Analysis of CDC’s Health Data and UX Improvement Experiments

Sydney Murphy

2023-12-11

The cities.csv dataset is a subset of the 500 Cities Project of the Centers for Disease Control and Prevention (CDC). It contains population of 123 cities of the US. In particular, these are the columns available in the cities.csv dataset.

Loaded the cities.csv dataset in R. Drop the City column, since we will not need it in our analysis. I renamed the other columns.

```
library(ISLR2)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(survival)
library(survminer)

## Loading required package: ggplot2

## Loading required package: ggpubr

##
## Attaching package: 'survminer'

## The following object is masked from 'package:survival':
##
##     myeloma

library(ggplot2)
library(readr)
library(stats)




#Load the dataset
```

```
cities <-
read_csv("https://raw.githubusercontent.com/sydneymcolumbia/CMU/main/cities.c
sv")

## Rows: 123 Columns: 9

## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): City
## dbl (8): Arthritis among adults aged >=18 Years, Chronic kidney disease
amon...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.

#Drop 'City'
cities <- cities[ , !(names(cities) %in% c("City"))]

#Rename columns
colnames(cities) <- c("arthritis", "kidney_disease", "copd", "heart_disease",
                      "no_health_insurance", "diabetes", "high_cholesterol",
"no_exercise")

head(cities)

## # A tibble: 6 × 8
##   arthritis kidney_disease  copd heart_disease no_health_insurance
diabetes
##       <dbl>          <dbl> <dbl>         <dbl>               <dbl>
<dbl>
## 1     0.294          0.032 0.08          0.072               0.234
0.131
## 2     0.225          0.028 0.067         0.061               0.242
0.106
## 3     0.203          0.027 0.061         0.058               0.231
0.105
## 4     0.178          0.03  0.046         0.055               0.248
0.131
## 5     0.217          0.03  0.069         0.067               0.295
0.128
## 6     0.31           0.035 0.09          0.076               0.198
0.168
## # ℹ 2 more variables: high_cholesterol <dbl>, no_exercise <dbl>
```

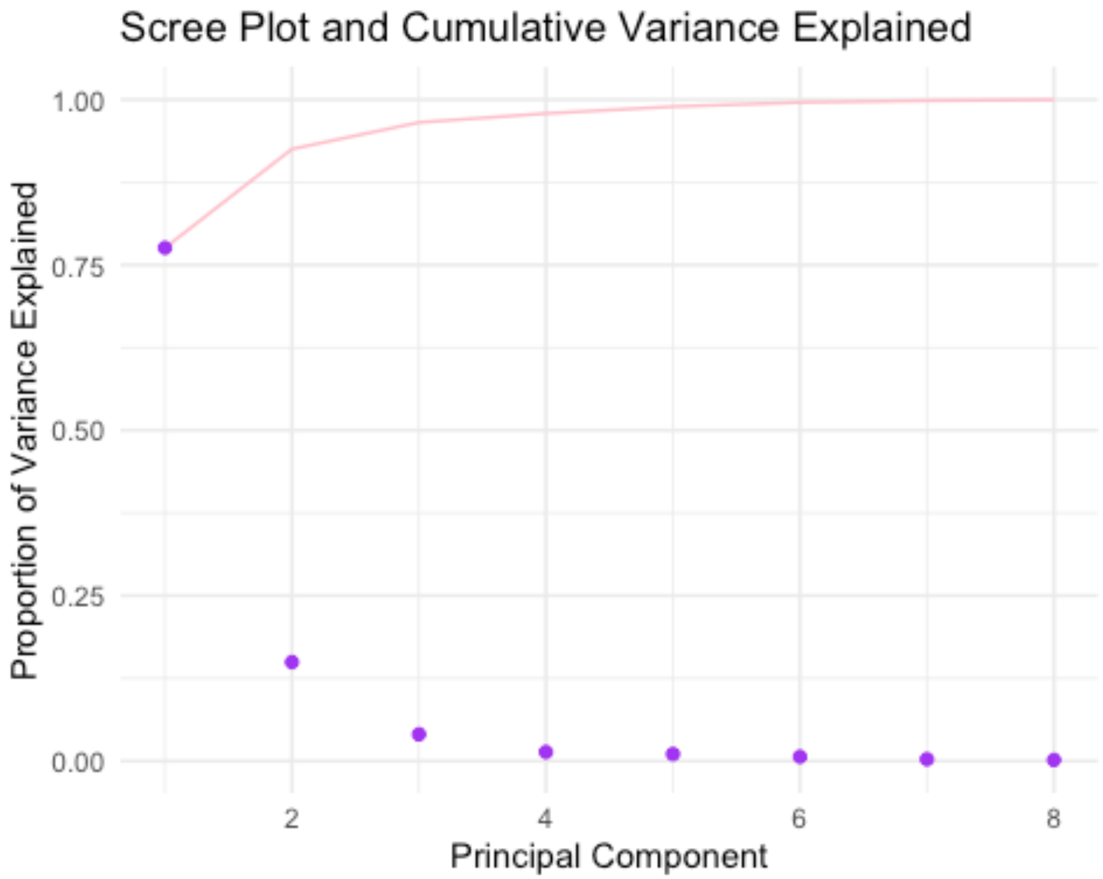
Applied Principal Component Analysis (PCA) to the dataset. Making sure to specify that the variables are centered (i.e., their empirical mean is set to 0) and also scaled (i.e., their empirical standard deviation is set to 1) in the prcomp function.

```
#PCA
pca_result <- prcomp(cities, center = TRUE, scale. = TRUE)

summary(pca_result)
```

```
## Importance of components:
##                            PC1    PC2     PC3     PC4     PC5     PC6
PC7
## Standard deviation     2.4915 1.0934 0.56754 0.32880 0.28829 0.22550
0.14525
## Proportion of Variance 0.7759 0.1494 0.04026 0.01351 0.01039 0.00636
0.00264
## Cumulative Proportion  0.7759 0.9254 0.96562 0.97913 0.98952 0.99588
0.99851
##                             PC8
## Standard deviation     0.10902
## Proportion of Variance 0.00149
## Cumulative Proportion  1.00000
```

