Global CLI Project

2022-01-19

Global CLI

Basic stuff

```
# Load data
cli <- read.csv('cost-of-living_v2.csv')

dim(cli)
```

```
## [1] 4956   58
```

```
varData <- setNames(stack(sapply(cli, class))[2:1], c('variable', 'class'))

descriptions <- read.csv('Descriptions.csv')
descriptions <- descriptions[, c('Column', 'Description')]
gt::gt(descriptions)
```

Column	Description
city	Name of the city
country	Name of the country
x1	Meal, Inexpensive Restaurant (USD)
x2	Meal for 2 People, Mid-range Restaurant, Three-course (USD)
x3	McMeal at McDonalds (or Equivalent Combo Meal) (USD)
x4	Domestic Beer (0.5 liter draught, in restaurants) (USD)
x5	Imported Beer (0.33 liter bottle, in restaurants) (USD)
x6	Cappuccino (regular, in restaurants) (USD)
x7	Coke/Pepsi (0.33 liter bottle, in restaurants) (USD)
x8	Water (0.33 liter bottle, in restaurants) (USD)
x9	Milk (regular), (1 liter) (USD)
x10	Loaf of Fresh White Bread (500g) (USD)
x11	Rice (white), (1kg) (USD)
x12	Eggs (regular) (12) (USD)
x13	Local Cheese (1kg) (USD)
x14	Chicken Fillets (1kg) (USD)
x15	Beef Round (1kg) (or Equivalent Back Leg Red Meat) (USD)
x16	Apples (1kg) (USD)
x17	Banana (1kg) (USD)
x18	Oranges (1kg) (USD)
x19	Tomato (1kg) (USD)
x20	Potato (1kg) (USD)
x21	Onion (1kg) (USD)
x22	Lettuce (1 head) (USD)
x23	Water (1.5 liter bottle, at the market) (USD)
x24	Bottle of Wine (Mid-Range, at the market) (USD)
x25	Domestic Beer (0.5 liter bottle, at the market) (USD)

