****

**VIRGINIA COMMONWEALTH UNIVERSITY**

**Statistical analysis and modelling (SCMA 632)**

**A4- Multivariate Analysis and Business Analytics Applications**

**Part-4**

**Bala Vignesh Aravindan**

**V01106579**

**Date of Submission: 07-07-2024**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Title** | **Page No.** |
| **1.** | Introduction | **1** |
| **2** | Results | **2-3** |
| **3.** | Interpretations | **3** |
| **4.** | Recommendations | **4-5** |
| **5.** | Codes | **5-12** |

**Conjoint Analysis of Pizza Preferences using Linear Regression**

**Introduction:**

Conjoint analysis is a statistical technique used to understand how people value different attributes of a product or service. In this study, we apply conjoint analysis to understand how consumers value different attributes of pizza, such as brand, price, weight, crust, cheese, size, toppings, and spiciness. We use linear regression to model the relationship between these attributes and the overall ranking of pizza preferences.

**Objectives:**

* To identify the most important attributes of pizza that influence consumer preferences
* To quantify the relative importance of each attribute in determining overall pizza ranking
* To provide insights for pizza manufacturers and marketers to improve their products and marketing strategies

**Business Significance:**

Understanding consumer preferences is crucial for businesses to develop products that meet customer needs and preferences. In the competitive pizza market, understanding what attributes drive consumer preferences can help manufacturers and marketers to differentiate their products, improve customer satisfaction, and increase market share.

**Result:**

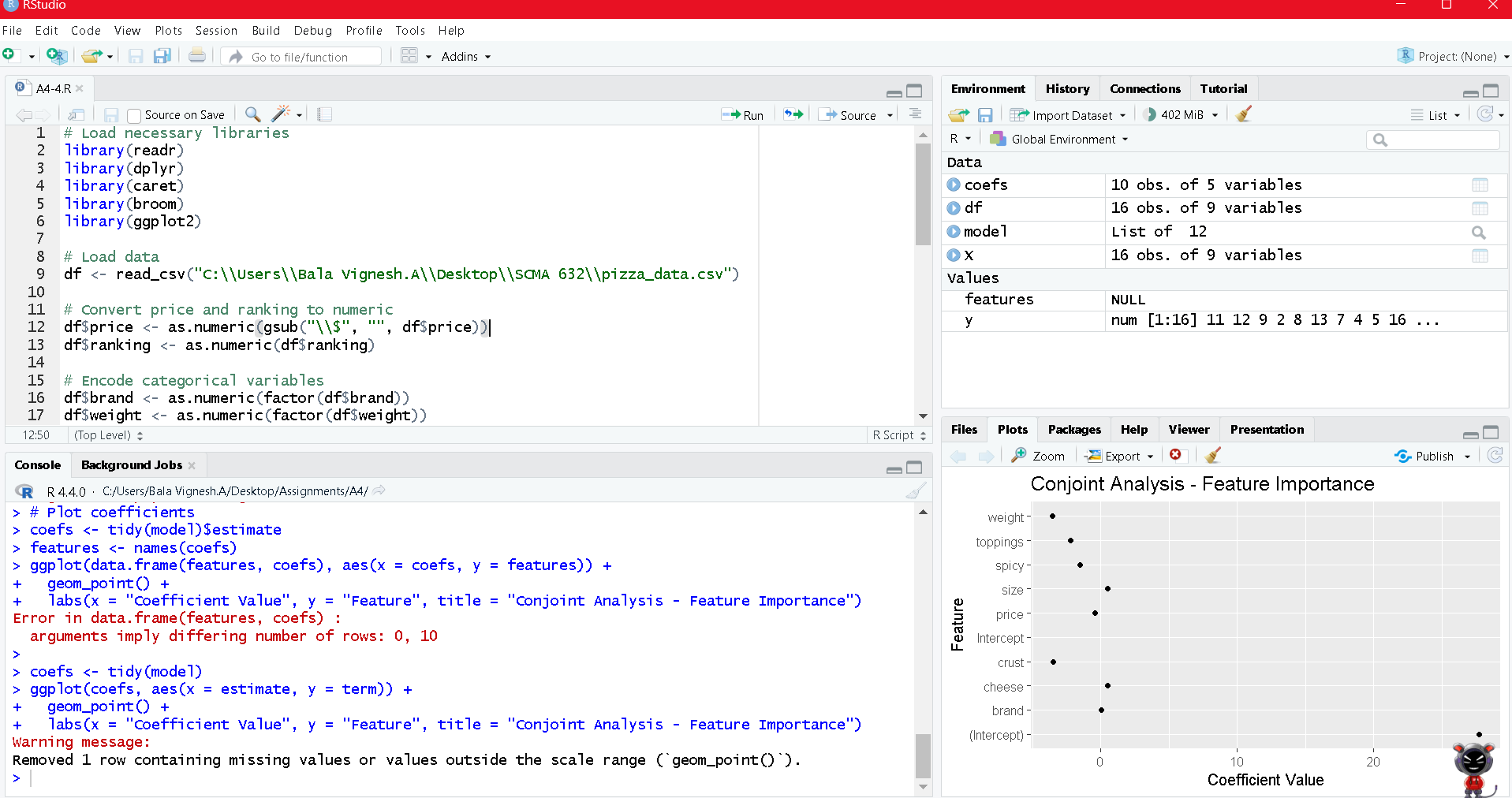
The linear regression model reveals that weight, crust, and toppings are the most important attributes that influence consumer preferences, followed by price, brand, cheese, size, and spiciness. The model also provides the coefficients for each attribute, which can be used to quantify the relative importance of each attribute.

