cor

This code creates a correlation matrix to look at relationships between variables. This helps identify specific relationships of interest

Link:

Getting Started

Libraries

Data

Cleaning

Remove non-numeric variables

```
bo<-
  bo %>%
  select(!c(date, ...1))

c2<-
  c2 %>%
  select(!c(date, ...1))

rd<-
  rd %>%
  select(!c(date, ...1))
```

Make Datasets for normal and non-normal variables

Variables that are normally distributed across all three sites will be in one df. All others will be in another.

```
#BLIND OSO
b_norm<-
  bo %>%
  select(!c(sal, secchi, nn, tdn, don, doc, toc, tn))
b_para<-
  bo %>%
  select(sal, secchi, nn, tdn, don, doc, toc, tn)

#CANALS
c_norm<-
  c2 %>%
  select(!c(sal, secchi, nn, tdn, don, doc, toc, tn))
c_para<-
  c2 %>%
  select(sal, secchi, nn, tdn, don, doc, toc, tn)

#GULF
r_norm<-
  rd %>%
  select(!c(sal, secchi, nn, tdn, don, doc, toc, tn))
r_para<-
  rd %>%
  select(sal, secchi, nn, tdn, don, doc, toc, tn)
```

Log10 Transform non-normally distributed data for Norm df

Distributions can be found in `1_distribution.qmd`

```
#BLIND OSO
b_norm<-
  b_norm %>%
  mutate(amm = log10(amm), #Log-transforms data
         din_dip = log10(din_dip),
         dust = log10(dust),
         chl = log10(chl),
         dust_lag = log10(dust_lag))
```

```r
#CANALS
c_norm<-
  c_norm %>%
  mutate(orthop = log10(orthop),
         dust = log10(dust),
         dust_lag = log10(dust_lag))


#GULF
r_norm<-
  r_norm %>%
  mutate(dust = log10(dust),
         dust_lag = log10(dust_lag))
```

Correlation Matrix

There are different methods for **correlation analysis** : **Pearson parametric correlation test**, **Spearman** and **Kendall** rank-based **correlation analysis**. The default is pearson correlation coefficient which measures the linear **dependence** between two variables. kendall and spearman correlation methods are non-parametric **rank-based correlation test**.

Compute Correlation Matrix

If your data contain missing values, use the following R code to handle missing values by case-wise deletion.

```r
cor(x, method = "pearson", use = "complete.obs")
```

Make Function to Format

Simple function for formatting a **correlation matrix** into a table with 4 columns containing : Column 1 : row names (variable 1 for the correlation test), Column 2 : column names (variable 2 for the correlation test), Column 3 : the **correlation coefficients**, Column 4 : the **p-values** of the correlations

```r
flattenCorrMatrix <- function(cormat, pmat) {
  ut <- upper.tri(cormat)
  data.frame(
    row = rownames(cormat)[row(cormat)[ut]],
```

```
    column = rownames(cormat)[col(cormat)[ut]],
    cor  =(cormat)[ut],
    p = pmat[ut])}
```

Make Matrix

The function **rcorr()** [in **Hmisc** package] can be used to compute the **significance levels** for **pearson** and **spearman correlations**. It returns both the correlation coefficients and the p-value of the correlation for all possible pairs of columns in the data table.

```
bclean<- rcorr(as.matrix(b_norm))
flattenCorrMatrix(bclean$r, bclean$P)
```

```
        row   column          cor            p
1  copies_mL      chl  0.424384223 1.691235e-01
2  copies_mL     temp -0.236364771 4.595296e-01
3        chl     temp -0.334160061 2.884230e-01
4  copies_mL   do_mgl  0.040715966 9.000217e-01
5        chl   do_mgl  0.399047726 1.987862e-01
6       temp   do_mgl -0.798746187 1.834246e-03
7  copies_mL   do_per -0.101928025 7.526039e-01
8        chl   do_per  0.333264084 2.897980e-01
9       temp   do_per -0.739913350 5.938779e-03
10    do_mgl   do_per  0.981998903 1.444345e-08
11 copies_mL       pH -0.090384104 7.799754e-01
12       chl       pH  0.118857022 7.129402e-01
13      temp       pH -0.206445779 5.197354e-01
14    do_mgl       pH  0.212156883 5.079865e-01
15    do_per       pH  0.233586970 4.649763e-01
16 copies_mL      amm  0.268597289 3.985937e-01
17       chl      amm  0.101391969 7.538694e-01
18      temp      amm -0.341512612 2.772798e-01
19    do_mgl      amm  0.312548446 3.226143e-01
20    do_per      amm  0.250623240 4.320520e-01
21        pH      amm -0.425690164 1.676752e-01
22 copies_mL   orthop  0.482379585 1.122269e-01
23       chl   orthop  0.427723591 1.654355e-01
24      temp   orthop -0.321965443 3.074536e-01
25    do_mgl   orthop  0.323036048 3.057556e-01
26    do_per   orthop  0.258084418 4.180008e-01
27        pH   orthop -0.024529552 9.396827e-01
```