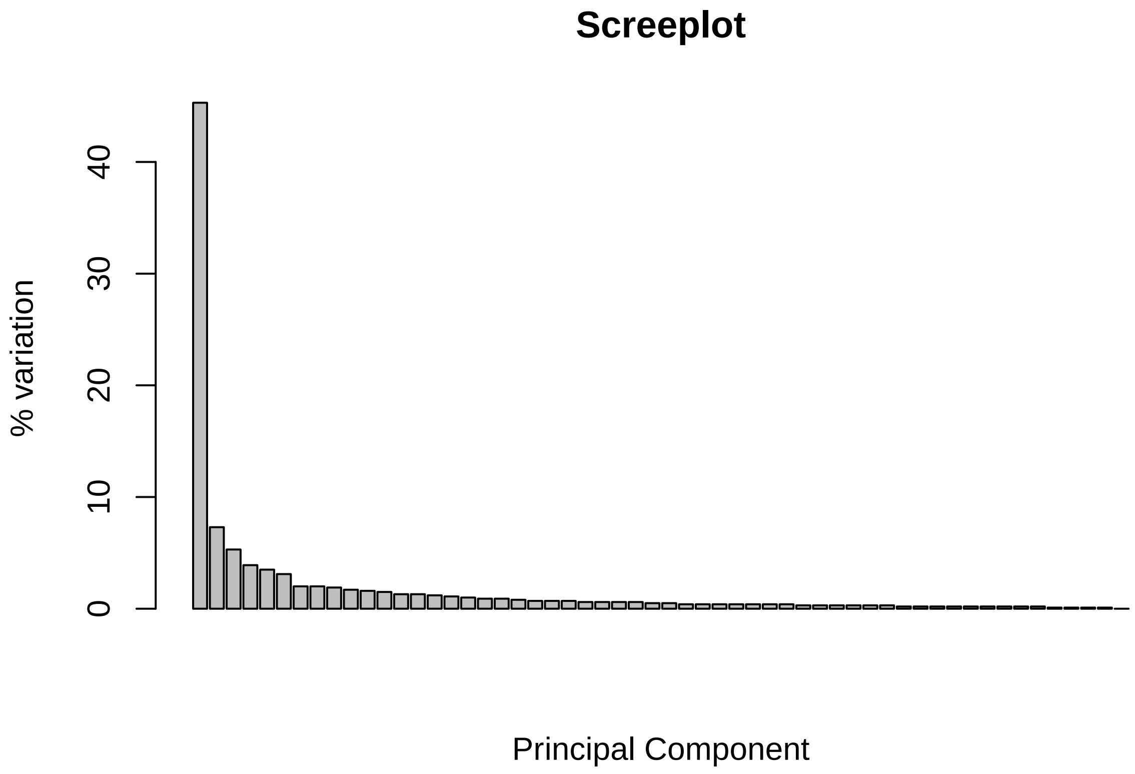
x26	Imported Beer (0.33 liter bottle, at the market) (USD)
x27	Cigarettes 20 Pack (Marlboro) (USD)
x28	One-way Ticket (Local Transport) (USD)
x29	Monthly Pass (Regular Price) (USD)
x30	Taxi Start (Normal Tariff) (USD)
x31	Taxi 1km (Normal Tariff) (USD)
x32	Taxi 1hour Waiting (Normal Tariff) (USD)
x33	Gasoline (1 liter) (USD)
x34	Volkswagen Golf 1.4 90 KW Trendline (Or Equivalent New Car) (USD)
x35	Toyota Corolla Sedan 1.6l 97kW Comfort (Or Equivalent New Car) (USD)
x36	Basic (Electricity, Heating, Cooling, Water, Garbage) for 85m2 Apartment (USD)
x37	1 min. of Prepaid Mobile Tariff Local (No Discounts or Plans) (USD)
x38	Internet (60 Mbps or More, Unlimited Data, Cable/ADSL) (USD)
x39	Fitness Club, Monthly Fee for 1 Adult (USD)
x40	Tennis Court Rent (1 Hour on Weekend) (USD)
x41	Cinema, International Release, 1 Seat (USD)
x42	Preschool (or Kindergarten), Full Day, Private, Monthly for 1 Child (USD)
x43	International Primary School, Yearly for 1 Child (USD)
x44	1 Pair of Jeans (Levis 501 Or Similar) (USD)
x45	1 Summer Dress in a Chain Store (Zara, H&M, . . .) (USD)
x46	1 Pair of Nike Running Shoes (Mid-Range) (USD)
x47	1 Pair of Men Leather Business Shoes (USD)
x48	Apartment (1 bedroom) in City Centre (USD)
x49	Apartment (1 bedroom) Outside of Centre (USD)
x50	Apartment (3 bedrooms) in City Centre (USD)
x51	Apartment (3 bedrooms) Outside of Centre (USD)
x52	Price per Square Meter to Buy Apartment in City Centre (USD)
x53	Price per Square Meter to Buy Apartment Outside of Centre (USD)
x54	Average Monthly Net Salary (After Tax) (USD)
x55	Mortgage Interest Rate in Percentages (%), Yearly, for 20 Years Fixed-Rate
data_quality	0 if Numbeo considers that more contributors are needed to increase data quality, else 1