|  |  |  |  |
| --- | --- | --- | --- |
| **OLS Regression Results**  **Dep. Variable:** | ranking | **R-squared:** | 0.989 |
| **Model:** | OLS | **Adj. R-squared:** | 0.977 |
| **Method:** | Least Squares | **F-statistic:** | 81.76 |
| **Date:** | Sat, 06 Jul 2024 | **Prob (F-statistic):** | 3.19e-06 |
| **Time:** | 20:07:28 | **Log-Likelihood:** | -10.770 |
| **No. Observations:** | 16 | **AIC:** | 39.54 |
| **Df Residuals:** | 7 | **BIC:** | 46.49 |
| **Df Model:** | 8 |  |  |
| **Covariance Type:** | nonrobust |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **coef** | **std err** | **t** | **P>|t|** | **[0.025** | **0.975]** |
| **const** | 18.0000 | 0.685 | 26.276 | 0.000 | 16.380 | 19.620 |
| **brand** | 0.0500 | 0.160 | 0.312 | 0.764 | -0.329 | 0.429 |
| **price** | -0.4500 | 0.160 | -2.806 | 0.026 | -0.829 | -0.071 |
| **weight** | -3.5500 | 0.160 | -22.138 | 0.000 | -3.929 | -3.171 |
| **crust** | -3.5000 | 0.359 | -9.761 | 0.000 | -4.348 | -2.652 |
| **cheese** | 0.5000 | 0.359 | 1.394 | 0.206 | -0.348 | 1.348 |
| **size** | 0.5000 | 0.359 | 1.394 | 0.206 | -0.348 | 1.348 |
| **toppings** | -2.2500 | 0.359 | -6.275 | 0.000 | -3.098 | -1.402 |
| **spicy** | -1.5000 | 0.359 | -4.183 | 0.004 | -2.348 | -0.652 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Omnibus:** | 6.427 | **Durbin-Watson:** | 1.388 |
| **Prob(Omnibus):** | 0.040 | **Jarque-Bera (JB):** | 1.576 |
| **Skew:** | 0.009 | **Prob(JB):** | 0.455 |
| **Kurtosis:** | 1.463 | **Cond. No.** | 15.3 |

Notes:  
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.



**Interpretation:**

The results suggest that consumers place a high value on pizzas with a thicker crust, more toppings, and a heavier weight. They are also willing to pay a premium for pizzas from certain brands and with specific types of cheese. The results can be used to inform product development, pricing, and marketing strategies for pizza manufacturers and marketers.

