Mixture model-based clustering of clients for a wholesale provider

STATS/CSE 790 Assignment 2

2024-02-01

Pao Zhu Vivian Hsu (400547994)

Introduction

In this paper, we perform model-based clustering methods to classify clients of a wholesale distributor using data on their annual spend for a range of product categories. Results of this study may help wholesale businesses better understand their clients to improve product offerings to clients. The data in this study comes from the UCI Machine Learning Repository (Cardoso, 2014). It contains 440 rows of client data and 8 variables describing the type of client (retail or hotel/restaurant/cafe), region, and the annual spend in various product categories. All spend amounts are in monetary units.

Methods

We first started the study by ensuring there were no missing data values and investigating the pattern of the data in a pairs plot. The plot revealed that there was heavy skewing in many of the product categories. This appeared to be caused by outliers as highlighted in the boxplot in Figure 1 below. To handle this, we capped the outliers using the 1.5 IQR rule. As a note, region was excluded from the study since we decided to focus the clustering on the type of client instead.

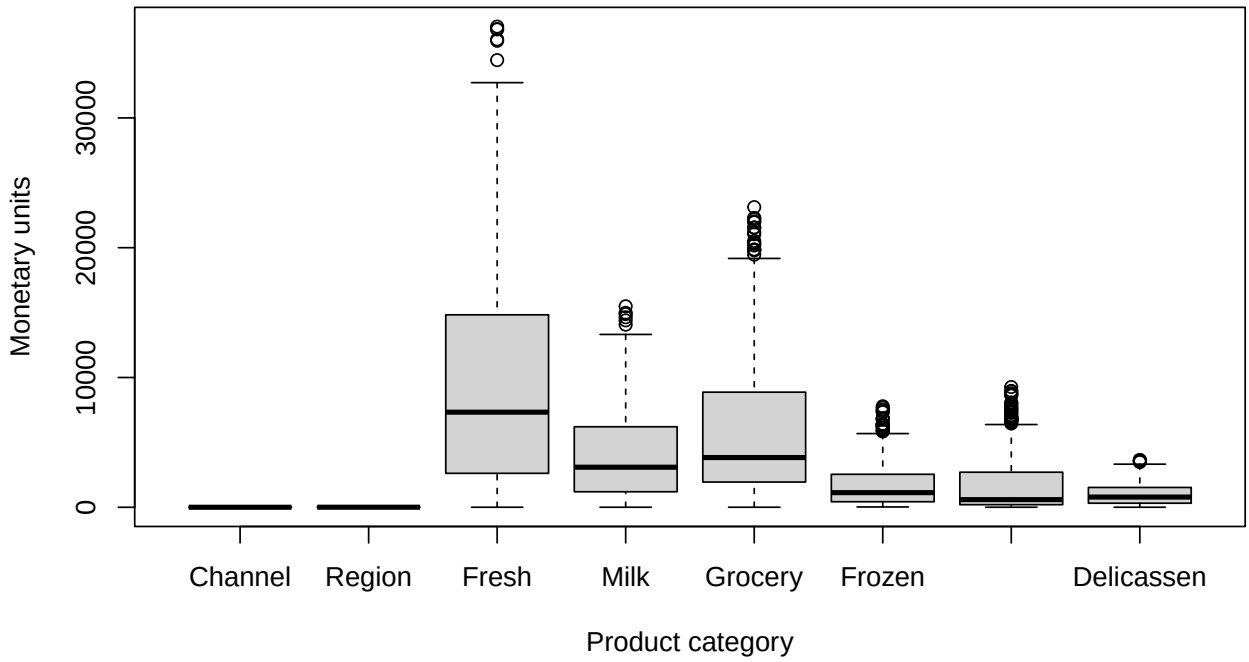


Figure 1: Outliers in the product categories

Next, we built two models using mixture model-based clustering methods. The first model used the Gaussian Parsimonious Clustering Model (GPCM) method. This form of clustering applies all model types within the GPCM family to the data, performs the expectation–maximization (EM) algorithm, and picks the best model using a penalizing criterion such as Bayesian Information Criterion (BIC) (King-Yu, 2024a). Since there are two classes, we tried running the model for 1 to 4 components using k-means for initialization. The best model had a covariance model type of VVI (variable volume, variable shape, and axis-aligned orientation) with 4 components and a BIC value of -5995.946.

