Country Environment Factors correlated with Happiness

Ritesh Malaiya

2018-11-22

# Introduction

## Context

Assessing country-level social and economic statistics are often limited to socio-economic data. Not any more! This dataset will be maintained and updated with miscellaneous environmental data for countries across the globe.

## Content

This data is all acquired through Google Earth Engine (<https://earthengine.google.com/>) where publicly available remote sensing datasets have been uploaded to the cloud to be manipulated by the average Joe like you and I. Most of the data is derived by calculating the mean for each country at a reduction scale of about 10km.

## Inspiration

Can you use environmental statistics to predict social and economic data? Are people more happy in sunny countries? How do economies in forested countries compare with those dominated by grassland/desert?

## Research Question (scope of this book)

How do the 137 countries differ on these variables?

# Dataset

* Data: Measurements of environment conditions in Countries
* Rows: There are 137 observations, 1 for each country.
* Columns: Total 29 variables
* Qualitative: Country (nominal), Happiness (Ordinal).
* Quantitative: Aspect, Slope Crop Land, Tree Canopy Wind Cloud & Multiple variables for Temp & Rain

## Structure of Data

## Structure of Data

## 'data.frame': 137 obs. of 29 variables:  
## $ Country : Factor w/ 137 levels "Afghanistan",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ Happiness\_Rank : Ord.factor w/ 3 levels "VH"<"H"<"U": 3 2 2 3 1 3 1 1 2 2 ...  
## $ accessibility\_to\_cities: num 317.7 73.8 1212.8 378.2 209.2 ...  
## $ elevation : num 1832 652 557 1061 683 ...  
## $ aspect : num 201 192 185 174 145 ...  
## $ slope : num 1.516 1.89 0.171 0.193 0.624 ...  
## $ cropland\_cover : num 9.51 23.35 3.69 2.79 21.96 ...  
## $ tree\_canopy\_cover : num 0.375 12.805 0.177 19.87 8.834 ...  
## $ isothermality : num 35.9 33.2 40.3 64.3 49.9 ...  
## $ rain\_coldestQuart : num 128.72 392.51 25.29 8.05 79.09 ...  
## $ rain\_driestMonth : num 1.722 40.088 0.935 0.26 17.183 ...  
## $ rain\_driestQuart : num 8.3 138.15 6.09 4.43 60.49 ...  
## $ rain\_mean\_annual : num 311.3 1151.1 79.5 1023.4 539.9 ...  
## $ rain\_seasonailty : num 91.6 38.5 67.1 91.5 48.3 ...  
## $ rain\_warmestQuart : num 12.69 138.33 9.51 318.54 183.14 ...  
## $ rain\_wettestMonth : num 67.8 159 13.4 202.2 79.2 ...  
## $ rain\_wettestQuart : num 175.8 435.9 33.3 524.3 211.7 ...  
## $ temp\_annual\_range : num 40.3 27.1 36.5 21.5 26.8 ...  
## $ temp\_coldestQuart : num -0.261 3.58 13.152 18.794 8.024 ...  
## $ temp\_diurnal\_range : num 14.72 9.11 14.87 13.85 13.46 ...  
## $ temp\_driestQuart : num 21.1 19.6 26.9 18.9 11.1 ...  
## $ temp\_max\_warmestMonth : num 32 26.3 41.5 31 28.2 ...  
## $ temp\_mean\_annual : num 11.5 11.5 23 21.6 14.2 ...  
## $ temp\_min\_coldestMonth : num -8.312 -0.806 5.058 9.549 1.443 ...  
## $ temp\_seasonality : num 88.2 62.7 75.1 18.5 47.6 ...  
## $ temp\_warmestQuart : num 22.7 19.6 32.5 23.3 20.2 ...  
## $ temp\_wettestQuart : num 3.95 5.27 20.81 22.76 16.48 ...  
## $ wind : num 3.43 2.47 4.03 2.16 4.27 ...  
## $ cloudiness : num 114.2 181.1 90.7 187.5 159 ...

## Datatable

## Datatable

Here is a nice table!

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Country | Happiness\_Rank | accessibility\_to\_cities | elevation | aspect | slope | cropland\_cover | tree\_canopy\_cover | isothermality | rain\_coldestQuart | rain\_driestMonth | rain\_driestQuart | rain\_mean\_annual | rain\_seasonailty | rain\_warmestQuart | rain\_wettestMonth | rain\_wettestQuart | temp\_annual\_range | temp\_coldestQuart | temp\_diurnal\_range | temp\_driestQuart | temp\_max\_warmestMonth | temp\_mean\_annual | temp\_min\_coldestMonth | temp\_seasonality | temp\_warmestQuart | temp\_wettestQuart | wind | cloudiness |
| Afghanistan | U | 317.71575 | 1831.74440 | 201.4298 | 1.5156001 | 9.511846 | 0.3746726 | 35.90442 | 128.718360 | 1.7224832 | 8.300540 | 311.32914 | 91.62876 | 12.689583 | 67.83925 | 175.8160 | 40.26161 | -0.2613516 | 14.724698 | 21.1203287 | 31.95010 | 11.533306 | -8.3115067 | 88.15351 | 22.67524 | 3.947660 | 3.432747 | 114.2290 |
| Albania | H | 73.83086 | 651.81554 | 192.1303 | 1.8900753 | 23.346087 | 12.8046289 | 33.16941 | 392.508789 | 40.0884801 | 138.154620 | 1151.09747 | 38.54663 | 138.334670 | 159.04607 | 435.8747 | 27.08155 | 3.5798958 | 9.114174 | 19.5732059 | 26.27564 | 11.467487 | -0.8059126 | 62.66852 | 19.57742 | 5.267450 | 2.472694 | 181.1311 |
| Algeria | H | 1212.79982 | 556.75832 | 184.9747 | 0.1708615 | 3.690864 | 0.1766562 | 40.29895 | 25.290778 | 0.9349921 | 6.088662 | 79.45607 | 67.08149 | 9.508621 | 13.36623 | 33.2941 | 36.49031 | 13.1515534 | 14.872674 | 26.9210724 | 41.54823 | 22.963946 | 5.0579265 | 75.11026 | 32.45083 | 20.813229 | 4.025770 | 90.6744 |
| Angola | U | 378.20239 | 1061.47899 | 174.2569 | 0.1926286 | 2.794477 | 19.8701092 | 64.33239 | 8.054832 | 0.2601221 | 4.430197 | 1023.37470 | 91.53816 | 318.535618 | 202.19713 | 524.2807 | 21.48916 | 18.7935162 | 13.845249 | 18.8668818 | 31.03791 | 21.611874 | 9.5487517 | 18.53582 | 23.28857 | 22.758982 | 2.164405 | 187.5170 |
| Argentina | VH | 209.21958 | 682.79925 | 145.0314 | 0.6238553 | 21.962504 | 8.8336096 | 49.85147 | 79.087676 | 17.1832114 | 60.486305 | 539.87247 | 48.32124 | 183.138160 | 79.21689 | 211.7121 | 26.77336 | 8.0238906 | 13.461877 | 11.0949601 | 28.21619 | 14.207192 | 1.4428340 | 47.56007 | 20.17117 | 16.477629 | 4.270904 | 159.0065 |
| Armenia | U | 97.29452 | 1850.48297 | 183.5375 | 2.3188956 | 21.338266 | 6.9929146 | 32.42359 | 69.674299 | 20.0054896 | 67.223098 | 501.65877 | 46.50160 | 127.161681 | 86.25816 | 213.7846 | 34.82843 | -4.5577218 | 11.433959 | -0.5852048 | 24.35806 | 6.165300 | -10.4703703 | 81.33372 | 16.19944 | 9.369816 | 1.968253 | 191.7617 |
| Australia | VH | 845.86802 | 278.03171 | 183.0250 | 0.0892681 | 7.938567 | 6.0561555 | 49.69498 | 68.505734 | 11.5888329 | 42.903430 | 467.50809 | 59.33698 | 170.644907 | 89.63448 | 234.7329 | 28.09456 | 14.7095377 | 13.980067 | 18.6189937 | 35.18608 | 21.466618 | 7.0915148 | 49.75149 | 27.37467 | 24.350216 | 4.151111 | 126.8140 |
| Austria | VH | 51.98273 | 950.53000 | 166.5693 | 1.4500095 | 33.003559 | 32.2642051 | 31.64129 | 189.495629 | 55.6065681 | 182.035657 | 1049.22861 | 30.45925 | 378.582134 | 132.78880 | 379.5294 | 28.32867 | -3.2035951 | 9.129945 | -1.7868646 | 20.71571 | 5.744182 | -7.6129561 | 68.33476 | 14.33271 | 14.330183 | 2.464692 | 231.8577 |
| Azerbaijan | H | 86.63017 | 640.06762 | 127.1189 | 1.4408533 | 36.145032 | 7.2671922 | 28.46190 | 85.279313 | 16.3500030 | 61.220735 | 453.30041 | 39.61667 | 98.757680 | 68.16093 | 174.0463 | 33.16762 | 1.5301839 | 9.605681 | 13.5284105 | 29.91986 | 11.989628 | -3.2477612 | 81.08114 | 22.22982 | 13.549606 | 2.662135 | 191.1590 |
| Bangladesh | H | 33.64179 | 29.93988 | 168.6095 | 0.0325392 | 68.131360 | 15.6179158 | 43.39718 | 34.113354 | 4.7619174 | 33.837177 | 2194.09854 | 91.06425 | 899.022021 | 466.05129 | 1273.2028 | 21.97629 | 19.8563773 | 9.626260 | 19.9846405 | 34.05148 | 25.521950 | 12.0751965 | 36.12041 | 28.64029 | 28.320933 | 2.450944 | 193.4797 |
| Belarus | H | 79.96755 | 161.15274 | 173.2924 | 0.0014257 | 51.466341 | 24.8728945 | 24.66370 | 111.503657 | 29.0028583 | 100.158781 | 624.14771 | 32.13826 | 231.674538 | 84.27929 | 231.6745 | 32.80025 | -5.3027945 | 8.228643 | -4.2013718 | 23.07259 | 6.244620 | -9.7276601 | 87.78211 | 17.10305 | 17.103049 | 3.264418 | 232.5725 |
| Belgium | VH | 14.38773 | 143.45749 | 218.8504 | 0.0953660 | 60.776216 | 15.0831524 | 33.25148 | 213.999746 | 56.2917139 | 183.645092 | 850.20472 | 11.33183 | 223.204347 | 83.65095 | 239.4220 | 22.43083 | 2.4340423 | 7.576102 | 5.9161187 | 21.70532 | 9.524934 | -0.7255174 | 55.15860 | 16.43188 | 7.706426 | 3.876680 | 237.9154 |
| Benin | U | 115.20319 | 267.04215 | 178.5135 | 0.0380440 | 25.279258 | 7.5916774 | 66.17602 | 477.592793 | 2.1062087 | 14.011089 | 1061.15181 | 91.49097 | 146.873797 | 228.95156 | 605.8449 | 18.64317 | 25.2216103 | 12.303683 | 26.8923644 | 37.07766 | 27.186348 | 18.4344933 | 16.81105 | 29.56815 | 25.534639 | 2.105127 | 212.7379 |
| Bhutan | H | 501.28316 | 2860.92535 | 181.5962 | 3.6983940 | 2.422378 | 49.8992422 | 44.83827 | 23.091998 | 3.8840301 | 21.313840 | 1509.81303 | 98.45824 | 899.312165 | 354.21080 | 927.5801 | 25.92725 | 2.9688695 | 11.756950 | 3.4260322 | 20.37446 | 9.953483 | -5.5527825 | 50.85974 | 15.88991 | 15.840825 | 2.439829 | 226.0796 |
| Bolivia | H | 445.12942 | 1290.02061 | 151.5460 | 0.6737767 | 4.057431 | 37.8147977 | 66.80037 | 100.328928 | 21.2518449 | 80.731905 | 1148.54296 | 75.84720 | 399.788309 | 198.38700 | 537.5875 | 19.96708 | 17.4872796 | 13.362064 | 18.0140993 | 29.44365 | 20.426654 | 9.4765654 | 19.79313 | 22.35332 | 22.148082 | 2.340412 | 196.9934 |
| Bosnia and Herzegovina | H | 73.43443 | 710.18800 | 160.8869 | 1.3528183 | 37.555606 | 34.4474441 | 31.82940 | 259.087867 | 64.4925311 | 204.197032 | 1035.95517 | 18.59864 | 250.690245 | 118.86808 | 324.2753 | 28.60093 | 0.0793670 | 9.273848 | 6.2444438 | 24.59435 | 9.072414 | -4.0065813 | 68.54096 | 17.70262 | 9.658404 | 2.315268 | 209.5248 |
| Botswana | U | 386.90666 | 1037.20212 | 162.7331 | 0.0062352 | 1.077684 | 1.6641639 | 55.61240 | 4.191614 | 0.6072688 | 3.661472 | 395.66263 | 91.84449 | 178.258424 | 88.15374 | 233.4610 | 29.01835 | 14.6822526 | 16.269263 | 14.6950611 | 33.27721 | 20.834293 | 4.2588571 | 42.94332 | 25.22013 | 24.748999 | 3.201352 | 112.3621 |
| Brazil | VH | 736.72980 | 326.64308 | 176.1375 | 0.1606525 | 17.282336 | 39.1752366 | 72.73447 | 366.626672 | 39.3132796 | 141.171795 | 1752.79241 | 60.92779 | 370.370421 | 277.12119 | 761.9858 | 15.49870 | 22.9490190 | 11.108500 | 23.6800075 | 32.01369 | 24.447749 | 16.5149899 | 10.75879 | 25.62179 | 24.754861 | 1.963017 | 218.4851 |
| Bulgaria | U | 89.12241 | 467.44587 | 160.3145 | 0.9749420 | 52.312497 | 20.3315042 | 30.87807 | 144.710998 | 35.2115842 | 117.985350 | 609.95772 | 22.86174 | 160.025342 | 73.87040 | 197.5502 | 30.76479 | 0.3883025 | 9.650659 | 9.7316973 | 26.43864 | 10.179378 | -4.3261514 | 74.67946 | 19.56625 | 13.330115 | 2.724888 | 191.3499 |
| Burkina Faso | U | 115.86369 | 302.30818 | 169.5040 | 0.0053619 | 14.993030 | 2.5775374 | 60.67141 | 174.003340 | 0.4366631 | 3.489308 | 760.54110 | 115.27002 | 107.650989 | 213.85117 | 517.2103 | 22.10615 | 25.6022003 | 13.474171 | 26.0392960 | 38.81851 | 28.070577 | 16.7123581 | 22.10523 | 31.17071 | 26.761712 | 2.838656 | 179.2996 |

# Principal Component Analysis

## Description

Principal component analysis (PCA), part of descriptive analytics, is used to analyze one table of quantitative data, specifically useful for *high dimensional data* and comparitively lesser data rows. PCA mixes the input variables to give new variables, called principal components. The first principal component is the line of best fit. It is the line that maximizes the inertia (similar to variance) of the cloud of data points. Subsequent components are defined as orthogonal to previous components, and maximize the remaining inertia.

