Contents

[Tayko Software Cataloger Case Study 2](#_Toc414657133)

[Estimating Gross Profit 2](#_Toc414657134)

[Classifying Customers as Purchasers and Non-Purchasers 2](#_Toc414657135)

[Predicting Spending Among Purchasers 4](#_Toc414657136)

[Classifying Purchasers and Non-Purchasers Using Logistic Regression 6](#_Toc414657137)

[Estimating Gross Profit Using Cumulative Lift Curve 8](#_Toc414657138)

[Executive Summary 8](#_Toc414657139)

# Tayko Software Cataloger Case Study

## Estimating Gross Profit

Question 1

Estimating the gross profit that the firm could expect from its remaining 180,000 customers if selected them randomly from the pool

The probability of a person responding and purchasing products from the initial random sample is 0.053

Applying this logic on to the larger data set, the probability of a person responding to a mail in population of 180, 000 is 180,000 \* 0.053 = 9540

The average spending was $103, applying this logic on to the purchasers from the larger data set, the average return would have been 9540 \* 103 = $982,620

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Transaction** | **# Count** | **Total $** |
| *Mailing Cost* | -2 | 180000 | -360000 |
| *Purchase* | 103 | 9540 | 982620 |
|  |  | **Profit** | 622620 |
|  |  | **Profit/ Catalog Mailed** | 3.459 |

Note:

Profit/Catalog Mailed = Profit/Mailing Cost count

## Classifying Customers as Purchasers and Non-Purchasers

Question 2

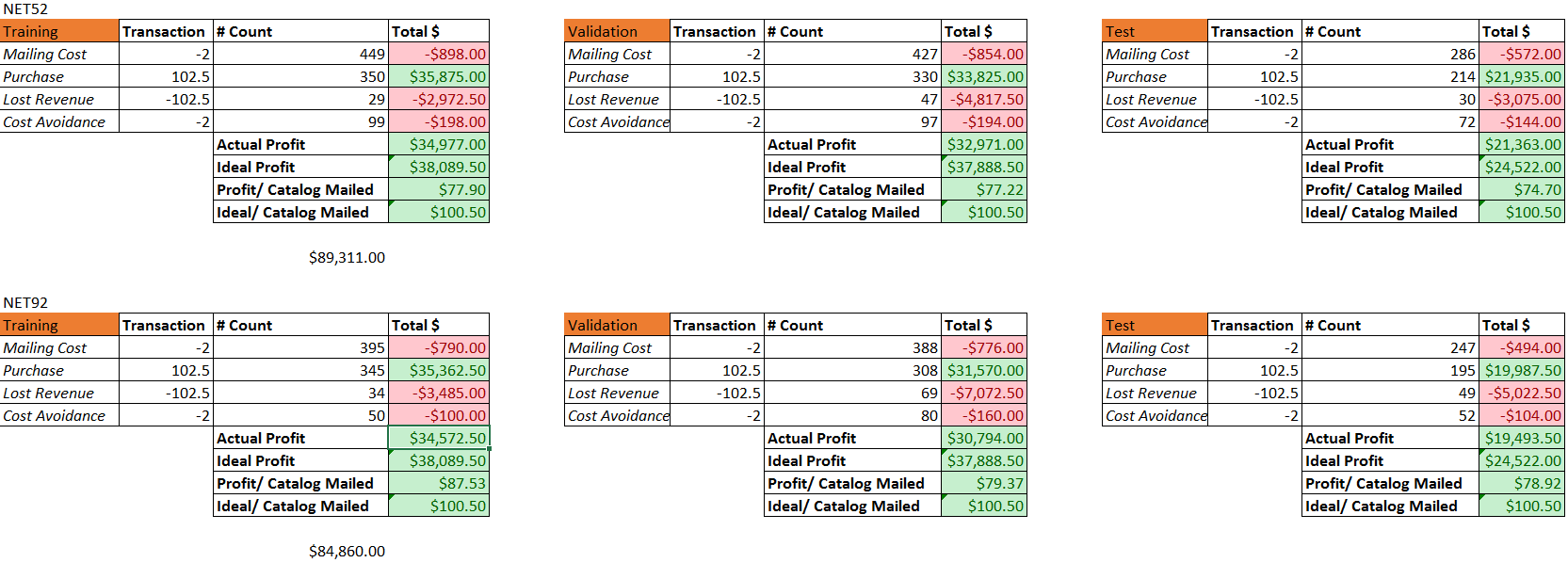
1. The data is partitioned using the partitioned using the partition variable. There are 800 training data, 700 validation data and 500 test data.
2. A classification is developed using neural networks with purchaser as output variable and other variable as input variables except for sequence number, source\_w and spending.

