VIRGINIA COMMONWEALTH UNIVERSITY

Statistical analysis and modelling (SCMA 632)

A3b: Limited Dependent Variable Models:

Probit Regression Analysis

FERAH SHAN SHANAVAS RABIYA

V01101398

Date of Submission: 04-07-2024

CONTENTS

Introduction

The objective of this report is to identify factors associated with non-vegetarian dietary habits using data from the National Sample Survey Office (NSSO) 68th round. This analysis aims to understand the demographic and socio-economic characteristics that influence dietary preferences, specifically the likelihood of an individual being non-vegetarian.

To achieve this, a probit regression model is employed. The probit model is well-suited for binary outcome variables, providing a robust framework for estimating the probability of an event occurring based on several predictor variables. In this context, the event of interest is whether an individual is non-vegetarian.

Objectives

- Identify factors influencing non-vegetarian dietary habits
- Apply probit regression to model and analyze the probability of individuals being non-vegetarian
- Provide insights for public health and policy interventions
- Evaluate model performance and applicability
- Educate stakeholders and decision makers
- Contribute to scientific understanding
- Provide recommendations for future research and data collection

Business Significance

The report on identifying non-vegetarians using probit regression in the "NSSO68.csv" dataset offers significant business implications, particularly in health, nutrition, and consumer behavior sectors. It provides consumer insights and market segmentation, enabling businesses to tailor marketing strategies and develop products that appeal specifically to non-vegetarian consumers. Health professionals and nutritionists can use this information to offer personalized dietary advice and interventions, while public health policymakers can use it to develop targeted interventions aimed at improving nutritional outcomes and reducing health disparities. Strategic decision-making is possible through business strategy adjustments, risk management, and competitive advantage by aligning product offerings and marketing messages with non-vegetarian preferences. Tailoring products and services to meet specific consumer needs enhances customer satisfaction and loyalty, potentially increasing market share and profitability.

Research and development opportunities arise from understanding consumer preferences for non-vegetarian diets, inspiring innovation in food technology and developing new ingredients or formulations. Collaborations between research institutions, businesses, and public health entities can further explore emerging trends in dietary behaviors.

Ethical and social responsibility are also important aspects of the report, as understanding consumer preferences for non-vegetarian diets can inform sustainable practices in agriculture, food production, and resource management. Businesses can align CSR initiatives with health and nutrition goals, contributing positively to public health outcomes and community well-being. Overall, the report's significance lies in its ability to provide actionable insights into consumer behavior related to non-vegetarian dietary habits, enhancing market responsiveness, improving product offerings, and contributing to societal well-being through informed decision-making and strategic initiatives.

Results and Interpretation using R

- **Create a binary variable for non-vegetarian status using dplyr pipeline, selecting relevant variables for the probit model and handling missing values**

```
# Create a binary variable for non-vegetarian status using dplyr pipeline
> data <- data %>%
+   mutate(non_veg = case_when(
+     eggsno_q > 0 ~ 1,
+     fishprawn_q > 0 ~ 1,
+     goatmeat_q > 0 ~ 1,
+     beef_q > 0 ~ 1,
+     pork_q > 0 ~ 1,
+     chicken_q > 0 ~ 1,
+     othrbirds_q > 0 ~ 1,
+     TRUE ~ 0
+   ))
> # Select relevant variables for the probit model and handle missing valu
es
> data_clean <- data %>%
+   select(non_veg, Age, Sex, hhdsz, Religion, Education, MPCE_URP, state,
State_Region) %>%
+   filter_all(all_vars(!is.na(.)))
```

Interpretation:

By creating the non_veg variable, you transform categorical data on food consumption into a binary indicator suitable for modeling dietary habits. Selecting relevant variables and handlin g missing values ensures that only necessary and complete data are used for subsequent analy sis. The use of the dplyr pipeline (%>%) facilitates efficient data manipulation and ensures co de readability and reproducibility. After preparing the data as described, you can proceed with fitting a probit model using packages like glm or brglm. This model will help us to analyze th e factors influencing non-vegetarian dietary habits based on the selected variables.

