**ALY6015 80472 Intermediate Analytics SEC 04 Spring 2023 CPS**

**Module 6 Group Assignment — Signature Assignment REPORT**

Logo

Description automatically generated

**NORTHEASTERN UNIVERSITY**

**College of Professional Studies, Boston, MA, 02215.**

**Submitted by**

Zihan Ma

ma.zihan1@northeastern.edu

**Instructor**

 Prof. Valeriy Shevchenko

**Date**

06/25/2023

**Signature Assignment**

**Assignment Summary:**

NA

**Tasks And Report:**

**Part 1: Data correlation matrix**

**Correlation Matrix Analysis:**

The initial step of our analysis focuses on examining the relationships between the variables in our dataset. For this, we employ a correlation matrix. This matrix presents the pairwise correlations between different variables, with a correlation score ranging between -1 and 1. A score of 1 signifies a perfect positive correlation, -1 a perfect negative correlation, and 0 no correlation.

Below is the code for generated correlation matrix plot for our cleaned dataset:

A picture containing text, font, screenshot, line

Description automatically generated

This command calculates the correlation matrix for our dataset and visualize these relationships.

Here's a snapshot of the calculated matrix:

A picture containing text, screenshot, font, diagram

Description automatically generated

Several interesting correlations stand out.

1. **Latitude and Neighbourhood Group (Correlation ~0.36):** The positive correlation suggests that as you move north (increase latitude), you might be more likely to be in certain neighbourhood groups. This could be due to how neighbourhoods are distributed geographically - perhaps certain neighbourhood groups are more concentrated in the northern part of the region covered by the dataset.
2. **Longitude and Neighbourhood Group (Correlation ~-0.35):** The negative correlation here indicates that as you move east (increase longitude), you might be more likely to be in certain other neighbourhood groups. As with latitude, this could be due to how neighbourhoods are arranged, perhaps some neighbourhood groups are more common in the western part of the region.
3. **Calculated Host Listings Count and Room Type Binary (Correlation ~0.36):** This strong positive correlation suggests that hosts with more listings (higher calculated\_host\_listings\_count) are more likely to offer a certain type of room (indicated by room\_type\_binary). For example, if room\_type\_binary is a binary variable where 1 represents a private room and 0 represents a shared room, this correlation might indicate that hosts with more listings are more likely to offer private rooms. Alternatively, if 1 represents a whole property and 0 represents a private or shared room, it could suggest that hosts with more listings are more likely to offer entire properties. This could be because hosts with multiple listings are more likely to be professional property managers, who might be more likely to offer certain types of accommodation.

**Task 2: Correlation Values Related to 'Price'**

In line with the aim of our project to discover key variables that significantly affect the pricing of Airbnb listings, we created a bar plot that shows the correlation of different variables with 'price'. A vital step in this analysis is discerning the influence of 'room\_type\_binary' (the variable showing the most substantial correlation with 'price') on other variables. This step will help in further refining our model by eliminating variables that are impacting the price indirectly through their relationship with 'room\_type\_binary'.

The corresponding bar plot:

A picture containing text, screenshot, diagram, line

Description automatically generated

The correlations and their explanations:

1. **'room\_type\_binary'** shows the most significant positive correlation (~0.18) with the price, meaning that the type of room could be a major determinant of the price of an Airbnb listing. A plausible explanation for this could be that listings offering entire properties or private rooms might demand higher prices compared to those offering shared spaces.
2. **'availability\_365'** and **'calculated\_host\_listings\_count'** both display a minimal positive correlation (~0.017) with the price. The positive correlation with 'availability\_365' could indicate that listings available for a larger portion of the year are priced slightly higher. This might be due to the higher maintenance and overhead costs associated with keeping a property available for longer periods. Similarly, hosts with more listings ('calculated\_host\_listings\_count') might price their listings slightly higher due to increased operational costs.
3. On the contrary, **'latitude'** shows the most substantial negative correlation with the price (~-0.074). This negative correlation could suggest that listings located further north (with higher latitude values) might have lower prices. This could be due to various factors such as the popularity of the location, the cost of living in the area, or the desirability of the destination among tourists.
4. **'number\_of\_reviews'** also has a negative correlation (~-0.042) with the price. This might suggest that listings with more reviews could be priced lower. It could be that more affordable listings attract more guests, leading to a higher number of reviews.
5. **'longitude'** (-0.03) and **'minimum\_nights'** (-0.012) show a negative correlation with the price as well. A potential explanation for the 'longitude' correlation could be similar to that for 'latitude', with locations further east (higher longitude values) being less expensive. The negative correlation with 'minimum\_nights' might suggest that listings requiring longer minimum stays are priced lower, potentially to attract guests willing to commit to longer stays.

