Analytics using Python

Learning outcomes

1. You will learn Python , a useful language

2. Use programming for problem solving

Great Lakes Institute of Management

A guide to learn python for analytics

P. V. Subramanian

**A workbook on Analytics using Python**

**P. V. SUBRAMANIAN**

**Visiting faculty – Great Lakes Institute of Management**

*Copyright @ 2019 Great Lakes Institute of Management : All contents in this book, including (but not limited to) its contents are proprietary to Great Lakes Institute of Management, Chennai. Any effort at re-using, modifying and circulating both through Digital or Physical means, without the express written permission of the author, will amount to infringement of copyrights, and will be liable for suitable action.*

**Chapter 6. Data Mining basics**

Table of Contents

[1) **Data Mining** 5](#_Toc9054219)

[**1.1.** **Data mining techniques** 6](#_Toc9054220)

[**1.1.1** **Association rules** 6](#_Toc9054221)

[**1.1.2** **Classification** 6](#_Toc9054222)

[**1.1.3** **Regression** 6](#_Toc9054223)

[**1.1.4** **Clustering** 6](#_Toc9054224)

[**1.1.5** **Prediction** 7](#_Toc9054225)

[2) **Machine Learning** 7](#_Toc9054226)

[**2.1.** **Types of machine learning** 8](#_Toc9054230)

[**2.1.1** **Supervised Learning** 8](#_Toc9054231)

[**2.1.2** **unSupervised Learning** 8](#_Toc9054232)

[**2.1.3** **reinforcement Learning** 8](#_Toc9054233)

[3) **MARKET BASKET ANALYSIS** 9](#_Toc9054234)

[**3.1.** **A few definitions** 9](#_Toc9054237)

[**3.2.** **Example MBA FOR ONLINE RETAIL DATA** 10](#_Toc9054238)

[4) **CLUSTERING** 16](#_Toc9054239)

[**4.1.** **common algorithms** 16](#_Toc9054244)

[**4.1.1** **Hierarchical clustering** 17](#_Toc9054247)

[4.1.2 **K-means clustering** 17](#_Toc9054248)

[**4.1.3** **K-modes and K-prototypes** 17](#_Toc9054249)

[**4.2.** **Linkage Methods** 18](#_Toc9054250)

[**4.3.** **Three distance metrics** 18](#_Toc9054253)

[**4.3.1** **Euclidean distance or l2** 18](#_Toc9054254)

[**4.3.2** **Manhattan or l1** 19](#_Toc9054255)

[**4.3.3** **Cosine** 19](#_Toc9054256)

[**4.4.** **hierarchical clustering** 19](#_Toc9054257)

[**4.4.1** **Steps in cluster analysis:** 20](#_Toc9054259)

[**4.5.** **K Means clustering** 25](#_Toc9054260)

[5) **PRINCIPAL COMPONENT ANALYSIS** 29](#_Toc9054261)

[**5.1.** **Introduction** 29](#_Toc9054267)

[**5.2.** **Example 3 –cereals data** 30](#_Toc9054268)

# **Data Mining**

* *Data mining is defined as extracting information that are hidden, valid and possibly useful patterns from huge sets of data. Data mining is also known as Knowledge discovery, Information harvesting etc.*
* *The insights derived from Data mining can be used for many applications including marketing analysis, fraud detection, customer retention etc.*

**Data Mining Implementation Process**

*For more details, refer:* [*https://www.guru99.com/data-mining-tutorial.html*](https://www.guru99.com/data-mining-tutorial.html)

## **Data mining techniques**

Some of the important, often used techniques are discussed here.

### **Association rules**

This technique finds the relationship between at least two products. It detects the hidden patterns in the data based on a relationship between at least two products that occur most of the times.

An example for association rule in the supermarket domain:

{ butter , bread } ⇒ { milk } meaning that if butter and bread are bought, customers also buy milk.

### **Classification**

Classification is a technique used grouping of related data into classes according to their known characteristics.

Example: How Gmail classifies the new emails into spam or not spam based on different attributes of the email.

### **Regression**

Regression analysis is a technique used for predicting the unknown value of a continuous numerical variable, called dependent variable from the known values of one or more variables, called independent variables.

An example: Predict a number of insurance claims on the past knowledge of the values of the predictor variables such as age, salary, and car location. Car location is the only categorical variable that can take on two possible values, carpark and street can be converted to a numerical value 0 (for car park) or 1 (street).

### **Clustering**

Clustering is a technique used for dividing data into groups in such a way that data in the same group (called a cluster) are more similar to each other than to those in other groups (clusters).

An example: to unfold customer segments for marketing points.

### **Prediction**

It is concerned with the discovery of information patterns in data resulting in reasonably good predictions about the future. An example: A Sales manager wants to predict how much a loyal customer will spend during a summer sale campaign.

Reference:

<https://www.tutorialride.com/data-mining/data-mining-tutorial.htm>

<https://acadgild.com/blog/5-best-data-mining-techniques>

<http://www.zentut.com/data-mining/data-mining-techniques/>

<https://www.guru99.com/data-mining-tutorial.html>

# **Machine Learning**

The name machine learning was coined in 1959 by Arthur Samuel.

Tom M. Mitchell provided a widely quoted, more formal definition of the algorithms studied in the machine learning field: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

Data mining is the process of extracting hidden patterns in large data sets involving methods such as machine learning, statistics, and database systems.

One major difference between machine learning and data mining is how they are used and applied in practice.

Data mining is often used by machine learning to see the connections between relationships. An example: Uber uses machine learning to calculate Estimated Arrival Time for rides.

Machine learning embodies the principles of data mining but can also infer automatic correlations and learn from them to apply to new algorithms. An Example: Self-driving cars that can quickly adjust to new driving conditions while driving.

References:

1. <https://en.wikipedia.org/wiki/Machine_learning>
2. https://en.wikipedia.org/wiki/Data\_mining
3. Samuel, Arthur (1959). "Some Studies in Machine Learning Using the Game of Checkers". IBM Journal of Research and Development. 3 (3): 210–229. CiteSeerX 10.1.1.368.2254. doi:10.1147/rd.33.0210. https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.368.2254
4. Mitchell, T. (1997). Machine Learning. McGraw Hill. p. 2. ISBN 978-0-07-042807-2.

https://www.import.io/post/data-mining-machine-learning-difference/

1. <https://en.wikipedia.org/wiki/Special:BookSources/978-0-07-042807-2>

## **Types of machine learning**

### **Supervised Learning**

In supervised learning, we have examples of correct input-output pairs that can be used to training the machine during the model building phase.