- **Converting categorical variables to factors and fitting the probit regression mode l using the glm function**

```
# Convert categorical variables to factors
> data_clean <- data_clean %>%
+   mutate(
+     Sex = as.factor(Sex),
+     Religion = as.factor(Religion),
+     state = as.factor(state),
+     State_Region = as.factor(State_Region)
+   )
# Fit the probit regression model using the glm function
> probit_model <- glm(non_veg ~ Age + Sex + hhdsz + Religion + Educatio
n + MPCE_URP + state + State_Region,
+                      data = data_clean, family = binomial(link = "prob
it"))
# Summarize the model
```

```
> summary(probit_model)

Call:
glm(formula = non_veg ~ Age + Sex + hhdsz + Religion + Education + 
    MPCE_URP + state + State_Region, family = binomial(link = "probit")
, 
    data = data_clean)

Coefficients: (34 not defined because of singularities)
                  Estimate Std. Error z value Pr(>|z|)    
(Intercept)     -2.014e-02  5.345e-02  -0.377 0.706315    
Age             -4.831e-03  3.843e-04 -12.569  < 2e-16 ***
Sex2            -2.399e-01  1.623e-02 -14.780  < 2e-16 ***
hhdsz            7.275e-02  2.428e-03  29.968  < 2e-16 ***
Religion2        1.278e+00  2.139e-02  59.745  < 2e-16 ***
Religion3        5.379e-01  3.750e-02  14.342  < 2e-16 ***
Religion4       -7.938e-02  4.141e-02  -1.917 0.055227 .  
Religion5       -1.985e+00  1.450e-01 -13.691  < 2e-16 ***
Religion6        8.171e-01  6.216e-02  13.145  < 2e-16 ***
Religion7       -4.675e-01  7.802e-01  -0.599 0.549079    
Religion9        3.624e-01  8.478e-02   4.274 1.92e-05 ***
Education       -3.436e-02  1.450e-03 -23.702  < 2e-16 ***
MPCE_URP         3.411e-06  1.590e-06   2.145 0.031958 *  
state2           2.444e-01  6.288e-02   3.886 0.000102 ***
state3          -7.275e-01  6.359e-02 -11.441  < 2e-16 ***
state4          -2.332e-01  8.912e-02  -2.617 0.008875 ** 
state5           2.412e-01  5.704e-02   4.228 2.36e-05 ***
state6          -1.437e+00  8.070e-02 -17.803  < 2e-16 ***
state7           2.996e-02  6.392e-02   0.469 0.639273    
state8          -1.085e+00  7.148e-02 -15.181  < 2e-16 ***
state9          -5.993e-01  5.658e-02 -10.593  < 2e-16 ***
state10          3.362e-01  5.673e-02   5.927 3.08e-09 ***
state11          1.067e+00  7.711e-02  13.840  < 2e-16 ***
state12          1.706e+00  8.343e-02  20.455  < 2e-16 ***
state13          2.587e+00  2.334e-01  11.083  < 2e-16 ***
state14          1.473e+00  1.087e-01  13.554  < 2e-16 ***
state15          2.480e+00  1.732e-01  14.318  < 2e-16 ***
state16          2.179e+00  8.154e-02  26.720  < 2e-16 ***
state17          1.765e+00  1.007e-01  17.536  < 2e-16 ***
state18          1.934e+00  1.102e-01  17.546  < 2e-16 ***
state19          1.993e+00  8.729e-02  22.825  < 2e-16 ***
state20          5.885e-01  6.028e-02   9.763  < 2e-16 ***
state21          1.321e+00  6.704e-02  19.710  < 2e-16 ***
state22          6.486e-01  8.419e-02   7.704 1.32e-14 ***
state23         -6.435e-01  7.334e-02  -8.775  < 2e-16 ***
state24         -1.121e+00  7.212e-02 -15.540  < 2e-16 ***
state25          1.248e+00  1.505e-01   8.292  < 2e-16 ***
state26          1.416e-01  1.040e-01   1.362 0.173237    
state27          6.161e-01  7.809e-02   7.889 3.05e-15 ***
state28          1.055e+00  6.507e-02  16.208  < 2e-16 ***
state29         -7.753e-02  5.725e-02  -1.354 0.175618    
state30          1.445e+00  9.717e-02  14.869  < 2e-16 ***
state31          6.500e-01  1.611e-01   4.034 5.49e-05 ***
state32          1.468e+00  6.087e-02  24.113  < 2e-16 ***
state33          1.011e+00  5.980e-02  16.914  < 2e-16 ***
state34          1.376e+00  8.348e-02  16.477  < 2e-16 ***
state35          1.655e+00  9.827e-02  16.841  < 2e-16 ***
State_Region12  -5.617e-02  7.013e-02  -0.801 0.423194    
State_Region13   1.218e+00  1.060e-01  11.489  < 2e-16 ***
State_Region14   1.263e+00  2.763e-01   4.572 4.84e-06 ***
```

```
State_Region21  -2.533e-01  5.669e-02  -4.468 7.89e-06 ***
State_Region22          NA         NA      NA       NA
State_Region31   2.654e-01  5.112e-02   5.191 2.09e-07 ***
State_Region32          NA         NA      NA       NA
State_Region41          NA         NA      NA       NA
State_Region51          NA         NA      NA       NA
State_Region61   7.027e-01  7.396e-02   9.502  < 2e-16 ***
State_Region62          NA         NA      NA       NA
State_Region71          NA         NA      NA       NA
State_Region81  -2.246e-01  8.054e-02  -2.788 0.005297 **
State_Region82   1.821e-01  6.605e-02   2.757 0.005827 **
State_Region83   4.674e-01  8.365e-02   5.587 2.31e-08 ***
State_Region84   3.237e-01  8.299e-02   3.900 9.61e-05 ***
State_Region85          NA         NA      NA       NA
State_Region91   2.146e-01  4.788e-02   4.482 7.41e-06 ***
State_Region92   1.821e-01  4.729e-02   3.851 0.000117 ***
State_Region93   3.665e-01  3.700e-02   9.905  < 2e-16 ***
State_Region94   4.826e-01  6.152e-02   7.844 4.36e-15 ***
State_Region95          NA         NA      NA       NA
State_Region101  4.630e-01  4.158e-02  11.135  < 2e-16 ***
State_Region102         NA         NA      NA       NA
State_Region111         NA         NA      NA       NA
State_Region121         NA         NA      NA       NA
State_Region131         NA         NA      NA       NA
State_Region141  9.655e-01  1.281e-01   7.534 4.92e-14 ***
State_Region142         NA         NA      NA       NA
State_Region151         NA         NA      NA       NA
State_Region161         NA         NA      NA       NA
State_Region171         NA         NA      NA       NA
State_Region181 -1.643e-01  1.233e-01  -1.333 0.182630
State_Region182  5.136e-02  1.281e-01   0.401 0.688446
State_Region183 -1.294e-01  1.423e-01  -0.910 0.362876
State_Region184         NA         NA      NA       NA
State_Region191  1.529e-01  1.365e-01   1.120 0.262770
State_Region192  2.235e-02  1.043e-01   0.214 0.830275
State_Region193 -4.395e-01  8.618e-02  -5.100 3.39e-07 ***
State_Region194 -4.057e-01  9.002e-02  -4.507 6.56e-06 ***
State_Region195         NA         NA      NA       NA
State_Region201  1.666e-01  5.562e-02   2.995 0.002746 **
State_Region202         NA         NA      NA       NA
State_Region211  2.074e-01  6.710e-02   3.091 0.001996 **
State_Region212 -2.204e-01  6.158e-02  -3.579 0.000345 ***
State_Region213         NA         NA      NA       NA
State_Region221  5.665e-01  1.217e-01   4.657 3.21e-06 ***
State_Region222 -3.483e-01  7.650e-02  -4.553 5.28e-06 ***
State_Region223         NA         NA      NA       NA
State_Region231  4.890e-01  7.022e-02   6.964 3.31e-12 ***
State_Region232  6.899e-02  7.628e-02   0.905 0.365707
State_Region233  1.990e-01  6.907e-02   2.881 0.003967 **
State_Region234  5.052e-01  7.142e-02   7.073 1.51e-12 ***
State_Region235  6.281e-01  7.759e-02   8.096 5.69e-16 ***
State_Region236         NA         NA      NA       NA
State_Region241  8.386e-01  6.565e-02  12.773  < 2e-16 ***
State_Region242  2.406e-01  7.264e-02   3.313 0.000925 ***
State_Region243  2.411e-01  1.113e-01   2.166 0.030322 *
State_Region244  7.419e-02  1.652e-01   0.449 0.653290
State_Region245         NA         NA      NA       NA
State_Region251         NA         NA      NA       NA
State_Region261         NA         NA      NA       NA
State_Region271  5.074e-02  6.919e-02   0.733 0.463298
```