The model generates 100 different networks with considering different input variables at random. We are sorting the networks generated based on the sensitivity. Normally a network which has high sensitivity and specificity is considered to be the best network

Here for demonstration let’s look at Net 52 and Net 92. Net 52 has a sensitivity of 92.35 and specificity of 76.48 and Net 92 has sensitivity of 91.03 and specificity of 88.12. These are the two best network generated.

Now let’s see the count of mailing cost, purchase, lost revenue and cost avoidance.

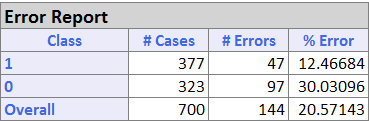
* Mailing Cost – Total number of customers the mail has been sent to
* Purchase – Number of people to whom the mail has been sent and those who responded
* Lost Revenue - Number of people who might have responded to but dint receive the mail
* Cost avoidance – Number of people who received the mail and did not respond



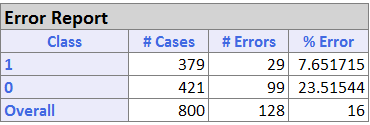
From the above table we derive that Net52 gives better profit than Net92 though Net92 has better specificity. This is because in Net52 has sent more mail to people who will accepted it. ***Having high specificity doesn’t necessarily mean that model is ideal***.

Now let’s take a look at the error report for all data

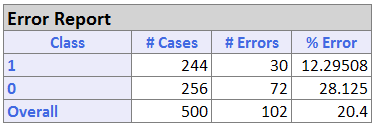
*Validation data error report*



*Training data error report*



*Test data error report*



## Predicting Spending Among Purchasers

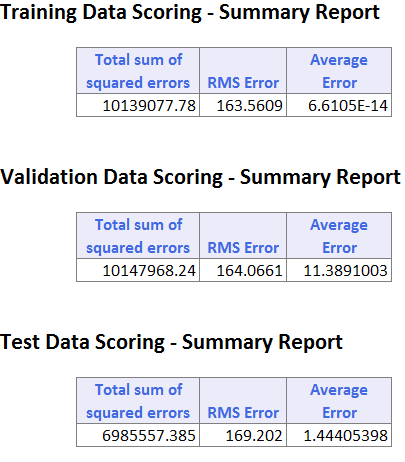
Question 3

#### Multiple Linear Regression Prediction

The dataset was partitioned into validation and training data based on partition variable. Now we are going to develop a model for predictive spending by best subset and backward elimination multiple linear regression

#### Best Subset

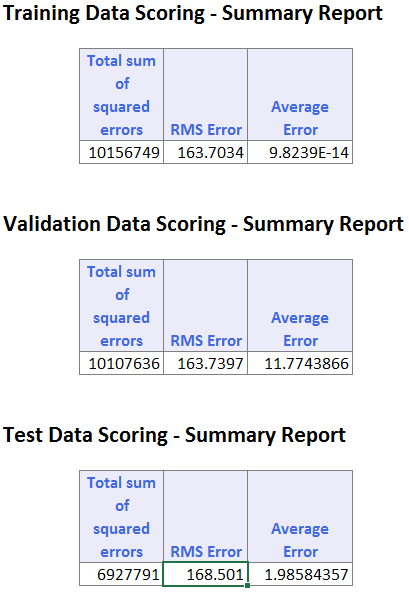
Let’s take a look at the error report if we use the best effort method



#### Backward Elimination

When we use backward elimination to predict spending in MLR, there will be a list of subsets generated. From that we have to choose the best subset and develop a model again to find the RMS error. We can choose the best subset based upon the R 2 value, Cp value and the Coeffs.

