Latent Topic Modeling on Twitter Data

Method: CA

Correspondence Analysis (CA) is a multivariate graphical technique designed to explore relationships among categorical variables. The outcome from correspondence analysis is a graphical display of the rows and columns of a contingency table that is designed to permit visualization of the salient relationships among the variable responses in a low-dimensional space. Such a representation reveals a more global picture of the relationships among row-column pairs which would otherwise not be detected through a pairwise analysis.

Calculate CA:

- Step 1: Compute row and column averages
- Step 2: Compute the expected values
- Step 3: Compute the residuals
- Step 4: Plotting labels with similar residuals close together
- Step 5: Interpreting the relationship between row and column labels

How to Interpret Correspondence Analysis Plots

Correspondence analysis does not show us which rows have the highest numbers, nor which columns have the highest numbers. It instead shows us the relativities.

- The further things are from the origin, the more discriminating they are.
- Look at the length of the line connecting the row label to the origin. Longer lines indicate that the row label is highly associated with some of the column labels (i.e., it has at least one high residual).
- Look at the length of the label connecting the column label to the origin. Longer lines again indicate a high association between the column label and one or more row labels.
- Look at the angle formed between these two lines. Really small angles indicate association. 90 degree angles indicate no relationship. Angles near 180 degrees indicate negative associations.

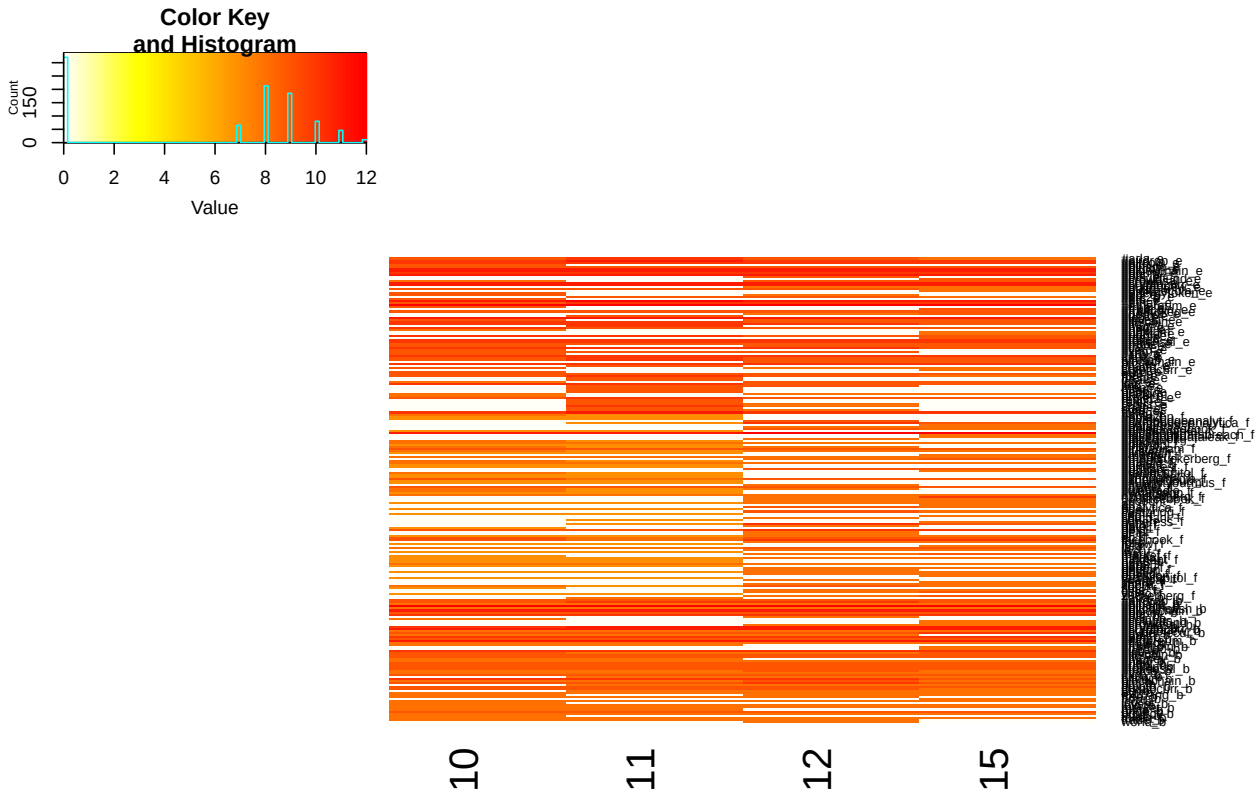
Dataset

```
##            10 11 12 15
## #ada_e      9 10  9  8
## #airdrop_e 10 11 10 10
## #altcoin_e 10 10 10 10
## #bch_e      9  0  0  0
## #binanc_e   9  9  9  9
## #bitcoin_e 11 11 11 11
```