28	amm	orthop	-0.249224323	4.347118e-01
29	copies_mL	sil	-0.152314048	6.365199e-01
30	chl	sil	-0.240499923	4.514774e-01
31	temp	sil	-0.379349806	2.239156e-01
32	do_mgl	sil	0.373777105	2.313547e-01
33	do_per	sil	0.396807301	2.015529e-01
34	pH	sil	-0.218608814	4.948563e-01
35	amm	sil	0.571850831	5.205678e-02
36	orthop	sil	-0.007650716	9.811735e-01
37	copies_mL	din	0.548177280	6.499395e-02
38	chl	din	0.359558395	2.509945e-01
39	temp	din	-0.543618288	6.771861e-02
40	do_mgl	din	0.467503637	1.253957e-01
41	do_per	din	0.329780358	2.951793e-01
42	pH	din	-0.203800072	5.252178e-01
43	amm	din	0.191467249	5.510959e-01
44	orthop	din	0.774072897	3.127218e-03
45	sil	din	0.208555520	5.153816e-01
46	copies_mL	din_dip	0.215717277	5.007219e-01
47	chl	din_dip	0.310276559	3.263321e-01
48	temp	din_dip	-0.716129317	8.801145e-03
49	do_mgl	din_dip	0.493488547	1.030132e-01
50	do_per	din_dip	0.394498473	2.044286e-01
51	pH	din_dip	-0.139062662	6.664474e-01
52	amm	din_dip	0.663275601	1.870811e-02
53	orthop	din_dip	0.028172577	9.307424e-01
54	sil	din_dip	0.509873632	9.036752e-02
55	din	din_dip	0.514231116	8.718991e-02
56	copies_mL	dust	0.368987692	2.378645e-01
57	chl	dust	0.401551076	1.957225e-01
58	temp	dust	-0.504965586	9.403916e-02
59	do_mgl	dust	0.714404445	9.042658e-03
60	do_per	dust	0.626559055	2.925072e-02
61	pH	dust	0.251170708	4.310133e-01
62	amm	dust	0.564496961	5.586262e-02
63	orthop	dust	0.072257076	8.234119e-01
64	sil	dust	0.283264128	3.723002e-01
65	din	dust	0.399525867	1.981988e-01
66	din_dip	dust	0.522786799	8.117270e-02
67	copies_mL	dust_lag	0.698802162	1.145807e-02
68	chl	dust_lag	0.727093796	7.376737e-03
69	temp	dust_lag	-0.593908784	4.173571e-02
70	do_mgl	dust_lag	0.646031933	2.323543e-02

```
71    do_per dust_lag  0.550716789 6.350957e-02
72        pH dust_lag  0.012884387 9.682993e-01
73       amm dust_lag  0.268809701 3.982063e-01
74    orthop dust_lag  0.656305678 2.045221e-02
75       sil dust_lag  0.076670381 8.127900e-01
76       din dust_lag  0.690183358 1.298173e-02
77   din_dip dust_lag  0.360213993 2.500682e-01
78      dust dust_lag  0.622673425 3.057347e-02
```

```r
cclean<- rcorr(as.matrix(c_norm))
flattenCorrMatrix(cclean$r, cclean$P)
```

```
        row   column          cor            p
1  copies_mL      chl  0.422398495 1.713409e-01
2  copies_mL     temp  0.063173362 8.453586e-01
3        chl     temp  0.652961803 2.132979e-02
4  copies_mL   do_mgl -0.149524858 6.427796e-01
5        chl   do_mgl -0.068547700 8.323608e-01
6       temp   do_mgl -0.453600118 1.385817e-01
7  copies_mL   do_per -0.128740158 6.900762e-01
8        chl   do_per -0.028510814 9.299127e-01
9       temp   do_per -0.416625174 1.778912e-01
10    do_mgl   do_per  0.998276278 1.194600e-13
11 copies_mL       pH -0.182827097 5.695324e-01
12       chl       pH  0.097087354 7.640515e-01
13      temp       pH  0.079501170 8.059920e-01
14    do_mgl       pH  0.172759583 5.913201e-01
15    do_per       pH  0.158984068 6.216394e-01
16 copies_mL      amm -0.108144710 7.379693e-01
17       chl      amm  0.326671917 3.000280e-01
18      temp      amm  0.872365160 2.144869e-04
19    do_mgl      amm -0.601754981 3.844305e-02
20    do_per      amm -0.578097323 4.896998e-02
21        pH      amm -0.033534377 9.175976e-01
22 copies_mL   orthop -0.165890855 6.063666e-01
23       chl   orthop -0.481702264 1.128057e-01
24      temp   orthop -0.157001675 6.260488e-01
25    do_mgl   orthop -0.493694850 1.028470e-01
26    do_per   orthop -0.505758703 9.343916e-02
27        pH   orthop -0.102317357 7.516851e-01
28       amm   orthop  0.132858057 6.806183e-01
```

```
29 copies_mL      sil -0.121687035 7.063701e-01
30       chl      sil  0.459947218 1.324558e-01
31      temp      sil  0.223780014 4.844441e-01
32    do_mgl      sil  0.563745981 5.626190e-02
33    do_per      sil  0.581788786 4.720759e-02
34        pH      sil  0.132684837 6.810153e-01
35       amm      sil  0.116490441 7.184483e-01
36    orthop      sil -0.725568926 7.563691e-03
37 copies_mL      din  0.046835143 8.850781e-01
38       chl      din -0.006894311 9.830346e-01
39      temp      din  0.557600165 5.960457e-02
40    do_mgl      din -0.872287147 2.151139e-04
41    do_per      din -0.862859406 3.020937e-04
42        pH      din -0.116239090 7.190340e-01
43       amm      din  0.752930699 4.705593e-03
44    orthop      din  0.622873031 3.050449e-02
45       sil      din -0.552619842 6.241273e-02
46 copies_mL  din_dip  0.349732772 2.651184e-01
47       chl  din_dip  0.632909856 2.717830e-02
48      temp  din_dip  0.459549730 1.328341e-01
49    do_mgl  din_dip -0.070507102 8.276314e-01
50    do_per  din_dip -0.051842656 8.728746e-01
51        pH  din_dip  0.018141273 9.553750e-01
52       amm  din_dip  0.156697851 6.267255e-01
53    orthop  din_dip -0.762075382 3.962779e-03
54       sil  din_dip  0.343844269 2.737984e-01
55       din  din_dip -0.139428555 6.656148e-01
56 copies_mL     dust  0.308382544 3.294492e-01
