FACS_analysis

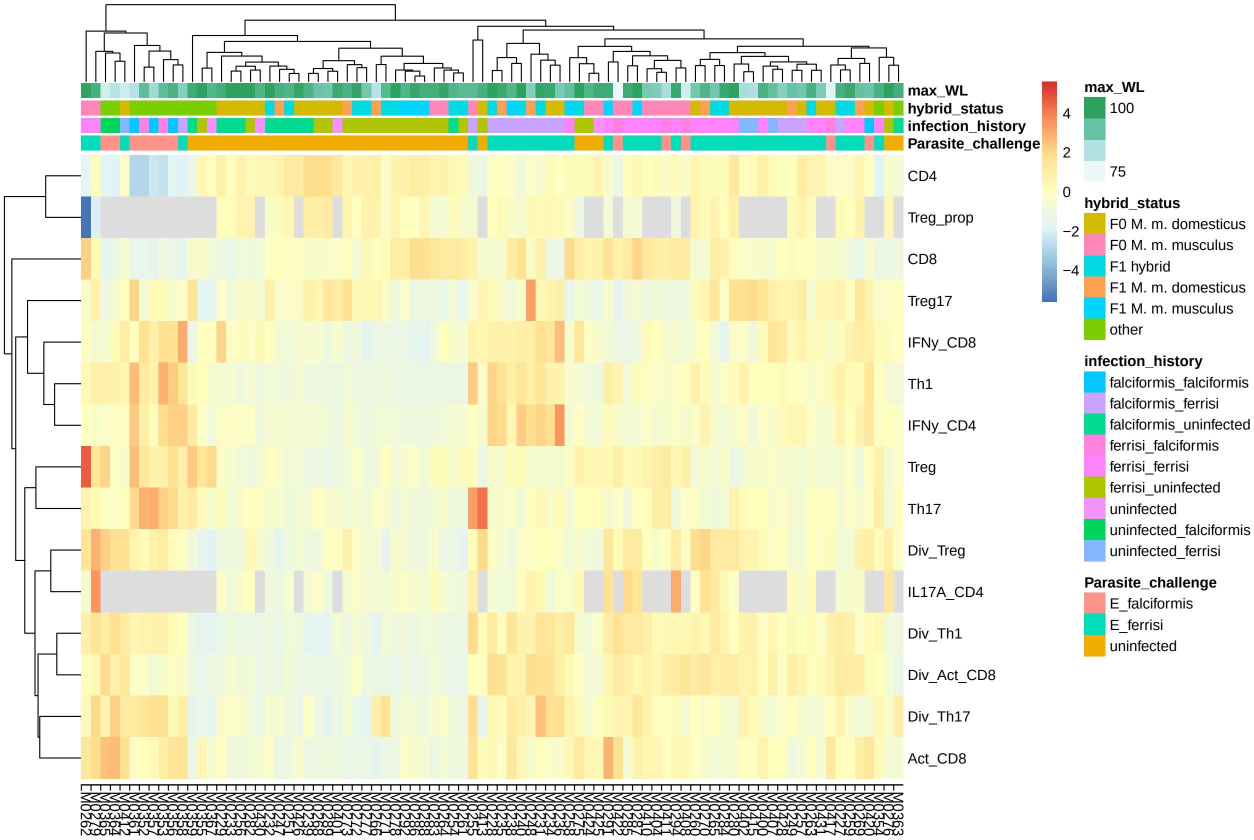
Fay

2022-05-18

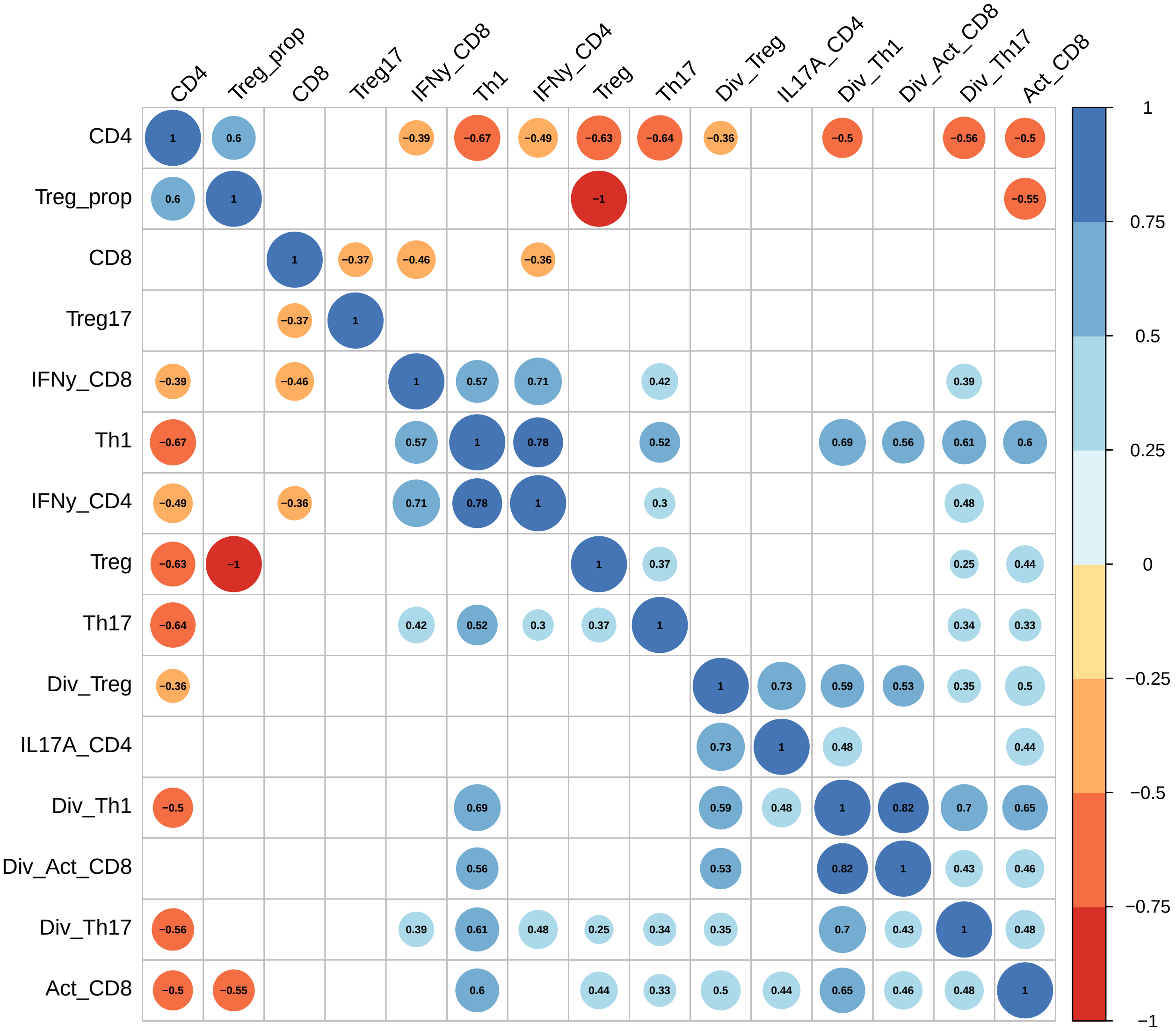
1. facs expression in the laboratory infections - Heatmap