PCA gives one map for the rows (called factor scores), and one map for the columns (called loadings). These 2 maps are related, because they both are described by the same components. However, these 2 maps project different kinds of information onto the components, and so they are *interpreted differently*. Factor scores are the coordinates of the row observations and Loadings describe the column variables. Both can be interpreted through their distance from origin. However, Factor scores are also interpreted by the distances between them and Loadings interpreted by the angle between them.

The distance from the origin is important in both maps, because squared distance from the mean is inertia (variance, information; see sum of squares as in ANOVA/regression). Because of the Pythagorean Theorem, the total information contributed by a data point (its squared distance to the origin) is also equal to the sum of its squared factor scores.

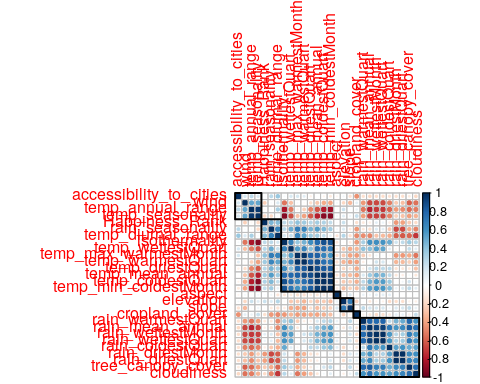
With both Factor and Loadings maps combined we can interpret which grouping criteria of rows of data is most impacted by which columns. This can interpreted visually by observing which a factors and loadings on a particular component and the distance on this component.

PCA also helps in *dimensionality reduction*. Using SVD, we get eigen values arranged in descending order in the diagonal matrix. We can simply ignore the lower eigen values to reduce dimensions. We can also take help of SCREE plot to visually analyze importance of eigen values.

There are multiple variables representing rain and Temp. Hence, for analysis purposes, lets choose annual mean for Rain and Temp.

## Correlation Plot

Visually analyze multicollinearity in the system.

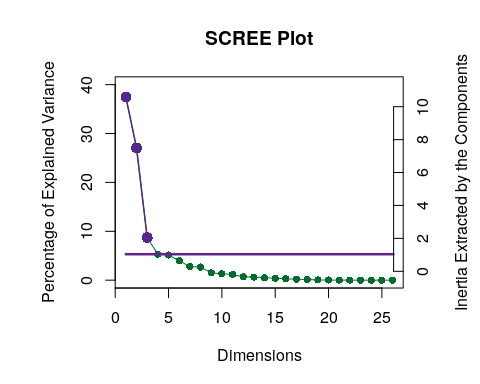


Now we have Factor scores and Loadings. \* Factor Scores are the new Data points w.r.t. new Components achieved with help of SVD. \* Loadings represent correlation between variables w.r.t the choosen Components. Can be interpreted in 3 ways + As slices of inertia of the contribution data table w.r.t. the choosen Components + As correlation between columns (features) of Original Data and Factor scores of each Components (latent features). + As coefficients of optimal linear combination i.e. Right Sigular Vectors (Q matrix of SVD)

## Scree Plot

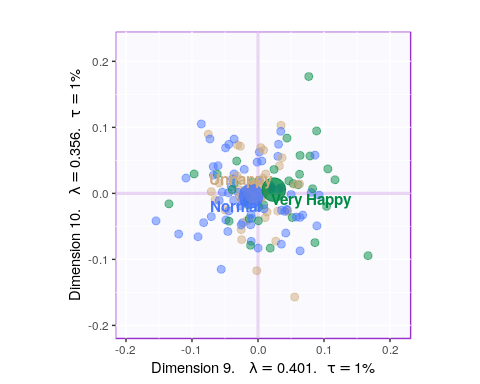
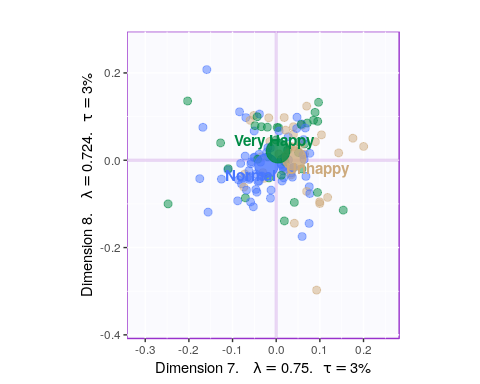
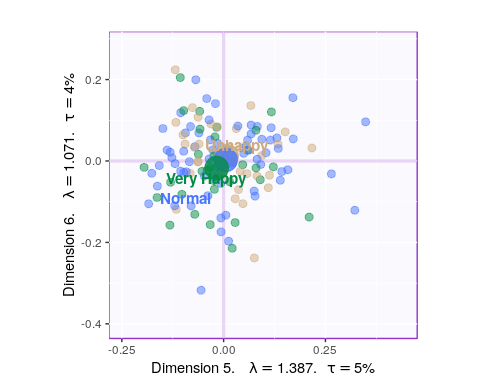
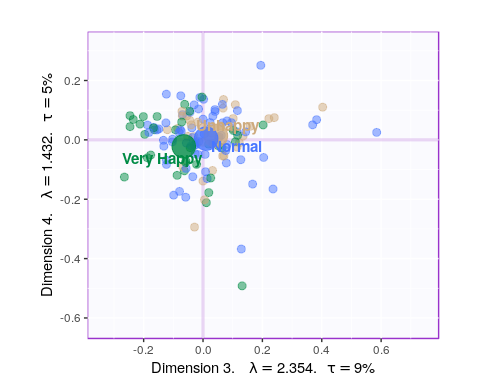
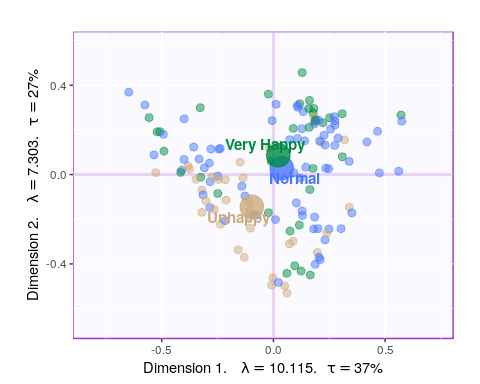
Gives amount of information explained by corresponding component. Gives an intuition to decide which components best represent data in order to answer the research question.

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



## Factor Scores

Lets visualize happiness categories for components 1-10, to make a decision (visually) on the most important components.



Since, it’s not very straightforward to decide which components may be best suited for the research question at hand, let’s represent, in a tabular format, which component helps to differentiate between which design variable values (Unhappy, Normal, Very Happy)

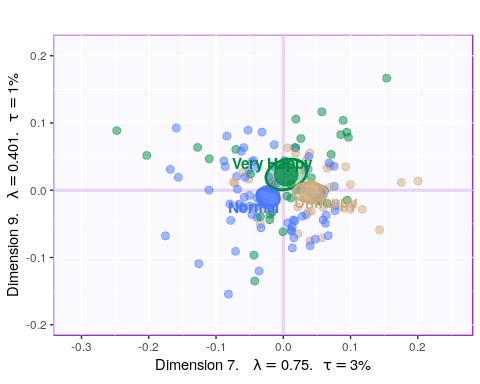
P.S. here -1 represents -ve quadrant of the component and +1 represent +ve quadrant. 0 represents that component was not decisive enough to clearly seperate happiness levels.

Identify Components best describing happiness levels

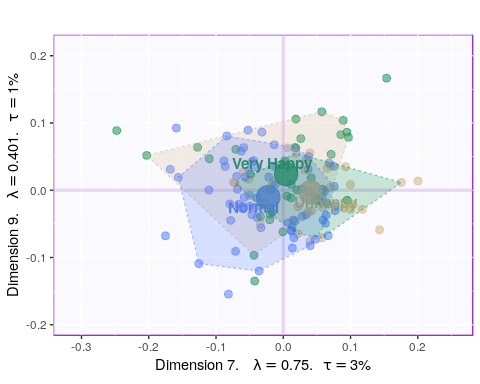
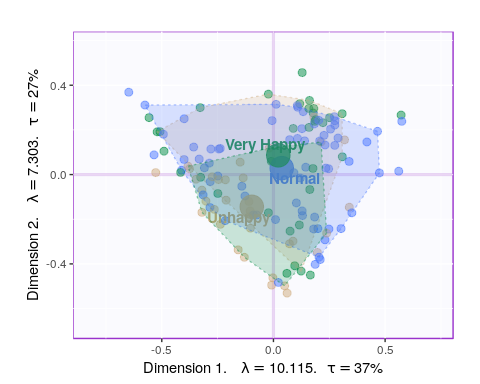
|  |  |  |  |
| --- | --- | --- | --- |
|  | happy | happier | happiest |
| Component 1 | -1 | 1 | 0 |
| Component 2 | -1 | 0 | 1 |
| Component 3 | 1 | 0 | -1 |
| Component 4 | 0 | 0 | 0 |
| Component 5 | 0 | 0 | 0 |
| Component 6 | 0 | 0 | 0 |
| Component 7 | 1 | -1 | 0 |
| Component 8 | 0 | 0 | 0 |
| Component 9 | 0 | -1 | 1 |
| Component 10 | 0 | 0 | 0 |

Looking at the table, it seems component 1, 2, 7, 9 may be able to best represent all 3 happiness levels. Although, SCREE Plot suggests that and components might be useful, from our above analysis we know otherwise. Also, SCREE plot suggests that component and onwards might not be useful which is contradicting our findings above. Hence, let’s plot components 1 vs 2 and 7 vs 9. Similarily, we will also plot Loading plots for these componenets.

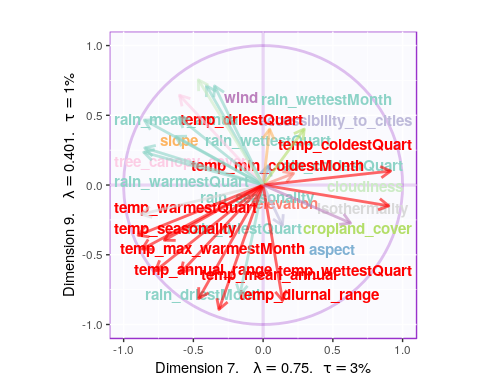
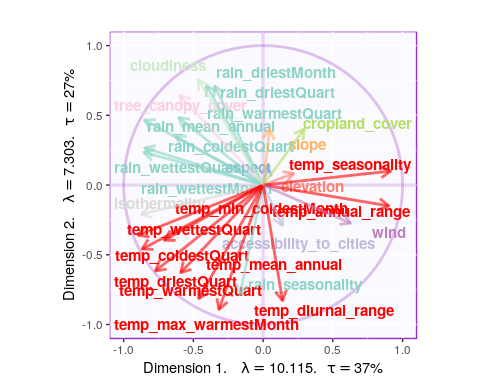
* With Confidence Interval



* With Tolerance Interval



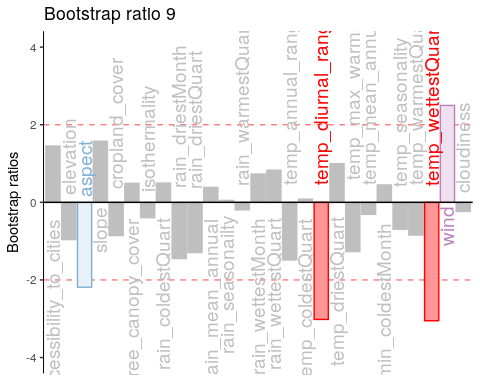
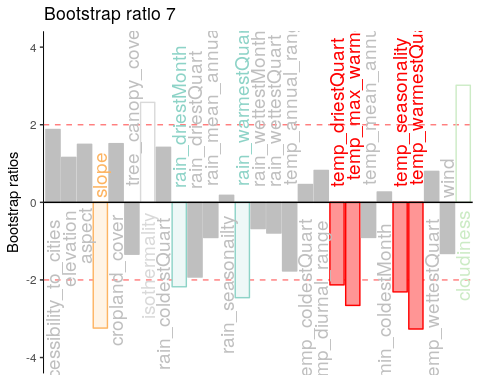
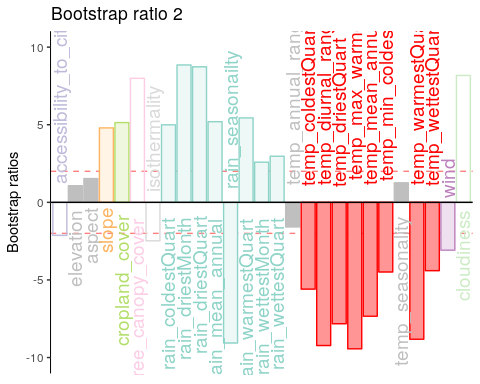
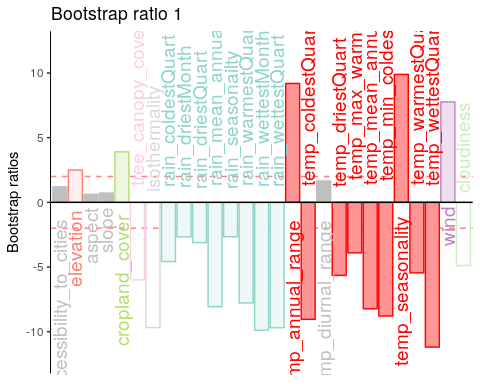
## Loadings



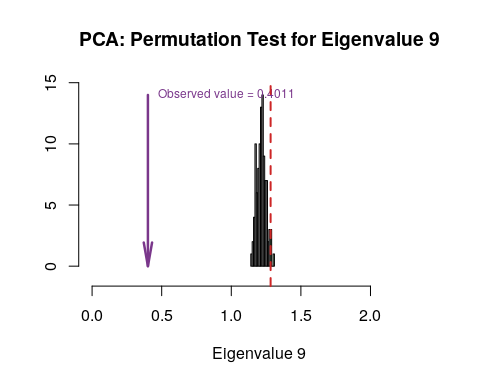
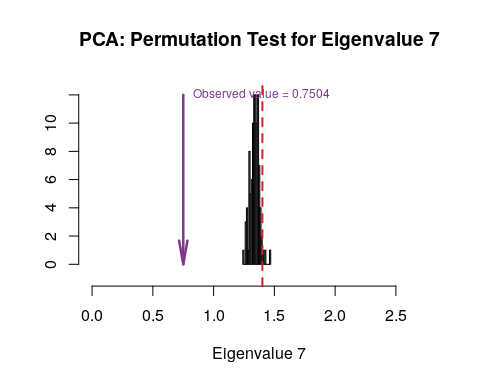
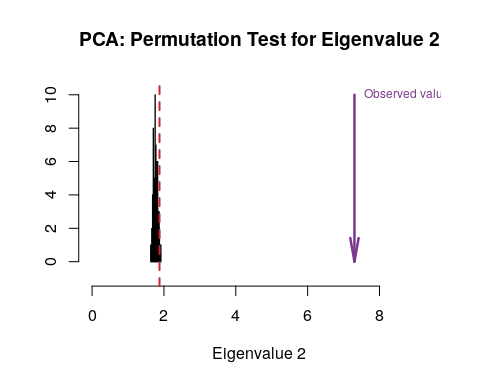
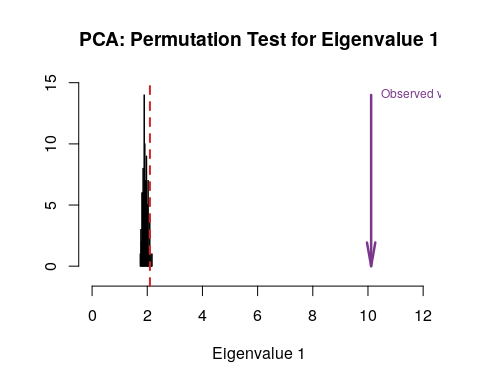
## Most Contributing Variables

Let’s plot variable contributions against each chosen components i.e. 1, 7, 9.