57       chl     dust  0.380630232 2.222269e-01
58      temp     dust  0.246445675 4.400184e-01
59    do_mgl     dust -0.525278678 7.947479e-02
60    do_per     dust -0.508465866 9.141058e-02
61        pH     dust  0.047651181 8.830877e-01
62       amm     dust  0.218564123 4.949467e-01
63    orthop     dust  0.340545296 2.787316e-01
64       sil     dust -0.411845524 1.834308e-01
65       din     dust  0.508319232 9.151969e-02
66   din_dip     dust  0.126471490 6.953044e-01
67 copies_mL dust_lag  0.747545414 5.189617e-03
68       chl dust_lag  0.691939899 1.265970e-02
69      temp dust_lag  0.110776455 7.317976e-01
70    do_mgl dust_lag  0.036626502 9.100255e-01
71    do_per dust_lag  0.069470043 8.301339e-01
```

```
72        pH dust_lag -0.138028896 6.688017e-01
73       amm dust_lag -0.203933228 5.249413e-01
74    orthop dust_lag -0.218820295 4.944285e-01
75       sil dust_lag  0.072668880 8.224196e-01
76       din dust_lag -0.156602903 6.269371e-01
77   din_dip dust_lag  0.355997752 2.560607e-01
78      dust dust_lag  0.401032196 1.963551e-01
```

```
rclean<- rcorr(as.matrix(r_norm))
flattenCorrMatrix(rclean$r, rclean$P)
```

```
        row   column          cor            p
1  copies_mL      chl  0.794313377 2.029048e-03
2  copies_mL     temp -0.694069633 1.227724e-02
3        chl     temp -0.443563509 1.486370e-01
4  copies_mL   do_mgl  0.053133959 8.697316e-01
5        chl   do_mgl -0.093135507 7.734296e-01
6       temp   do_mgl -0.298063032 3.467157e-01
7  copies_mL   do_per -0.277222477 3.830195e-01
8        chl   do_per -0.420595307 1.733703e-01
9       temp   do_per -0.104089537 7.475068e-01
10    do_mgl   do_per  0.849285771 4.728164e-04
11 copies_mL       pH -0.605752072 3.683882e-02
12       chl       pH -0.446839506 1.453050e-01
13      temp       pH  0.344394945 2.729798e-01
14    do_mgl       pH  0.181718635 5.719154e-01
15    do_per       pH  0.236204096 4.598438e-01
16 copies_mL      amm  0.818594338 1.131031e-03
17       chl      amm  0.652838885 2.136257e-02
18      temp      amm -0.754522513 4.569286e-03
19    do_mgl      amm  0.365888141 2.421346e-01
20    do_per      amm  0.018341958 9.548818e-01
21        pH      amm -0.253018859 4.275157e-01
22 copies_mL   orthop  0.372356247 2.332747e-01
23       chl   orthop  0.268756214 3.983039e-01
24      temp   orthop -0.232958050 4.662137e-01
25    do_mgl   orthop -0.259132952 4.160445e-01
26    do_per   orthop -0.406559303 1.896810e-01
27        pH   orthop -0.012512767 9.692133e-01
28       amm   orthop  0.145320512 6.522556e-01
29 copies_mL      sil -0.148270301 6.456022e-01
```

```
30       chl      sil  0.029808950 9.267289e-01
31      temp      sil  0.436241090 1.562603e-01
32    do_mgl      sil -0.694381992 1.222187e-02
33    do_per      sil -0.562797731 5.676890e-02
34        pH      sil -0.423805935 1.697674e-01
35       amm      sil -0.473660911 1.198284e-01
36    orthop      sil -0.001857214 9.954295e-01
37 copies_mL      din  0.824224135 9.758559e-04
38       chl      din  0.662898086 1.879968e-02
39      temp      din -0.754324335 4.586092e-03
40    do_mgl      din  0.359004781 2.517783e-01
41    do_per      din  0.008917153 9.780578e-01
42        pH      din -0.230646797 4.707741e-01
43       amm      din  0.997424631 8.884005e-13
44    orthop      din  0.201101449 5.308352e-01
45       sil      din -0.483244657 1.114905e-01
46 copies_mL  din_dip  0.689745678 1.306290e-02
47       chl  din_dip  0.568365081 5.383735e-02
48      temp  din_dip -0.661399605 1.916640e-02
49    do_mgl  din_dip  0.523777956 8.049442e-02
50    do_per  din_dip  0.216684979 4.987554e-01
51        pH  din_dip -0.235818220 4.605989e-01
52       amm  din_dip  0.942026284 4.677072e-06
53    orthop  din_dip -0.173125530 5.905225e-01
54       sil  din_dip -0.517084122 8.515090e-02
55       din  din_dip  0.922943116 1.878016e-05
56 copies_mL     dust  0.609074501 3.554219e-02
57       chl     dust  0.396820491 2.015366e-01
58      temp     dust -0.825081966 9.537341e-04
59    do_mgl     dust  0.088367808 7.847807e-01
60    do_per     dust -0.028003720 9.311566e-01
61        pH     dust -0.461536881 1.309496e-01
62       amm     dust  0.486749576 1.085398e-01
63    orthop     dust  0.540027877 6.991911e-02
64       sil     dust -0.064970788 8.410074e-01
65       din     dust  0.507091907 9.243639e-02
66   din_dip     dust  0.297131770 3.482972e-01
67 copies_mL dust_lag  0.666513887 1.793597e-02
68       chl dust_lag  0.620290805 3.140552e-02
69      temp dust_lag -0.507661770 9.200999e-02
70    do_mgl dust_lag -0.227706899 4.766046e-01
71    do_per dust_lag -0.390328132 2.096861e-01
72        pH dust_lag -0.837690963 6.709882e-04
```

```
73       amm dust_lag  0.350879809 2.634464e-01
74    orthop dust_lag  0.099840923 7.575342e-01
75       sil dust_lag  0.425220100 1.681956e-01
76       din dust_lag  0.332242196 2.913707e-01
77   din_dip dust_lag  0.287296281 3.652342e-01
78      dust dust_lag  0.622673425 3.057347e-02
```

