June 22, 2025

From: Robert Mayne, Thomas Sandoval, Yuqi Zeng, Graduate Student Consultant

To: Dr. Marvel Frankenstein

Statistical Exploration of Thermophysical Characteristics in Cheese: Differentiation, Variety Detection, and Texture Classification

EXECUTIVE SUMMARY

This report presents a statistical analysis of 89 cheese samples from two commercial manufacturers, each labeled with one of four textures: Hard, Pasta Filata, Semi-Hard, and Soft. Each sample was measured on six thermophysical properties relevant to classification and production.

Three research questions were addressed: whether thermophysical properties differ by texture, how many distinct cheese varieties exist based on these measurements, and whether texture can be predicted from them. MANOVA confirmed significant differences among textures, except between Pasta Filata and Soft. Clustering techniques consistently identified three natural cheese varieties. Linear Discriminant Analysis achieved 100 percent classification accuracy.

These results support the use of thermophysical measurements for distinguishing textures, identifying cheese varieties, and classifying new samples, valuable tools for quality control and product development.

1.0 - PROJECT DESCRIPTION

This project involves an observational study for Dr. Marvel Frankenstein of Get Cheese Right, PA, using data from 89 cheese samples collected from two commercial manufacturers. The samples represent four texture categories: Hard, Pasta Filata, Semi-Hard, and Soft. Each sample is described by six thermophysical properties relevant to manufacturing and classification.

With data collection complete, the study focuses on evaluating the relationship between physical properties and cheese texture. Multivariate statistical methods are applied to compare textures, identify natural groupings, and assess the potential to classify cheese types based on these measurements. The findings aim to support scientific research, enhance quality control, and inform the development of data-driven tools for use in cheese production.

1.1 - RESEARCH QUESTIONS

Question 1: Are the thermophysical properties of cheese textures different? If so, which textures are different?

Question 2: Based on thermophysical characteristics, how many cheese varieties (not textures) are present?

Question 3: Identify the texture of the cheese using the thermophysical characteristics for new cheese products

1.2 - STATISTICAL QUESTIONS

This section was intentionally omitted, as the research questions were explicitly defined by the client and required no additional context or refinement. Each question directly informed the structure of the analysis and aligned clearly with the variables and statistical methods used, making further elaboration unnecessary.

1.3 - VARIABLES

The data set that the company has collected includes data from two different manufacturers on 89 cheeses of various types from the four cheese textures. The data set includes nine variables listed in the table below.

VARIABLE	DESCRIPTION	TYPE	POSSIBLE VALUES
ID	Cheese identifier	ID	1-89
MANUFACTURER	Cheese manufacturer, Discrete	Discrete	1 or 2
TEXTURE	Texture of Cheese	Qualitative	Hard, Pasta Filata, Semi-hard, Soft
G80	Storage modulus at 80 C	Quantitative	43.18-413.84
VLTMAX	Temp vs at tan	Quantitative	67.15 – 79.08
VCO	Temp v at cross-over	Quantitative	49.97 – 59.69
FMAX	Max resistance force during extension	Quantitative	1.38 – 9.26
FD	Flowing degree	Quantitative	-0.59 – 12.34
FO	Free oil	Quantitative	23.03 – 51.02

The proposed questions by Dr. Frankenstein’s group allow us to answer the questions using the variables provided.

- For the first question, the response variables are the six quantitative thermophysical measurements provided. We could also use manufacturer if we wanted to see if the manufacturer had any effect on the modeling.
- For the second question, we will use the thermophysical measurement variables to perform Principal Component Analysis which creates natural groupings in the data. We will also use a method called k-means to also identify the groupings in the data
- For the third question, texture (Hard, Pasta Filata, Semi-hard and Soft) are the response variables and the thermophysical variables are the predictors/explanatory variables.

2.0 - EXPLORATORY DATA ANALYSIS (EDA)

Full exploratory data analysis was not required for this project. Data provided was already cleaned, with no missing values, outliers of concern, or structural anomalies identified during initial inspection. Because all variables were well-defined and appropriately scaled for analysis, no further EDA was necessary to proceed with the modeling and statistical procedures outlined in subsequent sections.

3.0 –STATISTICAL ANALYSIS

3.1 - MANOVA

To determine whether the thermophysical properties of cheese differ by texture, we used a Multivariate Analysis of Variance (MANOVA) with six quantitative variables: G80, vLTmax, vCO, Fmax, FD, and FO. The goal was to assess whether the multivariate profiles of these physical properties vary across four cheese textures: Hard, Pasta Filata, Semi-Hard, and Soft.

3.1.1 - MODEL ASSUMPTIONS

We assessed the assumptions required for MANOVA prior to analysis:

- Independence was met by design, as each cheese sample was collected and measured independently.
- Multivariate Normality was evaluated within each texture group using Mardia's test and Q-Q plots of Mahalanobis distances (*See A.1 and A.2, respectively)*. No substantial departures from normality were found.
- Equality of Covariance Matrices was confirmed using Box's M test ($\chi^2(63) = 53.80$, $p = 0.789$), indicating no significant difference in group variance-covariance structures.
- Multicollinearity was checked using a correlation matrix (*See B.1)*. While a few variables (e.g., G80 and Fmax) showed moderate correlation ($r < 0.7$), no relationships were strong enough to raise concern.

These results support the appropriateness of the MANOVA model using Wilks' Lambda as the test statistic.

3.1.2 - MODEL RESULTS AND INTERPRETATION

The MANOVA revealed statistically significant differences in the thermophysical characteristics among the four cheese textures. (Wilks' Lambda = 1.99×10^{-5}, $F(18, 226.76) = 565.81$, $p < 0.001$).

This indicates that at least one texture group has a distinct profile of thermophysical properties.

To determine which textures differed, we conducted pairwise MANOVA comparisons using 999 permutations. These tests showed significant differences between all textures except Pasta Filata and Soft ($p = 0.223$), suggesting these two textures share similar thermophysical profiles (*See A.3*).

3.2 - PCA AND K-MEANS ANALYSIS

For this research question, we are asked based on the thermophysical characteristics, how many cheese varieties (not textures) are present in our data. We used Principal Component Analysis (PCA) to reduce the number of dimensions and to minimize much of the variance and allow us to visualize the clusters. Additionally, k-means will be used as a secondary analysis to ensure that the PCA analysis makes overall suitable clusters. Based on the results of the clusters, we can obtain distinct cheese varieties and compare how the clusters align.

We then applied k-means clustering on the data that was created through the PCA process. This technique groups cheese samples based on their similar characteristics from the thermophysical variables provided. Because we did not know the groupings beforehand, we used both these unsupervised learning approaches to processes seeking out the natural groupings in the data

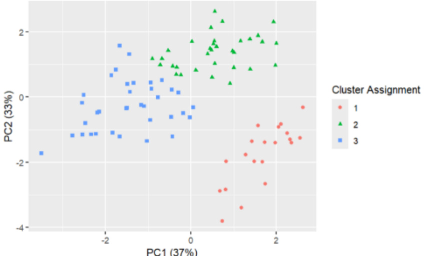
3.2.1 - MODEL ASSUMPTIONS

To perform PCA, there is not a formal list of assumptions to be met, however we needed to check if there were any strong correlations between the two variables and check if there were no major outliers. Before doing so, we first standardized the data (mean = 0, SD = 1). Then we created a correlation matrix (*See B.1)* to identify any correlations that appeared in the data to enable us to use PCA. Because no two variables were strongly correlated (all < 0.7), there were a few moderate correlations which support the use of PCA. Lastly, we checked for outliers with boxplots (*See A.5)*. There were a few potential outliers in the variables FD and FO, they were not extreme enough to need to change the processes and significantly change the analyses.

3.2.2 - INTERPRETATIONS

To determine the optimal number of clusters (i.e. Cheese varieties) there are, we first used the elbow method with k-means clustering. This allows us to evaluate how the variation in the cluster changes as the number of cluster increases. Based on the plot (*see B.2 & B.4*)– this occurs at k = 3, so three clusters best represented the structure of the data.

Next, we used PCA to reduce the data to two principal components and visualized the k-means cluster assignments using those PCAs. Using the first two principal components, which explained 70% of the data, we plotted the cheese sample and identified them by color based on their k-means cluster assignments (*See B.3)*. The plot clearly showed defined and well-separated groupings which supported that there were three distinct cheese varieties based on the thermophysical characteristics of the cheeses.