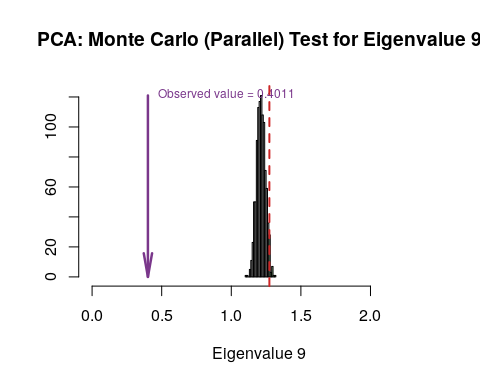
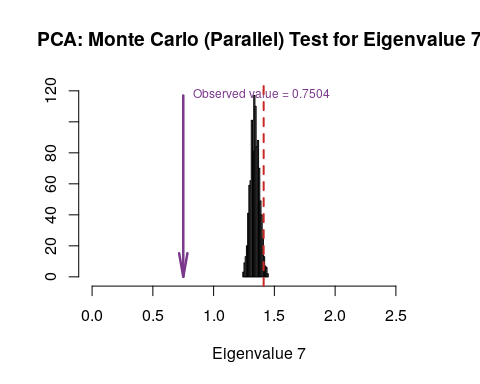
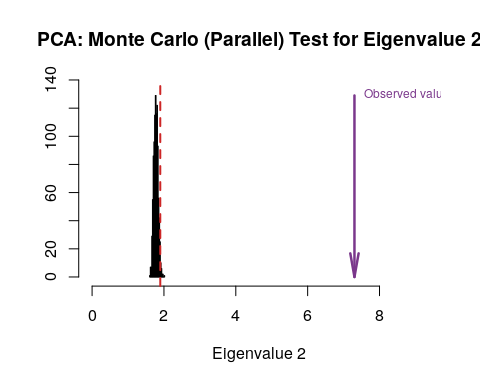
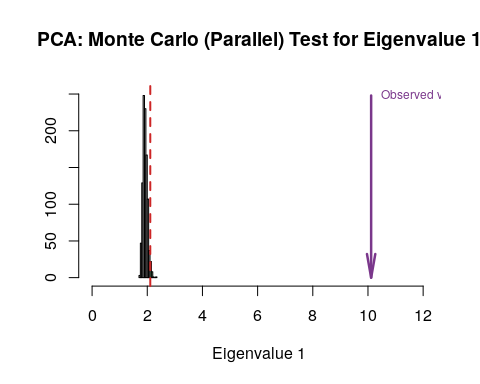
* With Bootstrap Ratio



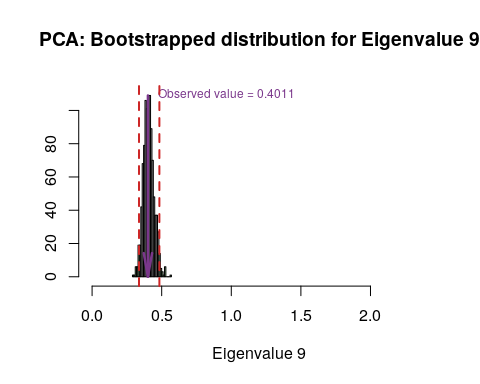
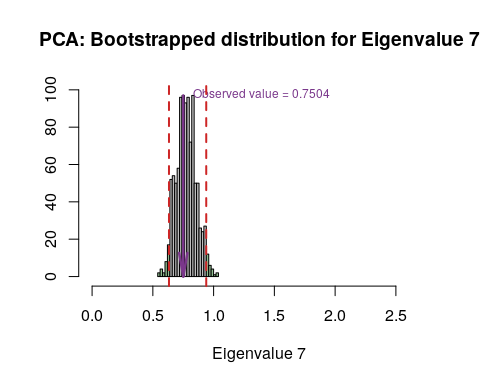
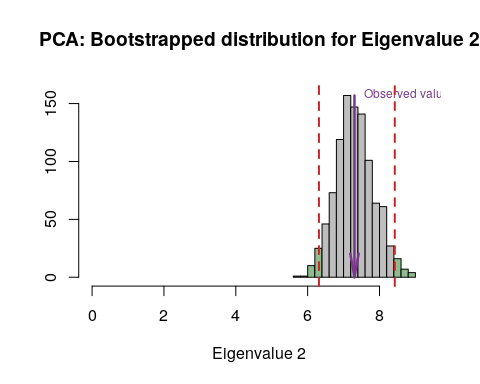
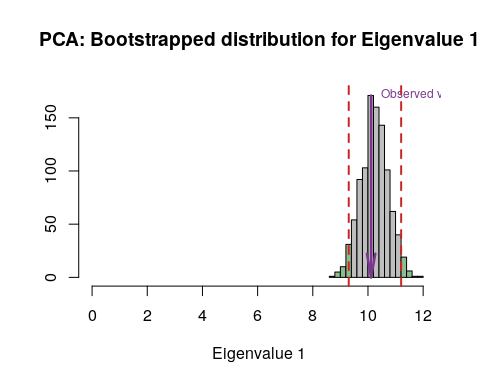
## Permutation Test



## Parallet Test



## Bootstrap Test



## Conclusion

* Component 1:
  + Rows: Normal & Happy
  + Columns: Cloudiness & Rain vs Cropland, Aspect, Elevation
  + Interpret: People in countries with more Cloudiness, Trees and Rain tends to be happier.
* Component 7:
  + Rows: Happy & Unhappy
  + Columns: Temp and Rain vs Accessibility and Cropland
  + Interpret: Rain and Temp seems to be main reason for unhappiness and Cropland is important for Happiness.
* Component 9:
  + Rows: Happy & Very Happy
  + Columns: Temp vs Rain
  + Interpret: Rain and Temp seems to be main reason for Happiness. *This contradicts with Component 7 and 1*.

# Multiple Component Analysis

## Description

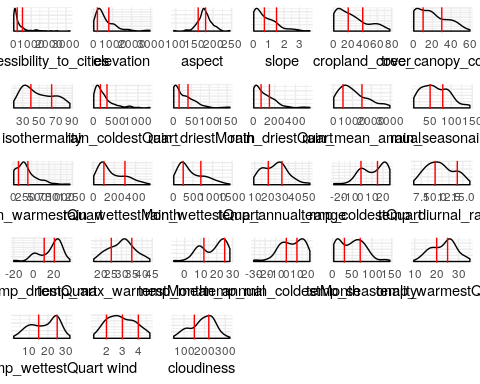
Multiple correspondence analysis (MCA) is an extension of correspondence analysis(CA) which allows one to analyze the pattern of relationships of several categorical dependent variables. As such, it can also be seen as a generalization of principal component analysis when the variables to be analyzed are categorical instead of quantitative. Because MCA has been (re)discovered many times, equivalent methods are known under several different names such as optimal scaling, optimal or appropriate scoring, dual scaling, homogeneity analysis,scalogram analysis, and quantiﬁcation method.

**Interpreting MCA** Multiple correspondence analysis locates all the categories in a Euclidean space.

* The first two dimensions of this space are plotted to examine the associations among the categories.
* The top-right quadrant of the plot shows the categories.
* The bottom-left quadrant shows the association.
* This interpretation is based on points found in approximately the same direction from the origin and in approximately the same region of the space. Distances between points do not have a straightforward interpretation.

## Density Plot

Let’s observe the distribution of each variables to get an intuition of how we can bin these variables. It’s important to have nearly equal number of observations in the each bin and to try to cut the variables in a way to so that each new binned distribution is nearly Gaussian. We can also verify that our binning is appropiate by calculating Spearman Correlation for each of original variable and binned variable, the correlation coefficient should be close to 1.

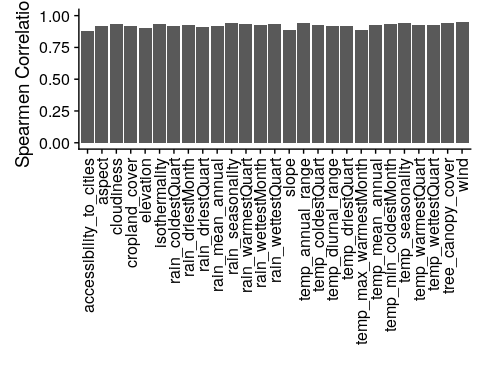


## Binning

Structure of Data after binning based on above observation.

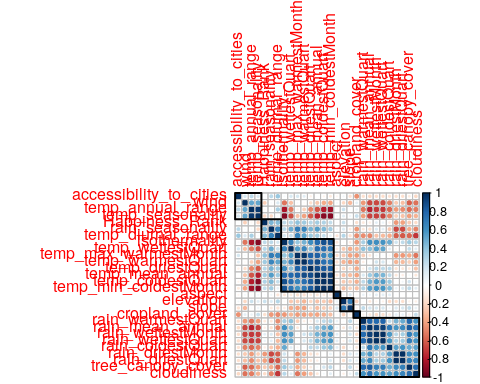
## 'data.frame': 137 obs. of 27 variables:  
## $ accessibility\_to\_cities: Factor w/ 3 levels "1","2","3": 2 1 3 2 2 1 3 1 1 1 ...  
## $ elevation : Factor w/ 3 levels "1","2","3": 3 2 2 3 2 3 2 3 2 1 ...  
## $ aspect : Factor w/ 3 levels "1","2","3": 3 3 3 2 1 3 3 2 1 2 ...  
## $ slope : Factor w/ 3 levels "1","2","3": 3 3 1 1 1 3 1 2 2 1 ...  
## $ cropland\_cover : Factor w/ 3 levels "1","2","3": 1 2 1 1 2 2 1 2 2 3 ...  
## $ tree\_canopy\_cover : Factor w/ 3 levels "1","2","3": 1 2 1 2 1 1 1 3 1 2 ...  
## $ isothermality : Factor w/ 3 levels "1","2","3": 1 1 2 2 2 1 2 1 1 2 ...  
## $ rain\_coldestQuart : Factor w/ 3 levels "1","2","3": 1 3 1 1 1 1 1 2 1 1 ...  
## $ rain\_driestMonth : Factor w/ 3 levels "1","2","3": 1 3 1 1 2 2 1 3 2 1 ...  
## $ rain\_driestQuart : Factor w/ 3 levels "1","2","3": 1 2 1 1 1 1 1 3 1 1 ...  
## $ rain\_mean\_annual : Factor w/ 3 levels "1","2","3": 1 2 1 2 2 2 1 2 1 3 ...  
## $ rain\_seasonailty : Factor w/ 3 levels "1","2","3": 3 1 2 3 1 1 2 1 1 3 ...  
## $ rain\_warmestQuart : Factor w/ 3 levels "1","2","3": 1 2 1 3 2 2 2 3 1 3 ...  
## $ rain\_wettestMonth : Factor w/ 3 levels "1","2","3": 1 2 1 2 1 1 1 2 1 3 ...  
## $ rain\_wettestQuart : Factor w/ 3 levels "1","2","3": 1 2 1 2 1 1 1 2 1 3 ...  
## $ temp\_annual\_range : Factor w/ 3 levels "1","2","3": 3 2 3 2 2 3 2 2 3 2 ...  
## $ temp\_coldestQuart : Factor w/ 3 levels "1","2","3": 1 2 2 3 2 1 2 1 2 3 ...  
## $ temp\_diurnal\_range : Factor w/ 3 levels "1","2","3": 3 1 3 2 2 2 2 1 1 1 ...  
## $ temp\_driestQuart : Factor w/ 3 levels "1","2","3": 3 2 3 2 2 1 2 1 2 2 ...  
## $ temp\_max\_warmestMonth : Factor w/ 3 levels "1","2","3": 2 2 3 2 2 1 3 1 2 2 ...  
## $ temp\_mean\_annual : Factor w/ 3 levels "1","2","3": 1 1 2 2 2 1 2 1 1 3 ...  
## $ temp\_min\_coldestMonth : Factor w/ 3 levels "1","2","3": 1 1 2 2 2 1 2 1 1 3 ...  
## $ temp\_seasonality : Factor w/ 3 levels "1","2","3": 3 2 3 1 2 3 2 2 3 2 ...  
## $ temp\_warmestQuart : Factor w/ 3 levels "1","2","3": 2 1 3 2 2 1 3 1 2 3 ...  
## $ temp\_wettestQuart : Factor w/ 3 levels "1","2","3": 1 1 2 2 2 1 2 1 1 3 ...  
## $ wind : Factor w/ 4 levels "1","2","3","4": 3 2 4 2 4 1 4 2 2 2 ...  
## $ cloudiness : Factor w/ 3 levels "1","2","3": 1 2 1 2 2 2 1 3 2 2 ...

## Spearman Correlation

Let’s observe correlation between original data and binned data to make sure that neither the correlation ceofficient is too low or perfect. 

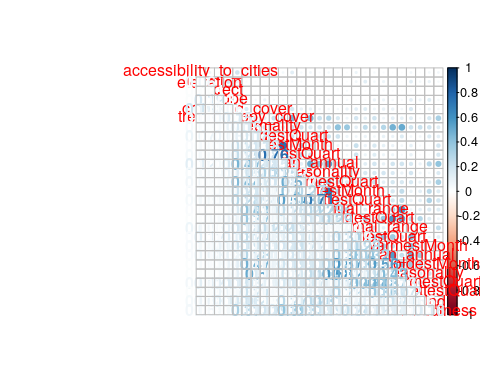
## Correlation Plot

Visually analyze multicollinearity in the system of the original data



## Heatmap

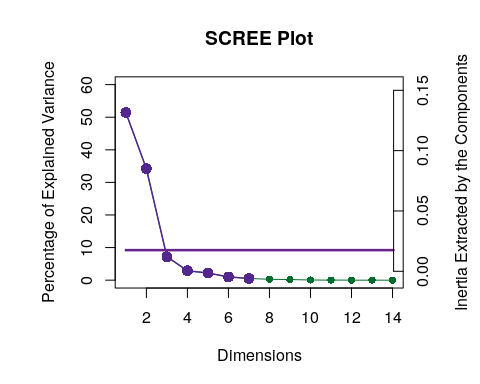
For binned data.



