Assignment_5

Olayinka Sikiru

2024-03-25

```
# Setting the working directory
library(readr)
Cereals <- read_csv("C:/Users/DELL/OneDrive/Desktop/Fundamentals of Machine Learning/Dataset/Cer
eals.csv")
```

```
## Rows: 77 Columns: 16
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (3): name, mfr, type
## dbl (13): calories, protein, fat, sodium, fiber, carbo, sugars, potass, vita...
##
## 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.
```

Data Exploration

```
# To view first few rows of dataset
head(Cereals)
```

```
## # A tibble: 6 × 16
##   name       mfr   type  calories protein   fat sodium fiber carbo sugars potass
##   <chr>      <chr> <chr>    <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>  <dbl>  <dbl>
## 1 100%_Bran  N     C           70       4     1    130  10     5        6    280
## 2 100%_Natu… Q     C          120       3     5     15   2     8        8    135
## 3 All-Bran   K     C           70       4     1    260   9     7        5    320
## 4 All-Bran_… K     C           50       4     0    140  14     8        0    330
## 5 Almond_De… R     C          110       2     2    200   1    14        8     NA
## 6 Apple_Cin… G     C          110       2     2    180   1.5  10.5     10     70
## # ℹ 5 more variables: vitamins <dbl>, shelf <dbl>, weight <dbl>, cups <dbl>,
## #   rating <dbl>
```

```
# Checking the summary and statistics of dataset
summary(Cereals)
```

```
##      name               mfr                type              calories
##  Length:77          Length:77          Length:77          Min.   : 50.0
##  Class :character   Class :character   Class :character   1st Qu.:100.0
##  Mode  :character   Mode  :character   Mode  :character   Median :110.0
##                                                           Mean   :106.9
##                                                           3rd Qu.:110.0
##                                                           Max.   :160.0
##
##     protein            fat             sodium           fiber
##  Min.   :1.000   Min.   :0.000   Min.   :  0.0   Min.   : 0.000
##  1st Qu.:2.000   1st Qu.:0.000   1st Qu.:130.0   1st Qu.: 1.000
##  Median :3.000   Median :1.000   Median :180.0   Median : 2.000
##  Mean   :2.545   Mean   :1.013   Mean   :159.7   Mean   : 2.152
##  3rd Qu.:3.000   3rd Qu.:2.000   3rd Qu.:210.0   3rd Qu.: 3.000
##  Max.   :6.000   Max.   :5.000   Max.   :320.0   Max.   :14.000
##
##      carbo           sugars           potass          vitamins
##  Min.   : 5.0   Min.   : 0.000   Min.   : 15.00   Min.   :  0.00
##  1st Qu.:12.0   1st Qu.: 3.000   1st Qu.: 42.50   1st Qu.: 25.00
##  Median :14.5   Median : 7.000   Median : 90.00   Median : 25.00
##  Mean   :14.8   Mean   : 7.026   Mean   : 98.67   Mean   : 28.25
##  3rd Qu.:17.0   3rd Qu.:11.000   3rd Qu.:120.00   3rd Qu.: 25.00
##  Max.   :23.0   Max.   :15.000   Max.   :330.00   Max.   :100.00
##  NA's   :1      NA's   :1        NA's   :2
##      shelf            weight          cups            rating
##  Min.   :1.000   Min.   :0.50   Min.   :0.250   Min.   :18.04
##  1st Qu.:1.000   1st Qu.:1.00   1st Qu.:0.670   1st Qu.:33.17
##  Median :2.000   Median :1.00   Median :0.750   Median :40.40
##  Mean   :2.208   Mean   :1.03   Mean   :0.821   Mean   :42.67
##  3rd Qu.:3.000   3rd Qu.:1.00   3rd Qu.:1.000   3rd Qu.:50.83
##  Max.   :3.000   Max.   :1.50   Max.   :1.500   Max.   :93.70
##
```

```
# Checking structure of Cereals dataset
str(Cereals)
```

```
## spc_tbl_ [77 × 16] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ name    : chr [1:77] "100%_Bran" "100%_Natural_Bran" "All-Bran" "All-Bran_with_Extra_Fibe
r" ...
##  $ mfr     : chr [1:77] "N" "Q" "K" "K" ...
##  $ type    : chr [1:77] "C" "C" "C" "C" ...
##  $ calories: num [1:77] 70 120 70 50 110 110 110 130 90 90 ...
##  $ protein : num [1:77] 4 3 4 4 2 2 2 3 2 3 ...
##  $ fat     : num [1:77] 1 5 1 0 2 2 0 2 1 0 ...
##  $ sodium  : num [1:77] 130 15 260 140 200 180 125 210 200 210 ...
##  $ fiber   : num [1:77] 10 2 9 14 1 1.5 1 2 4 5 ...
##  $ carbo   : num [1:77] 5 8 7 8 14 10.5 11 18 15 13 ...
##  $ sugars  : num [1:77] 6 8 5 0 8 10 14 8 6 5 ...
##  $ potass  : num [1:77] 280 135 320 330 NA 70 30 100 125 190 ...
##  $ vitamins: num [1:77] 25 0 25 25 25 25 25 25 25 25 ...
##  $ shelf   : num [1:77] 3 3 3 3 3 1 2 3 1 3 ...
##  $ weight  : num [1:77] 1 1 1 1 1 1 1 1.33 1 1 ...
##  $ cups    : num [1:77] 0.33 1 0.33 0.5 0.75 0.75 1 0.75 0.67 0.67 ...
##  $ rating  : num [1:77] 68.4 34 59.4 93.7 34.4 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   name = col_character(),
##   ..   mfr = col_character(),
##   ..   type = col_character(),
##   ..   calories = col_double(),
##   ..   protein = col_double(),
##   ..   fat = col_double(),
##   ..   sodium = col_double(),
##   ..   fiber = col_double(),
##   ..   carbo = col_double(),
##   ..   sugars = col_double(),
##   ..   potass = col_double(),
##   ..   vitamins = col_double(),
##   ..   shelf = col_double(),
##   ..   weight = col_double(),
##   ..   cups = col_double(),
##   ..   rating = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Data Preprocessing