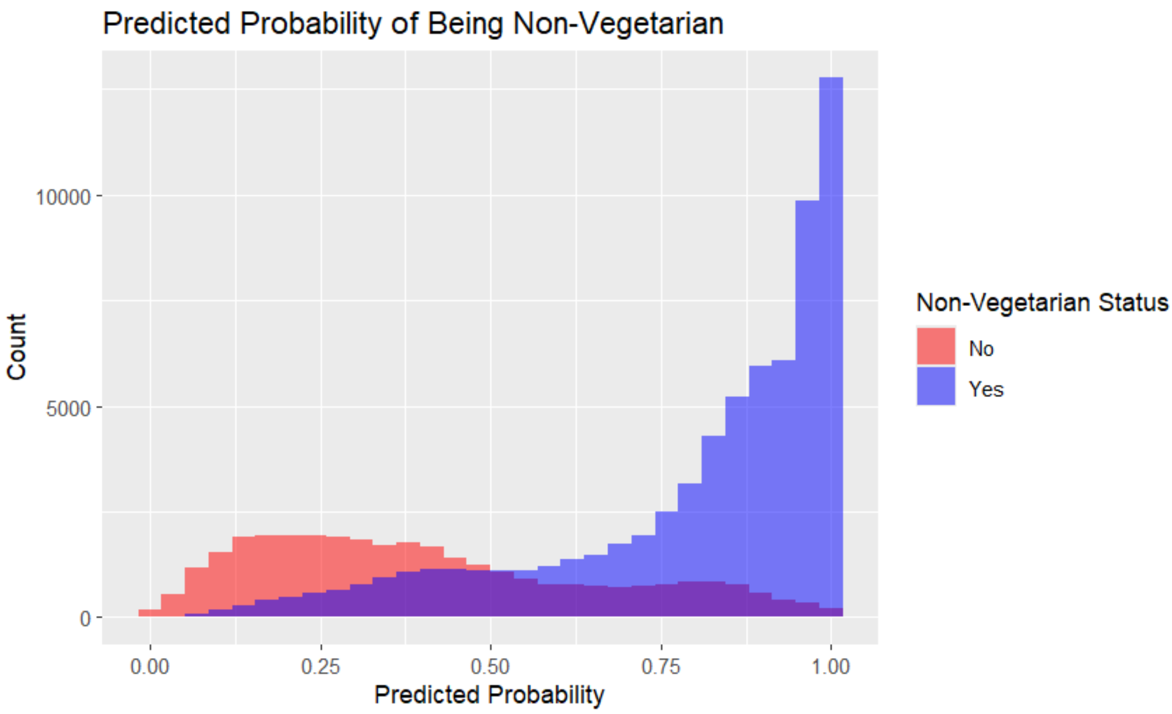
```
State_Region272 -1.506e-01  6.843e-02  -2.201 0.027721 *
State_Region273 -5.623e-01  7.456e-02  -7.541 4.65e-14 ***
State_Region274 -7.695e-01  7.119e-02 -10.809  < 2e-16 ***
State_Region275 -5.138e-01  7.121e-02  -7.215 5.38e-13 ***
State_Region276         NA         NA      NA       NA
State_Region281  3.281e-01  6.164e-02   5.323 1.02e-07 ***
State_Region282  1.084e-01  6.213e-02   1.745 0.081044 .
State_Region283  2.982e-01  5.974e-02   4.993 5.96e-07 ***
State_Region284  6.408e-01  7.566e-02   8.469  < 2e-16 ***
State_Region285         NA         NA      NA       NA
State_Region291  9.428e-01  8.122e-02  11.608  < 2e-16 ***
State_Region292  1.203e+00  8.058e-02  14.927  < 2e-16 ***
State_Region293  7.774e-01  4.751e-02  16.363  < 2e-16 ***
State_Region294         NA         NA      NA       NA
State_Region301         NA         NA      NA       NA
State_Region311         NA         NA      NA       NA
State_Region321  8.915e-02  6.154e-02   1.449 0.147432
State_Region322         NA         NA      NA       NA
State_Region331  1.110e-01  4.908e-02   2.262 0.023711 *
State_Region332  7.012e-02  5.534e-02   1.267 0.205176
State_Region333  3.248e-01  5.302e-02   6.125 9.08e-10 ***
State_Region334         NA         NA      NA       NA
State_Region341         NA         NA      NA       NA
State_Region351         NA         NA      NA       NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 128251  on 101651  degrees of freedom
Residual deviance:  83536  on 101552  degrees of freedom
AIC: 83736

Number of Fisher Scoring iterations: 7
```