## Scree Plot

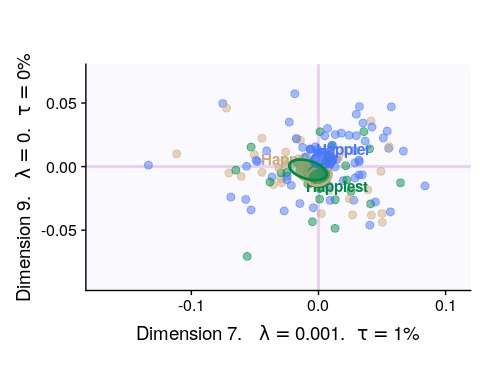
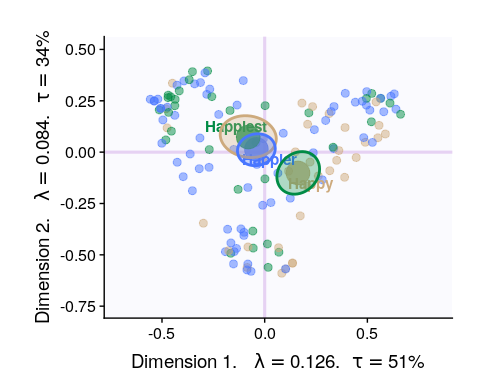
Gives amount of information explained by corresponding component. Gives an intuition to decide which components best represent data in order to answer the research question.

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

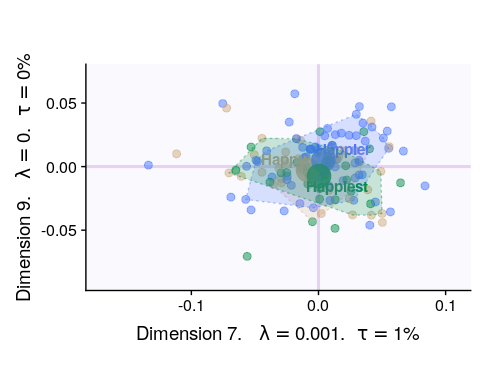
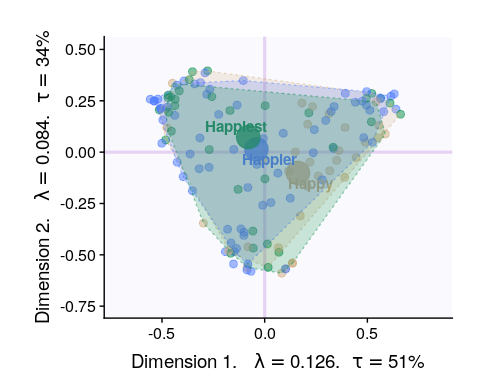


## Factor Scores

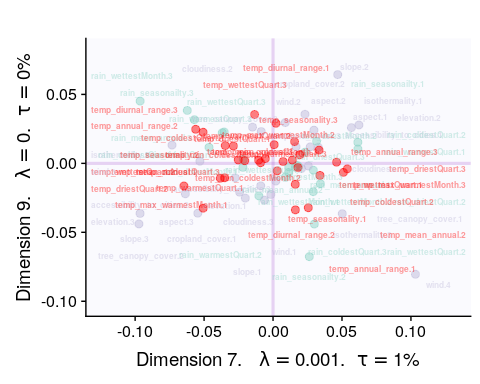
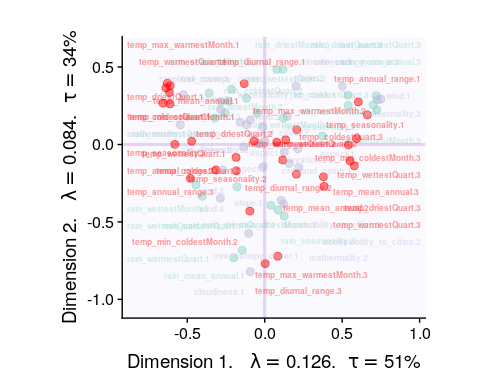
Lets visualize happiness categories for components 1-10, to make a decision (visually) on the most important components.

**With Confidence Interval** 

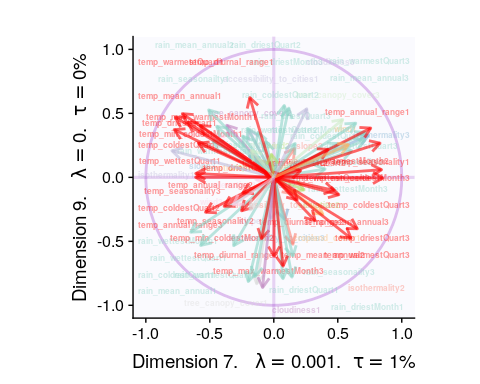
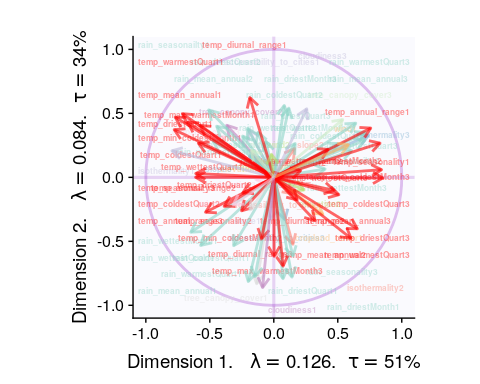
**With Tolerance Interval**



## Loadings



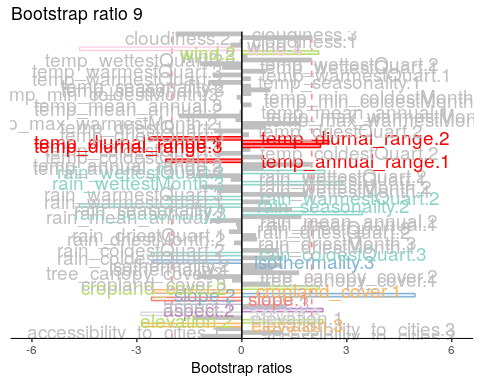
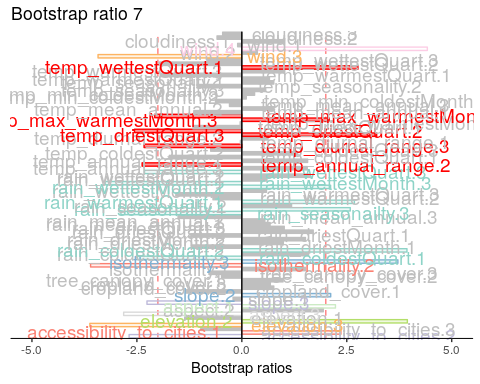
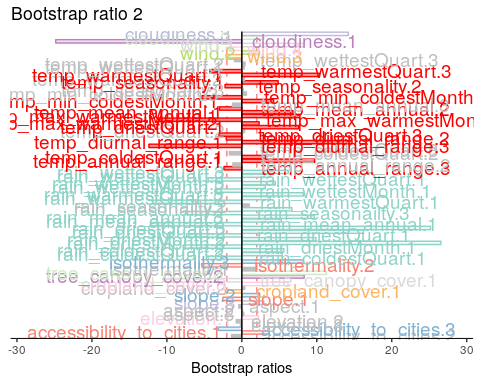
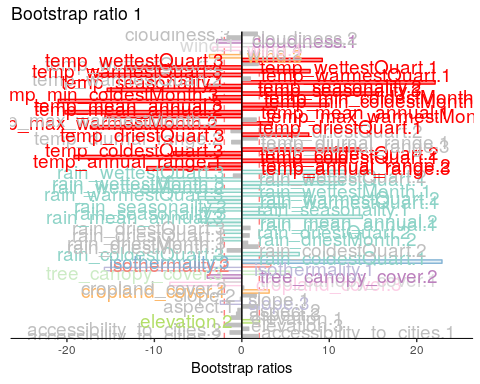
## Loadings (correlation plot)



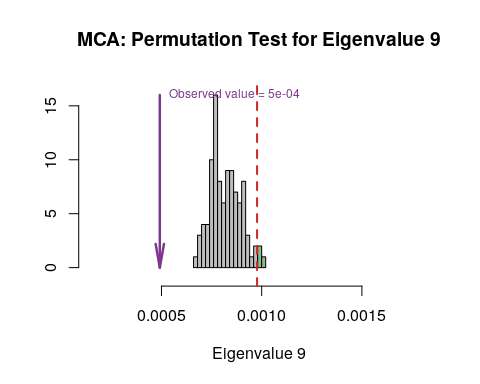
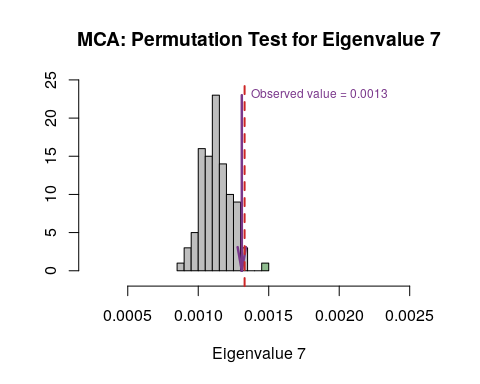
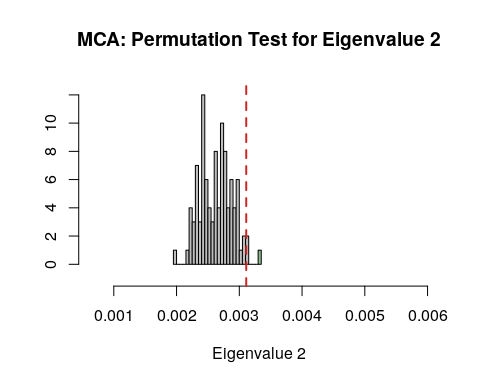
## Most Contributing Variables (Inference)

Let’s plot variable contributions against each chosen components i.e. 1, 2, 7, 9.

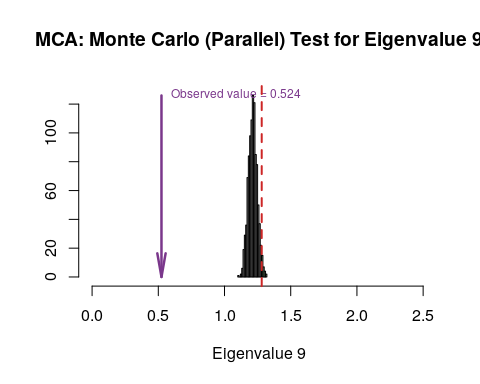
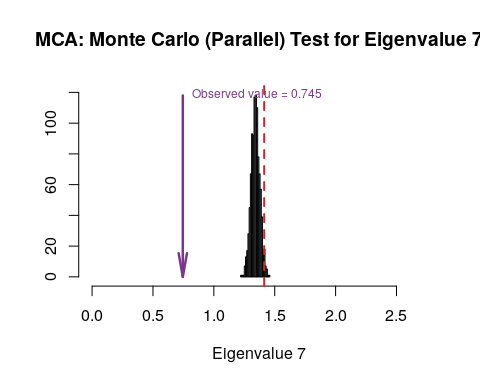
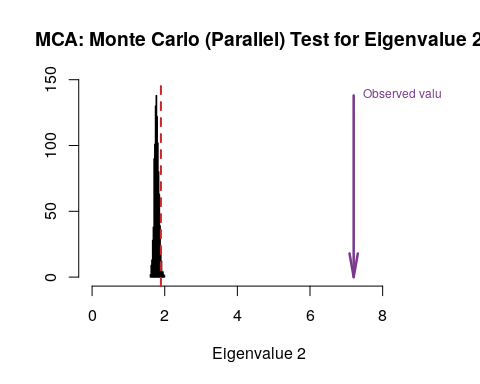
* With Bootstrap Ratio



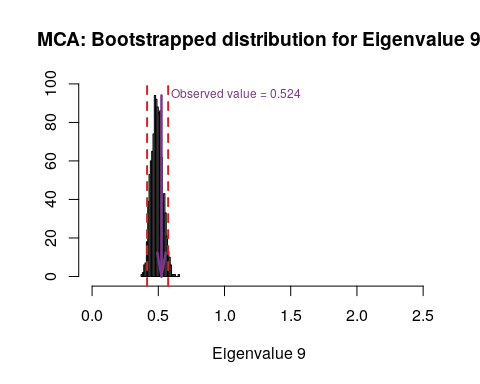
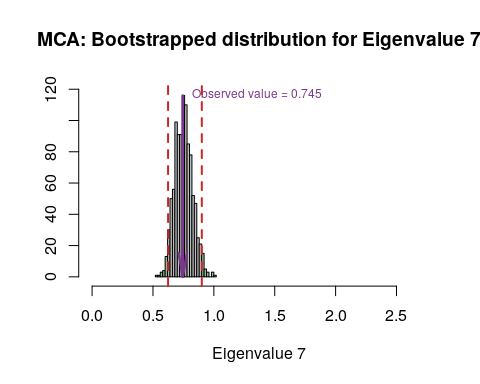
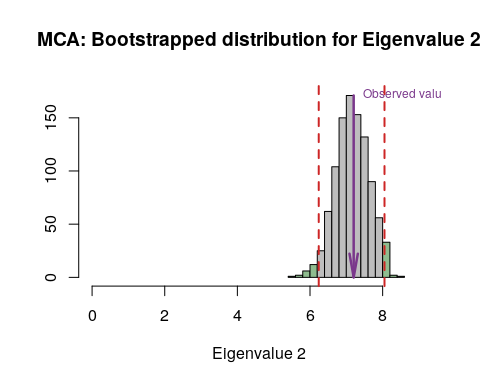
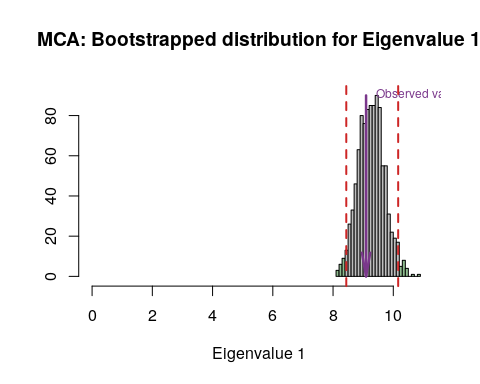
## Permutation Test



## Parallet Test



## Bootstrap Test



## Conclusion

* Component 1:
  + Rows: Normal & Happy
  + Columns: Cloudiness & Rain vs Cropland, Aspect, Elevation
  + Interpret: People in countries with more Cloudiness, Trees and Rain tends to be happier.
* Component 7:
  + Rows: Happy & Unhappy
  + Columns: Temp and Rain vs Accessibility and Cropland
  + Interpret: Rain and Temp seems to be main reason for unhappiness and Cropland is important for Happiness.
* Component 9:
  + Rows: Happy & Very Happy
  + Columns: Temp vs Rain
  + Interpret: Rain and Temp seems to be main reason for Happiness. *This contradicts with Component 7 and 1*.