The second model used the Mixture of Factor Analyzers (MFA) method. This form of clustering is similar to GPCM, but applies the data to all model types within the Parsimonious Gaussian Mixture Models (PGMM) family, and uses the an extension of the EM algorithm called the Alternating Expectation-Conditional Maximization algorithm (AECM) (King-Yu, 2024b). Since there are two classes and 6 predictor variables, we tried running the model for 1 to 4 components and 1 to 6 factors. The best model used a CUU model with 4 components, a q value of 1, and a BIC of -5939.351.

Results

Table 1 below summarizes the results of each clustering method. The accuracy is quite low for both models with an Adjusted Rand Index of 0.201 and 0.200 for the GPCM model and MFA models respectively. While the Gaussian Parsimonious Clustering model performed slightly better compared to the Mixture of Factor Analyzers model, the poor performance of both models indicate more work can be done to improve results.

Table 1: Clustering performance comparison

Model	Accuracy	Kappa	Rand Index	Adjusted Rand Index
Gaussian Parsimonious Clustering	0.341	0.136	0.577	0.201
Mixture of Factor Analyzers	0.250	-0.033	0.576	0.200

Conclusion

Based on the results of our study, we conclude that the two mixture model-based clustering models do not classify clients for the wholesale distributor very accurately. Some ways to improve results for future studies include feature engineering, performing more extensive data cleansing prior to modelling, or incorporating subject-matter expertise to interpret results.

References

Cardoso, M. (2014). *Wholesale customers*. UC Irvine Machine Learning Repository. https://doi.org/10.24432/C5030X

King-Yu, S. (2024a). *Statistical learning lecture 4: Model-based clustering i*.

King-Yu, S. (2024b). *Statistical learning lecture 5: Model-based clustering II*.

Appendix

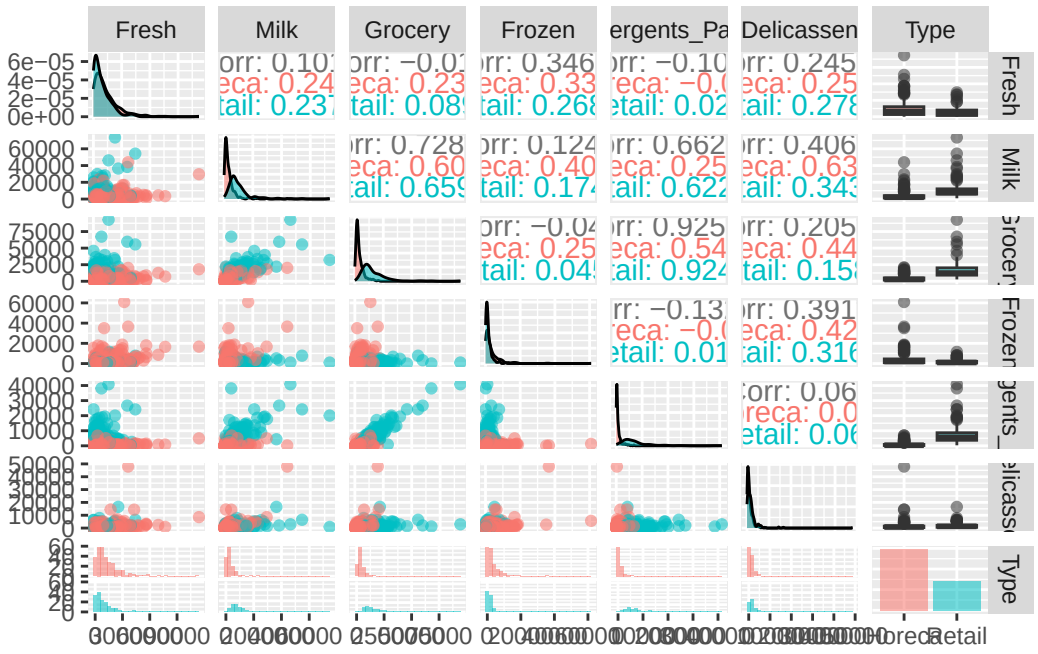
```
# ----- PACKAGE & DATA SETUP ----- #
rm(list=ls())
library(tidyverse)
library("GGally")
library(mixture)
library(pgmm)
library(e1071)
library(kableExtra)

# Load data
wholesale <- read_csv("Wholesale customers data.csv")

# ----- DATA VISUALIZATION & TRANSFORMATION ----- #
# Check for missing data
sum(sapply(wholesale, function(col){ifelse(is.na(col), 1, 0)}))
```

```
[1] 0
```