Interpretation:

The output from the probit regression model provides valuable insights into the factors influe ncing non-vegetarian dietary status. The intercept represents the estimated log-odds of being n on-vegetarian when all other predictors are zero. In this case, it's not statistically significant (p = 0.706), indicating no evidence of a non-vegetarian baseline when other factors are controlle d. For every one unit increase in age, the log-odds of being non-vegetarian decrease by -0.004 8 (p < 0.001). This suggests that younger individuals are more likely to be non-vegetarian. Be ing male (coded as 2) decreases the log-odds of being non-vegetarian by -0.2399 (p < 0.001) c ompared to females (coded as 1). Each unit increase in household size increases the log-odds of being non-vegetarian by 0.0728 (p < 0.001). Higher education levels decrease the log-odds of being non-vegetarian by -0.0344 (p < 0.001). Each unit increase in MPCE_URP (presumab ly a measure of economic status) increases the log-odds of being non-vegetarian by 0.000003 41 (p = 0.032). State and State_Region represent different geographic regions. Each level indi cates how being in that state or region affects the log-odds of being non-vegetarian compared to a reference state or region. The deviance goodness-of-fit test compares the model with a model that has no predictors (intercept-only model). A lower null deviance indicates better model fit. Residual deviance compares the fitted model with the saturated model (perfect fit). A lower residual deviance suggests a better fit of the model. AIC is used for model selection. Lower AIC values indicate a better trade-off between model complexity and goodness of fit. This probit regression model provides a comprehensive understanding of how demographic, socio-economic, and geographic factors influence the likelihood of being non-vegetarian. It identifies significant predictors such as age, sex, household size, religion, education, and economic status (MPCE_URP), highlighting their roles in shaping dietary habits. Businesses, policymakers, and health professionals can use these insights to target interventions, develop tailored strategies, and promote healthier dietary behaviors among different demographic groups.

- **Make predictions and visualize the results**

```
# Make predictions
> data_clean <- data_clean %>%
+   mutate(predicted_prob = predict(probit_model, type = "response"))
> # Visualize the results
> ggplot(data_clean, aes(x = predicted_prob, fill = as.factor(non_veg))
) +
+   geom_histogram(position = "identity", alpha = 0.5, bins = 30) +
+   labs(title = "Predicted Probability of Being Non-Vegetarian", x = "
Predicted Probability", y = "Count") +
+   scale_fill_manual(values = c("1" = "blue", "0" = "red"), name = "No
n-Vegetarian Status", labels = c("No", "Yes"))
```



Interpretation:

The predicted probabilities represent the model's estimate of the likelihood of an individual be ing non-vegetarian based on their characteristics as captured by the model. The histogram bin s the predicted probabilities. The x-axis represents the predicted probability values. The y-axi s shows the count of individuals falling into each predicted probability bin. Non-vegetarians (non_veg = 1) are colored blue. Vegetarians (non_veg = 0) are colored red. According to the h istogram, most of them are non-vegetarians.