Then, we then applied hierarchical clustering using Euclidean distance and complete linkage. The dendrogram (*See B.4)* supported the three-group solution, which is compatible with the k-mean number of groupings. Lastly, we used the hierarchical cluster assignments on the same PCA-space (*See B.5)* which showed very similar groupings to the PCA space completed first again supporting that there were three natural cheese varieties.

3.2.3 - CLUSTERING

Each cluster can be summarized by the following characteristics:

Cluster 1	Cluster 2	Cluster 3
High firmness and low flow	Moderate firmness, lower oil, and meltability	Soft and oily, high flow

You can see the assignments in table (*see B.7)* as well as the characteristics of each trait given that grouping. This then allows us to determine the characteristics of each cluster/ variety of cheese.

3.3 - DISCRIMINANT ANALYSIS

For research question 3, our goal is to predict the texture type of a new cheese sample based on six thermophysical characteristics: G80, vLTmax, vCO, Fmax, FD, and FO. Each cheese sample was labeled as one of four texture types: Hard, Pasta Filata, Semi-Hard, or Soft. Therefore, this part involves a supervised classification problem. To build a reliable classification model, we employed Linear Discriminant Analysis (LDA) — a statistical technique commonly used for classifying observations into predefined groups based on features.

3.3.1 - MODEL ASSUMPTIONS

Before applying LDA, we verified two key assumptions. The results showed that the assumptions for LDA were reasonably satisfied. Thus, linear discriminant analysis is appropriate for these data.

- **Normality of Features within Each Group**: Shapiro-Wilk tests(*See C.1*) showed that most texture groups and thermophysical characteristics satisfy the normality assumption (p-values>0.05). And a few exceptions showed mild departure from normality (p-values<0.05). However, it was still appropriate to use LDA model, because Shapiro-Wilk tests can be sensitive with small sample sizes, and LDA is robust to mild normality violations.
- **Homogeneity of Covariance Matrices Across Groups**: Box's M test produced a p-value of 0.789 (>0.05) with degree of freedom of 63(*See C.2*), indicating no significant differences between the variance-covariance matrices across texture groups, supporting the use of LDA.

3.3.2 - MODEL INTERPRETATION

The LDA model calculates a discriminant score for each group using a linear combination of the six features. Each equation describes how the values of the six characteristics are weighted to compute a score for a given cheese sample. These discriminant functions can be interpreted as scoring rules: for a new cheese sample, the group with the highest score determines the predicted texture.

TEXTURE	FUNCTION
Hard	$\widehat{d^L_{Hard}}(x) = -1465.316 + 0.294 * G80 + 18.879 * vLTmax + 34.808 * vCO$ $-53.061 * Fmax + 1.607 * FD - 4.096 * FO$
Pasta Filata	$d^L_{\widehat{PastaFilata}}(x) = -1439.382 + 0.273 * G80 + 18.845 * vLTmax + 34.523 * vCO$ $-52.501 * Fmax + 1.613 * FD \pm 4.289 * FO$
Semi-Hard	$d^L_{\widehat{Semi-Hard}}(x) = -1447.824 + 0.274 * G80 + 18.642 * vLTmax + 34.891 * vCO$ $-52.626 * Fmax + 0.984 * FD - 4.063 * FO$
Soft	$\widehat{d^L_{Soft}}(x) = -1478.557 + 0.277 * G80 + 18.858 * vLTmax + 35.080 * vCO$ $-53.445 * Fmax + 1.480 * FD - 3.943 * FO$

Then, to evaluate how well the LDA model generalizes to unseen data, we used **10-fold cross-validation** on the training set. This method splits the data into 10 parts, using 9 for training and 1 for testing in each round. This process repeats 10 times so that each sample is tested once. The resulting confusion matrix showed no misclassification across the folds — the model achieved 100% accuracy during cross-validation, meaning all texture types were correctly identified (*see C.5*).

Furthermore, we held out a separate set of 25 cheese samples that were not used in model training. This final test set was used to evaluate the model's predictive ability to truly new data. The model again achieved 100% accuracy (*see C.3*), correctly identifying the texture type for every cheese sample in the test set.

4.0 – RECOMMENDATIONS

Question 1: The MANOVA provided strong evidence that cheese texture is associated with differences in thermophysical properties. All textures were significantly different from one another, except for Pasta Filata and Soft, which appeared statistically similar in their physical profiles. The assumptions were adequately met, supporting the validity of conclusions. Supporting figures, test results, and tables are provided in the Appendix.

Question 2: Based on the analysis, there appears to be three different varieties of cheese in the dataset. We used an unsupervised learning task because we did not know the true groupings beforehand. We verified that there were three groupings by using PCA, k-means clustering, and hierarchical clustering, which showed each time that there were three natural groupings based on the thermophysical properties.

Question 3: The six thermophysical properties of cheese can be used to accurately classify cheese into known texture categories. Specifically, the LDA model, a supervised classifier, effectively distinguishes between texture types. All model assumptions were reasonably satisfied, and both cross-validation and final testing confirmed excellent classification performance with no misclassification and 100% accuracy.

5.0 - RESOURCES

This report is based on data from two commercial cheese manufacturers, consisting of six thermophysical measurements across 89 cheese samples. Analyses and visualizations were performed in RStudio using relevant statistical packages. Multivariate analysis, supervised learning (LDA), and unsupervised techniques (PCA, k-means, hierarchical clustering) were used to explore variable relationships, evaluate group differences, and identify natural clusters in support of the research questions.

6.0 - CONSIDERATIONS

While the results of our statistical analysis are promising, there are several considerations to keep in mind when interpreting the findings and applying the model:

Sample size limitation: The dataset contains 89 observations, which is relatively small given the number of cheese texture types and thermophysical variables involved. While the models performed exceptionally well, the limited sample size may inflate apparent accuracy. Additional data collection would strengthen the reliability of the conclusions and allow for more robust model validation.

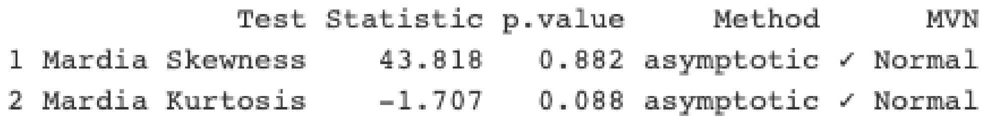
Assumption Sensitivity: Although LDA is known to be robust to moderate violations of normality, the mild normality violation should be acknowledged, especially if the model is applied to broader or more varied cheese products in the future.

We thank Dr. Marvel Frankenstein and their research team for the opportunity to contribute to this interesting and meaningful project. It has been a pleasure supporting your work, and we hope these insights will help guide further product development, quality control, and research initiatives.

APPENDIX

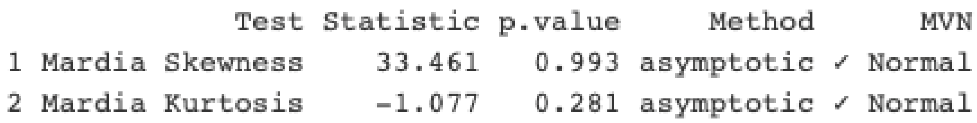
A.1 Mardia's Test Values

Texture: Hard



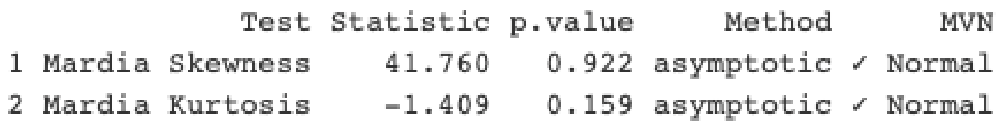
```
             Test Statistic p.value     Method      MVN
1 Mardia Skewness    43.818   0.882 asymptotic ✓ Normal
2 Mardia Kurtosis    -1.707   0.088 asymptotic ✓ Normal
```