PCA

```r
# Function for plotting on GG plot. Takes in data and PCs(int) to plot
pca.gg <- function(d, n1, n2) {
  # Variation
  p.var <- d$sdev^2
  p.var.per <- round(p.var / sum(p.var)*100, 1)

  ggD <- data.frame(Sample=rownames(d$x), X=d$x[,n1], Y=d$x[,n2])

  ggplot(data=ggD, aes(x=X, y=Y, label=Sample)) +
    geom_text() +
    xlab(paste("PC", n1, p.var.per[n1], "%", sep = " ")) +
    ylab(paste("PC", n2, p.var.per[n2], "%", sep=" ")) +
    ggtitle("PCA Graph")
}

# Loading scores
load_scores <- function(d, n, onCols) {
  loading_scores <- d$rotation[, n]
  col_scores <- abs(loading_scores)
  c_ranked <- sort(col_scores, decreasing = TRUE)
  test <- data.frame(c_ranked[1:10])
  colnames(test) <- "Loading Score"
  test$Column <- row.names(test)
  if (onCols) {
    test <- left_join(test, descriptions, by="Column")
    return(gt::gt(test))
  } else {
    return(gt::gt(test))
  }
}

# Scree plot
s_plot <- function(d){
  pca.var <- d$sdev^2
  pca.var.per <- round(pca.var/sum(pca.var)*100, 1)
  barplot(pca.var.per, main="Screeplot", xlab="Principal Component",
          ylab="% variation")
}
```

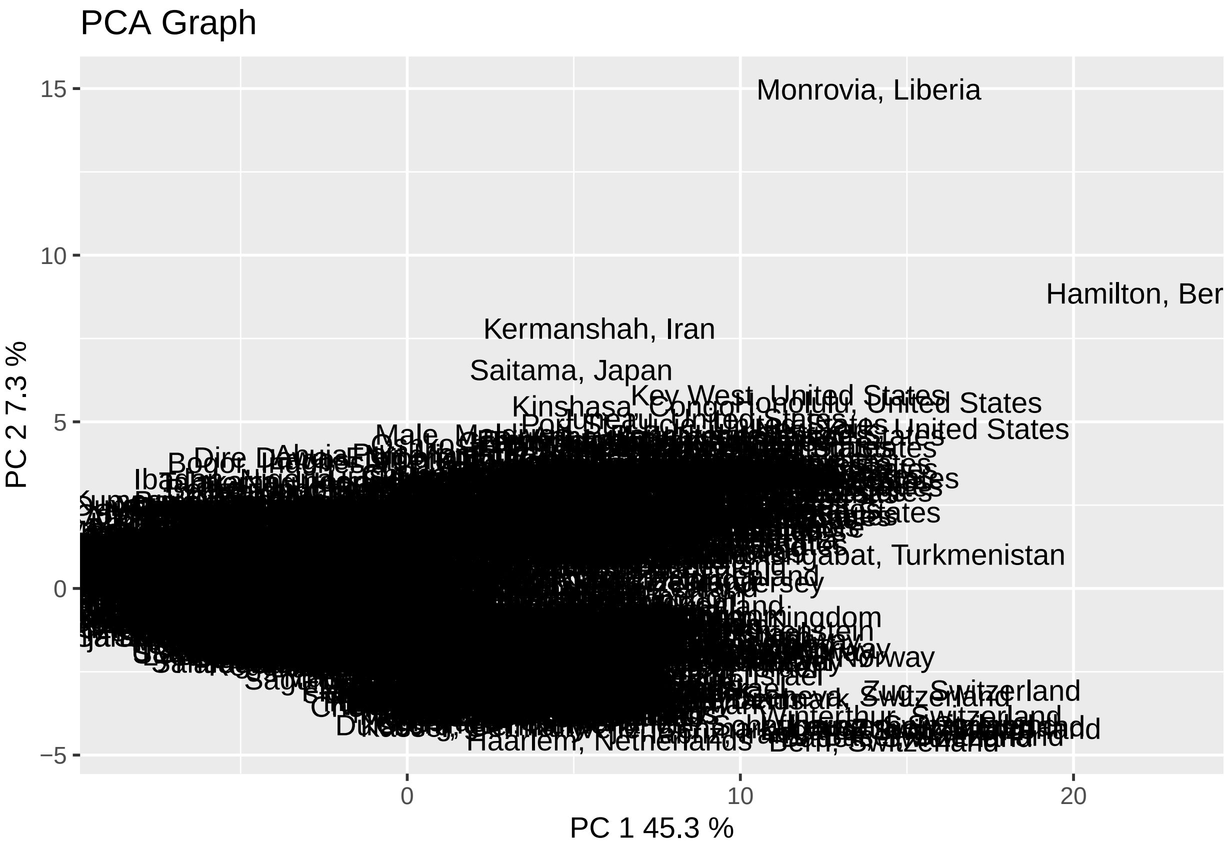
Setup

```
PCA.data <- cli[complete.cases(cli), ]
row.names(PCA.data) <- paste(PCA.data$city, PCA.data$country, sep = ", ")

PCA.data <- subset(PCA.data, select=-c(city, country))
PCA.1 <- prcomp(PCA.data, scale=TRUE, tol = 0.1)
s_plot(PCA.1)
```



