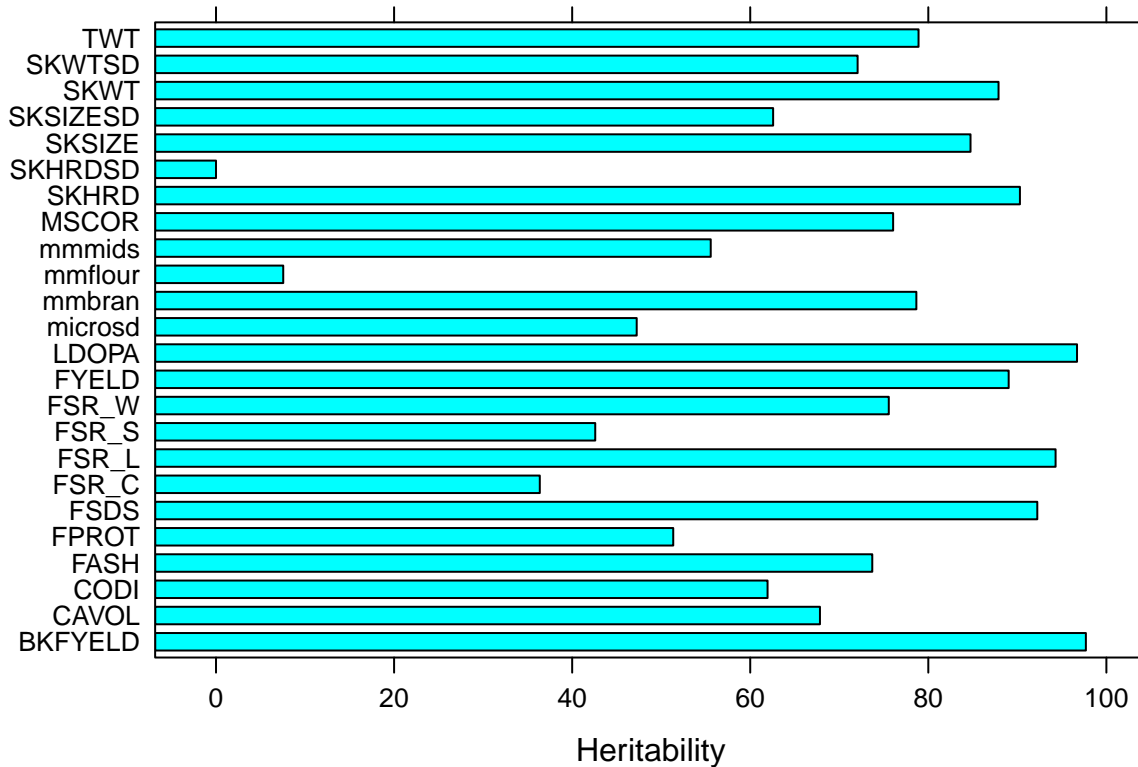
Mixed Model for Line, Location, Year and interactions.

Mean-line basis version of heritability

```
library(lme4)
#This model is the same as above, but includes the interaction effects in order to calculate the mean-l
models41 <- lapply(names(qdata[,12:35]), function(x) lmer(formula(paste0(x,
  "~(1|NAMET)+(1|LOCATION)+(1|year0)+(1|NAMET:LOCATION)+(1|NAMET:year0)")), data=qdata))
names(models41)<-names(qdata[,12:35])
an<-list()
for(i in 1:length(models41)){
  x=models41[[i]]
  y=models4[[i]]
  z=as.data.frame(anova(x,y))
  name=names(models41[i])
  an[[name]]=z
}

#This is the same thing as above, but each interaction effect is used instead of the explicit year and
hert11=c()
for(i in 1:length(models41)){
  x=models41[[i]]
  x=(as.data.frame(VarCorr(x))[,c(1,4)][3,2]/
    (as.data.frame(VarCorr(x))[,c(1,4)][3,2]+
      (as.data.frame(VarCorr(x))[,c(1,4)][1,2]/4)+
      (as.data.frame(VarCorr(x))[,c(1,4)][2,2]/4)+
      (as.data.frame(VarCorr(x))[,c(1,4)][6,2]/16))))*100
  x=as.numeric(x)
  hert11=c(hert11,x)
}

hert11=cbind(names(models41),round(hert11,5))
hert11=as.data.frame(hert11)
names(hert11)<-c("Trait", "Heritability")
hert11$Heritability=as.numeric(as.character(hert11$Heritability))
barchart(Trait~Heritability,data=hert11)
```



hert11

##	Trait	Heritability
## 1	mmbran	78.65814
## 2	mmflour	7.54919
## 3	mmmids	55.56241
## 4	microsd	47.24858
## 5	FSDS	92.24167
## 6	FSR_C	36.36493
## 7	FSR_L	94.29544
## 8	FSR_W	75.55939
## 9	FSR_S	42.59276
## 10	LDOPA	96.70367
## 11	TWT	78.90226
## 12	SKHRD	90.29002
## 13	SKHRDSD	0.00000
## 14	SKWT	87.88917
## 15	SKWTSD	72.06066
## 16	SKSIZE	84.74199
## 17	SKSIZESD	62.57208
## 18	FPROT	51.35112
## 19	FASH	73.70655
## 20	FYELD	89.02749
## 21	MSCOR	76.04960
## 22	BKFYELD	97.69701
## 23	CODI	61.94515
## 24	CAVOL	67.83120

Environment Variable

Mixed Model for Line, Location, Year and interactions but the location and year is combined into an environment variable

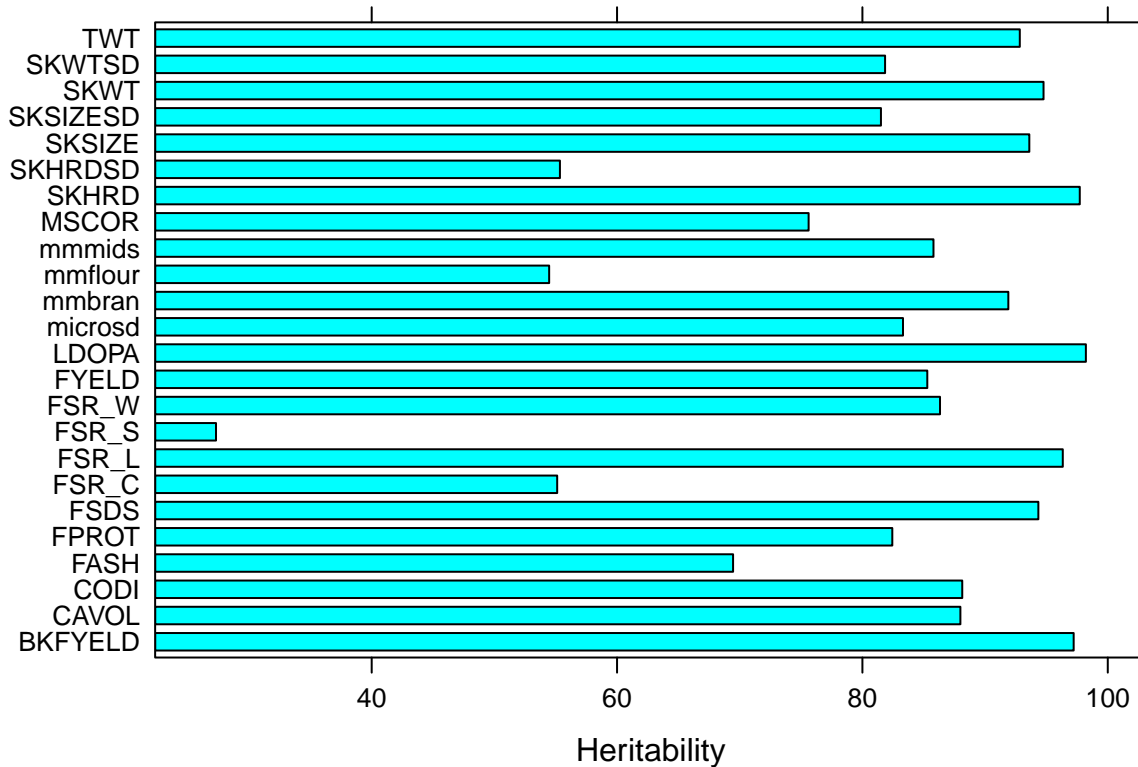
#This is the same as previous models but the location and year variables are combined into a specific environment variable

```
qdata$ENV2=paste(qdata$year0,qdata$LOCATION,sep=":")
qdata=transform(qdata, ENV2=factor(ENV2))
models44 <- lapply(names(qdata[,12:35]), function(x) lmer(formula(paste0(x,
  "~(1|NAMET)+(1|ENV2)+(1|NAMET:ENV2)")), data=qdata))
names(models44)<-names(qdata[,12:35])

an4<-list()
for(i in 1:length(models44)){
  x=models44[[i]]
  y=models44[[i]]
  z=as.data.frame(anova(x,y))
  name=names(models44[i])
  an4[[name]]=z
}

hert14=c()
for(i in 1:length(models44)){
  x=models44[[i]]
  x=(as.data.frame(VarCorr(x))[,c(1,4)][2,2]/
    (as.data.frame(VarCorr(x))[,c(1,4)][2,2]+
      (as.data.frame(VarCorr(x))[,c(1,4)][1,2]/10)+
      (as.data.frame(VarCorr(x))[,c(1,4)][4,2]/10)))*100
  x=as.numeric(x)
  hert14=c(hert14,x)
}

hert14=cbind(names(models44),round(hert14,5))
hert14=as.data.frame(hert14)
names(hert14)<-c("Trait","Heritability")
hert14$Heritability=as.numeric(as.character(hert14$Heritability))
barchart(Trait~Heritability,data=hert14)
```

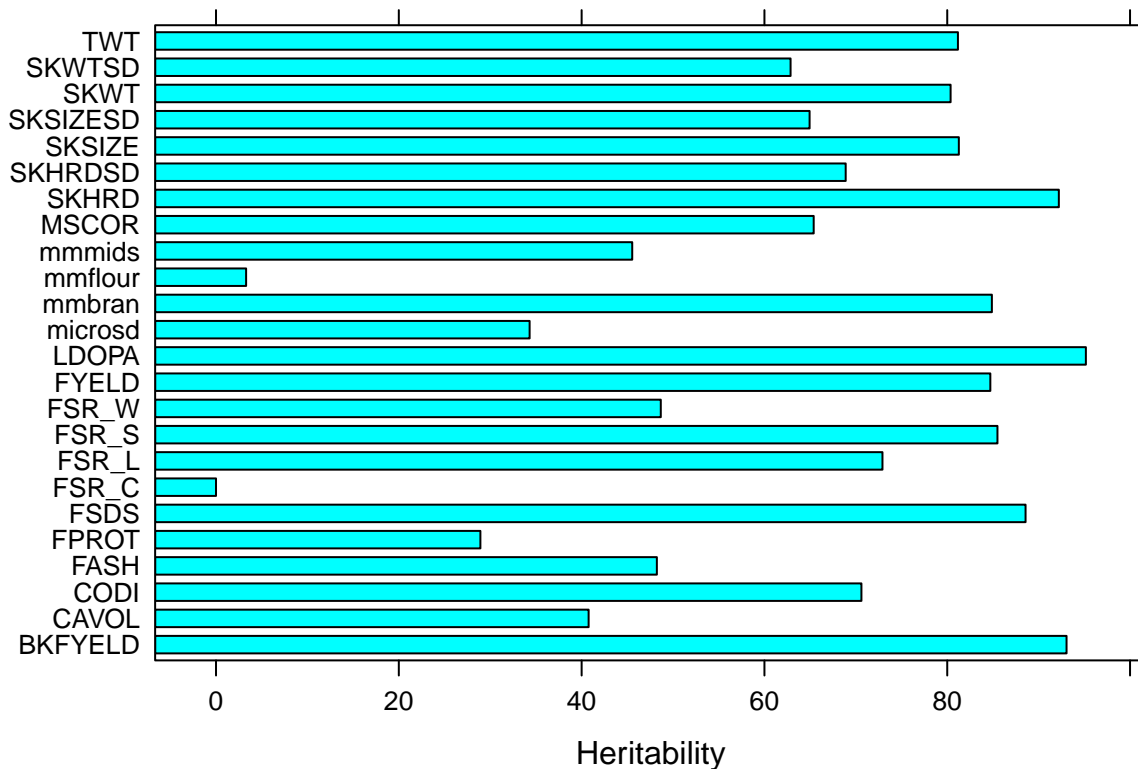


hert14

##	Trait	Heritability
## 1	mmbran	91.89373
## 2	mmflour	54.46640
## 3	mmmids	85.79259
## 4	microsd	83.30816
## 5	FSDS	94.34107
## 6	FSR_C	55.12064
## 7	FSR_L	96.33178
## 8	FSR_W	86.32406
## 9	FSR_S	27.31125
## 10	LDOPA	98.21429
## 11	TWT	92.82840
## 12	SKHRD	97.71688
## 13	SKHRDSD	55.34420
## 14	SKWT	94.76382
## 15	SKWTSD	81.84729
## 16	SKSIZE	93.60504
## 17	SKSIZESD	81.50991
## 18	FPROT	82.43652
## 19	FASH	69.45777
## 20	FYELD	85.28851
## 21	MSCOR	75.61500
## 22	BKFYELD	97.21998
## 23	CODI	88.13691
## 24	CAVOL	87.98431

Mean-line basis (Soft Elite and Pendleton Subset)

```
#This is the mean-line basis model but for the subsetting soft elite pendleton lines
qdsep=droplevels(qdsep)
models45 <- lapply(names(qdsep[,12:35]), function(x) lmer(formula(paste0(x,
  "~(1|NAMET)+(1|year0)")), data=qdsep))
names(models45)<-names(qdsep[,12:35])
hert15=c()
for(i in 1:length(models45)){
  x=models45[[i]]
  x=(as.data.frame(VarCorr(x))[,c(1,4)][1,2]/
    (as.data.frame(VarCorr(x))[,c(1,4)][1,2]+
      (as.data.frame(VarCorr(x))[,c(1,4)][3,2]/3)))*100
  x=as.numeric(x)
  hert15=c(hert15,x)
}
hert15=cbind(names(models45),round(hert15,5))
hert15=as.data.frame(hert15)
names(hert15)<-c("Trait","Heritability")
hert15$Heritability=as.numeric(as.character(hert15$Heritability))
barchart(Trait~Heritability,data=hert15)
```



hert15

```
##      Trait Heritability
## 1   mmbran    84.88377
## 2   mmflour     3.29170
```

```
## 3    mmmids      45.53447
## 4    microsd     34.30985
## 5      FSDS      88.56850
## 6      FSR_C      0.00000
## 7      FSR_L      72.91284
## 8      FSR_W      48.66058
## 9      FSR_S      85.49383
## 10    LDOPA      95.16737
## 11      TWT      81.17048
## 12    SKHRD      92.20976
## 13    SKHRDSD     68.88948
## 14      SKWT      80.36765
## 15    SKWTSD     62.85864
## 16    SKSIZE      81.26888
## 17    SKSIZESD     64.94047
## 18    FPROT      28.92192
## 19      FASH      48.22954
## 20    FYELD      84.71424
## 21    MSCOR      65.38999
## 22    BKFYELD     93.05314
## 23      CODI      70.60985
## 24    CAVOL      40.76587
```

Comparison of Heritability

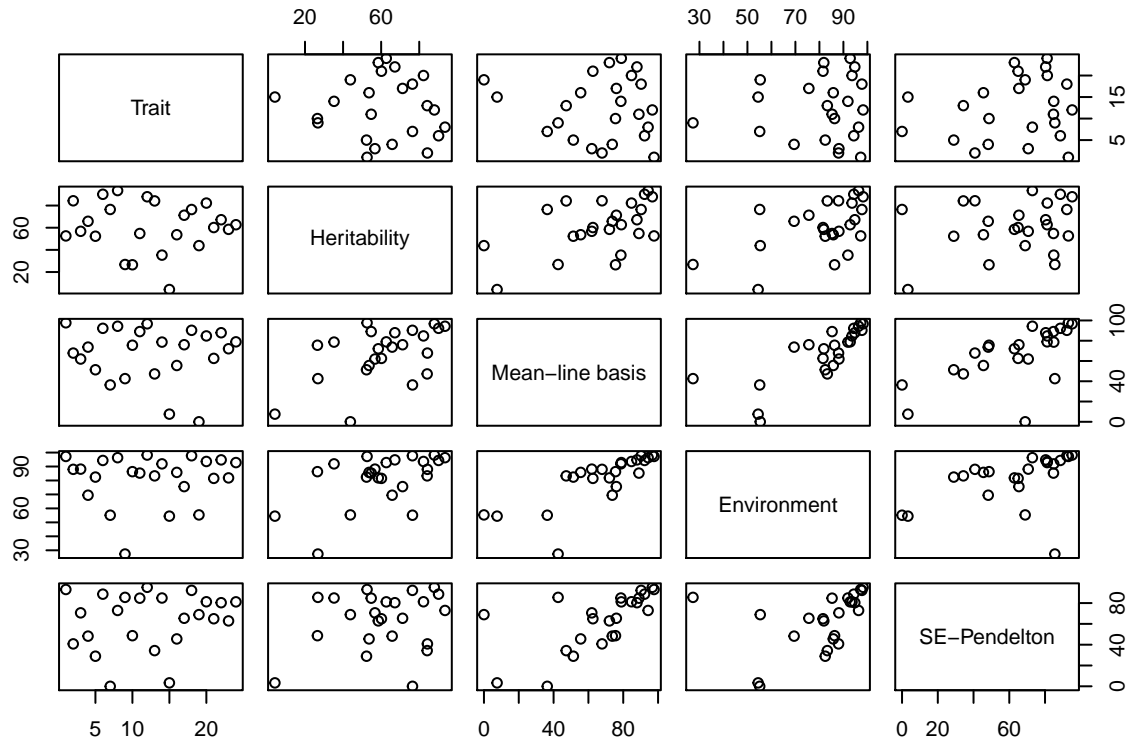
```
#this creates a dataset that subsets the values from the previous calculations
hert2=cbind(hert1,hert11[,2],hert14[,2],hert15[,2])
names(hert2)<-c("Trait","Heritability","Mean-line basis","Environment","SE-Pendelton")
hert2
```

##	Trait	Heritability	Mean-line basis	Environment	SE-Pendelton
## 1	mmbran	35.25424	78.65814	91.89373	84.88377
## 2	mmflour	4.18815	7.54919	54.46640	3.29170
## 3	mmmids	53.65209	55.56241	85.79259	45.53447
## 4	microsd	84.16729	47.24858	83.30816	34.30985
## 5	FSDS	90.15717	92.24167	94.34107	88.56850
## 6	FSR_C	76.45715	36.36493	55.12064	0.00000
## 7	FSR_L	93.52582	94.29544	96.33178	72.91284
## 8	FSR_W	26.53077	75.55939	86.32406	48.66058
## 9	FSR_S	26.77419	42.59276	27.31125	85.49383
## 10	LDOPA	87.92276	96.70367	98.21429	95.16737
## 11	TWT	62.68798	78.90226	92.82840	81.17048
## 12	SKHRD	76.41711	90.29002	97.71688	92.20976
## 13	SKHRDSD	43.82516	0.00000	55.34420	68.88948
## 14	SKWT	67.19506	87.88917	94.76382	80.36765
## 15	SKWTSD	58.49594	72.06066	81.84729	62.85864
## 16	SKSIZE	82.23043	84.74199	93.60504	81.26888
## 17	SKSIZESD	60.14340	62.57208	81.50991	64.94047
## 18	FPROT	52.30967	51.35112	82.43652	28.92192
## 19	FASH	65.78485	73.70655	69.45777	48.22954
## 20	FYELD	54.76967	89.02749	85.28851	84.71424
## 21	MSCOR	71.24918	76.04960	75.61500	65.38999
## 22	BKFYELD	52.59335	97.69701	97.21998	93.05314
## 23	CODI	56.73987	61.94515	88.13691	70.60985


```
## 24      CAVOL      84.28723      67.83120      87.98431      40.76587
```

Plot of Heritability

```
plot(hert2)
```



Comparison of MSE

A good way to compare models is to calculate the mean squared error of the residuals. These MSEs were run on Test weight for model selection because test weight was normally distributed and had no missing data.

#Mean-square error is a good way to measure the error for models, the smaller it is the better fit the model

```
MSE1 <- mean((predict(models4$TWT, qdata,type = "response") - qdata$TWT)^2)
MSE2 <- mean((predict(models41$TWT, qdata,type = "response") - qdata$TWT)^2)
MSE5 <- mean((predict(models44$TWT, qdata,type = "response") - qdata$TWT)^2)

name=c("No GE","GE-0","ENV-0")
dat_mse=cbind(name,rbind(MSE1,MSE2,MSE5))
colnames(dat_mse)<-c("Model","MSE")
dat_mse
```

```
##      Model  MSE
## MSE1 "No GE" "0.893796485791088"
## MSE2 "GE-0"  "0.28307285047641"
## MSE5 "ENV-0" "0.104706415207486"
```

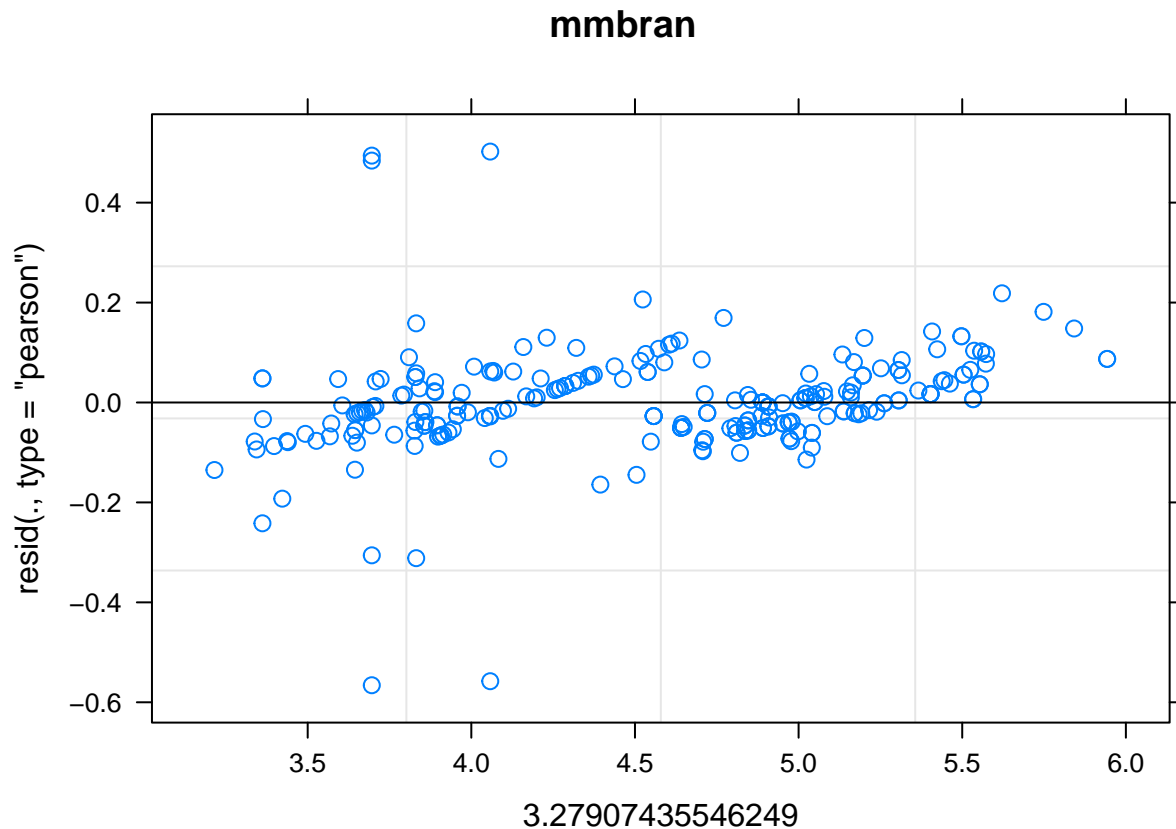
Best model was the Mean-line basis version with interactions

