Principles of Statistical Modeling

Concluding Miniproject.

by
Atit Bashyal

JACOBS
UNIVERSITY

Prof. Dr. Herbert Jaeger
Jacobs University Bremen

1 Introduction

The objective of this project is to explore and investigate statistical structures present in the dataset obtained from Nepal Living Standards Survey (NLSS) 2011. The NLSS survey is aimed at collecting data with the objective of measuring the living standards of the people in Nepal to determine the level of poverty in the country. The survey covers a wide range of topics related to household welfare which include demographic structure, monthly and weekly consumption, income, access to facilities, health, education etc. Results published from the NLSS survey has been of prime importance for government agencies and other organisations to assess the impact of policies and programs on socioeconomic changes in Nepal.

For me, this project serves as opportunity to utilize the knowledge I have gained through the semester in the Principles of Statistical Modelling course to explore the statistical concept behind the workings of an classification algorithm, namely: The Linear Discriminant Analysis. I also use the Linear Discriminant Analysis algorithm present in the python Sklearn package to explore the plot of the class distribution of poor and non-poor households present in the Dataset.

2 Data Description and Data Exploration

The dataset that I got access to from the Central Bureau of Statistics Nepal contains data and metadata from the Nepal Living Standard Survey(NLSS) conducted in the year 2011. The NLSS 2011 data obtained for this project included:

- The survey questionnaire
- Stata (version 14) data files: household data and individual data.

Each of the Stata data files contain:

- Set of 258 response variables generated from the survey questionnaire answered by each household.
- The sample weight variable for the household weight named ”wt_hh”
- A binary variable named “poor”.

The subsection below describes the dataset in correct mathematical formalism.

2.1 Data description in Mathematical Formalism

When statistical data are collected, the data collection process can be described using different components. Understanding these components of the data collection process helps us understand the dataset better and generate a mathematical abstraction of the dataset. The components that we will explore are as follows:

- **Universe and Elementary Events :** The Universe is a part of the real world from which we collect data. In mathematical terms the universe is represented by a set Ω. In the particular case of considering the NLSS dataset the Universe set is the collection of all the Households of Nepal in the year 2011.

  Elementary events are elements of the universal set from which the observation data is collected. Mathematically the elementary events are represented by ω are elements of the universe set Ω. Following the definition of the universe set provided for the NLSS data, the elementary events from which the NLSS data was collected are the households present in the universal set Ω.

- **Data Value Space:** The data value space mathematically is also set which contains all the possible outcomes of an observation procedure acting on elementary events, which we also can call the "observation act". In particular when considering the NLSS data, each question present in the survey questionnaire is an observation procedure and a household members answering the questions during the survey is the observation act on an elementary event(the household). A review of the survey questionnaire, shows that data values generated as answers to the questions are of three different types:

  - **categorical value :** The survey questions that output a categorical value produces a data value space of finite sets. These finite sets can further be divided in to two types depending on the type of values the elements of the sets have.

    * **Nominal valued elements:** These sets are finite and have elements that have no natural order and no numeric value. For the purpose of this project we will represent sets with nominal valued elements by c_i where i represents the index of questions in the survey questionnaire that output Nominal values. One such question in the questionnaire asked which ethnic group in Nepal does the household head identify with. The data value space generated by this particular question will be the finite set of 11 different ethnic groups recognized by the government of Nepal. Figure 1 below shows the total count of households that belong to these 11 different ethnic groups, based on the NLSS dataset. Next to make our task of defining the data value space easier we create a single nominal data value space represented by $\mathcal{C}$ by defining it to be the set product of c_i

$$\mathcal{C} = \times_i \, c_i \tag{1}$$

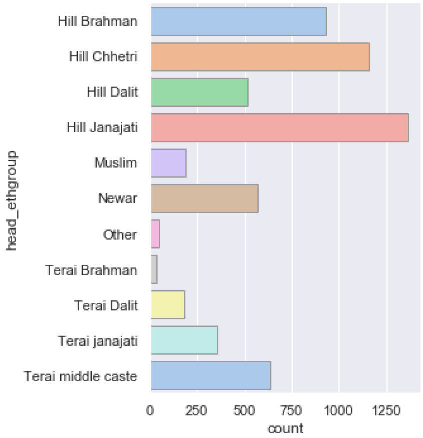


Figure 1: total count of households belonging to different ethnic groups in Nepal