- Research Question
  - Do we see new words appearing on a particular week

Analysis

Heatmap

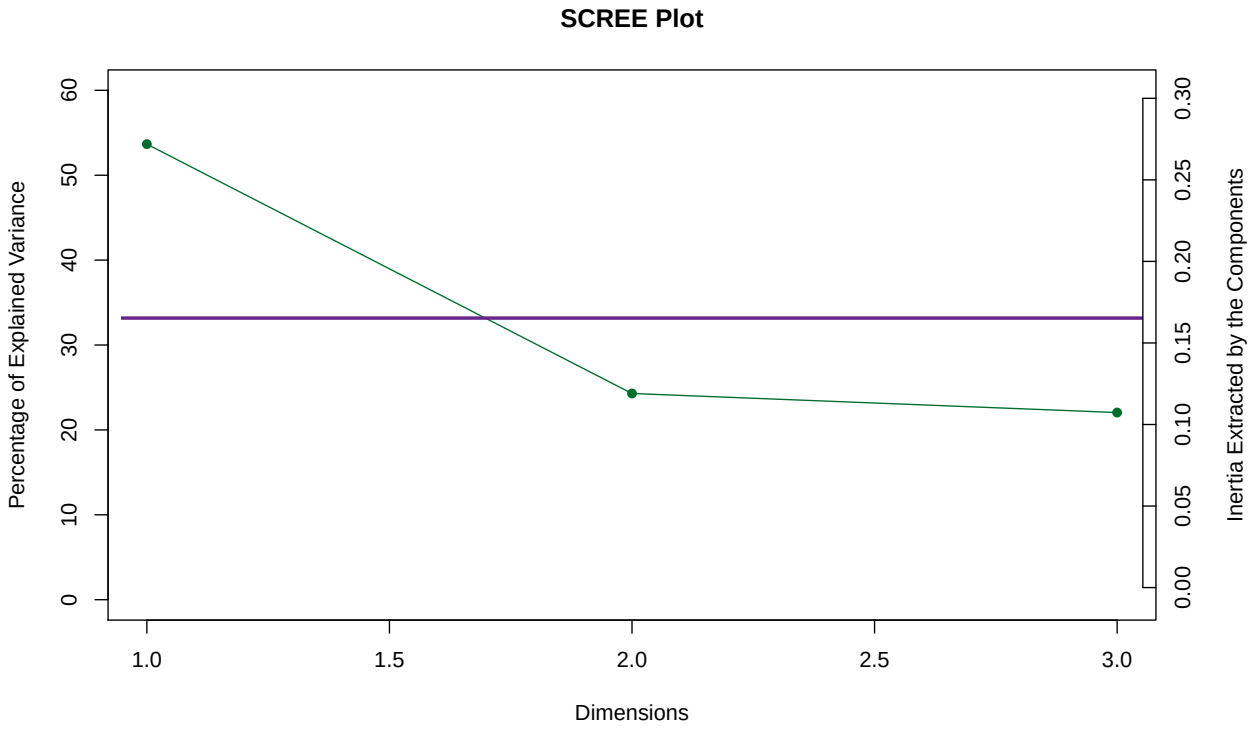


Scree Plot

Gives amount of information explained by corresponding component. Gives an intuition to decide which components best represent data in order to answer the research question.

P.S. The most contribution component may not always be most useful for a given research question.

```
PTCA4CATA::PlotScree(ev = resCA.sym$ExPosition.Data$eigs,
                     #p.ev =  we_data_inf$Inference.Data$components$p.vals,
                     title = 'SCREE Plot',
                     plotKaiser = TRUE
)
```