Example: Handwriting recognition

Supervised machine learning tasks can be broadly classified into two subgroups: regression and classification.

### **unSupervised Learning**

In unsupervised learning, find patterns of cluster membership where we do not have examples of correct input-output pairs that can be used to training the machine during the model building phase.

Unsupervised machine learning tasks can be broadly classified into two subgroups: Clustering and association.

### **reinforcement Learning**

Reinforcement learning technique allows software agents to automatically determine the ideal behavior within a specific context, to maximize its performance by observing the results.

# **MARKET BASKET ANALYSIS**

Market Basket Analysis uses association rule mining on the data to predict the items that are bought together.

Association mining is frequently done on transactions in the retail domain such as retail market or e-commerce store. The apriori algorithm finds the patterns or rules in the huge data quickly.



## **A few definitions**

**Rule**

A rule is notation that represents which item(s) is frequently bought with what items and has an Left Hand Side (LHS) part and an Right Hand Side (RHS) part.

Example: {Milk} -> {Bread, Butter}

This implies the items on the right (Bread and Butter) were frequently purchased along with the items on the left (Milk).

**How do you measure the strength of a rule?**

The apriori algorithm generates the most relevant set of rules from a given transaction data set.

It shows support, confidence and lift of those rules to decide the relative strength of the rules.

Consider the rule {X} => {Y} and compute the following metrics:

**Support** = Number of transactions with both X and Y / Total number of transactions = P(X∩Y)

It is the fraction of the transactions in the data set that contain product, X.

**Confidence** = Number of transactions with both X and Y / Total number of transactions with X =

P(X∩Y) / P(X)

Confidence of the rule is 60% implies that if a customer buys product X, he / she is 60% likely to buy product Y too.

*A caveat: Confidence does not measure if the association between X and Y is random or not.*

**Expected confidence** = Number of transactions with Y / Total number of transactions = P(Y)

**Lift** = Confidence / Expected Confidence = P(X∩Y) / P(X). P(Y)

Lift is the factor by which the co-occurrence of X and Y exceeds the expected probability of X and Y co-occurring, had they been independent. Higher the lift, higher the chance of X and Y occurring together.

**A lift value of 1.50 implies that the chance of buying the product Y would increase by 50%, whenever product X is purchased.**

**Reference:**

* *http://r-statistics.co/Association-Mining-With-R.html*
* *https://datascienceplus.com/a-gentle-introduction-on-market-basket-analysis%E2%80%8A-%E2%80%8Aassociation-rules/*

## **Example MBA FOR ONLINE RETAIL DATA**

For performing Market Basket Analysis, we shall use the transactional data set which contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail.

**Source:**

Dr Daqing Chen, Director: Public Analytics group. chend '@' lsbu.ac.uk, School of Engineering, London South Bank University, London SE1 0AA, UK.

For more details, refer to http://archive.ics.uci.edu/ml/datasets/online+retail#

<http://archive.ics.uci.edu/ml/machine-learning-databases/00352/Online%20Retail.xlsx>

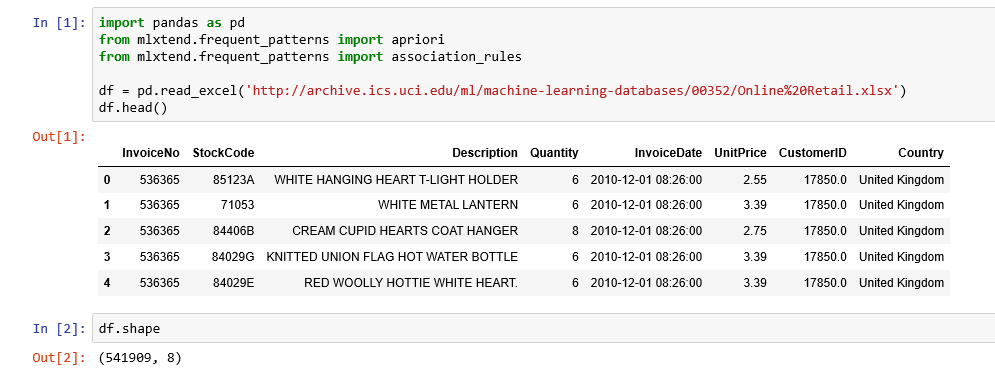
<https://pbpython.com/market-basket-analysis.html>

**Data dictionary**

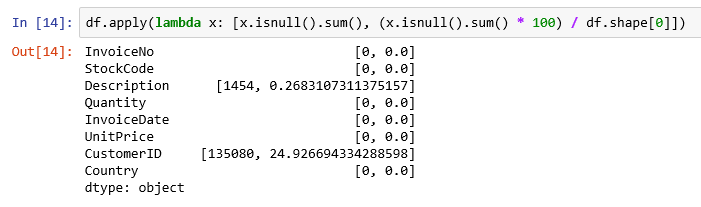
|  |  |
| --- | --- |
| InvoiceNo | Invoice number. Nominal, a 6-digit integral number uniquely assigned to each transaction. If this code starts with letter 'c', it indicates a cancellation |
| StockCode | Product (item) code. Nominal, a 5-digit integral number uniquely assigned to each distinct product. |
| Description | Product (item) name. Nominal. |
| Quantity | The quantities of each product (item) per transaction. Numeric |
| InvoiceDate | Invoice Date and time. Numeric, the day and time when each transaction was generated. |
| UnitPrice | Unit price. Numeric, Product price per unit in sterling. |
| CustomerID | Customer number. Nominal, a 5-digit integral number uniquely assigned to each customer. |
| Country | Country name. Nominal, the name of the country where each customer resides. |

**We shall use mlxtend for performing Market Basket Analysis.**

Load the required modules and the data set from the given URL.:.

  
:   
We observe that there are 541909 observations and 8 variables in the data frame.

Let us check if there are any missing values in the data.



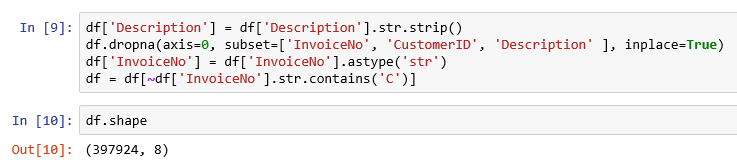
We observe that the columns, Description has 1454 missing values accounting to 0.26% of the total observations and also the column, CustomerID has 135080 missing values accounting to 24.9% of the missing values. These two columns are very important. It is very difficult to impute these variables to give a reasonable result. Let us drop all these missing values, row wise.