Extract cor and p values

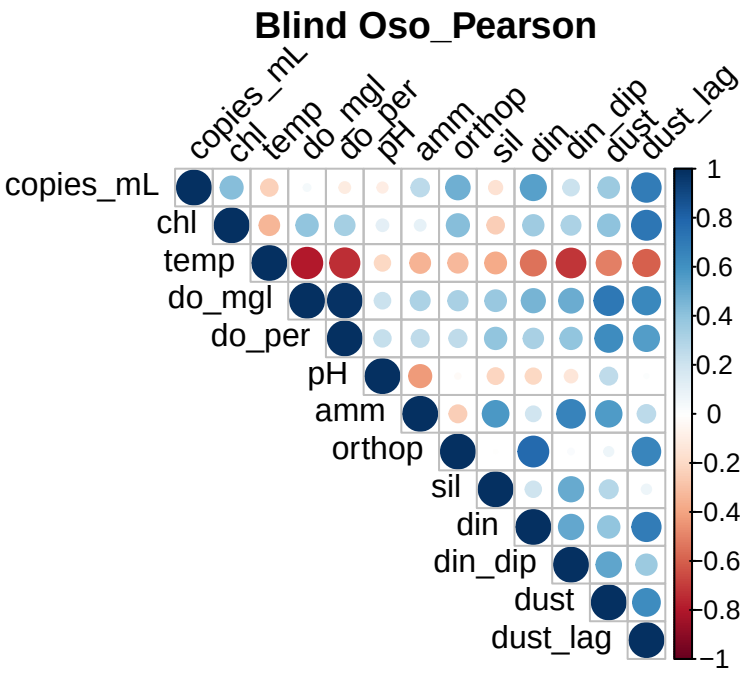
```
b_p_values<-
 as.data.frame(bclean$P)
```

Visualize with `corrplot()` function

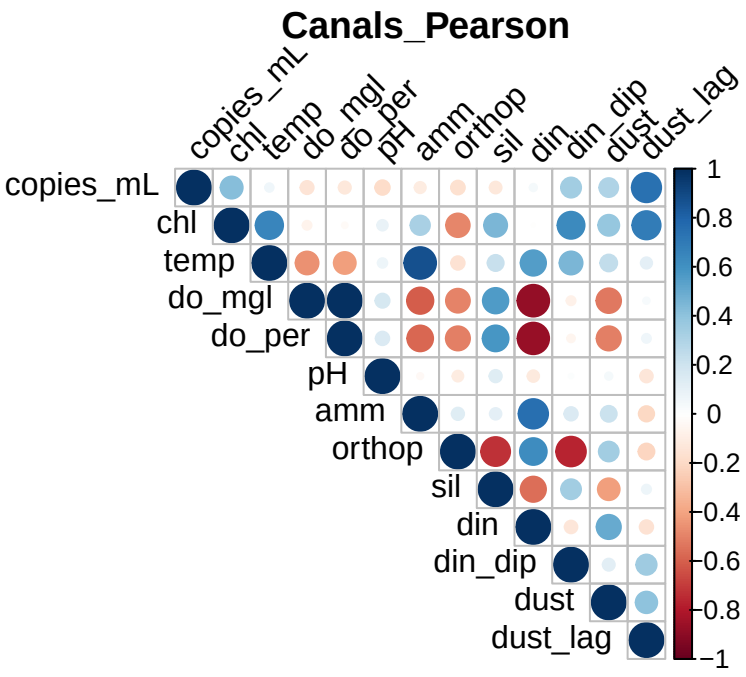
The function **corrplot()** takes the **correlation matrix** as the first argument. The second argument (type="upper") is used to display only the upper triangular of the **correlation matrix**.