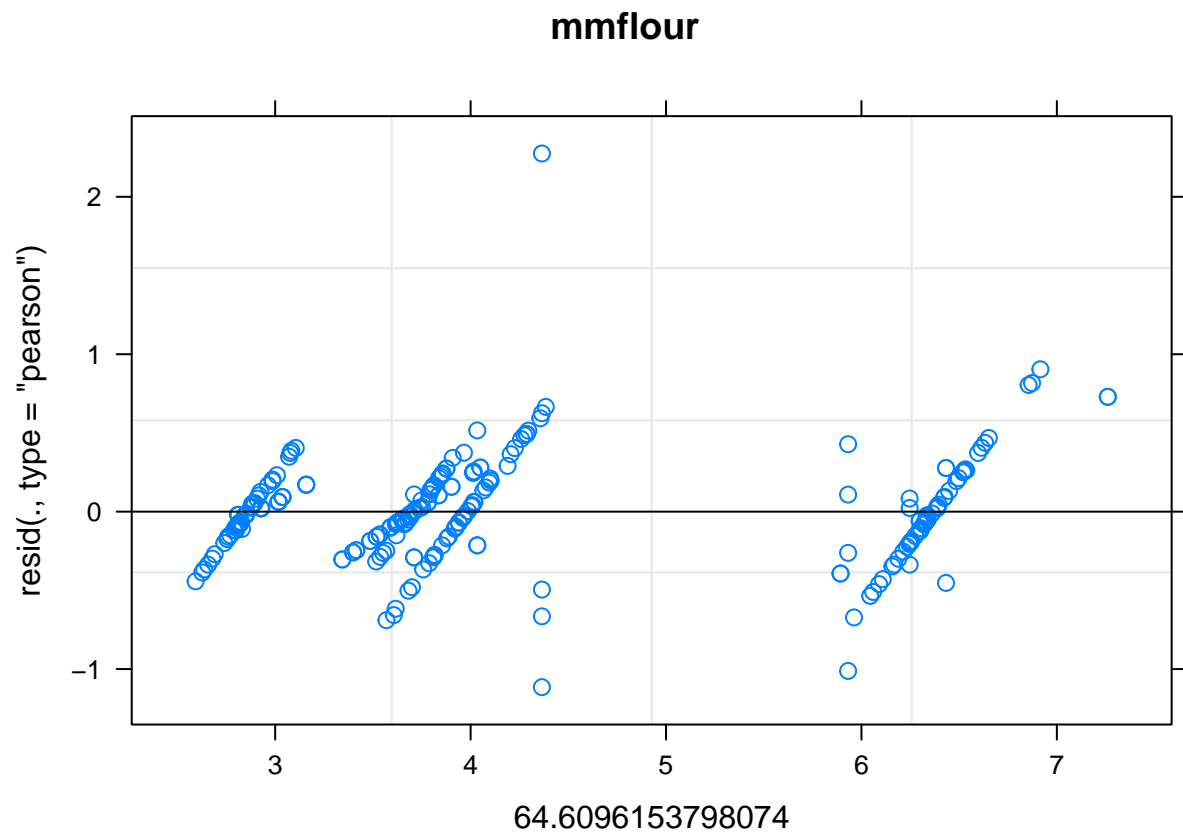
Plot Residuals and Mean Square Error

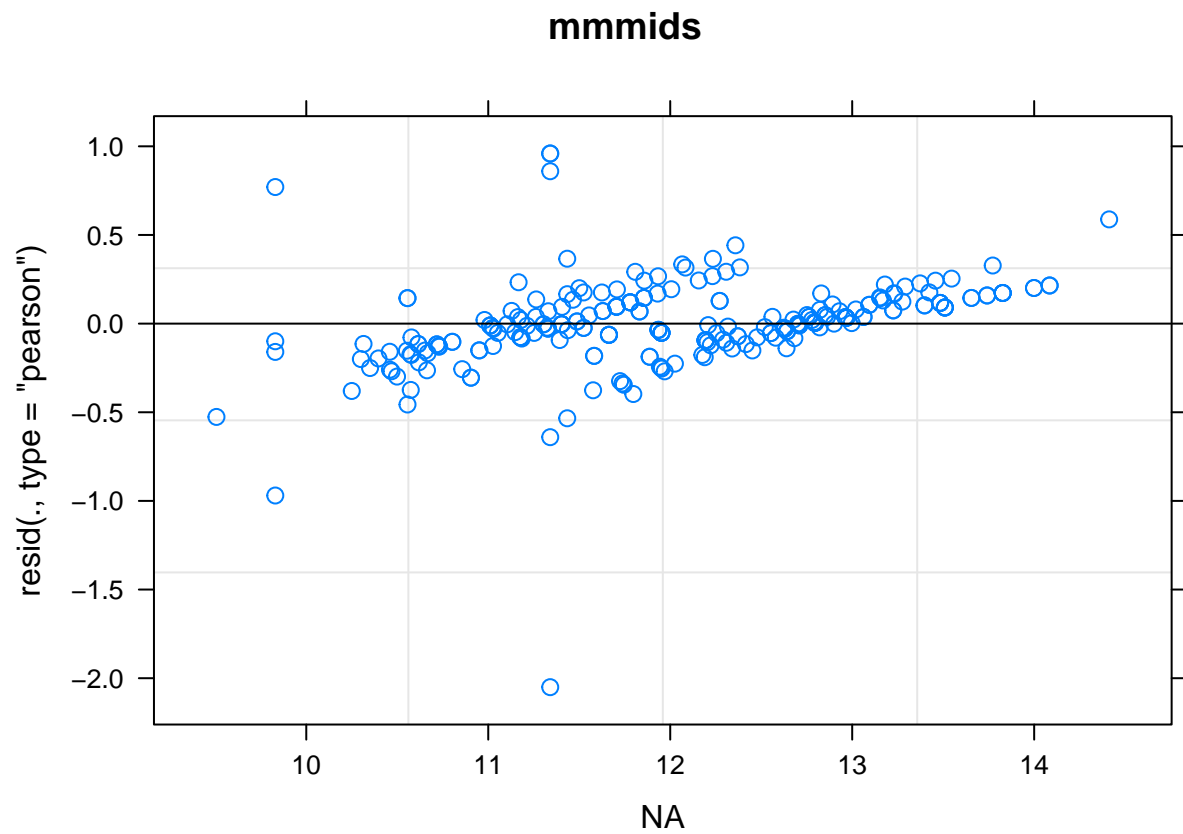
```
result = vector("list", length(models41))  
#This is a loop to calculate the MSE for each trait.  
for(i in 1:length(models41)){  
  result[[i]] = tryCatch(mean((predict(models41[[i]], qdata,type = "response") - qdata[[12+i]])^2),  
    error = function(e) paste("something wrong here"))  
}
```

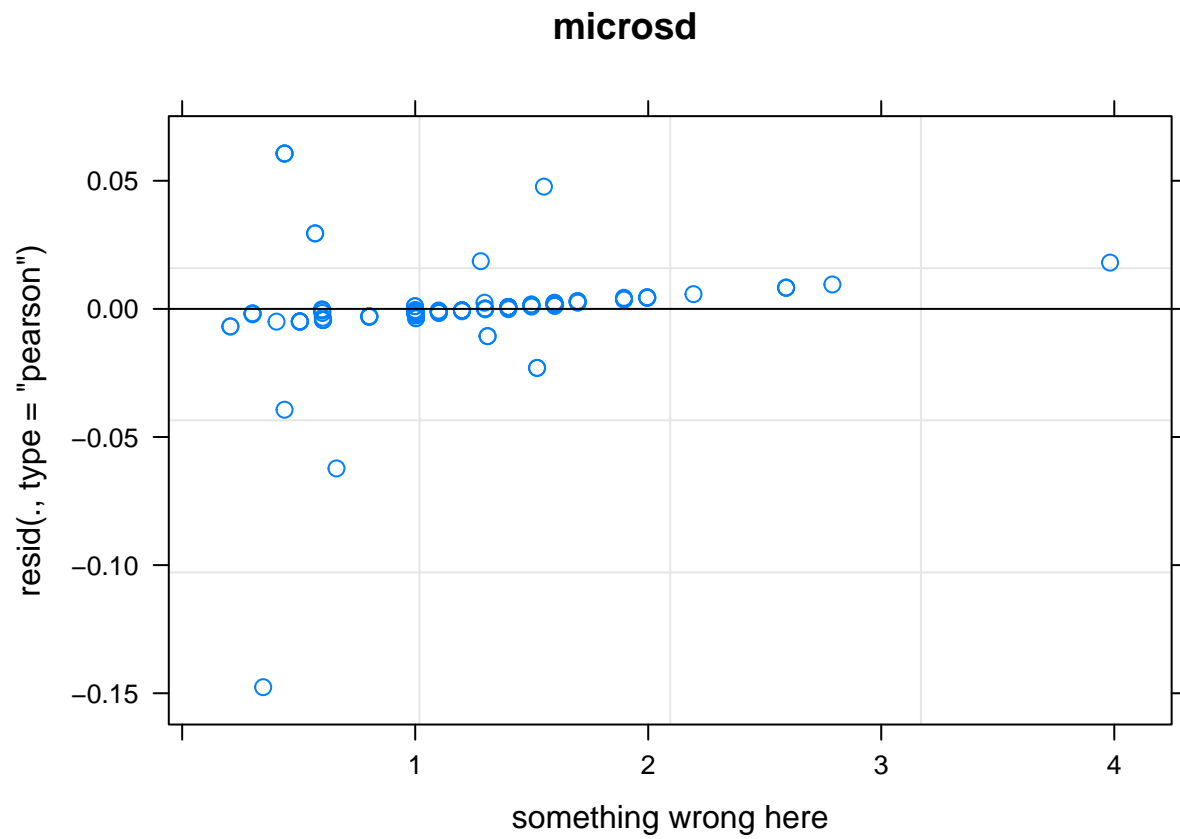
Some of the variables having missing data which messes up the prediction to calculate the MSE

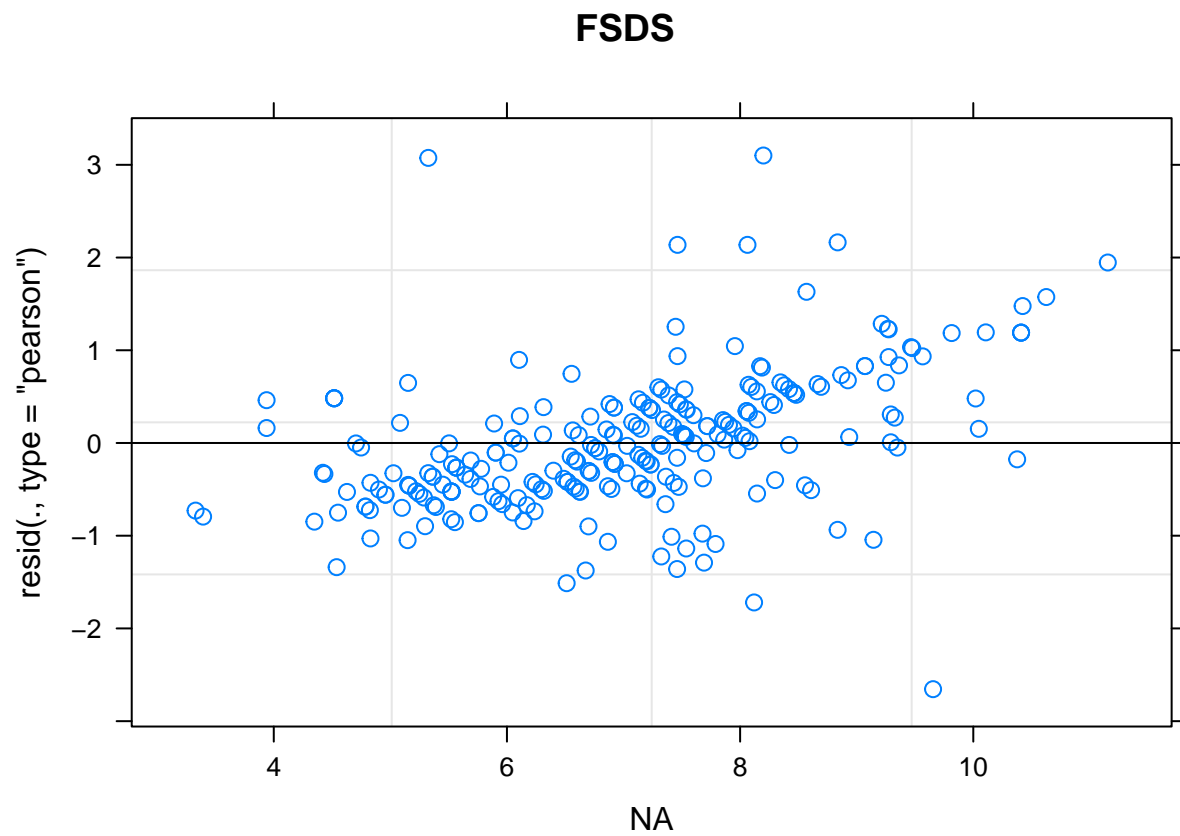
```
#par(mfrow=c(2,2))  
#This plots the predicted vs residual values to explore heterodascasity and the validation of the assum  
for(i in 1:length(models41)){  
  z=models41[[i]]  
  print(plot(z,main=names(models41)[i],xlab=as.character(result[[i]])))  
}
```