Results and Interpretation using Python

- **Fitting the probit regression model**

```
# Add a constant term for the intercept
# Define dependent variable (y) and independent variables (X)
y = df1['NV']
X = df1[['HH_type', 'Religion', 'Social_Group', 'Regular_salary_earner',
          'Possess_ration_card', 'Sex', 'Age', 'Marital_Status', 'Educatio
n',
          'Meals_At_Home', 'Region', 'hhdsz', 'NIC_2008', 'NCO_2004']]

# Assuming X is your DataFrame containing the independent variables
X['Social_Group'] = X['Social_Group'].astype('category')
X['Regular_salary_earner'] = X['Regular_salary_earner'].astype('category')
X['HH_type'] = X['HH_type'].astype('category')
X['Possess_ration_card'] = X['Possess_ration_card'].astype('category')
X['Sex'] = X['Sex'].astype('category')
X['Marital_Status'] = X['Marital_Status'].astype('category')
X['Education'] = X['Education'].astype('category')
X['Region'] = X['Region'].astype('category')

X= sm.add_constant(X)

# Fit the probit regression model
probit_model = Probit(y, X).fit()

# Print the summary of the model
print(probit_model.summary())
```

```
Optimization terminated successfully.
         Current function value: 0.589533
         Iterations 5
                          Probit Regression Results
==============================================================================
Dep. Variable:                     NV   No. Observations:                93096
Model:                         Probit   Df Residuals:                    93081
Method:                           MLE   Df Model:                           14
Date:            Wed, 03 Jul 2024   Pseudo R-squ.:                 0.05196
Time:                    23:36:13   Log-Likelihood:                -54883.
converged:                       True   LL-Null:                       -57891.
Covariance Type:            nonrobust   LLR p-value:                     0.000
==============================================================================
```

	coef	std err	z	P>|z|	[0.025	0.975]
const	0.0501	0.056	0.902	0.367	-0.059	0.159
HH_type	0.0174	0.004	4.677	0.000	0.010	0.025
Religion	0.1878	0.005	37.169	0.000	0.178	0.198
Social_Group	0.0464	0.001	-32.205	0.000	-0.049	-0.044
Regular_salary_earner	-0.0321	0.011	-2.904	0.004	-0.054	-0.010
Possess_ration_card	0.0222	0.012	1.897	0.058	-0.001	0.045
Sex	-0.0262	0.020	-1.305	0.192	-0.065	0.013
Age	-0.0020	0.000	-5.265	0.000	-0.003	-0.001
Marital_Status	-0.0228	0.016	-1.438	0.150	-0.054	0.008
Educati	-0.0127	0.001	-8.534	0.000	-0.016	-0.010
Meals_At_Home	0.0103	0.000	36.460	0.000	0.010	0.011
Region	-0.0789	0.003	-23.916	0.000	-0.085	-0.072
hhdsz	-0.0070	0.002	-3.342	0.001	-0.011	-0.003
NIC_2008	2.4e-06	1.81e-07	13.247	0.000	2.05e-06	2.76e-06
NCO_2004	6.919e-05	2.17e-05	3.196	0.001	2.68e-05	0.000

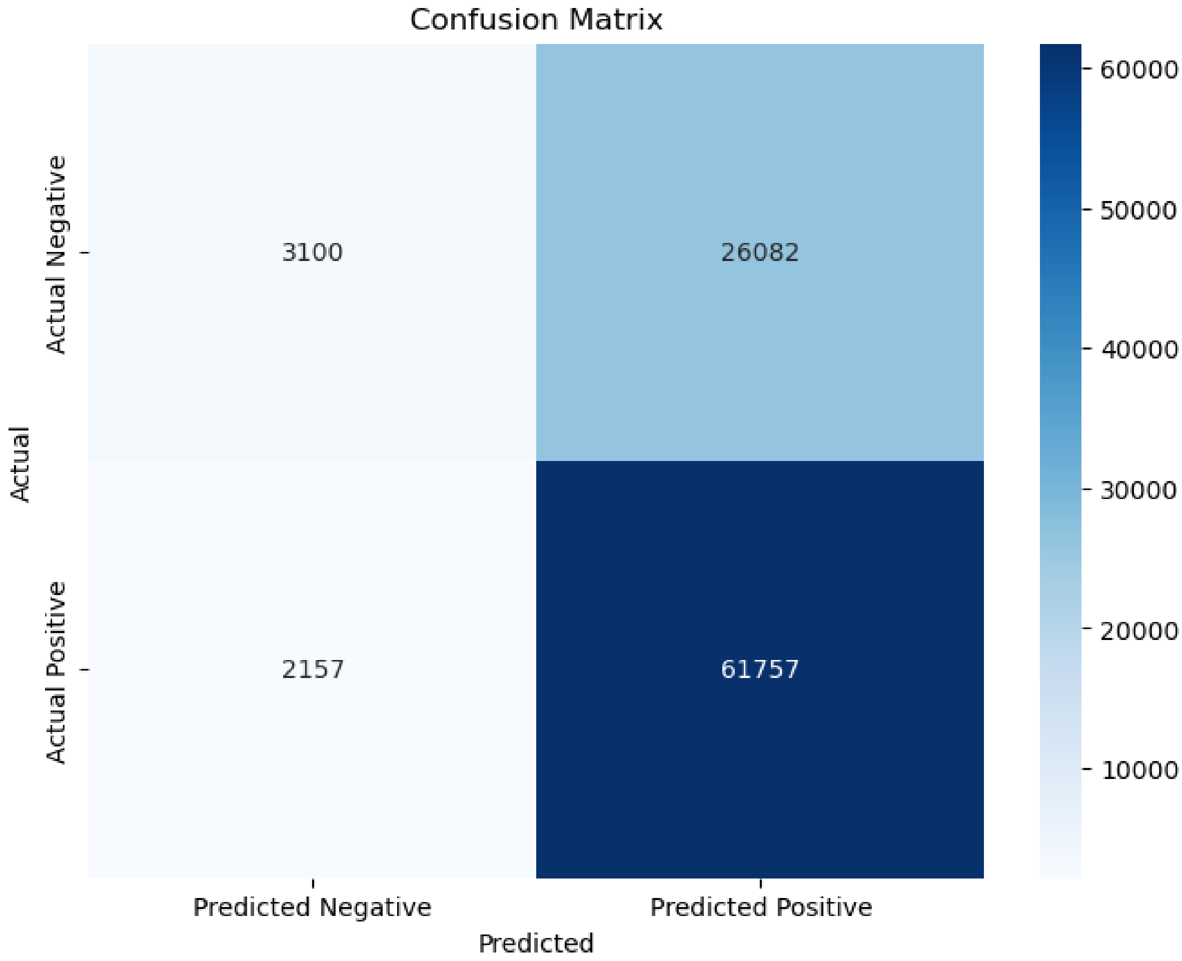
Interpretation:

The study analyzed 93,096 observations and used a Probit regression model to predict non-vegetarian status. The model fit was -54,883, with a log-likelihood of -57,891 and a pseudo-R-squared of 0.05196. The coefficients represented the estimated effect of an independent variable on the probability of being non-vegetarian, holding all other variables constant. The baseline probability of being non-vegetarian when all independent variables were zero was 0.0501. HH_type, Religion, Social_Group, Age, Education, Meals_At_Home, Region, hhdsz, NIC_2008, and NCO_2004 had p-values less than 0.05, suggesting they have a significant impact on the likelihood of being non-vegetarian. Regular_salary_earner also significantly affected the likelihood, albeit more marginally. An example of the results is that an increase in age by one year decreases the log odds of being non-vegetarian by 0.0020, all else being equal. Individuals in certain social groups (represented by Social_Group) have a higher log odds of being non-vegetarian by 0.0464 compared to those in the reference group. The results provide insights into the factors influencing non-vegetarian status and help quantify the direction and strength of these relationships, aiding in understanding the determinants of dietary preferences within the studied population.

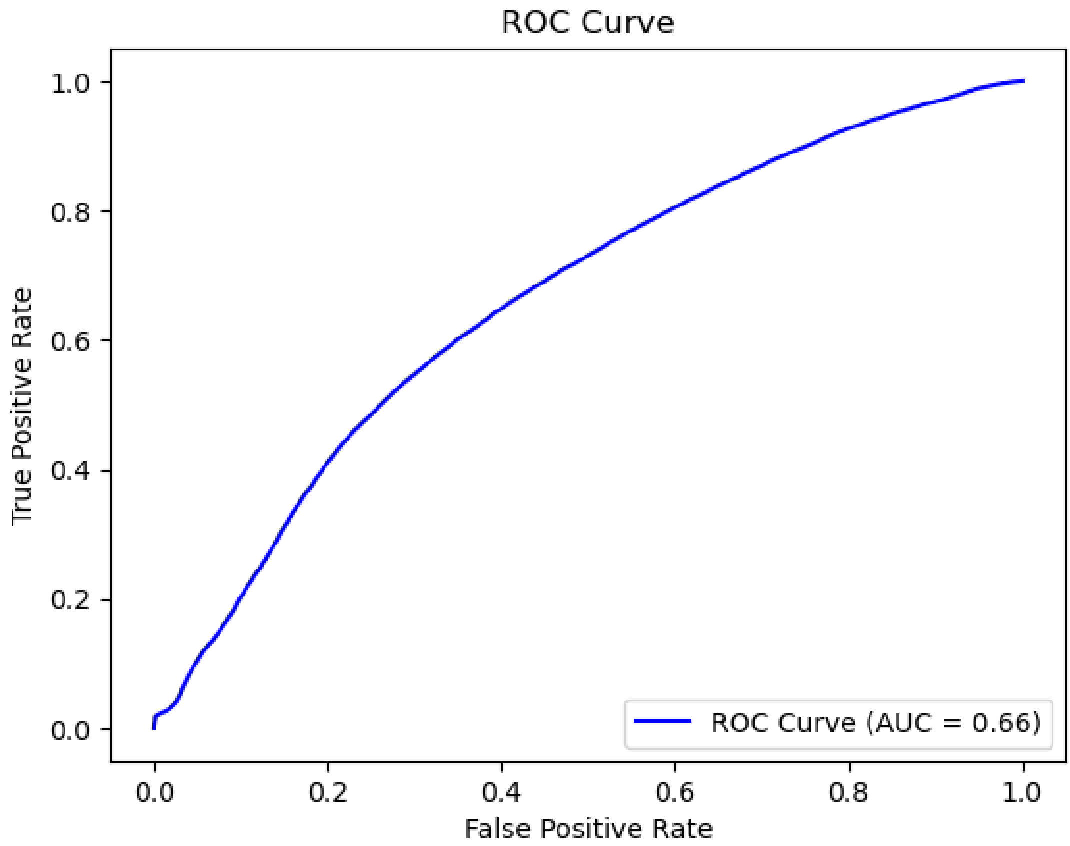
- Printing confusion matrix and ROC curve for Logistic Regression

```
# Predict probabilities
predicted_probs = probit_model.predict(X)

# Convert probabilities to binary predictions using a threshold of 0.5
predicted_classes = (predicted_probs > 0.5).astype(int)

# Confusion Matrix
conf_matrix = confusion_matrix(y, predicted_classes)
conf_matrix_df = pd.DataFrame(conf_matrix, index=['Actual Negative', 'Actual
Positive'], columns=['Predicted Negative', 'Predicted Positive'])
print("Confusion Matrix:\n", conf_matrix_df)

# Plotting the Confusion Matrix
plt.figure(figsize=(8, 6))
sns.heatmap(conf_matrix_df, annot=True, fmt='d', cmap='Blues')
plt.ylabel('Actual')
plt.xlabel('Predicted')
plt.title('Confusion Matrix')
plt.show()
```

```
Confusion Matrix:
                  Predicted Negative  Predicted Positive
Actual Negative                 3100               26082
Actual Positive                 2157               61757
```



```
# ROC curve and AUC value
fpr, tpr, _ = roc_curve(y, predicted_probs)
auc_value = roc_auc_score(y, predicted_probs)
plt.plot(fpr, tpr, color='blue', label=f'ROC Curve (AUC = {auc_value:.2f})')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve')
plt.legend(loc='lower right')
plt.show()
print(f"AUC: {auc_value}")

# Accuracy, Precision, Recall, F1 Score
accuracy = accuracy_score(y, predicted_classes)
precision = precision_score(y, predicted_classes)
recall = recall_score(y, predicted_classes)
f1 = f1_score(y, predicted_classes)

print(f"Accuracy: {accuracy}")
print(f"Precision: {precision}")
print(f"Recall: {recall}")
print(f"F1 Score: {f1}")
```




```
AUC: 0.6624909652546589
Accuracy: 0.6966679556586749
Precision: 0.7030703901456073
Recall: 0.9662515254873737
F1 Score: 0.8139147166777593
```