Just Normal Data

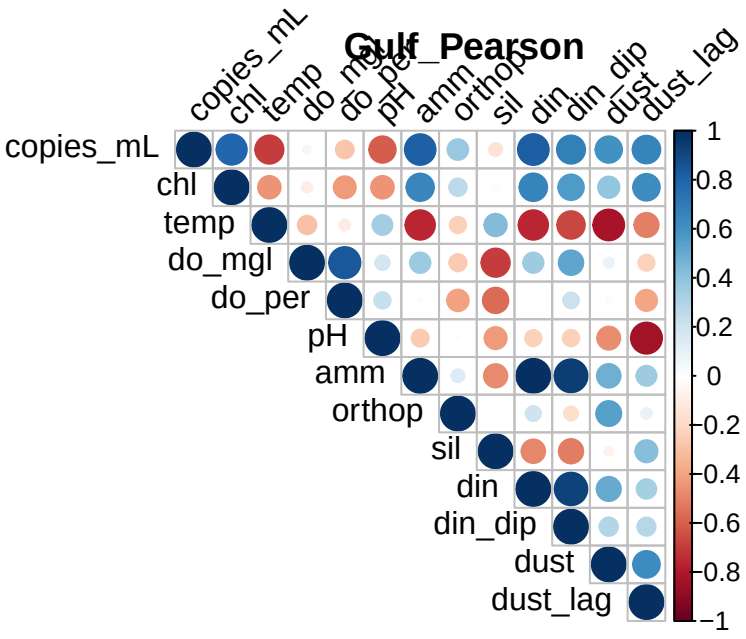
```
#Normal Data
 corrplot(b, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Blind Oso_Pearson",  mar=c(0,0,1,0)) #Adds and lowers title
```




```
corrplot(c, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Canals_Pearson", mar=c(0,0,1,0))
```

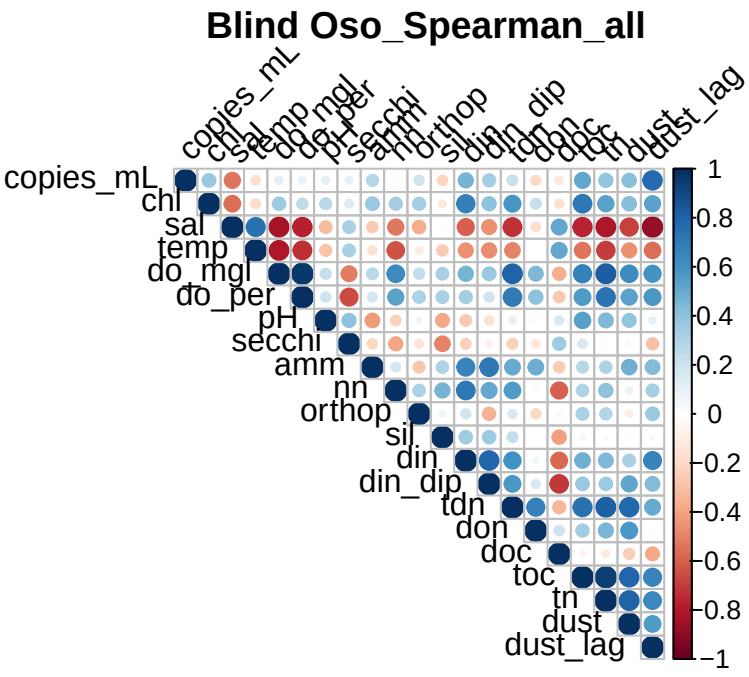



```
corrplot(r, type = "upper",
         tl.col = "black", tl.srt = 45, mar=c(0,0,1,0))
         title(main = "Gulf_Pearson")
```