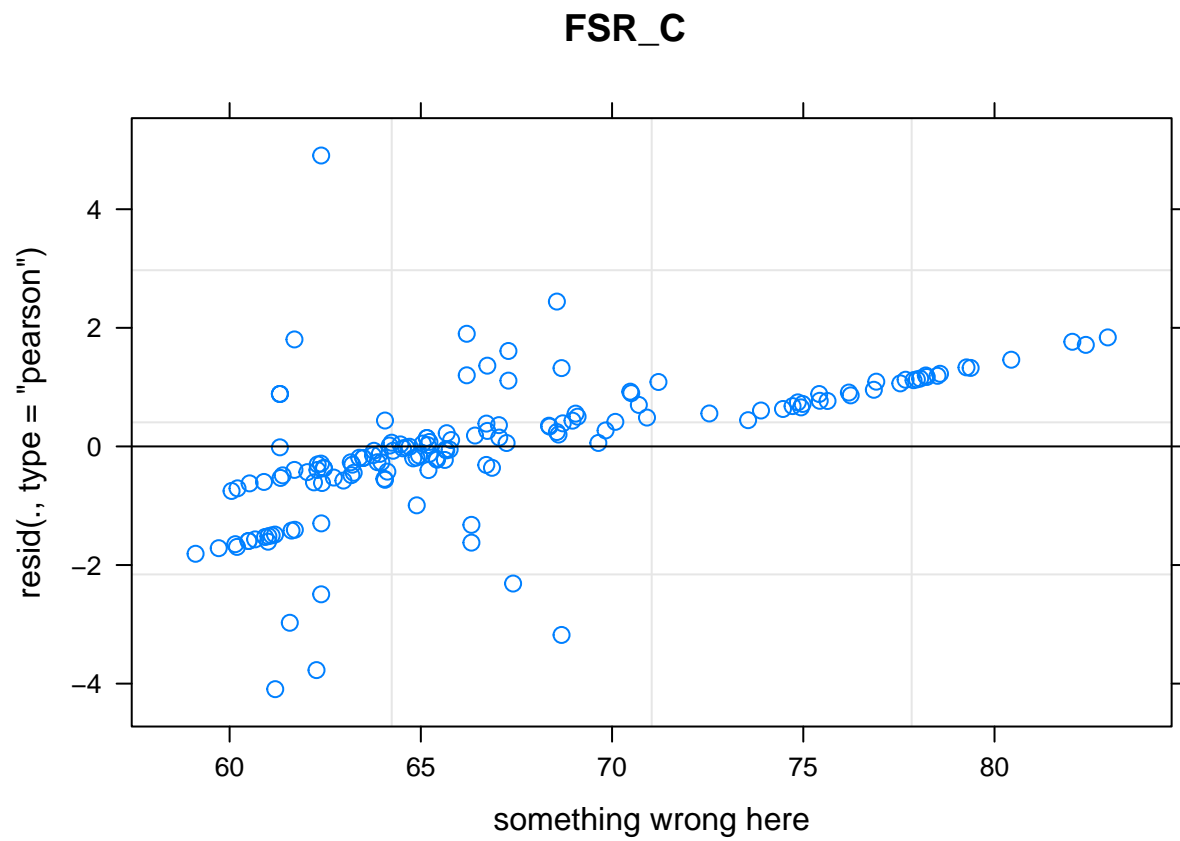


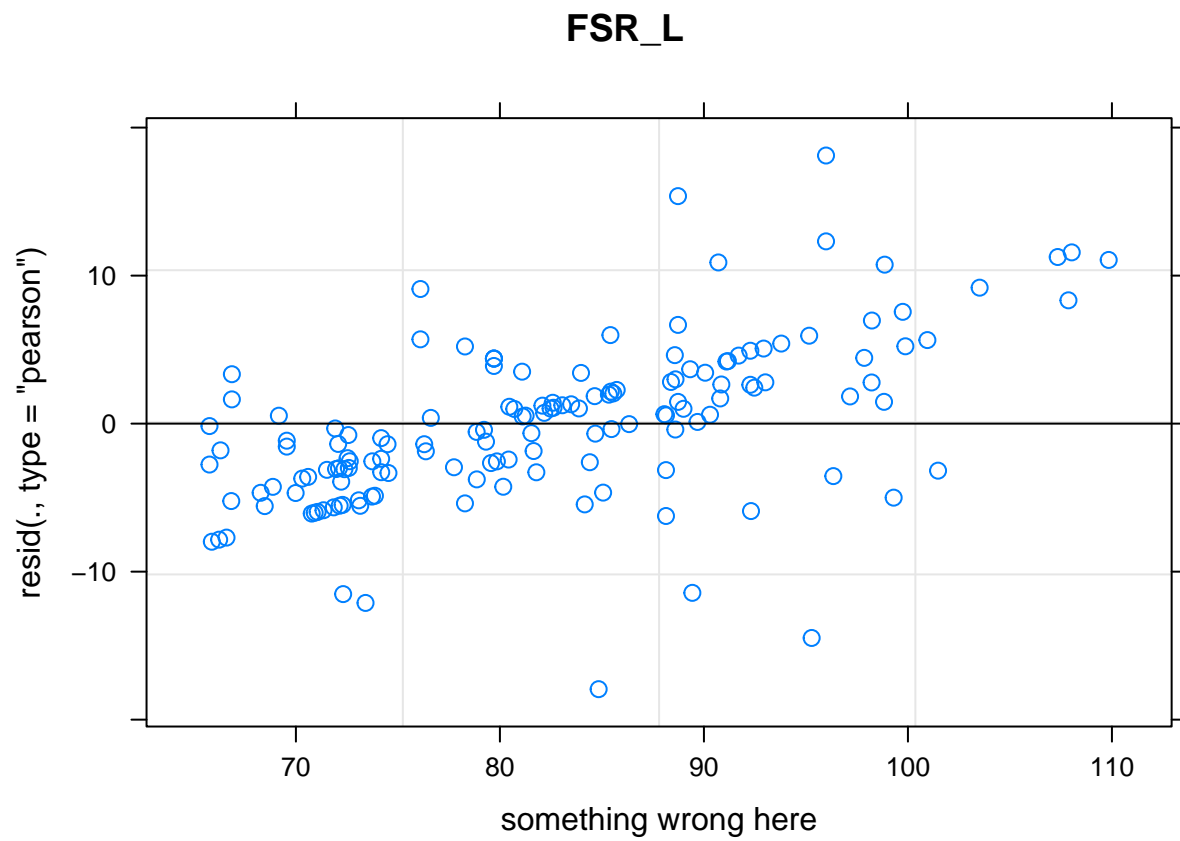


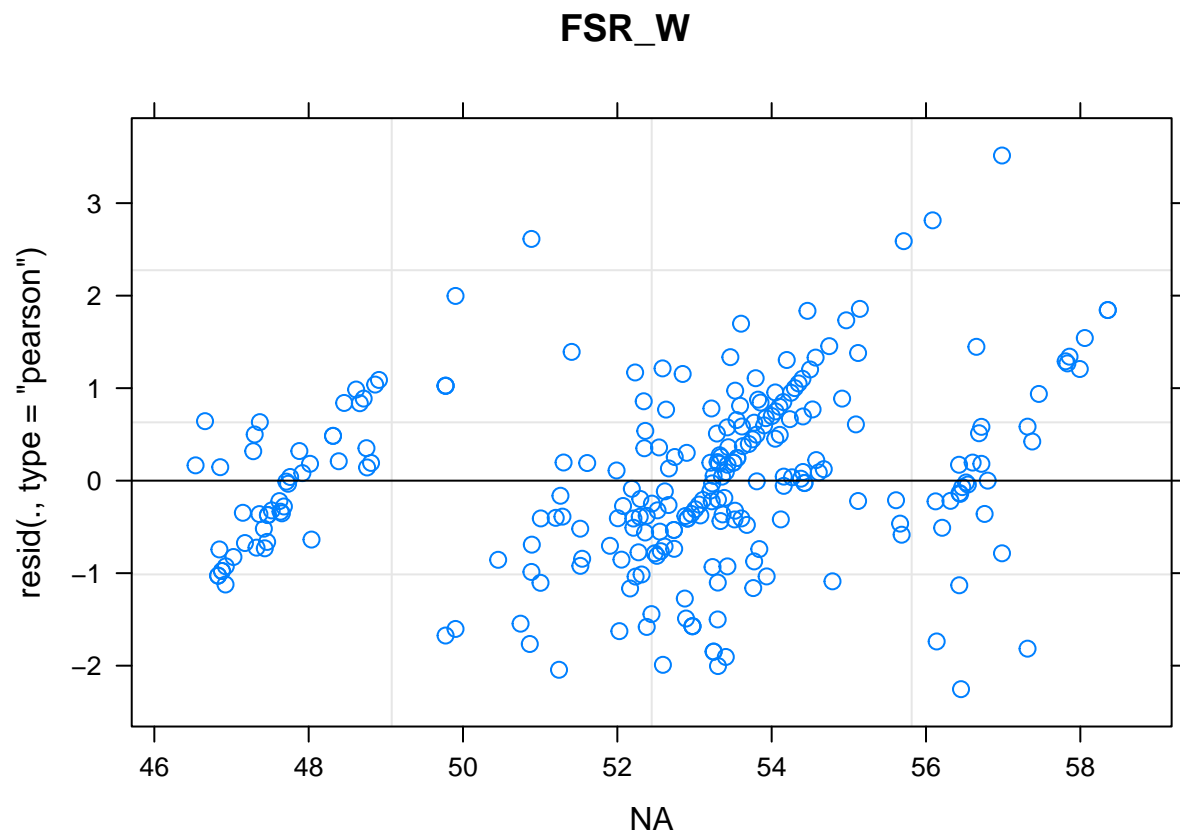


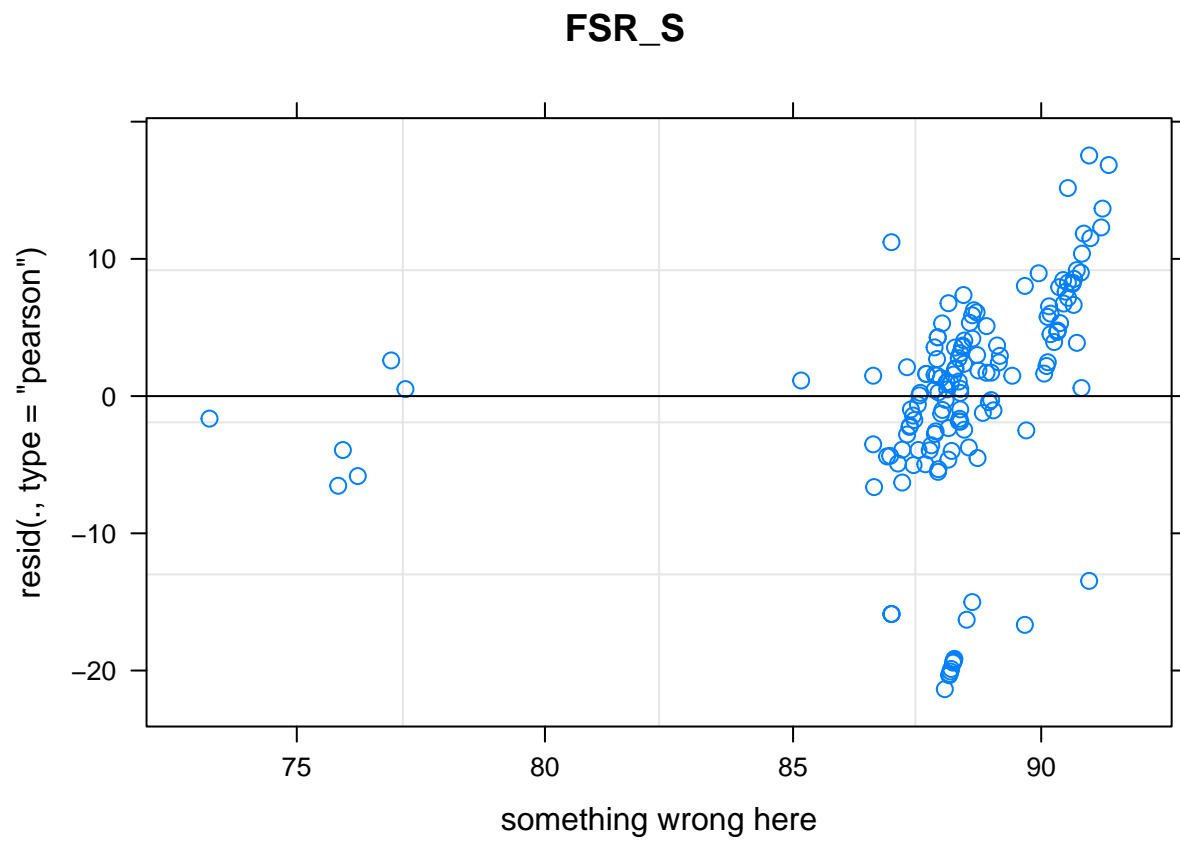




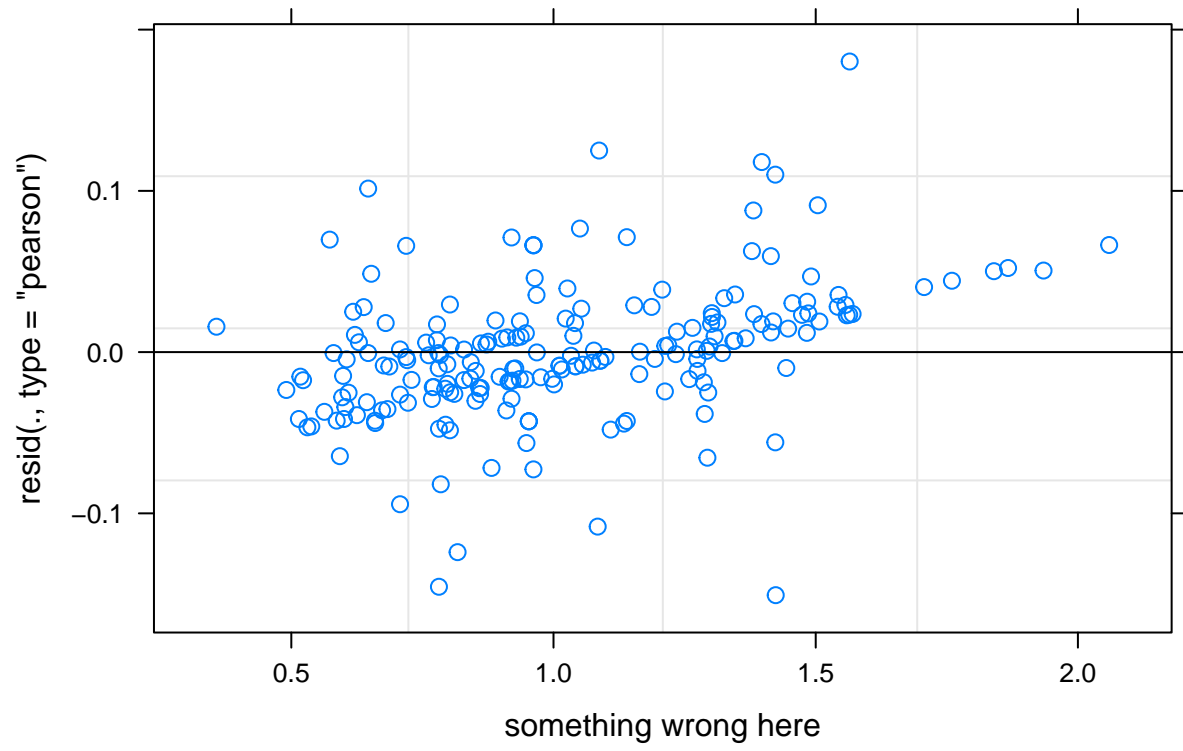


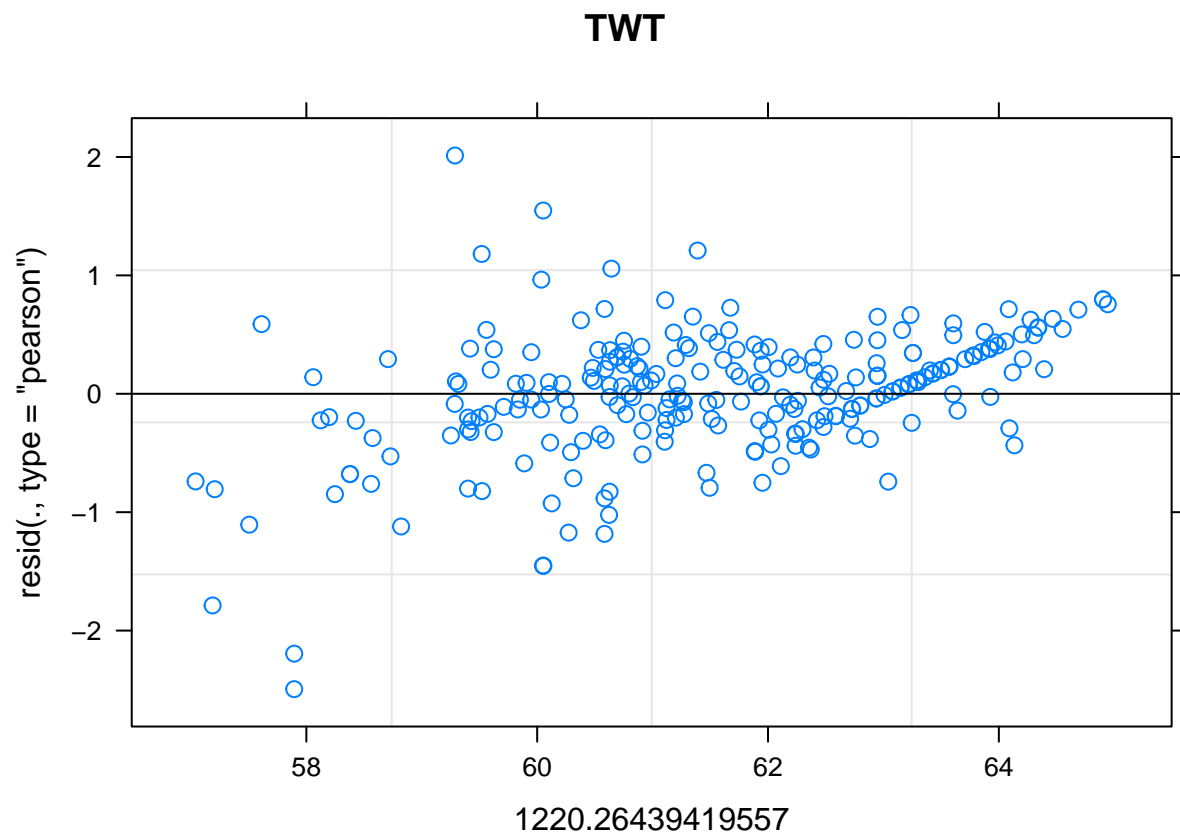




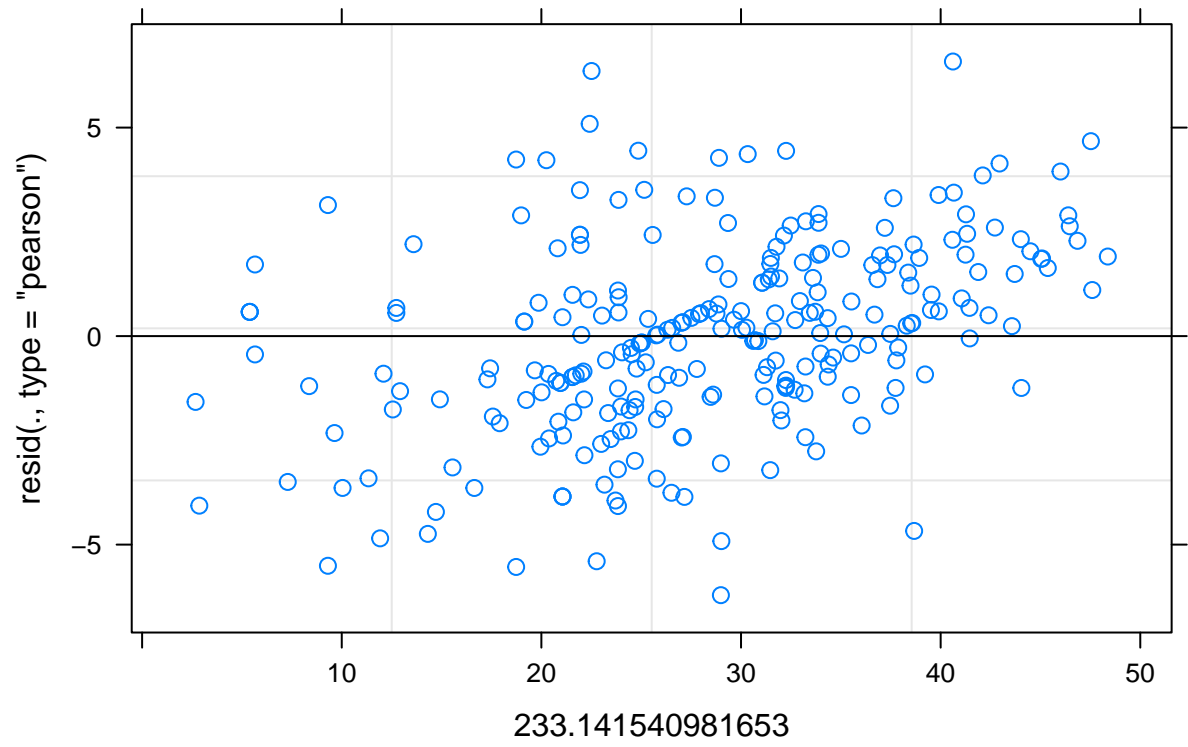


LDOPA

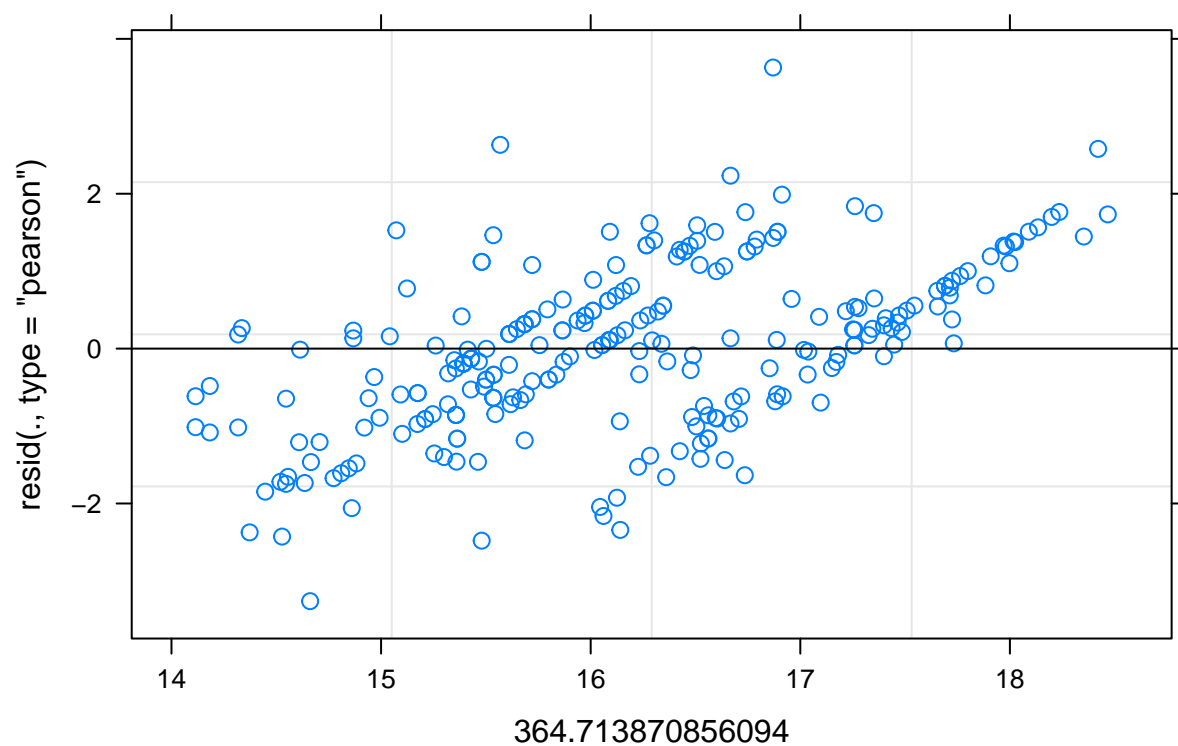




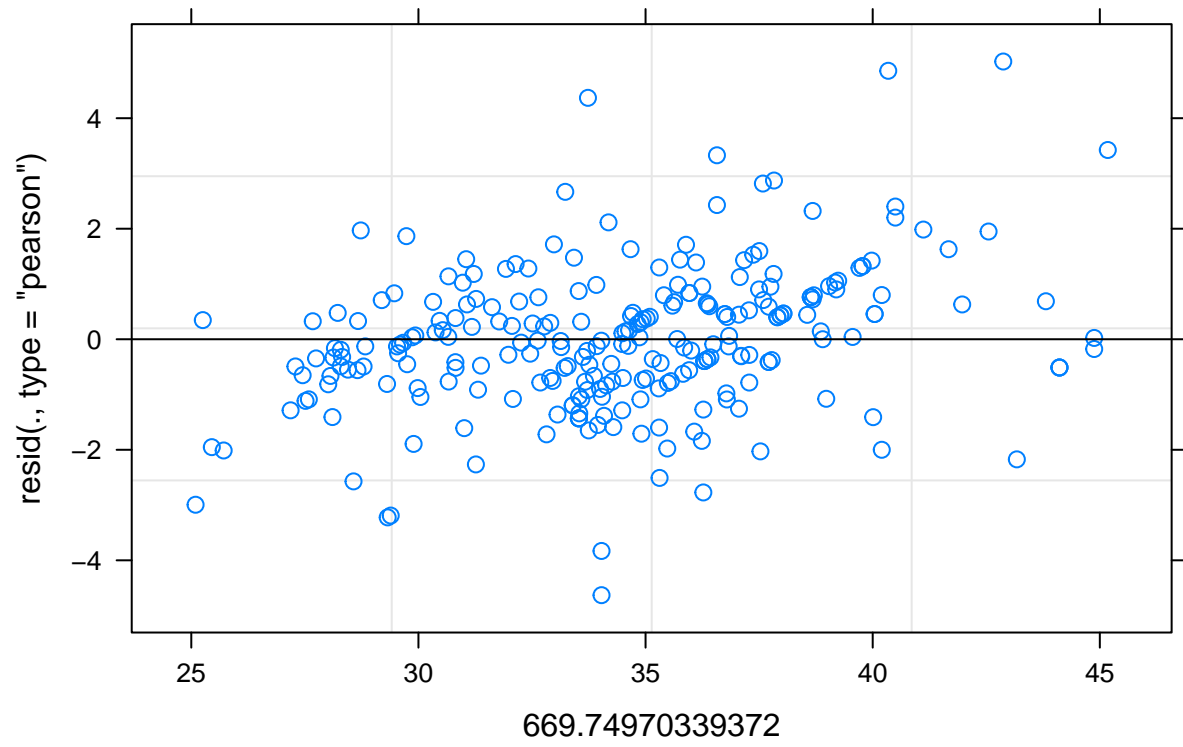
SKHRD



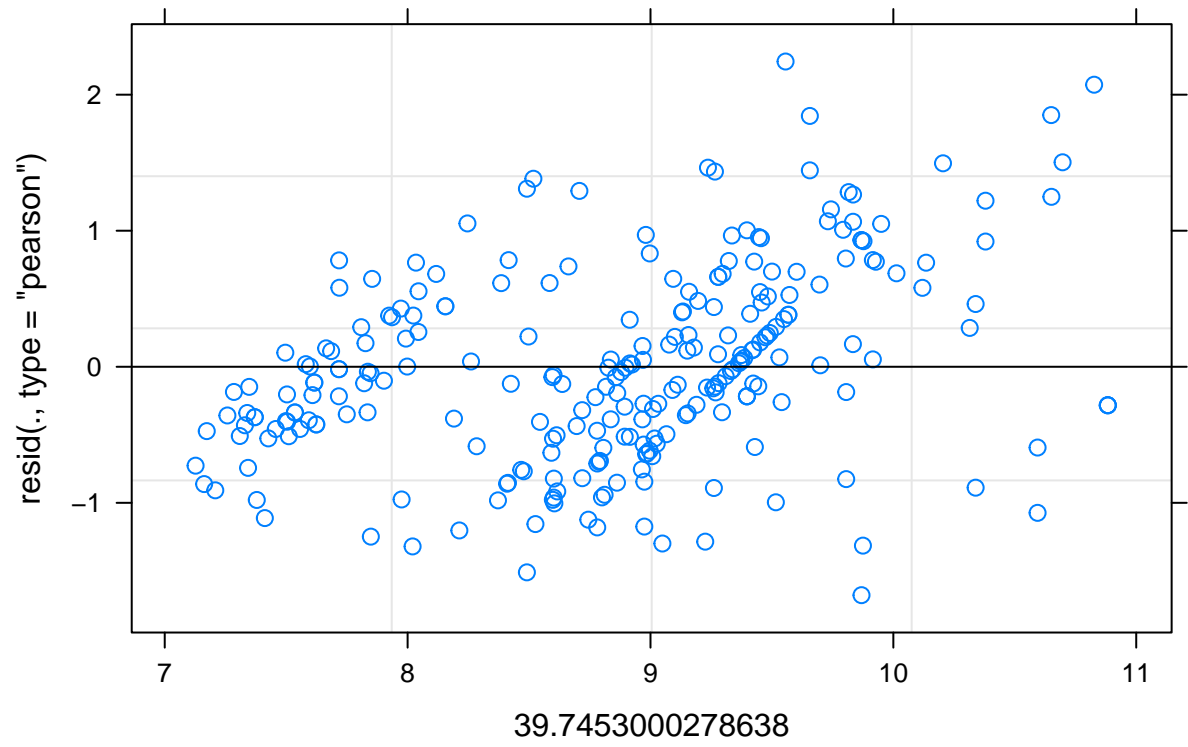
SKHRDSD

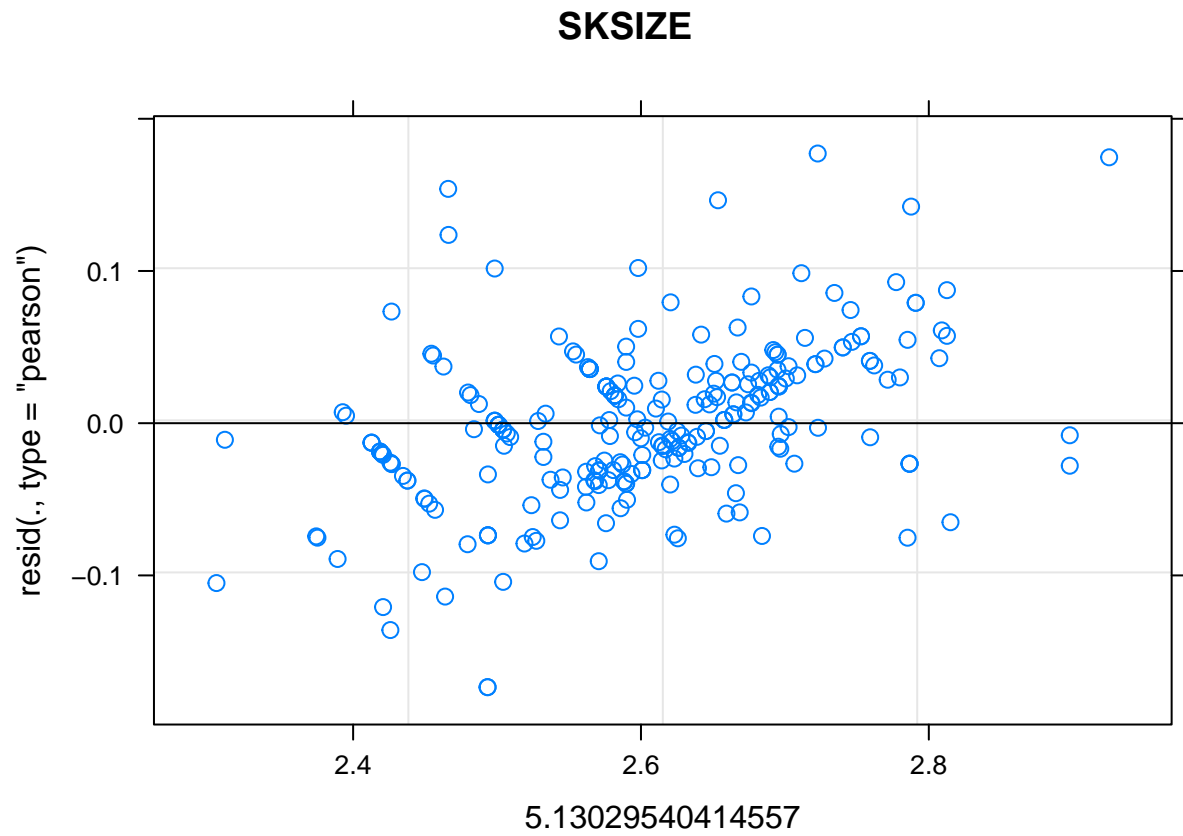


SKWT

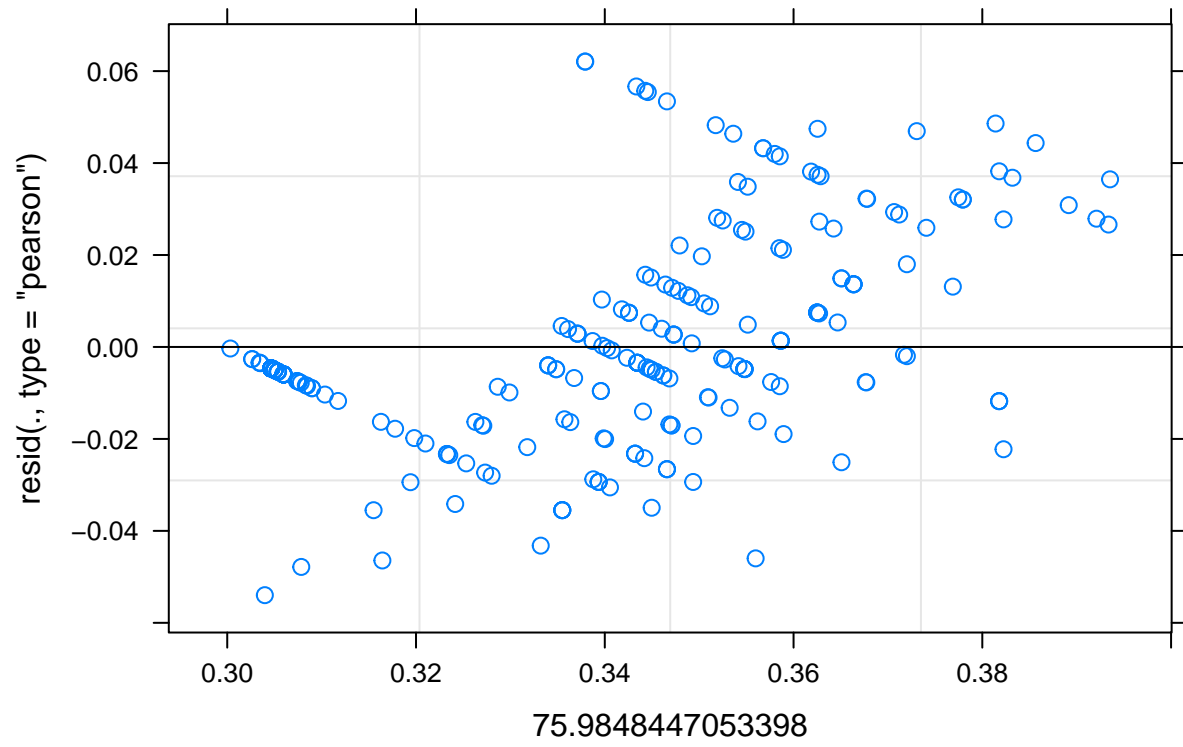


SKWTSD

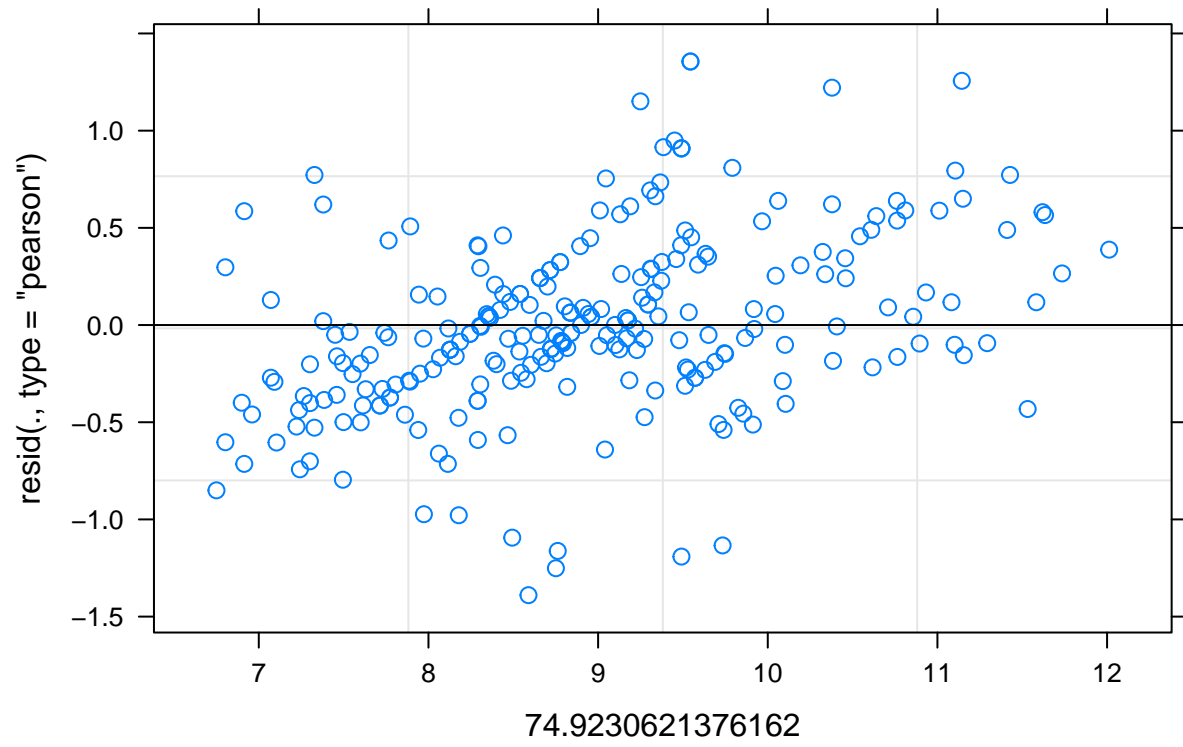




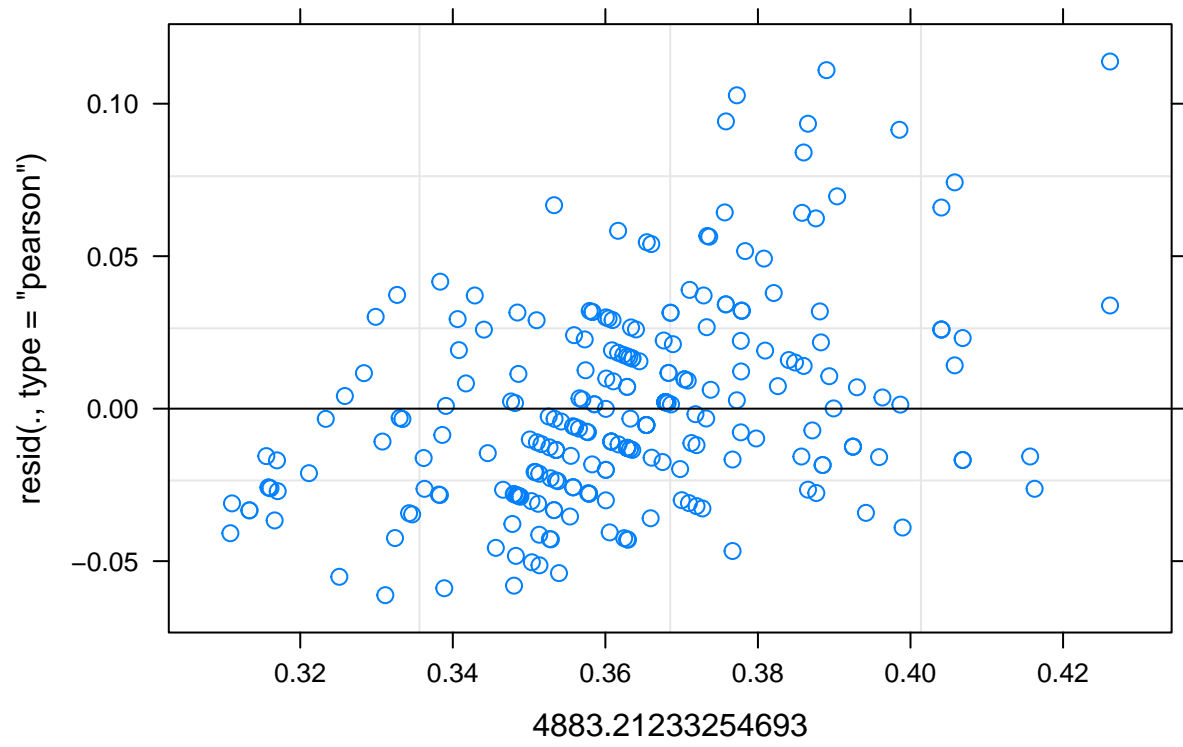
SKSIZESD

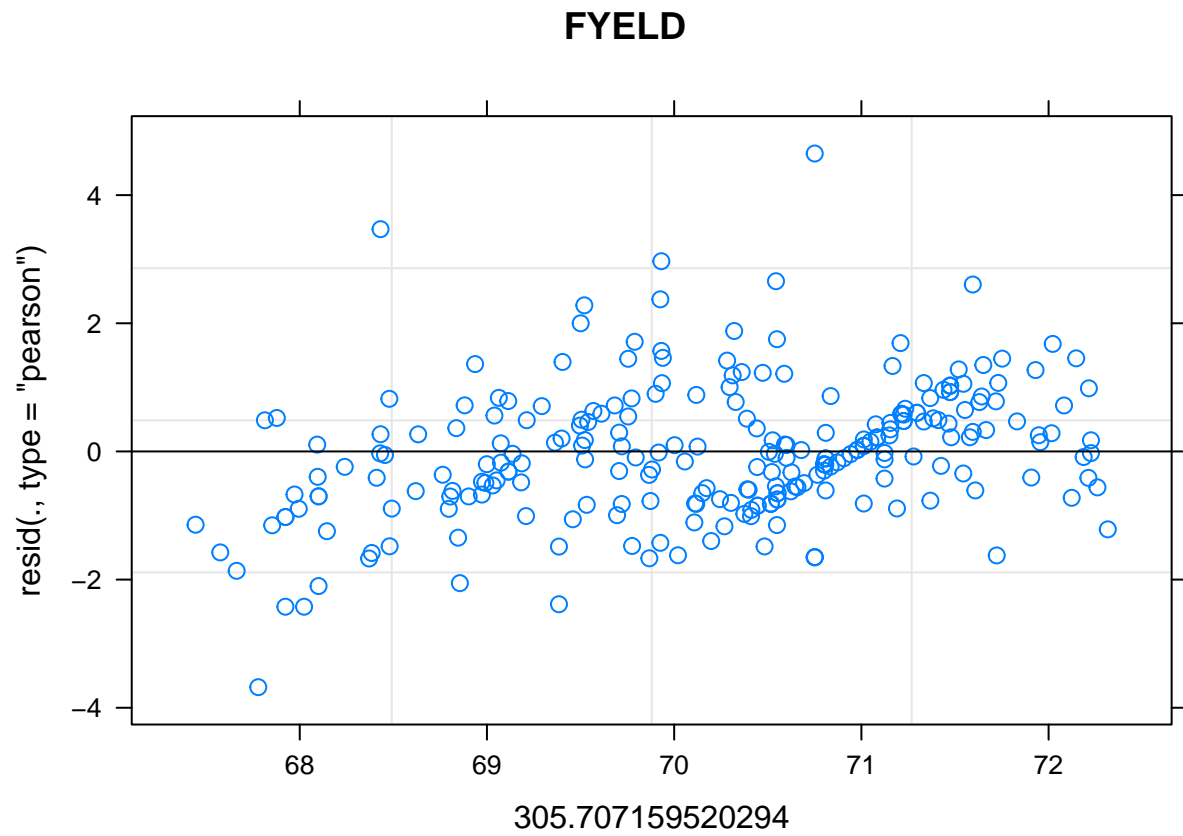


FPROT

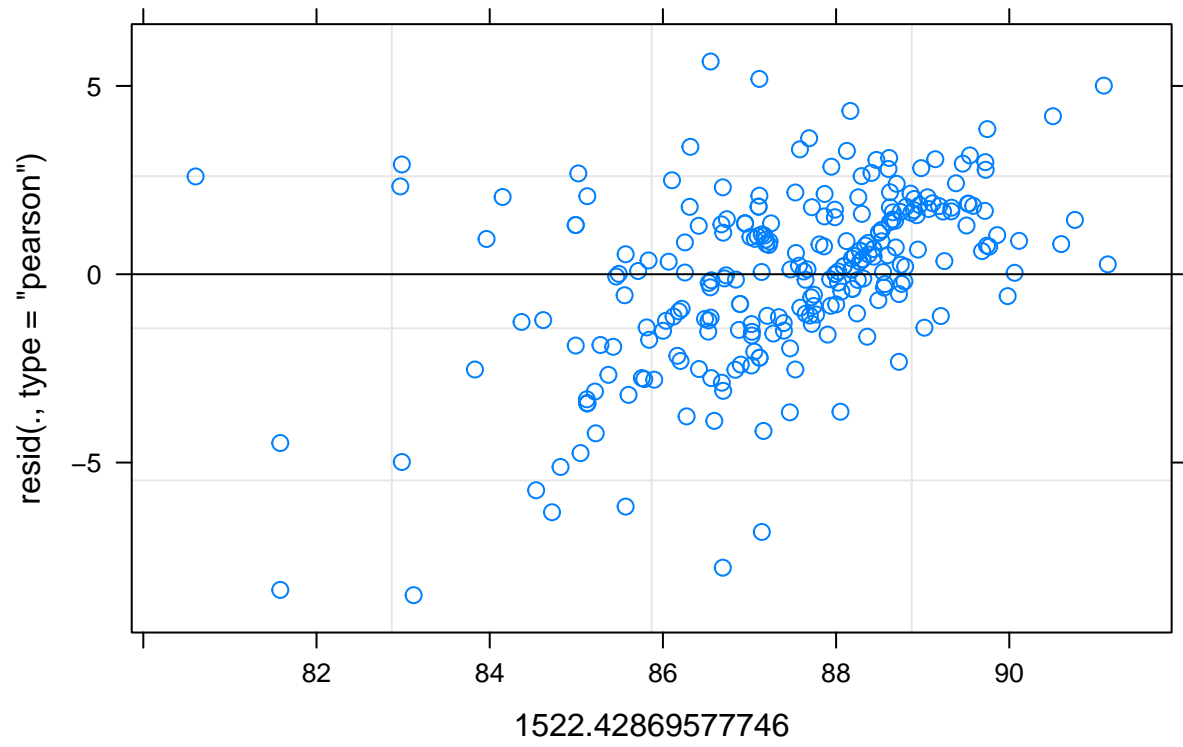


FASH

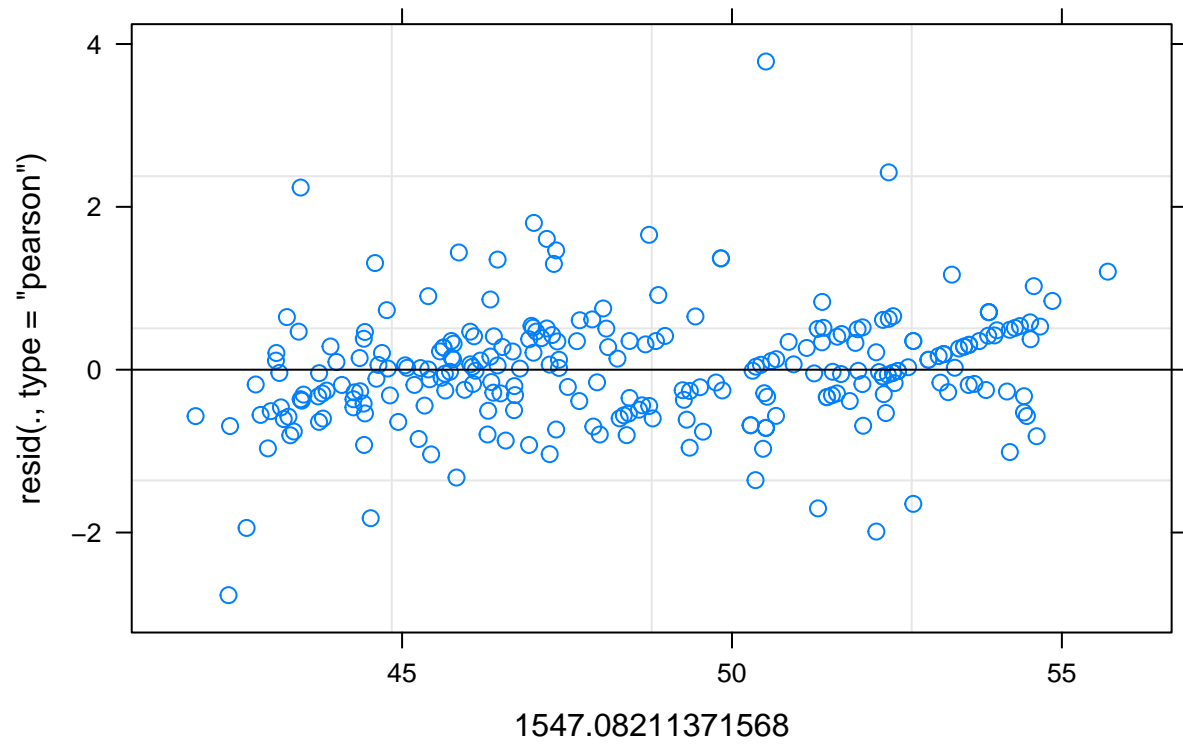


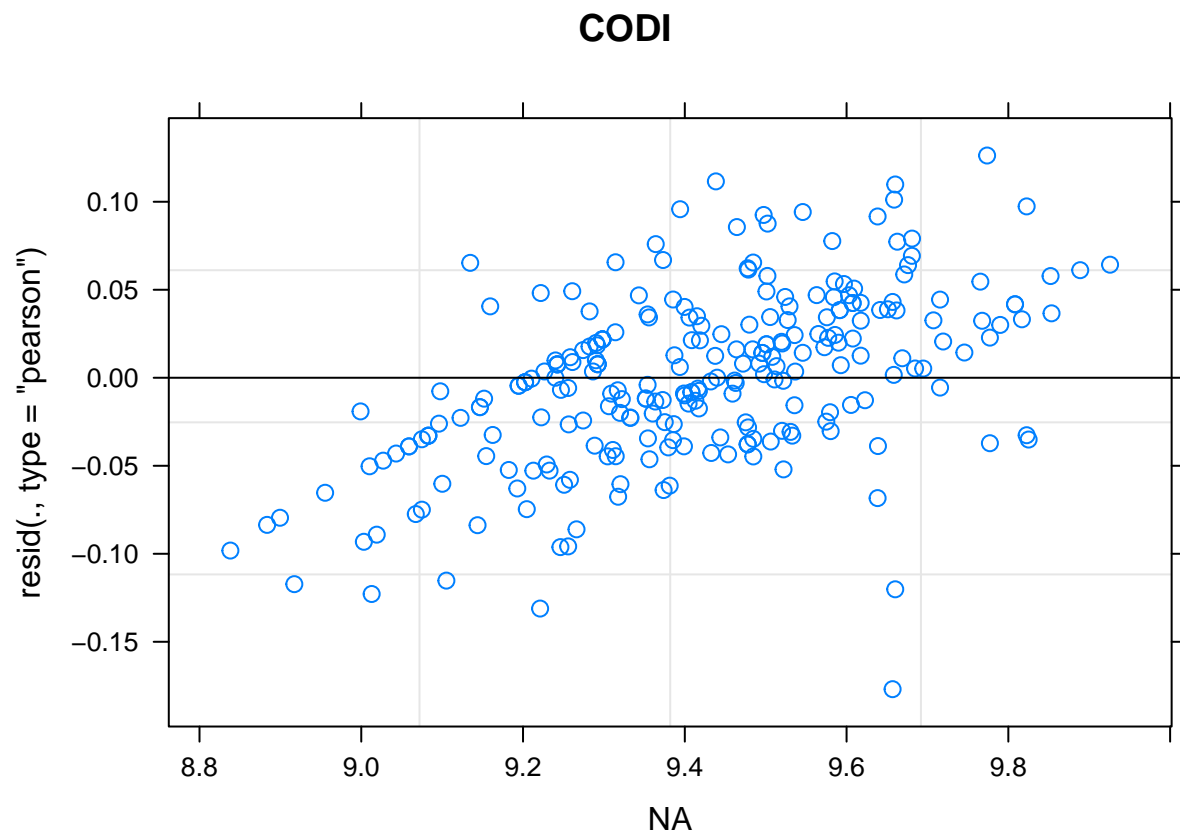


MSCOR