Texture: Semi-Hard



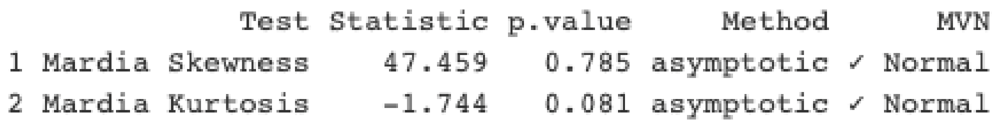
```
             Test Statistic p.value     Method      MVN
1 Mardia Skewness    33.461   0.993 asymptotic ✓ Normal
2 Mardia Kurtosis    -1.077   0.281 asymptotic ✓ Normal
```

Texture: Soft



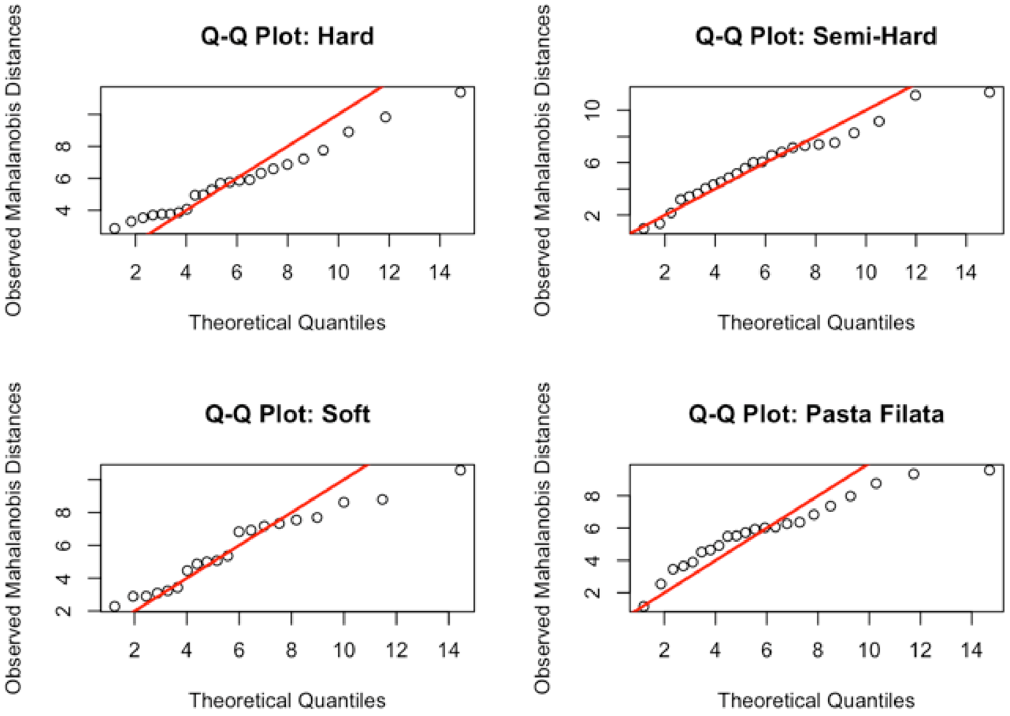
```
             Test Statistic p.value     Method      MVN
1 Mardia Skewness    41.760   0.922 asymptotic ✓ Normal
2 Mardia Kurtosis    -1.409   0.159 asymptotic ✓ Normal
```

Texture: Pasta Filata



```
             Test Statistic p.value     Method      MVN
1 Mardia Skewness    47.459   0.785 asymptotic ✓ Normal
2 Mardia Kurtosis    -1.744   0.081 asymptotic ✓ Normal
```

A.2 Q-Q plots of Mahalanobis distances



Q-Q Plot: Hard

Q-Q Plot: Semi-Hard

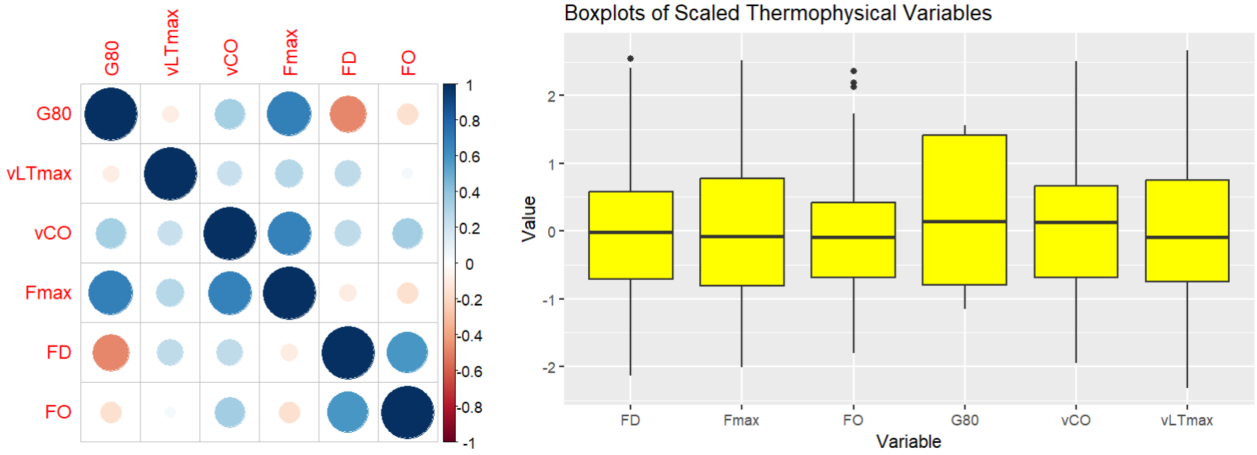
Q-Q Plot: Soft

Q-Q Plot: Pasta Filata

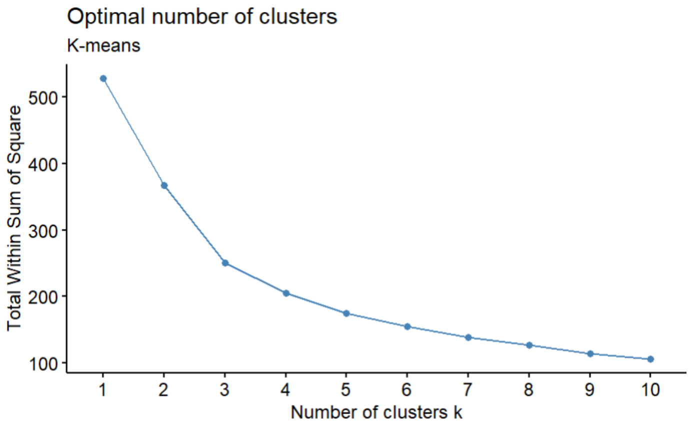
A.3 Pairwise distances between means

	d <dbl>	UCL (95%) <dbl>	Z <dbl>	Pr > d <dbl>
Hard:Pasta Filata	304.36709	76.93177	4.7812479	0.001
Hard:Semi–Hard	173.65328	79.73413	3.0621406	0.001
Hard:Soft	351.11959	76.28911	5.0826463	0.001
Pasta Filata:Semi–Hard	130.78664	76.89193	2.5966559	0.001
Pasta Filata:Soft	50.28927	79.50654	0.8204222	0.223
Semi–Hard:Soft	177.84004	79.37222	3.1233759	0.001

