

# ALY 6070: COMMUNICATION AND VISUALIZATION FOR DATA ANALYTICS

Assignment 1: Cryptocurrency and Valuations: Bitcoin

Submitted to

Prof. Fatemeh Ahmadi Abkenari

Submitted by

Abhilash Dikshit

Milan Prajapati

Shamim Sherafati

Smit Parmar

## Assignment 1: Cryptocurrency Prices and Valuations: Bitcoin

Abhilash Dikshit, Milan Prajapati, Shamim Sherafati, Smit Parmar

*College of Professional Studies Northeastern University Vancouver, Canada*

## Abstract:

This study aims to use Principal Component Analysis (PCA) in R to perform feature reduction and determine the optimal number of components for a given dataset. The dataset used in this study is the Bitcoin cryptocurrency network data from Coin metrics, which contains many variables related to the Bitcoin network. PCA is used to transform the original variables into a smaller set of uncorrelated variables, which can provide insights into the underlying structure of the data and simplify subsequent analyses. The optimal number of components is determined by visualizing the variance explained by each component and selecting the number of components that captures enough variance. The results of this study provide a practical example of how PCA can be used in R to perform feature reduction and improve the efficiency and interpretability of subsequent analyses in large datasets.

## Introduction

Principal Component Analysis (PCA) is a commonly used technique in data analysis and machine learning for dimensionality reduction and feature extraction. PCA is particularly useful in cases where there are many variables in a dataset, and the objective is to identify a smaller set of variables that explain most of the variance in the data.

The study begins by introducing the concept of PCA and its usefulness in feature reduction. We then provide an overview of the dataset used in this study and describe the variables in the dataset. Next, we demonstrate how to implement PCA in R and visualize the results to decide on the optimal number of components. Finally, we interpret the results of the PCA and discuss the implications for future analysis. The study concludes by highlighting the potential applications of PCA in data analysis and its usefulness in simplifying complex datasets.

### Part A: Dataset Description

We have 2991 rows and 10 columns, and we are considering 10 columns for our data analysis i.e., Date, High, Low, Open, Close, Volume, and Market cap. As there are no NA values, data cleanup is not preformed. The column “Date” class type was changed from char to Date. The dataset headTail() function values are shown below. We have also shown a summary of the dataset to get a better understanding of the variables in our dataset.

Realtime Dataset has been taken from [Coinmetrics.io](https://coinmetrics.io/community-network-data/) website.

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

Table

Description automatically generated

Text

Description automatically generated with medium confidence

### Part B: Variable Description

|  |  |
| --- | --- |
| Name | Crypto Name |
| Symbol | Crypto Symbol |
| Date | date of observation |
| High | Highest price on the given day |
| Low | Lowest price on the given day |
| Open | Opening price on the given day |
| Close | Closing price on the given day |
| Volume | Volume of transactions on the given day |
| Marketcap | Market capitalization in USD |

The following are the variables in our dataset

## Explore Data features with different types of graphs in R.

## Now, let’s create different types of graphs to explore the data features:

## Chart Description automatically generated

## Histogram - to show the distribution of Bitcoin closing prices:

## Graphical user interface, application, table Description automatically generated

## Box plot - to show the distribution of Bitcoin closing prices by year:

## Graphical user interface, chart Description automatically generated

## Scatter plot - to show the relationship between Bitcoin closing prices and trading volume:

## Chart Description automatically generated

## Bar chart - to show the total trading volume of Bitcoin by year:

## Chart, histogram Description automatically generated

## Run a PCA approach in R to do a feature reduction.

## Let’s run a Principal Component Analysis (PCA) approach in R to do a feature reduction on the cryptocurrency price history dataset. First, let’s load the dataset and prepare it for PCA. We will use the dataset, and select only the Open, High, Low, Close, Volume, and Market cap columns:

## Graphical user interface, text, application, email Description automatically generated

## Next, we will scale the data to have zero mean and unit variance:

## 

## Now, we can run the PCA using the prcomp() function:

## 

## We can now explore the results of the PCA, starting with the summary of the PCA object:

## Calendar Description automatically generated

## This shows us that the first principal component (PC1) explains 66.33% of the variance in the data, while the second principal component (PC2) explains 14.52% of the variance, and so on.

## We can also plot a scree plot to visualize the proportion of variance explained by each principal component:

## A picture containing chart Description automatically generated

## Decide on the best number of components based on the PCA visualization.

## Finally, we can extract the principal components using the predict() function:

## We can then use these principal components for further analysis or modeling.

## 

## Plot the cumulative variance plot.

## Graphical user interface Description automatically generated

## References

* Silge, M. K. A. J. n.d. 16 Dimensionality Reduction | Tidy Modeling with r. <https://www.tmwr.org/dimensionality.html>.
* Wood, R. 2021, December 14. Learn Principal Component Analysis in r - Towards Data Science. <https://towardsdatascience.com/learn-principle-component-analysis-in-r-ddba7c9b1064>.

## Appendix

my\_packages = c("plyr", "plotly", "ggplot2", "psych", "tidyr", "tidyverse","dplyr","lubridate","readr","caret")

*#install.packages(my\_packages)*

lapply(my\_packages, **require**, character.only = T)

*#Dataset used from https://coinmetrics.io/community-network-data/*

bitcoin <- read.csv("/Users/abidikshit/R\_Projects/Data/coin\_Bitcoin.csv", header = T)

cat("Number of Rows before cleanup:", nrow(bitcoin), "\n") *# Printing string and variable row count on the same line*

cat("Number of Columns before cleanup:", ncol(bitcoin), "\n")

cat("Blank cells count before cleanup:", sum(!complete.cases(bitcoin))) *# Displaying Blank Cells Count for uncleaned data*

bitcoin$Date <- as.Date(bitcoin$Date)

headTail(bitcoin, top = 4, bottom = 4, ellipsis = F)

summary(bitcoin)

ggplot(bitcoin, aes(x = Date, y = Close)) + geom\_line() + xlab("Date") + ylab("Close Price (USD)") + ggtitle("Bitcoin Price Trend Over Time")

ggplot(bitcoin, aes(x = Close)) + geom\_histogram(binwidth = 50) + xlab("Close Price (USD)") + ylab("Frequency") + ggtitle("Histogram of Bitcoin Closing Prices")

bitcoin$Year <- year(bitcoin$Date)

ggplot(bitcoin, aes(x = Year, y = Close)) + geom\_boxplot() + xlab("Year") + ylab("Close Price (USD)") + ggtitle("Box Plot of Bitcoin Closing Prices by Year")

ggplot(bitcoin, aes(x = Close, y = Volume, colour = Date)) +

geom\_point() +

xlab("Close Price (USD)") +

ylab("Trading Volume") +

ggtitle("Scatter Plot of Bitcoin Closing Prices and Trading Volume") +

scale\_color\_viridis\_c() +

guides(colour = guide\_legend(title = "Date")) +

geom\_smooth(method="lm", se=FALSE, color="blue")

total\_volume <- aggregate(Volume ~ Year, data = bitcoin, sum)

mean\_volume <- mean(total\_volume$Volume)

ggplot(total\_volume, aes(x = reorder(Year, -Volume), y = Volume)) +

geom\_bar(stat = "identity", fill = "steelblue") +

xlab("Year") +

ylab("Total Trading Volume") +

ggtitle("Total Trading Volume of Bitcoin by Year") +

geom\_text(aes(label = format(Volume, big.mark = ",")), vjust = -0.5) +

geom\_hline(yintercept = mean\_volume, linetype = "dashed", color = "coral") +

annotate("text", x = Inf, y = mean\_volume, vjust = -1, hjust = 1, label = paste0("Mean: $", format(mean\_volume, big.mark = ","))) +

geom\_text(aes(label = "Mean Volume"), hjust = 1.5, color = "red", size = 3) +

theme\_classic()

bitcoin\_pca <- read\_csv("/Users/abidikshit/R\_Projects/Data/coin\_Bitcoin.csv", col\_types = cols\_only(Date = col\_date(), Open = col\_double(), High = col\_double(), Low = col\_double(), Close = col\_double(), Volume = col\_double(), Marketcap = col\_double()))

bitcoin\_pca <- bitcoin\_pca[,c(2:7)]

scaled\_bitcoin <- scale(bitcoin\_pca)

pca\_bitcoin <- prcomp(scaled\_bitcoin, scale = TRUE)

summary(pca\_bitcoin)

scree\_plot <- ggplot(data.frame(PC = 1:6, Variance = pca\_bitcoin$sdev^2 / sum(pca\_bitcoin$sdev^2)), aes(x = PC, y = Variance)) + geom\_bar(stat = "identity", fill = "steelblue") + geom\_line(aes(x = PC, y = cumsum(Variance)), color = "red") + xlab("Principal Component") + ylab("Proportion of Variance") + ggtitle("Scree Plot of Bitcoin Price History Dataset")

scree\_plot

pcs\_bitcoin <- predict(pca\_bitcoin, scaled\_bitcoin)

plot(cumsum(pca\_bitcoin$sdev^2 / sum(pca\_bitcoin$sdev^2)), xlab = "Number of components", ylab = "Cumulative variance")

*# Choose the number of components*

n\_components <- length(which(cumsum(pca\_bitcoin$sdev^2 / sum(pca\_bitcoin$sdev^2)) < 0.8)) + 1

cat("Number of components:", n\_components, "\n")

df\_pca\_components <- predict(pca\_bitcoin, scaled\_bitcoin) %>%

as.data.frame() %>%

rename\_all(~ paste0("PC", .))

*#df\_pca\_components$Date <- df$Date*

headTail(df\_pca\_components,top = 3, bottom = 3, ellipsis = 0)

*## NA*