:   
We will also drop the rows that:

a) do have null values,

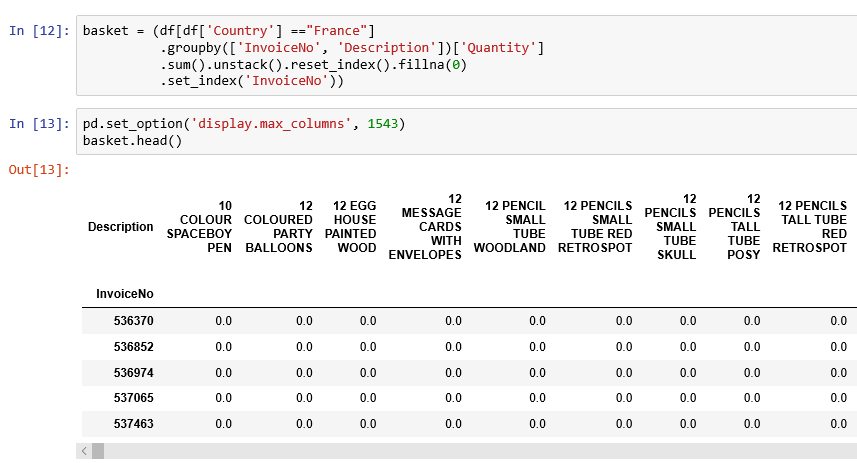
b) do not have invoice numbers and

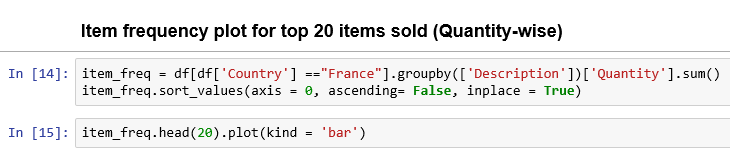
c) the credit transactions (those with invoice numbers containing C)

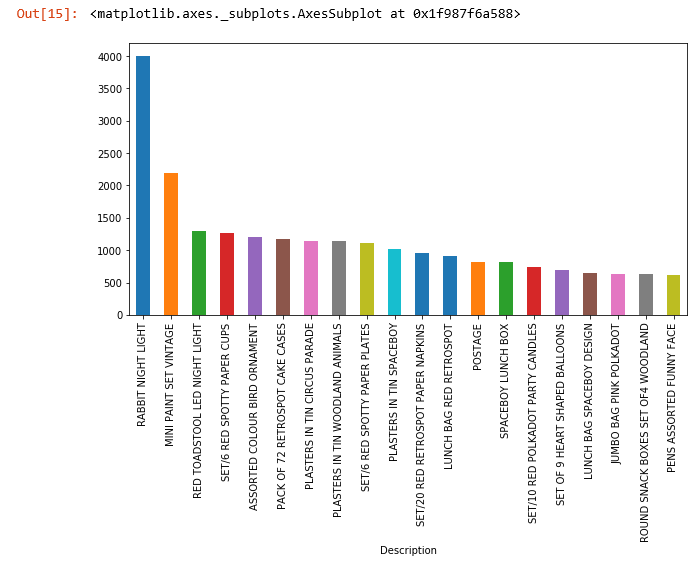


**Now we have 397924 observations and 8 columns in our data frame, df.**

**After the cleanup, consolidate the items into 1 transaction per row with each product 1 hot encoded. Consider sales for France.**

****

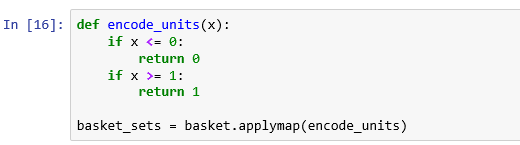
****

: 

We observe that the item, RABBIT NIGHT LIGHT is sold more than any other item.

We re interested in the pair of items that are sold together most.

There are a lot of zeros in the data, but we also need to make sure any positive values are converted to a 1 and anything less the 0 is set to 0. This step will complete the one hot encoding of the data

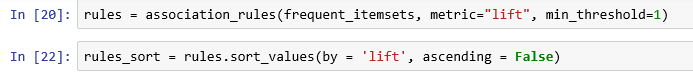


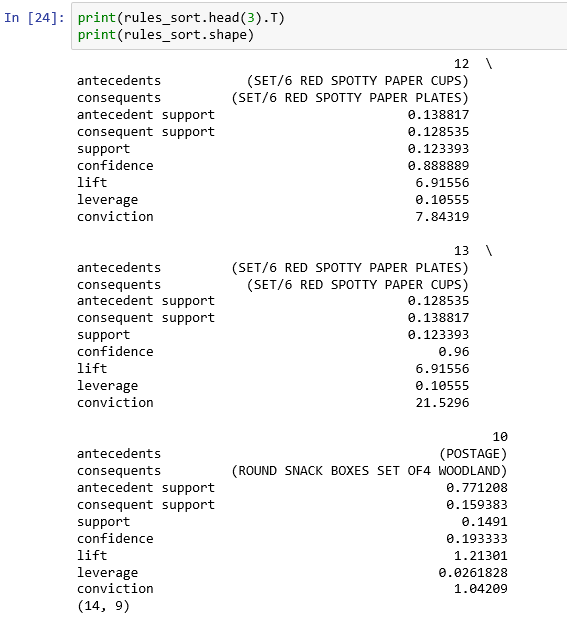
Apriori is a popular algorithm for extracting frequent item sets with applications in association rule learning. The apriori algorithm has been designed to operate on databases containing transactions, such as purchases by customers of a store. An itemset is considered as "frequent" if it meets a user-specified support threshold.

Generate frequent item sets that have a support of at least 12%.



The final step is to generate the rules with their corresponding support, confidence and lift:



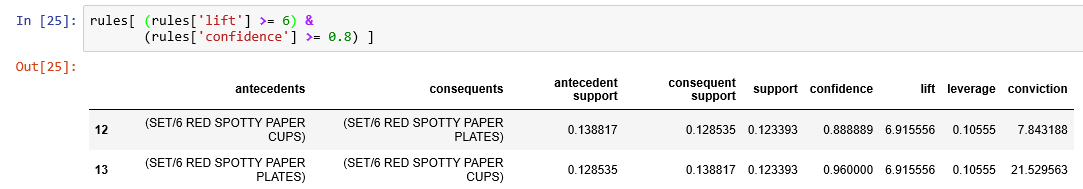


**There are 14 observations and 9 columns in the data frame, rule\_sort.**

**We observe the following:**

1. *There are quite a few rules with a high lift value which means that it occurs more frequently than would be expected given the number of transaction and product combinations.*
2. *We observe high confidence values for a few rules.*

**Now, check for a large lift (6) and high confidence (.8)**

****

*Looking at the rules, we observe that the red paper cups and plates are purchased together in a manner that is higher than the overall probability would suggest.*

# **CLUSTERING**

In Machine Learning, unsupervised learning is a class of problems in which one seeks to determine how the data are organized.