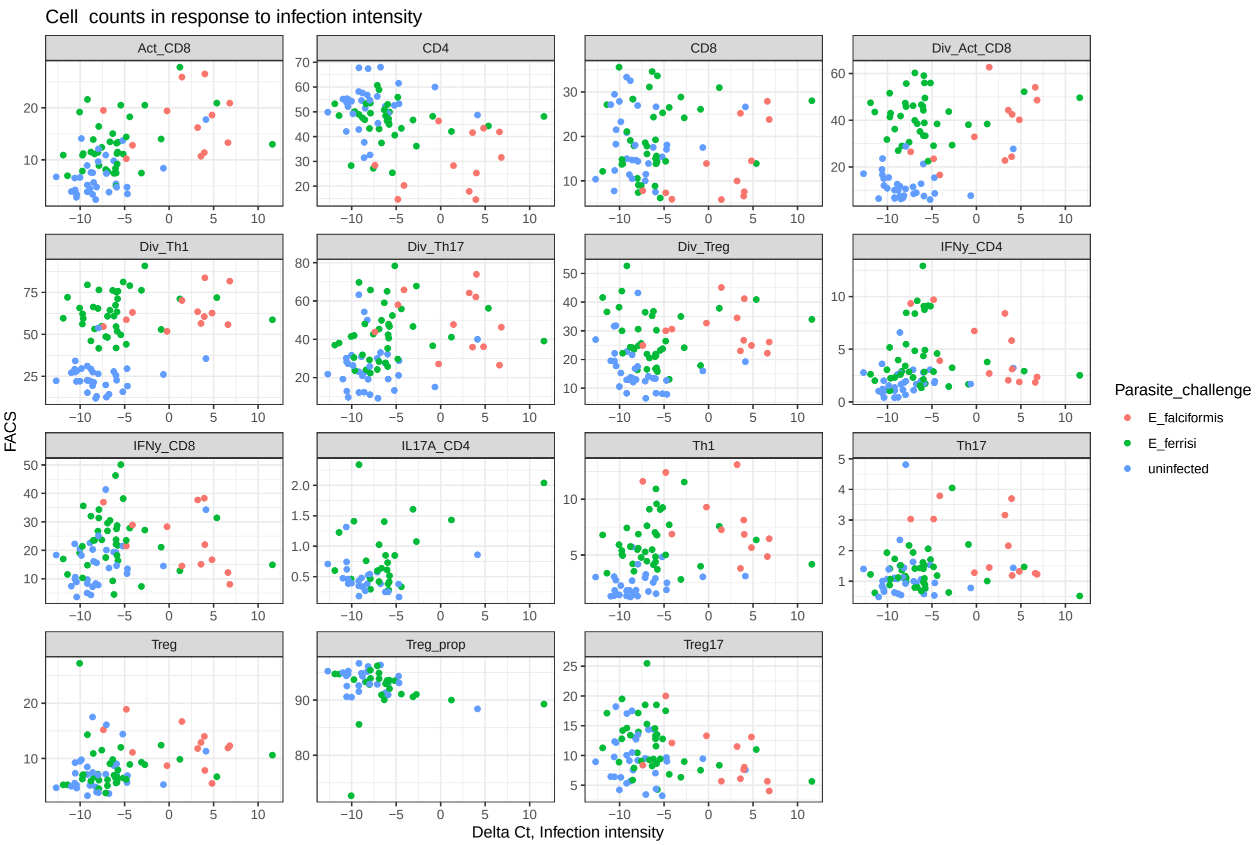
FACS

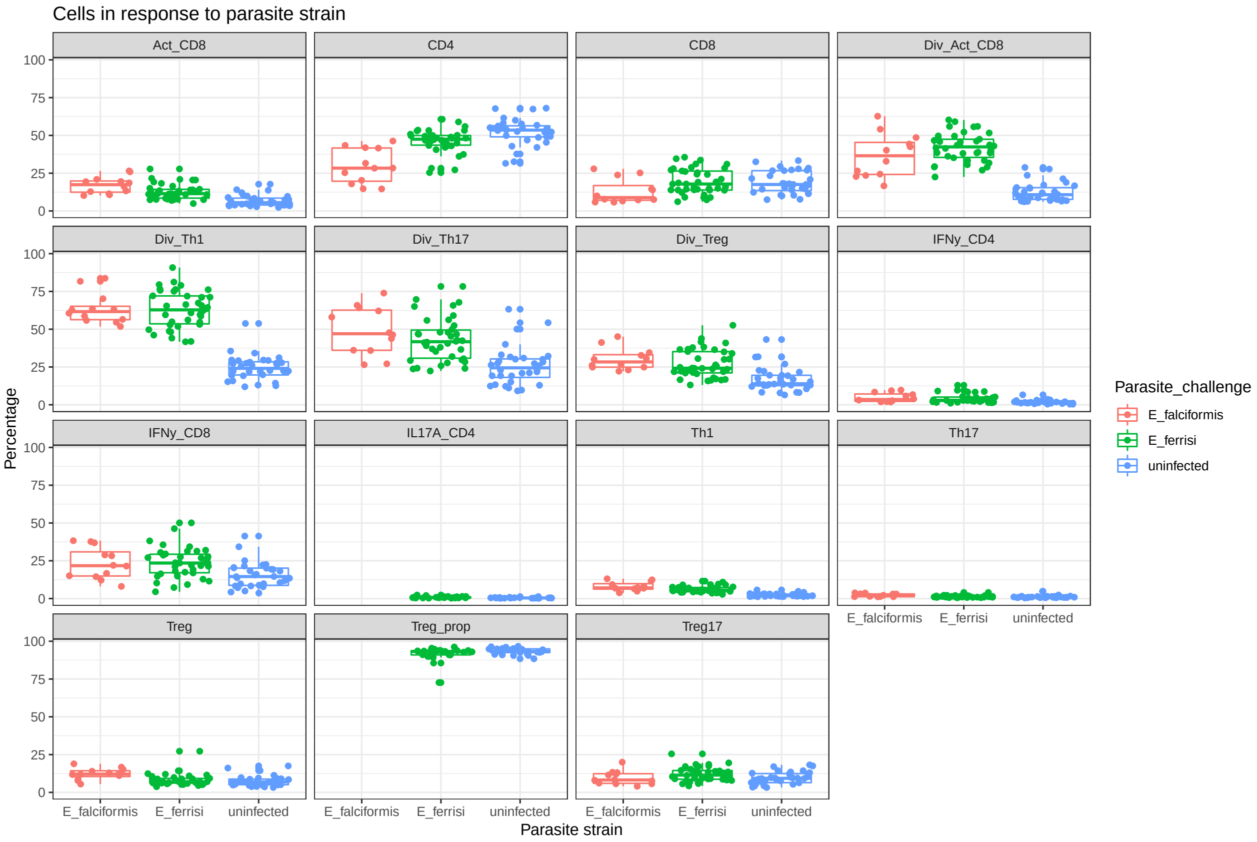
```
heatmap_data %>%
  pheatmap(annotation_col = annotation_df, scale = "row")
```

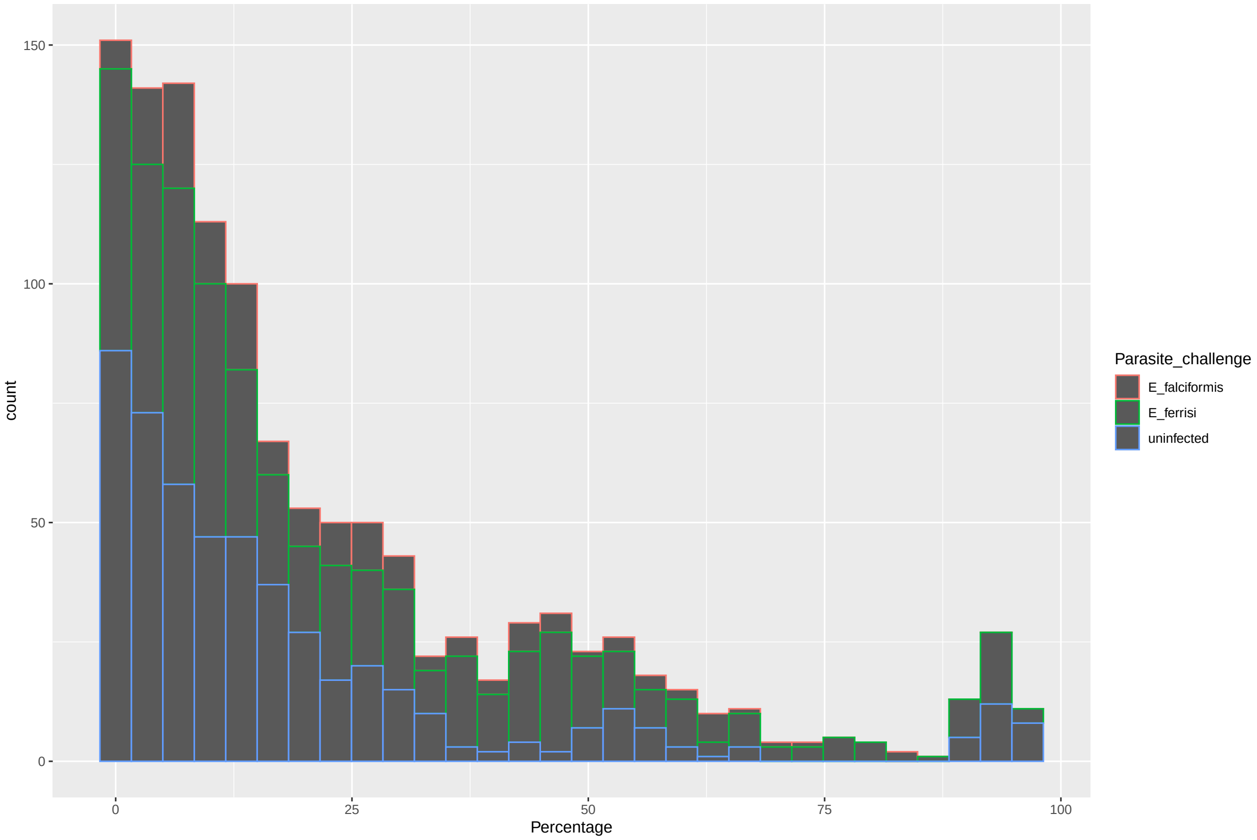


2. Correlations between the cells

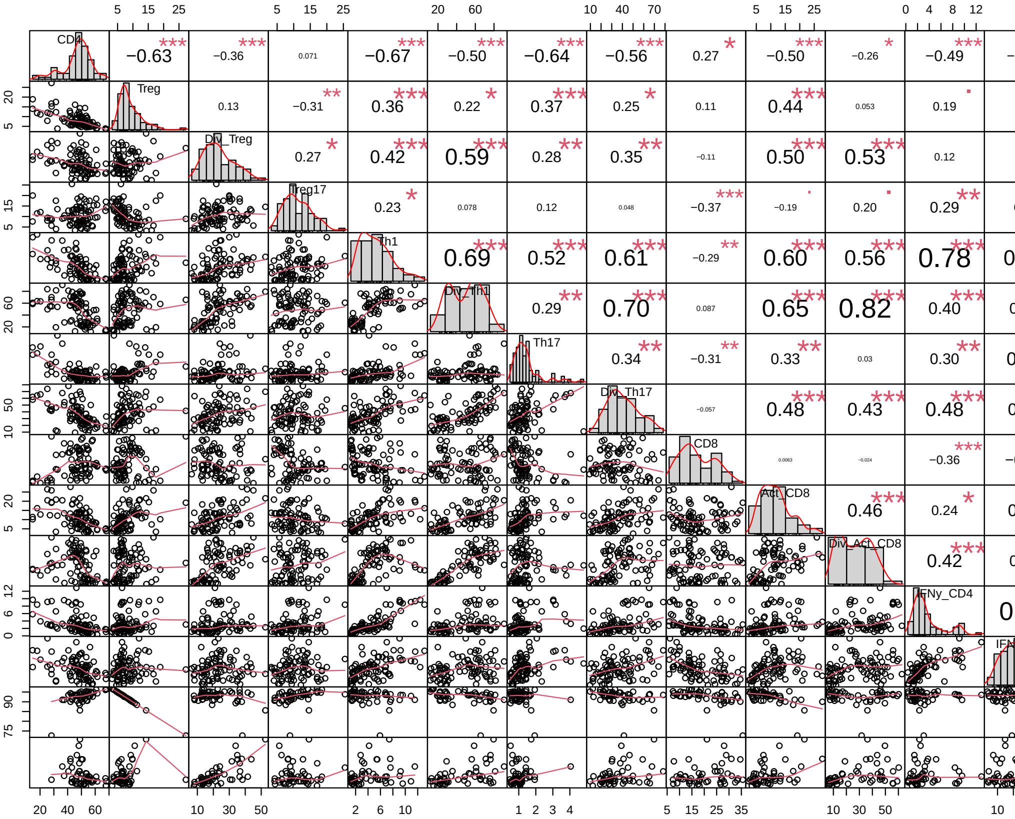



Cell counts in response to infection intensity
Act_CD8
CD4
CD8
Div_Act_CD8
Div_Th1
Div_Th17
Div_Treg
IFNy_CD4
IFNy_CD8
IL17A_CD4
Th1
Th17
Treg
Treg_prop
Treg17
FACS
Delta Ct, Infection intensity
−10
−5
0
5
10
Parasite_challenge
E_falciformis
E_ferrisi
uninfected