- **Ordinal valued elements:** These sets are finite and have elements that have a natural order and a numeric value. The survey questions that output such values are questions that have yes/no answer. The natural ordering given to the answer of such a question is most commonly yes=1 and no =0. we represent such sets for the purpose of this project by $d_j = \{0, 1\}$ where j is is the index representing questions in the survey questionnaire that have a yes/no answer. One such question in the questionnaire asks if the household gets access to hospital facilities within 30 minutes. The data value space generated by this particular question will be the finite set {0,1}. Figure 2 below shows the total count of households according to access of hospital within 30 minutes, based on the NLSS dataset. Similar to defining the nominal data value space, we can define a ordinal data value space represented by $\mathcal{D}$ as the set product of d_j.

$$\mathcal{D} = \times_j d_j \tag{2}$$

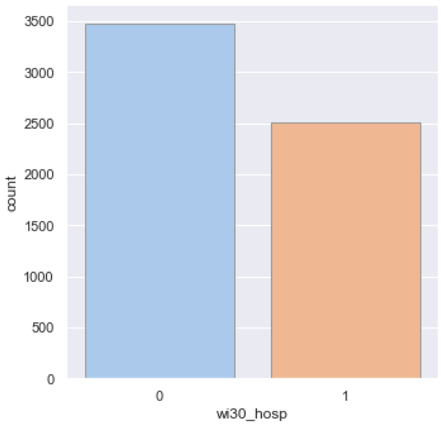
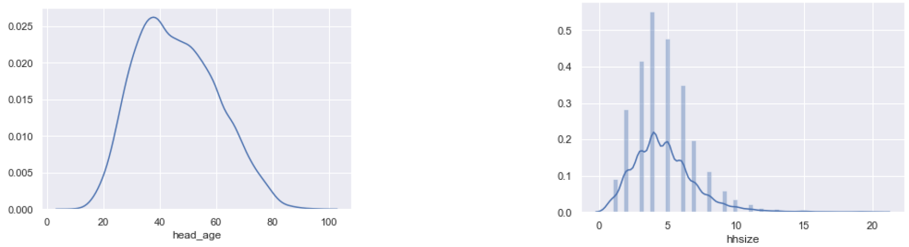


Figure 2: total count of household based on access to hospital within 30 minutes.

- **Numerical values:** Some survey questions output a numeric value. These questions will create either a discrete data value space or a continuous data value space. The discrete valued data value space are again sets of various lengths that contain elements belonging to the real numbers. We will represent these sets by g_k where k is the index of question that output a discrete data value space. Then we can further define a discrete valued data space such that,

$$\mathcal{G} = \times_k \, g_k \tag{3}$$

  on the other hand the continuous data value space contains different intervals of the real number line. To make the task of defining this continuous data value space we can merge the real line interval $[0, \infty]^n$ where n is the number of questions that require a continuous numerical answer. I have defined the upper limit of the line interval to be ∞ because, the data value space must contain all the possible outcome values, defining these intervals for each questions will require a long section and detailed discussion of the particular questions therefore to make the task simpler, I have defined the interval with an upper bound of ∞. Most of the numerical values resulting from the questions are discrete and a few of them are continuous. One question which outputs a discrete numerical value is the household size. Similarly the age of the household head outputs a continuous numerical value. Figure 3 below shows plots of the distribution of the household size and age of the household head, based on the NLSS data



(a) distribution of household head age (continuous numerical)

(b) distribution of household size(discrete numerical)

Figure 3: distribution of the variables household head age and household size the y axis has been normalized to make the area under the distribution curve 1

These different data value spaces created depending on the type of answer each question of the questionnaire requires can be merged together into a big data value space $\mathcal{S}$ which is mathematically represented as

$$\mathcal{S} = \mathcal{C} \times \mathcal{D} \times \mathcal{G} \times [0, \infty]^n \tag{4}$$