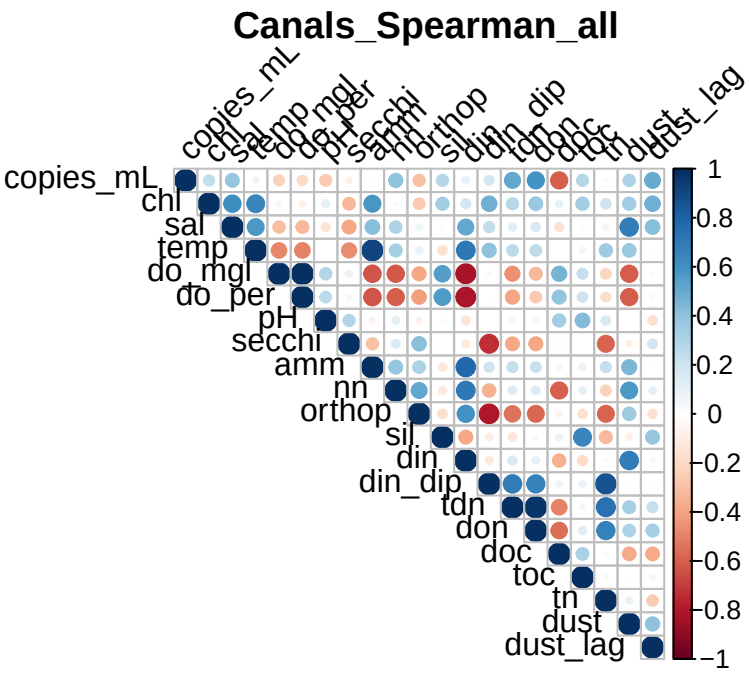


All Spearman

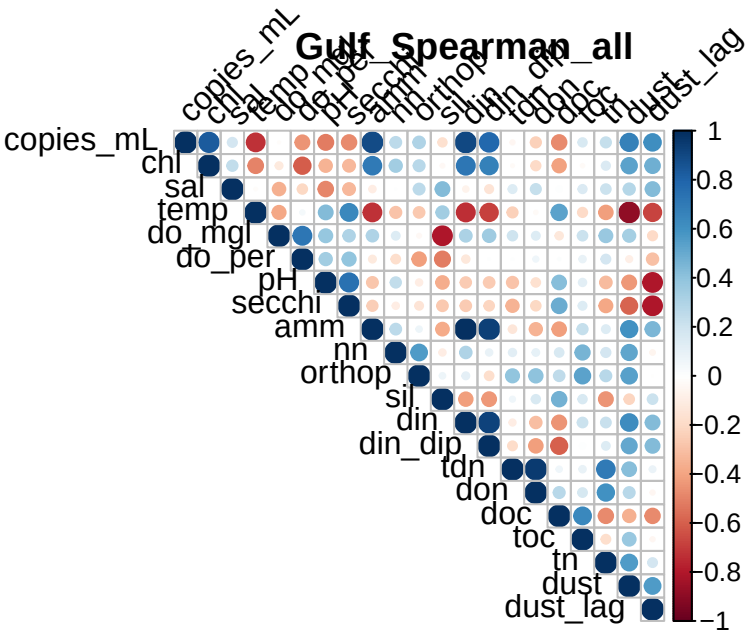
```
#Non-Parametric Data
 corrplot(b2, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Blind Oso_Spearman_all",  mar=c(0,0,1,0)) #Adds and lowers title
```




```
corrplot(c22, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Canals_Spearman_all", mar=c(0,0,1,0))
```




```
corrplot(r2, type = "upper",
         tl.col = "black", tl.srt = 45, mar=c(0,0,1,0))
         title(main = "Gulf_Spearman_all" )
```



Everything

```
par(mfrow=c(3,3))
#Normal Data
 corrplot(b, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Blind Oso_Pearson",  mar=c(0,0,1,0)) #Adds and lowers title

 corrplot(c, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Canals_Pearson", mar=c(0,0,1,0))

 corrplot(r, type = "upper",
         tl.col = "black", tl.srt = 45, mar=c(0,0,1,0))
         title(main = "Gulf_Pearson")

#Non-Parametric Data
 corrplot(b1, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Blind Oso_Spearman",  mar=c(0,0,1,0)) #Adds and lowers title
```

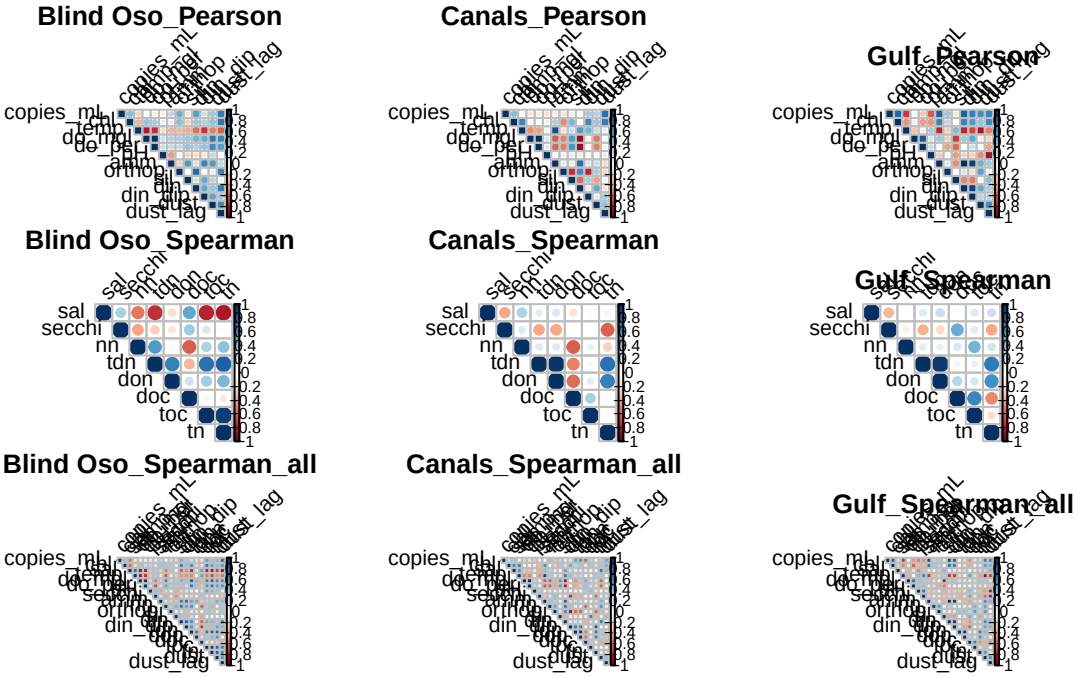
```
corrplot(c1, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Canals_Spearman", mar=c(0,0,1,0))

corrplot(r1, type = "upper",
         tl.col = "black", tl.srt = 45, mar=c(0,0,1,0))
         title(main = "Gulf_Spearman" )

#Non-Parametric Data
 corrplot(b2, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Blind Oso_Spearman_all",  mar=c(0,0,1,0)) #Adds and lowers title

corrplot(c22, type = "upper",
         tl.col = "black", tl.srt = 45,
         title = "Canals_Spearman_all", mar=c(0,0,1,0))

corrplot(r2, type = "upper",
         tl.col = "black", tl.srt = 45, mar=c(0,0,1,0))
         title(main = "Gulf_Spearman_all" )
```



