**DSC 680**

**Project Two**

**Milestone 1**

**Milan Sherman**

**Topic:** I am going to build a model to predict funnel metrics for the next 30 days for each of our marketing channels, including:

* Site visits
* Started applications
* Submitted applications
* Approved applications
* Bound policies

**Business Problem:** in order to anticipate and plan for revenue, expenses, and cost per acquisition (CPA), we need to be able to generate the metrics noted above by channel. As our pay structure varies by channel, e.g., cost per click, started app, bound policy, etc., we need to be able to forecast these metrics by channel in order to get an accurate CPA. For revenue, it is sufficient to get the number of policies across all channels and use the average revenue per policy. Right now, the growth team is doing this forecast in a very manual way, using historical data and anticipated changes in spend for the following month. They would like to develop a more sophisticated model that can be automated.

Research Questions:

1. Can we create a model that is better at predicting a user’s approved risk class at quote than our current system?
2. What data is needed beyond what we currently collect at quote to achieve this?
3. While accurately predicting a user’s approved risk class at quote is the ultimate goal, can we also reduce the error for users that we make incorrect predictions for? We have a total of 10 risk classes, and the further away a user is at approved from where they were at quote, the larger the price difference they will see. So in addition to making more accurate predictions, can we also reduce the size of the error, and therefore the price discrepancy, for users that the mode fails to correctly predict?

**Data:** as noted in the second research question above, we will need to use data that is not currently collected when a user receives a quote in order to build a better model. I think the key will be identifying the features that we currently collect in the application that are most predictive of approved risk class, and moving them to the quote portion of the user experience. Fortunately, this is something that our product team is already working on, i.e., collecting more data at quote in order to provide product recommendations and a more personalized quote experience for users.

We have more than 100 fields in our application data table, and my plan is to first create a model using all of these features, and then conduct some feature importance analysis and determine a subset of features that we could collect at quote that would achieve the goals of this project.

**Methods**: my first thought was to build a multiclass classification model, since we are trying to predict class membership, i.e., approved risk class. However, in speaking with members of the Data Science team, they suggested that I also try using the relative mortality associated with each risk class, which would convert the problem to a regression model at its core, before converting predicted mortality rates back to a risk class. In addition, I will need to conduct some feature importance analysis in order to identify a subset of features that can be collected at quote.

**Ethical considerations:** I think the main ethical consideration is that the model does not discriminate against any protected classes, i.e., women or minorities, in terms of its predictions of approved risk class and the associated price. For example, do the approved risk class predictions result in showing women an approved price that is closer or further from the quoted price than what men see? In particular, if the resulting approved price tends to be higher for women (in relation to the quoted price they shown), then the model could deter women from applying. The Data Science team has recently started conducting Adverse Impact Ratio analysis on its models in order to assess that impact, and this project could be a good opportunity to learn how to do that.

References:

Multiclass Classification:

1. <https://builtin.com/machine-learning/multiclass-classification>
2. <https://towardsdatascience.com/comprehensive-guide-on-multiclass-classification-metrics-af94cfb83fbd>

Regression:

<https://towardsdatascience.com/random-forest-regression-5f605132d19d>

<https://machinelearningmastery.com/random-forest-ensembles-with-xgboost/>

Another resource that I plan to leverage is ChatGPT. I think this is an important tool to understand how to leverage in creating and tuning machine learning models. I think it has already become such an important tool in this field (and many others) that anyone who has not learned how to use it well will be at a disadvantage.