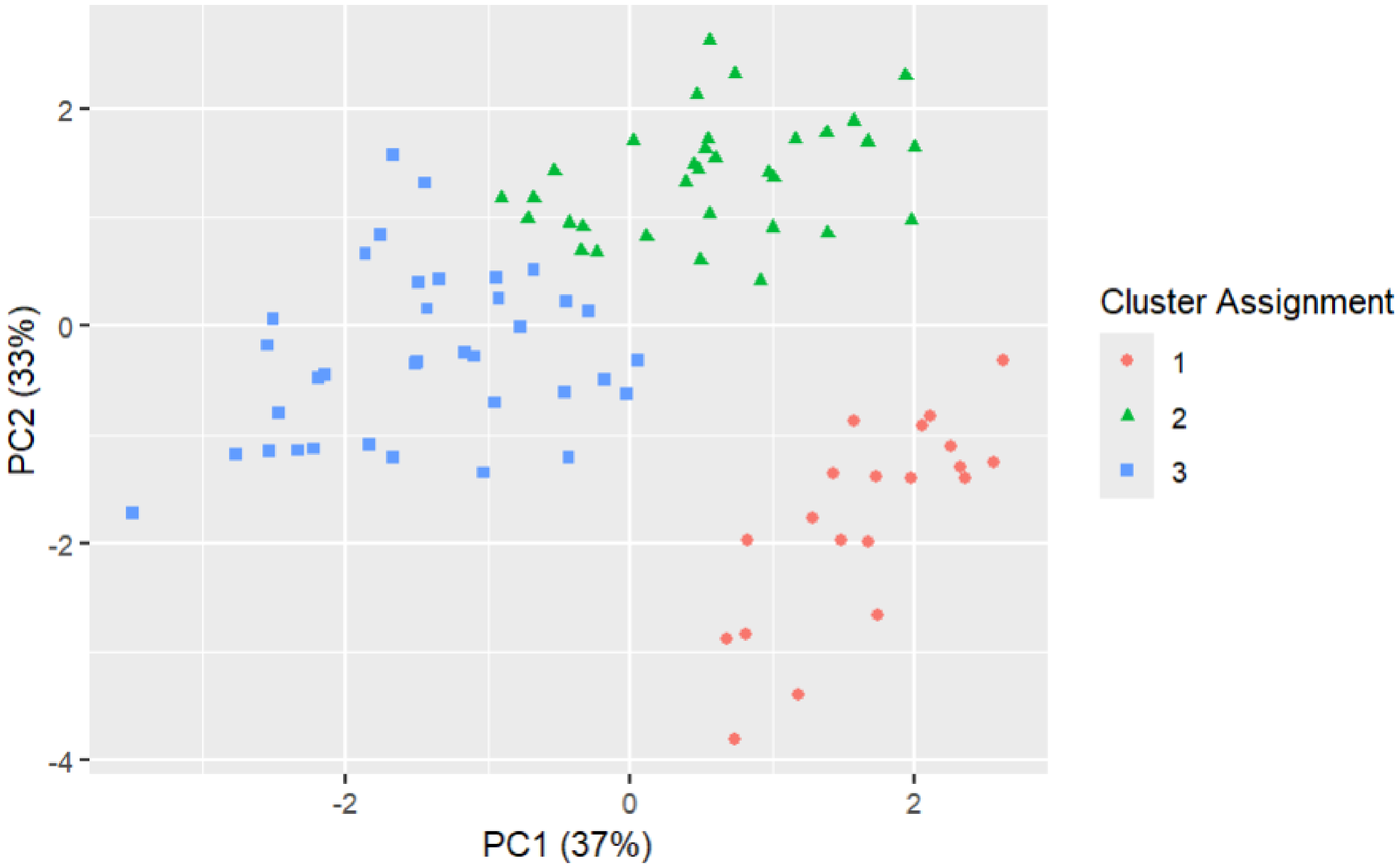
B.1 Correlation Matrix and A.5 boxplots of variable data



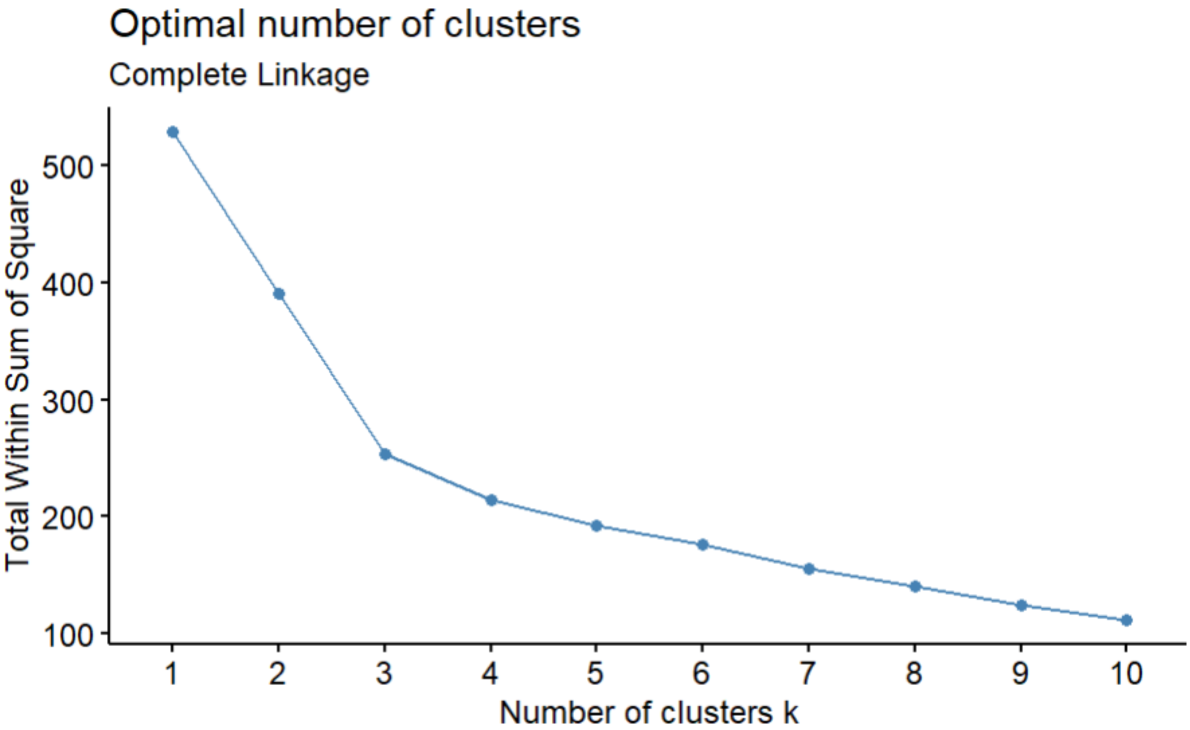
B.2 Elbow Plot for Optimal number of clusters using PCA