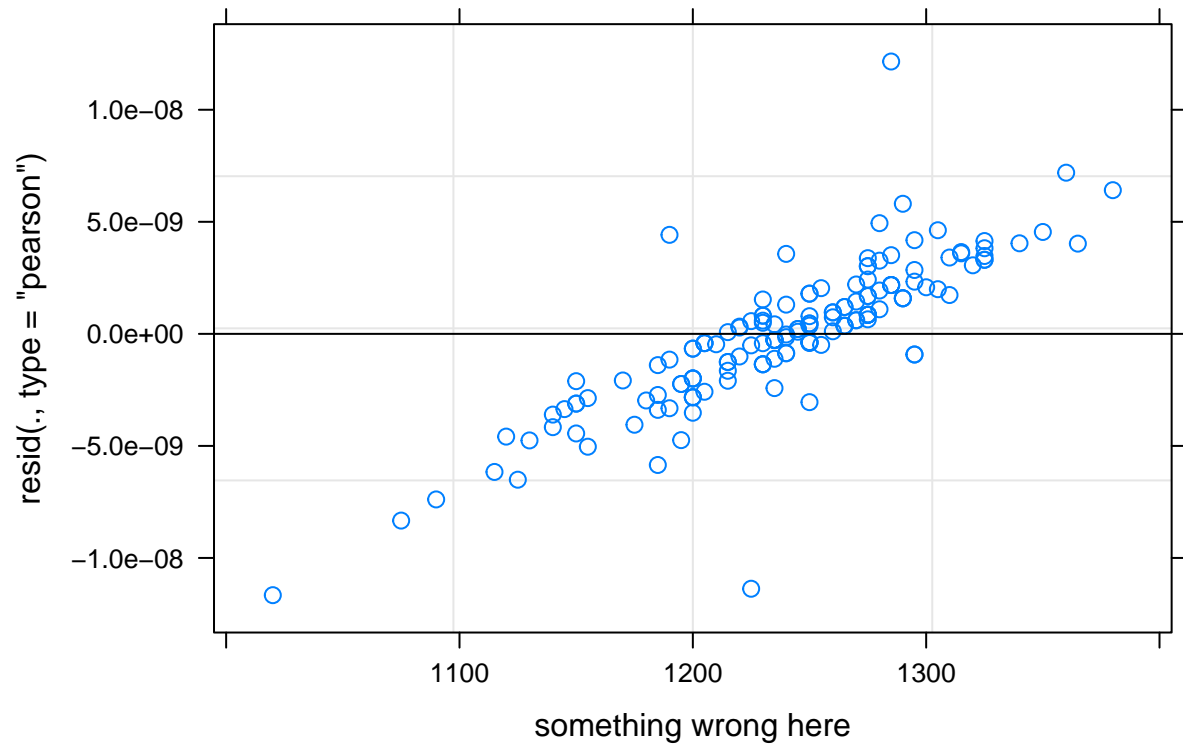


BKIFYELD





CAVOL



Amount of Lines with calculateble BLUPs for each trait.

```
for(i in 1:length(models41)){
  z=models41[[i]]
  rr1 <- ranef(z)
  resline=rr1$NAMET
  print(length(rownames(resline)))
}
```

```
## [1] 138
## [1] 138
## [1] 138
## [1] 74
## [1] 138
## [1] 89
## [1] 89
## [1] 138
## [1] 89
## [1] 126
## [1] 138
## [1] 138
## [1] 138
## [1] 138
## [1] 138
## [1] 138
## [1] 138
```

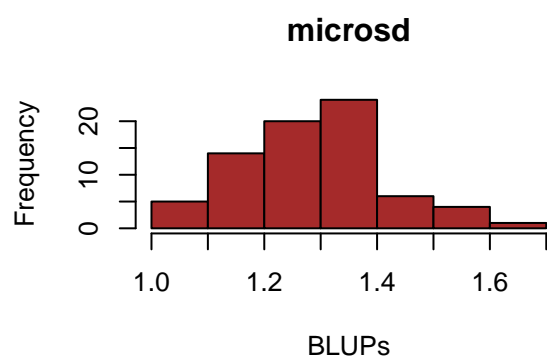
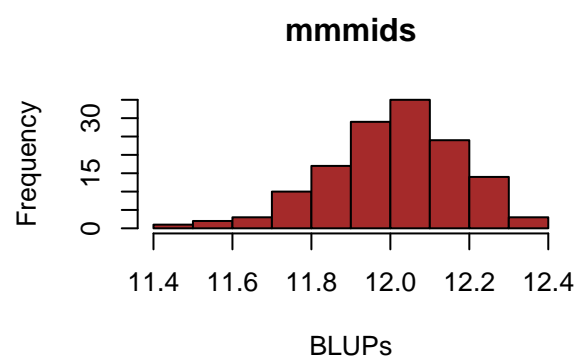
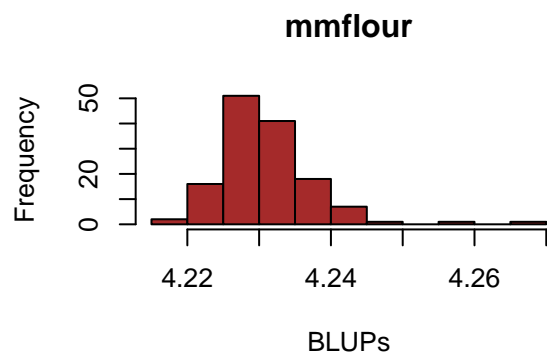
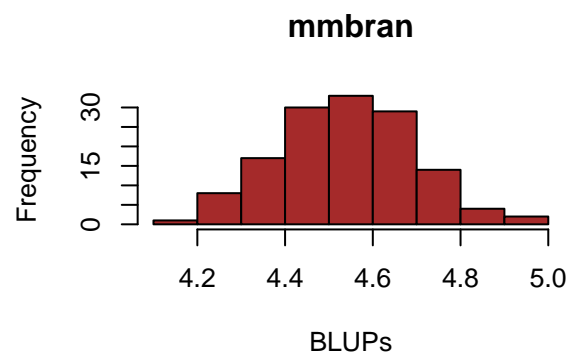
```
## [1] 138
## [1] 138
## [1] 138
## [1] 138
## [1] 138
## [1] 138
## [1] 123
```

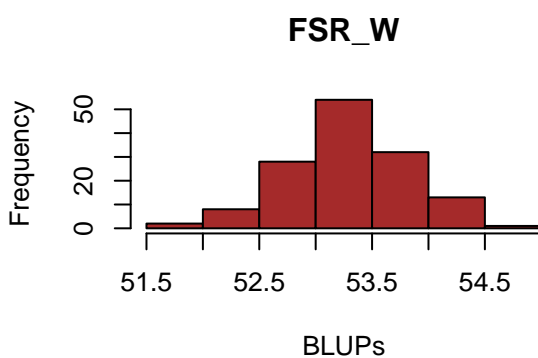
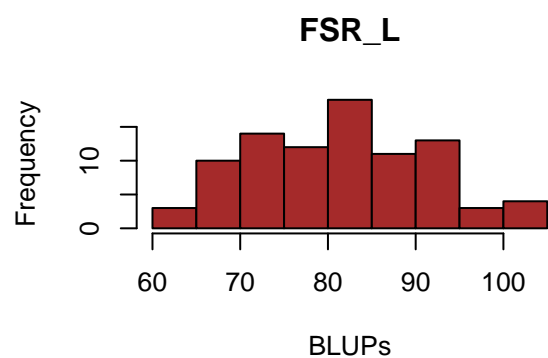
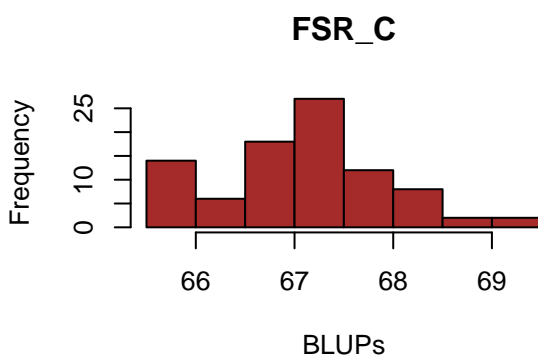
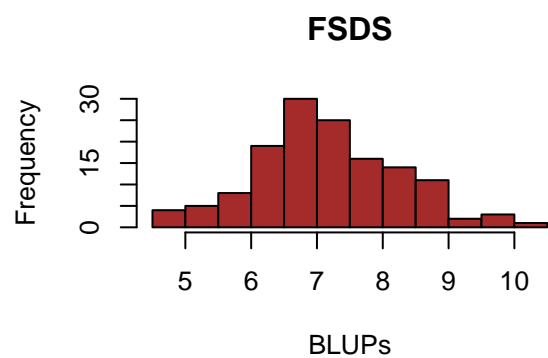
BLUPs for each trait