```
# To remove all cereals with missing values
cereals <- na.omit(Cereals)

# Retrieve the dimensions (number of rows and columns) of the 'cereals' object
dim(cereals)
```

```
## [1] 74 16
```

Solution 1

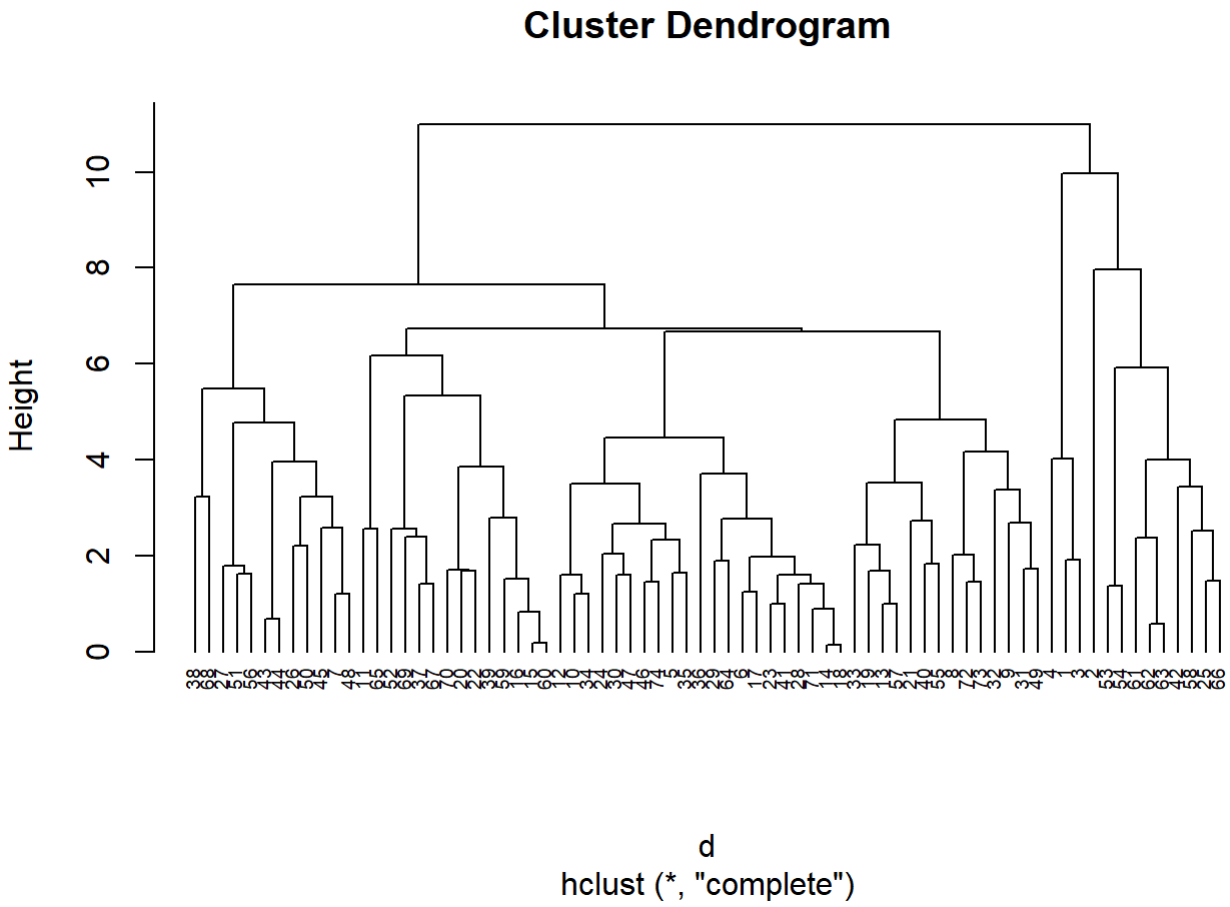
```
# Exclude non-numeric columns
cereals_numeric <- cereals[, sapply(cereals, is.numeric)]