Interpretation:

The probit regression model's performance was evaluated using a confusion matrix and evalu ation metrics. The model showed a moderate level of discrimination ability in distinguishing b etween non-vegetarians and vegetarians, with an AUC of 0.66 indicating a moderate level of a ccuracy. The model's accuracy was 0.697, with 69.7% of predictions correct. Precision was 0. 703, with a 70.3% accuracy rate when predicting non-vegetarian status. Recall was 0.966, ind icating 96.6% of all actual non-vegetarians correctly identified. The F1 score was 0.814, indic ating a good balance between precision and recall. The model showed a relatively good abilit y to predict non-vegetarian status based on the provided features, with high recall and reasona ble precision. However, the moderate AUC suggests room for improvement in the model's dis criminatory power. The confusion matrix provided a detailed breakdown of how well the mod el performs in correctly classifying non-vegetarians and vegetarians. Adjustments or enhance ments to the model could focus on improving AUC and overall predictive accuracy.

Recommendation

The probit regression model for predicting non-vegetarian status has been analyzed, revealing its strengths in high recall and moderate AUC. However, the model's overall discriminatory power could be improved. The model correctly predicts 69.7% of cases, with a 70.3% accuracy rate when predicting non-vegetarian status. It captures 96.6% of all actual non-vegetarians. The F1 score provides a balanced assessment of precision and recall, indicating good overall performance.

The confusion matrix analysis reveals the distribution of true positives, false positives, false negatives, and false negatives, providing insights into the model's strengths and weaknesses. The ROC curve and AUC are discussed, with 0.66 indicating moderate discrimination ability. Recommendations for improvement include exploring additional features or refining existing features to enhance the model's discriminatory power. Different thresholds for binary classification can be evaluated to optimize the balance between precision and recall. Potential strategies for model refinement include feature engineering, regularization techniques, or exploring different modeling algorithms like random forests or gradient boosting.
In conclusion, the report summarizes the findings and emphasizes the practical implications of the model's performance in predicting non-vegetarian status. It concludes with a statement on the model's current utility and potential future directions for improving its predictive accuracy and reliability.