It's also possible to **combine correlogram with the significance test**. We'll use the result *res.cor2* generated in the previous section with **rcorr**() function [in **Hmisc** package]:

```
# Insignificant correlation are crossed
#corrplot(bclean$r, type="upper", order="hclust",
        # p.mat = bclean$P, sig.level = 0.05, insig = "blank")

#corrplot(cclean$r, type="upper", order="hclust",
        # p.mat = cclean$P, sig.level = 0.05, insig = "blank")

#corrplot(rclean$r, type="upper", order="hclust",
         #p.mat = rclean$P, sig.level = 0.05, insig = "blank")
```

Correlation Tables

Load in data again to bypass earlier code

Rename COpies –> TOtal Vib

```
bo<-
  bo %>%
  rename("Vibrio" = "copies_mL")

c2<-
  c2 %>%
  rename("Vibrio" = "copies_mL")

rd<-
  rd %>%
  rename("Vibrio" = "copies_mL")
```

Create Species Dataframes and Site-Species Dataframes

```
#SPECIES DF
vc<-
  sp %>%
  filter(type %in% "V. cholerae") %>%
```

```
  rename("Vc" = "copies_mL")
vp<-
  sp %>%
  filter(type %in% "V. parahaemolyticus") %>%
  rename("Vp" = "copies_mL")
vv<-
  sp %>%
  filter(type %in% "V. vulnificus") %>%
  rename("Vv" = "copies_mL")

#SITE_SPECIES DF
b_vc <-
  vc %>%
  filter(site %in% "Blind Oso") %>%
  select(Vc, date)
b_vv <-
  vv %>%
  filter(site %in% "Blind Oso") %>%
  select(Vv, date)
b_vp <-
  vp %>%
  filter(site %in% "Blind Oso") %>%
  select(Vp, date)

c_vc <-
  vc %>%
  filter(site %in% "Canals") %>%
  select(Vc, date)
c_vv <-
  vv %>%
  filter(site %in% "Canals") %>%
  select(Vv, date)
c_vp <-
  vp %>%
  filter(site %in% "Canals") %>%
  select(Vp, date)

r_vc <-
  vc %>%
  filter(site %in% "Gulf") %>%
  select(Vc, date)
```

```
r_vv <-
  vv %>%
  filter(site %in% "Gulf") %>%
  select(Vv, date)
r_vp <-
  vp %>%
  filter(site %in% "Gulf") %>%
  select(Vp, date)
```

Combine Species with Site Masters

```
b_list<- list(b_vc, b_vv, b_vp, bo)
c_list<- list(c_vc, c_vv, c_vp, c2)
r_list<- list(r_vc, r_vv, r_vp, rd)

bo<-
  b_list %>% reduce(full_join, by = 'date') %>%
  select(!c(date, ...1, Vp))

c2<-
  c_list %>% reduce(full_join, by = 'date') %>%
  select(!c(date, ...1))

rd<-
  r_list %>% reduce(full_join, by = 'date') %>%
  select(!c(date, ...1))
```

This is using Spearman's for all

Make Correlation Function

p* = < 0.05

** = < 0.01

*** = <0.001

```
bo_table<-
  kable(as.data.frame(correlation_matrix(bo, type = "spearman", digits = 2)) %>%
  select(1:4) %>%
    filter(!row_number() %in% c(1:4)),
```

Table 1: Blind Oso: Correlations between Environmental and Biological Variables using Spearman's Rank correlations

	Vc	Vv	Vibrio	chl
sal	-0.52	-0.78**	-0.41	-0.63*
temp	-0.61*	-0.67*	-0.24	-0.30
do_mgl	0.64*	0.83***	0.04	0.43
do_per	0.55	0.78**	-0.10	0.38
pH	-0.16	0.04	-0.09	0.21
secchi	-0.24	-0.37	-0.27	-0.24
amm	0.69*	0.71**	0.19	-0.09
nn	0.61*	0.45	0.55	0.30
orthop	0.12	0.24	0.48	0.45
sil	0.51	0.53	-0.15	-0.26
din	0.62*	0.47	0.55	0.29
din_dip	0.63*	0.66*	-0.05	0.02
tdn	0.81**	0.90***	0.29	0.41
don	0.37	0.64*	-0.20	0.21
doc	-0.10	0.01	-0.18	0.19
toc	0.29	0.64*	0.31	0.80**
tn	0.65*	0.89***	0.37	0.65*
dust	0.77**	0.91***	0.27	0.35
dust_lag	0.42	0.69*	0.65*	0.76**

```
  caption = "Blind Oso: Correlations between Environmental and Biological Variables using

c2_table<-
  kable(as.data.frame(correlation_matrix(c2, type = "spearman", digits = 2)) %>%
  select(1:5) %>%
    filter(!row_number() %in% c(1:5)),
  caption = "Canals: Correlations between Environmental and Biological Variables using Spe

rd_table<-
  kable(as.data.frame(correlation_matrix(rd, type = "spearman", digits = 2)) %>%
  select(1:5) %>%
    filter(!row_number() %in% c(1:5)),
  caption = "Gulf: Correlations between Environmental and Biological Variables using Spear

bo_table
```

Table 2: Canals: Correlations between Environmental and Biological Variables using Spearman's Rank correlations

	Vc	Vv	Vp	Vibrio	chl
sal	-0.30	0.27	0.33	0.38	0.64*
temp	0.18	-0.27	0.04	0.06	0.65*
do_mgl	-0.26	0.21	0.17	-0.15	-0.07
do_per	-0.27	0.18	0.18	-0.13	-0.03
pH	-0.01	0.38	-0.02	-0.18	0.10
secchi	0.28	0.06	-0.52	-0.02	-0.34
amm	0.14	-0.34	-0.09	-0.11	0.33
nn	-0.12	0.20	-0.11	0.21	-0.42
orthop	-0.13	0.18	-0.14	0.11	-0.48
sil	0.08	-0.24	0.06	-0.12	0.46
din	0.04	-0.15	-0.15	0.05	-0.01
din_dip	0.13	-0.15	0.14	0.35	0.63*
tdn	0.18	-0.35	0.06	0.43	0.39
don	0.17	-0.32	0.10	0.41	0.39
doc	-0.14	0.48	-0.08	-0.33	-0.05
toc	-0.21	0.33	-0.23	0.03	0.16
tn	-0.13	0.01	0.10	0.16	0.39
dust	-0.25	0.35	0.38	0.31	0.32
dust_lag	-0.08	-0.25	-0.16	0.78**	0.76**

```
c2_table
```

```
rd_table
```

Table with Just Dust and Vibrio/Chl

```
bo_table2<-
  kable(as.data.frame(correlation_matrix(bo, type = "spearman", digits = 2)) %>%
  select(1:4) %>%
    filter(row_number() %in% c(22,23)),
  caption = "Blind Oso: Correlations between Dust Depsoition and Biological Variables usin

c2_table2<-
  kable(as.data.frame(correlation_matrix(c2, type = "spearman", digits = 2)) %>%
  select(1:5) %>%
```

Table 3: Gulf: Correlations between Environmental and Biological Variables using Spearman's Rank correlations

	Vc	Vv	Vp	Vibrio	chl
sal	0.43	0.20	-0.34	0.23	0.32
temp	-0.30	-0.07	0.28	-0.69*	-0.44
do_mgl	-0.25	0.10	-0.16	0.05	-0.09
do_per	-0.37	0.11	0.01	-0.28	-0.42
pH	-0.48	0.40	0.43	-0.61*	-0.45
secchi	-0.45	0.26	0.58*	-0.58*	-0.39
amm	0.08	0.11	-0.08	0.82**	0.65*
nn	-0.20	-0.09	0.26	0.26	0.29
orthop	-0.01	-0.04	0.32	0.37	0.27
sil	0.30	-0.19	0.04	-0.15	0.03
din	0.06	0.10	-0.05	0.82***	0.66*
din_dip	0.08	0.15	-0.20	0.69*	0.57
tdn	-0.13	-0.43	-0.11	0.06	-0.07
don	-0.14	-0.44	-0.09	-0.12	-0.22
doc	-0.36	0.30	0.73**	-0.46	-0.43
toc	-0.26	0.11	0.54	0.18	-0.04
tn	0.01	-0.42	-0.34	0.33	0.06
dust	0.29	-0.25	-0.05	0.54	0.26
dust_lag	0.77**	-0.16	-0.47	0.65*	0.68*

Table 4: Blind Oso: Correlations between Dust Depsoition and Biological Variables using Spearman's Rank correlations

	Vc	Vv	Vibrio	chl
dust	0.77**	0.91***	0.27	0.35
dust_lag	0.42	0.69*	0.65*	0.76**

Table 5: Canals: Correlations between Dust Depsoition and Biological Variables using Spearman's Rank correlations

	Vc	Vv	Vp	Vibrio	chl
dust	-0.25	0.35	0.38	0.31	0.32
dust_lag	-0.08	-0.25	-0.16	0.78**	0.76**

```
    filter(row_number() %in% c(23,24)),
  caption = "Canals: Correlations between Dust Depsoition and Biological Variables using S

rd_table2<-
  kable(as.data.frame(correlation_matrix(rd, type = "spearman", digits = 2)) %>%
  select(1:5) %>%
    filter(row_number() %in% c(23,24)),
  caption = "Gulf: Correlations between Dust Depsoition and Biological Variables using Spe

bo_table2
```

```
c2_table2
```

```
rd_table2
```

Table 6: Gulf: Correlations between Dust Depsoition and Biological Variables using Spearman's Rank correlations

	Vc	Vv	Vp	Vibrio	chl
dust	0.29	-0.25	-0.05	0.54	0.26
dust_lag	0.77**	-0.16	-0.47	0.65*	0.68*