Cells in response to parasite strain
Act_CD8
CD4
CD8
Div_Act_CD8
Div_Th1
Div_Th17
Div_Treg
IFNy_CD4
IFNy_CD8
IL17A_CD4
Th1
Th17
Treg
Treg_prop
Treg17
Percentage
0
25
50
75
100
E_falciformis
E_ferrisi
uninfected
Parasite strain
Parasite_challenge
E_falciformis
E_ferrisi
uninfected


count
150
100
50
0
0
25
50
75
100
Percentage
Parasite_challenge
E_falciformis
E_ferrisi
uninfected

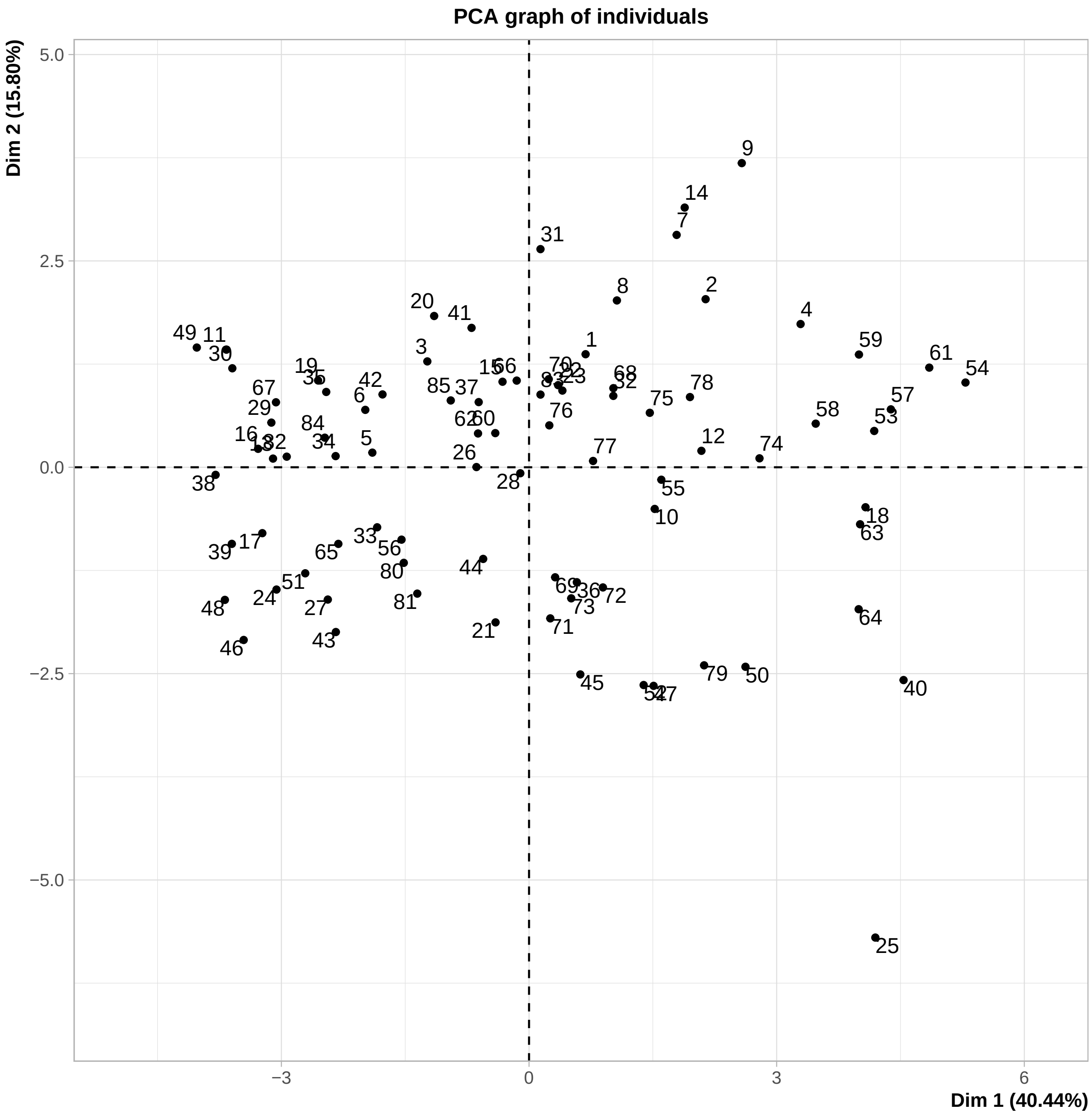
3. PCA

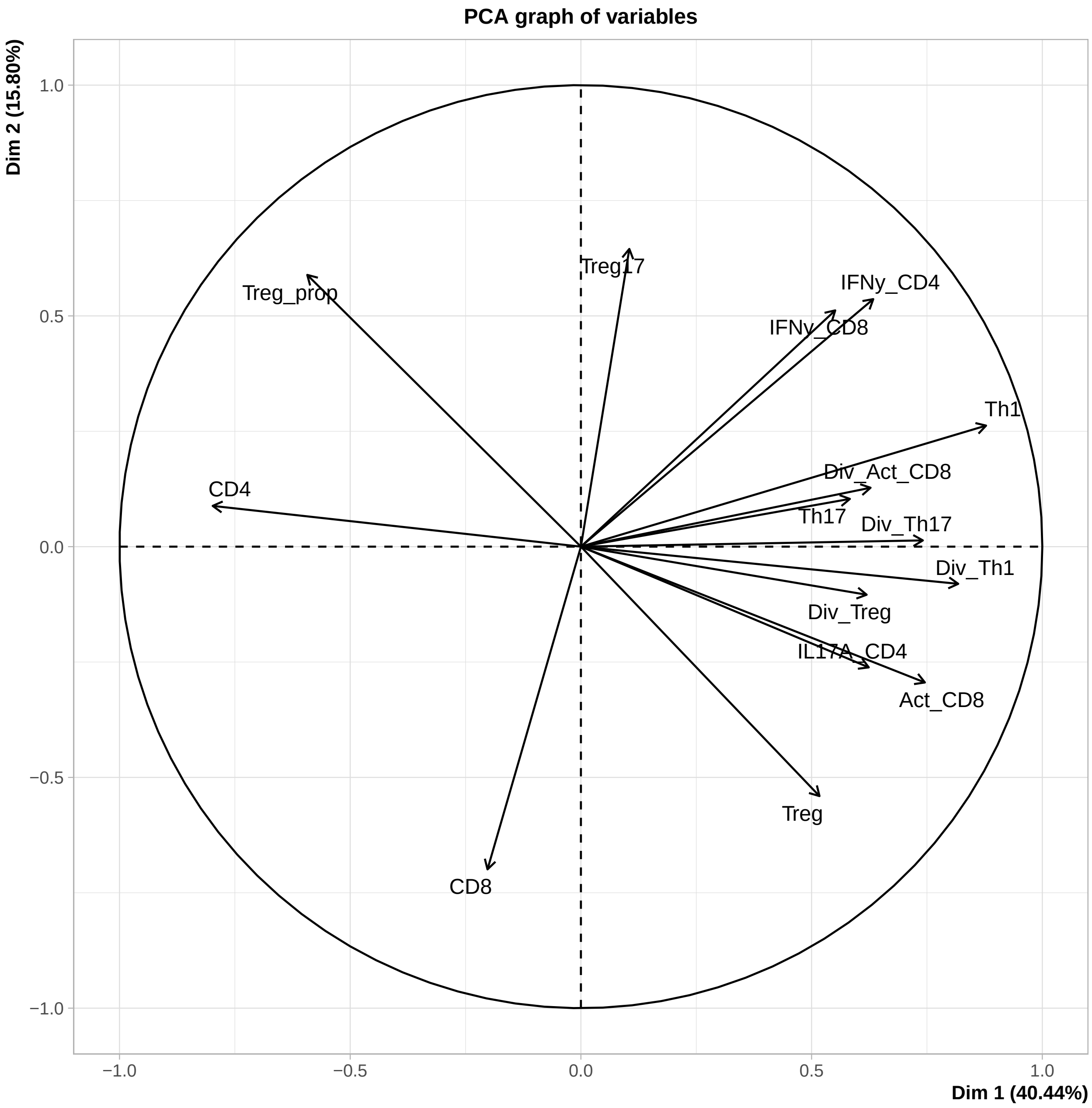


Handling missing data in a pca:

We will now continue by using an iterative pca to impute missing data A. Initialization: impute using the mean B. Step lampda: # a. do pca on imputed data table S dimensions retained # b. missing data imputed using pca # c. means (and standard deviations) updated C. Iterate the estimation and imputation steps (until convergence) (convergence: the act of converging and especially moving toward union or uniformity)

Overfitting is a common problem due to believing too much in links between variables. –> regularized iterative PCA (This version is what is being implented in missMDA) This is a way of taking less risk when imputing the missing data. The algorithm estimates the missing data values with values that have no influence on the PCA results, i.e., no influence on the coordinates of the individuals or variables.


PCA graph of individuals
Dim 2 (15.80%)
Dim 1 (40.44%)
5.0
2.5
0.0
−2.5
−5.0
−3
0
3
6
9
14
7
31
8
2
20
41
4
49
11
30
59
1
3
61
54
19
35
15
66
70
83
22
23
68
32
42
67
78
85
37
6
57
75
29
76
58
62
60
84
53
16
13
82
34
5
77
12
74
26
38
28
55
10
18
63
33
17
39
65
56
80
44
51
69
36
72
24
48
27
81
73
64
71
21
43
46
79
50
45
40
52
47
25



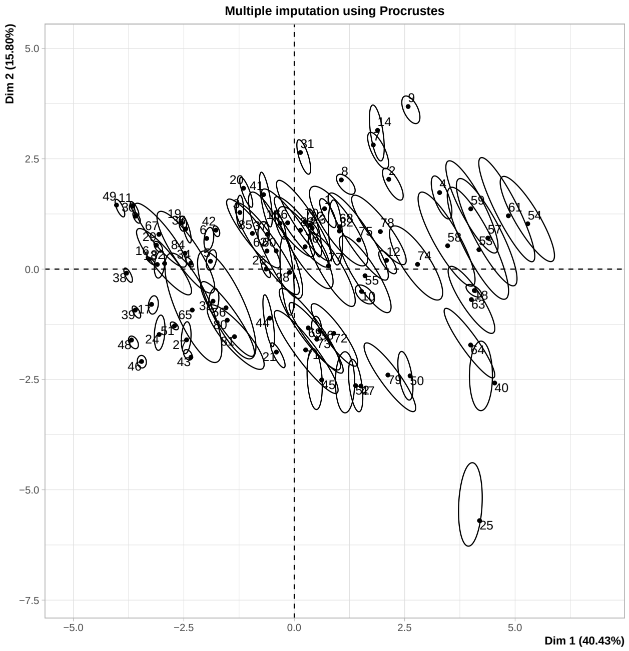
Caution: When imputing data, the percentages of inertia associated with the first dimensions will be overestimated.