- **Random Variables:** are functions which map the elementary events from the universe set into data values in a data value space. The random variables are the mathematical formalism of representing the observation procedure. This follows that each question of the questionnaire, $q_l \mid l \in \{1, ...n\}$ where n=258 is the total number of question in the questionnaire, is a random variables that maps the elementary events into the data value space. These individual random variables q_l can also be expressed as a compound random variable $\mathcal{Q}$, where:

$$\mathcal{Q} := \bigotimes_{i=1}^{258} q_l \tag{5}$$

This compound random variable $\mathcal{Q}$ is thus the mathematical representation of the survey questionnaire. Following these definitions of the random variable and data value space the process of one household filling out the survey questionnaire can be formally represented mathematically as:

$$\mathcal{Q} : \omega \rightarrow \mathcal{S} \tag{6}$$

The whole survey process of collecting the household data is then represented in mathematical formalism by

$$\mathcal{Q} : \mathbf{\Omega} \rightarrow \mathcal{S}^N \tag{7}$$

where N is the number of households who were included in the survey.

Mathematically formalizing the data value space and the random variables gives us an mathematical description of how the data values of 258 columns of the dataset are generated from the elementary events in the universe, and defines a data-value space where the household characteristics have been defined mathematically. There are two other columns in our dataset namely the binary variable ”poor” and the survey weight variable ”wt_hh”. The values that these two variables take are not a direct result of the random variable $\mathcal{Q}$ acting on the elementary events. These variables have been generated using information from the mathematical representation of the households and household characteristics in the data value space generated by $\mathcal{Q}$ and the design/modelling choices made by the data collection agency, the Central Bureau of Statics Nepal (CBS).

Survey weights are typically a value assigned to each case in the data file. These values are used to make statistics computed from the data more representative of the population. In general, the survey weight is calculated for each case of the data file (in our case each household) using the following formula:

$$wt\_hh = 1/(sampling fraction) \tag{8}$$

where sampling fraction for each household in a particular strata of the sample is calculated using:

$$Sampling fraction = n/N \tag{9}$$

where n represents the sample size of a particular strata and N is the population size. The division of households in a particular strata is a consequence of how the CBS defines each particular stratum. The metadata does not discuss the process of choosing the sampling strata and the sample size of each strata so formalizing this process further is out of scope of this project.

On the other hand the variable ”poor” which categorizes each household as ”poor” or ”non-poor” has been computed using a method called the Proxy Mean Test. In formal mathematical terms that we use in the course, the Proxy Mean test is an estimator, which estimates for each household present in the dataset the level of household income based on the different household characteristics described by the dataset, as a further step this income estimate is then compared with a predetermined standard income level to categorize the household as ”poor” or ”non-poor”. In formal mathematical definition the Proxy mean estimator (which we represent by ρ for discussion in this project) is a function,such that:

$$\rho \circ \mathcal{Q} : \Omega \rightarrow \{0, 1\} \tag{10}$$

where a household categorized by a label of 0 is ”non-poor” and with a label of 1 ”poor”.

2.2 Dataset Exploration

In this section we plot various graphs to explore and identify presence of distinct relationships between the household characteristics and the poverty status of the household (characterised by the ”poor” label).

For numerical variables we can plot the distribution of features. Since each households have been labeled as poor or not poor we can further compare the distributions of the numeric variables between poor and non-poor responses. comparing these distributions will be interesting, because if the distributions are separable, it gives us a hint as to if the particular variable will be in useful as a predictor if the raw dataset is used for classification. Figure below shows the comparison of eight such features between poor and non-poor responses.