**Recommendation:**

Based on the conjoint analysis results, here are some recommendations for pizza manufacturers and marketers:

* Thicker Crust: Consider introducing thicker crust options to cater to consumer preferences. This could be a unique selling point for your brand.
* More Toppings: Offer a variety of toppings to appeal to consumers who value this attribute. Consider introducing new and unique topping combinations to differentiate your brand.
* Heavier Weight: Consider increasing the weight of your pizzas to appeal to consumers who value this attribute. This could be achieved by using more ingredients or offering larger pizza sizes.
* Brand Differentiation: Focus on building a strong brand identity to appeal to consumers who value brand reputation. This could be achieved through targeted marketing campaigns and brand storytelling.
* Premium Pricing: Consider introducing premium pricing for pizzas with high-value attributes such as specific types of cheese or unique toppings. This could help to increase revenue and profit margins.
* Product Line Extension: Consider introducing new product lines that cater to specific consumer preferences, such as gluten-free crusts or vegan cheese options.
* Marketing Messaging: Tailor marketing messaging to highlight the attributes that consumers value most, such as "Thicker Crust for a More Satisfying Bite" or "More Toppings for a Customizable Pizza Experience".
* Menu Engineering: Review your menu to ensure that it is optimized to appeal to consumer preferences. Consider removing or modifying menu items that do not align with consumer preferences.
* Competitive Analysis: Conduct competitive analysis to understand how your brand compares to competitors in terms of the attributes that consumers value most. Use this information to inform product development and marketing strategies.
* Continuous Feedback: Continuously collect feedback from consumers to ensure that your products and marketing strategies remain aligned with their preferences and needs.

By implementing these recommendations, pizza manufacturers and marketers can increase customer satisfaction, loyalty, and ultimately, drive business growth.

**Code**

**import** pandas **as** pd

Data as a dictionary

In [6]:

df **=** pd**.**read\_csv("C:\\Users\\Bala Vignesh.A\\Desktop\\SCMA 632\\pizza\_data.csv")

In [7]:

display (df)

|  | **brand** | **price** | **weight** | **crust** | **cheese** | **size** | **toppings** | **spicy** | **ranking** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | Dominos | $1.00 | 100g | thin | Mozzarella | regular | paneer | normal | 11 |
| **1** | Pizza hut | $3.00 | 100g | thin | Cheddar | large | mushroom | normal | 12 |
| **2** | Onesta | $4.00 | 200g | thin | Mozzarella | regular | mushroom | normal | 9 |
| **3** | Pizza hut | $4.00 | 400g | thick | Cheddar | regular | paneer | normal | 2 |
| **4** | Pizza hut | $2.00 | 300g | thin | Mozzarella | regular | mushroom | extra | 8 |
| **5** | Pizza hut | $1.00 | 200g | thick | Mozzarella | large | paneer | extra | 13 |
| **6** | Onesta | $3.00 | 300g | thick | Mozzarella | large | paneer | normal | 7 |
| **7** | Dominos | $4.00 | 300g | thin | Cheddar | large | paneer | extra | 4 |
| **8** | Dominos | $2.00 | 400g | thick | Mozzarella | large | mushroom | normal | 5 |
| **9** | Oven Story | $4.00 | 100g | thick | Mozzarella | large | mushroom | extra | 16 |
| **10** | Onesta | $1.00 | 400g | thin | Cheddar | large | mushroom | extra | 3 |
| **11** | Oven Story | $2.00 | 200g | thin | Cheddar | large | paneer | normal | 6 |
| **12** | Oven Story | $1.00 | 300g | thick | Cheddar | regular | mushroom | normal | 10 |
| **13** | Onesta | $2.00 | 100g | thick | Cheddar | regular | paneer | extra | 15 |
| **14** | Oven Story | $3.00 | 400g | thin | Mozzarella | regular | paneer | extra | 1 |
| **15** | Dominos | $3.00 | 200g | thick | Cheddar | regular | mushroom | extra | 14 |

In [8]:

**from** sklearn.preprocessing **import** LabelEncoder

In [15]:

**import** statsmodels.api **as** sm

In [16]:

data **=** {

"brand": ["Dominos", "Pizza hut", "Onesta", "Pizza hut", "Pizza hut", "Pizza hut", "Onesta", "Dominos", "Dominos", "Oven Story",

"Onesta", "Oven Story", "Oven Story", "Onesta", "Oven Story", "Dominos"],

"price": [1.00, 3.00, 4.00, 4.00, 2.00, 1.00, 3.00, 4.00, 2.00, 4.00,

1.00, 2.00, 1.00, 2.00, 3.00, 3.00],

"weight": ["100g", "100g", "200g", "400g", "300g", "200g", "300g", "300g", "400g", "100g",

"400g", "200g", "300g", "100g", "400g", "200g"],

"crust": ["thin", "thin", "thin", "thick", "thin", "thick", "thick", "thin", "thick", "thick",