Computed and plotted the proportion of variance explained by the principal components and the cumulative proportion of variance explained by the principal components.

```
#Extract proportion
var_explained <- summary(pca_result)$importance[2, ]

#Compute cumulative proportion
cum_var_explained <- cumsum(var_explained)

# Create a data frame for plotting
pc_data <- data.frame(PC = 1:length(var_explained),
                      Variance = var_explained,
                      CumulativeVariance = cum_var_explained)

#Plot
ggplot(pc_data, aes(x = PC)) +
  geom_line(aes(y = CumulativeVariance), colour = "pink") +
  geom_point(aes(y = Variance), colour = "purple") +
  labs(title = "Scree Plot and Cumulative Variance Explained",
       x = "Principal Component",
       y = "Proportion of Variance Explained") +
  theme_minimal()
```



The nominal dimension of this dataset is 8 (i.e., we have 8 variables available in total). Based on the plot of the cumulative proportion of variance explained by the principal components that I produced, I explain what I think is the effective dimensionality of this dataset (i.e., are the observations in these data concentrated on a smaller subspace and what is the dimension of this subspace).

```
cat ("The effective dimensionality of this dataset is 2. Most of the
information can be summarized just by looking at 2 new variables created from
the original 8. Based on these results, the first principal component alone
explains 77.59% of the variance, and the first two principal components
together explain 92.54% of the variance. These 2 new variables (first 2
principal components from the PCA) together explain over 90% of the variation
in the entire dataset.")
```

```
## The effective dimensionality of this dataset is 2. Most of the information
can be summarized just by looking at 2 new variables created from the
original 8. Based on these results, the first principal component alone
explains 77.59% of the variance, and the first two principal components
together explain 92.54% of the variance. These 2 new variables (first 2
principal components from the PCA) together explain over 90% of the variation
in the entire dataset.
```