# Partial Least Squares - Correlation

## Method: PLS-C

PLS is used to find the fundamental relations between two matrices (X and Y), i.e. a latent variable approach to modeling the covariance structures in these two spaces. A PLS model will try to find the multidimensional direction in the X space that explains the maximum multidimensional variance direction in the Y space. PLS regression is particularly suited when the matrix of predictors has more variables than observations, and when there is multicollinearity among X values. PLS bears some relation to principal components regression; instead of finding hyperplanes of maximum variance between the response and independent variables, it finds a linear regression model by projecting the predicted variables and the observable variables to a new space. Because both the X and Y data are projected to new spaces, the PLS family of methods are known as bilinear factor models.

* Research Question

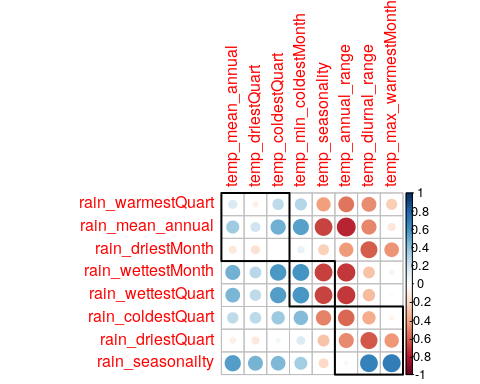
Which variables in Rain and Temperature contribute most towards happiness

## Analysis

## Correlation Plot

Visually analyze multicollinearity between all varaibles in Rain and Temperature tables.

## Warning in as.dist.default(1 - corr): non-square matrix



## PLS-C

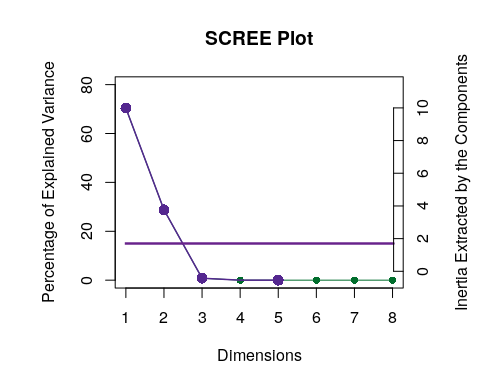
## ------------------------------------------------------------------------------  
## Results of Permutation Test for PLSC of X'\*Y = R   
## for Omnibus Inertia and Eigenvalues   
## ------------------------------------------------------------------------------  
## $ fixedInertia the Inertia of Matrix X  
## $ fixedEigenvalues an L\*1 vector of the eigenvalues of X  
## $ pOmnibus the probablity associated to the Inertia  
## $ pEigenvalues an L\* 1 matrix of p for the eigenvalues of X  
## $ permInertia vector of the permuted Inertia of X  
## $ permEigenvalues matrix of the permuted eigenvalues of X  
## ------------------------------------------------------------------------------

Now we have Latent Variables and Saliences. \* Latent Variables are the new Data points w.r.t. correlation between both the tables. Latent Variables exists for each table. \* Saliences represent correlation between variables of each table.

## 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.



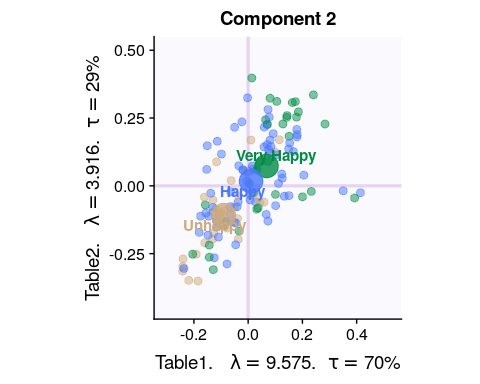
## Latent Variables

Lets visualize happiness categories for Components 1 of each table

### Component 1 for both Tables: Rain and Temperature

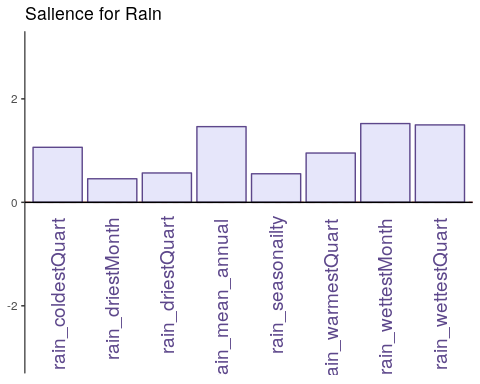


### Component 2 for both Tables: Rain and Temperature

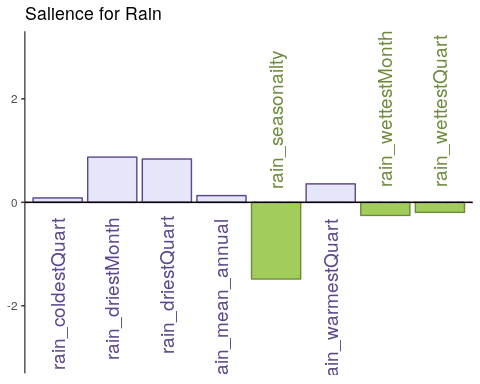


## Salience for Rain

### Components 1



### Component 2

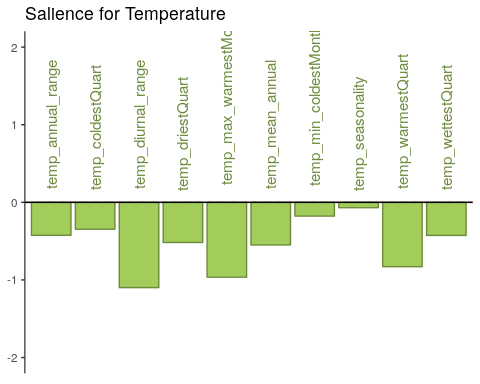


## Salience for Temperature

### Component 1



### Component 2



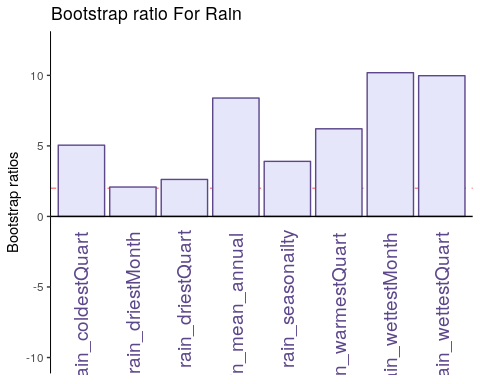
## Most Contributing Variables - PLS-C (with Inference)

## ------------------------------------------------------------------------------  
## Bootstraped Factor Scores (BFS) and Bootstrap Ratios (BR)   
## for the I and J-sets of a PLSC (obtained from multinomial resampling of X & Y)   
## ------------------------------------------------------------------------------  
## $ bootstrapBrick.i an I\*L\*nIter Brick of BFSs for the I-Set  
## $ bootRatios.i an I\*L matrix of BRs for the I-Set  
## $ bootRatiosSignificant.i an I\*L logical matrix for significance of the I-Set  
## $ bootstrapBrick.j a J\*L\*nIter Brick of BFSs for the J-Set  
## $ bootRatios.j a J\*L matrix of BRs for the J-Set  
## $ bootRatiosSignificant.j a J\*L logical matrix for significance of the J-Set  
## ------------------------------------------------------------------------------

### Bootstrap Test

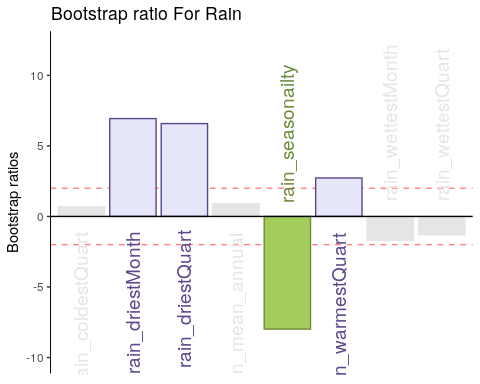
* Rain - Component 1

BR = resBoot4PLSC$bootRatios.i  
  
PrettyBarPlot2(BR[,1],  
 threshold = 2,  
 font.size = 5,  
 main = 'Bootstrap ratio For Rain ',  
 ylab = 'Bootstrap ratios',  
 horizontal = TRUE,  
 ylim = c(-10,12)  
 )



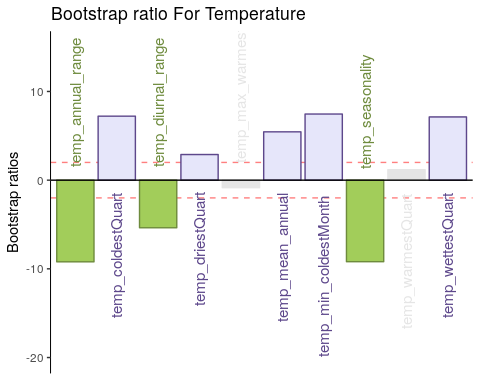
* Rain - Component 2

BR = resBoot4PLSC$bootRatios.i  
  
PrettyBarPlot2(BR[,2],  
 threshold = 2,  
 font.size = 5,  
 #color4bar = gplots::col2hex(col4J), # we need hex code  
 main = 'Bootstrap ratio For Rain ',  
 ylab = 'Bootstrap ratios',  
 horizontal = TRUE,  
 ylim = c(-10,12)  
 )



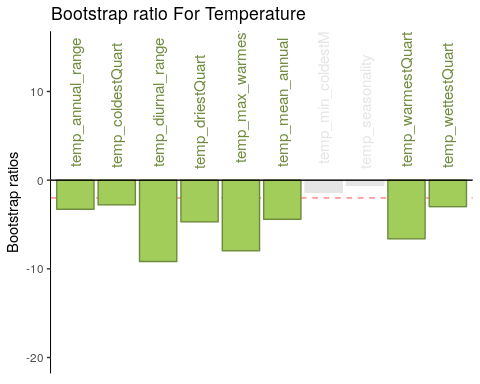
* Temperature - Component 1

BR = resBoot4PLSC$bootRatios.j  
  
PrettyBarPlot2(BR[,1],  
 threshold = 2,  
 font.size = 4,  
 #color4bar = gplots::col2hex(col4J), # we need hex code  
 main = 'Bootstrap ratio For Temperature ',  
 ylab = 'Bootstrap ratios',  
 horizontal = TRUE,  
 ylim = c(-20,15)  
 )



* Temperature - Component 2

BR = resBoot4PLSC$bootRatios.j  
  
PrettyBarPlot2(BR[,2],  
 threshold = 2,  
 font.size = 4,  
 #color4bar = gplots::col2hex(col4J), # we need hex code  
 main = 'Bootstrap ratio For Temperature ',  
 ylab = 'Bootstrap ratios',  
 horizontal = TRUE,  
 ylim = c(-20,15)  
 )



## Conclusion

Here Component 2 seems to best seperate Happiness levels. Let’s compare Component 2 for both tables.

* Table 1 & 2 Component 2
  + Latent Variables: Very Happy vs Unhappy (for Rain and Temperature both)
  + Salience:
    - Rain: It seems dryness and wetness at a montly scale have more effect than coldness or yearly patterns.
    - Temperature: All temperature variations at a monthly and yearly scale seems to impact happiness.

# Barycentric Discriminant Analysis

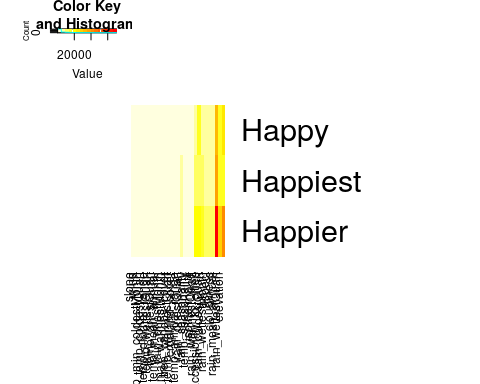
## Method: BADA

## Heatmap

Visually analyze multicollinearity in the system.

happiness\_dummies = as.data.frame(dummy(country\_env\_df$Happiness\_Rank))  
colnames(happiness\_dummies) <- c('Happiest', 'Happier', 'Happy' )  
  
#heatmap(t(happiness\_dummies) %\*% as.matrix(country\_env\_df\_for\_pca))

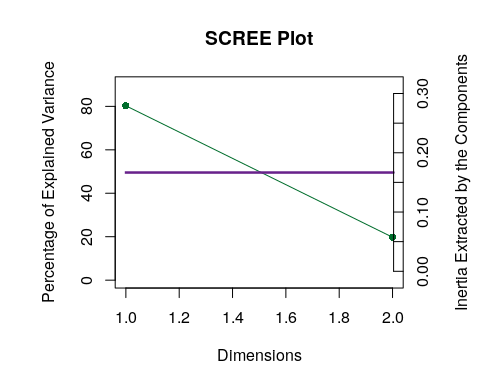
heatmap.2(t(happiness\_dummies) %\*% as.matrix(country\_env\_df\_for\_pca), col = rev(heat.colors(16)), dendrogram = 'none', trace = 'none', margins = c(8, 16)) #tracecol=NA)



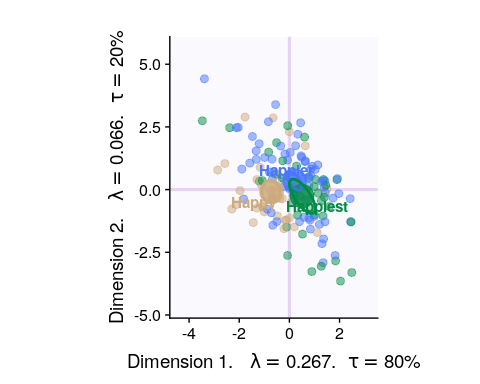
## Scree Plot

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.

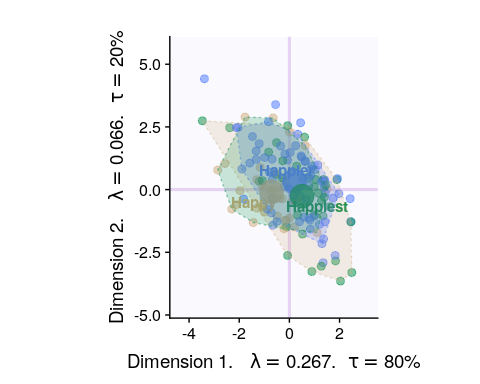
P.S. The most contribution component may not always be most useful for a given research question.