First one!!

```
pca.gg(PCA.1, 1, 2)
```



```
# Which variables were most influential on where the companies were plotted for PC1 (x-axis?)
load_scores(PCA.1, 1, TRUE)
```

Loading Score	Column	Description
0.1787700	x41	Cinema, International Release, 1 Seat (USD)
0.1778548	x2	Meal for 2 People, Mid-range Restaurant, Three-course (USD)
0.1753416	x54	Average Monthly Net Salary (After Tax) (USD)
0.1730512	x1	Meal, Inexpensive Restaurant (USD)
0.1700596	x14	Chicken Fillets (1kg) (USD)
0.1684964	x12	Eggs (regular) (12) (USD)
0.1661474	x6	Cappuccino (regular, in restaurants) (USD)
0.1656012	x7	Coke/Pepsi (0.33 liter bottle, in restaurants) (USD)
0.1645299	x3	McMeal at McDonalds (or Equivalent Combo Meal) (USD)
0.1635456	x4	Domestic Beer (0.5 liter draught, in restaurants) (USD)

```
# Which variables were most influential on where the companies were plotted for PC2 (y-axis?)
load_scores(PCA.1, 2, TRUE)
```

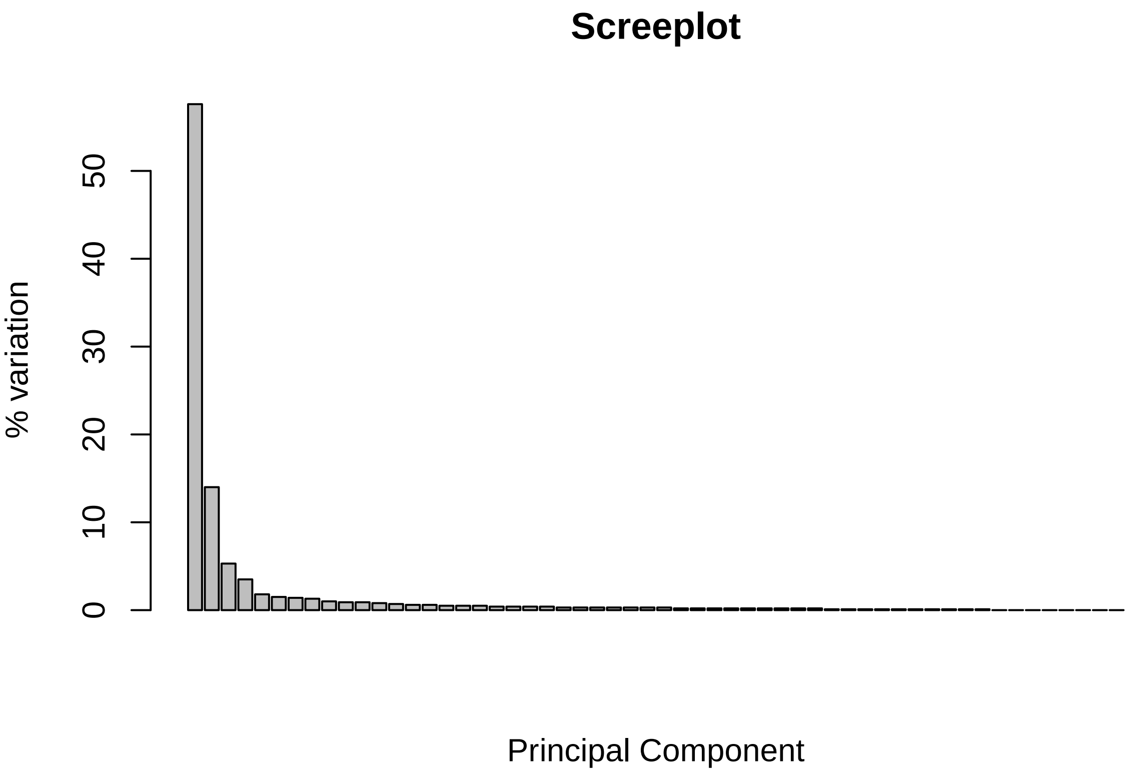
Loading Score	Column	Description
0.3234178	x33	Gasoline (1 liter) (USD)
0.3014550	x44	1 Pair of Jeans (Levis 501 Or Similar) (USD)
0.2508365	x16	Apples (1kg) (USD)
0.2166533	x21	Onion (1kg) (USD)