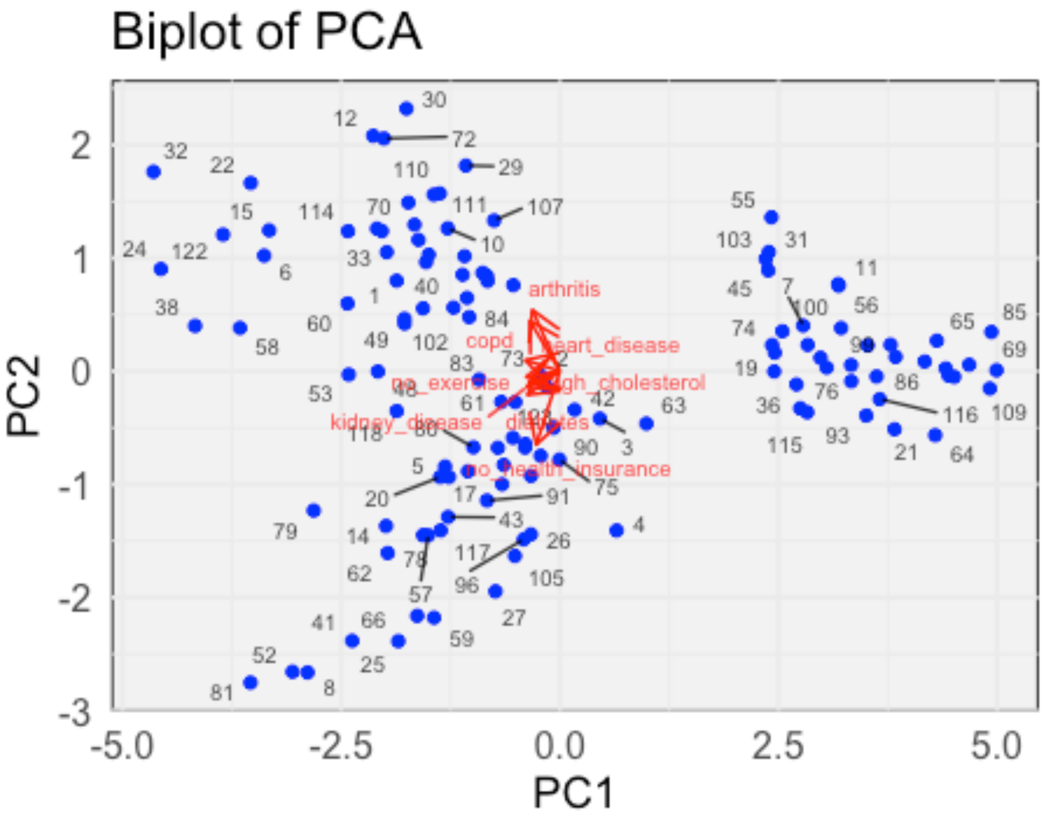
Computed the correlation matrix for the variables of the cities.csv dataset. After inspecting the correlation matrix. I was not surprised that PCA was successful in reducing the dimensionality of this dataset.

```
#Correlation matrix
correlation_matrix <- cor(cities)
print(correlation_matrix)

##                     arthritis kidney_disease      copd heart_disease
## arthritis           1.0000000      0.6454906 0.9426813     0.8262763
## kidney_disease      0.6454906      1.0000000 0.7317619     0.9257972
## copd                0.9426813      0.7317619 1.0000000     0.9023038
## heart_disease       0.8262763      0.9257972 0.9023038     1.0000000
## no_health_insurance 0.1252281      0.7249104 0.2450656     0.5910475
## diabetes            0.5838917      0.9657743 0.6671002     0.8838351
## high_cholesterol    0.6254636      0.7469367 0.6867782     0.8235408
## no_exercise         0.7213266      0.8732492 0.7822419     0.9278511
##                     no_health_insurance  diabetes high_cholesterol
no_exercise
## arthritis                     0.1252281 0.5838917        0.6254636
0.7213266
## kidney_disease                0.7249104 0.9657743        0.7469367
0.8732492
## copd                          0.2450656 0.6671002        0.6867782
0.7822419
## heart_disease                 0.5910475 0.8838351        0.8235408
0.9278511
## no_health_insurance           1.0000000 0.7441073        0.6748323
0.6885669
## diabetes                      0.7441073 1.0000000        0.7574795
0.8775762
## high_cholesterol              0.6748323 0.7574795        1.0000000
0.8461454
## no_exercise                   0.6885669 0.8775762        0.8461454
1.0000000

cat ("Given the high correlations between several of the variables (ie.
arthritis is highly correlated with COPD and heart disease), it is not
surprising that PCA was successful in reducing the dimensionality of this
dataset.")

## Given the high correlations between several of the variables (ie.
arthritis is highly correlated with COPD and heart disease), it is not
surprising that PCA was successful in reducing the dimensionality of this
dataset.
```

Focusing on the first 2 principal components found for the cities.csv dataset, I produced the biplot for the first 2 principal components and interpreted it.

```
library(ggplot2)
library(ggrepel)
```

```
biplot_data <- as.data.frame(pca_result$x[, 1:2])
names(biplot_data) <- c("PC1", "PC2")

#Data frame for variable vectors
loadings <- pca_result$rotation[, 1:2]
loadings_df <- as.data.frame(loadings)
loadings_df$variable <- rownames(loadings)

#Plot
p <- ggplot() +
  geom_point(data = biplot_data, aes(x = PC1, y = PC2), linewidth = 1, color
= "blue") +
  geom_text_repel(data = biplot_data, aes(x = PC1, y = PC2, label =
rownames(biplot_data)),
                  size = 2.5, alpha = 0.8) +
  geom_segment(data = loadings_df, aes(x = 0, y = 0, xend = PC1, yend = PC2),
arrow = arrow(length = unit(0.15, "inches")),
               color = "red", linewidth = 0.5) +
  geom_text_repel(data = loadings_df, aes(x = PC1, y = PC2, label =
variable),
                  size = 2.5, alpha = 0.8, color = "red") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "none",
        plot.margin = unit(c(1, 1, 1, 1), "lines"),
        panel.background = element_rect(fill = "grey95"),
        axis.text = element_text(size = 12),  # Adjust axis text size
        axis.title = element_text(size = 14)) +  # Adjust axis title size
  labs(title = "Biplot of PCA", x = "PC1", y = "PC2")

## Warning in geom_point(data = biplot_data, aes(x = PC1, y = PC2), linewidth
= 1,
## : Ignoring unknown parameters: `linewidth`

#Size of plot
ggsave("biplot_improved.png", plot = p, width = 24, height = 16, dpi = 300)

print(p)

## Warning: ggrepel: 38 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```