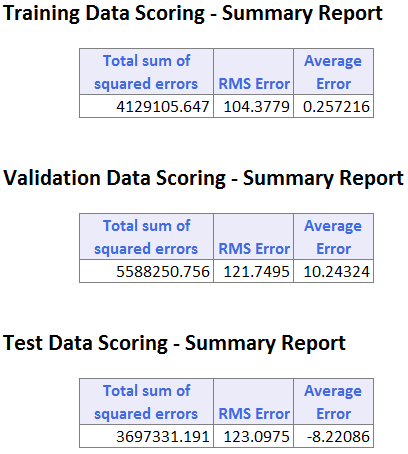
For choosing the best subset the value of R 2 should be as low as possible and the Cp should be as close to 1+Coeffs. Based on these factors we choose subset 15 to be the best subset of all and we develop another model with subset 15 to get the RMS error values. Now the take a look at the RMS error of validation, training and test data.



#### Neural Network Prediction

Now let’s develop a model with neural network with the same partitioned dataset and see how different the values are from MLR prediction. After trying out different Epochs and gradient descent values for the best result, we have set Epochs as 50 and gradient descent as 0.6.

Let’s take a look at the RMS error values if the model is developed using neural network prediction



In both cases there is a high RMS error for validation data. For best fit MLR its 164.0661, for backward elimination MLR it’s 163.797 and for neural network prediction is 121.7495. The average spending according to the dataset given is 205. The RMS values got after developing model using different techniques are quite high. This shows that both the models does perform well under unseen data. Both the models are not ideal for predicting the spending of purchaser. When compared to both MLR and neural network, neural network performs slightly well then MLR prediction.

## Classifying Purchasers and Non-Purchasers Using Logistic Regression

Question 4

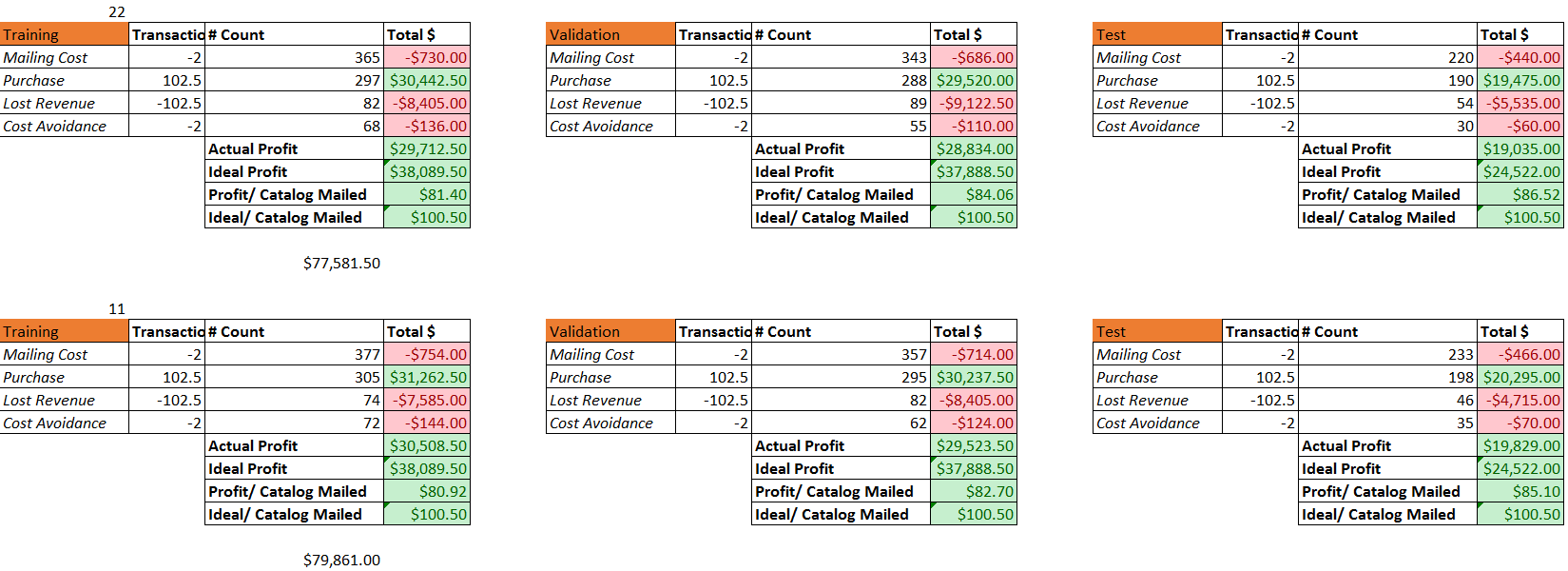
Here we are trying to develop a model to classify customers as purchasers and non-purchasers using logistic regression. The data is first partitioned based on the partition variable and a classification model is developed with purchasers as the output variable. This LR classification develops different subsets for us to choose from. The best subset can be chosen by looking at the Cp value. The Cp value should be as close to the value of sum of probability and Coeffs. Based on this now let’s compare two subsets.