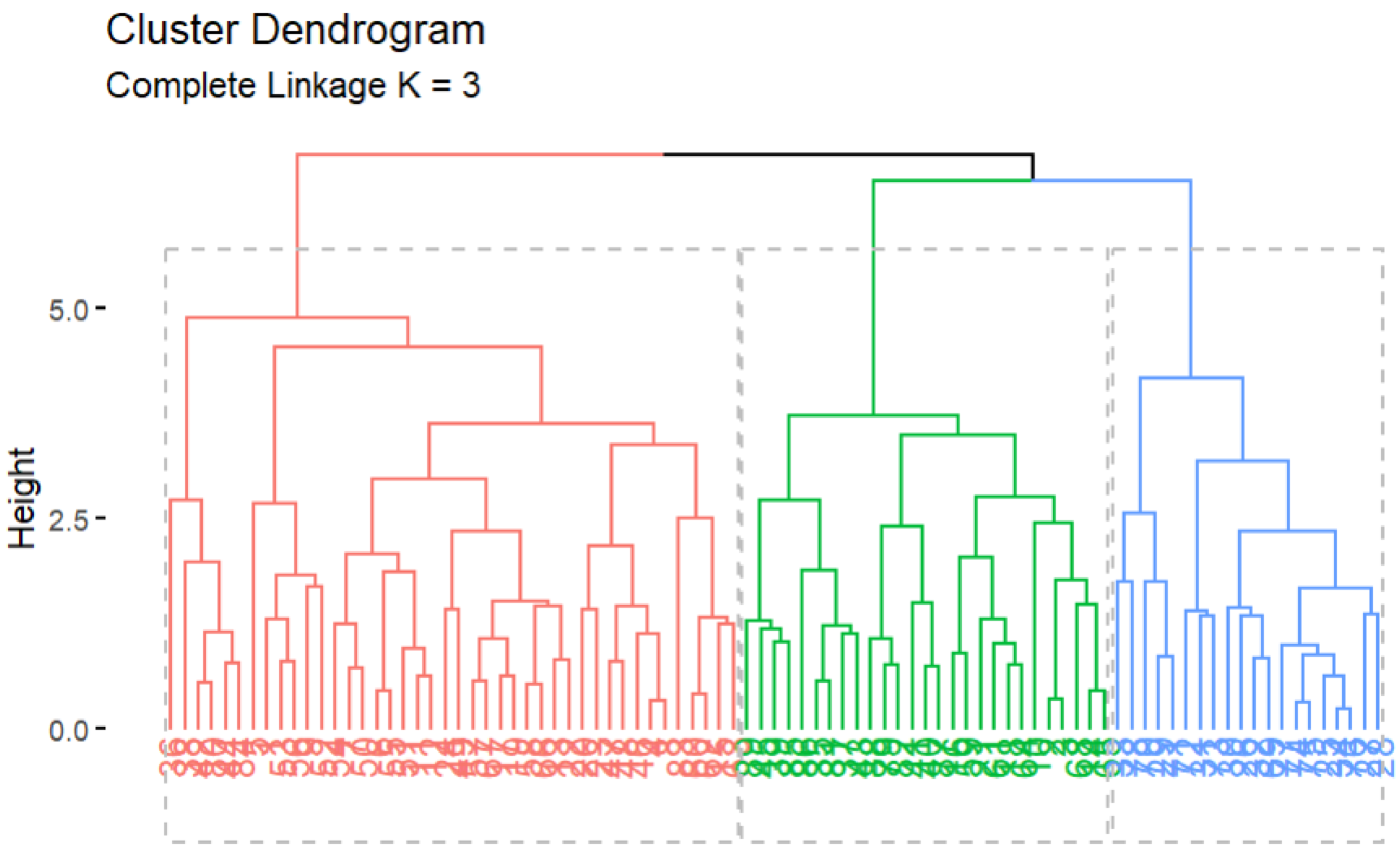
B.3 PCA visualization with 3 clusters



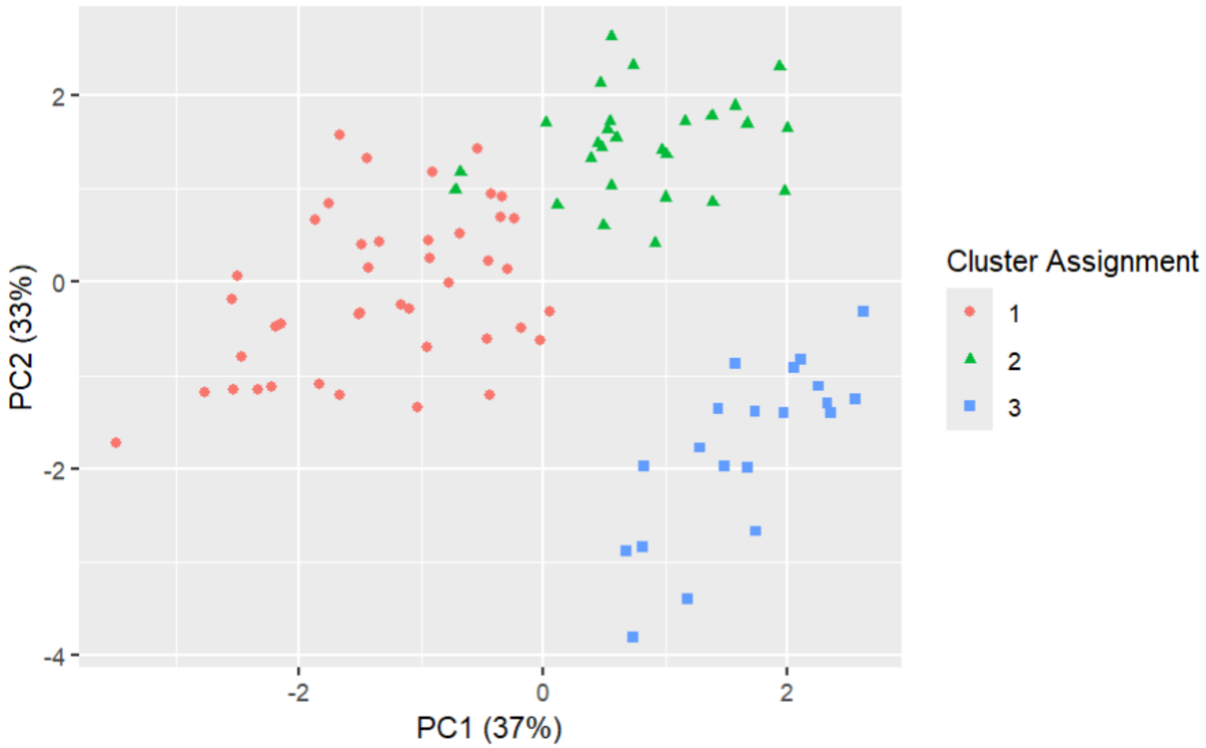
B.4 Elbow Plot for Optimal number of clusters using k-means



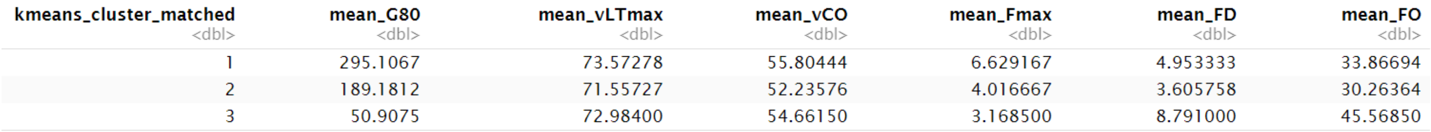
B.5 Hierarchical Clustering Dendrogram



B.6 k-means visualization with 3 clusters



B.6 Characteristics of the Cluster Groupings



kmeans_cluster_matched	mean_G80	mean_vLTmax	mean_vCO	mean_Fmax	mean_FD	mean_FO
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	295.1067	73.57278	55.80444	6.629167	4.953333	33.86694
2	189.1812	71.55727	52.23576	4.016667	3.605758	30.26364
3	50.9075	72.98400	54.66150	3.168500	8.791000	45.56850

B.7 Cluster assignments and if the clusters match per procedure

ID	Hierarchical Clustering	k-means Clustering	Match
1	1	1	Match
2	2	2	Match
3	2	2	Match
4	1	1	Match
5	1	1	Match
6	1	1	Match
7	1	1	Match
8	1	1	Match
9	1	1	Match
10	2	2	Match
11	1	1	Match
12	1	1	Match
13	1	1	Match
14	2	2	Match
15	1	1	Match
16	2	2	Match
17	1	1	Match
18	2	2	Match
19	1	1	Match
20	1	1	Match
21	2	2	Match
22	1	1	Match
23	1	1	Match
24	1	2	No Match
25	3	3	Match
26	3	3	Match
27	3	3	Match
28	3	3	Match
29	3	3	Match
30	3	3	Match
31	3	3	Match
32	3	3	Match
33	3	3	Match
34	3	3	Match
35	2	2	Match
36	1	1	Match

37	2	2	Match
38	1	1	Match
39	2	2	Match
40	2	2	Match
41	2	2	Match
42	1	1	Match
43	2	2	Match
44	1	1	Match
45	2	2	Match
46	1	1	Match
47	1	1	Match
48	1	1	Match
49	1	1	Match
50	1	2	No Match
51	1	1	Match
52	1	1	Match
53	1	1	Match
54	1	1	Match
55	1	1	Match
56	1	1	Match
57	1	2	No Match
58	1	2	No Match
59	2	2	Match
60	1	1	Match
61	2	2	Match
62	1	1	Match
63	1	1	Match
64	2	2	Match
65	2	2	Match
66	1	2	No Match
67	1	2	No Match
68	2	2	Match
69	3	3	Match
70	3	3	Match
71	3	3	Match
72	3	3	Match
73	3	3	Match
74	3	3	Match
75	3	3	Match
76	3	3	Match
77	3	3	Match
78	3	3	Match
79	2	2	Match
80	1	1	Match

81	2	2	Match
82	2	2	Match
83	1	1	Match
84	1	1	Match
85	2	2	Match
86	2	2	Match
87	2	2	Match
88	2	2	Match
89	2	2	Match

C.1 Normality Assumption Tests: Shapiro-Wilk Results by Texture Group

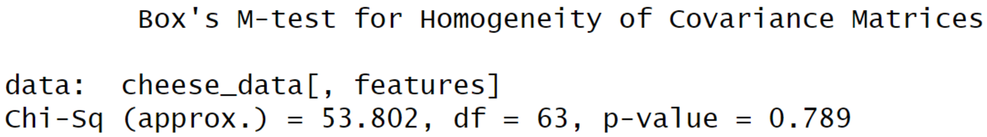
Shapiro-Wilk tests were conducted for each thermophysical feature within each texture group. The table below summarizes the p-values. A p-value below 0.05 suggests potential deviation from normality.