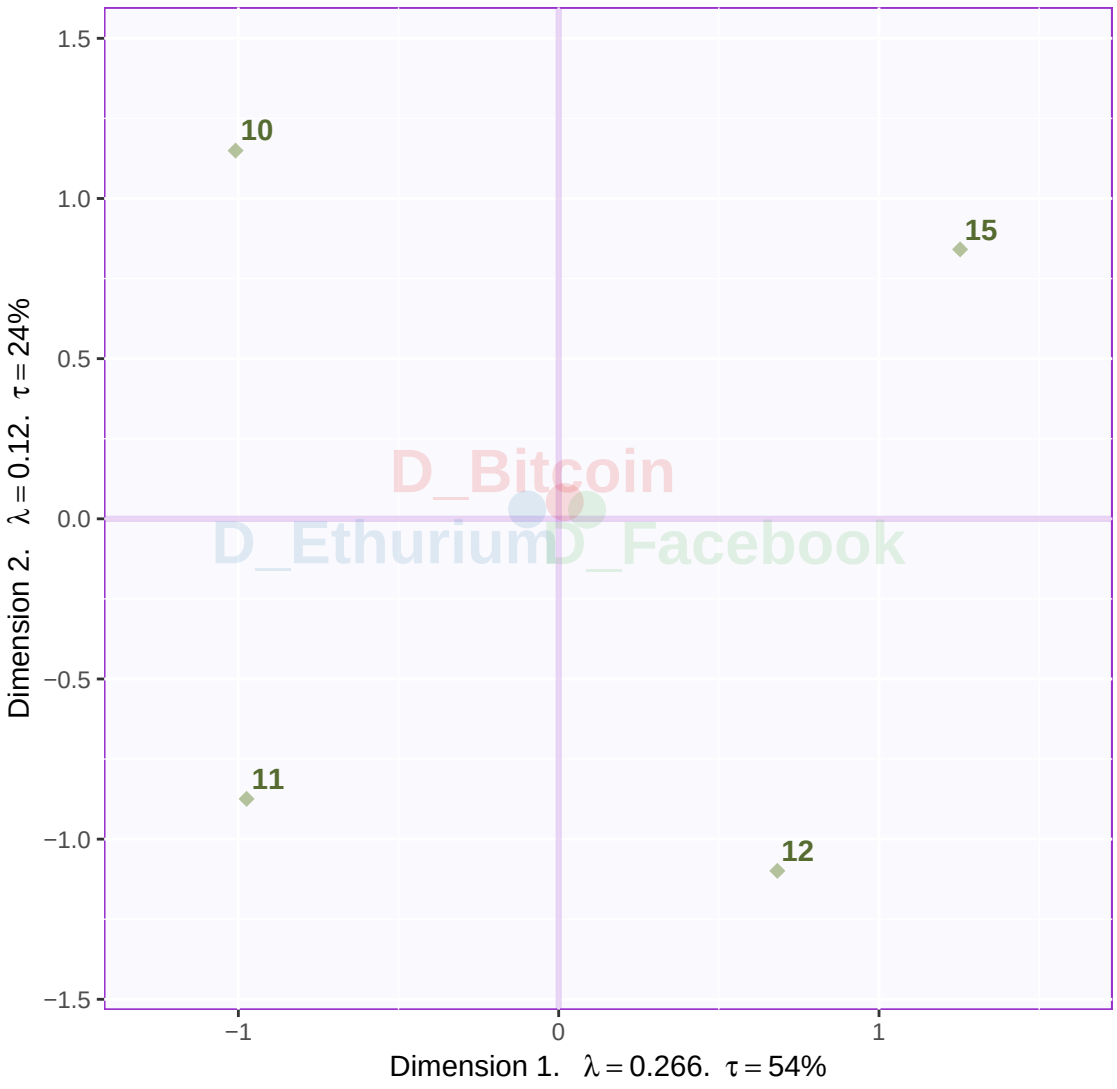


Factor Scores

Asymmetric Plot

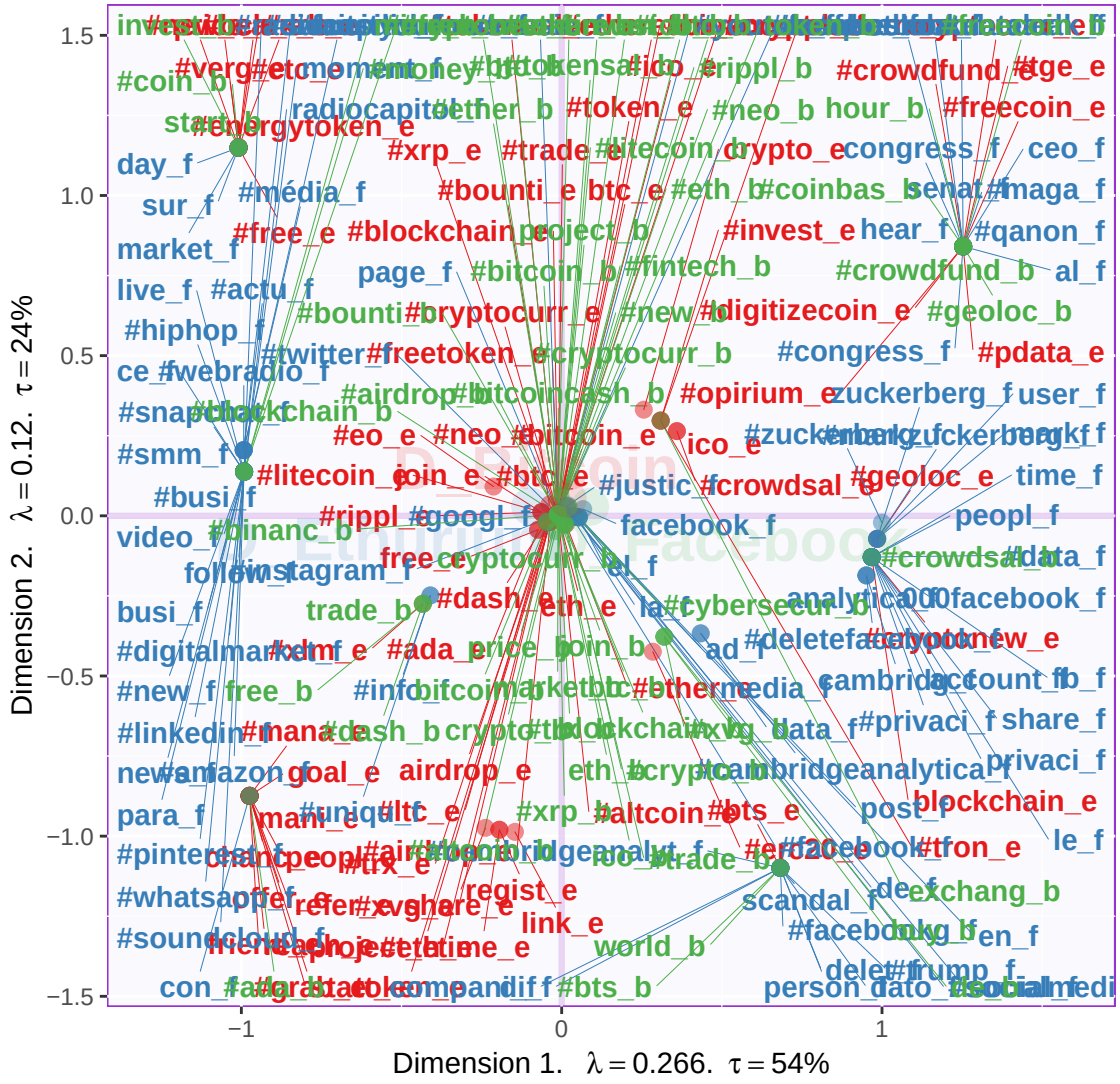
```
#### Asymmetric Plot

map.IJ.asym <- asymMap$baseMap + asymMap$J_labels +
  asymMap$J_points + labels4CA + legend$zeMap_dots + legend$zeMap_text
print(map.IJ.asym)
```




```
#### Asymmetric Plot

map.IJ.asym <- asymMap$baseMap + asymMap$I_labels +
  asymMap$I_points + labels4CA + legend$zeMap_dots + legend$zeMap_text
print(map.IJ.asym)
```