By identifying these relationships, we could streamline our model to focus on the most relevant variables, ensuring that our analysis is efficient and effective in predicting Airbnb prices.

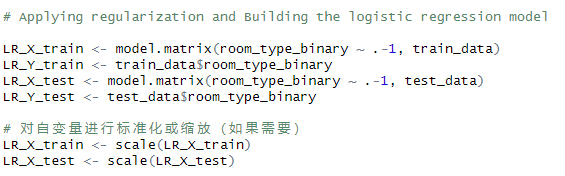
**Task 3: Dataset and Preprocessing**

we transition into the modeling phase, constructing a logistic regression model with L1 regularization (also known as Lasso regression). This regression technique is renowned for its ability to perform feature selection and diminish the impact of irrelevant variables by reducing their coefficients to zero. This will help us to find the variables that are important to room type predictions and compare them to the variables that are important to price predictions.

We begin by splitting our dataset into training and testing sets. The model matrix function is used to remove the intercept from our linear model formulation, as the glmnet package we use to apply the logistic regression model incorporates its intercept. We also standardized the predictor variables to ensure that they operate on the same scale, as it aids in accurately interpreting the feature importance and promotes the algorithm's stability and convergence.

A picture containing text, line, font, screenshot

Description automatically generated



We then fit a logistic regression model with Lasso regularization (alpha=1) to the training data and employ cross-validation to identify the optimal value for the regularization parameter, lambda. The cross-validation process involves training the model on a subset of the data and evaluating it on a validation set. The lambda that minimizes the binomial deviance (a measure of error for a binary classification model) is selected as the optimal lambda.

A picture containing text, font, receipt, screenshot

Description automatically generated

To test our model's performance, we made predictions on both the training and testing data, converting the probability outputs into binary outcomes (0 or 1) using a 0.5 threshold. We evaluated the model's accuracy by comparing these predictions to the actual room types. We obtained an accuracy of about 0.79 on the training data and 0.82 on the testing data, suggesting that our model generalizes well and is not overfitting the training data.

A close-up of a computer code

Description automatically generated with low confidence

Further, we constructed a confusion matrix, a critical tool in machine learning that provides insight into how well our model can distinguish between classes. The values in the matrix indicate the number of times the model accurately predicted a class or incorrectly classified observations. For instance, our model correctly predicted 983 + 955 = 1938 times and made incorrect predictions 167 + 268 = 435 times on the test data.

A picture containing text, screenshot, font, number

Description automatically generated

Here's what each of the four segments of the matrix represent:

1. **True Negative (TN):** These are cases in which we predicted 0 (negative), and the actual class was also 0. In your case, there are 983 such instances. This is a correct prediction.
2. **False Positive (FP):** These are cases in which we predicted 1 (positive), but the actual class was 0. In your case, there are 167 such instances. This is also known as a Type I error.
3. **False Negative (FN):** These are cases in which we predicted 0 (negative), but the actual class was 1. In your case, there are 268 such instances. This is also known as a Type II error.
4. **True Positive (TP):** These are cases in which we predicted 1 (positive), and the actual class was also 1. In your case, there are 955 such instances. This is a correct prediction.

The plot(LR\_cv\_model$glmnet.fit, "lambda", label=FALSE) command is used to plot a path diagram to visualize how the model's coefficients change as lambda varies. This visualization helps in interpreting the influence of lambda on the model's complexity. A higher lambda value leads to more coefficients shrinking towards zero, resulting in a simpler model.

For instance, with a lambda of 0.0000336 (minimum lambda where the model performance is best), the model includes 8 predictor variables and explains approximately 26.06% of the total deviance.