Clustering is a method of unsupervised learning for performing statistical analysis used in many fields. Clustering is about dividing the data set into subsets, or clusters, wherein the observations should be like those in the same cluster but differ greatly from the observations in other clusters.

<http://www.cad.zju.edu.cn/home/zhx/csmath/lib/exe/fetch.php?media=2011:presentation_ml_by_ibrar.pdf>



## **common algorithms**

1. Hierarchical clustering
2. K-mean clustering
3. K-mode clustering
4. K-prototype clustering

### **Hierarchical clustering**

Hierarchical (aka agglomerative) clustering tries to link each data point by a distance measure, to its nearest neighbor, creating a cluster. Reiterating the algorithm using different linkage methods, the algorithm gathers all the available points into a rapidly diminishing number of clusters, until in the end all the points reunite into a single group.

### **K-means clustering**

K-means clustering is an algorithm to group your objects based on attributes into k number of groups and k > 0. Grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid.

http://www.cad.zju.edu.cn/home/zhx/csmath/lib/exe/fetch.php?media=2011:presentation\_ml\_by\_ibrar.pdf

### **K-modes and K-prototypes**

**What if our data is… non-numerical?**

The basic foundation of k-means stands on mathematical calculations (means, Euclidian distances). When our data is non-numerical or, in other words, categorical, what do you do with K-means?

You may transform categorical values in numerical values by using dummy variables and then apply k-means. Caveat: k-means uses numerical distances, so it could consider close two distant objects that merely have been assigned two close numbers.

Think of k-modes algorithm presented in a paper of 1998 by Zhexue Huang.

k-modes is an extension of k-means. Instead of distances it uses dissimilarities (that is, quantification of the total mismatches between two objects: the smaller this number, the more similar the two objects). And instead of means, it uses modes. A mode is a vector of elements that minimizes the dissimilarities between the vector itself and each object of the data.

k-modes is used for clustering categorical variables. It defines clusters based on the number of matching categories between data points. The k-prototypes algorithm combines k-modes and k-means and is able to cluster mixed numerical / categorical data.

## **Linkage Methods**

The linkage criterion determines the distance between sets of observations as a function of the pairwise distances between observations.

For more details, refer: <https://www.dummies.com/programming/big-data/data-science/data-science-performing-hierarchical-clustering-with-python/>



1. **Ward**

This looks for spherical clusters, very cohesive inside and extremely differentiated from other groups. This finds clusters of similar size and works well with Euclidean distance.

1. **Complete**

This looks for most dissimilar data points and links clusters. Clusters created using this method comprises highly similar observations, making the clusters compact.

1. **Average**

This looks for centroids of data points, ignoring their boundaries and links clusters. These clusters are can be of different sizes and shapes making it attractive technique to be used in the field of biological sciences.

## **Three distance metrics**

### **Euclidean distance or l2**

The Euclidean distance between points a and b is the length of the line segment connecting them .



### **Manhattan or l1**

The Manhattan distance between points is calculate by adding the absolute value of the difference between the dimensions.



### **Cosine**

Cosine distance is squared Euclidean distance with the data normalized to unit length.

## **hierarchical clustering**

In hierarchical clustering, we don't use a set number of clusters but rather we arrange the data in a hierarchy where on top of the hierarchy there is a single big cluster and at the bottom of the hierarchy we have as many clusters as many observations in the data set.

**Two common methods of hierarchical clustering algorithms:**

**Agglomerative Hierarchical Clustering Algorithms**

In this approach, we initially assign different clusters to each observation. Based on similarity we consolidate until we arrive at one single big cluster.

**Divisive Hierarchical Clustering Algorithm**

In this approach, we initially assign a single cluster to all the observations. The clusters are divided continuously until we have one cluster for each observation.

We will use Agglomerative Clustering method.

**Example 1:**

Use the inbuilt iris data set and perform cluster analysis,

We shall use a Dendrogram to decide the number of clusters required for the dataset. A Dendrogram is a tree diagram illustrating the arrangement of clusters.

We will import the package AgglomerativeClustering for building an agglomerative clustering model. We shall import other required packages like pandas, matplotlib, numpy etc.

We shall import datasets for obtaining the iris dataset.

We shall import the package dendrogram which allows us to create dendrogram.



### **Steps in cluster analysis:**

1. **Select the right variables**

Select the variables which is considered important for identifying and understanding differences among groups of observation within the data. Since we want to cluster the iris flower attributes, we select all the attributes describing the dimension of the flower.

1. **Scale the data**

This needs to be done when the samples of data are drawn from different sources and grouped in different scales. In our case, iris flower attributes measured at the same time by the same person with the same apparatus. There is no need to scale the data.

1. **Calculate distances and link the clusters**

Our objective is to group similar points together in one cluster.

The following agglomerative algorithms define similarity between clusters in different ways:

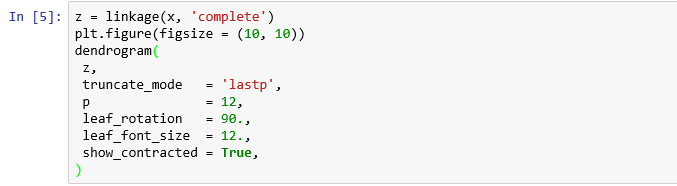
1. Single Linkage (aka nearest neighbor method) defines similarity between clusters as the shortest distance from any point in one cluster to any point in another
2. Complete Linkage (aka farthest neighbor method) defines similarity between clusters as the maximum distance from any point in one cluster to any point in another
3. Average Linkage defines similarity between clusters as the average distance between all pairs of the two clusters' members
4. Centroid Method defines the distance between the two clusters as the distance between to centroids. Cluster centroids are the mean values of the observation on the variables of the cluster.
5. Ward's Method defines similarity between clusters as the sum of squares within the clusters summed over all the variables.
6. Median linkage method defines the distance between two clusters is the median distance between an observation in one cluster and an observation in the other cluster.
7. **Choose the right number of clusters**