# Normalize the data
cereals_norm <- scale(cereals_numeric)

# We use the euclidean distance measure
d <- dist(cereals_norm, method = "euclidean")

# Hierachical Clustering with the complete linkage method
hc1 <- hclust(d, method = "complete")

# Plot the dendrogram
plot(hc1, main = "Cluster Dendrogram", cex = 0.6, hang = -1)
```




```
# This computation uses the Agglomerative coefficients to compare clustering with different link
age methods

# Perform agglomerative hierarchical clustering using the single linkage method
hc_single <- agnes(cereals_norm, method = "single")

print(hc_single$ac)
```

```
## [1] 0.6067859
```

```
# Perform agglomerative hierarchical clustering using the complete linkage method
hc_complete <- agnes(cereals_norm, method = "complete")

print(hc_complete$ac)
```

```
## [1] 0.8353712
```

```
# Perform agglomerative hierarchical clustering using the average linkage method
hc_average <- agnes(cereals_norm, method = "average")

print(hc_average$ac)
```

```
## [1] 0.7766075
```

```
# Perform agglomerative hierarchical clustering using the ward linkage method
hc_ward <- agnes(cereals_norm, method = "ward")

print(hc_ward$ac)
```

```
## [1] 0.9046042
```

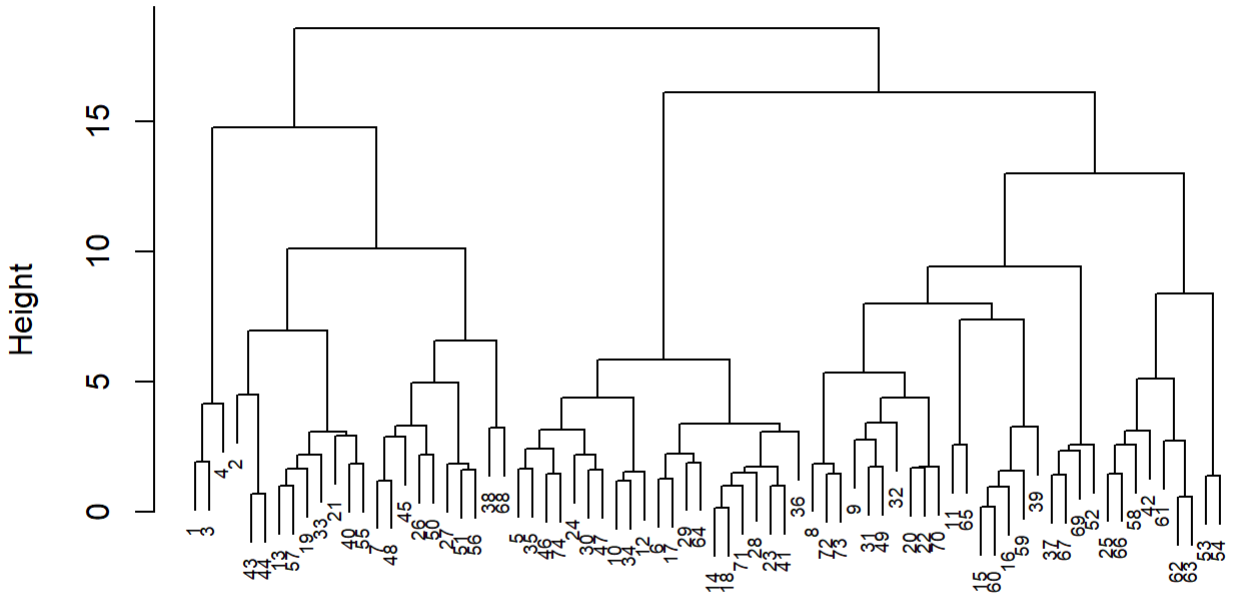
```
# Compare clustering solutions using Agglomerative Coefficients
ac_values <- c(
  single = hc_single$ac,
  complete = hc_complete$ac,
  average = hc_average$ac,
  ward = hc_ward$ac
)

# Print Agglomerative Coefficients for comparison
print(ac_values)
```

```
##    single  complete   average      ward
## 0.6067859 0.8353712 0.7766075 0.9046042
```

```
# To plot the dendrogram for the best method - Ward linkage)

plot(hc_ward,which.plot = 2, cex = 0.6, main = "Ward Linkage Dendrogram")
```



Choosing the best method- Ward Linkage (0.9046042) has the highest agglomerative coefficient and provides the strongest clustering structure, it appears to be the best method for clustering the data, as it yields the most compact clusters.

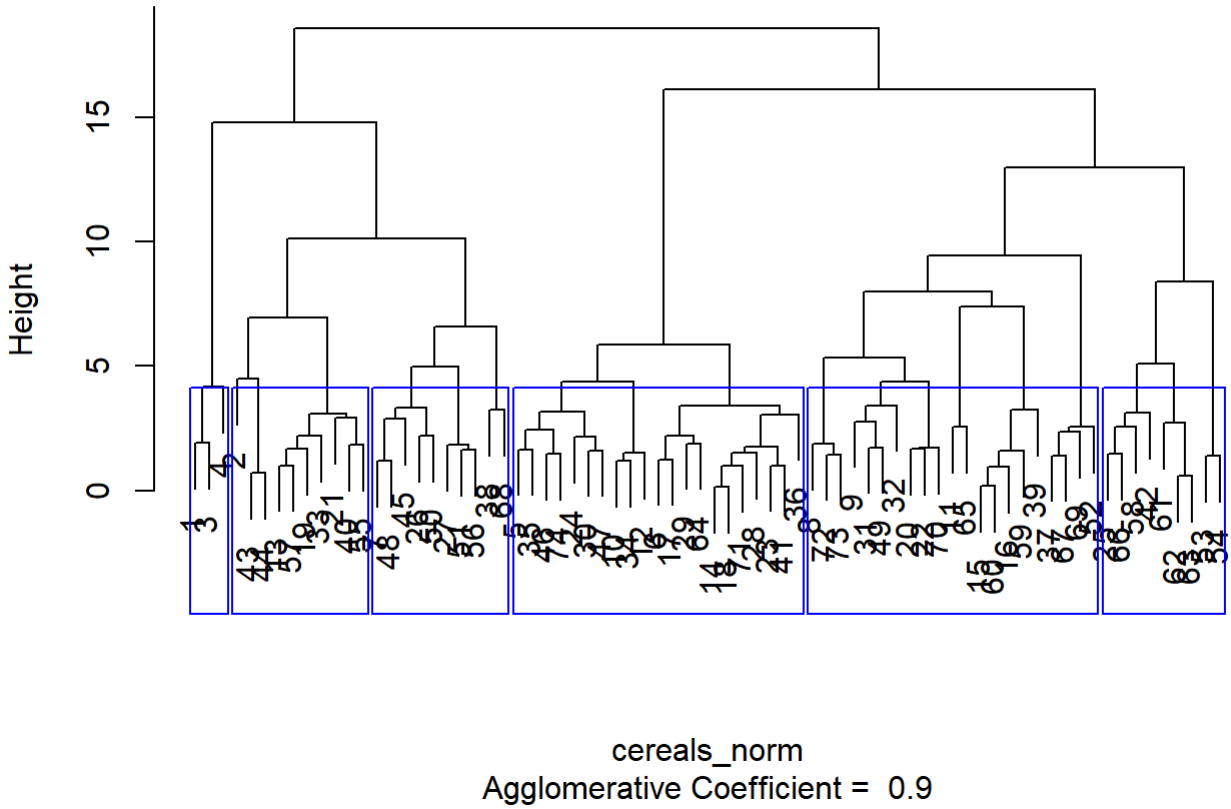
Solution 2

