Our first task was to load the dataset and check it for any issues. 332,624 rows of data and 19 columns were loaded and a summary was generated in R. The variables C\_previous, duration\_previous, and risk\_factor contained “NA” values. The variable car\_value had missing values, which did not appear as “NA” when loaded into R. Code was required to convert the corresponding missing values to “NA” in order to then apply a conversion for missing data to the car\_value variable.

After loading the data and applying code to ensure that all missing data appeared as “NA,” the NA values were converted four each of the four variables. There were 9,366 rows with NA values for both C\_previous and duration\_previous. The definition of C\_previous indicated that the variable values could be 0 for “nothing” or 1, 2, 3, or 4 for each type of coverage possible. There were no values of 0 in the original dataset so we chose to convert NA to 0, believing that the value of 0 was somehow changed to NA during handling of the data. We then chose to convert the NA values for the duration\_previous variable, which all corresponded to NA values for C\_previous, to 0 as well, believing that the same logic applied. The duration\_previous indicates the amount of time that a customer was covered previously so if they didn’t have previous coverage their duration had to be 0.

Car\_value is a categorical variable with 9 categories ranging from “a” to “i.” 763 rows had NA values. We checked the frequency of each category assignment in the entire dataset and “e” was the mean value assigned so we chose to recode NA values to “e.” There were relatively few rows with NA values for car\_value so we didn’t want to create a new value category just for those rows.

Risk\_factor is a categorical variable with 4 categories 1, 2, 3, or 4. 120,291 rows had NA values, which was the largest number of NA for any of the variables. A boxplot of risk\_factor vs. cost showed that the mean cost differed significantly for each category, including NA, so we decided to assign the NA values their own category of 5. We also explored building a regression model to assign risk\_factor to NA, but the RMSE for risk\_factor vs. cost was worse with the modelled risk\_factor than it was for the category 5 risk factor so we chose to stick with 5 categories.

After loading the dataset and addressing the NA values we performed EDA to look for any issues or obvious associations of variables with one another. Some important issues are as follows. As described already, we looked at the mean cost for each risk\_factor and found that it was significantly different. We looked at age\_oldest vs. age\_youngest and discovered that they were highly correlated with one another. We also looked at age\_oldest vs. age\_youngest by group size and discovered a logical inconsistency. If a group\_size is 1 it should be logical to assume that the age\_oldest and age\_youngest would be the same, since there is just one individual to insure. That actually was not the case and there were instances of group size 1 with age\_oldest not equal to age\_youngest. We looked at married\_couple vs. group\_size and found a similar logical inconsistency. The married\_couple variable is a binary “yes” “no” variable with “yes” indicating that the customer group to be insured contains a married couple. However, there were instances of married\_couple “yes” and group\_size 1. The EDA results seemed to indicate that there was noise in the data, which could limit the usefulness in the model, but it did not prompt us to eliminate variables from the model based on EDA.

A final step in EDA was to look at univariate analyses of the dataset variables vs. cost. We looked at the r square and p-values associated with the F statistic in order to determine which variables to include in our initial model.

The following variables were not included in our model:

1. “Transaction\_ID”: this was the first (unlabeled) column and served as the primary key for the table
2. Customer\_ID: categorical, n = 94,934
3. Shopping\_pt: categorical, p-value = .9662
4. Record\_type: binary, categorical, p-value = .6331
5. Time: we wrote code to convert time to a 24 category variable and also a continuous decimal variable ranging from 0.01 to 23.99. Although there appeared to be some significance, we determined that there was no improvement in the model by including time in either form.
6. Location: we included location as a continuous integer variable (p-value = .2362) as well as a categorical variable (n=6,227) and determined that there was significance using the categorical variable but we would likely over fit the model due to the very high number of degrees of freedom.

We constructed a glm model in R to predict y = cost for each insurance policy. The following variables were included in our model:

1. Day: categorical variable for day of week, used as defined in the data dictionary
2. State: categorical variable for state, used as defined in the data dictionary. 36 states were in the dataset.
3. Group\_size: integer variable ranging from 1 to 4, used as defined in the data dictionary
4. Homeowner: binary categorical variable indicating customer owns home, used as defined in the data dictionary
5. Car\_age: integer variable ranging from 0 to 85 indicating age of car in years, used as defined in the data dictionary
6. Car\_value: categorical variable described previously. No new categories introduced.
7. Risk\_factor: categorical variable described previously. One new category was created to address NA values.
8. Age\_oldest: integer variable ranging from 18 to 75 years, used as defined in the data dictionary
9. Age\_youngest: integer variable ranging from 16 to 75 years, used as defined in the data dictionary
10. Married\_couple: binary categorical variable, used as defined in the data dictionary
11. C\_previous: categorical variable described previously. No new categories introduced.
12. Duration\_previous: integer variable ranging from 0 to 15 years, used as defined in data dictionary

We explored interaction terms and determined that there weren’t any combinations that improved the RMSE or adjusted r squared when added to our baseline glm model. We didn’t explore higher order terms in our model.