0.2107541	x24	Bottle of Wine (Mid-Range, at the market) (USD)
0.2101449	x20	Potato (1kg) (USD)
0.2009413	x18	Oranges (1kg) (USD)
0.1853395	x7	Coke/Pepsi (0.33 liter bottle, in restaurants) (USD)
0.1845426	x25	Domestic Beer (0.5 liter bottle, at the market) (USD)
0.1807534	x26	Imported Beer (0.33 liter bottle, at the market) (USD)

```
load_scores(PCA.1, 3, TRUE)
```

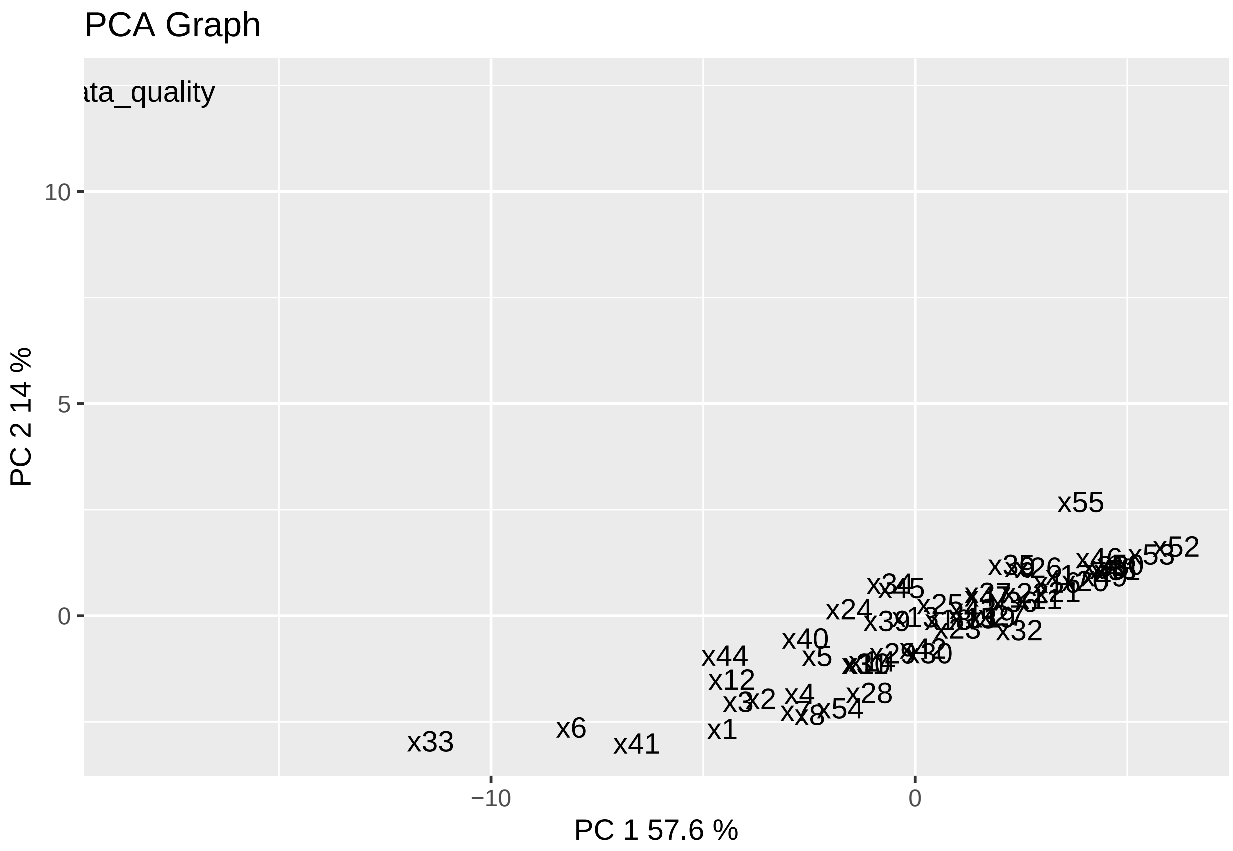
Loading Score	Column	Description
0.4072216	x46	1 Pair of Nike Running Shoes (Mid-Range) (USD)
0.3773023	x35	Toyota Corolla Sedan 1.6l 97kW Comfort (Or Equivalent New Car) (USD)
0.3521002	x34	Volkswagen Golf 1.4 90 KW Trendline (Or Equivalent New Car) (USD)
0.3183403	x45	1 Summer Dress in a Chain Store (Zara, H&M, . . .) (USD)
0.2455854	x9	Milk (regular), (1 liter) (USD)
0.2289242	x47	1 Pair of Men Leather Business Shoes (USD)
0.2197251	x17	Banana (1kg) (USD)
0.1804443	x55	Mortgage Interest Rate in Percentages (%), Yearly, for 20 Years Fixed-Rate
0.1575918	x28	One-way Ticket (Local Transport) (USD)
0.1553375	x44	1 Pair of Jeans (Levis 501 Or Similar) (USD)