```
# Assign clusters using cutree function, specifying k=6 clusters
fit_hc <- cutree(hc_ward, k = 6)

# Store the clusters in a data frame along with the cereals data
cereals_fit_hc <- cbind(fit_hc, cereals_norm)

# Plot the dendrogram of the chosen clusters
plot(hc_ward, main = "Dendrogram of Chosen Clusters", which.plot = 2)

# Draw rectangles around the 6 clusters
rect.hclust(hc_ward, k = 6, border = "blue")
```




```
# Print the number of clusters chosen
print(paste("Number of clusters chosen:", length(unique(fit_hc))))
```

```
## [1] "Number of clusters chosen: 6"
```

Solution 3

Comment on the structure of the clusters

The clusters are visualized using a dendrogram, which shows how the observations are merged together based on their similarities and various attributes such as calories, protein, fat, sodium, fiber, carbo, sugars, potass, vitamins, shelf, weight, cups, and rating. The rectangles drawn around the dendrogram highlight the 6 clusters formed by the hierarchical clustering algorithm. Each cluster represents a group of cereal products that share similar characteristics based on the attributes included in the analysis.

```
# To check for stability of the Clusters

# For reproducibility
set.seed(123)

# Create cluster partitions A and B
partition_A <- sample(1:2, nrow(cereals_norm), replace = TRUE)
partition_B <- 3 - partition_A

# Fit cluster on partition A and select number of clusters based on dendrogram
cluster_A <- cutree(hc_ward, k = 6)

# Use cluster centroids from A to assign records in partition B
cluster_centroids_A <- aggregate(cereals_norm[partition_A, ], by = list(cluster_A), FUN = mean)
[, -1]
cluster_B <- apply(cereals_norm[partition_B, ], 1, function(x) which.min(colSums((t(cluster_cent
roids_A) - x)^2)))

# Assess cluster consistency
min_len <- min(length(cluster_A), length(cluster_B))  # Ensure both vectors have the same length
consistency <- sum(cluster_A[1:min_len] == cluster_B[1:min_len]) / min_len

# Print the consistency measure
print(paste("Consistency of cluster assignments compared between partitions A and B:", consisten
cy))
```