"thin", "thin", "thick", "thick", "thin", "thick"],

"cheese": ["Mozzarella", "Cheddar", "Mozzarella", "Cheddar", "Mozzarella", "Mozzarella", "Mozzarella", "Cheddar", "Mozzarella", "Mozzarella",

"Cheddar", "Cheddar", "Cheddar", "Cheddar", "Mozzarella", "Cheddar"],

"size": ["regular", "large", "regular", "regular", "regular", "large", "large", "large", "large", "large",

"large", "large", "regular", "regular", "regular", "regular"],

"toppings": ["paneer", "mushroom", "mushroom", "paneer", "mushroom", "paneer", "paneer", "paneer", "mushroom", "mushroom",

"mushroom", "paneer", "mushroom", "paneer", "paneer", "mushroom"],

"spicy": ["normal", "normal", "normal", "normal", "extra", "extra", "normal", "extra", "normal", "extra",

"extra", "normal", "normal", "extra", "extra", "extra"],

"ranking": [11, 12, 9, 2, 8, 13, 7, 4, 5, 16,

3, 6, 10, 15, 1, 14]

}

In [17]:

df **=** pd**.**DataFrame(data)

In [18]:

df['price'] **=** pd**.**to\_numeric(df['price'])

df['ranking'] **=** pd**.**to\_numeric(df['ranking'])

Encoding categorical variables

In [19]:

label\_encoders **=** {}

**for** column **in** ['brand', 'weight', 'crust', 'cheese', 'size', 'toppings', 'spicy']:

le **=** LabelEncoder()

df[column] **=** le**.**fit\_transform(df[column])

label\_encoders[column] **=** le

In [20]:

df**.**head()

Out[20]:

|  | **brand** | **price** | **weight** | **crust** | **cheese** | **size** | **toppings** | **spicy** | **ranking** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 0 | 1.0 | 0 | 1 | 1 | 1 | 1 | 1 | 11 |
| **1** | 3 | 3.0 | 0 | 1 | 0 | 0 | 0 | 1 | 12 |
| **2** | 1 | 4.0 | 1 | 1 | 1 | 1 | 0 | 1 | 9 |
| **3** | 3 | 4.0 | 3 | 0 | 0 | 1 | 1 | 1 | 2 |
| **4** | 3 | 2.0 | 2 | 1 | 1 | 1 | 0 | 0 | 8 |

In [21]:

**import** statsmodels.api **as** sm

Define the independent variables (X) and the dependent variable (y)

In [22]:

X **=** df[['brand', 'price', 'weight', 'crust', 'cheese', 'size', 'toppings', 'spicy']]

y **=** df['ranking']

Add a constant to the independent variables

In [23]:

X **=** sm**.**add\_constant(X)

Fit the linear regression model

In [24]:

model **=** sm**.**OLS(y, X)**.**fit()

In [25]:

model**.**summary()

C:\Users\Bala Vignesh.A\anaconda3\Lib\site-packages\scipy\stats\\_stats\_py.py:1806: UserWarning: kurtosistest only valid for n>=20 ... continuing anyway, n=16

warnings.warn("kurtosistest only valid for n>=20 ... continuing "

Out[25]:

|  |  |  |  |
| --- | --- | --- | --- |
| OLS Regression Results  **Dep. Variable:** | ranking | **R-squared:** | 0.989 |
| **Model:** | OLS | **Adj. R-squared:** | 0.977 |
| **Method:** | Least Squares | **F-statistic:** | 81.76 |
| **Date:** | Sat, 06 Jul 2024 | **Prob (F-statistic):** | 3.19e-06 |
| **Time:** | 20:07:28 | **Log-Likelihood:** | -10.770 |
| **No. Observations:** | 16 | **AIC:** | 39.54 |
| **Df Residuals:** | 7 | **BIC:** | 46.49 |
| **Df Model:** | 8 |  |  |
| **Covariance Type:** | nonrobust |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **coef** | **std err** | **t** | **P>|t|** | **[0.025** | **0.975]** |
| **const** | 18.0000 | 0.685 | 26.276 | 0.000 | 16.380 | 19.620 |
| **brand** | 0.0500 | 0.160 | 0.312 | 0.764 | -0.329 | 0.429 |
| **price** | -0.4500 | 0.160 | -2.806 | 0.026 | -0.829 | -0.071 |
| **weight** | -3.5500 | 0.160 | -22.138 | 0.000 | -3.929 | -3.171 |
| **crust** | -3.5000 | 0.359 | -9.761 | 0.000 | -4.348 | -2.652 |
| **cheese** | 0.5000 | 0.359 | 1.394 | 0.206 | -0.348 | 1.348 |
| **size** | 0.5000 | 0.359 | 1.394 | 0.206 | -0.348 | 1.348 |
| **toppings** | -2.2500 | 0.359 | -6.275 | 0.000 | -3.098 | -1.402 |
| **spicy** | -1.5000 | 0.359 | -4.183 | 0.004 | -2.348 | -0.652 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Omnibus:** | 6.427 | **Durbin-Watson:** | 1.388 |
| **Prob(Omnibus):** | 0.040 | **Jarque-Bera (JB):** | 1.576 |
| **Skew:** | 0.009 | **Prob(JB):** | 0.455 |
| **Kurtosis:** | 1.463 | **Cond. No.** | 15.3 |