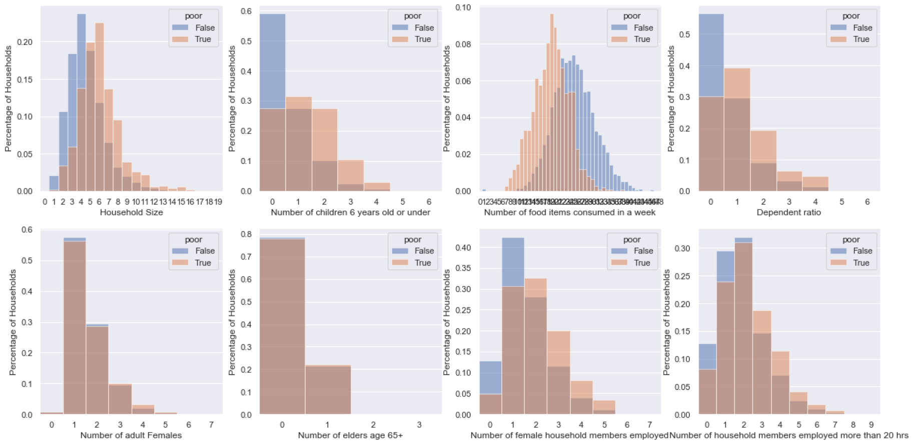


Figure 4: distribution of numerical features between poor and non-poor responses

The distributions in the plot above do not show distinct separation between poor and non-poor responses. The distribution of household size and the number of food items consumed in a week are two exceptions where some separation can be seen. In general it is expected that these two numerical features will function better as predictors for classification compared to the other two variables. The separable distribution of the food items consumed in a week for poor and non-poor households hints that other variables present in the dataset related to household consumption will play an important role as better predictors.

3 Linear Discriminant Analysis for classification of households

Classification is a supervised learning task, where the dataset used must be represented as a labelled pair:

$$(x_i, y_j) \tag{11}$$

where x_i are the different features that represent a case in the dataset that has been labelled using a class y_j. In the NLSS dataset we have each household represented by features (the household characteristics, collected from the Questionnaire) and a class label represented by the value of the ”poor” variable which we will refer to as the class variable. A classification algorithm is also an estimator, which takes as input a dataset and produces an class estimate for objects in the dataset. Using classification algorithm to classify objects in a dataset requires two steps, the first step which we classically call the ”training” step and the second step the ”testing” step. The training step is where we input a subset of our dataset as training-data into the classifier and allow the algorithm to ”learn” from the training data. The objective of the learning process is to obtain a ”well learnt” algorithm that when fed with the test dataset generalizes in a meaningful way what it has ”learnt” from the training sample. The ”learning” task looked upon mathematically, is trying to estimate the conditional distribution:

$$P(Y = y_j \mid X = x_i) \tag{12}$$

3.1 Linear Discriminate Analysis algorithm

The Linear discriminant analysis (LDA) is a supervised learning algorithm, which is most commonly used to reduce dimensionality in various machine learning problems. The algorithm is also used for classification problems. In principle LDA uses the training data $x_i \in \mathcal{R}^n$ to construct a lower dimensional space $\mathcal{R}^k for (1 < k < n-1)$ where the means of the classes present in the data set are maximally separated while the variances are minimized. The reduction of dimension is thus achieved by projecting the data features into the lower dimensional space. The use of the constructed lower dimension can then be extended for classification. The following section discusses in detail how LDA constructs this lower dimensional space and how classification can be achieved using LDA

3.1.1 LDA Explanation and Theory

As discussed above, The objective of the learning process of an classification task looked upon mathematically, is find an estimate of the conditional distribution:

$$P(Y = y_j \mid X = x_i) \tag{13}$$

To estimate the conditional distribution $P(Y = y_j \mid X = x_i)$, LDA uses Bayes' theorem which mathematically is written as:

$$P(Y = y_j \mid X = x_i) = \frac{P(Y = y_j)P(X = x_i \mid Y = y_j)}{\sum_j P(Y = y_j)P(X = x_i \mid Y = y_j)} \tag{14}$$

To use this formula, LDA makes important assumptions regarding the conditional distribution $P(X = x_i \mid Y = y_j)$. More specifically, LDA assumes that this conditional distribution $P(X = x_i \mid Y = y_j)$ is a normal (Gaussian) distribution. Another assumption that LDA makes is that, While the means of different classes may differ, their standard deviations are assumed to be equal. In other words, we assume that $P(X = x_i \mid Y = y_j) \sim N(\mu_j, \sigma_j)$ where, $\sigma_j = \sigma$. If we represent the gaussian distribution $N(\mu_j, \sigma_j)$ by $f_{\mathrm{x|y}}$ and $P(Y = y_j)$ by p_{y_j} means equation 10 can be written as,

$$P(Y = y_j \mid X = x_i) = \frac{p_{y_j} * f_{\mathrm{x|y}}}{\sum_j p_{y_j} * f_{\mathrm{x|y}}} \tag{15}$$