```
PCA.data2 <- t(data.matrix(PCA.data))
PCA.data2 <- t(apply(PCA.data2, 1, function(x)(x-min(x))/(max(x)-min(x))))
PCA.2 <- prcomp(PCA.data2)

s_plot(PCA.2)
```



Second one!!

```
# Graph
pca.gg(PCA.2, 1, 2)
```



```
# What companies were most influential on where the variables were plotted for PC1?
load_scores(PCA.2, 1, FALSE)
```

Loading Score	Column
0.05501724	Basel, Switzerland
0.05468856	Zurich, Switzerland
0.05338798	Bern, Switzerland
0.05312936	Lucerne, Switzerland
0.05308558	Lausanne, Switzerland
0.05306766	Winterthur, Switzerland
0.05239151	Zug, Switzerland
0.05129129	Geneva, Switzerland
0.05064648	Vejle, Denmark
0.05026382	Odense, Denmark

```
# PC2?
load_scores(PCA.2, 2, FALSE)
```

Loading Score	Column
0.09309229	Baden, Switzerland
0.09103455	Schaffhausen, Switzerland
0.08035352	Vaduz, Liechtenstein

0.07368335	Drammen, Norway
0.06942890	Skien, Norway
0.06881351	Alesund, Norway
0.06415414	Santa Monica, United States
0.06397552	Esbjerg, Denmark
0.06333390	Svendborg, Denmark
0.06254870	Roskilde, Denmark