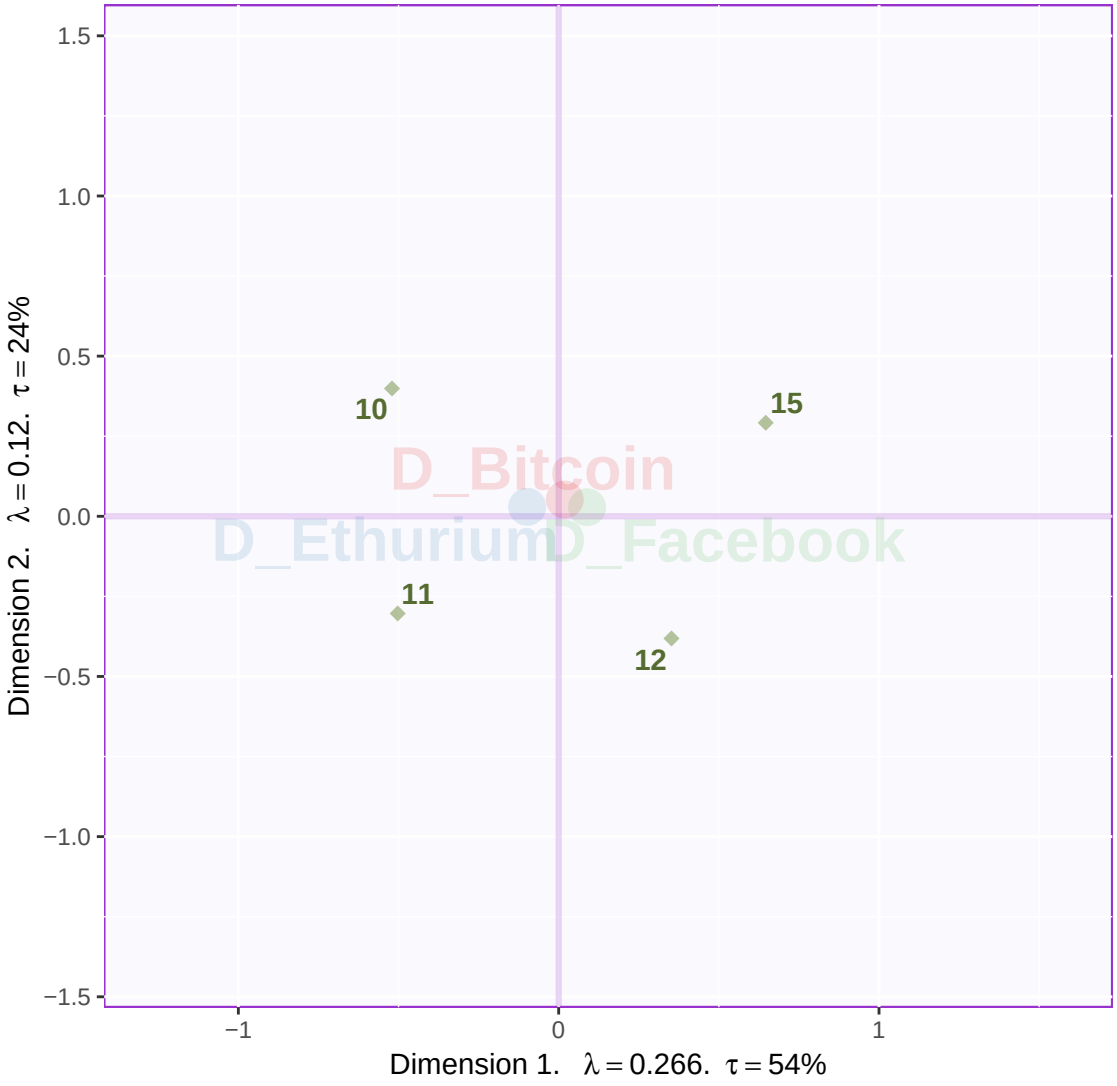


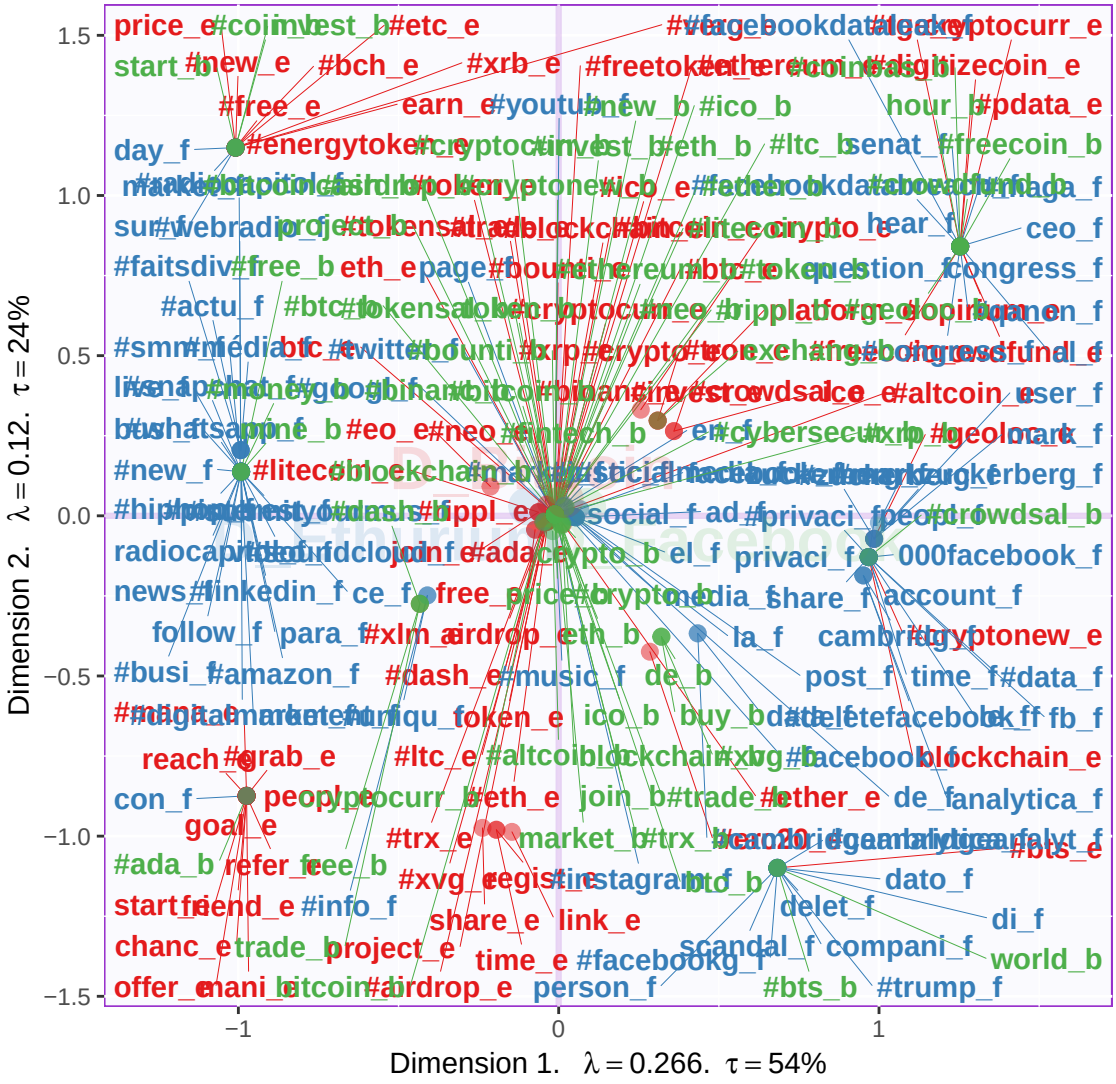

```
#### Asymmetric Plot

map.IJ.asym <- asymMap$baseMap + asymMap$I_labels +
  asymMap$I_points + asymMap$J_labels +
  asymMap$J_points + labels4CA + legend$zeMap_dots + legend$zeMap_text
print(map.IJ.asym)
```


Dimension 1. λ = 0.266. τ = 54%
Dimension 2. λ = 0.12. τ = 24%
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−1
0
1