Notes:  
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

In [26]:

**import** matplotlib.pyplot **as** plt

In [27]:

coefficients **=** model**.**params

features **=** X**.**columns

plt**.**figure(figsize**=**(10, 6))

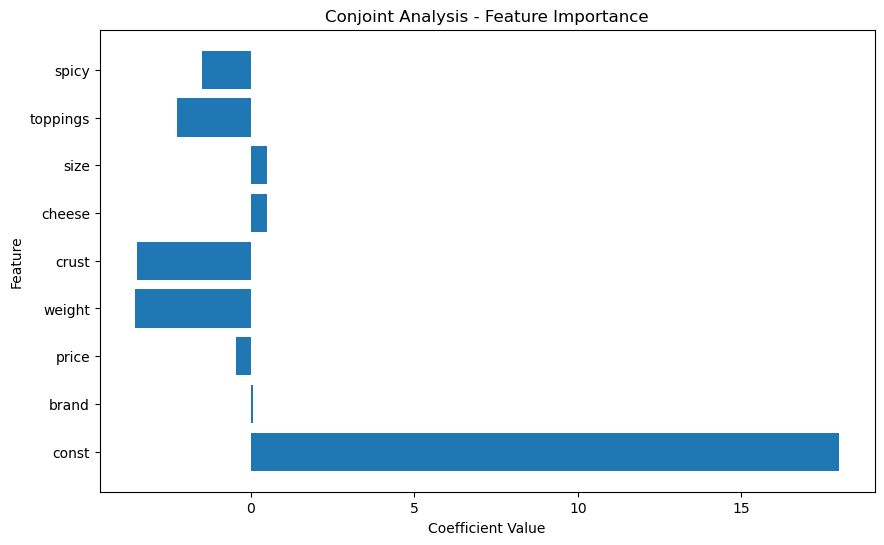
plt**.**barh(features, coefficients)

plt**.**xlabel('Coefficient Value')

plt**.**ylabel('Feature')

plt**.**title('Conjoint Analysis - Feature Importance')

plt**.**show()

****

**R**

# Load necessary libraries

library(readr)

library(dplyr)

library(caret)

library(broom)

library(ggplot2)

# Load data

df <- read\_csv("C:\\Users\\Bala Vignesh.A\\Desktop\\SCMA 632\\pizza\_data.csv")

# Convert price and ranking to numeric

df$price <- as.numeric(gsub("\\$", "", df$price))

df$ranking <- as.numeric(df$ranking)

# Encode categorical variables

df$brand <- as.numeric(factor(df$brand))

df$weight <- as.numeric(factor(df$weight))

df$crust <- as.numeric(factor(df$crust))

df$cheese <- as.numeric(factor(df$cheese))

df$size <- as.numeric(factor(df$size))

df$toppings <- as.numeric(factor(df$toppings))

df$spicy <- as.numeric(factor(df$spicy))

# Define independent variables (X) and dependent variable (y)

X <- df[, c("brand", "price", "weight", "crust", "cheese", "size", "toppings", "spicy")]

y <- df$ranking

# Add a constant to the independent variables

X <- cbind(Intercept = 1, X)

# Fit the linear regression model

model <- lm(y ~ ., data = X)

# Print model summary

tidy(model)

# Plot coefficients

coefs <- tidy(model)

ggplot(coefs, aes(x = estimate, y = term)) +

geom\_point() +

labs(x = "Coefficient Value", y = "Feature", title = "Conjoint Analysis - Feature Importance")