```
cat ("Each point represents an individual city. Their placement on the plot
shows their scores on the principal components. Cities that are close
together on the plot have similar profiles in terms of the health variables
considered. The horizontal line (PC1) captures the most common trend, and the
vertical line (PC2) shows the second most common trend. The farther a city's
dot is from the center along these lines, the more it is characterized by
those trends. An example is the high correlation between arthritis and COPD
shown with long arrows pointing in similar directions as eachother.")
```

```
## Each point represents an individual city. Their placement on the plot
shows their scores on the principal components. Cities that are close
together on the plot have similar profiles in terms of the health variables
considered. The horizontal line (PC1) captures the most common trend, and the
vertical line (PC2) shows the second most common trend. The farther a city's
dot is from the center along these lines, the more it is characterized by
those trends. An example is the high correlation between arthritis and COPD
shown with long arrows pointing in similar directions as eachother.
```

In the last month, the Product organization of this web company ran 100 experiments to evaluate ideas to improve the User Experience (UX) of its customers. In each experiment, a Product Engineering team were responsible to enable a different UX for a randomly selected group of users. For instance, randomly selected users could see different colors for some of the navigation buttons, different positioning of the search bar on the page, modified text for different components of the page, etc. At the end of each experiment, the Product Manager in charge of the experiment used a tool to compute the p-value for the one-sided t-test associated with following statistical hypothesis test: (H0 : user engagement is not higher with the new user experience H1 : user engagement higher with the new user experience. Here is the loeaded experiments.csv file containing the p-values of the 100 experiments that were run in the last month.

```
library(ISLR2)
library(dplyr)
library(survival)
library(survminer)
library(ggplot2)
library(readr)
library(stats)

experiments_data <-
read.csv("https://raw.githubusercontent.com/sydneymcolumbia/CMU/main/experime
nts.csv")

str(experiments_data)

## 'data.frame':    100 obs. of  2 variables:
##  $ experiment: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ p         : num  0.0075 0.0418 0.0431 0.2733 0.2355 ...

head(experiments_data)

##   experiment      p
## 1          1 0.0075
## 2          2 0.0418
## 3          3 0.0431
## 4          4 0.2733
## 5          5 0.2355
## 6          6 0.4834
```

The Product organization of the web company has an internal policy by which the default significance level that should be used when evaluating the results of UX experiments for the company's website is α = 0.10. I looked at how many experiments were found to generate a statistically significant UX improvement at the α = 0.10 level over the last month.

```
significant_experiments <- sum(experiments_data$p < 0.10)

print(significant_experiments)

## [1] 47

print(paste("This means that", significant_experiments, "experiments showed a
statistically significant improvement in user experience according to the
company's internal policy for significance level."))
```

```
## [1] "This means that 47 experiments showed a statistically significant
improvement in user experience according to the company's internal policy for
significance level."
```

The Family-Wise Error Rate (FWER) across 100 statistical tests - each carried out at the α = 0.10 significance level - is much larger than 0.10. Assuming that these statistical tests were independent, I found the effective FWER that the Product team incurred into by not accounting for the problem of multiple testing.

```
alpha = 0.10
n = 100

fwer = 1 - (1 - alpha)**n

print(paste("The FWER is:" , fwer, ". This high value indicates a very high
probability of making at least one false positive error across the 100 tests
when each test is conducted at a 10% significance level."))
```

```
## [1] "The FWER is: 0.999973438601112 . This high value indicates a very
high probability of making at least one false positive error across the 100
tests when each test is conducted at a 10% significance level."
```

Using the Benjamini-Hochberg method to account for the problem of multiple testing, I provided the list of experiment IDs that likely resulted in an improvement of the user experience. I controlled the False Discovery Rate (FDR) at the level q = 0.10. I performed different types of multiple hypothesis tests, including the Benjamini-Hochberg method.

```
adjusted_p_values <- p.adjust(experiments_data$p, method = "BH", n =
length(experiments_data$p))

significant_experiments <- which(adjusted_p_values < 0.10)

#Convert IDs to comma-separated string
ids_string <- paste(experiments_data$experiment[significant_experiments],
collapse = ", ")

IDs <- paste("The experiments with the IDs", ids_string, "were identified as
having p-values low enough to be considered statistically significant.")

print(IDs)
```

```
## [1] "The experiments with the IDs 1, 10, 12, 27, 29, 33, 34, 38, 58, 66,
67, 68, 76, 78, 82, 83 were identified as having p-values low enough to be
considered statistically significant."
```