Symmetric Plot

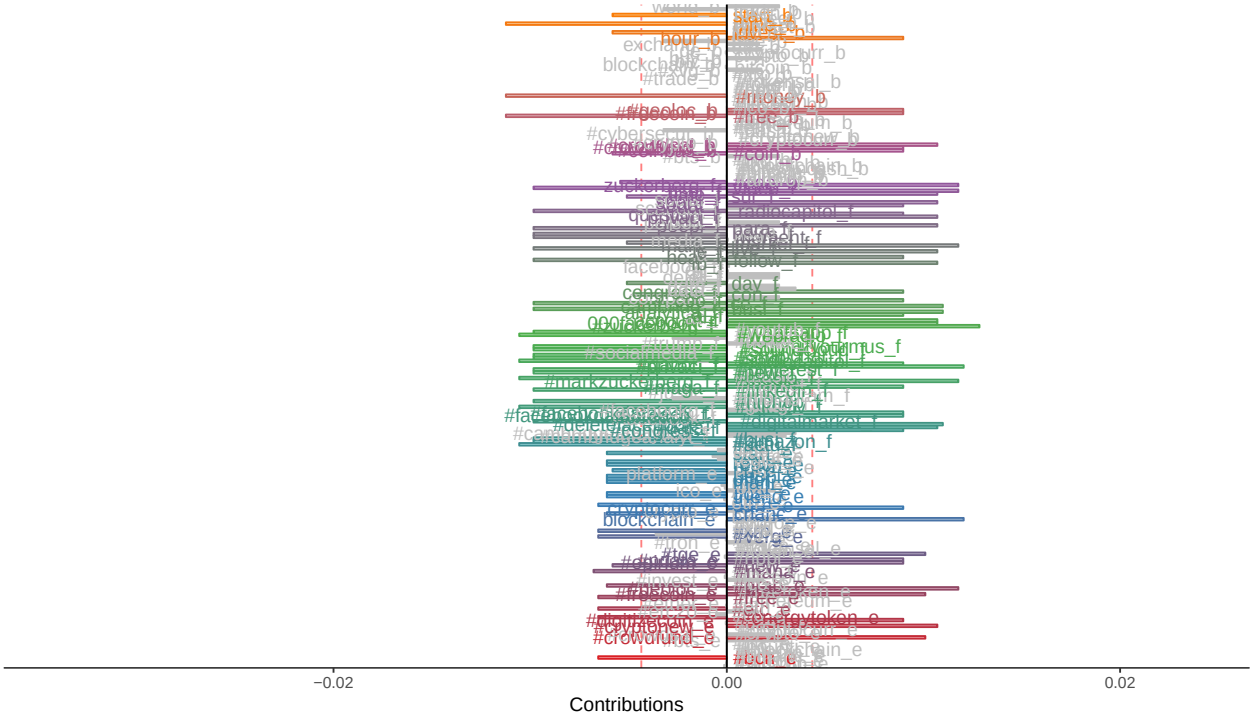




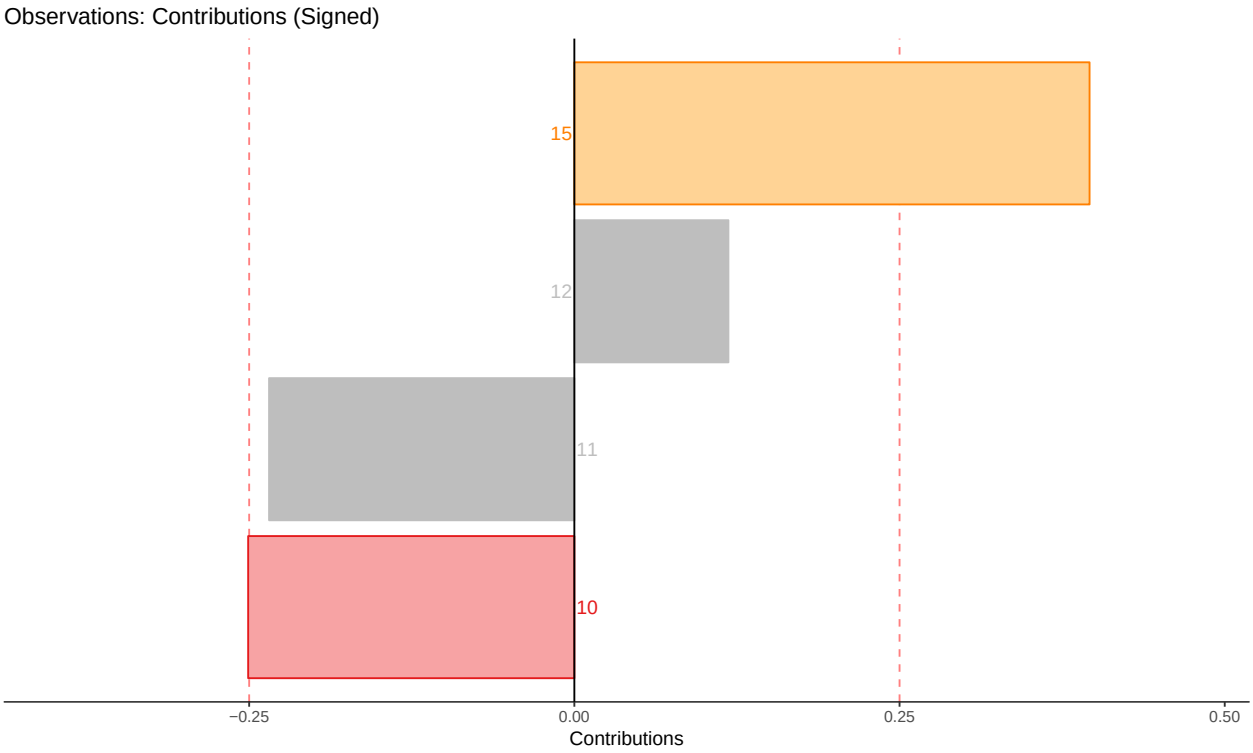
Most Contributing Variables

```
PTCA4CATA::PrettyBarPlot2(ctr.I[,1],
                          threshold = 1 / NROW(ctr.I),
                          font.size = 4,
                          color4bar = gplots::col2hex(color4I),
                          color4ns = 'grey',
                          main = 'Observations: Contributions (Signed)',
                          ylab = 'Contributions', ylim = c(1.2*min(ctr.I),
                           1.2*max(ctr.I) ),
                          horizontal = FALSE )
```

Observations: Contributions (Signed)