Texture	Feature	Shapiro-Wilk p-value
Hard	G80	0.4755
Hard	vLTmax	0.2152
Hard	vCO	0.9701
Hard	Fmax	0.8914
Hard	FD	0.3294
Hard	FO	0.5388
Pasta Filata	G80	**0.004928**
Pasta Filata	vLTmax	0.9304
Pasta Filata	vCO	**0.01773**
Pasta Filata	Fmax	0.0702
Pasta Filata	FD	0.5862
Pasta Filata	FO	0.8335
Semi-Hard	G80	0.7858
Semi-Hard	vLTmax	0.8615
Semi-Hard	vCO	0.45
Semi-Hard	Fmax	0.5407
Semi-Hard	FD	0.9664
Semi-Hard	FO	0.3548
Soft	G80	0.9938
Soft	vLTmax	**0.0425**
Soft	vCO	0.3874
Soft	Fmax	0.3748

Soft	FD	0.7229
Soft	FO	0.637

C.2 Homogeneity of Covariances Assumption test: Box’s M Test

The p-value > 0.05 suggests no significant difference in covariance matrices between groups, satisfying the assumption of homogeneity.



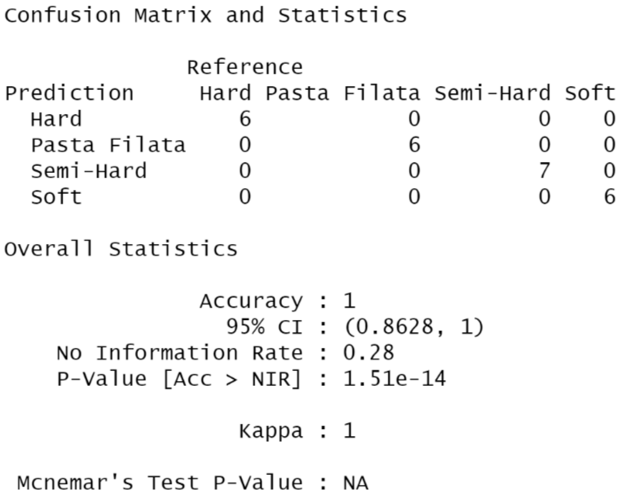
```
              Box's M-test for Homogeneity of Covariance Matrices

data:  cheese_data[, features]
Chi-Sq (approx.) = 53.802, df = 63, p-value = 0.789
```

C.3 Test Set Performance (Held out 25 Observations)

Prediction Accuracy: 100%

Number of misclassifications: 0

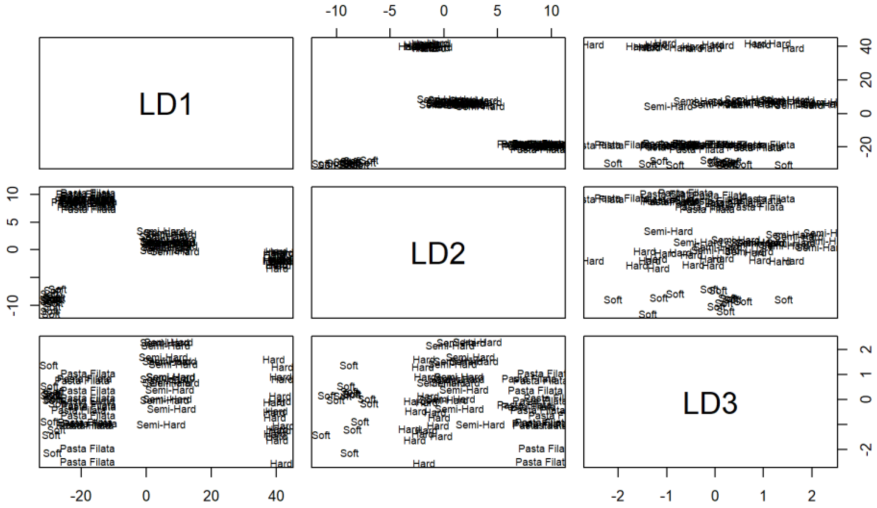
All 25 test samples were correctly classified into their respective texture types.



```
Confusion Matrix and Statistics

              Reference
Prediction     Hard Pasta Filata Semi-Hard Soft
  Hard            6            0         0    0
  Pasta Filata    0            6         0    0
  Semi-Hard       0            0         7    0
  Soft            0            0         0    6

Overall Statistics

               Accuracy : 1
                 95% CI : (0.8628, 1)
    No Information Rate : 0.28
    P-Value [Acc > NIR] : 1.51e-14

                  Kappa : 1

 Mcnemar's Test P-Value : NA
```

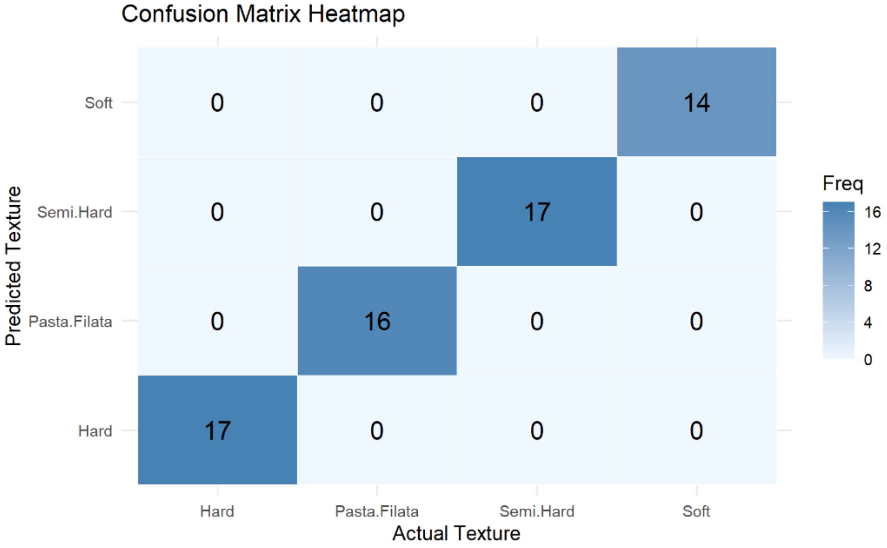
C.4 Discriminant Score Plot

Samples are plotted using their scores on the three linear discriminant axes. Each point represents a cheese sample with its texture type. Clear separation indicates strong discriminative power of the LDA model.



C.5 Cross-Validation Confusion Matrix Heatmap

The heatmap provides a visual representation of prediction accuracy across all texture categories. The heatmap below is generated from cross-validation. In this plot, the x-axis represents the actual texture type, while the y-axis represents the predicted texture type assigned by our LDA model. The numbers inside the tiles indicate the number of samples classified into each actual-predicted combination. Darker tiles represent higher frequencies. The plot displays a perfect diagonal pattern, suggesting that every observation was correctly classified into its actual category. No off-diagonal entries were observed, indicating zero misclassification.