R Codes

```
# Load necessary libraries
library(readr)
library(dplyr)
library(ggplot2)
library(magrittr)

# Read the dataset
data <- read_csv("E:\\VCU\\Summer 2024\\Statistical Analysis & Modeling\\NSSO68.csv")

# Create a binary variable for non-vegetarian status using dplyr pipeline
data <- data %>%
  mutate(non_veg = case_when(
    eggsno_q > 0 ~ 1,
    fishprawn_q > 0 ~ 1,
    goatmeat_q > 0 ~ 1,
    beef_q > 0 ~ 1,
    pork_q > 0 ~ 1,
    chicken_q > 0 ~ 1,
    othrbirds_q > 0 ~ 1,
    TRUE ~ 0
  ))

# Select relevant variables for the probit model and handle missing values
data_clean <- data %>%
  select(non_veg, Age, Sex, hhdsz, Religion, Education, MPCE_URP, state, State_Region) %
>%
  filter_all(all_vars(!is.na(.)))

# Convert categorical variables to factors
data_clean <- data_clean %>%
  mutate(
    Sex = as.factor(Sex),
```

```
  Religion = as.factor(Religion),
  state = as.factor(state),
  State_Region = as.factor(State_Region)
 )

# Fit the probit regression model using the glm function
probit_model <- glm(non_veg ~ Age + Sex + hhdsz + Religion + Education + MPCE_URP +
state + State_Region,
                    data = data_clean, family = binomial(link = "probit"))

# Summarize the model
summary(probit_model)

# Make predictions
data_clean <- data_clean %>%
  mutate(predicted_prob = predict(probit_model, type = "response"))

# Visualize the results
ggplot(data_clean, aes(x = predicted_prob, fill = as.factor(non_veg))) +
  geom_histogram(position = "identity", alpha = 0.5, bins = 30) +
  labs(title = "Predicted Probability of Being Non-Vegetarian", x = "Predicted Probability", y
= "Count") +
  scale_fill_manual(values = c("1" = "blue", "0" = "red"), name = "Non-Vegetarian Status", la
bels = c("No", "Yes"))

# Save the plot
ggsave("predicted_probabilities.png", width = 8, height = 6)
```

Python Codes

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
from statsmodels.discrete.discrete_model import Probit
from sklearn.metrics import confusion_matrix, roc_curve, roc_auc_score, accuracy_score, pr
ecision_score, recall_score, f1_score
import matplotlib.pyplot as plt
import seaborn as sns
import os

os.chdir("C:\\Users\\Ferah Shan\\Downloads")
# Load the dataset
data = pd.read_csv('NSSO68.csv', encoding='Latin-1', low_memory=False)

# Display basic information about the dataset
print(data.info())
# Display first few rows to understand the data
print(data.head())
list(data.columns)

# Create a new feature called NV
data['NV'] = data[['eggsno_q', 'fishprawn_q', 'goatmeat_q', 'beef_q', 'pork_q', 'chicken_q', 'oth
rbirds_q']].sum(axis=1).apply(lambda x: 1 if x > 0 else 0)
data.shape
df= data.copy()
df.dropna(how= 'all',inplace=True)
df1 = df[['NV','HH_type', 'Religion', 'Social_Group', 'Regular_salary_earner',
        'Possess_ration_card', 'Sex', 'Age', 'Marital_Status', 'Education',
        'Meals_At_Home', 'Region', 'hhdsz', 'NIC_2008', 'NCO_2004']]
df1.dropna(how='any',inplace=True)
df1
```

```
# Add a constant term for the intercept
# Define dependent variable (y) and independent variables (X)
y = df1['NV']
X = df1[['HH_type', 'Religion', 'Social_Group', 'Regular_salary_earner',
        'Possess_ration_card', 'Sex', 'Age', 'Marital_Status', 'Education',
        'Meals_At_Home', 'Region', 'hhdsz', 'NIC_2008', 'NCO_2004']]

# Assuming X is your DataFrame containing the independent variables
X['Social_Group'] = X['Social_Group'].astype('category')
X['Regular_salary_earner'] = X['Regular_salary_earner'].astype('category')
X['HH_type'] = X['HH_type'].astype('category')
X['Possess_ration_card'] = X['Possess_ration_card'].astype('category')
X['Sex'] = X['Sex'].astype('category')
X['Marital_Status'] = X['Marital_Status'].astype('category')
X['Education'] = X['Education'].astype('category')
X['Region'] = X['Region'].astype('category')

X= sm.add_constant(X)

# Fit the probit regression model
probit_model = Probit(y, X).fit()

# Print the summary of the model
print(probit_model.summary())

# Predict probabilities
predicted_probs = probit_model.predict(X)

# Convert probabilities to binary predictions using a threshold of 0.5
predicted_classes = (predicted_probs > 0.5).astype(int)

# Confusion Matrix
conf_matrix = confusion_matrix(y, predicted_classes)
```

```
conf_matrix_df = pd.DataFrame(conf_matrix, index=['Actual Negative', 'Actual Positive'], co
lumns=['Predicted Negative', 'Predicted Positive'])
print("Confusion Matrix:\n", conf_matrix_df)

# Plotting the Confusion Matrix
plt.figure(figsize=(8, 6))
sns.heatmap(conf_matrix_df, annot=True, fmt='d', cmap='Blues')
plt.ylabel('Actual')
plt.xlabel('Predicted')
plt.title('Confusion Matrix')
plt.show()

# ROC curve and AUC value
fpr, tpr, _ = roc_curve(y, predicted_probs)
auc_value = roc_auc_score(y, predicted_probs)
plt.plot(fpr, tpr, color='blue', label=f'ROC Curve (AUC = {auc_value:.2f})')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve')
plt.legend(loc='lower right')
plt.show()
print(f"AUC: {auc_value}")

# Accuracy, Precision, Recall, F1 Score
accuracy = accuracy_score(y, predicted_classes)
precision = precision_score(y, predicted_classes)
recall = recall_score(y, predicted_classes)
f1 = f1_score(y, predicted_classes)

print(f"Accuracy: {accuracy}")
print(f"Precision: {precision}")
print(f"Recall: {recall}")
print(f"F1 Score: {f1}")
```

References

1. www.github.com
2. www.geeksforgeeks.com
3. www.datacamp.com