For comparison let’s take subset 22 and subset 11 and apply the same methodology that we applied for question 2.

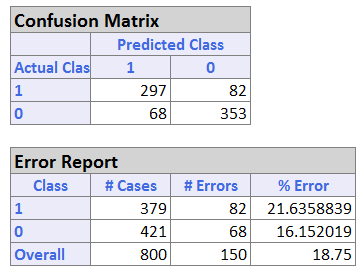
Now let’s see the count of mailing cost, purchase, lost revenue and cost avoidance.

* **Mailing Cost** – Total number of customers the mail has been sent to
* **Purchase** – Number of people to whom the mail has been sent and those who responded
* **Lost Revenue** - Number of people who might have responded to but dint receive the mail
* **Cost avoidance** – Number of people who received the mail and did not respond

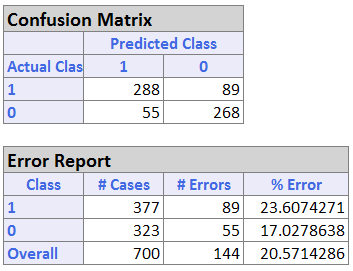
From the above table you can see that subset 11 performs slightly better than subset 22. So now lets the look at the error percentage for both validation and training data for subset 11.



*Training data – summary report*

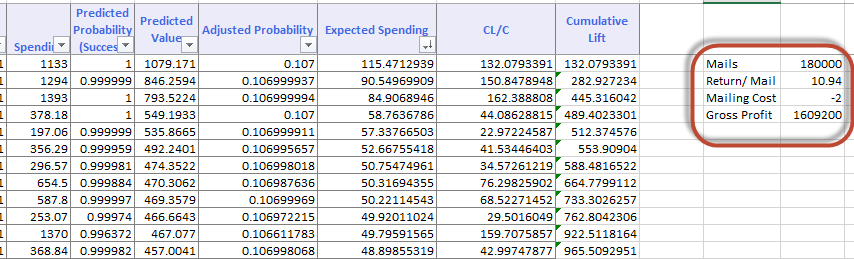


*Validation data – summary report*



This clearly shows that the classification regression model performs well on the training data rather than on the unseen data. The error percentages are lower in training data when compared to validation data.

## Estimating Gross Profit Using Cumulative Lift Curve



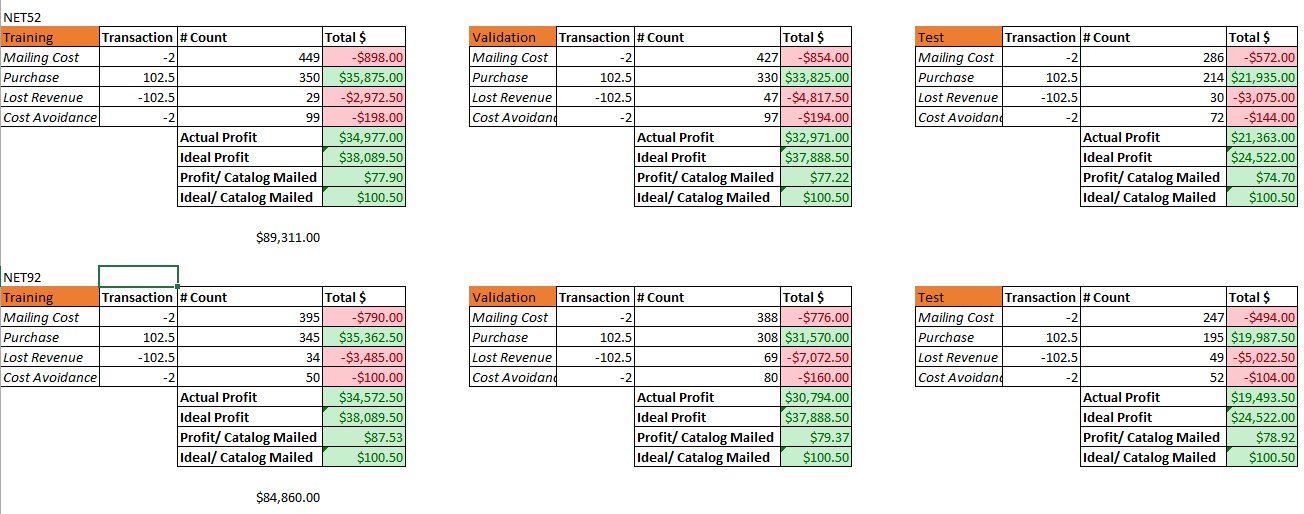
The average profit/ catalog mailed is 10.94. This value was identified by aggregating the cumulative lift of each entry.