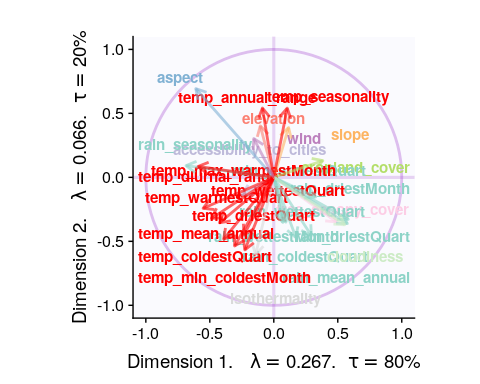
## Factor Scores



* With Tolerance Interval

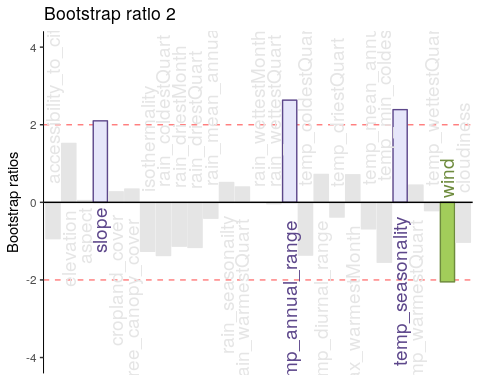
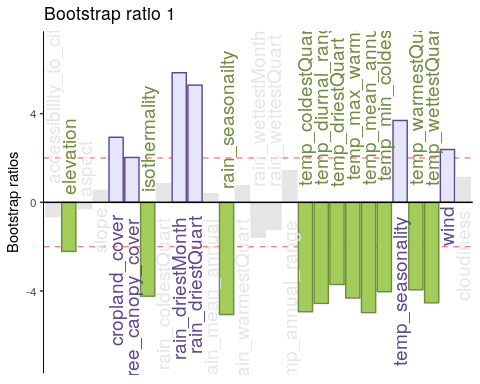


## Loadings

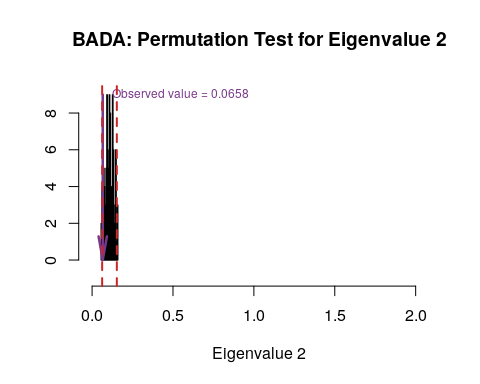
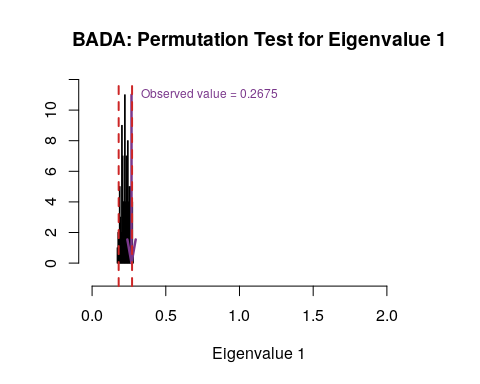


## Most Contributing Variables

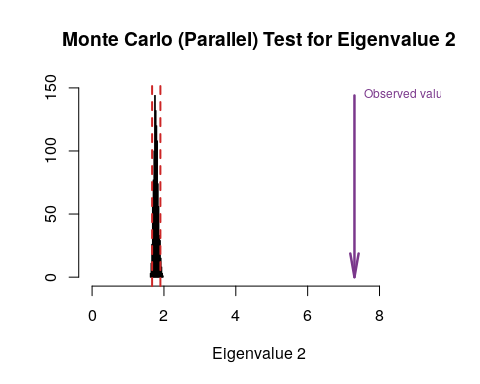
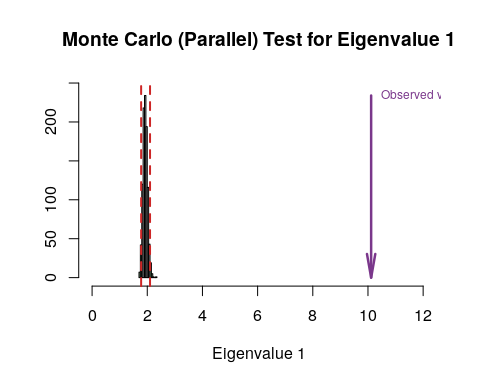
* With Bootstrap Ratio



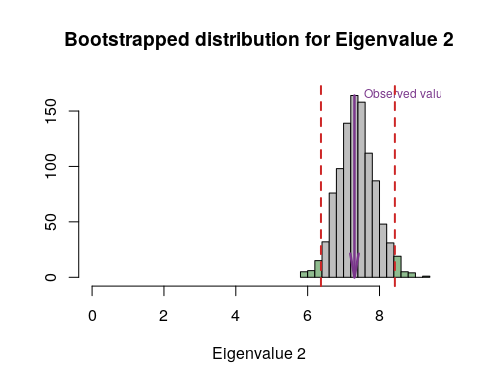
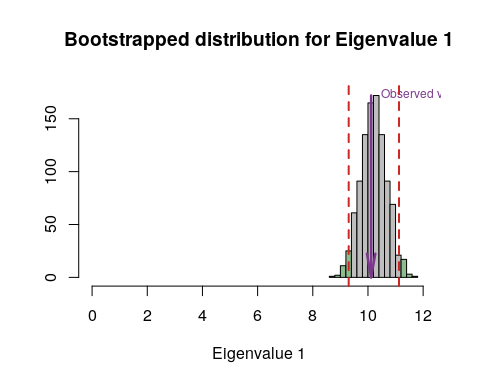
## Permutation Test



## Parallet Test



## Bootstrap Test



## Conclusion

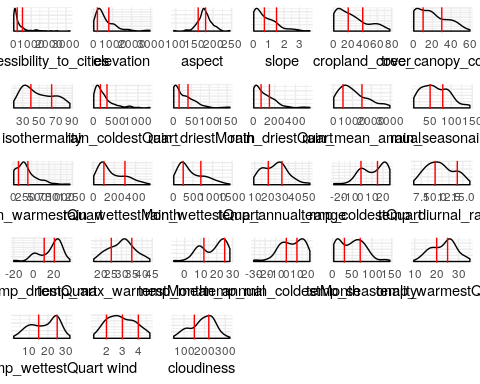
* Component 1:
  + Rows: Normal & Happy
  + Columns: Cloudiness & Rain vs Cropland, Aspect, Elevation
  + Interpret: People in countries with more Cloudiness, Trees and Rain tends to be happier.
* Component 7:
  + Rows: Happy & Unhappy
  + Columns: Temp and Rain vs Accessibility and Cropland
  + Interpret: Rain and Temp seems to be main reason for unhappiness and Cropland is important for Happiness.
* Component 9:
  + Rows: Happy & Very Happy
  + Columns: Temp vs Rain
  + Interpret: Rain and Temp seems to be main reason for Happiness. *This contradicts with Component 7 and 1*.

# Discriminant Correspondence Analysis

## Method: DiCA

## Density plot

Let’s observe the distribution of each variables to get an intuition of how we can bin these variables. It’s important to have nearly equal number of observations in the each bin and to try to cut the variables in a way to so that each new binned distribution is nearly Gaussian. We can also verify that our binning is appropiate by calculating Spearman Correlation for each of original variable and binned variable, the correlation coefficient should be close to 1.



## Binning

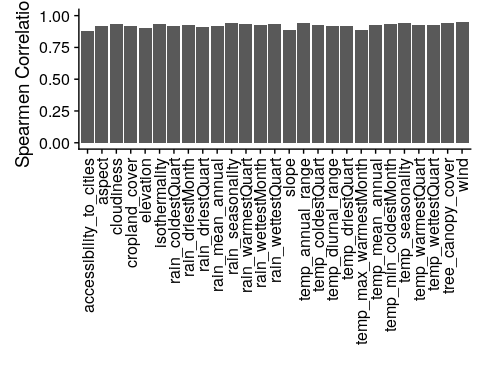
Structure of Data after binning based on above observation.

## 'data.frame': 137 obs. of 27 variables:  
## $ accessibility\_to\_cities: int 2 1 3 2 2 1 3 1 1 1 ...  
## $ elevation : int 3 2 2 3 2 3 2 3 2 1 ...  
## $ aspect : int 3 3 3 2 1 3 3 2 1 2 ...  
## $ slope : int 3 3 1 1 1 3 1 2 2 1 ...  
## $ cropland\_cover : int 1 2 1 1 2 2 1 2 2 3 ...  
## $ tree\_canopy\_cover : int 1 2 1 2 1 1 1 3 1 2 ...  
## $ isothermality : int 1 1 2 2 2 1 2 1 1 2 ...  
## $ rain\_coldestQuart : int 1 3 1 1 1 1 1 2 1 1 ...  
## $ rain\_driestMonth : int 1 3 1 1 2 2 1 3 2 1 ...  
## $ rain\_driestQuart : int 1 2 1 1 1 1 1 3 1 1 ...  
## $ rain\_mean\_annual : int 1 2 1 2 2 2 1 2 1 3 ...  
## $ rain\_seasonailty : int 3 1 2 3 1 1 2 1 1 3 ...  
## $ rain\_warmestQuart : int 1 2 1 3 2 2 2 3 1 3 ...  
## $ rain\_wettestMonth : int 1 2 1 2 1 1 1 2 1 3 ...  
## $ rain\_wettestQuart : int 1 2 1 2 1 1 1 2 1 3 ...  
## $ temp\_annual\_range : int 3 2 3 2 2 3 2 2 3 2 ...  
## $ temp\_coldestQuart : int 1 2 2 3 2 1 2 1 2 3 ...  
## $ temp\_diurnal\_range : int 3 1 3 2 2 2 2 1 1 1 ...  
## $ temp\_driestQuart : int 3 2 3 2 2 1 2 1 2 2 ...  
## $ temp\_max\_warmestMonth : int 2 2 3 2 2 1 3 1 2 2 ...  
## $ temp\_mean\_annual : int 1 1 2 2 2 1 2 1 1 3 ...  
## $ temp\_min\_coldestMonth : int 1 1 2 2 2 1 2 1 1 3 ...  
## $ temp\_seasonality : int 3 2 3 1 2 3 2 2 3 2 ...  
## $ temp\_warmestQuart : int 2 1 3 2 2 1 3 1 2 3 ...  
## $ temp\_wettestQuart : int 1 1 2 2 2 1 2 1 1 3 ...  
## $ wind : int 3 2 4 2 4 1 4 2 2 2 ...  
## $ cloudiness : int 1 2 1 2 2 2 1 3 2 2 ...

## Spearman Correlation

Let’s observe correlation between original data and binned data to make sure that neither the correlation ceofficient is too low or perfect.

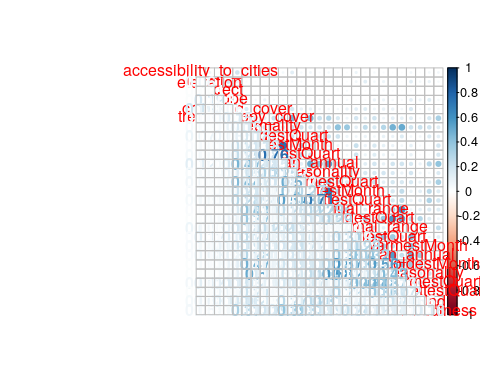
cor\_spear <- mapply(function(x,y) cor(x, as.integer(y),method = "spearman"), country\_env\_df\_for\_pca, country\_env\_df\_for\_mca)  
#columns = colnames(country\_env\_df\_for\_pca)  
#cor\_df <- data.frame(col = columns, corr = cor\_spear)  
cor\_p <- as.data.frame(cor\_spear)  
  
ggplot(data=cor\_p, aes(x=rownames(cor\_p), y=cor\_p$cor\_spear)) +  
 geom\_bar(stat="identity") + theme(axis.text.x=element\_text(angle=90,hjust=1,vjust=0.5)) +  
 xlab("") + ylab("Spearmen Correlation") + ylim(0, 1)



## Heatmap

* For binned data

Visually analyze multicollinearity in the system of the original data



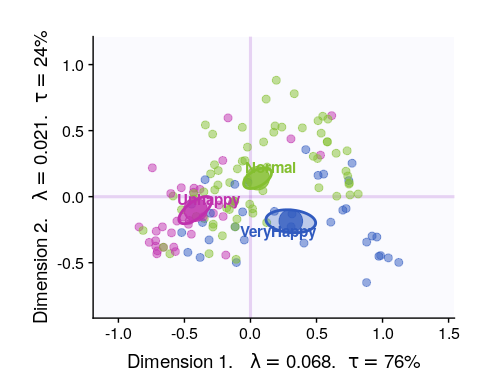
## Scree Plot

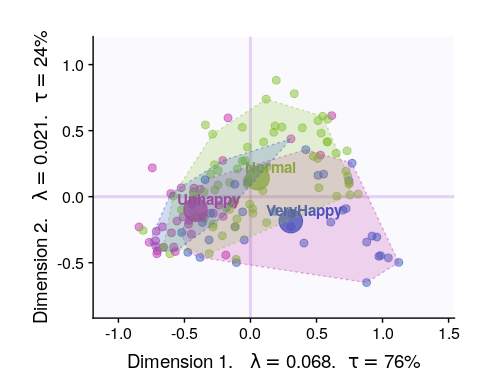
Gives amount of information explained by corresponding component. Gives an intuition to decide which components best represent data in order to answer the research question.

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

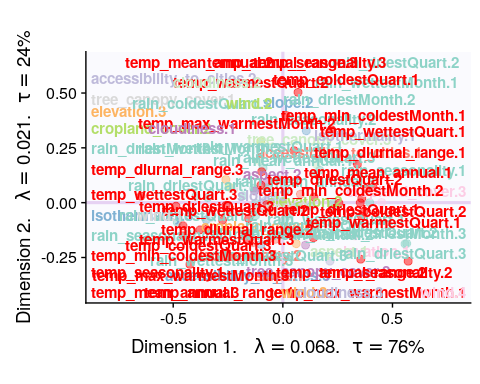


## Factor Scores

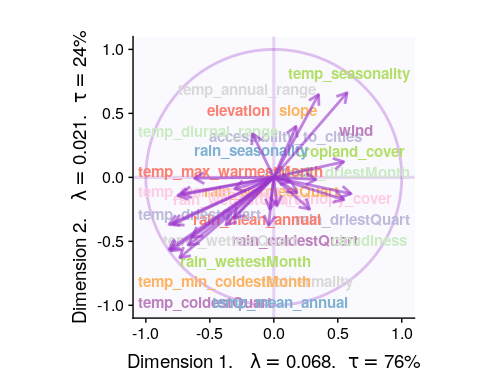
* With Confidence Interval 
* With Tolerance Interval



## Loadings



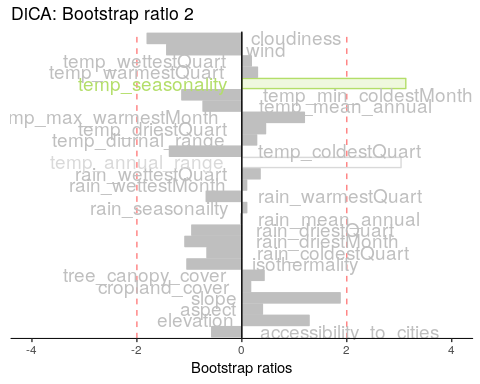
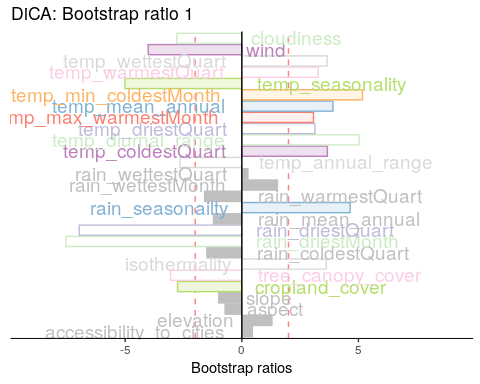
## Loadings (correlation plot)



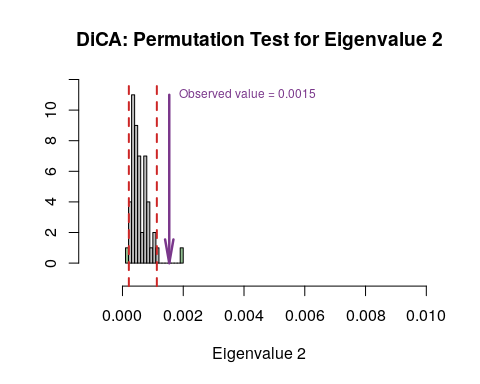
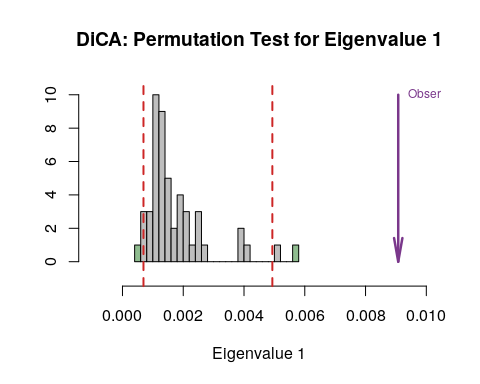
## Most Contributing Variables (Inference)

Let’s plot variable contributions against each chosen components i.e. 1, 2.

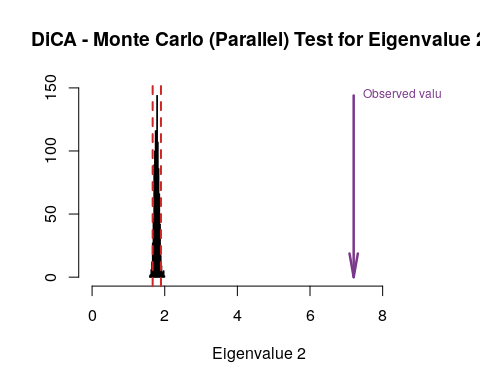
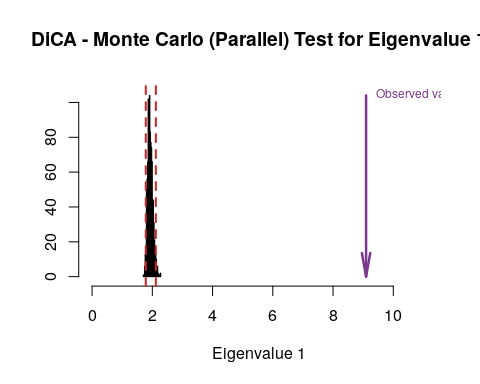
* With Bootstrap Ratio



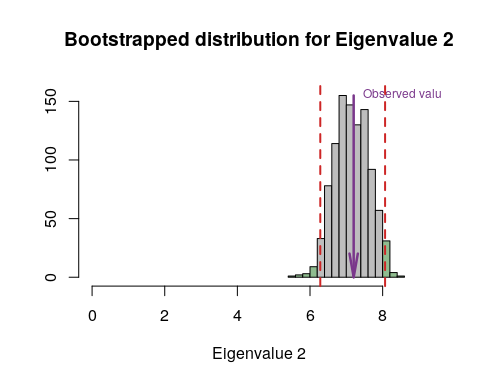
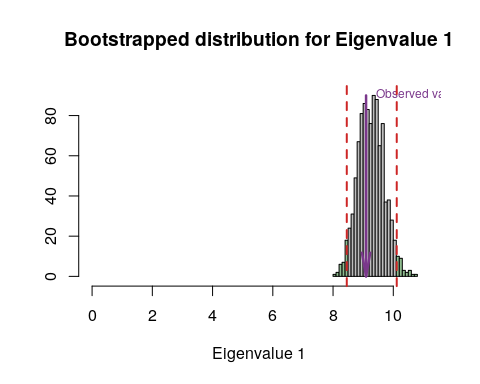
## Permutation Test



## Parallet Test



## Bootstrap Test



## Conclusion

Among PCA, MCA, BADA and DiCA, DiCA is able to best seperate the data based on levels of happiness. It seems increase in temperature is correlated with increase in happiness and increase in rain decreases happiness.

# Multiple Factor Analyis

## Method: MFA

## MFA

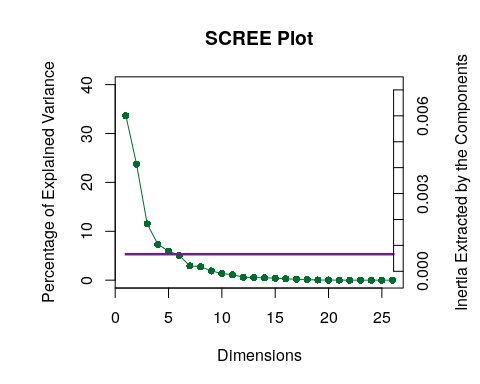
We have divided the data into 3 tables, separate tables for rain and temperature related columns and 3rd table for rest of the columns.

## [1] "Preprocessed the Rows of the data matrix using: None"  
## [1] "Preprocessed the Columns of the data matrix using: Center\_1Norm"  
## [1] "Preprocessed the Tables of the data matrix using: MFA\_Normalization"  
## [1] "Preprocessing Completed"  
## [1] "Optimizing using: None"  
## [1] "Processing Complete"

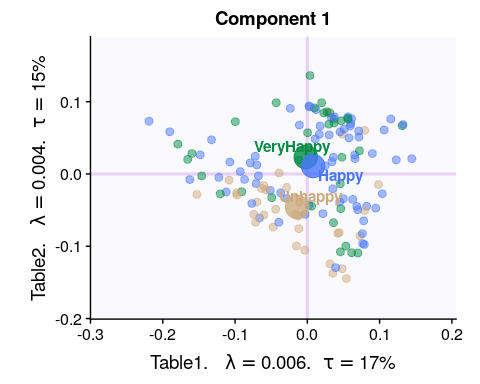
## Scree Plot

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.

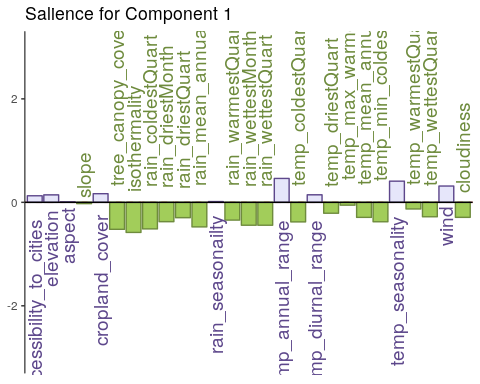
P.S. The most contribution component may not always be most useful for a given research question.

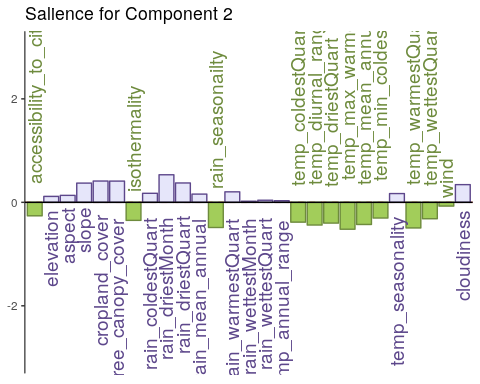


## Factor Scores

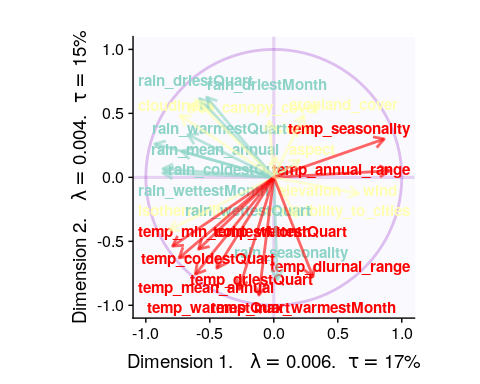


## Loadings





## Correlation Circle



## Conclusion

Here Component 2 seems to best seperate Happiness levels. Let’s compare Component 2 for both tables.

* Table 1 & 2 Component 2
  + Latent Variables: Very Happy vs Unhappy (for Rain and Temperature both)
  + Salience:
    - Rain: It seems dryness and wetness at a montly scale have more effect than coldness or yearly patterns.
    - Temperature: All temperature variations at a monthly and yearly scale seems to impact happiness.

# Other Methods

## 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

* Data: Measurements of Weekly Earnings per Race
* Rows: There are 6 observations representing Asian/White/Black, Men/Woman.
* Columns: Total 6 variables grouping people based on Decile and Quartile ranges of their weekly income.

## White.men White.women Black.men Black.Women  
## 1st decile 412 374 361 331  
## 1st quartile 594 506 483 423  
## 2nd quartile 920 743 680 615  
## 3rd quartile 1467 1140 1046 935  
## 9th decile 2278 1726 1551 1453  
## Total people (in thousands) 48746 36698 6445 7142  
## Asian.Men Asian.Women Hispanic.Men  
## 1st decile 420 385 358  
## 1st quartile 648 551 451  
## 2nd quartile 1129 877 631  
## 3rd quartile 1860 1411 979  
## 9th decile 2699 2024 1498  
## Total people (in thousands) 3684 2954 11142  
## Hispanic.Women  
## 1st decile 320  
## 1st quartile 404  
## 2nd quartile 566  
## 3rd quartile 830  
## 9th decile 1266  
## Total people (in thousands) 7168

However, here we can see that it may not be advisable to include Quartile and Decile intervals in the same analysis. Hence, we go ahead with Quartile Ranges only.

4 Pianist for each of 3 Composers

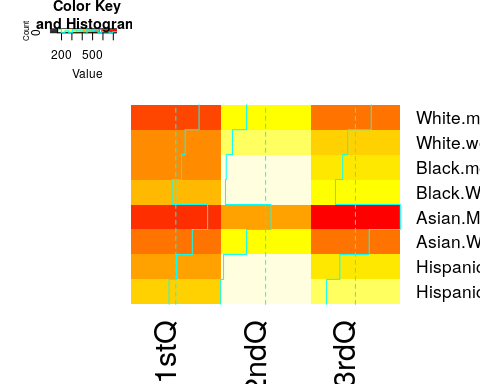
|  |  |  |  |
| --- | --- | --- | --- |
|  | 1stQ | 2ndQ | 3rdQ |
| White.men | 594 | 326 | 547 |
| White.women | 506 | 237 | 397 |
| Black.men | 483 | 197 | 366 |
| Black.Women | 423 | 192 | 320 |
| Asian.Men | 648 | 481 | 731 |
| Asian.Women | 551 | 326 | 534 |
| Hispanic.Men | 451 | 180 | 348 |
| Hispanic.Women | 404 | 162 | 264 |

* Research Question
  + Does total earning of different races differ.
  + Which race get less than median salary (2nd Quartile)

### Heatmap

## Warning in heatmap.2(WE\_data, Colv = FALSE, Rowv = FALSE, col =  
## rev(heat.colors(16))): Discrepancy: Rowv is FALSE, while dendrogram is  
## `both'. Omitting row dendogram.

## Warning in heatmap.2(WE\_data, Colv = FALSE, Rowv = FALSE, col =  
## rev(heat.colors(16))): Discrepancy: Colv is FALSE, while dendrogram is  
## `column'. Omitting column dendogram.

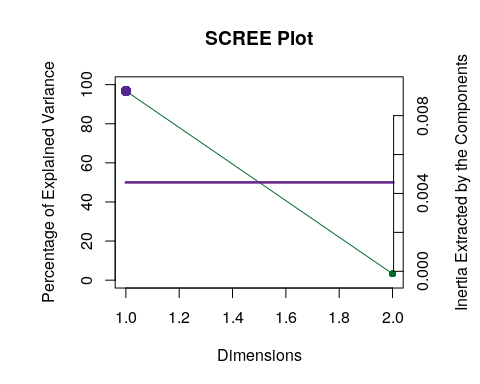


### Scree Plot

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.