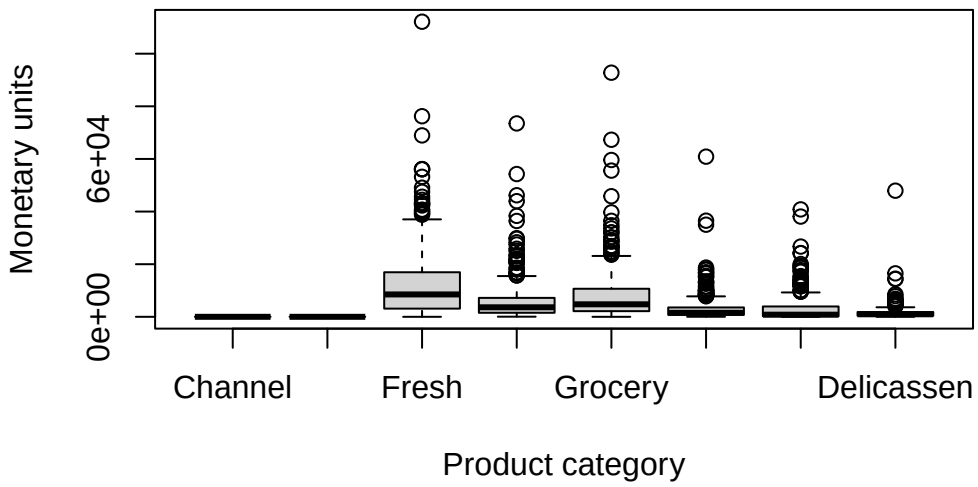
```
sum(sapply(wholesale, function(col){ifelse(sum(col == ""), 1, 0)}))
```

```
[1] 0
```

```
# Pairs plot
wholesaleV <- within(wholesale, Type <- ifelse(Channel==1,"Horeca", "Retail"))
ggpairs(data=wholesaleV[,-c(1,2)], aes(colour=Type, alpha=0.4))
```




```
# Handle outliers

bplot <- boxplot(wholesale, ylab ="Monetary units", xlab ="Product category")
```




```
outlier_val <- bplot$out

outlier_index <- bplot$group

bplot_stats <- bplot$stats

wholesale <- wholesale %>%

  mutate(Fresh = ifelse(Fresh %in% outlier_val[outlier_index==3],

                                          bplot_stats[1,3], Fresh),
```

```
          Milk = ifelse(Milk %in% outlier_val[outlier_index==4],
                                        bplot_stats[1,5], Milk),
          Grocery = ifelse(Grocery %in% outlier_val[outlier_index==5],
                                        bplot_stats[1,5], Grocery),
          Frozen = ifelse(Frozen %in% outlier_val[outlier_index==6],
                                        bplot_stats[1,6], Frozen),
          Detergents_Paper = ifelse(Detergents_Paper %in% outlier_val[outlier_index==7],
                                        bplot_stats[1,7], Detergents_Paper),
          Delicassen = ifelse(Delicassen %in% outlier_val[outlier_index==8],
                                        bplot_stats[1,8], Delicassen))

# Scale data
x <- scale(wholesale[,-c(1,2)])

# ----- GAUSSIAN PARSIMONIOUS CLUSTERING ------ #
# Use k-means
wholesale_gpcm <- gpcm(x, G=1:4, start=2)
summary(wholesale_gpcm)
```

```
BIC for each model, number of components (rows), and covariance structure (columns).
        EVV       VEE       VVE       EVE       VVV       VEV       EEV
1 -7152.203 -7152.203 -7152.203 -7152.203 -7152.203 -7152.203 -7152.203
2 -6663.152 -6623.256 -6653.696 -6629.694 -6218.577 -6521.561 -6673.612
3 -6511.270 -6529.305 -6282.949 -6460.694 -6121.494 -6234.645 -6532.697
4 -6383.535 -6434.080 -6260.572 -6312.064 -6153.306 -6107.112 -6469.229
        EEE       VVI       EVI       VEI       EEI       VII       EII
1 -7152.203 -7559.030 -7559.030 -7559.030 -7559.030 -7528.596 -7528.596
2 -6928.789 -6240.565 -6645.156 -6787.236 -6977.979 -6933.651 -7166.078
3 -6820.343 -6040.759 -6478.474 -6496.013 -6943.891 -6795.270 -7100.033