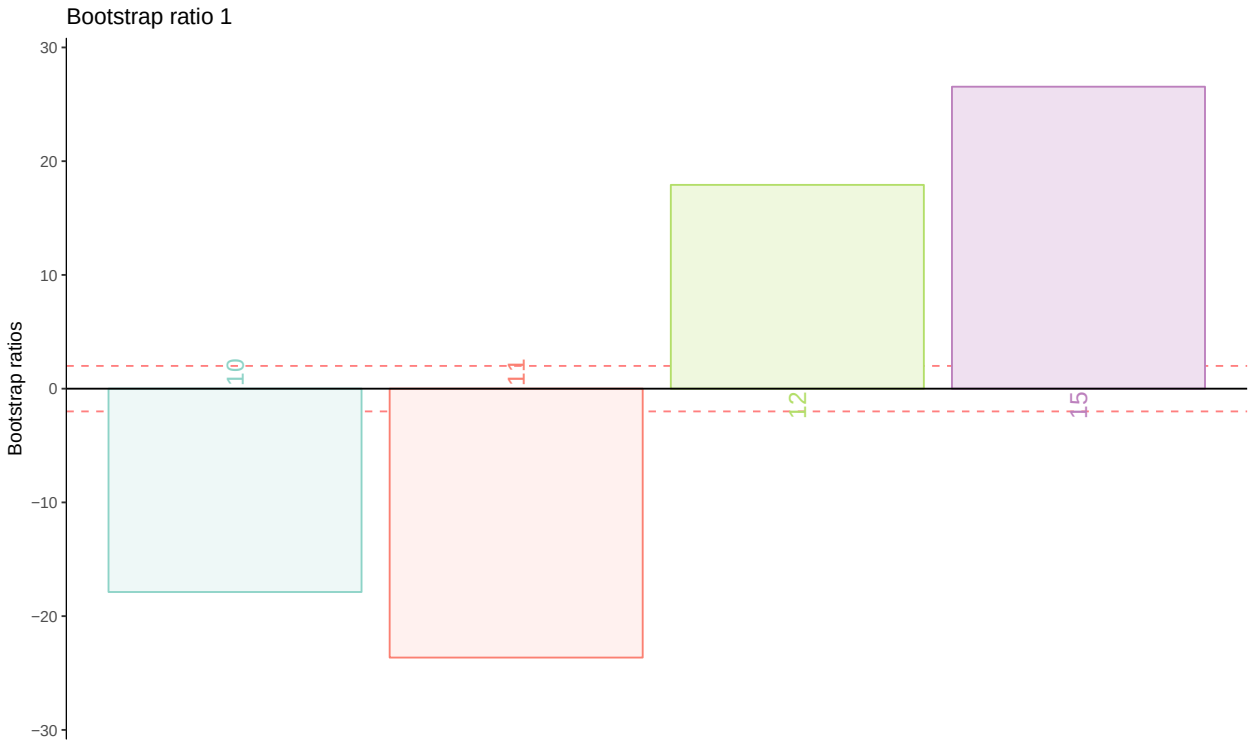

```
PTCA4CATA::PrettyBarPlot2(ctr.J[,1],
                          threshold = 1 / NROW(ctr.J),
                          font.size = 4,
                          color4bar = gplots::col2hex(color4J),
                          color4ns = 'grey',
                          main = 'Observations: Contributions (Signed)',
                          ylab = 'Contributions', ylim = c(1.2*min(ctr.J),
                           1.2*max(ctr.J) ),
                          horizontal = FALSE )
```



Inference CA

```
ba001.BR1 <- PrettyBarPlot2(BR[,laDim],
                            threshold = 2,
                            font.size = 5,
                            color4bar = gplots::col2hex(col4J), # we need hex code
                            main = paste0('Bootstrap ratio ',laDim),
                            ylab = 'Bootstrap ratios',
                            ylim = c(60*min(BR[,laDim]), 1.2*max(BR[,laDim]))
)
print(ba001.BR1)
```


```r
wedata.BR1 <- PrettyBarPlot2(BR[,laDim],
                             threshold = 2,
                             font.size = 5,
                             color4bar = gplots::col2hex(col4J), # we need hex code
                             main = paste0('Bootstrap ratio ',laDim),
                             ylab = 'Bootstrap ratios'
                             #ylim = c(1.2*min(BR[,laDim]), 1.2*max(BR[,laDim]))
)
print(wedata.BR1)
```


Bootstrap ratio 1
Bootstrap ratios
30
20
10
0
−10
−20
−30
10
11
12
15