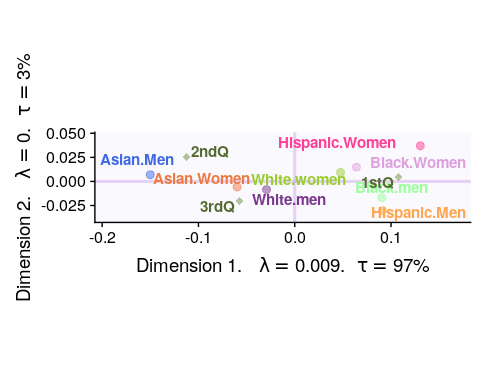
PTCA4CATA::PlotScree(ev = resCA.sym$ExPosition.Data$eigs,  
 p.ev = we\_data\_inf$Inference.Data$components$p.vals,  
 title = 'SCREE Plot',  
 plotKaiser = TRUE  
)



### Factor Scores

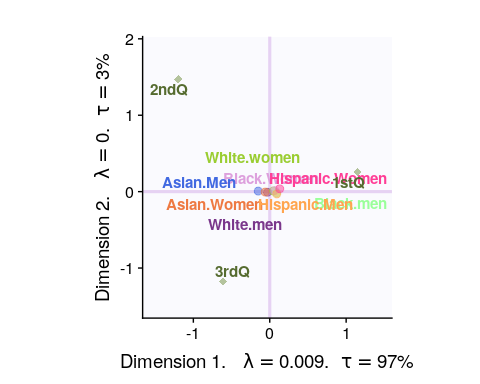
#### Symmetric Plot

map.IJ.sym <- symMap$baseMap + symMap$I\_labels + symMap$I\_points +  
 symMap$J\_labels + symMap$J\_points + labels4CA  
print(map.IJ.sym)



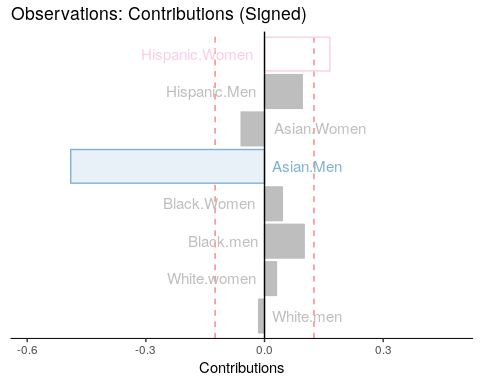
#### Asymmetric Plot

map.IJ.asym <- asymMap$baseMap + asymMap$I\_labels +   
 asymMap$I\_points + asymMap$J\_labels +   
 asymMap$J\_points + labels4CA  
print(map.IJ.asym)

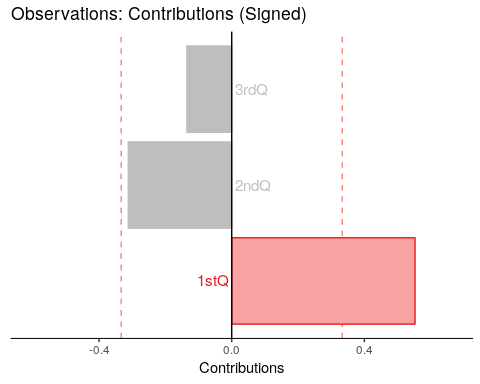


### Most Contributing Variables

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 )

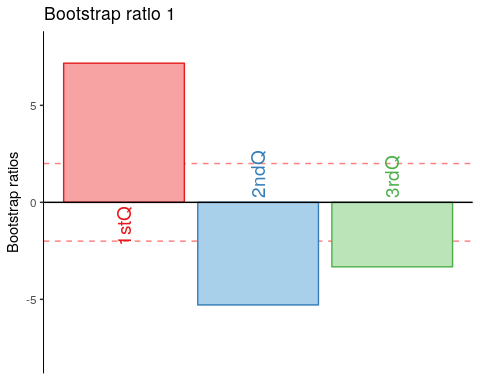


PTCA4CATA::PrettyBarPlot2(ctr.J[,1],   
 threshold = 1 / NROW(ctr.J),   
 font.size = 4,   
 color4bar = color4J,   
 color4ns = 'grey',   
 main = 'Observations: Contributions (Signed)',   
 ylab = 'Contributions', ylim = c(1.2\*min(ctr.J),  
 1.2\*max(ctr.J) ),   
 horizontal = FALSE )

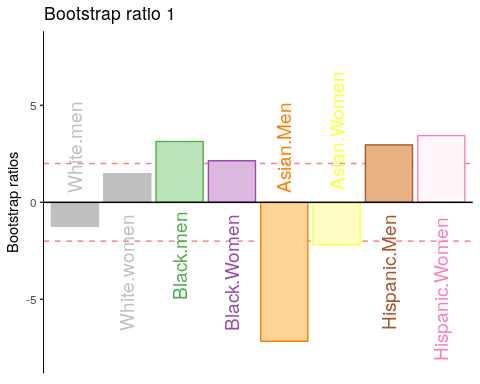


### Inference CA

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(1.2\*min(BR[,laDim]), 1.2\*max(BR[,laDim]))  
)  
print(ba001.BR1)



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)



rm(list = ls())  
  
devtools::install\_github('HerveAbdi/PTCA4CATA')

## Skipping install of 'PTCA4CATA' from a github remote, the SHA1 (0a982b85) has not changed since last install.  
## Use `force = TRUE` to force installation

suppressMessages(library(PTCA4CATA))  
# PTCA4CATA should first to avoid conflict with TInPosition  
suppressMessages(library(ExPosition))  
#suppressMessages(library(InPosition))  
#suppressMessages(library(TInPosition))  
suppressMessages(library(ggplot2))  
suppressMessages(library(dplyr))  
suppressMessages(library(officer))  
suppressMessages(library(flextable))  
suppressMessages(library(rvg))  
suppressMessages(library(useful))  
suppressMessages(library(DistatisR))  
library(RColorBrewer)  
  
  
# Source the function file:  
#  
# install.packages('TExPosition')  
# install.packages('MExPosition')  
library(MExPosition)  
rm(list = ls())

## DiSTATIS

To be filled

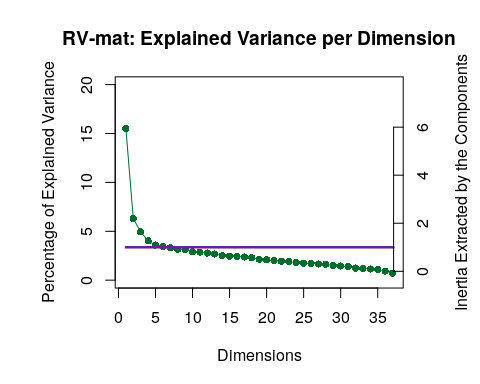
### Dataset

4 Pianist for each of 3 Composers

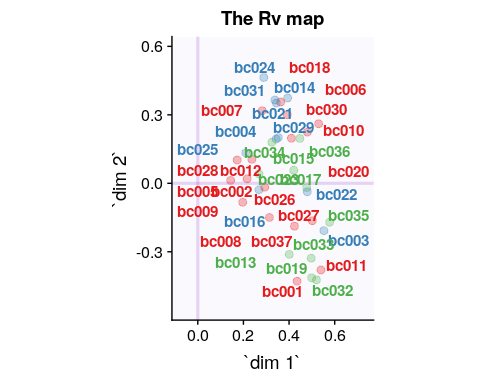
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | bc001 | bc002 | bc003 | bc004 | bc005 | bc006 | bc007 | bc008 | bc009 | bc010 | bc011 | bc012 | bc013 | bc014 | bc015 | bc016 | bc017 | bc018 | bc019 | bc020 | bc021 | bc022 | bc023 | bc024 | bc025 | bc026 | bc027 | bc028 | bc029 | bc030 | bc031 | bc032 | bc033 | bc034 | bc035 | bc036 | bc037 |
| Bach.Arrau.1 | 1 | 2 | 1 | 2 | 3 | 1 | 1 | 3 | 1 | 1 | 2 | 2 | 2 | 2 | 3 | 1 | 1 | 2 | 1 | 3 | 3 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 3 | 1 | 1 | 1 | 3 | 3 | 1 | 2 |
| Bach.Arrau.2 | 2 | 1 | 1 | 2 | 3 | 1 | 1 | 2 | 1 | 1 | 2 | 3 | 1 | 3 | 2 | 3 | 2 | 3 | 1 | 1 | 1 | 2 | 2 | 3 | 2 | 2 | 3 | 3 | 2 | 3 | 3 | 1 | 2 | 3 | 3 | 3 | 1 |
| Bach.Arrau.3 | 1 | 3 | 2 | 1 | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 2 | 3 | 2 | 1 | 3 | 1 | 1 | 1 | 1 | 3 | 1 | 3 | 1 | 1 | 1 | 3 | 3 | 1 | 2 | 2 | 3 | 3 | 3 | 1 | 1 | 1 |
| Bach.Baren.4 | 1 | 3 | 2 | 2 | 2 | 1 | 2 | 3 | 1 | 1 | 3 | 1 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 3 | 1 | 1 | 1 | 1 | 1 | 3 | 3 | 1 | 3 | 1 | 3 | 3 | 3 | 1 | 1 | 1 |
| Bach.Baren.5 | 1 | 1 | 2 | 3 | 3 | 3 | 3 | 3 | 1 | 2 | 2 | 2 | 3 | 3 | 1 | 3 | 1 | 3 | 1 | 3 | 2 | 1 | 1 | 3 | 3 | 1 | 3 | 1 | 2 | 1 | 1 | 2 | 3 | 1 | 3 | 3 | 3 |
| Bach.Baren.6 | 1 | 3 | 3 | 2 | 3 | 1 | 2 | 1 | 3 | 1 | 3 | 2 | 3 | 1 | 2 | 2 | 3 | 2 | 1 | 2 | 1 | 1 | 3 | 3 | 1 | 1 | 2 | 2 | 1 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 2 |
| Bach.Pires.7 | 1 | 2 | 3 | 3 | 2 | 3 | 1 | 3 | 3 | 1 | 2 | 1 | 3 | 3 | 2 | 3 | 1 | 3 | 3 | 3 | 1 | 3 | 2 | 1 | 2 | 3 | 1 | 3 | 1 | 1 | 1 | 2 | 3 | 1 | 3 | 3 | 1 |
| Bach.Pires.8 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 2 | 1 | 3 | 1 | 3 | 2 | 2 | 3 | 2 | 2 | 2 | 2 | 2 | 3 | 2 | 1 | 1 | 1 | 2 | 2 | 2 | 3 | 3 | 2 | 2 | 2 | 3 | 2 | 1 | 3 |
| Bach.Pires.9 | 1 | 3 | 2 | 1 | 3 | 3 | 1 | 2 | 2 | 2 | 1 | 2 | 1 | 3 | 2 | 3 | 3 | 3 | 1 | 2 | 1 | 2 | 2 | 1 | 3 | 2 | 3 | 1 | 1 | 1 | 1 | 1 | 2 | 1 | 3 | 3 | 1 |
| Bach.Richt.10 | 2 | 1 | 1 | 3 | 2 | 2 | 2 | 2 | 2 | 3 | 1 | 3 | 2 | 1 | 3 | 2 | 2 | 2 | 2 | 1 | 2 | 3 | 2 | 2 | 3 | 3 | 2 | 3 | 3 | 2 | 2 | 1 | 1 | 2 | 2 | 1 | 3 |
| Bach.Richt.11 | 3 | 3 | 3 | 2 | 1 | 3 | 1 | 3 | 3 | 1 | 3 | 3 | 3 | 2 | 2 | 2 | 3 | 1 | 3 | 2 | 1 | 1 | 3 | 1 | 2 | 2 | 1 | 1 | 2 | 3 | 1 | 3 | 3 | 2 | 1 | 2 | 2 |
| Bach.Richt.12 | 2 | 1 | 1 | 1 | 1 | 2 | 3 | 3 | 3 | 3 | 1 | 3 | 2 | 1 | 3 | 2 | 1 | 2 | 2 | 1 | 2 | 3 | 3 | 2 | 3 | 2 | 2 | 3 | 3 | 2 | 3 | 1 | 1 | 2 | 2 | 1 | 2 |
| Beet.Arrau.13 | 1 | 2 | 2 | 2 | 3 | 1 | 2 | 1 | 1 | 2 | 2 | 3 | 1 | 2 | 2 | 1 | 1 | 2 | 3 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 2 | 3 | 3 | 2 | 2 | 3 | 3 | 2 | 1 | 1 | 1 |
| Beet.Arrau.14 | 3 | 2 | 3 | 3 | 1 | 3 | 1 | 1 | 2 | 1 | 3 | 1 | 3 | 3 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 2 | 1 | 1 | 2 | 1 | 1 | 2 | 1 | 1 | 3 | 3 | 3 | 1 | 1 | 2 | 1 |
| Beet.Arrau.15 | 3 | 2 | 3 | 1 | 2 | 2 | 1 | 1 | 2 | 3 | 3 | 2 | 1 | 1 | 3 | 2 | 1 | 1 | 3 | 1 | 2 | 1 | 1 | 3 | 3 | 3 | 1 | 2 | 3 | 3 | 1 | 3 | 2 | 3 | 1 | 1 | 1 |
| Beet.Baren.16 | 1 | 3 | 3 | 3 | 2 | 3 | 2 | 1 | 2 | 1 | 3 | 1 | 3 | 1 | 1 | 1 | 1 | 3 | 3 | 1 | 3 | 2 | 3 | 3 | 3 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 3 | 2 | 1 | 2 | 2 |
| Beet.Baren.17 | 1 | 3 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 1 | 3 | 3 | 1 | 1 | 3 | 3 | 3 | 3 | 3 | 1 | 2 | 1 | 1 | 1 | 3 | 3 | 2 | 1 | 1 | 2 | 3 | 1 | 3 | 2 | 1 |
| Beet.Baren.18 | 3 | 1 | 3 | 2 | 3 | 1 | 1 | 3 | 2 | 1 | 3 | 2 | 3 | 2 | 1 | 1 | 1 | 2 | 3 | 2 | 1 | 1 | 1 | 2 | 1 | 2 | 1 | 3 | 2 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 2 |
| Beet.Pires.19 | 1 | 3 | 3 | 3 | 2 | 2 | 3 | 1 | 2 | 1 | 3 | 3 | 2 | 2 | 2 | 1 | 3 | 2 | 3 | 3 | 1 | 1 | 1 | 1 | 3 | 2 | 1 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 2 |
| Beet.Pires.20 | 3 | 3 | 3 | 2 | 1 | 1 | 2 | 1 | 2 | 2 | 2 | 3 | 3 | 2 | 1 | 1 | 1 | 1 | 3 | 2 | 3 | 1 | 3 | 1 | 2 | 2 | 1 | 3 | 1 | 3 | 1 | 3 | 3 | 2 | 1 | 2 | 2 |
| ``` |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

## [1] Bootstrap On Factor Scores. Iterations #:   
## [2] 1000

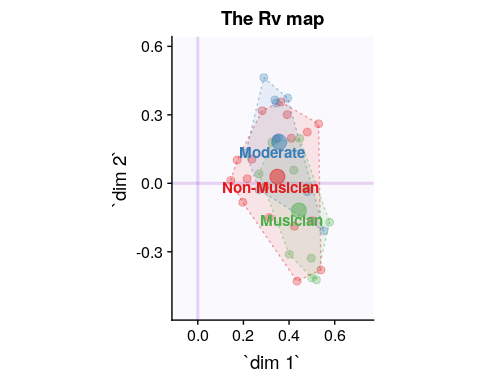
### SCREE Plot

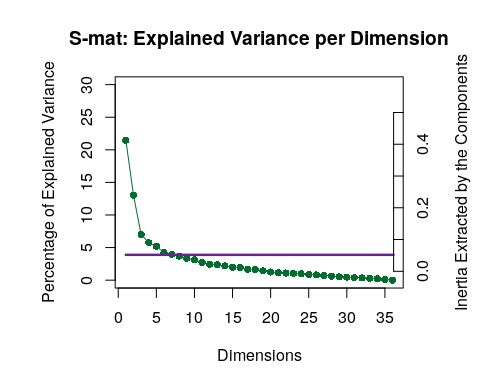


### Plotting Assessor Matrix

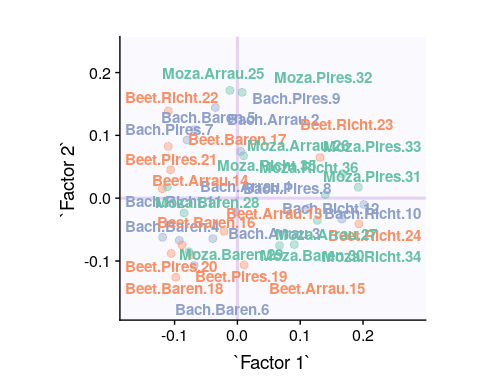


### ConvexHull





### I Set



## Warning: Removed 3 rows containing missing values (geom\_text\_repel).