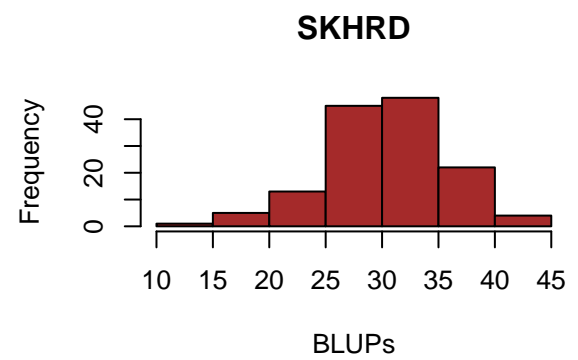
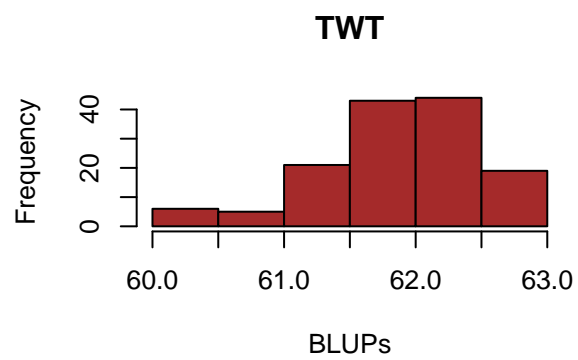
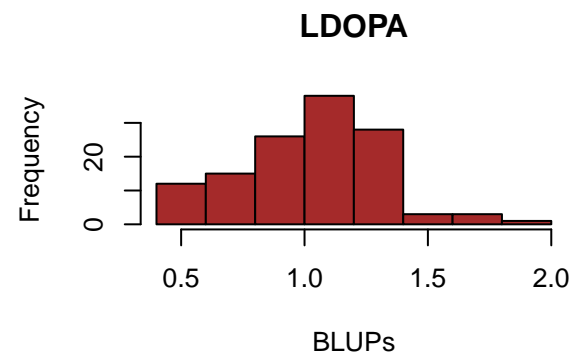
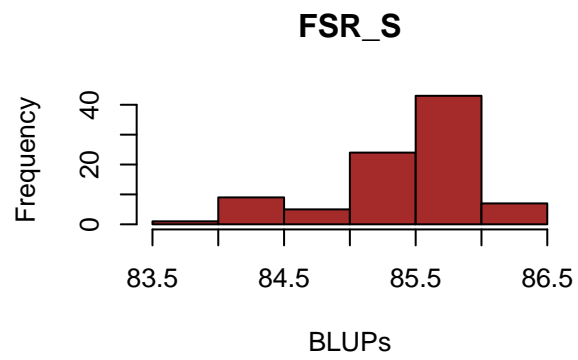
```
#I built these loops to extract the random predicted effects or BLUPs for each trait using the mean-line
BLUES=list()
for(i in 1:length(models41)){ #Extract the random and fixed effect for each model and line.
  z=models41[[i]]
  rr1 <- ranef(z)
  mu=fixef(z)[[1]]
  resline=rr1$NAMET
## Creating plots with the BLUPs
# Create a numeric vector with the BLUP for each line
  LINEBLUP = resline[,1] #create a vector of just the predicted effects
  BLUPs=c()
  for(j in 1:length(LINEBLUP)){ #adds the predicted effect of each line to the populatio mean to have the
    x=resline[j,1]
    y=x+mu
    BLUPs=c(BLUPs,y)
  }
  resline=cbind(resline,BLUPs) #create a data frame for all of the BLUPs for each trait and line
  resline1=as.data.frame(resline[,2])
  rownames(resline1)=rownames(resline)
  d <- cbind(rownames(resline1), data.frame(resline1, row.names=NULL))
  names(d)<-c("NAMET",names(models41)[i])
  BLUES[[names(models41)[i]]]<-d
}

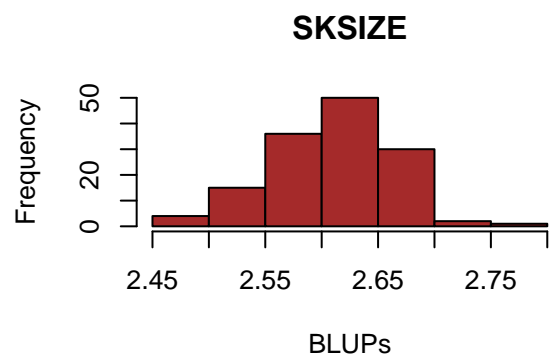
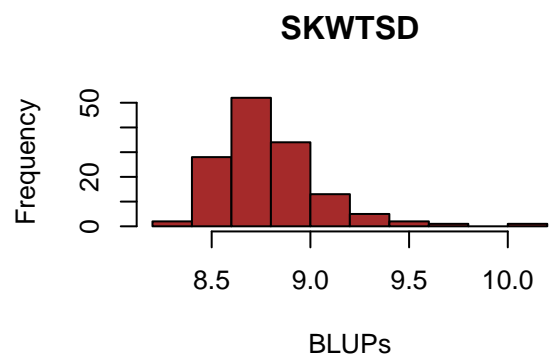
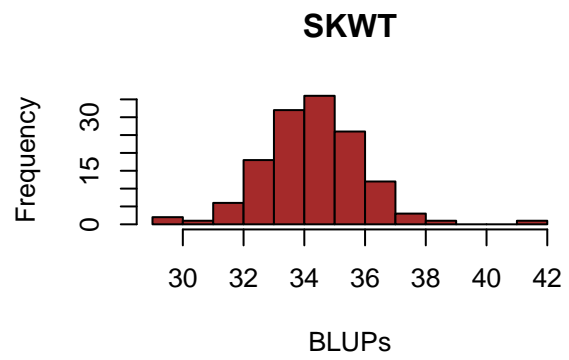
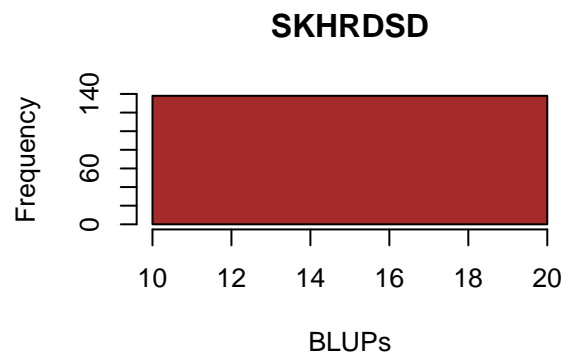
BLUEz=qdata[,c(6,7,8,10)]
for(i in 1:length(models41)){
  k=BLUES[[i]]
  BLUEz=left_join(BLUEz,k,by=c("NAMET"))
}

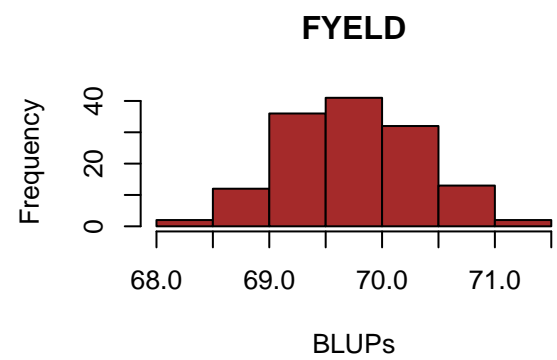
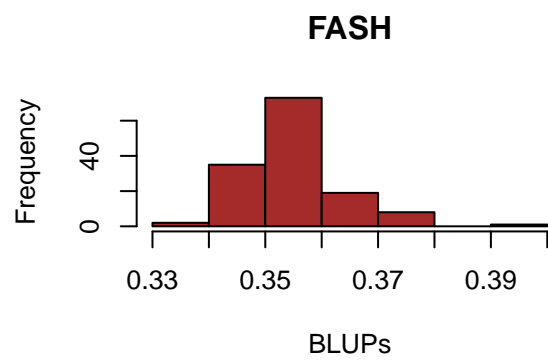
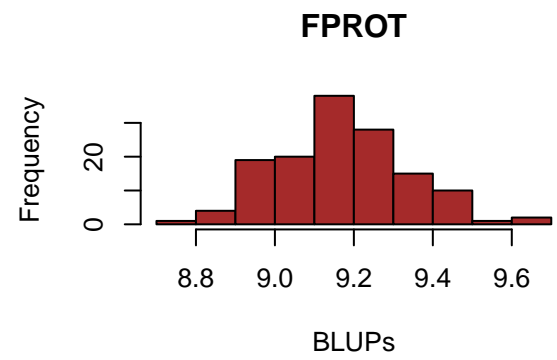
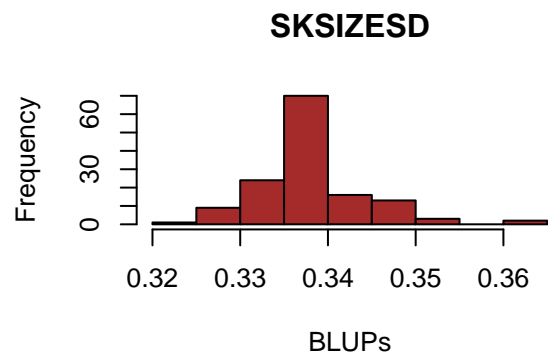
BLUEz1=aggregate(BLUEz[,5:28],list(BLUEz$NAMET),mean)
names(BLUEz1)[names(BLUEz1) == "Group.1"] <- "NAMET"
BLUEz1$NAMET=as.factor(BLUEz1$NAMET)
# Create a histogram with the BLUP for each line
par(mfrow=c(2,2))
for(i in 2:length(BLUEz1)){
  j=BLUEz1[,c(1,i)]
  j=j[complete.cases(j),]
  hist(j[,2],main=names(BLUEz1)[i],xlab="BLUPs",col="brown",na.rm=TRUE)
}
```

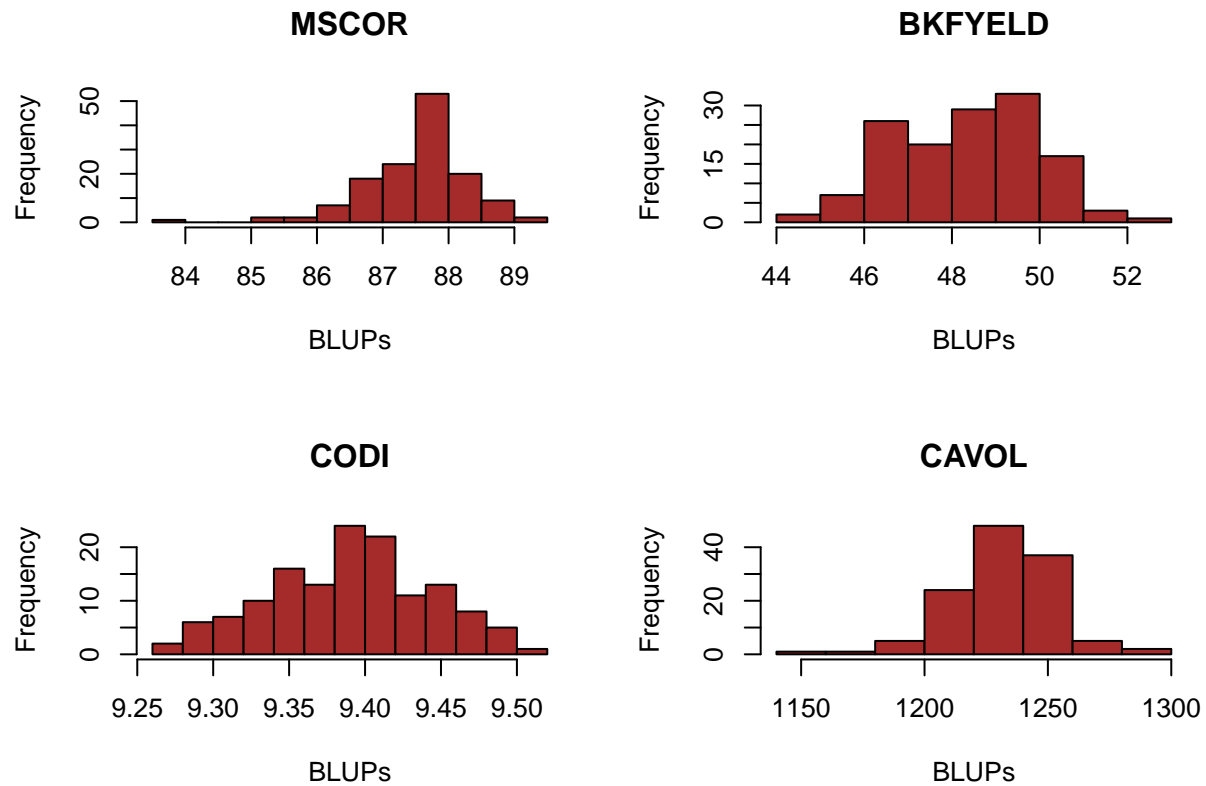








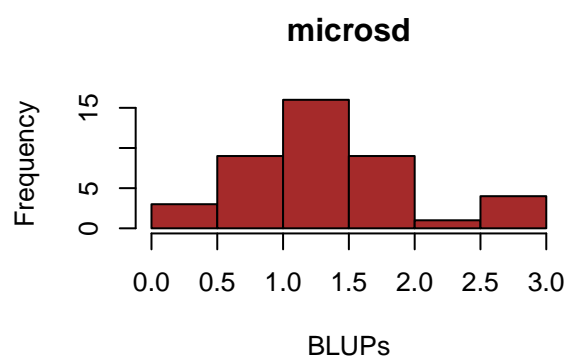
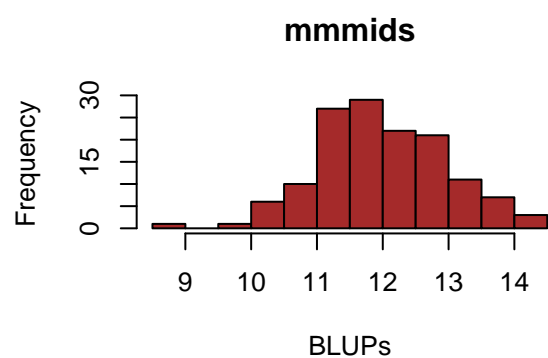
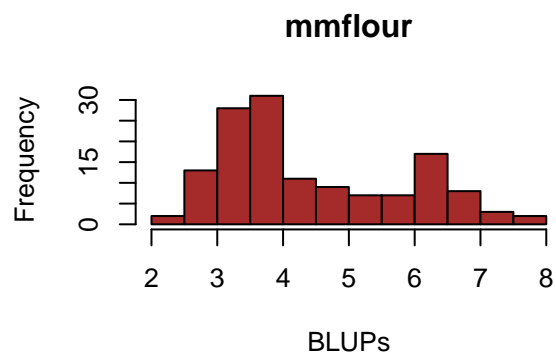
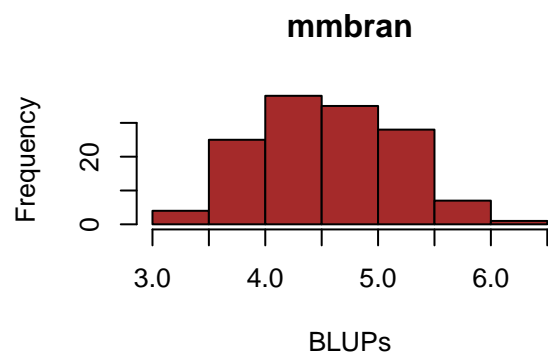


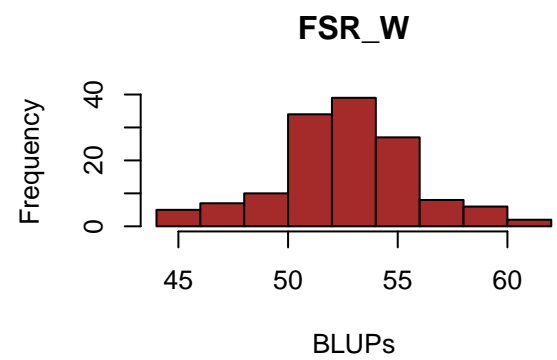
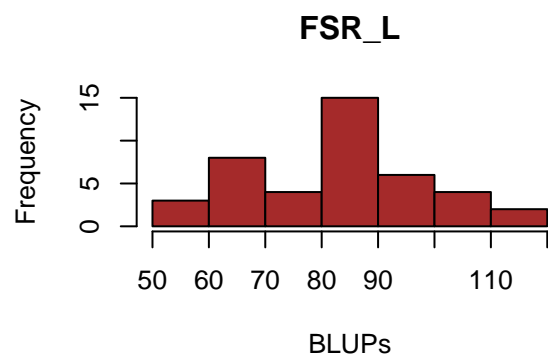
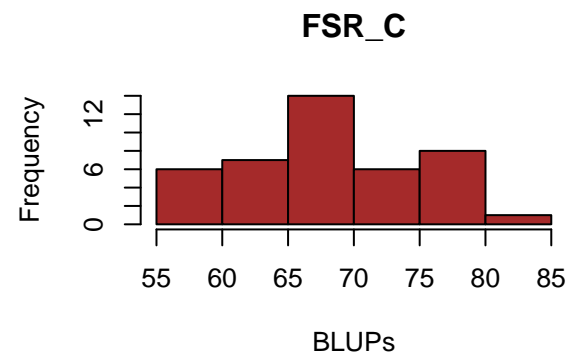
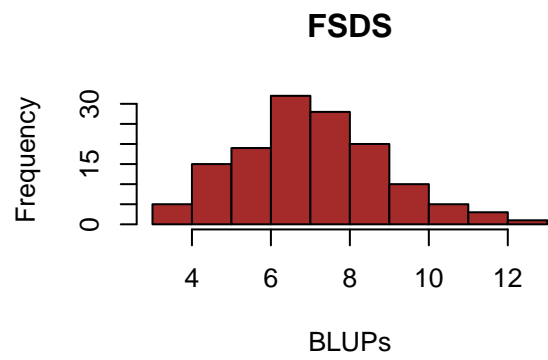


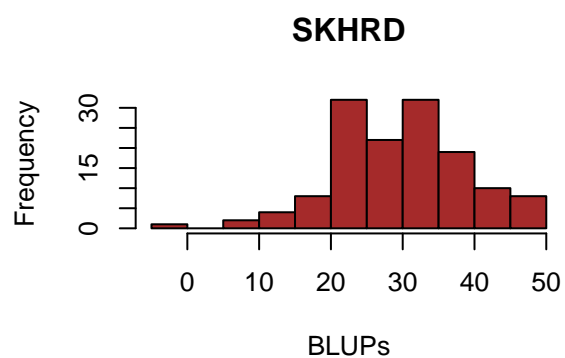
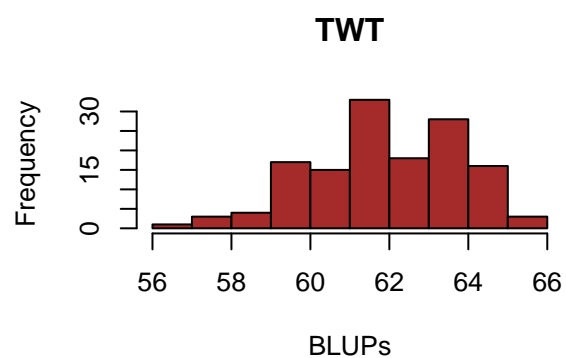
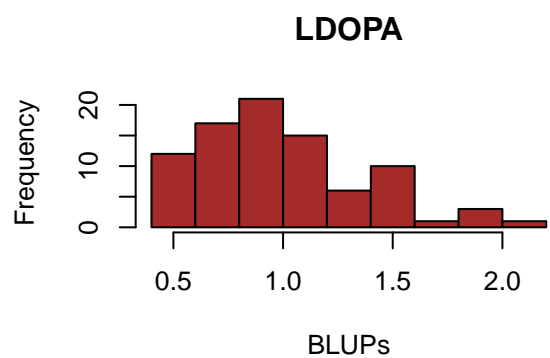
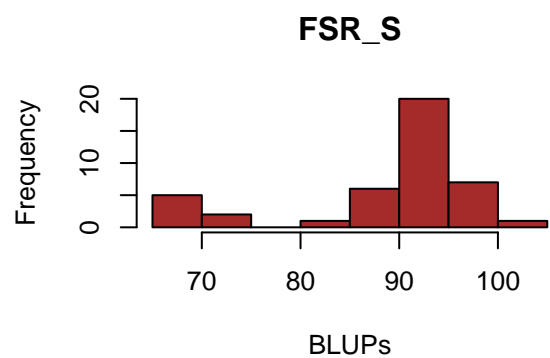
```
## Compare BLUP to line averages on a scatterplot
```

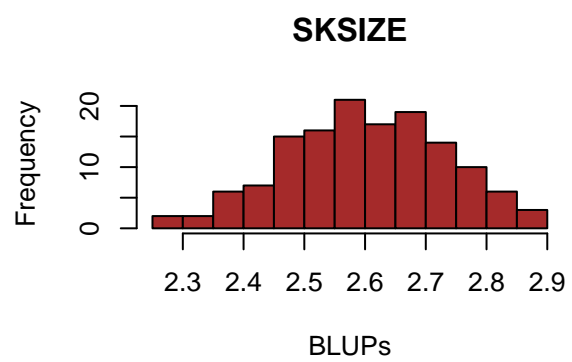
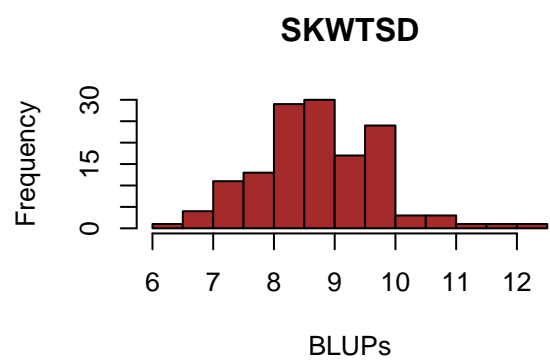
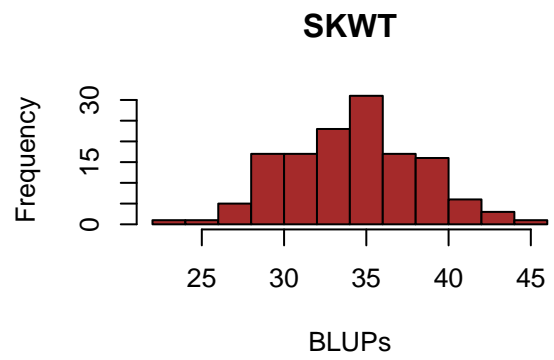
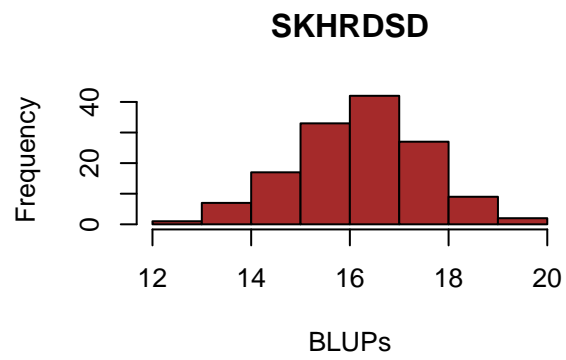
Histograms for line averages

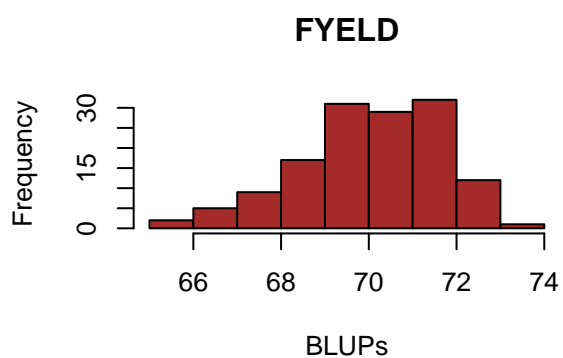
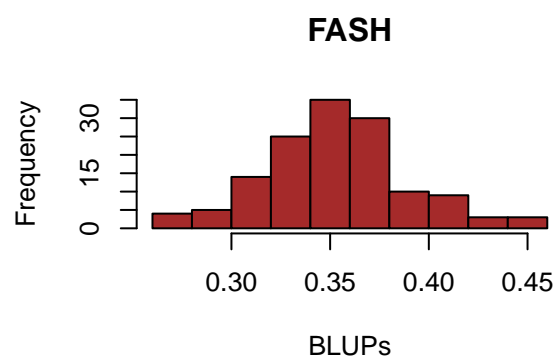
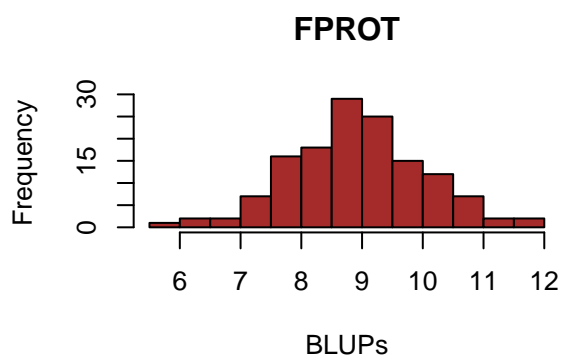
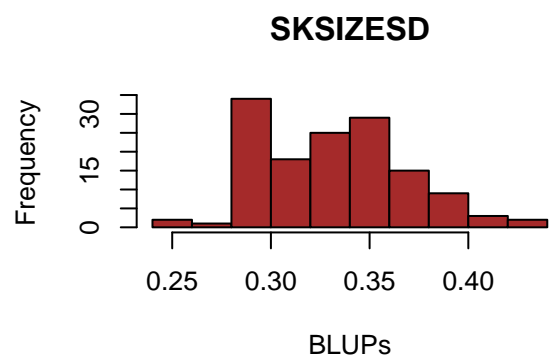
```
qnamet=aggregate(qdata[,12:35],list(qdata$NAMET),mean)
names(qnamet)[names(qnamet) == "Group.1"] <- "NAMET"
qnamet$NAMET=as.factor(qnamet$NAMET)
# Create a histogram with the line averages for each line
par(mfrow=c(2,2))
for(i in 2:length(qnamet)){
  j=qnamet[,c(1,i)]
  j=j[complete.cases(j),]
  hist(j[,2],main=names(qnamet)[i],xlab="BLUPs",col="brown")
}
```