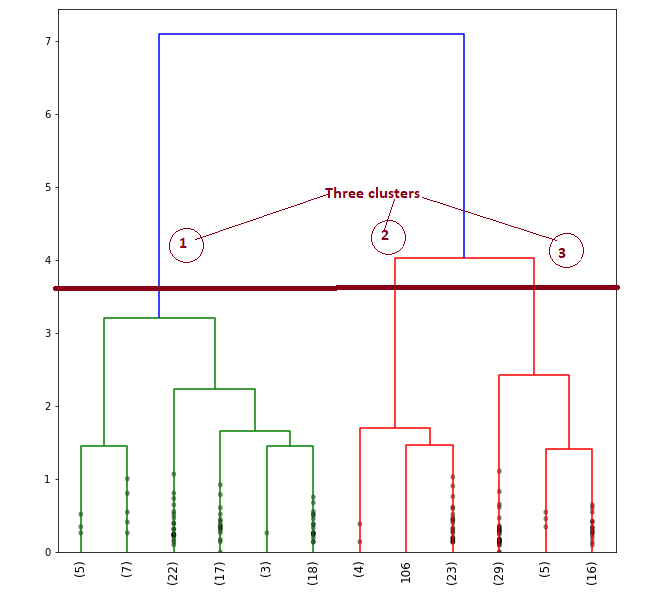
We shall load the following modules in addition to pandas, numpy in python:

* 1. scipy module to draw a dendrogram
  2. sklearn.datasets to get the iris data set
  3. sklearn.cluster module for Hierarchical clustering model building
  4. matplotlib.pyplot for drawing graphs



**Draw a dendrogram to identify the number of clusters.**





1. Dendrogram visually displays a cluster configuration. Rows that are close together (have small dissimilarity) will be linked near the right side of the plot. To determine the number of clusters, draw a horizontal line at that value and counting the number of lines that the horizontal line intersects.
2. At three points the horizontal line intersects. So the number of clusters identified is 3.

**Parameters:**

1. **z** is the linkage matrix encoding the hierarchical clustering to render as a dendrogram. See the linkage function for more information on the format of z.
2. **truncate\_mode** (optional): This is used for condensing the dendrogram to make it easy to read

**Several modes**:

1. None: No truncation is performed

2. **lastp**: The last p non-singleton clusters formed in the linkage are the only non-leaf nodes in the linkage and others are contracted into leaf nodes.

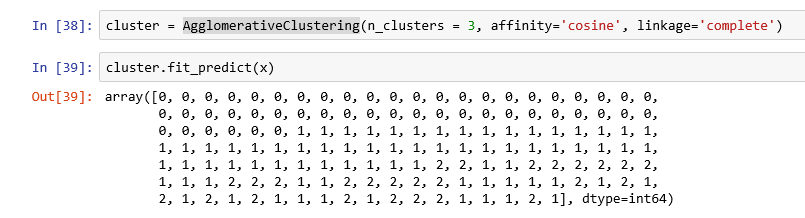
1. **p** parameter, an integer (optional) for the truncate mode
2. **leaf\_rotation** (optional): Specifies the angle in degrees to rotate the leaf labels.
3. **leaf\_font\_size** (optional): Specifies the font size (in points) of the leaf labels,
4. **show\_contracted** (optional): When True, the heights of the non-singleton nodes contracted into a leaf node are plotted as crosses along the link connecting that leaf node.

For more details, please refer to:

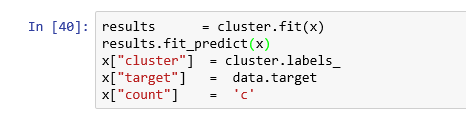
[*https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.dendrogram.html*](https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.dendrogram.html)

1. **Apply AgglomerativeClustering method**

Recursively merge the pair of clusters that minimally increases a given linkage distance.



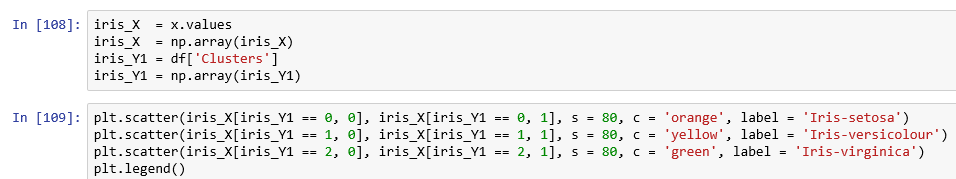
**Group the data points in clusters derived as above.**

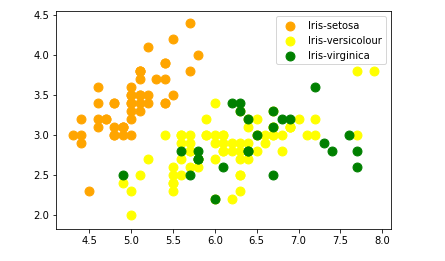


**Compare the clusters formed with the original target variable, Species of the iris data set.**



**Visual representation of the above**

****

****

**Target 0 represents Iris Setosa and is correctly predicted.**

**Target 1 represents Iris Versicolour and is correctly predicted**

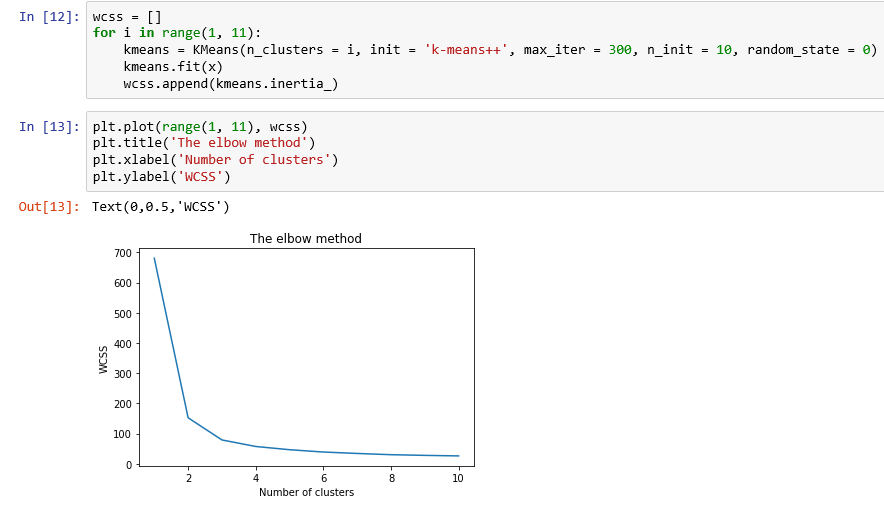
**Target 2 represents Iris Virginica and partially correctly predicted.**

## **K Means clustering**

* K-means is the most commonly used method for clustering. It requires us to define the number of clusters.
* Decide the number of clusters, k by performing Elbow analysis by running K-means for the value of k from 1 to 11, we find the optimum value of k. for i in range(1, 11):

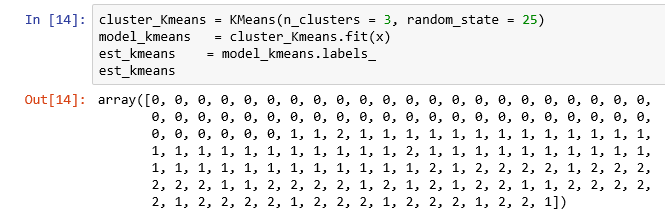
**Example 2:**

Using the same iris data set, perform K Means clustering analysis.



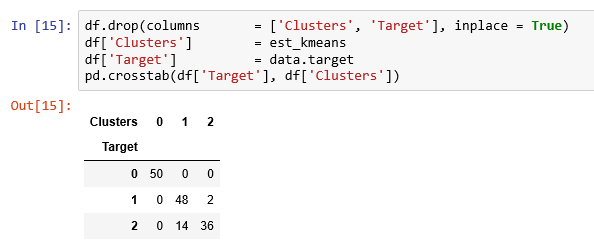
* Refer the above elbow graph. Observe the points where the drop falls, and the line smoothens out. So, k=3 because till k = 3, the decline is sharp.

**Apply Kmeans() method since we know the number of clusters to be used**

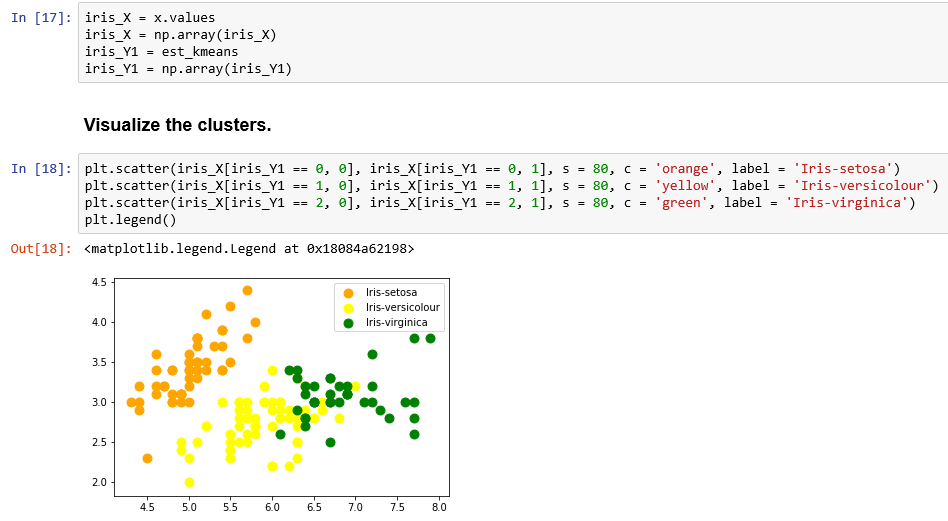
****

**Group the data points in clusters derived as above and compare the clusters formed with the original target variable, Species of the iris data set.**

*Note: Take care to remove the old clusters formed by hierarchical clustering method*

****

*The above table shows that we have classified correctly 134 observations (50 + 48 + 36) out of 150 observations.*



*We observe that Iris-setosa has been correctly formed into a separate well-defined cluster, but the other two classes are not formed well.*

# **PRINCIPAL COMPONENT ANALYSIS**



## **Introduction**

* PCA was invented in 1901 by Karl Pearson. Later it was independently developed and named by Harold Hotelling in the 1930s.

Ref: <https://en.wikipedia.org/wiki/Principal_component_analysis>

* Principal Component Analysis (PCA) is one of the most valuable result from Applied Linear Algebra.
* With minimal effort, PCA provides a method of reducing a complex multi-dimensional dataset to a lower dimension that carries maximum information.
* The main idea of PCA is to reduce the dimensionality of a data set consisting of many variables correlated with each other, either heavily or lightly, while retaining the variation present in the dataset up to the maximum extent. This is achieved by transforming the variable to a new set of variables, known as Principal Components (PC),
* PCs are orthogonal, ordered such that the retention of variation present in the original variables decreases as we progress. So, the first principal component retains maximum variation that was present in the original components.
* PCs are the underlying structure in the data. They are the directions where there is most variance, the directions where the data is most spread out.
* The main goal of a PCA analysis is to identify patterns in data.
* PCA aims to detect the correlation between variables. If a strong correlation between variables exists, the attempt to reduce the dimensionality only makes sense.
* PCA finds of maximum variance in high-dimensional data and project it onto a smaller dimensional sub-space while retaining most of the information.
* Eigen vectors and eigen values of a covariance matrix represent the core of PCA. The eigen vectors (PCA) determine the directions of the new feature (reduced) space and the eigen values determine their magnitude. Covariance matrix describes the variance of the data and the covariance among the variables.

## **Example 3 –cereals data**

**Data set**

As part of a study of consumer consideration of ready-to-eat cereals sponsored by Kellogg Australia, Roberts and Lattin (1991) surveyed consumers regarding their perceptions of their favorite brands of cereals. Each respondent was asked to evaluate three preferred brands on each of 25 different attributes. Respondents used a five point Likert scale to indicate the extent to which each brand possessed the given attribute.

For the purpose of this assignment, a subset of the data collected by Roberts and Lattin, reflecting the evaluations of the 12 most frequently cited cereal brands in the sample (in the original study, a total of 40 different brands were evaluated by 121 respondents, but the majority of brands were rated by only a small number of consumers). The 25 attributes and 12 brands are listed below:

Cereal Brand Attributes 1-12 Attributes 13-25 All Bran Filling Family Cerola Muesli Natural Calories Just Right Fibre Plain Kellogg’s corn falkes Sweet Crisp Komplete Easy Regular Nutrigrain Salt Sugar Purina Muesli Satisfying Fruit Rice Bubbles Energy Process Special K Fun Quality Sustain Kids Treat Vitabrit Soggy Boring Weetbix Economical Nutritious Health

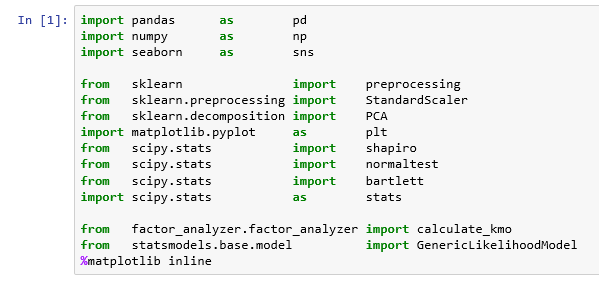
In total 116 respondents provided 235 observations of the 12 selected brands.

Refer: <https://rpubs.com/ssindw/274704>

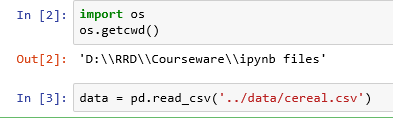
We use the PCA to find latent variables in a data set that describes the quality of cereals. We shall compare how well a set of latent variables works in predicting the quality of cereals against the original data set.

Refer <https://www.kindsonthegenius.com/2019/01/12/principal-components-analysispca-in-python-step-by-step/>

**Step 1 Import the necessary modules**

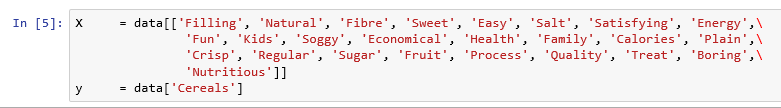
****

**Step 2: Obtain the Dataset**

****

* pd.read\_csv() is a function in pandas. The first argument is the path to the data.

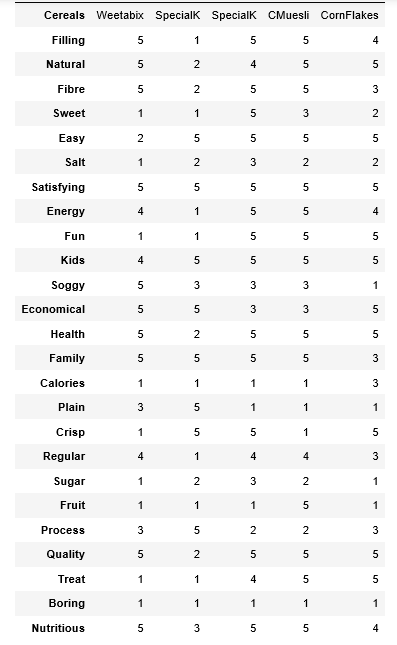




* In the first line, you create a data frame of independent variables by specifying the columns you need within double square bracket, [] In the second line, you create a series, y containing the dependent variable, Cereals.

**Step 3: Preview the data**

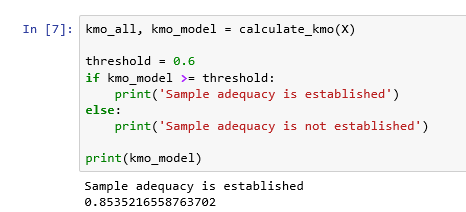
****

****

**Step 4 Check assumptions**

1. The variables used for PCA should **be measured at the continuous level**, although ordinal variables are very frequently used. Example of ordinal variable is measured in Likert scales (e.g., a 5-point scale from 'strongly agree' through to 'strongly disagree'). In our data set, all the variables are ordinal variables and measured in Likert scale.
2. Sample size: You should have sampling adequacy, which simply means that for PCA to produce a reliable result, large enough sample sizes are required.

ideally, there should be 150+ cases and there should be ratio of at least five cases for each variable (Pallant, 2010)

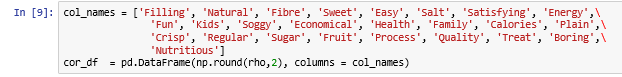


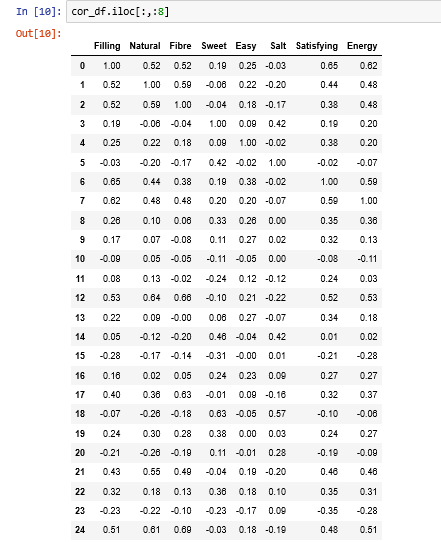
Kaiser-Meyer-Olkin (KMO) Test measures the suitability of data for PCA. It determines the adequacy for each observed variable and for the complete model. KMO estimates the proportion of variance among all the observed variable. KMO values range between 0 and 1. Value of KMO less than 0.6 is considered inadequate.

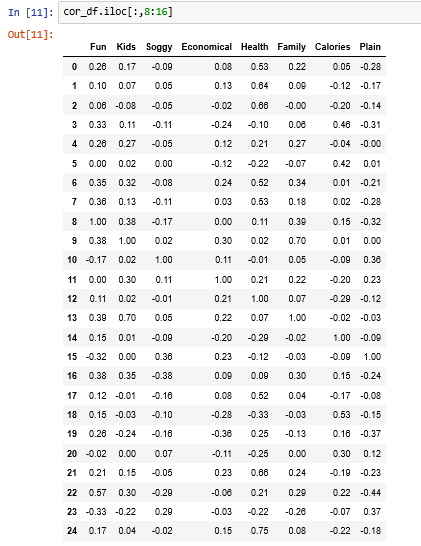
1. Correlations: there should be some correlation among the factors to be considered for PCA

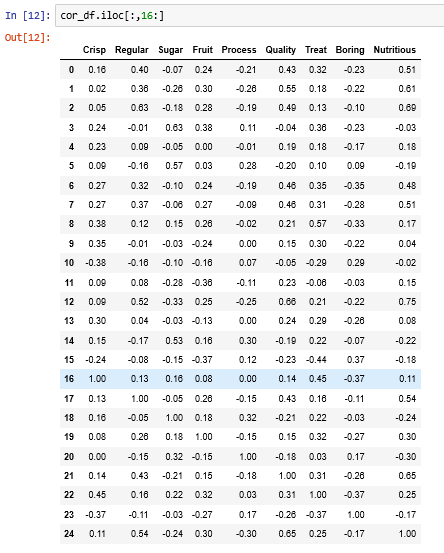
The Spearman Rank Correlation is a test of association for ordinal or interval variables.