4 -6795.443 -5995.946 -6319.816 -6434.024 -6687.973 -6662.693 -6973.706
```

```
wholesale_gpcm$best_model
```

```
============================
Best Model According To BIC
============================
Status:  Converged
Covariance Model Type:  VVI
Number of Components:  4
Initalization:  kmeans
BIC:  -5995.946
============================
```

```
tab_gpcm <- table(as.vector(wholesale[,1])$Channel, wholesale_gpcm$map)
tab_gpcm
```

```
    1   2   3   4
1  97  48  15 138
2   9  53  70  10
```

```
classAgreement(tab_gpcm)
```

```
$diag
[1] 0.3409091

$kappa
[1] 0.1359112

$rand
[1] 0.5765686

$crand
[1] 0.2006987
```

```
# ----- MIXTURE OF FACTOR ANALYZERS ----- #
# Use k-means
wholesale_pgmm = pgmmEM(x, rG=1:4, rq=1:6, relax=TRUE)
```

```
You are running pgmm for values of q outisde the recommended range. This is not recommended for non-
The BIC for this model is -5939.351.
```

```
summary(wholesale_pgmm)
```

```
Based on k-means starting values, the best model (BIC) for the range of factors and components used
The BIC for this model is -5939.351.
```

```
tab_pgmm <- table(as.vector(wholesale[,1])$Channel, wholesale_pgmm$map)
tab_pgmm
```

```
    1   2   3   4
1 100 135  47  16
2   9  10  51  72
```

```
classAgreement(tab_pgmm)
```

```
$diag
[1] 0.25

$kappa
[1] -0.03324604

$rand
[1] 0.5759474

$crand
[1] 0.1998278
```

```r
# RESULTS
model <- c("Gaussian Parsimonious Clustering", "Mixture of Factor Analyzers")
diag <- c(classAgreement(tab_gpcm)$diag,
          classAgreement(tab_pgmm)$diag)
kappa <- c(classAgreement(tab_gpcm)$kappa,
           classAgreement(tab_pgmm)$kappa)
rand <- c(classAgreement(tab_gpcm)$rand,
          classAgreement(tab_pgmm)$rand)
crand <- c(classAgreement(tab_gpcm)$crand,
           classAgreement(tab_pgmm)$crand)
summary <- data.frame("Model" = model,
                      "Accuracy" = round(diag,3),
                      "Kappa" = round(kappa,3),
                      "Rand Index" = round(rand,3),
                      "Adjusted Rand Index" = round(crand,3),
                      check.names = FALSE)
kable(summary)
```

Model	Accuracy	Kappa	Rand Index	Adjusted Rand Index
Gaussian Parsimonious Clustering	0.341	0.136	0.577	0.201
Mixture of Factor Analyzers	0.250	-0.033	0.576	0.200