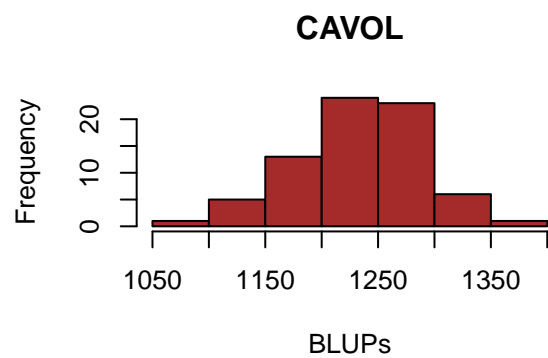
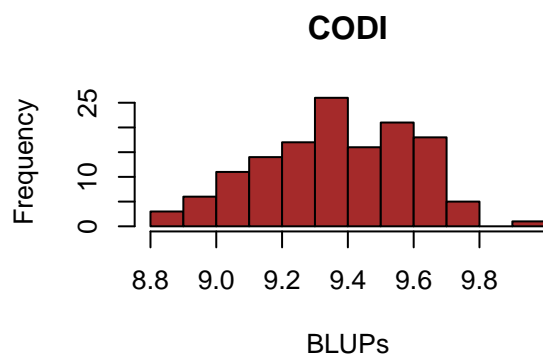
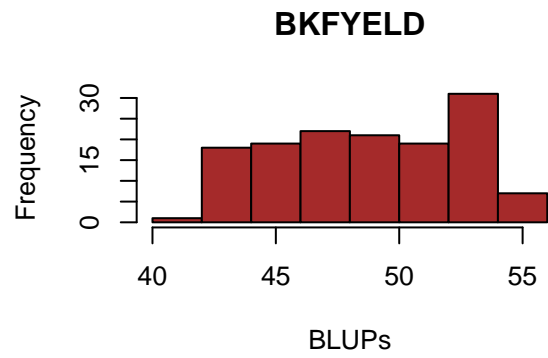
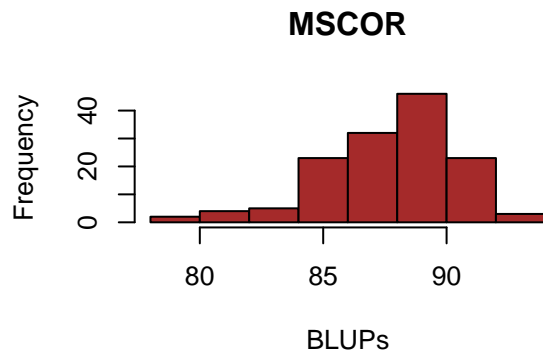







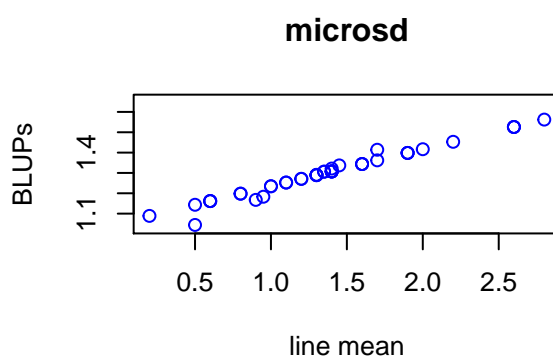
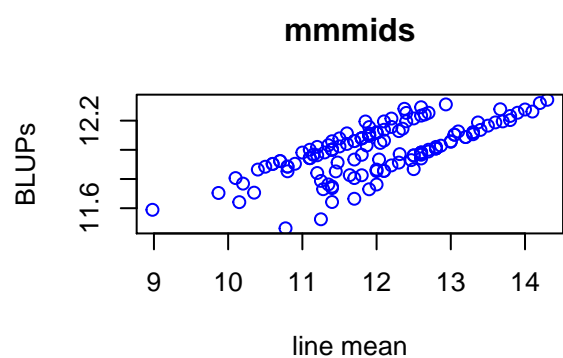
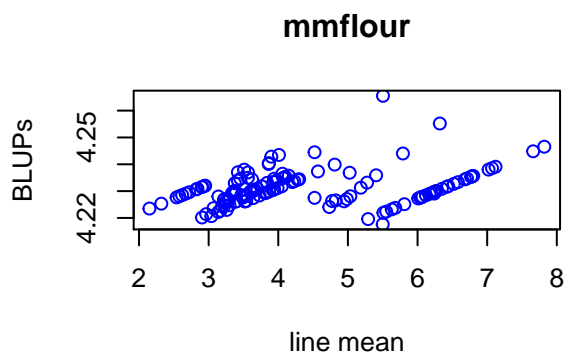
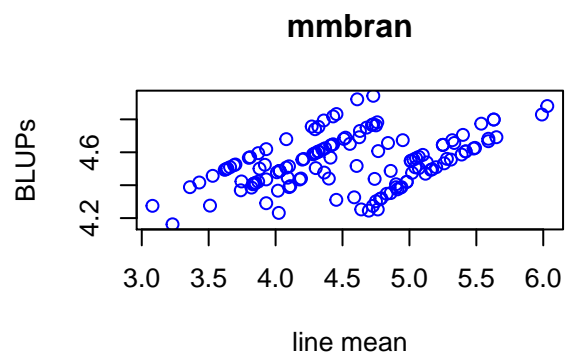


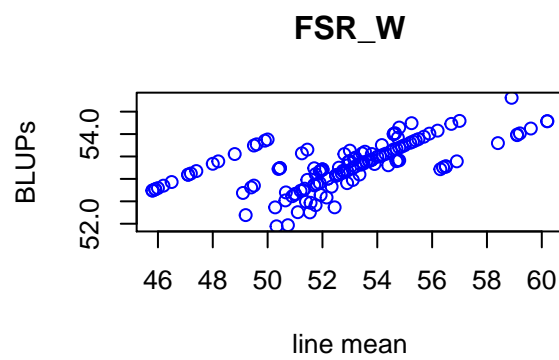
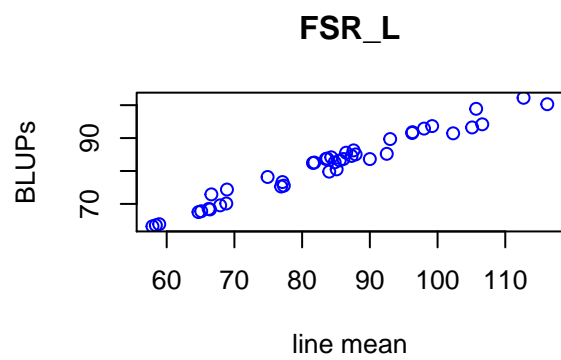
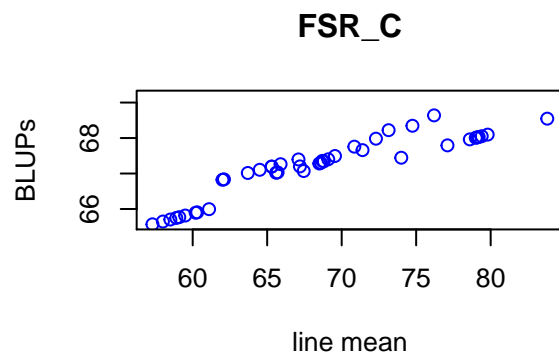
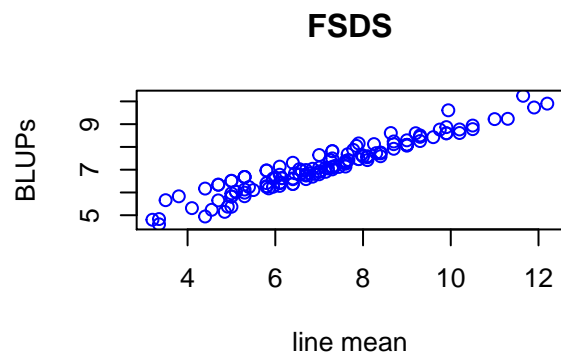


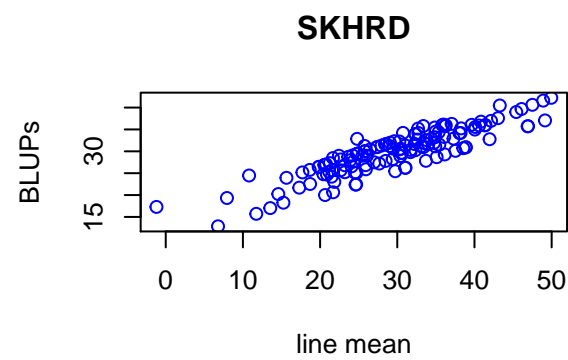
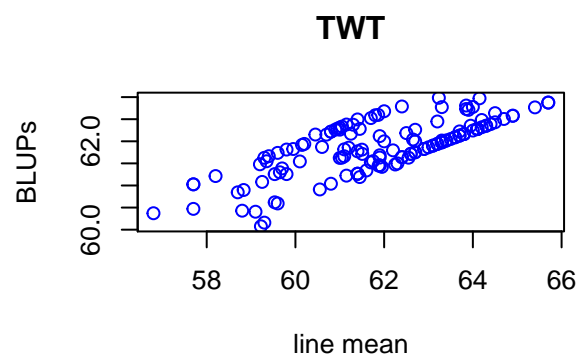
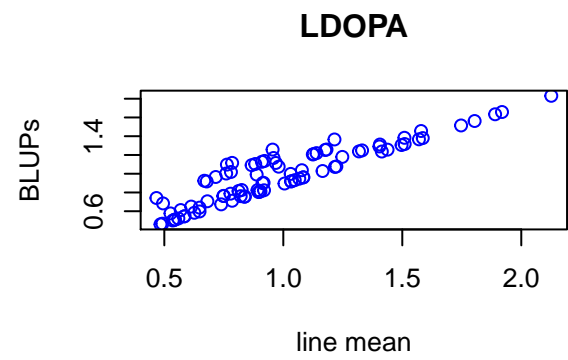
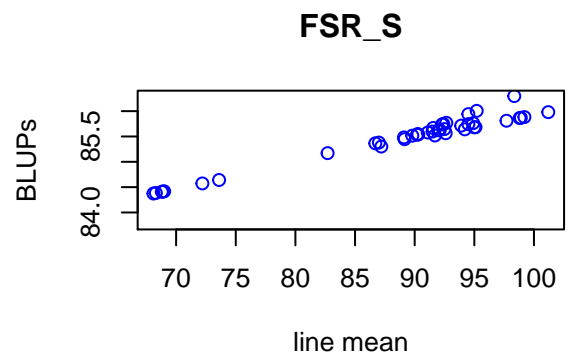


Compare BLUPs with line averages

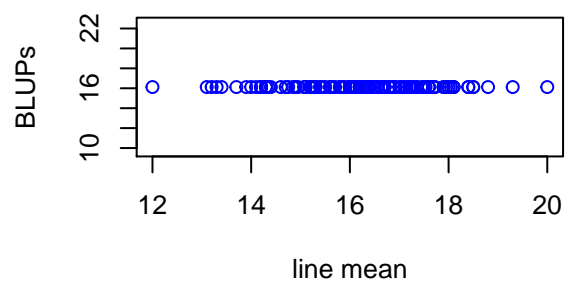
```
## Compare BLUP to line averages on a scatterplot
par(mfrow=c(2,2))
for(i in 2:length(qnamet)){
  j=qnamet[,c(1,i)]
  k=BLUEz1[,c(1,i)]
  plot(j[,2], k[,2], col="blue",main=names(qnamet)[i],ylab="BLUPs",xlab="line mean")
}
```



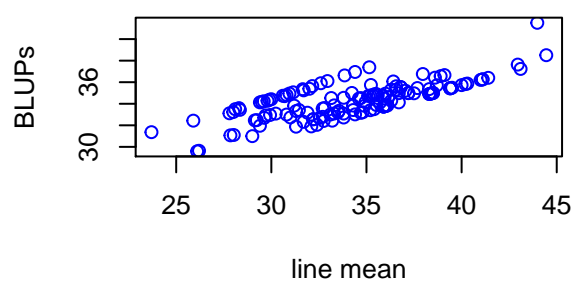




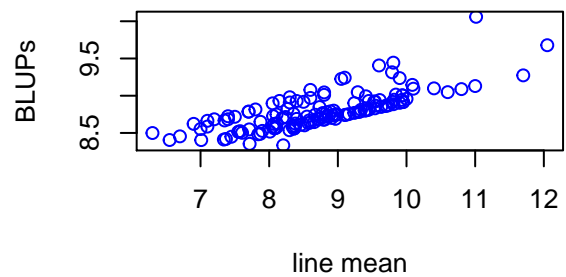
SKHRDSD



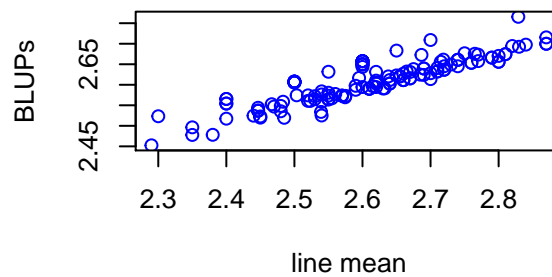
SKWT



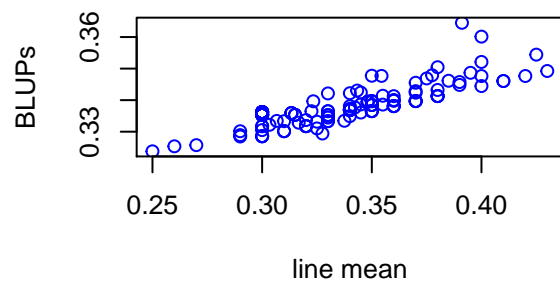
SKWTSD



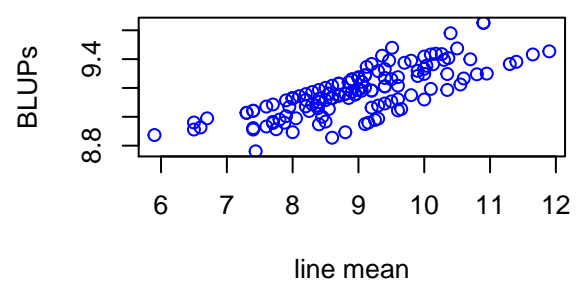
SKSIZE



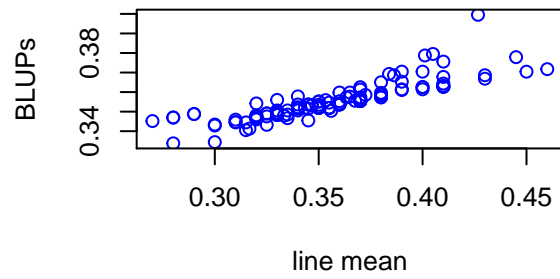
SKSIZESD



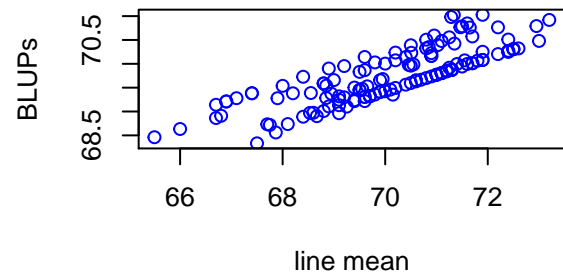
FPROT

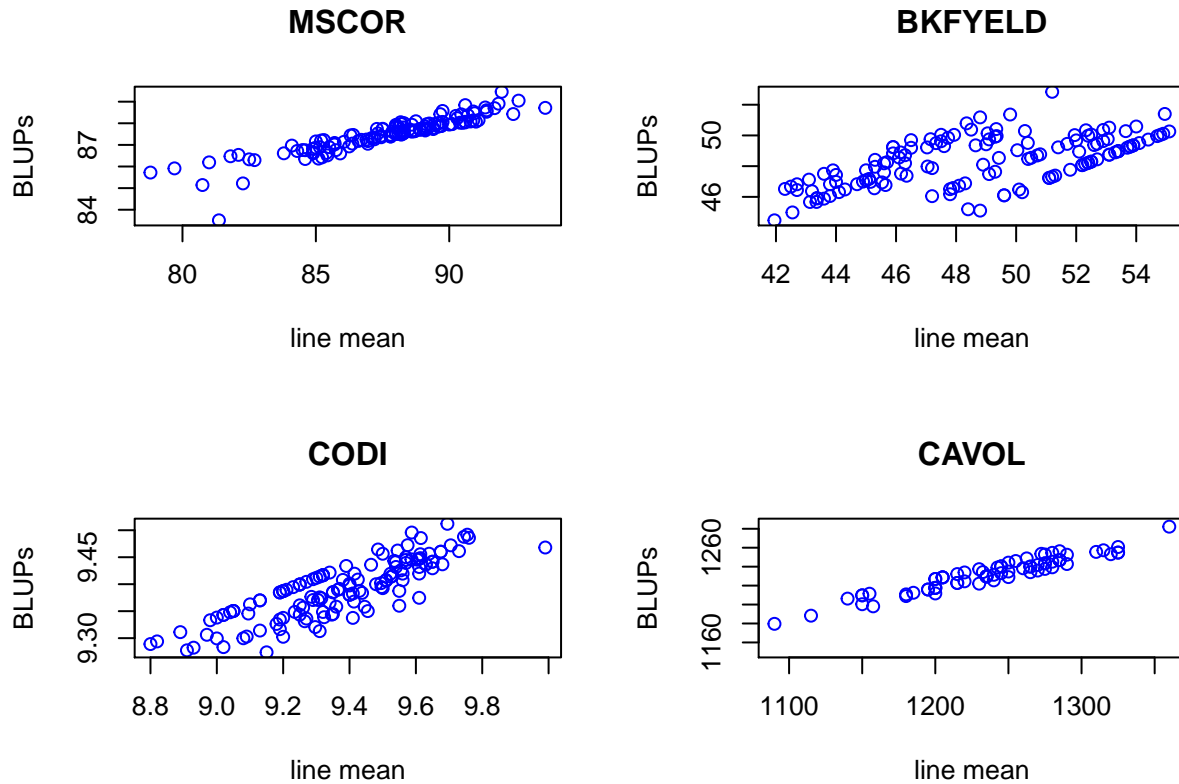


FASH



FYIELD





You can see that a normal mixed linear model may not be the correct model for each traits. Traits such as mmicrosd clearly do not follow a normal distribution. Therefore, we can try genearlized models or generalized mixed linear models using glm and glmer, respectively. Also, you can use additive spline models,or GEE models to account for non-normal distribution. Also, using a log distribution or transformation may help. You may also use non-parametric regression models such as principal component regression or partial least squares.

Generalized Linear Mixed Model

```
library(glmmTMB)
m2 <- glmmTMB(I(microsd+10)~(1|NAMET)+(1|LOCATION)+(1|year0)+(1|NAMET:LOCATION)+(1|NAMET:year0), family=
summary(m2)

## Family: nbinom2 ( log )
## Formula:
## I(microsd + 10) ~ (1 | NAMET) + (1 | LOCATION) + (1 | year0) +
## (1 | NAMET:LOCATION) + (1 | NAMET:year0)
## Data: qdata
##
##      AIC      BIC    logLik deviance df.resid
##      NA      NA      NA      NA      110
##
## Random effects:
##
## Conditional model:
## Groups      Name      Variance Std.Dev.
```

```
## NAMET (Intercept) 2.087e-11 4.568e-06
## LOCATION (Intercept) 6.929e-24 2.632e-12
## year0 (Intercept) 1.530e-29 3.911e-15
## NAMET:LOCATION (Intercept) 4.490e-11 6.701e-06
## NAMET:year0 (Intercept) 3.018e-11 5.493e-06
## Number of obs: 117, groups:
## NAMET, 74; LOCATION, 3; year0, 2; NAMET:LOCATION, 92; NAMET:year0, 95
##
## Overdispersion parameter for nbinom2 family (): 1.28e+08
##
## Conditional model:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) 2.416 NA NA NA

#z=m2
#rr1 <- ranef(z)
#mu=fixef(z)[[1]]
#rr1$zi
#resline=rr1['NAMET']
#resline
#LINEBLUP = resline[,1]
#BLUPs=c()
#x=resline[j,1]
#y=x+mu
#BLUPs=c(BLUPs,y)
```

Model Selection (microsd)

```
library(glmulti)
srt.model <-
  glmulti(microsd ~ NAMET+LOCATION+year0,
    data = qdata,
    level = 2, #interactions considered
    method = "h", #Exhaustive approach
    crit = "aic", #AIC as criteria
    confsetsize = 100, # Keep 100 best models
    plotty = T,
    report = T, #plot or interim reports
    fitfunction = "glm")

## Initialization...
## TASK: Exhaustive screening of candidate set.
## Fitting...
## Completed.

AIC <- rep(0, length(srt.model@formulas))
MODEL <- rep(NA, length(srt.model@formulas))
AUC <- rep(0, length(srt.model@formulas))
RSQUARED <- rep(0, length(srt.model@formulas))
for(i in 1:length(srt.model@formulas)){
  fit <- glm(paste(as.character(srt.model@formulas[i])), data = qdata)
  MODEL[i] <- paste(as.character(srt.model@formulas[i]))
  AIC[i] <- fit$aic
  predictpr <- predict(fit, type = "response")
  #ROC <- pROC::roc(qdata$microsd ~ predictpr)
```

```

#AUC[i] <- pROC::auc(ROC)
RSQUARED[i] <- 1 - (fit$deviance/fit$null.deviance)
}
INDEX <- seq(1:length(srt.model@formulas))
srt.model.fits <- data.frame(INDEX, MODEL, AIC, RSQUARED, AUC)
srt.model.fits$MODEL <- as.character(srt.model.fits$MODEL)
srt.model.fits$AIC <- as.numeric(srt.model.fits$AIC)
srt.model.fits$RSQUARED <- as.numeric(srt.model.fits$RSQUARED)
srt.model.fits$AUC <- as.numeric(srt.model.fits$AUC)
srt.model.fits

```

```

##      INDEX
## 1         1
## 2         2
## 3         3
## 4         4
## 5         5
## 6         6
## 7         7
## 8         8
## 9         9
## 10        10
## 11        11
## 12        12
## 13        13
## 14        14
## 15        15
## 16        16
## 17        17
## 18        18
##
##                                     MODEL
## 1  microsd ~ 1 + NAMET + LOCATION + year0 + LOCATION:NAMET + year0:NAMET + year0:LOCATION
## 2                microsd ~ 1 + NAMET + LOCATION + year0 + LOCATION:NAMET + year0:NAMET
## 3                microsd ~ 1 + NAMET + LOCATION + year0 + year0:NAMET + year0:LOCATION
## 4                        microsd ~ 1 + NAMET + LOCATION + year0 + year0:NAMET
## 5                                microsd ~ 1 + NAMET + year0 + year0:NAMET
## 6                microsd ~ 1 + NAMET + LOCATION + year0 + LOCATION:NAMET + year0:LOCATION
## 7                        microsd ~ 1 + NAMET + LOCATION + year0 + year0:LOCATION
## 8                                microsd ~ 1 + NAMET + LOCATION + year0
## 9                microsd ~ 1 + NAMET + LOCATION + year0 + LOCATION:NAMET
## 10                        microsd ~ 1 + NAMET + LOCATION
## 11                microsd ~ 1 + NAMET + LOCATION + LOCATION:NAMET
## 12                        microsd ~ 1 + NAMET + year0
## 13                                microsd ~ 1 + NAMET
## 14                        microsd ~ 1 + LOCATION
## 15                                microsd ~ 1 + LOCATION + year0
## 16                microsd ~ 1 + LOCATION + year0 + year0:LOCATION
## 17                        microsd ~ 1
## 18                                microsd ~ 1 + year0
##
##      AIC      RSQUARED  AUC
## 1 -411.8003 0.9992014353  0
## 2 -400.7307 0.9991070575  0
## 3 -349.4100 0.9985174243  0
## 4 -338.8804 0.9983498413  0

```

```
## 5 -336.7252 0.9982607034 0
## 6 160.8530 0.8756152276 0
## 7 170.9827 0.8217009328 0
## 8 171.6085 0.8176541180 0
## 9 173.3185 0.8592456360 0
## 10 184.1595 0.7935058007 0
## 11 188.5009 0.8369797640 0
## 12 188.8998 0.7812604383 0
## 13 191.0272 0.7734061425 0
## 14 211.5969 0.0907287674 0
## 15 213.5952 0.0907420653 0
## 16 214.1919 0.1015820145 0
## 17 218.7250 0.0000000000 0
## 18 220.6559 0.0005897104 0
```

Model Selection (TWT)

```
library(glmulti)
srt.model <-
  glmulti(TWT ~ NAMET+LOCATION+year0,
    data = qdata,
    level = 2,           #interactions considered
    method = "h",        #Exhaustive approach
    crit = "aic",        #AIC as criteria
    confsetsize = 100,   # Keep 100 best models
    plotty = T,
    report = T,          #plot or interim reports
    fitfunction = "glm")

## Initialization...
## TASK: Exhaustive screening of candidate set.
## Fitting...
## Completed.

AIC <- rep(0, length(srt.model@formulas))
MODEL <- rep(NA, length(srt.model@formulas))
AUC <- rep(0, length(srt.model@formulas))
RSQUARED <- rep(0, length(srt.model@formulas))
for(i in 1:length(srt.model@formulas)){
  fit <- glm(paste(as.character(srt.model@formulas[i])), data = qdata)
  MODEL[i] <- paste(as.character(srt.model@formulas[i]))
  AIC[i] <- fit$aic
  predictpr <- predict(fit, type = "response")
  #ROC <- pROC::roc(qdata$microsd ~ predictpr)
  #AUC[i] <- pROC::auc(ROC)
  RSQUARED[i] <- 1 - (fit$deviance/fit$null.deviance)
}
INDEX <- seq(1:length(srt.model@formulas))
srt.model.fits <- data.frame(INDEX, MODEL, AIC, RSQUARED, AUC)
srt.model.fits$MODEL <- as.character(srt.model.fits$MODEL)
srt.model.fits$AIC <- as.numeric(srt.model.fits$AIC)
srt.model.fits$RSQUARED <- as.numeric(srt.model.fits$RSQUARED)
srt.model.fits$AUC <- as.numeric(srt.model.fits$AUC)
srt.model.fits
```