D.1 Program for Question 1 (R code):

```
# Question 1 – MANOVA
# List Library
library(tidyverse)   # for data manipulation and visualization
library(car)         # for MANOVA tools and Pillai’s trace
library(ggplot2)     # for plotting
library(dplyr)
library(RRPP)       # for pirwise MANOVA
library(MVN)        # for assessing Multivariate Normality
library(biotools)   # for Box's M test
library(corrplot)

#Load Datasets
cheese <- read.csv("/Users/robertmayne/Documents/STAT 580/cheeseThermophysical.csv",
header=TRUE)

# Check structure of the data
# Here we see that the cheese is a character that needs to be converted to a factor.
glimpse(cheese)

# Convert texture to a factor (important for MANOVA)
cheese$texture <- factor(cheese$texture)

# Multivariate Normality


# Subset the dataset to include only samples with the specific textures,ckeeping only the
six thermophysical response variables.
# This allows us to assess multivariate normality within this texture group.
hard_data <- subset(cheese, texture == "Hard")[, c("G80", "vLTmax", "vCO", "Fmax", "FD",
"FO")]
semi_data <- subset(cheese, texture == "Semi-Hard")[, c("G80", "vLTmax", "vCO", "Fmax",
"FD", "FO")]
pasta_data <- subset(cheese, texture == "Pasta Filata")[, c("G80", "vLTmax", "vCO", "Fmax",
"FD", "FO")]
soft_data <- subset(cheese, texture == "Soft")[, c("G80", "vLTmax", "vCO", "Fmax", "FD",
"FO")]

# Run Mardia's test with plot
mvn(hard_data, mvn_test = "mardia")
mvn(semi_data, mvn_test = "mardia")
mvn(pasta_data, mvn_test = "mardia")
mvn(soft_data, mvn_test = "mardia")

# Function to create Q-Q plot of Mahalanobis distances manually
qq_mahalanobis <- function(data, group_name) {
  data <- na.omit(data)
  if (nrow(data) <= ncol(data)) {
    warning(paste("Skipping", group_name, "- not enough observations"))
    return(NULL)
  }

  # Manually compute Mahalanobis distances
  center <- colMeans(data)
  cov_matrix <- cov(data)
  if (det(cov_matrix) == 0) {
    warning(paste("Skipping", group_name, "- singular covariance matrix"))
    return(NULL)
  }
  d2 <- mahalanobis(data, center, cov_matrix)
  theoretical <- qchisq(ppoints(length(d2)), df = ncol(data))

  # Create base R plot
  qqplot(theoretical, sort(d2),
         main = paste("Q-Q Plot:", group_name),
         xlab = "Theoretical Quantiles",
         ylab = "Observed Mahalanobis Distances")
  abline(0, 1, col = "red", lwd = 2)
}
# Layout for 2x2 plots
par(mfrow = c(2, 2))

# Run manually for each group
for (group in unique(cheese$texture)) {
  data_subset <- subset(cheese, texture == group)[, c("G80", "vLTmax", "vCO", "Fmax", "FD",
"FO")]
  qq_mahalanobis(data_subset, group)
}

# Homogeneity of Covariance Matrices
responses <- cheese[, c("G80", "vLTmax", "vCO", "Fmax", "FD", "FO")]
# Run Box's M test
boxM(responses, cheese$texture)

# Absence of Multicollinearity
# Correlation matrix
cor(responses)

# # Create correlation matrix
cor_matrix <- round(cor(responses, use = "complete.obs"), 3)  # Rounded to 3 decimals

# Print it nicely with spacing
print(cor_matrix)

# Start MANOVA
# Select thermophysical variables as response
response_vars <- c("G80", "vLTmax", "vCO", "Fmax", "FD", "FO")

# Create the MANOVA model
manova_model <- manova(as.matrix(cheese[, response_vars]) ~ texture, data = cheese)
manova_model

# View MANOVA summary using Wilks as the conditions for MANOVA have been met.
summary(manova_model, test = "Wilks")

# Run the multivariate model
manova_rrpp <- lm.rrpp(as.matrix(cheese[, c("G80", "vLTmax", "vCO", "Fmax", "FD", "FO")]) ~
texture,
                       data = cheese, iter = 999)

# Run pairwise MANOVA comparisons
pairwise_results <- pairwise(manova_rrpp, groups = cheese$texture)

# Summarize results
summary(pairwise_results)
```

D.2 Program for Question 2 (R code):

~~~
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```{r}
library(tidyverse)
library(mvtnorm) #used for randomly generating data from multivariate normal
library(factoextra)
library(gridExtra)
library(corrplot)
library(knitr)
### Scale the quantitative Variables

```{r}
ProjQuant <-
  Proj %>%
  select(G80, vLTmax, vCO, Fmax, FD, FO)

PQScale <- as.data.frame(scale(x = ProjQuant, center = TRUE, scale = TRUE))

```
### Check Corrrelation and Outliers
```{r}
cor(PQScale)
corrplot(cor(PQScale))

PQScale_long <- PQScale %>%
  pivot_longer(cols = everything(), names_to = "Variable", values_to = "Value")

ggplot(PQScale_long, aes(x = Variable, y = Value)) +
  geom_boxplot(fill = "yellow") +
  labs(title = "Boxplots of Scaled Thermophysical Variables")

#Finding the outliers
# Example: find outliers in FO
boxplot.stats(PQScale$FO)$out

# Find which rows they belong to
which(PQScale$FO %in% boxplot.stats(PQScale$FO)$out)

```
~~~

~~~~
### Create elbow plot
```{r}
set.seed(123)
fviz_nbclust(x = PQScale,
             FUNcluster = kmeans,
             method = "wss",
             k.max = 10,
             nstart = 25,
             iter.max = 25) +
  labs(subtitle = "K-means")
```

### Part c. - Perform K-means with K = 3 and extract Total Within Cluster Variation

```{r}
set.seed(123)
kmeansPQScale <- kmeans(x = PQScale, centers = 3, iter.max = 25, nstart = 15)
set.seed(NULL)

kmeansPQScale$tot.withinss
```

Based on kmeans with 3 clusters, the following is the total within-cluster variation:
250.4097

### How many in each cluster?

```{r}
table(kmeansPQScale$cluster)
```

Based on kmeans with 5 clusters, the followoing is the number of observations assigned with
each cluster:

### Part e. - Perform PCA and create visualization for K = 3 (K-means)
```{r}
#Using prcomp to use and standardize the data
ProjPCA <- prcomp(x = PQScale, center = TRUE, scale. = TRUE)

#Calculate PVE (used in the next steps)
PVE <- ProjPCA$sdev^2 / sum(ProjPCA$sdev^2)

#Create the df for the plots
Projtemp_df <- as.data.frame(x = ProjPCA$x[ , 1:2])

#Add cluster assignment to the dataset
Projtemp_df <-
  Projtemp_df %>%
  mutate(cluster = kmeansPQScale$cluster)

#Create Plot
ggplot(data = Projtemp_df, mapping = aes(x = PC1, y = PC2,
~~~~

~~~
                                  color = as.factor(cluster),
                                  shape = as.factor(cluster))) +
  geom_point() +
  labs(x = paste("PC1 (", round(100*PVE[1], digits = 1), "%)", sep = ""),
       y = paste("PC2 (", round(100*PVE[2], digits = 1), "%)", sep = ""),
       color = "Cluster Assignment",
       shape = "Cluster Assignment")
```


## Heirarchical Clustering
### Part a - Perform hierarchical clustering using Euclidean distance and complete linkage
```{r}
eucDistProj <- stats::dist(x = PQScale, method = "euclidean")
hcComp <- hclust(d = eucDistProj, method = "complete")

fviz_nbclust(x = PQScale, FUNcluster = hcut,
             method = "wss",
             k.max = 10,
             hc_func = "hclust",
             hc_metric = "euclidean",
             hc_method = "complete") +
  labs(subtitle = "Complete Linkage")
```


## Heirarchical clustering
```{r}
  fviz_dend(x = hcComp, k = 3, rect = TRUE) +
  labs(subtitle = "Complete Linkage K = 3") +
  theme(legend.position = "none")
```

### Part d. - Perform PCA and create visualization for K = 3 (Hierarchical)
```{r}
#Extract cluster assignments
hcCompClust <- cutree(tree = hcComp, k = 3)

hcCompClustdf <- as.data.frame(hcCompClust)


#Using prcomp to use and standardize the data
ProjPCA <- prcomp(x = PQScale, center = TRUE, scale. = TRUE)

#Calculate PVE (used in the next steps)
PVE <- ProjPCA$sdev^2 / sum(ProjPCA$sdev^2)

#Create the df for the plots
Projtemp_dfQ3 <- as.data.frame(x = ProjPCA$x[ , 1:2])

#Add cluster assignment to the dataset
Projtemp_dfQ3 <-
  Projtemp_dfQ3 %>%
~~~

~~~
  mutate(cluster = hcCompClustdf$hcCompClust)

