Due: Sunday, Feb 3, by 5pm

1. On connect, you will find a data set called “car93”. It is a cleaned up version of the “Cars93” data in the “MASS” library.

   (a) Perform a principal components analysis on the numeric variables within the car93 data set. Provide a summary of the fitted model and a biplot. Ensure you scale the data.

   (b) Interpret the loadings of the first principal component.

   (c) Interpret the loadings of the second principal component.

   (d) How many principal components should be kept...

      i. according to the Kaiser criterion?

      ii. if we wish to retain at least 90% of the variance in the data?

      iii. according to the scree plot?

   (e) Keep the components suggested by the Kaiser criterion and...

      i. perform logistic regression (with a manual loop for leave one out cross-validation) with our response being “Small” or “Not Small” for the “Type” of car and the predictors being the components retained. What is the cross-validated logloss of this model?

      ii. perform LDA (with built-in leave-one-out cross-validation) with our response being “Small” or “Not Small” for the “Type” of car and the predictors being the components retained. What is the cross-validated logloss of this model?

      iii. perform LDA (with built-in leave-one-out cross-validation) using all categories from the original “Type” variable as the response. What is the logloss of this model?

   (f) Do the results from the above classification runs approximately match the discussion surrounding the interpretation of the first and/or second principal components? Explain.

2. Fit a neural network on the car93 data set with one hidden layer and 5 hidden layer variables to predict car price using all other quantitative variables. In order to provide a proper fit, you will need to standardize the variables. If you have an R object called `car_numeric`, which contains only the quantitative data, then we can use the following command to quickly normalize the way briefly described in class:

   ```
   scar <- apply(car_numeric, 2, function(v) (v-min(v))/(max(v)-min(v)))
   ```

   Use set.seed(4521) prior to the model fitting.

   (a) Provide the MSE of this model.

   (b) Of course, this could be overfitting, let’s set up a training and testing scenario.

   ```
   set.seed(217)
   ind <- sample(1:nrow(scar), 41)
   train <- scar[ind,]
   test <- scar[-ind,]
   ```

   Now fit the same neural network, then find the MSE for the test set.

   (c) On average, how far off is your model in its prediction of price? Give me this amount in dollars.