##	INDEX
## 1	1
## 2	2
## 3	3
## 4	4
## 5	5
## 6	6
## 7	7
## 8	8
## 9	9
## 10	10
## 11	11
## 12	12
## 13	13
## 14	14
## 15	15
## 16	16
## 17	17
## 18	18

##	MODEL
## 1	TWT ~ 1 + NAMET + LOCATION + year0 + LOCATION:NAMET + year0:NAMET + year0:LOCATION
## 2	TWT ~ 1 + NAMET + LOCATION + year0 + year0:NAMET + year0:LOCATION
## 3	TWT ~ 1 + NAMET + LOCATION + year0 + LOCATION:NAMET + year0:NAMET
## 4	TWT ~ 1 + NAMET + LOCATION + year0 + LOCATION:NAMET + year0:LOCATION
## 5	TWT ~ 1 + NAMET + LOCATION + year0 + year0:NAMET
## 6	TWT ~ 1 + NAMET + year0 + year0:NAMET
## 7	TWT ~ 1 + NAMET + LOCATION + year0 + year0:LOCATION
## 8	TWT ~ 1 + NAMET + LOCATION + year0 + LOCATION:NAMET
## 9	TWT ~ 1 + NAMET + LOCATION + year0
## 10	TWT ~ 1 + NAMET + year0
## 11	TWT ~ 1 + NAMET + LOCATION + LOCATION:NAMET
## 12	TWT ~ 1 + LOCATION + year0 + year0:LOCATION
## 13	TWT ~ 1 + NAMET + LOCATION
## 14	TWT ~ 1 + LOCATION + year0
## 15	TWT ~ 1 + year0
## 16	TWT ~ 1 + NAMET
## 17	TWT ~ 1 + LOCATION
## 18	TWT ~ 1

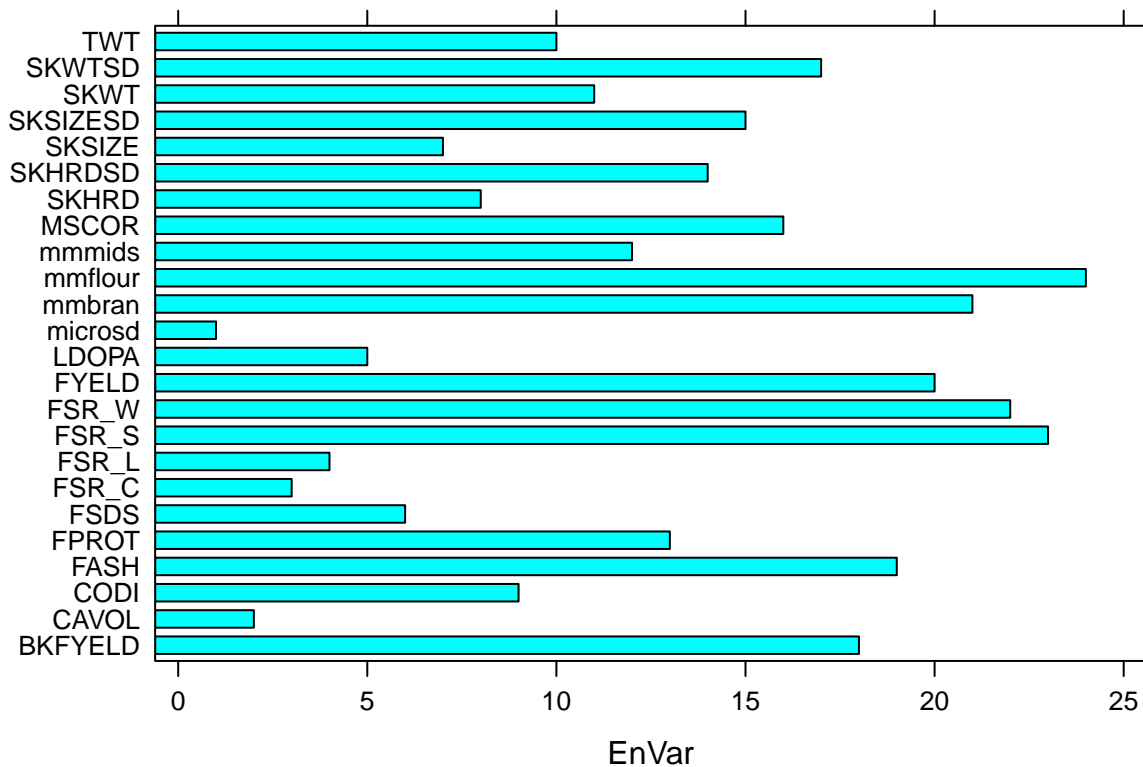
##	AIC	RSQUARED	AUC
## 1	407.5452	0.9889426	0
## 2	588.3793	0.9707662	0
## 3	600.6091	0.9762658	0
## 4	689.1073	0.9594015	0
## 5	726.1749	0.9489567	0
## 6	734.3794	0.9460676	0
## 7	772.2164	0.9079821	0
## 8	838.8734	0.9257472	0
## 9	888.9204	0.8519456	0
## 10	895.7806	0.8443787	0
## 11	905.9628	0.9014300	0
## 12	917.7895	0.5321091	0
## 13	940.5321	0.8149029	0
## 14	960.9101	0.4339825	0
## 15	971.3476	0.3967473	0


```
## 16 994.0974 0.7668332 0
## 17 1016.9744 0.2800481 0
## 18 1095.7457 0.0000000 0
```

How does the variation due to environment differ among the traits?

All Environmental and Residual variation for each trait

```
#Extract all environmental variation
envar=c()
for(i in 1:length(models41)){
  x=models41[[i]]
  x=((as.data.frame(VarCorr(x))[,c(1,4)][4,2])+
    (as.data.frame(VarCorr(x))[,c(1,4)][5,2])+
    (as.data.frame(VarCorr(x))[,c(1,4)][6,2]))/
    sum((as.data.frame(VarCorr(x))[,c(1,4)][,2]))*100
  envar=c(envar,x)
}
envar=cbind(names(models41),round(envar,5))
envar=as.data.frame(envar)
names(envar)<-c("Trait", "EnVar")
barchart(Trait~EnVar,data=envar)
```



```
envar
```

```
##      Trait      EnVar
## 1  mmbran 84.08524
```

```

## 2    mmflour  96.3896
## 3      mmmids 66.40228
## 4    microsd  0.63005
## 5      FSDS  40.42003
## 6      FSR_C 27.44757
## 7      FSR_L 32.79525
## 8      FSR_W 89.51964
## 9      FSR_S 94.91282
## 10    LDOPA  34.89465
## 11      TWT   65.6657
## 12    SKHRD  55.10161
## 13  SKHRDSD 70.64851
## 14      SKWT  66.10717
## 15    SKWTSD 75.58521
## 16    SKSIZE 52.39343
## 17  SKSIZESD 72.12458
## 18    FPROT  69.48906
## 19      FASH 79.38474
## 20    FYELD  82.20365
## 21    MSCOR  73.62557
## 22  BKFYELD  79.1029
## 23      CODI 61.57348
## 24    CAVOL  19.92223

```

Mean-line basis and Environmental Variation

hert11

```

##      Trait Heritability
## 1    mmbran    78.65814
## 2    mmflour    7.54919
## 3      mmmids  55.56241
## 4    microsd  47.24858
## 5      FSDS   92.24167
## 6      FSR_C   36.36493
## 7      FSR_L   94.29544
## 8      FSR_W   75.55939
## 9      FSR_S   42.59276
## 10    LDOPA   96.70367
## 11      TWT   78.90226
## 12    SKHRD   90.29002
## 13  SKHRDSD    0.00000
## 14      SKWT   87.88917
## 15    SKWTSD   72.06066
## 16    SKSIZE   84.74199
## 17  SKSIZESD   62.57208
## 18    FPROT   51.35112
## 19      FASH   73.70655
## 20    FYELD   89.02749
## 21    MSCOR   76.04960
## 22  BKFYELD   97.69701
## 23      CODI   61.94515
## 24    CAVOL   67.83120

```

```

hert11$Heritability=as.numeric(as.character(hert11$Heritability))
envdif=c()
for(i in 1:length(models41)){
  y=hert11$Heritability[i]
  z=100-y

  envdif=c(envdif,z)
}
hert21=cbind(hert11,envvar[,2],envdif)
names(hert21)<-c("Trait","Mean-line basis-Year0","Env Variation","Difference Env")
hert21

```

##	Trait	Mean-line basis-Year0	Env Variation	Difference Env
## 1	mmbran	78.65814	84.08524	21.34186
## 2	mmflour	7.54919	96.3896	92.45081
## 3	mmmids	55.56241	66.40228	44.43759
## 4	microsd	47.24858	0.63005	52.75142
## 5	FSDS	92.24167	40.42003	7.75833
## 6	FSR_C	36.36493	27.44757	63.63507
## 7	FSR_L	94.29544	32.79525	5.70456
## 8	FSR_W	75.55939	89.51964	24.44061
## 9	FSR_S	42.59276	94.91282	57.40724
## 10	LDOPA	96.70367	34.89465	3.29633
## 11	TWT	78.90226	65.6657	21.09774
## 12	SKHRD	90.29002	55.10161	9.70998
## 13	SKHRDSD	0.00000	70.64851	100.00000
## 14	SKWT	87.88917	66.10717	12.11083
## 15	SKWTSD	72.06066	75.58521	27.93934
## 16	SKSIZE	84.74199	52.39343	15.25801
## 17	SKSIZESD	62.57208	72.12458	37.42792
## 18	FPROT	51.35112	69.48906	48.64888
## 19	FASH	73.70655	79.38474	26.29345
## 20	FYELD	89.02749	82.20365	10.97251
## 21	MSCOR	76.04960	73.62557	23.95040
## 22	BKFYELD	97.69701	79.1029	2.30299
## 23	CODI	61.94515	61.57348	38.05485
## 24	CAVOL	67.83120	19.92223	32.16880

AMMI

It is only calculatable for data without missing values. Test Weight was a try, but not the best method for unbalanced data.

```

#AMMI is one of my favorite ways to display GEI but this function through agricolae only can display ba
library(agricolae)
model<- AMMI(qdata$LOCATION, qdata$NAMET, qdata$year0, qdata$TWT, console=FALSE)
model$ANOVA
#model$genXenv
#model$means
#model$biplot
#summary(model)
plot(model,main= "Biplot for TWT")
plot(model, type=2,main= "Triplot for TWT")
plot(model, first=0,second=1,main= "PC1 vs Yield for TWT")
plot(model,main= "Biplot Contour for TWT")

```

```
AMMI.contour(model,distance=0.7,shape=8,col="red",lwd=2,lty=5)
Idxbu=index.AMMI(model)
print(Idxbu[order(Idxbu[,3]),])
print(Idxbu[order(Idxbu[,4]),])
```

```
###Principal Components
```

#Principal components has to use only balanced data with no missing values. Therefore I used the means

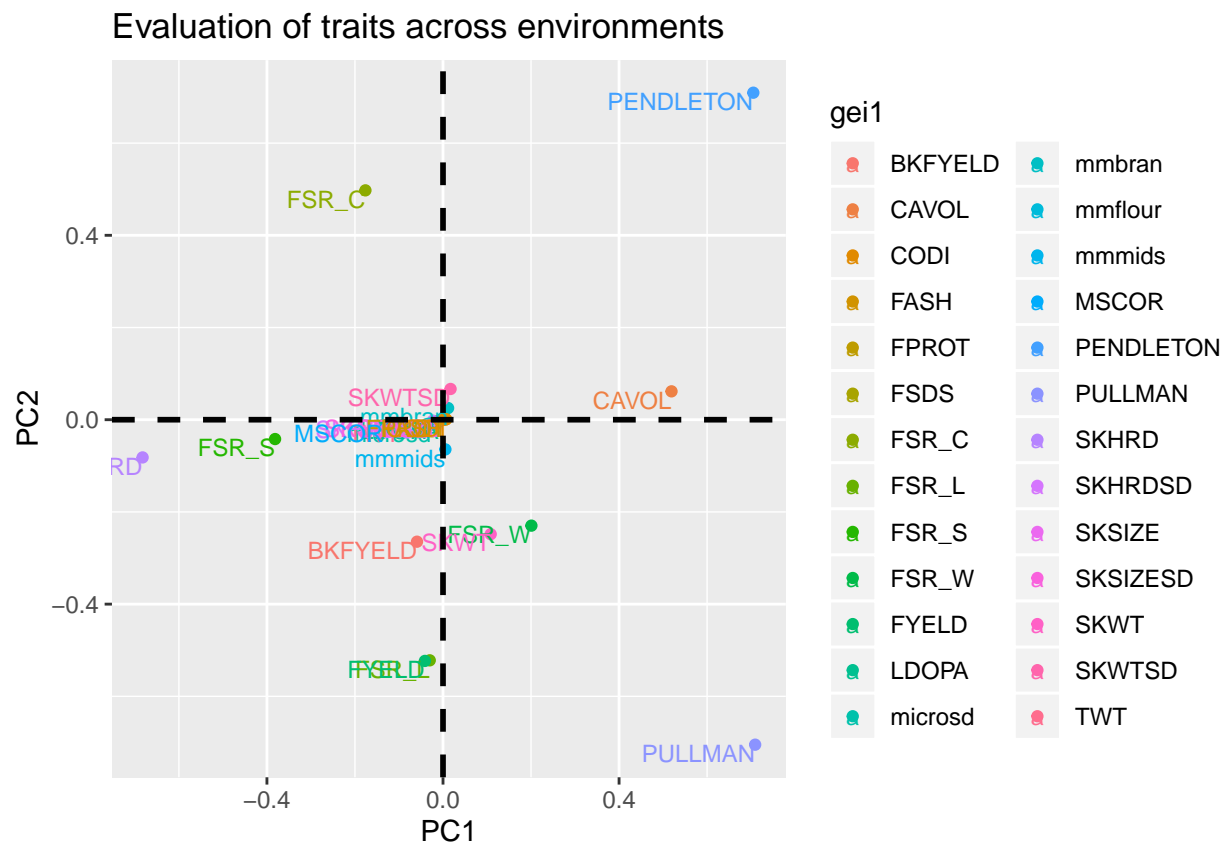
```
set.seed(702)
qdhc=aggregate(qdata[,12:35],list(qdata$LOCATION),mean,na.rm=TRUE)
qdhne=qdhc[,-1]
rownames(qdhne)=qdhc[,1]
qdhne2=qdhne[complete.cases(qdhne),]
```

```
pr.out1e <- prcomp(qdhne2)
prout1e=as.data.frame(pr.out1e$rotation)
pnam1e=rownames(prout1e)
```

```
pr.out1e1 <- prcomp(t(qdhne2))
prout1e1=as.data.frame(pr.out1e1$rotation)
pnam1e1=rownames(prout1e1)
```

```
gei=rbind(prout1e,prout1e1)
gei1=rownames(gei)
```

```
ggplot(gei,aes(x=gei$PC1,y=gei$PC2,col=gei1))+geom_point(position="jitter")+geom_text(aes(label=gei1),s
```



Are there any pedigrees that perform particularly well or particularly bad? Any clusters?

This section was focused on clustering and principal components. Good way to visualize overall correlations.

```
qdh=aggregate(qdata[,12:35],list(qdata$pedigree),mean,na.rm=TRUE)
qdh1=qdh[,,-1]
rownames(qdh1)=qdh[,1]

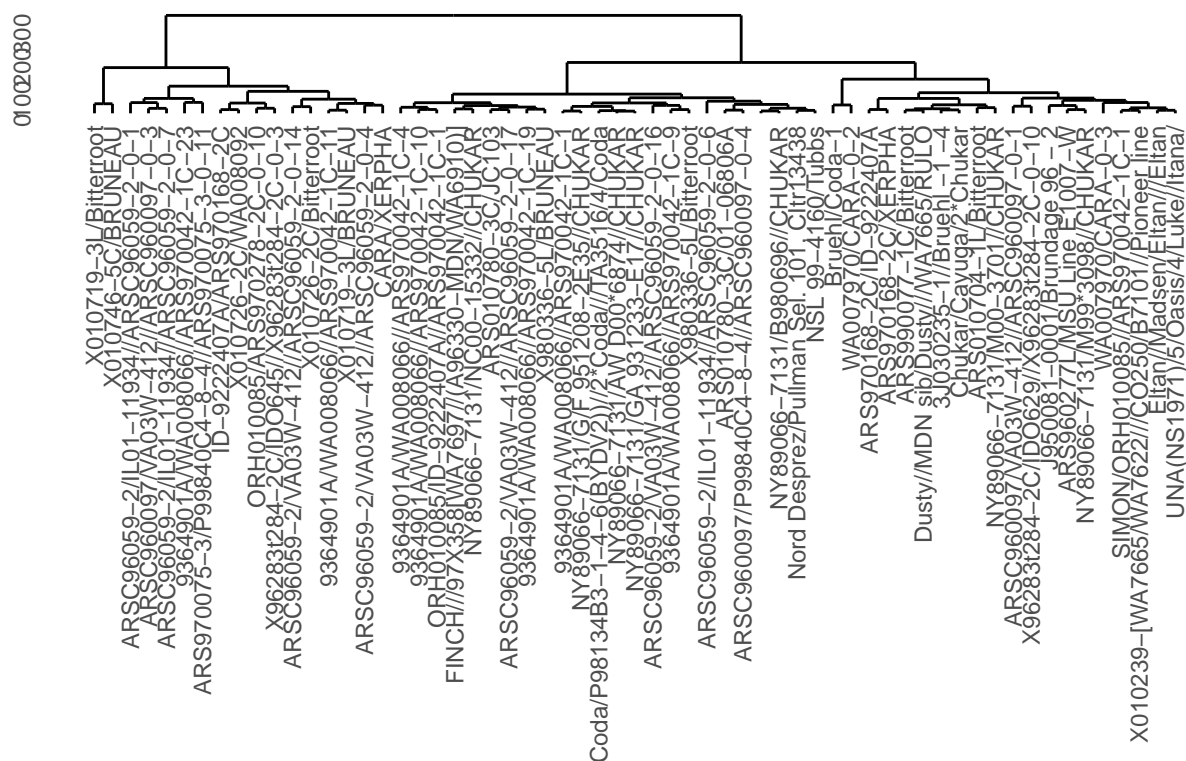
qdh1=aggregate(qdata[,12:35],list(qdata$NAMET),mean,na.rm=TRUE)
qdh1=qdh1[,,-1]
rownames(qdh1)=qdh1[,1]
```

Using hierarchical clustering with complete linkage and Euclidean distance for Pedigrees.

Complete Clustering

```
library(ggdendro)
library(ggplot2)
set.seed(702)
hc.complete <- hclust(dist(qdh1), method = "complete")
ggdendrogram(hc.complete)+labs(title="Complete Cluster Dendrogram")
```

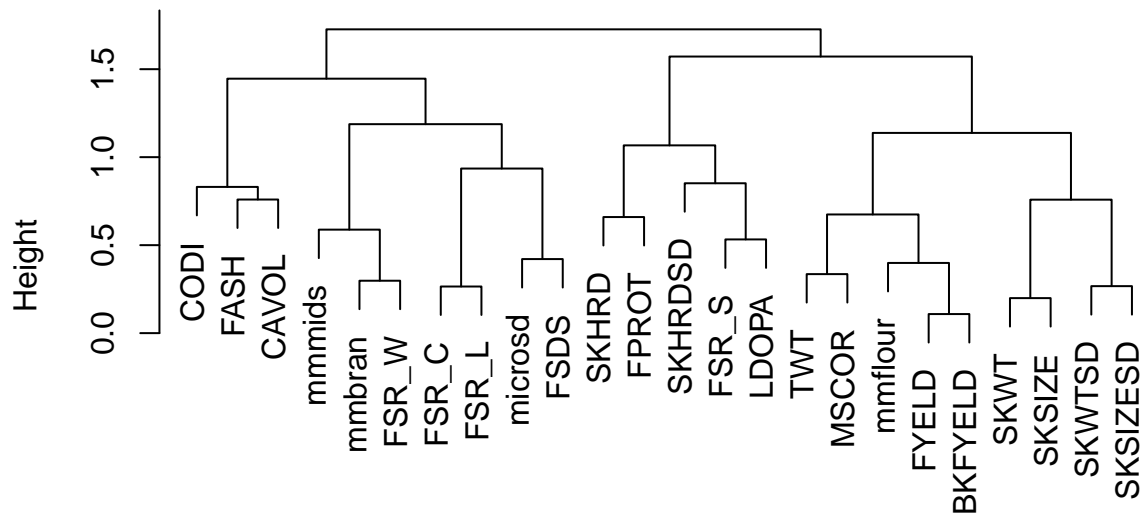
Complete Cluster Dendrogram



Complete Clustering with Scaling

```
set.seed(702)
sd.data <- scale(qdh1)
```


Complete Cluster Dendrogram

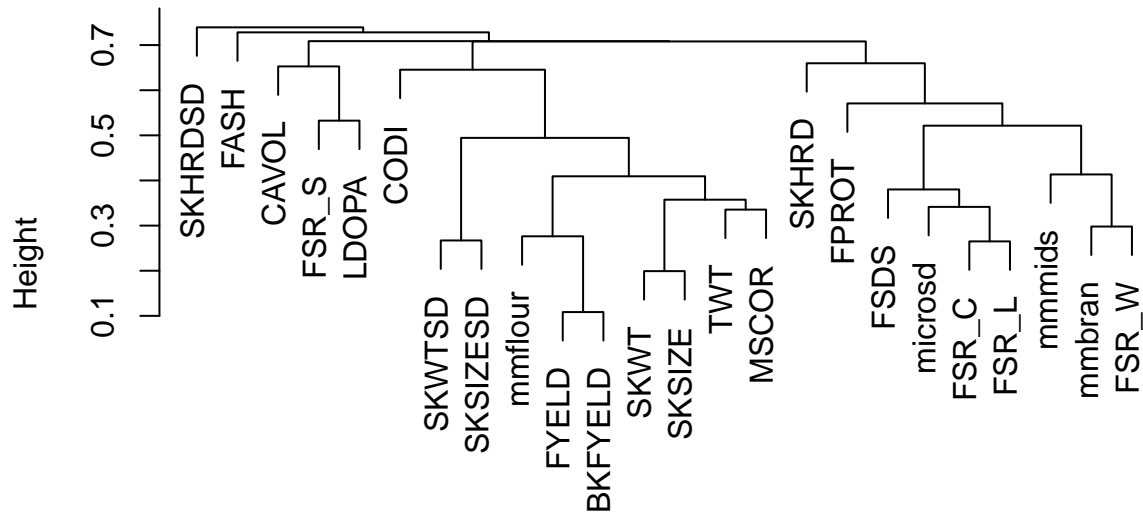


```
as.dist(1 - cor(qdhn2))
hclust (*, "complete")
```

Single Clustering

```
set.seed(702)
hc.single <- hclust(as.dist(1 - cor(qdhn2)), method = "single")
plot(hc.single, main="Single Cluster Dendrogram")
```

Single Cluster Dendrogram

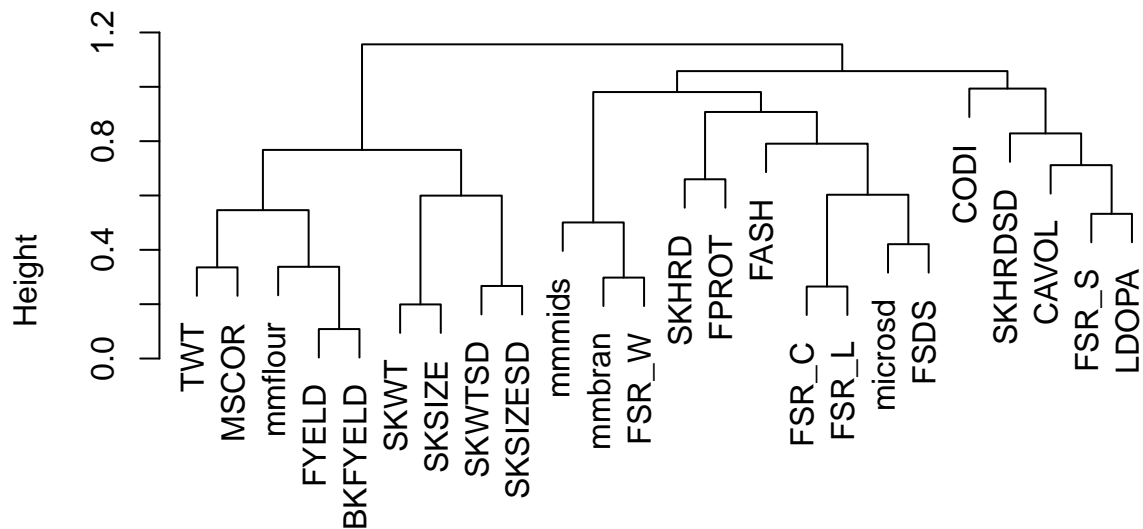


```
as.dist(1 - cor(qdhn2))
hclust (*, "single")
```