#Create Plot
ggplot(data = Projtemp_dfQ3, mapping = aes(x = PC1, y = PC2,
                                  color = as.factor(cluster),
                                  shape = as.factor(cluster))) +
  geom_point() +
  labs(x = paste("PC1 (", round(100*PVE[1], digits = 1), "%)", sep = ""),
       y = paste("PC2 (", round(100*PVE[2], digits = 1), "%)", sep = ""),
       color = "Cluster Assignment",
       shape = "Cluster Assignment")
```

### Add the cluster assignments to the dataset
```{r}
# Add k-means cluster assignment to original dataset
Proj_with_kmeans <-
  Proj %>%
  mutate(kmeans_cluster = kmeansPQScale$cluster)
# Add hierarchical cluster assignment to original dataset
Proj_with_clusters <-
  Proj_with_kmeans %>%
  mutate(hierarchical_cluster = hcCompClust)
# View the first few rows with both cluster assignments

Proj_with_clusters <- Proj_with_clusters %>%
  mutate(kmeans_cluster_matched = case_when(
    kmeans_cluster == 3 ~ 1,
    kmeans_cluster == 1 ~ 3,
    TRUE ~ kmeans_cluster  # leaves 2 unchanged
  ))

Proj_with_clusters <- Proj_with_clusters %>%
  select(-kmeans_cluster)

Proj_with_clusters <- Proj_with_clusters %>%
  mutate(cluster_match = if_else(kmeans_cluster_matched == hierarchical_cluster, "Match",
"No Match"))


print(Proj_with_clusters)

cluster_comparison_df <- Proj_with_clusters %>%
  select(ID, kmeans_cluster_matched, hierarchical_cluster, cluster_match)

print(cluster_comparison_df)
```



```{r}
Proj_with_clusters %>%
  group_by(kmeans_cluster_matched) %>%
  summarise(across(c(G80, vLTmax, vCO, Fmax, FD, FO), mean, .names = "mean_{.col}"))
~~~

~~~~
```

From the clusters, we can identify the clusters have the following traits:
Cluster 1: High firmness and low flow
Cluster 2: Moderate Firmness, lower oil, and meltability
Cluster 3: Soft and oily, high flow.
~~~~

D.3 Program for Question 3 (R code):

```
# cheese data Q3: discriminant analysis
# Load necessary libraries
library(MASS)       # For lda()
library(caret)      # For data splitting and evaluation
library(ggpubr)     # For normality checks (Shapiro)
library(biotools)   # For Box's M test

# data input & EDA
cheese_data = read.csv("cheeseThermophysical.csv")
head(cheese_data)
table(cheese_data$texture)# mild unbalanced
table(cheese_data$texture)/length(cheese_data$ID)# prior prob.

# Convert target to factor if not already
cheese_data$texture <- as.factor(cheese_data$texture)
# str(cheese_data)
# 'data.frame':  89 obs. of  7 variables:
#  $ texture  : Factor w/ 4 levels
#  $ G80   : num
#  $ vLTmax: num
#  $ vCO   : num
#  $ Fmax  : num
#  $ FD    : num
#  $ FO    : num

# ----------------------------------------
# 1. Check Assumptions
# ----------------------------------------

# 1.1 Normality check for each feature within each group
features <- c("G80", "vLTmax", "vCO", "Fmax", "FD", "FO")
for (f in features) {
  cat("\nShapiro test for", f, "by group:\n")
  print(by(cheese_data[[f]], cheese_data$texture, shapiro.test))
}
# all p-value>0.05, cannot reject H0(normality), conclude that data statisfies normality
assumption.
# G80-Pasta Filata, vLTmax-Soft, VCO-Pasta Filata <0.05

# 1.2 Box's M test for equality of covariance matrices
boxM_result <- boxM(cheese_data[, features], cheese_data$texture)
print(boxM_result)
# H0: the observed covariance matrices for the dependent variables are equal across groups
```

```
# all p-value>0.05, cannot reject H0, conclude that covariance matrices are equal

# ----------------------------------------
# 2. Train-Test Split
# ----------------------------------------
set.seed(615)
train_index <- createDataPartition(cheese_data$texture, p = 0.7, list = FALSE)
train_data <- cheese_data[train_index, ]
test_data  <- cheese_data[-train_index, ]
head(train_data)
table(train_data$texture)
head(test_data)
table(test_data$texture)

# ----------------------------------------
# 3. Fit LDA Model
# ----------------------------------------
lda_model <- lda(texture ~ G80+vLTmax+vCO+Fmax+FD+FO, data = train_data)

# Model summary
print(lda_model)

# extract parameters
mu <- lda_model$means                    # group mean
pi_k <- lda_model$prior                  # prior prob.
Sigma <- lda_model$scaling               # LDA
cov_pooled <- lda_model$svd             # MASS::lda

# linear discriminant functions needs sum*mu
# here we compute by solve():
X <- train_data[, c("G80", "vLTmax", "vCO", "Fmax", "FD", "FO")]
Sigma_pool <- cov(X)                    #
inv_Sigma <- solve(Sigma_pool)          # sum

# discriminant function
coefs <- lapply(1:nrow(mu), function(i) {
  mu_k <- mu[i, ]
  a <- as.numeric(inv_Sigma %*% mu_k)
  b <- -0.5 * t(mu_k) %*% inv_Sigma %*% mu_k + log(pi_k[i])
  list(name = rownames(mu)[i], intercept = as.numeric(b), coefs = a)
})
coefs

feature_names <- colnames(mu)

for (group in coefs) {
  cat(paste0("\nDiscriminant function for group: ", group$name, "\n"))
  cat(sprintf("d_%s(x) = %.3f", group$name, group$intercept))
  for (j in seq_along(group$coefs)) {
    cat(sprintf(" + %.3f * %s", group$coefs[j], feature_names[j]))
  }
  cat("\n")
}

# ----------------------------------------
# 4. Predict on Test Set
# ----------------------------------------
lda_pred <- predict(lda_model, test_data)

# View first few predictions
head(lda_pred$class)

# levels
# str(test_data$texture)
levels(test_data$texture)

# ----------------------------------------
# 5. Evaluate the Model
# ----------------------------------------
# method 1: Resubstitution
# ----------------------------------------
conf_matrix <- confusionMatrix(lda_pred$class, test_data$texture)
print(conf_matrix)

# Optionally visualize
plot(lda_model)

# dataframe
cm_df <- as.data.frame(conf_matrix$table)
colnames(cm_df) <- c("Predicted", "Actual", "Freq")

# hot
ggplot(data = cm_df, aes(x = Actual, y = Predicted, fill = Freq)) +
  geom_tile(color = "white") +
  geom_text(aes(label = Freq), size = 6, color = "black") +
  scale_fill_gradient(low = "#e0f3f8", high = "#08589e") +
  labs(title = "Confusion Matrix (Test Set)", x = "Actual Texture", y = "Predicted
Texture") +
  theme_minimal(base_size = 14)

# ----------------------------------------
# method 2: 10-fold Cross-Validation
# ----------------------------------------
# factor levels renamed (delete space)
levels(cheese_data$texture) <- make.names(levels(cheese_data$texture))
levels(train_data$texture) <- make.names(levels(train_data$texture))

# new levels
levels(cheese_data$texture)
levels(train_data$texture)

set.seed(615)
train_control <- trainControl(
  method = "cv",
  number=10,
  savePredictions = "all",
  classProbs = TRUE)
lda_cv_model <- train(
  texture ~ G80+vLTmax+vCO+Fmax+FD+FO,
  data = train_data,
  method = "lda",
  trControl = train_control
)

# Model summary
print(lda_cv_model)

# Accuracy under cross-validation
cat("\nCross-Validated Accuracy:\n")
print(lda_cv_model$results)

# Confusion Matrix
predictions <- lda_cv_model$pred
true_levels <- levels(cheese_data$texture)
pred <- factor(predictions$pred, levels = true_levels)
obs  <- factor(predictions$obs,  levels = true_levels)
conf_matrix <- confusionMatrix(pred, obs)
print(conf_matrix)

library(reshape2)

# confusion Matrix data frame
conf_table <- as.data.frame(conf_matrix$table)

# heat map
ggplot(conf_table, aes(x = Reference, y = Prediction, fill = Freq)) +
  geom_tile(color = "white") +
  geom_text(aes(label = Freq), color = "black", size = 5) +
  scale_fill_gradient(low = "#F0F8FF", high = "#4682B4") +
  labs(title = "Confusion Matrix Heatmap", x = "Actual Texture", y = "Predicted Texture") +
  theme_minimal()
```