Project 3B

1. Copy the dataset to the name MyLoans.csv and change the last 5 loans to default (known\_value = 1).

-- See the attached file Project-3B-Kungulio.html:

* Data Preprocessing – line 3
* credit\_data[(nrow(credit\_data) - 4):nrow(credit\_data), "known\_value"] <- c(1, 1, 1, 1, 1)

1. Change the column headings to inc, “attained age”, borrowed, redundant\_feature, and answer.

-- See the attached file Project-3B-Kungulio.html:

* Data Preprocessing – lines 4 & 5
* new\_names <- c("clientid", "inc", "attained age", "borrowed", "redundant\_feature", "answer")
* colnames(credit\_data) <- new\_names

1. Adjust the code accordingly to train on this new data set.

-- See the attached file Project-3B-Kungulio.html

1. Report the values you get for the test set and for the validation set.

-- See the attached file Project-3B-Kungulio.html:

* Testing Neural Network – line 14 **0.9975**
* Validating Neural Network – line 18 **0.965**

1. Report the value you get for the training set (hint: see the plot produced).

-- See the attached file Project-3B-Kungulio.html:

* Training Neural Network
* **Error:** 1.539125 **Steps:** 27795

1. We have used this as classification application (0 or 1 for output). How would you change this to make it regression application (probability of default)?

To modify the neural network for a regression problem (predicting the probability of default), follow these steps:

* In the training neural network code (line #9) I would change the “linear.output = TRUE”
* Remove the rounding step so that predictions are continuous values between 0 and 1, representing probabilities.

1. What was the purpose of this assignment? What can you do now as a result of accomplishing it?

Purpose:

* To build and evaluate a neural network for predicting loan defaults.
* To understand the process of data normalization, model training, testing, and validation.
* To visualize and interpret the structure and performance of a neural network.

Skills Gained:

* Splitting datasets into training, testing, and validation sets.
* Training neural networks using the neuralnet package in R.
* Evaluating model performance using accuracy metrics.
* Plotting and interpreting neural network structures.
* Adapting models for different types of machine learning tasks (classification vs. regression).