180000 \* (10.94 - 2) = 1609200

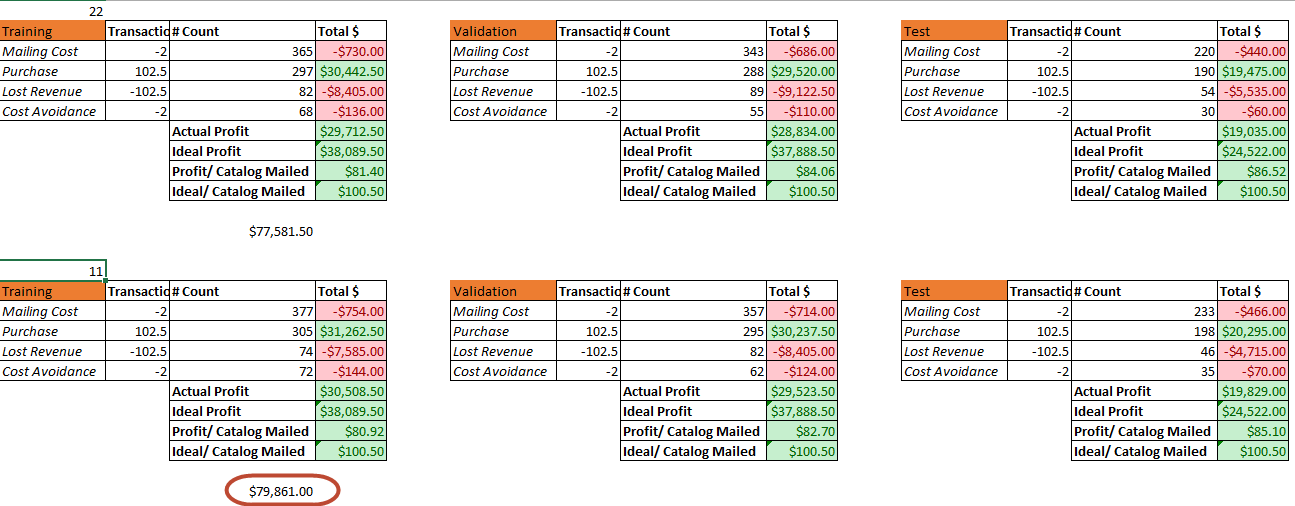
## Executive Summary

This report consists of an analysis on Tayko mailing experiment.

***Models with high sensitivity were chosen for analysis (identify to as many responders as possible) as the cost of mailing participants who don’t respond (false positives) was less (2$) when compared to average spending of the customers who were identified for mailing by the model.***

An analysis of classification of data (will the customer purchase or not) using Neural Networks was performed. Different neural network models were tried out to get the best model (to maximize profit). Various factors considered were the data partition, number of hidden layers, gradient step size and number of epochs. 

An alternative analysis using logistic regression was also done to improve the performance by using the ‘predicted probability of success’ and obtaining the ‘probability of success’ cutoff used in extending credit.



Logistic regression models were chosen for predicting the expected spending as neural networks (Black Box) sometimes tend to over fit the training data and don’t perform well on unseen data.