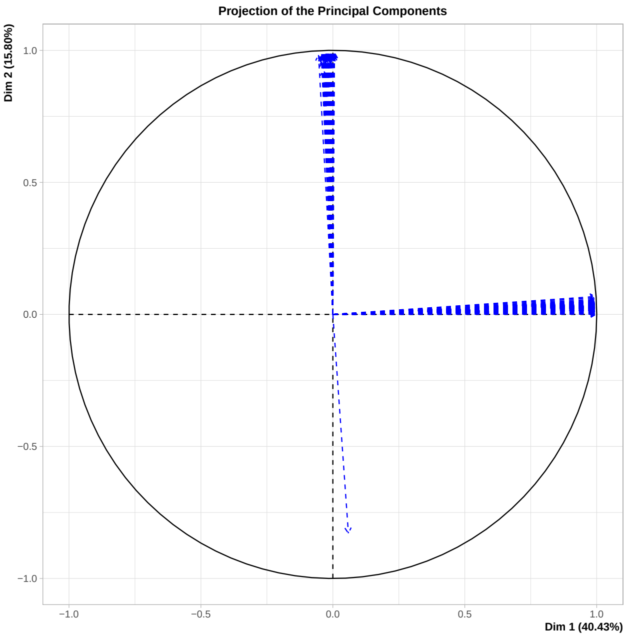
Another problem: the imputed data are, when the pca is performed considered like real observations. But they are estimations!!

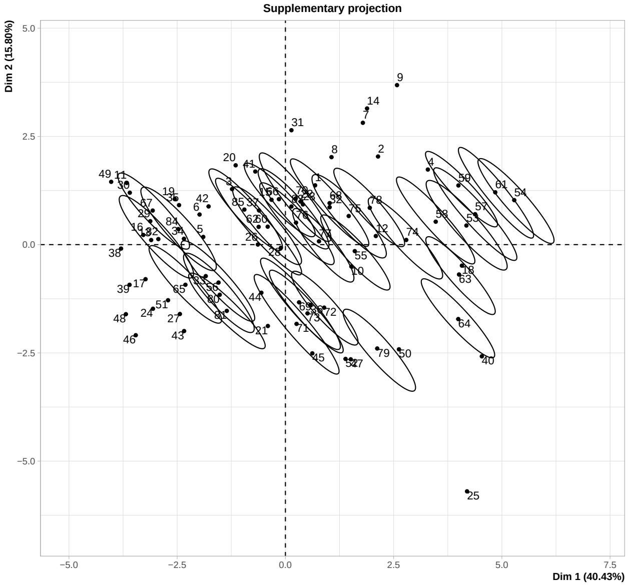
Visualizing uncertainty due to issing data:

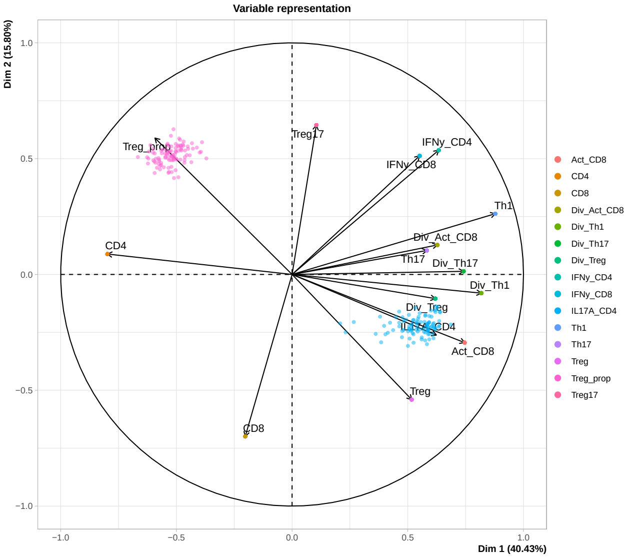
–> mulrimple imputation: facsrate several plausible values for each missing data point

We here visualize the variability, that is uncertainty on the plane defined by two pca axes.

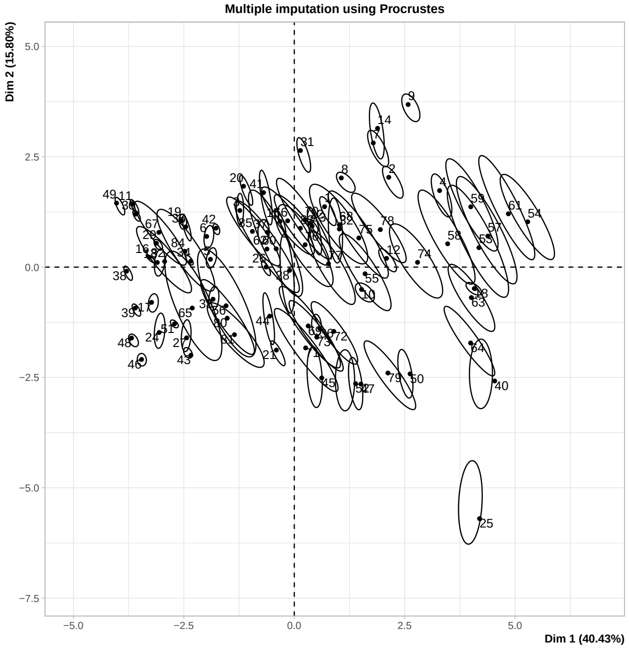




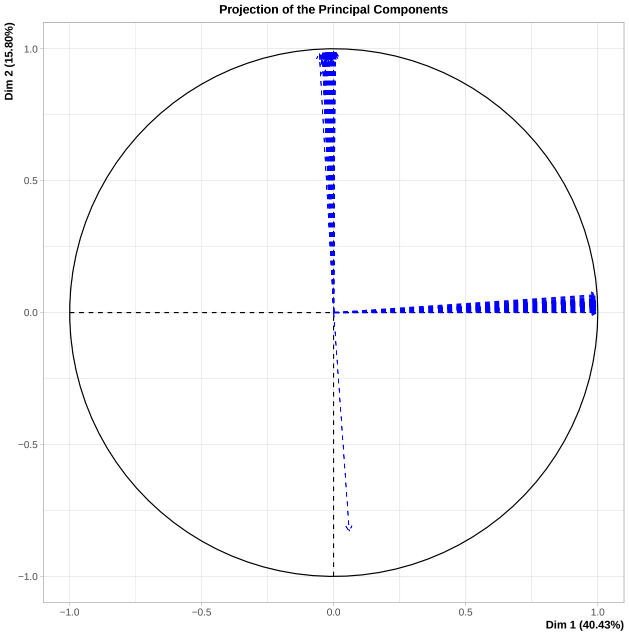




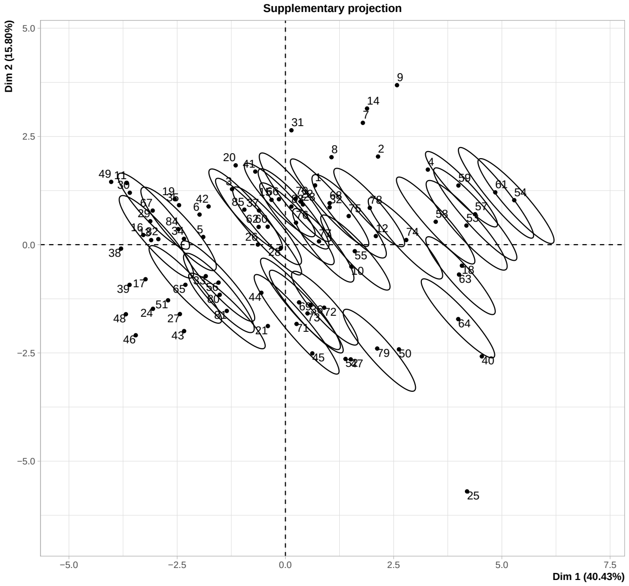
```
## $PlotIndProc
```



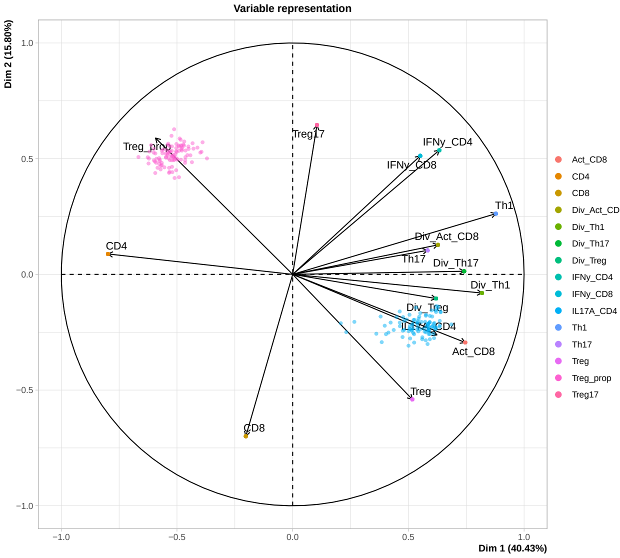
```
##
## $PlotDim
```



```
##
## $PlotIndSupp
```



```
##
## $PlotVar
```



Individuals lying on the axis have no missing data, but individuals that far away have many missing data. big ellipse = big uncertainty tight elipse (line) = low uncertainty

Variable representation: Poins tight together)look like one) - have no missing variables –> low uncertainty Points spread − > higher variability − > higher uncertainty

High uncertainty–> we should interpret the result with care

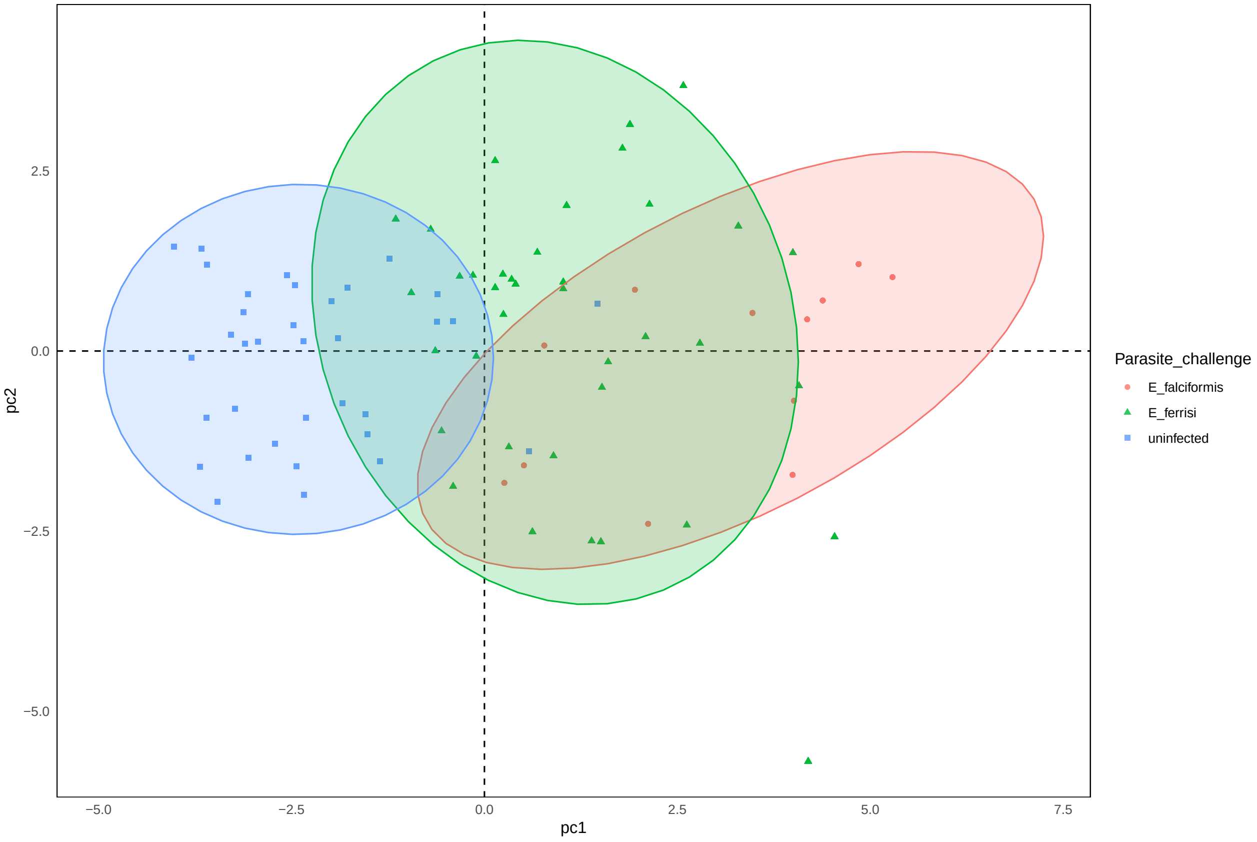
The individuals with many missing data values make the axes move, and thus the positions of all individuals

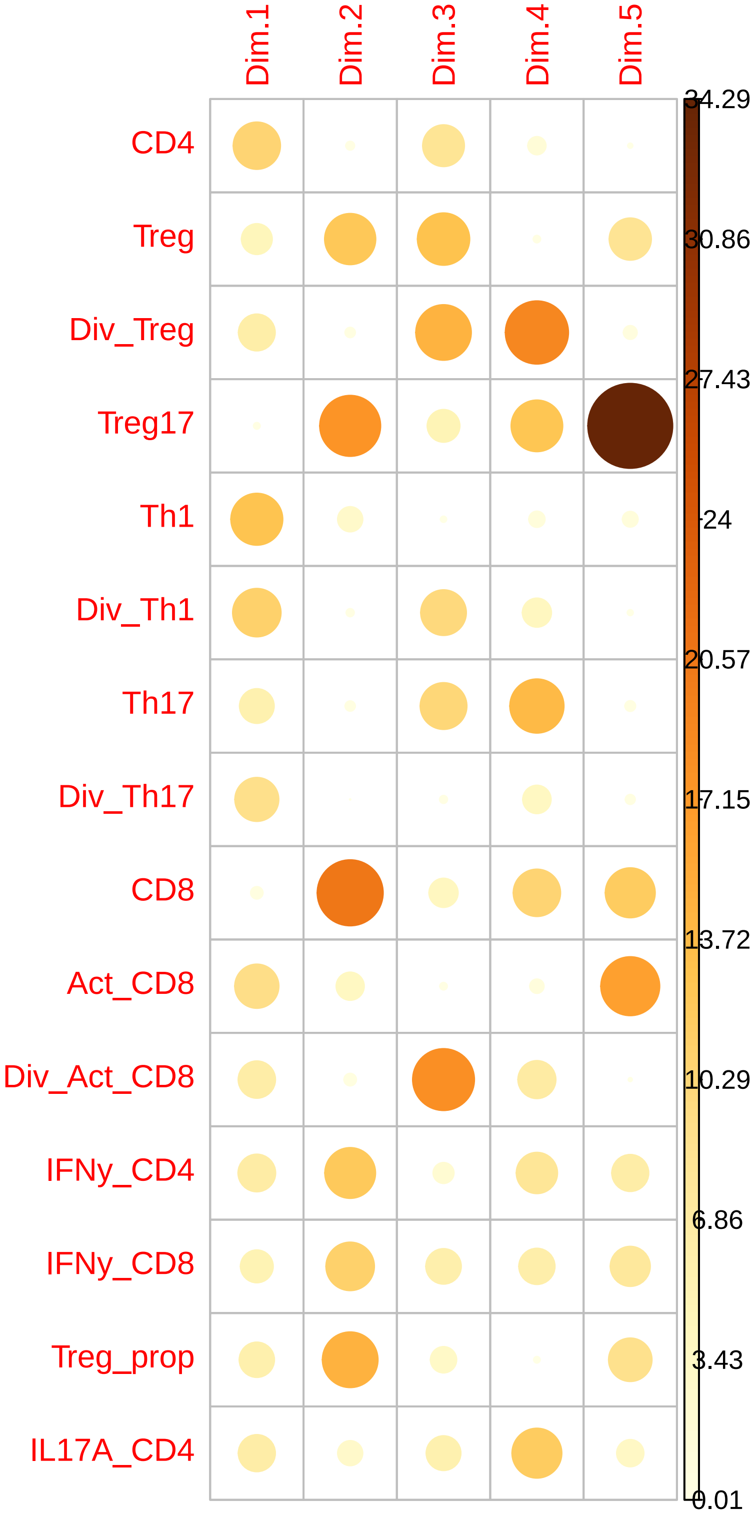
Therefore in the last plots every individual is getting an eclipse as they are as well influenced by the missing data of the others.

THe plot with the dimensions shows the projections of the pca dimensions of each imputed table on the pca plane obtained using the original imputed data table

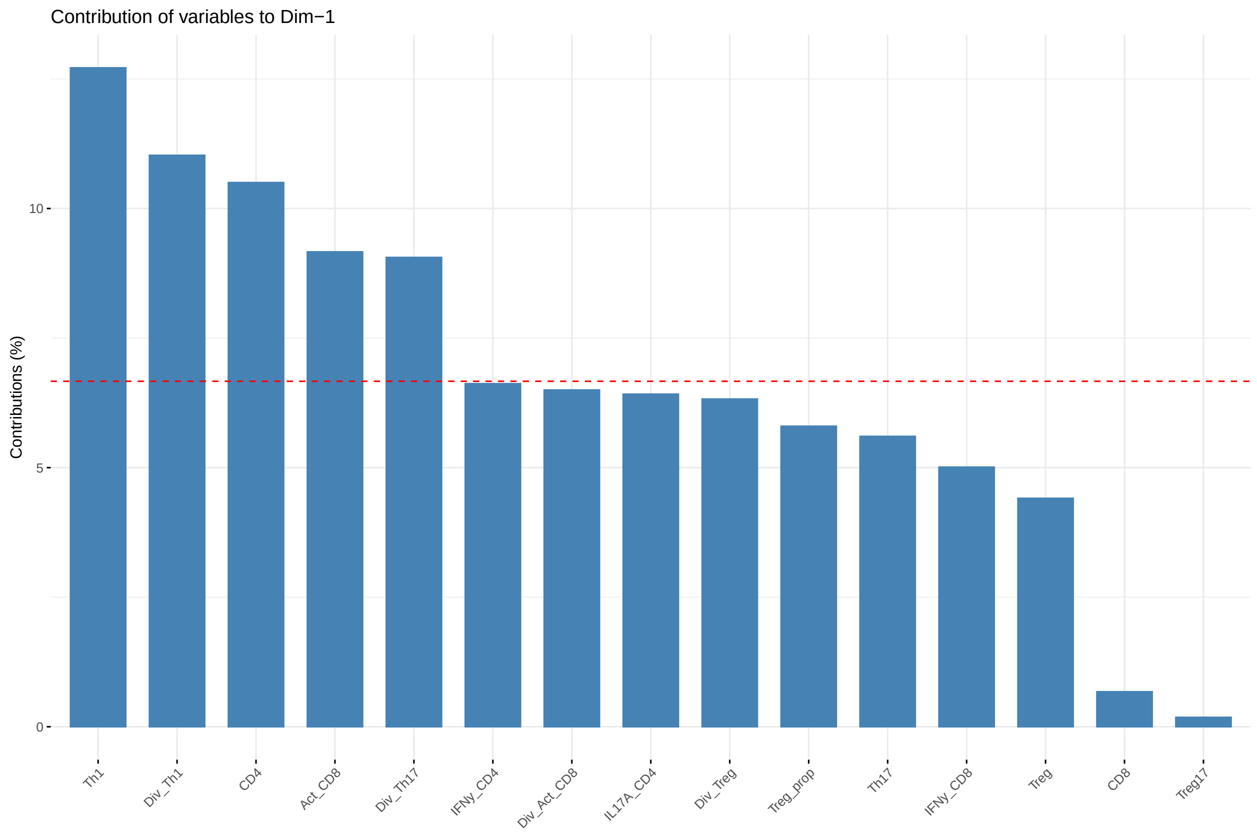
As all of the arrows are close to either the first or second axes, this means that the axes are stable with respect to the set of imputed tables –> we don't have evidence of instability here.

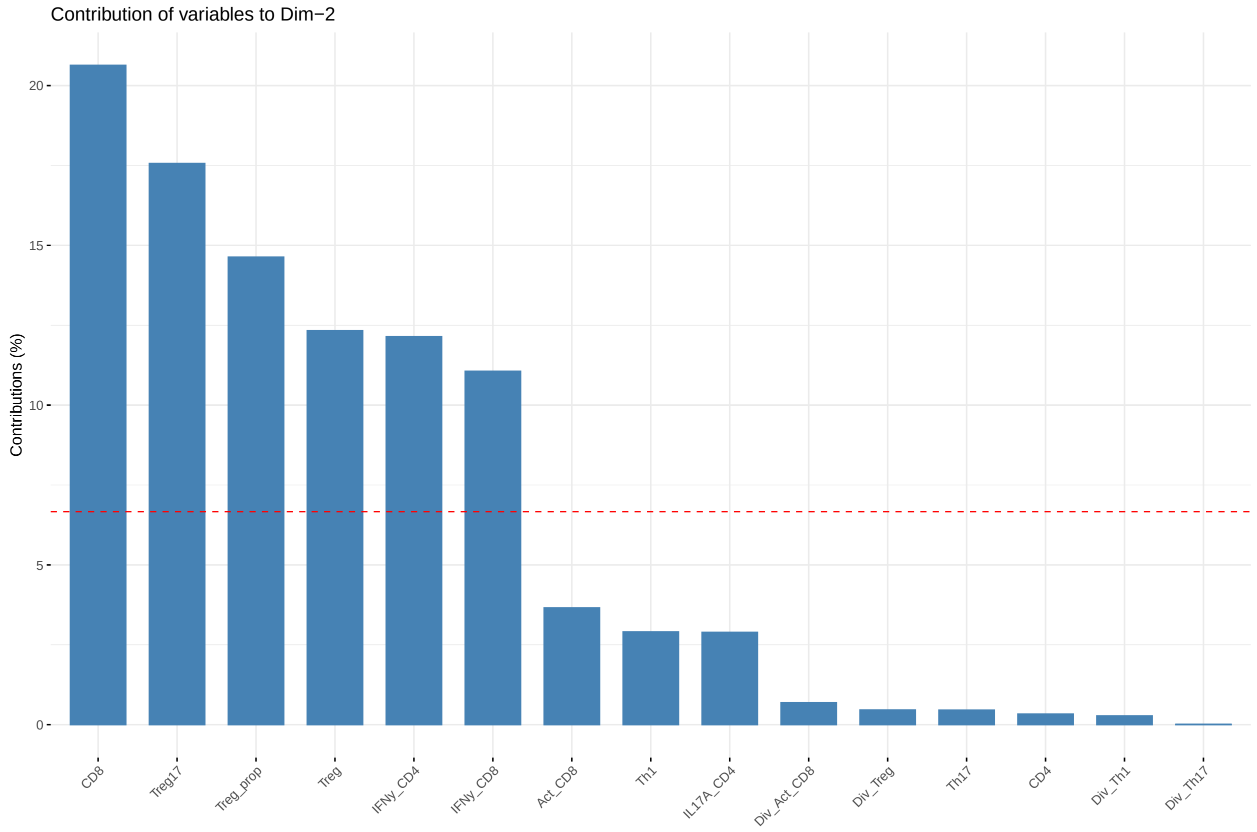
```
#Now we can make our initial plot of the PCA.


imputed_facs %>%
  pivot_longer(10:24, names_to = "Cells", values_to = "Proportion") %>%
  ggplot(aes(x = pc1, y = pc2, color = Parasite_challenge, shape = Parasite_challenge)) +
  geom_hline(yintercept = 0, lty = 2) +
  geom_vline(xintercept = 0, lty = 2) +
  geom_point(alpha = 0.8) +
  stat_ellipse(geom="polygon", aes(fill = Parasite_challenge), alpha = 0.2, show.legend = FALSE,
               level = 0.95) +
  theme_minimal() +
  theme(panel.grid = element_blank(), panel.border = element_rect(fill= "transparent"))
```

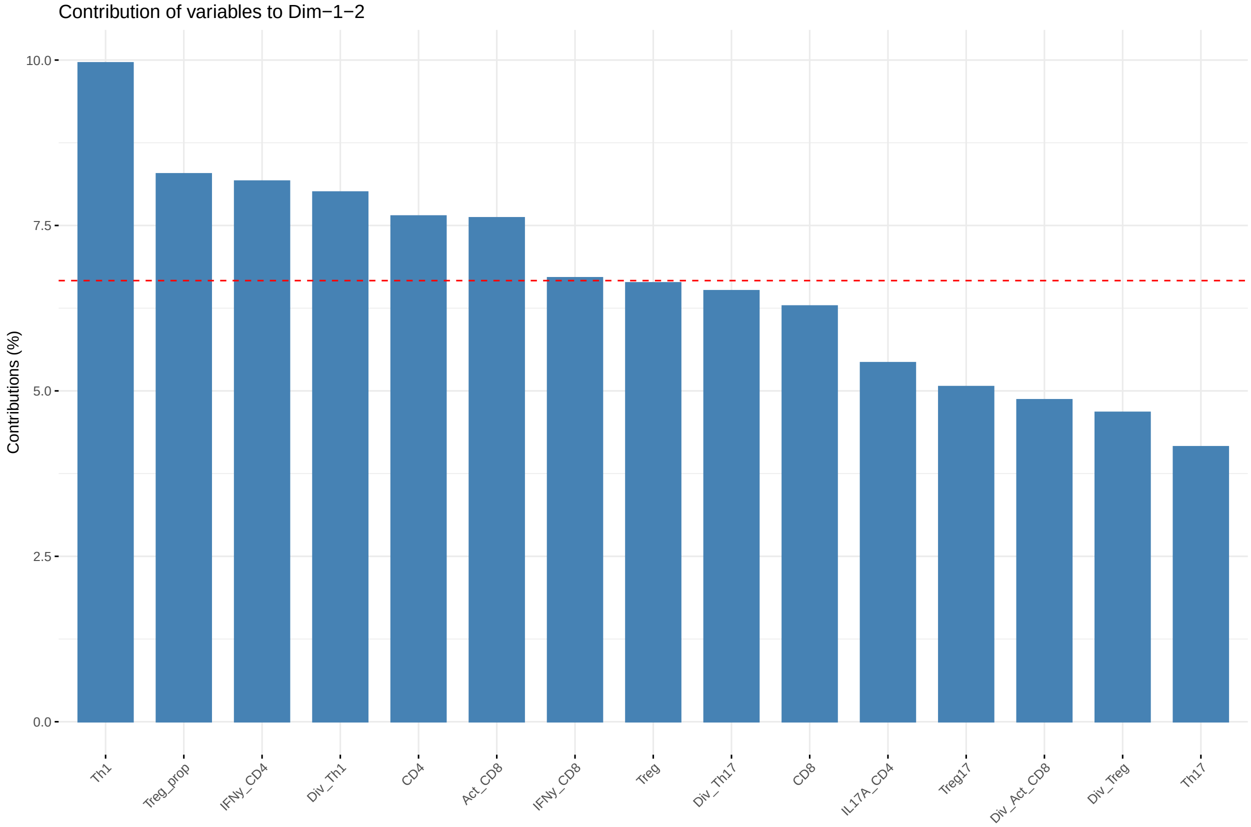




The function fviz_contrib() [factoextra package] can be used to draw a bar plot of variable contributions. If your data contains many variables, you can decide to show only the top contributing variables. The R code below shows the top 10 variables contributing to the principal components:


Contribution of variables to Dim−1
Contributions (%)
10
5
0
Th1
Div_Th1
CD4
Act_CD8
Div_Th17
IFNy_CD4
Div_Act_CD8
IL17A_CD4
Div_Treg
Treg_prop
Th17
IFNy_CD8
Treg
CD8
Treg17

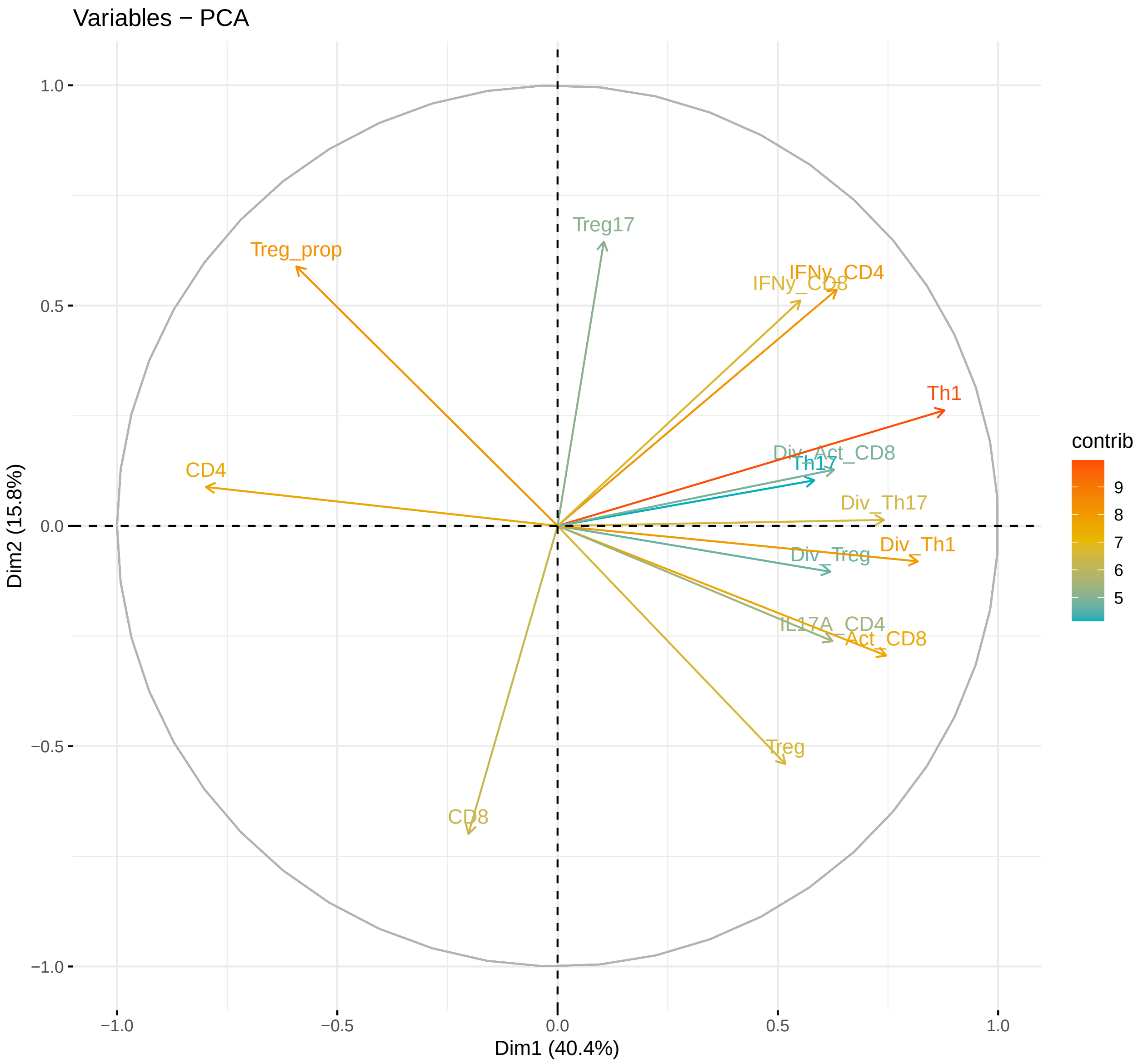

Contribution of variables to Dim−2
Contributions (%)
20
15
10
5
0
CD8
Treg17
Treg_prop
Treg
IFNy_CD4
IFNy_CD8
Act_CD8
Th1
IL17A_CD4
Div_Act_CD8
Div_Treg
Th17
CD4
Div_Th1
Div_Th17



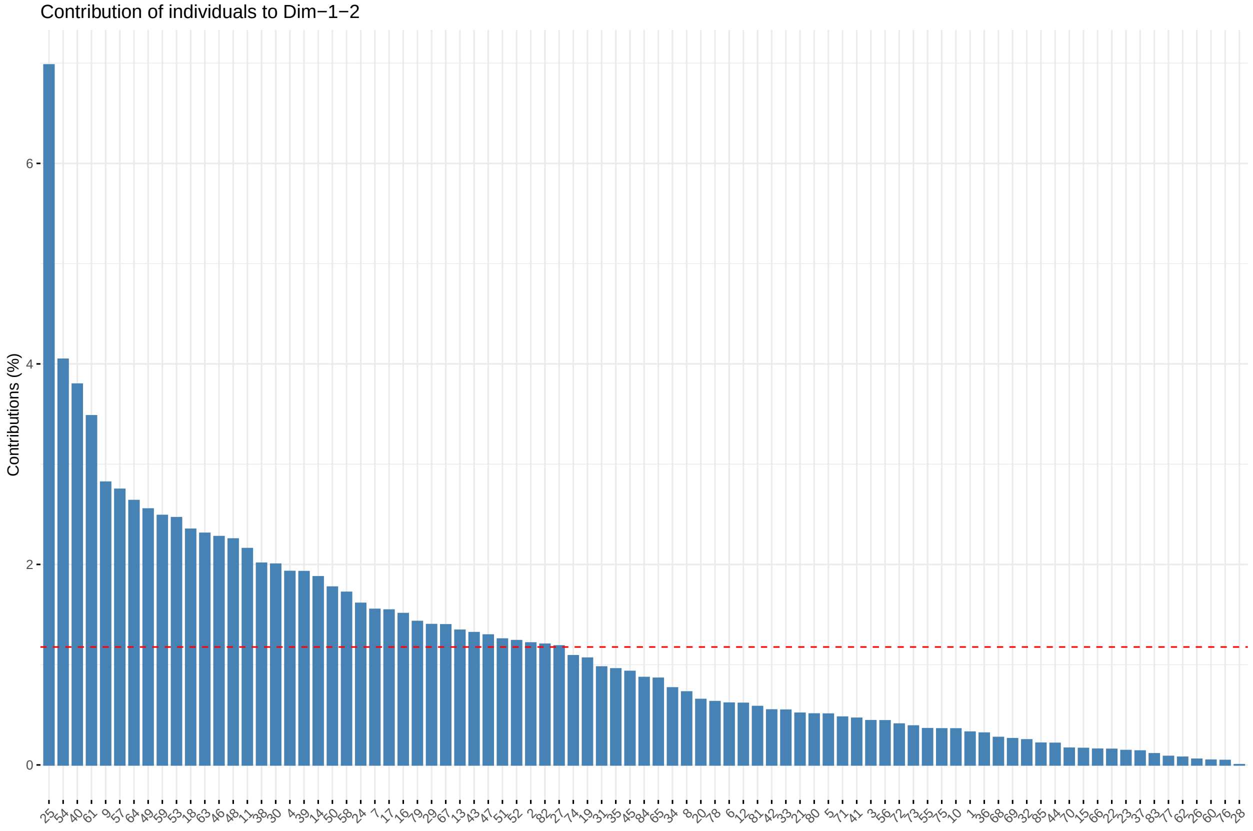
The red dashed line on the graph above indicates the expected average contribution. If the contribution of the variables were uniform, the expected value would be 1/length(variables) = 1/10 = 10%. For a given component, a variable with a contribution larger than this cutoff could be considered as important in contributing to the component.

Note that, the total contribution of a given variable, on explaining the variations retained by two principal components, say PC1 and PC2, is calculated as contrib = [(C1 * Eig1) + (C2 * Eig2)]/(Eig1 + Eig2), where

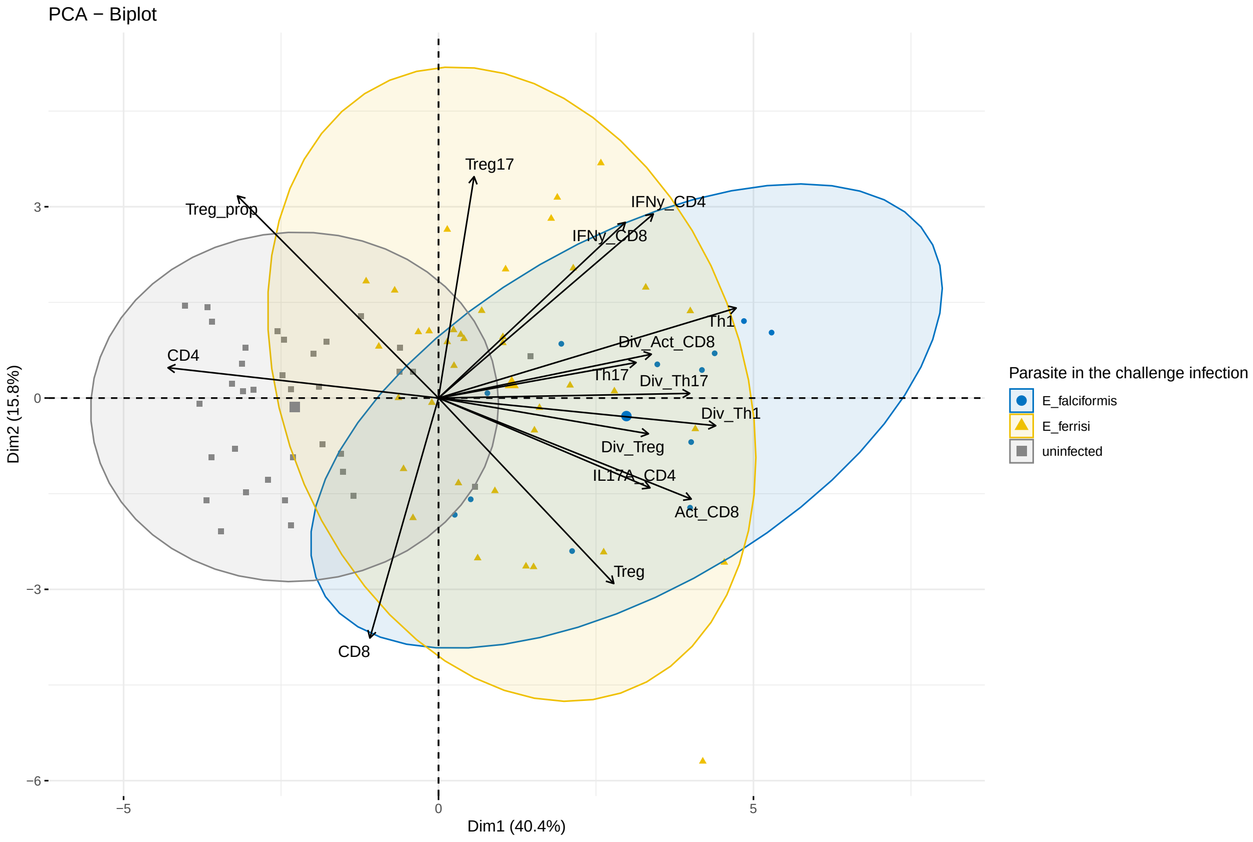
C1 and C2 are the contributions of the variable on PC1 and PC2, respectively Eig1 and Eig2 are the eigenvalues of PC1 and PC2, respectively. Recall that eigenvalues measure the amount of variation retained by each PC. In this case, the expected average contribution (cutoff) is calculated as follow: As mentioned above, if the contributions of the 10 variables were uniform, the expected average contribution on a given PC would be 1/10 = 10%. The expected average contribution of a variable for PC1 and PC2 is : [(10* Eig1) + (10 * Eig2)]/(Eig1 + Eig2)



To visualize the contribution of individuals to the first two principal components:



PCA + Biplot combination



In the following example, we want to color both individuals and variables by groups. The trick is to use pointshape = 21 for individual points. This particular point shape can be filled by a color using the argument fill.ind. The border line color of individual points is set to "black" using col.ind. To color variable by groups, the argument col.var will be used.

Linear models:

```
##
## Call:
## lm(formula = max_WL ~ pc1 + pc2 + Parasite_challenge, data = imputed_facs)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.5521  -2.9263   0.1456   3.6345   9.9540
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   80.9739     1.8237  44.401  < 2e-16 ***
## pc1                            0.9508     0.3772   2.521   0.0137 *
## pc2                           -0.1475     0.3542  -0.416   0.6782
## Parasite_challengeE_ferrisi   11.0501     1.7898   6.174 2.60e-08 ***
## Parasite_challengeuninfected  17.1202     2.5903   6.609 3.95e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.975 on 80 degrees of freedom
## Multiple R-squared:  0.4265, Adjusted R-squared:  0.3978
## F-statistic: 14.87 on 4 and 80 DF,  p-value: 3.976e-09
```

```
## [1] 520.8357
```

```
##
## Call:
## lm(formula = max_WL ~ pc1 + pc2 + Parasite_challenge + hybrid_status,
##     data = imputed_facs)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.9842  -3.1816   0.5616   3.5633   9.4894
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       79.8812     2.1017  38.007  < 2e-16 ***
## pc1                                0.9103     0.4255   2.140   0.0356 *
## pc2                                0.3741     0.5915   0.633   0.5290
## Parasite_challengeE_ferrisi       10.6313     1.9281   5.514 4.76e-07 ***
## Parasite_challengeuninfected      16.6312     2.5658   6.482 8.55e-09 ***
## hybrid_statusF0 M. m. musculus     1.5746     2.5368   0.621   0.5367
## hybrid_statusF1 hybrid             3.5399     1.6592   2.134   0.0362 *
## hybrid_statusF1 M. m. domesticus  -0.9093     2.0039  -0.454   0.6513
## hybrid_statusF1 M. m. musculus     4.2002     2.7063   1.552   0.1249
## hybrid_statusother                 1.6583     2.1713   0.764   0.4474
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.91 on 75 degrees of freedom
## Multiple R-squared:  0.4763, Adjusted R-squared:  0.4134
## F-statistic: 7.578 on 9 and 75 DF,  p-value: 8.217e-08
```

```
## [1] 523.1115
```

```
##
## Call:
## lm(formula = max_WL ~ pc1 + pc2 + hybrid_status, data = imputed_facs)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.0089  -3.8001   0.7976   4.3336  11.4052
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      91.3120     1.4723  62.020   <2e-16 ***
## pc1                              -0.6637     0.3125  -2.124   0.0369 *
## pc2                               0.4622     0.7284   0.635   0.5276
## hybrid_statusF0 M. m. musculus     1.6599     3.1596   0.525   0.6009
## hybrid_statusF1 hybrid            4.5068     2.0539   2.194   0.0312 *
## hybrid_statusF1 M. m. domesticus  -0.5451     2.4849  -0.219   0.8270
## hybrid_statusF1 M. m. musculus     5.0932     3.3323   1.528   0.1305
## hybrid_statusother                0.7101     2.4448   0.290   0.7723
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.118 on 77 degrees of freedom
## Multiple R-squared:  0.1654, Adjusted R-squared:  0.08956
## F-statistic:  2.18 on 7 and 77 DF,  p-value: 0.04505
```

```
## [1] 558.7177
```

```
##
## Call:
## lm(formula = max_WL ~ pc1 + pc2 + infection_history, data = imputed_facs)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5834  -2.8158  -0.4039   3.2253   8.7078
##
## Coefficients:
##                                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)                            84.39648    2.65405  31.799  < 2e-16
## pc1                                     0.98773    0.36148   2.732 0.007858
## pc2                                    -0.09311    0.38599  -0.241 0.810049
## infection_historyfalciformis_ferrisi     7.62628    2.63621   2.893 0.005011
## infection_historyfalciformis_uninfected 15.00027    3.40217   4.409 3.46e-05
## infection_historyferrisi_falciformis    -4.18836    3.02105  -1.386 0.169791
## infection_historyferrisi_ferrisi         9.29476    2.78012   3.343 0.001301
## infection_historyferrisi_uninfected     13.15007    3.36444   3.909 0.000204
## infection_historyuninfected             13.65829    3.90798   3.495 0.000805
## infection_historyuninfected_falciformis -8.53966    3.95493  -2.159 0.034073
## infection_historyuninfected_ferrisi     -1.22495    3.30747  -0.370 0.712174
##
## (Intercept)                            ***
## pc1                                    **
## pc2
## infection_historyfalciformis_ferrisi    **
## infection_historyfalciformis_uninfected ***
## infection_historyferrisi_falciformis
## infection_historyferrisi_ferrisi        **
## infection_historyferrisi_uninfected     ***
## infection_historyuninfected             ***
## infection_historyuninfected_falciformis *
## infection_historyuninfected_ferrisi
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.48 on 74 degrees of freedom
## Multiple R-squared:  0.5699, Adjusted R-squared:  0.5117
## F-statistic: 9.804 on 10 and 74 DF,  p-value: 3.216e-10
```

```
## [1] 508.3793
```

```
##
## Call:
## lm(formula = max_WL ~ pc1 + pc2, data = imputed_facs)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.168  -3.267   1.382   4.218   8.878
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  92.9634     0.6727 138.191  < 2e-16 ***
## pc1          -0.7582     0.2732  -2.776  0.00682 **
## pc2          -0.1078     0.4370  -0.247  0.80584
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.202 on 82 degrees of freedom
## Multiple R-squared:  0.08651,    Adjusted R-squared:  0.06423
## F-statistic: 3.883 on 2 and 82 DF,  p-value: 0.02448
```

```
##                    df      AIC
## weight_lm           6 520.8357
## weight_lm_exp_only  4 556.3975
```

repeating the heatmap on the now imputed data

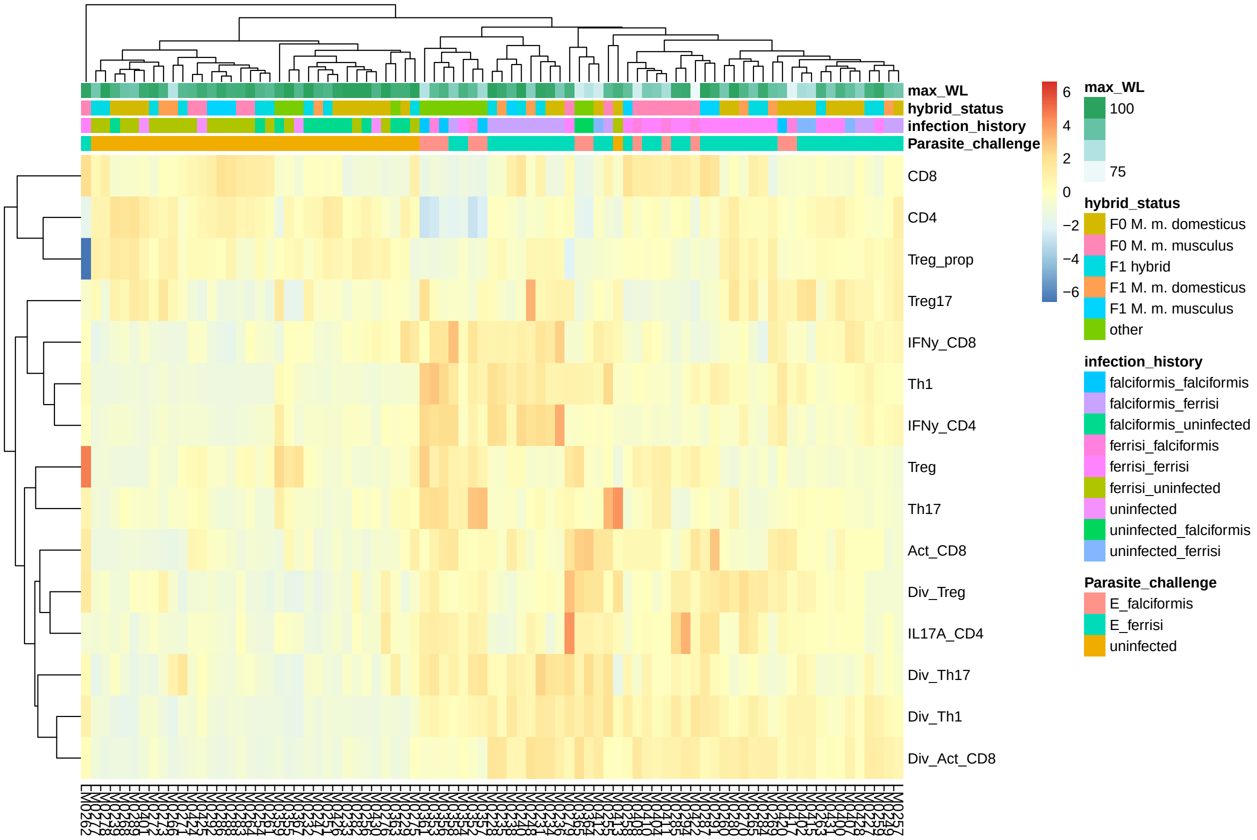
```
facs <- imputed_facs %>%
  dplyr::select(c(EH_ID, all_of(CellCount.cols))) %>%
  dplyr::select(-Position)

 # turn the data frame into a matrix and transpose it. We want to have each cell
 # type as a row name
 facs <- t(as.matrix(facs))

 #switch the matrix back to a data frame format
 facs <- as.data.frame(facs)
```

```
# turn the first row into column names
facs %>%
    row_to_names(row_number = 1) -> heatmap_data


table(rowSums(is.na(heatmap_data)) == nrow(heatmap_data))
```

```
##
## FALSE
##    15
```

```
# turn the columns to numeric other wise the heatmap function will not work
heatmap_data[] <- lapply(heatmap_data, function(x) as.numeric(as.character(x)))
# remove columns with only NAs
heatmap_data <- Filter(function(x)!all(is.na(x)), heatmap_data)

#remove rows with only Nas
heatmap_data <-  heatmap_data[, colSums(is.na(heatmap_data)) != nrow(heatmap_data)]
rownames(annotation_df) <- colnames(heatmap_data)
```

Heatmap on facs expression data:


max_WL
hybrid_status
infection_history
Parasite_challenge
CD8
CD4
Treg_prop
Treg17
IFNy_CD8
Th1
IFNy_CD4
Treg
Th17
Act_CD8
Div_Treg
IL17A_CD4
Div_Th17
Div_Th1
Div_Act_CD8
max_WL
hybrid_status
F0 M. m. domesticus
F0 M. m. musculus
F1 hybrid
F1 M. m. domesticus
F1 M. m. musculus
other
infection_history
falciformis_falciformis
falciformis_ferrisi
falciformis_uninfected
ferrisi_falciformis
ferrisi_ferrisi
ferrisi_uninfected
uninfected
uninfected_falciformis
uninfected_ferrisi
Parasite_challenge
E_falciformis
E_ferrisi
uninfected