Average Clustering

```
set.seed(702)
hc.average <- hclust(as.dist(1 - cor(qdhn2)), method = "average")
plot(hc.average, main="Average Cluster Dendrogram")
```


Average Cluster Dendrogram



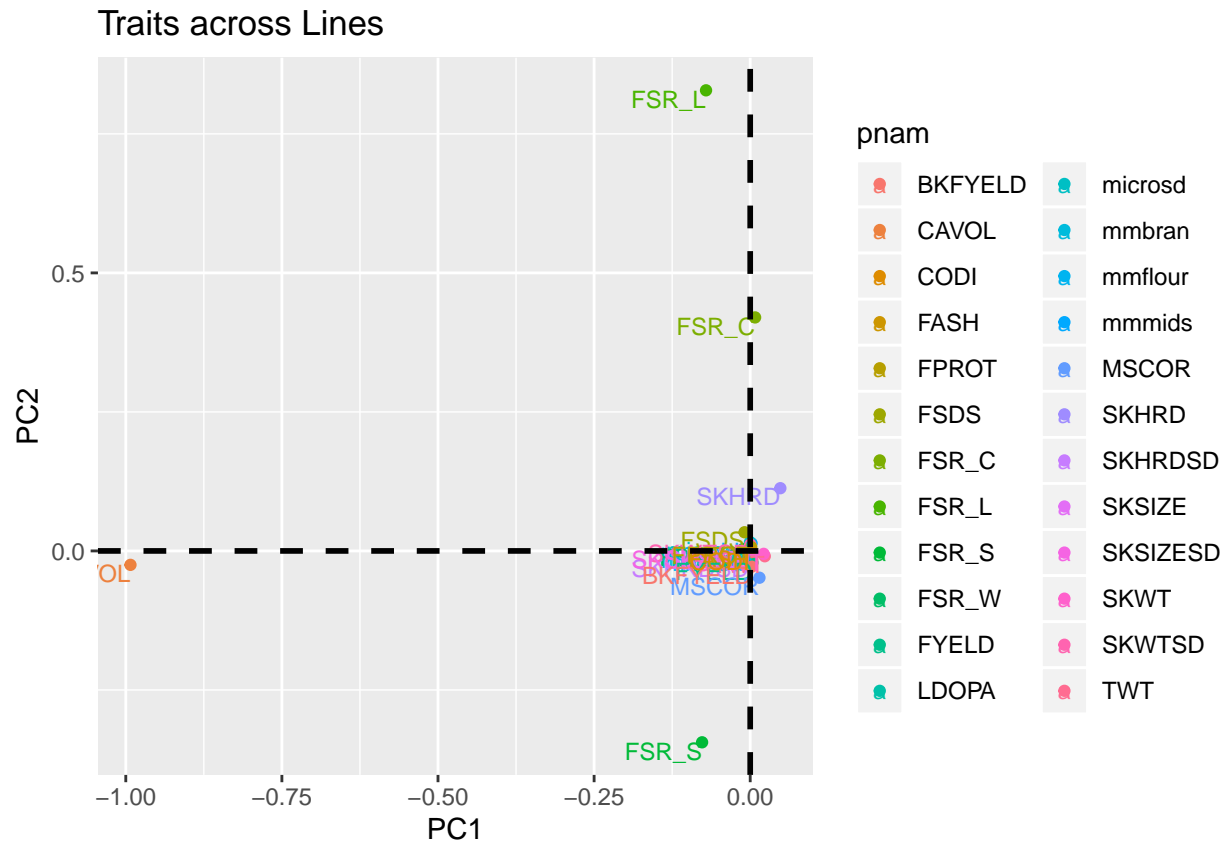
```
as.dist(1 - cor(qdhn2))
hclust (*, "average")
```

Principal Components

Raw clustering

```
set.seed(702)
pr.out <- prcomp(qdhn2)
prout=as.data.frame(pr.out$rotation)
pnam=row.names(prout)
```

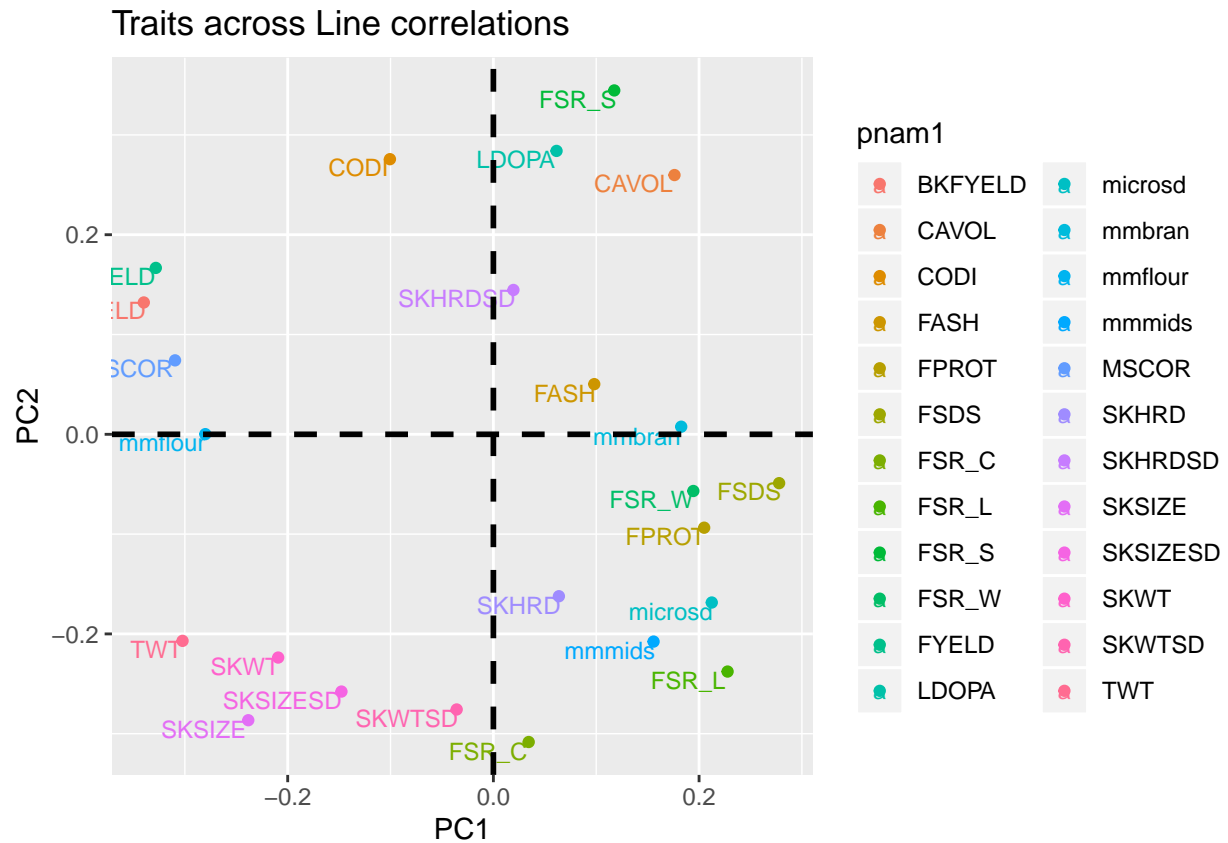
```
ggplot(prout,aes(x=prout$PC1,y=prout$PC2,col=pnam))+geom_point(position="jitter")+geom_text(aes(label=pnam))
```



We'll try again but with Correlations.

```
set.seed(702)
pr.out1 <- prcomp(1-cor(qdhn2))
prout1=as.data.frame(pr.out1$rotation)
pnam1=rownames(prout1)

ggplot(prout1,aes(x=prout1$PC1,y=prout1$PC2,col=pnam1))+geom_point(position="jitter")+geom_text(aes(lab=
```

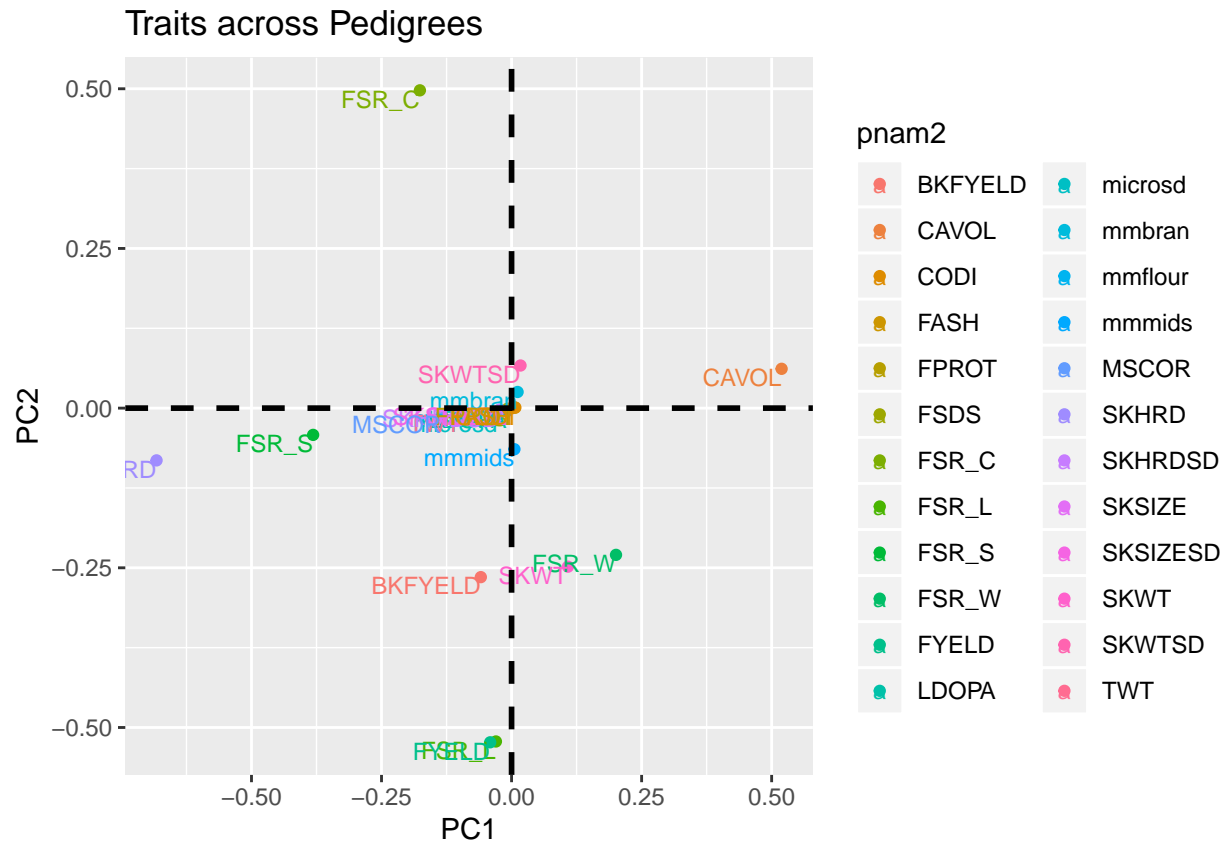


Try with Pedigree

```
set.seed(702)
qdh3=aggregate(qdata[,12:35],list(qdata$LOCATION),mean,na.rm=TRUE)
qdh3=qdh3[, -1]
rownames(qdh3)=qdh3[,1]
qdh3e=qdh3[complete.cases(qdh3),]

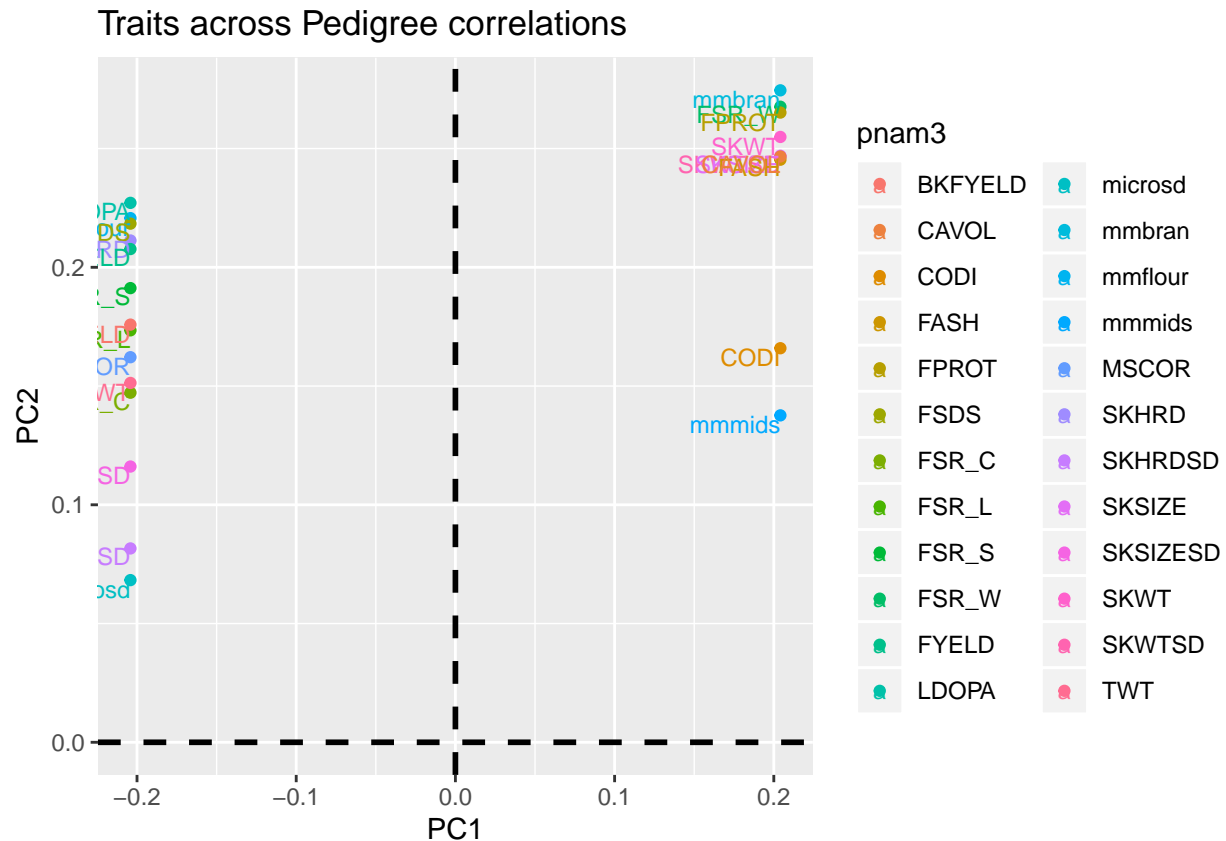
pr.out2 <- prcomp(qdh3e)
prout2=as.data.frame(pr.out2$rotation)
pnam2=rownames(prout2)

ggplot(prout2,aes(x=prout2$PC1,y=prout2$PC2,col=pnam2))+geom_point(position="jitter")+geom_text(aes(label=pnam2))
```



```
pr.out3 <- prcomp(1-cor(qdhn3e))
prout3=as.data.frame(pr.out3$rotation)
pnam3=rownames(prout3)

ggplot(prout3,aes(x=prout3$PC1,y=prout3$PC2,col=pnam3))+geom_point(position="jitter")+geom_text(aes(label=
```



Differing Traits

```
set.seed(702)
total.load <- apply(pr.out$rotation, 1, sum)
index <- order(abs(total.load), decreasing = TRUE)
total.load[index[1:10]]
```

```
##      FPROT      MSCOR      FSR_L      microsd      SKWTSD      CODI      CAVOL
## -2.001679 -1.567846  1.555306 -1.489243  1.265369 -1.171405 -1.150147
##      FSR_W      SKSIZE      BKFYELD
## -1.132228  1.025835  1.007656
```

These are the 10 most different traits using principal components.

What should I select?

A good club wheat has high cake volume, low gluten strength, high test weight, high break flour and high flour yield. Our stakeholders have asked me to increase kernel size and test weight and reduce LDOPA as well. Can you make any suggestions for crosses that I might want to make?

Lets compare BLUPs

We want to focus on cake volume (CAVOL), Gluten Strength (Microsd, FSR-L), Test weight (TWT), High Break flour (BKFYELD), kernel size (SKSIZE), LDOPA (LDOPA).

```

bp=BLUEz[,c(1:4,8,11,14,15,20,24,26,28)]
bp=bp %>% mutate(msd_rank = rank(bp$microsd, ties.method = 'first'))
bp=bp %>% mutate(fsrl_rank = rank(bp$FSR_L, ties.method = 'first'))
bp=bp %>% mutate(ldopa_rank = rank(bp$LDOPA, ties.method = 'first'))
bp=bp %>% mutate(twt_rank = dense_rank(desc(bp$TWT)))
bp=bp %>% mutate(sksize_rank = dense_rank(desc(bp$SKSIZE)))
bp=bp %>% mutate(fy_rank = dense_rank(desc(bp$FYELD)))
bp=bp %>% mutate(bkfy_rank = dense_rank(desc(bp$BKFYELD)))
bp=bp %>% mutate(cavol_rank = dense_rank(desc(bp$CAVOL)))
bp$avg=rowMeans(bp[,13:20])
print(head(bp[order(bp[,21]),]))
rnks=bp[,c(1,13:21)]
print(head(rnks[order(rnks[,10]),]))

```

Ranks of Lines

```

bp1=BLUEz[,c(1:4,8,11,14,15,20,24,26,28)]
bp2=aggregate(bp1[,5:12],list(bp1$NAMET),mean)
bp2=bp2 %>% mutate(msd_rank = rank(bp2$microsd, ties.method = 'first')) #For traits with low as better
bp2=bp2 %>% mutate(fsrl_rank = rank(bp2$FSR_L, ties.method = 'first'))
bp2=bp2 %>% mutate(ldopa_rank = rank(bp2$LDOPA, ties.method = 'first')) #For traits with high as better
bp2=bp2 %>% mutate(twt_rank = dense_rank(desc(bp2$TWT)))
bp2=bp2 %>% mutate(sksize_rank = dense_rank(desc(bp2$SKSIZE)))
bp2=bp2 %>% mutate(fy_rank = dense_rank(desc(bp2$FYELD)))
bp2=bp2 %>% mutate(bkfy_rank = dense_rank(desc(bp2$BKFYELD)))
bp2=bp2 %>% mutate(cavol_rank = dense_rank(desc(bp2$CAVOL)))
bp2$avg=rowMeans(bp2[,10:17])
#print(head(bp2[order(bp2[,18]),]))
rnks2=bp2[,c(1,10:18)]
print(head(rnks2[order(rnks2[,10]),]))

```

##	Group.1	msd_rank	fsrl_rank	ldopa_rank	twt_rank	sksize_rank
## 12	09X492-0-0-6*CBW	4	10	3	94	9
## 17	09X500-0-0-17*CBW	30	7	7	20	43
## 9	09X488-0-0-3*CBW	11	42	39	15	6
## 15	09X500-0-0-14*CBW	35	11	4	12	43
## 8	09X488-0-0-1*CBW	7	35	10	39	48
## 27	10X022-0-9*CBW	27	15	23	55	98
##		fy_rank	bkfy_rank	cavol_rank	avg	
## 12		4	2	36	20.250	
## 17		37	33	32	26.125	
## 9		16	7	76	26.500	
## 15		27	27	72	28.875	
## 8		43	53	13	31.000	
## 27		14	6	18	32.000	

Ranks of Pedigrees

```

bp3=aggregate(bp1[,5:12],list(bp1$pedigree),mean)
bp3=bp3 %>% mutate(msd_rank = rank(bp3$microsd, ties.method = 'first'))
bp3=bp3 %>% mutate(fsrl_rank = rank(bp3$FSR_L, ties.method = 'first'))
bp3=bp3 %>% mutate(ldopa_rank = rank(bp3$LDOPA, ties.method = 'first'))
bp3=bp3 %>% mutate(twt_rank = dense_rank(desc(bp3$TWT)))
bp3=bp3 %>% mutate(sksize_rank = dense_rank(desc(bp3$SKSIZE)))

```

```

bp3=bp3 %>% mutate(fy_rank = dense_rank(desc(bp3$FYELD)))
bp3=bp3 %>% mutate(bkfy_rank = dense_rank(desc(bp3$BKFYELD)))
bp3=bp3 %>% mutate(cavol_rank = dense_rank(desc(bp3$CAVOL)))
bp3$avg=rowMeans(bp3[,10:17])
#print(head(bp3[order(bp3[,18]),]))
rnks3=bp3[,c(1,10:18)]
print(head(rnks3[order(rnks3[,10]),]))

```

```

##                               Group.1 msd_rank fsrl_rank ldopa_rank
## 22  ARSC96059-2/IL01-11934//ARSC96059-2-0-6          4          9          2
## 26  ARSC96059-2/VA03W-412//ARSC96059-2-0-17         17          7          4
## 20    ARSC960097/VA03W-412//ARSC960097-0-3          7         21         28
## 24  ARSC96059-2/VA03W-412//ARSC96059-2-0-14         19         10          3
## 19    ARSC960097/VA03W-412//ARSC960097-0-1          6         18          7
## 58 X96283t284-2C/ID0629//X96283t284-2C-0-10         20         27         25
##   twt_rank sksize_rank fy_rank bkfy_rank cavol_rank   avg
## 22      43          5        2          1         21 10.875
## 26       8         22       17         17         19 13.875
## 20       5          3        6          5         48 15.375
## 24       4         22       10         13         46 15.875
## 19      15         26       26         27          8 16.625
## 58       9         13       28         10          6 17.250

```

Lines

Ranks for Good Club Wheat

```

bp2$cavg=rowMeans(bp2[,c(10,11,13,15,16,17)])
#print(head(bp2[order(bp2[,19]),]))
rnks4=bp2[,c(1,18:19)]
print(head(rnks4[order(rnks4[,3]),]))

```

```

##                Group.1   avg   cavg
## 62      ARS-CRESCENT 37.500 18.66667
## 27    10X022-0-9*CBW 32.000 22.50000
## 12    09X492-0-0-6*CBW 20.250 25.00000
## 23    10X022-0-11*CBW 36.125 26.00000
## 17    09X500-0-0-17*CBW 26.125 26.50000
## 9     09X488-0-0-3*CBW 26.500 27.83333

```

Ranks for Stakeholder and Parents

```

bp2$pavg=rowMeans(bp2[,c(12:14)])
#print(head(bp2[order(bp2[,20]),]))
rnks5=bp2[,c(1,18:20)]
print(head(rnks5[order(rnks5[,4]),]))

```

```

##                Group.1   avg   cavg   pavg
## 40    12X024-0-1-37-CBW 47.000 61.83333  2.666667
## 115    DH08X117-30-1 44.625 53.66667 13.000000
## 68    ARS20060126-0-13C 47.500 55.16667 16.666667
## 10    09X489-0-0-1*CBW 34.000 42.83333 17.666667
## 15    09X500-0-0-14*CBW 28.875 30.66667 19.666667
## 9     09X488-0-0-3*CBW 26.500 27.83333 20.000000

```

Pedigrees

Ranks for Good Club Wheat

```
bp3$cavg=rowMeans(bp3[,c(10,11,13,15,16,17)])  
#print(head(bp3[order(bp3[,19]),]))  
rnks4=bp3[,c(1,18:19)]  
print(head(rnks4[order(rnks4[,3]),]))
```

##		Group.1	avg	cavg
## 32	Dusty//MDN sib/Dusty//WA7665/RUL0	19.250	10.33333	
## 9	9364901A/WA008066//ARS970042-1C-9	17.625	12.16667	
## 22	ARSC96059-2/IL01-11934//ARSC96059-2-0-6	10.875	13.33333	
## 26	ARSC96059-2/VA03W-412//ARSC96059-2-0-17	13.875	14.16667	
## 5	9364901A/WA008066//ARS970042-1C-11	18.250	14.50000	
## 20	ARSC960097/VA03W-412//ARSC960097-0-3	15.375	15.33333	

Ranks for Stakeholder and Parents

```
bp3$pavg=rowMeans(bp3[,c(12:14)])  
#print(head(bp3[order(bp3[,20]),]))  
rnks5=bp3[,c(1,18:20)]  
print(head(rnks5[order(rnks5[,4]),]))
```

##		Group.1	avg	cavg	pavg
## 11	ARS010780-3C/01-06806A	23.750	29.16667	5.333333	
## 24	ARSC96059-2/VA03W-412//ARSC96059-2-0-14	15.875	17.00000	9.666667	
## 26	ARSC96059-2/VA03W-412//ARSC96059-2-0-17	13.875	14.16667	11.333333	
## 20	ARSC960097/VA03W-412//ARSC960097-0-3	15.375	15.33333	12.000000	
## 27	ARSC96059-2/VA03W-412//ARSC96059-2-0-4	22.875	24.83333	12.333333	
## 21	ARSC96059-2/IL01-11934//ARSC96059-2-0-1	22.500	28.16667	14.000000	