Using the assumptions we discussed above, the LDA estimates the class means and the standard deviations from the training data. With these quantities in hand, the new lower dimensional new space is determined by LDA where the space $w = \Sigma^{-1}(\mu_1 - \mu_0)$ maximizes the separation between class means while minimizing variance between the two clusters. While making predictions for the observations in the test dataset the LDA classifier uses these estimates of the mean and standard deviation to compute values of the distribution $P(Y = y_j \mid X = x_i)$ for each observation and classifies the observation to the class j for which the value of $P(Y = y_j \mid X = x_i)$ is maximum. The formula used by the algorithm make this decision varies form the from shown in equation 11, the actual form used by the algorithm is known as the discriminant formula, which arises by taking the log of equation 11, plugging in the formula of normal distribution for $f_{x|y}$ and algebraically reducing the equation into a simpler which has a form:

$$\delta(x_i) = x_i^T \sigma^{-1} \mu_j - 0.5\mu_j^T \sigma^{-1} \mu_j + \log(p_j) \tag{16}$$

This equation is also known as the discriminant equation. Therefore in its essence LDA assigns a observations to the class j for which the discriminant value $\delta(x_i)$ is maximum. Geometrically understanding this classification steps is simple much simpler, since LDA assumes a Gaussian distribution of the classes with the same standard deviation, if we project and plot our test data into the LDA transformed axis we expect to see the data as two normal distributions. Then when LDA is used as a classifier, data points from the test data is also projected into the LDA transformed axis, the model then assigns the data point to the class with the nearest mean to the data point.

3.2 Using LDA on NLSS Data

This section provides the results of applying LDA to the NLSS dataset. To run the LDA on the NLSS data Set the data was first divided into training and test data with a split ratio of 75:25. In the first model that is run the full set of variables present in the dataset was used as features. Then in subsequent models built later, features are selected based on the correlation between the LDA transformed axis of the initial model and each of our features. The classification performance of each model is evaluated using different metrics. Figure 5 shows the plot of the transformed LDA axis and the resulting class distributions of the initial model.

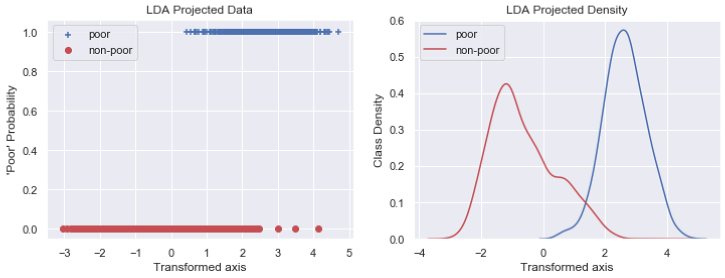
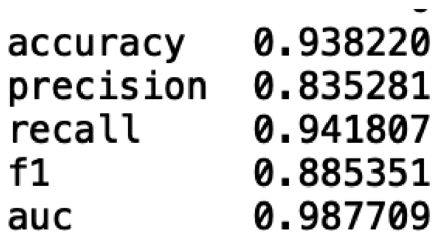


Figure 5: plot of the transformed training data in the LDA transformed axis and the resulting class distributions

Classification Performance:



```
accuracy   0.938220
precision  0.835281
recall     0.941807
f1         0.885351
auc        0.987709
```

Figure 6: Model evaluation metrics

3.2.1 Feature Selection Using LDA

One way in which we can utilize the Transformed axis of the LDA classifier is to use is for feature selection. In this method we will calculate the correlation between each feature of our NLSS data and the transformed LDA axis. We can then choose to exclude the features with very low correlation, since a low correlation means they have they have little impact on the transformed axis. In this section we run three new models, where we select features based is different correlation thresholds values and present the result of running LDA classification with the selected features.