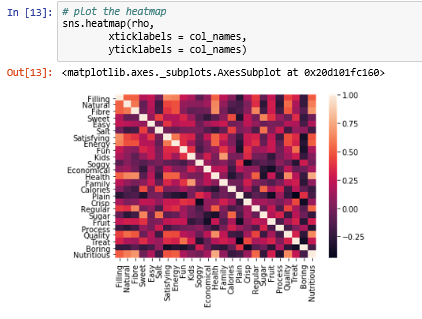






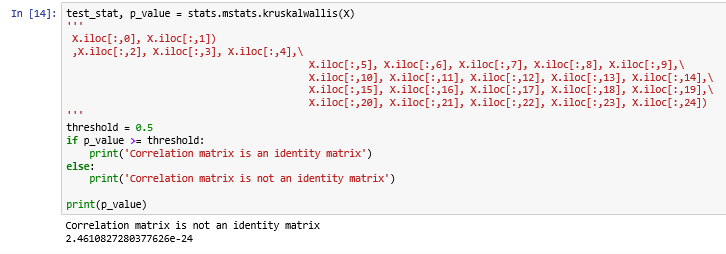






**From the above he Spearman Rank Correlation matrix we can see that all the variables value of correlation coefficient greater than 0.3 with at least one other variable. 17 variables have correlation coefficient of at least 0.5 and only 8 variables (Easy, Soggy, Economical, Plain, Crisp, Fruit, Treat, Boring) have at least 0.3. Hence, we can assume that variables are fairly correlated with each other and we can run Factor Analysis on this data.**

**Perform kruskalwallis’ test for equal variances:**

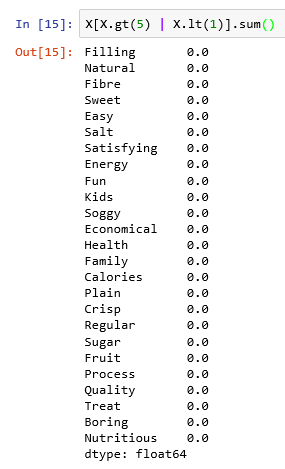
****

Kruskal–Wallis H test was conducted in Python and it was found to be significant

( P < 0.001) . The significance of this test tells us that the correlation matrix is not an identity matrix. Hence, we assume that there is some relationship between the variable.

Except for the variables, residual sugar, chlorides, sulphates and alcohol, other seven variables are moderately correlated with value above 0.50.

1. Linearity: It is not appropriate to determine the normality of ordinal data. So, we are not testing for normality assumption.
2. Outliers: Respondents have used a five point Likert scale. There are no chances for any outliers as shown below:



**Step 5 Scale the data**

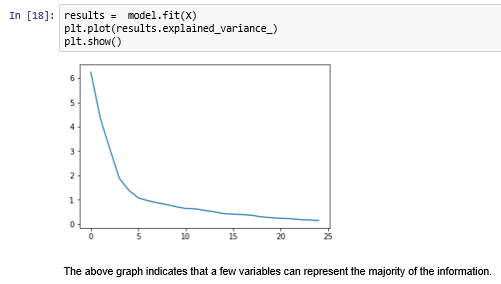
Since the variables in the data are ordinal, we don't have to scale the data.

**Step 6 Apply PCA**

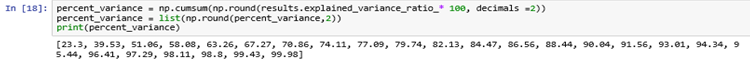
To perform PCA, we use PCA module from sklearn, we have imported in step 1.

****

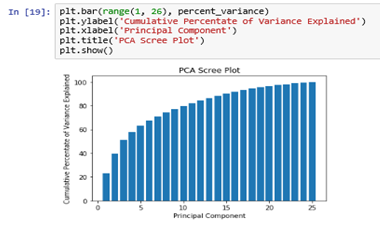
**a) Find the number of components needed**

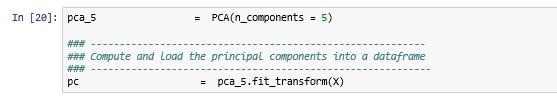
****

1. **b) Perform a Scree Plot of the Principal Components to visualize the cumulative % of variation explained by each component**

****

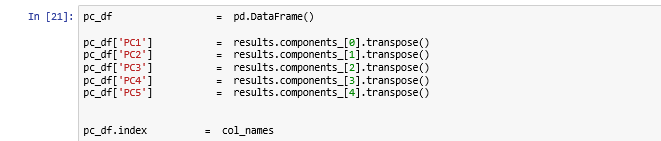
**Let us find out the cumulative percentage of variation of each component.**

1. 
2. **Note:** We observe that first 5 variables explain about 63% of the total information.
3. **c. Perform the PCA with the selected number of components**

****

1. **d. Form the matrix of variable loadings (i.e., a matrix whose columns contain the eigenvectors)**

**How does PCA calculate the 25 original variables' correlation with 11 latent variables?**

****

****

We have observed in Step 6 (b) that the first latent variable, PC1 captures 23.3% of the total information in the data set and the latent variables, PC1 and PC2 captures 39.53% of the total information and so on. Principal Components are linear combination of the 25 variables.

**For example: PC1 is calculated by using the following formula:**

PC1 = - (Filling \* 0.2478) - (Natural \* 0.2174) - (Fibre \* 0.2418) - ( Sweet \* 0.1278) - (Easy \* 0.1073) + (Salt \* 0.0363) - (Satisfying \* 0.2321) - (Energy \* 0.2505) - (Fun \* 0.3013) - (Kids \* 0.1530) + (Soggy \* 0.1050) - (Economical \* 0.0479) - (Health \* 0.2302) - (Family \* 0.1765) + ( Calories \* 0.0130) + (Plain \* 0.1938) - (Crisp \* 0.2314) - (Regular \* 0.2605) + (Sugar \* 0.0282) - (Fruit \* 0.1903) + (Process \* 0.1217) - (Quality \* 0.2446) - (Treat \* 0.3352) + (Boring \* 0.1856) - (Nutritious \* 0.2442)

1. **e. Component loading can be classified on their magnitude.**

|  |  |
| --- | --- |
| Value | Interpretation |
| > 30 | Minimum consideration level |
| > 40 | More important |
| > 50 | Practically significant |

**Variables that load highly on PC1:**

1. Fun
2. Treat

These are related to youth aspects of the Cereal, so we can label this component as Youth.

**Variables that load highly on PC2:**

1. Sweet
2. Sugar

These are related to taste aspects of the Cereal, so we can label this component as Tasty.

**Variables that load highly on PC3:**

1. Kids
2. Economical
3. Family
4. Fruit

These are related to the Family, so we can label this factor as For Family.

**Variables that load highly on PC4:**

Soggy

This is related to the moist aspect of the cereal, so we can label this factor as Over Moist.

**Variables that load highly on PC5:**

1. Crisp
2. Regular

These are related to the crunchy aspect of the cereal, so we can label this factor as Crunchy.