```
## [1] "Consistency of cluster assignments compared between partitions A and B: 0.14864864864864
9"
```

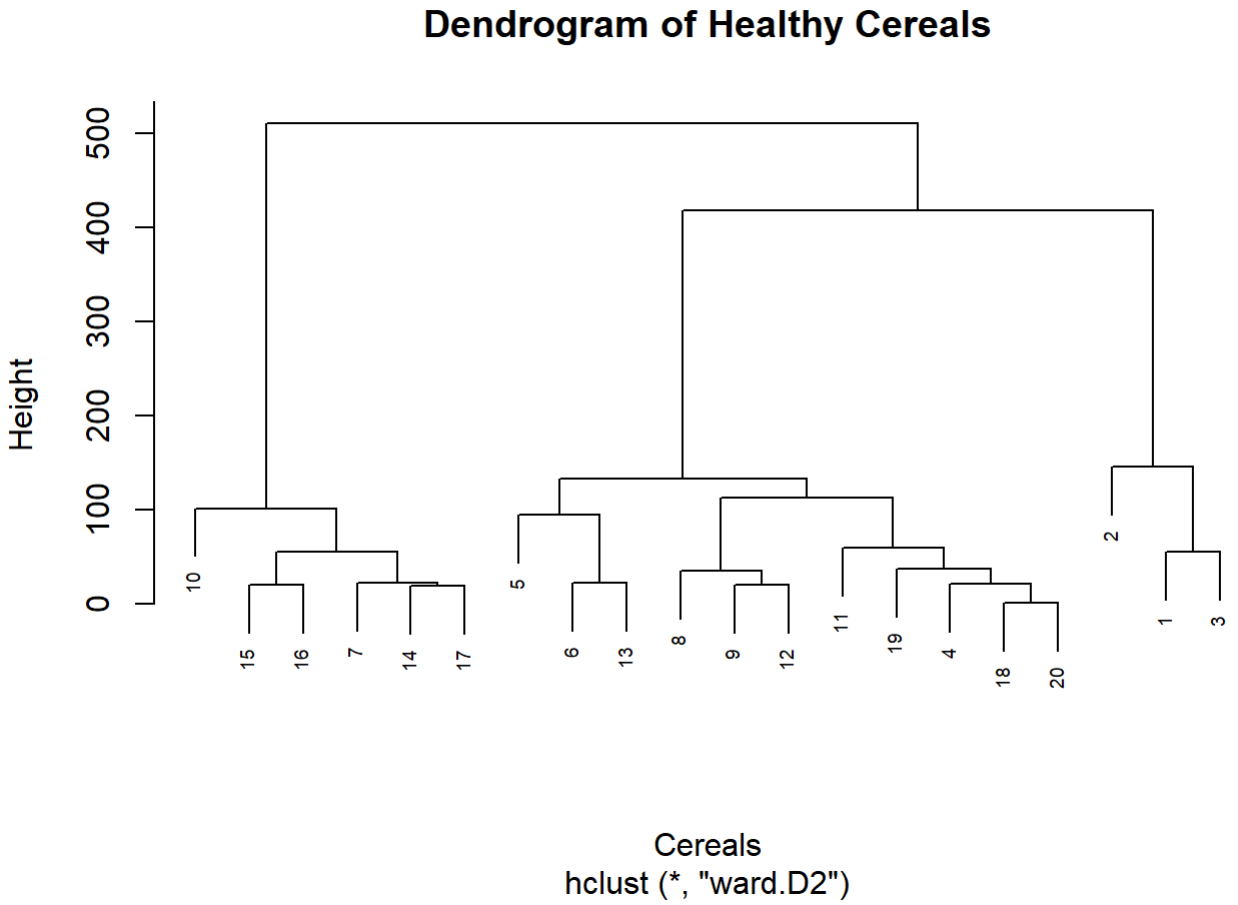
Solution 4

```
# Defining criteria for Healthy Cereals Low sugar content (< 10) and high fiber content (> 2)
healthy_criteria <- cereals$sugars < 10 & cereals$fiber > 2

# To filter Cereals Based on Healthy Criteria
healthy_cereals <- cereals[healthy_criteria, ]

# To Perform hierarchical clustering on the filtered healthy cereals
hc_ward <- hclust(dist(healthy_cereals[, c("calories", "protein", "fat", "sodium", "fiber", "car
bo", "sugars", "potass")]), method = "ward.D2")

# Visualize the Dendrogram
plot(hc_ward, main = "Dendrogram of Healthy Cereals", xlab = "Cereals", sub = NULL, cex = 0.6)
```



The dendrogram visually represents the hierarchical clustering of healthy cereals based on their nutritional attributes which includes calories, protein, fat, sodium, fiber, carbo, sugars, and potass. Cereals that are closer together on the dendrogram have similar nutritional profiles, while those farther apart are dissimilar. The height of each branch indicates the distance at which clusters were merged during the clustering process. This visualization helps identify natural groupings or clusters of healthy cereals based on their nutritional composition.

Overall, Normalization would not be necessary for identifying “healthy cereals” in the cereals dataset due to similar scales among attributes representing nutritional values. The criteria for defining “healthy cereals,” such as low sugar and high fiber content, align well with these attributes.