- **Correlation $>$ 0.25**:

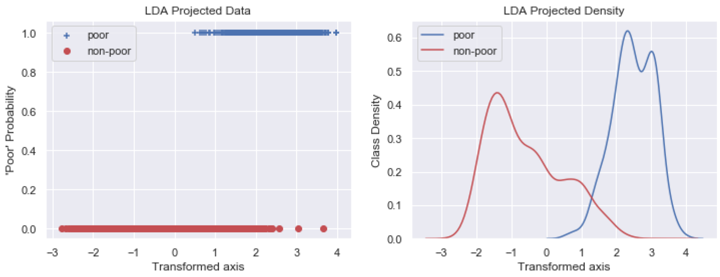
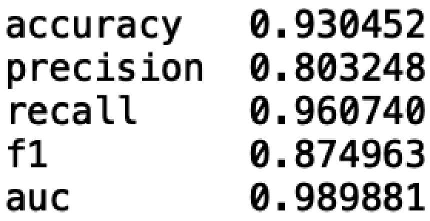


Figure 7: plot of the transformed training data in the LDA transformed axis and the resulting class distributions

Classification Performance:



```
accuracy    0.930452
precision   0.803248
recall      0.960740
f1          0.874963
auc         0.989881
```

Figure 8: Model evaluation metrics

- **Correlation > 0.5**:

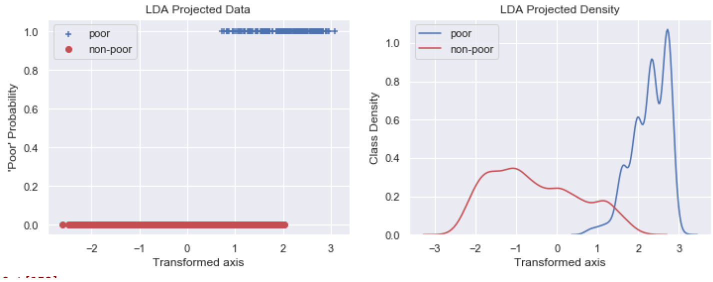
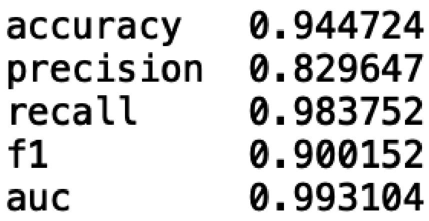


Figure 9: plot of the transformed training data in the LDA transformed axis and the resulting class distributions

Classification Performance:



```
accuracy    0.944724
precision   0.829647
recall      0.983752
f1          0.900152
auc         0.993104
```

Figure 10: Model evaluation metrics

- **Correlation > 0.75**:

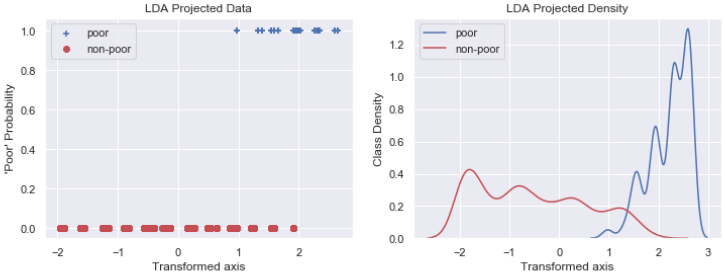
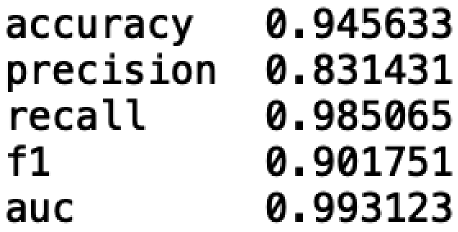


Figure 11: plot of the transformed training data in the LDA transformed axis and the resulting class distributions

Classification Performance:



```
accuracy   0.945633
precision  0.831431
recall     0.985065
f1         0.901751
auc        0.993123
```

Figure 12: Model evaluation metrics

We can notice that the class distributions plotted by the various models do not change in terms of the separability when the features change. But one thing we can notice is that the number of peaks(maximas) that the distributions have change with the features.More specifically, the number of peaks seem to increase as the number of features decrease which shows that the features that we have consequently selected using the correlation information do not just separate the classes, but also could create subgroups/clusters of households within each class.

4 References

- 1. Lecture Notes Principles of Statistical Modeling 2019: Herbert Jaeger.
- 2. Lecture Notes Machine Learning 2019: Herbert Jaeger.
- 3. An Introduction to Statistical Learning with Applications in R: Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani.