**DSC 630**

**Predictive Analytics**

**Project Proposal**

Sang Phil Han &

Milan Sherman

**What types of model or models do you plan to use and why?**

As we are trying to predict a binary outcome, i.e. whether a user will purchase a life insurance policy, we will use a classification model. We plan to use a logistic regression model as our baseline model, and to also develop a random forest classifier to see if we can improve upon the logistic regression model. After assessing the accuracy of these models and gaining a better understanding of our data, we will discuss whether or not another model has the potential to improve accuracy over these two models, and which may be most promising.

**How do you plan to evaluate your results?**

Accuracy will be an important metric for any models we create, but given the current scale of users who are approved for a policy and the number of agents we have contacting them, we have more tolerance for a false positive than a false negative. Currently, our agents have plenty of capacity for contacting users who have been approved, so we would rather err on the side of misidentifying a user as a good candidate to contact than to not contact a user who has been misclassified as not a good candidate. Since we would like to minimize false negatives over false positive, i.e., we would like our model to be optimistic, an important metric for us to evaluate results will be recall. Especially of our classes are imbalanced, this may be a more important metric than accuracy when comparing models.

**What do you hope to learn?**

There are two questions that we hope to answer with this model:

1. Who should we target with marketing?
2. How important is the difference between the quoted and approved premiums in influencing whether a user will buy a policy?

The purpose of this model is very specific: the users identified in this model have applied for a life insurance policy and been approved but have not bought a policy.  Given that only 30-40% of users who apply for a policy are approved, this is an important group of potential customers for the company to focus on as they’ve already cleared an important barrier to becoming a customer by getting approved.  This model will help the company understand who to target with marketing.  Currently the marketing action that the company is taking with these customers is to have an agent contact the user via phone to follow up and answer questions.  This model would allow agents to focus on users with a good chance of purchasing while at the same time not wasting time contacting users with a high probability of purchasing even if they are not contacted.

The other thing that we hope to learn is related to how our software generates insurance premium quotes.  We have found that more than half of users are shown a higher premium after they are approved than what they saw when they generated an instant quote at the start of the process.  By including the difference between these two premiums as a feature in the model and understanding its importance to the accuracy of the model, the company can begin to quantify how this feature is impacting the business and take appropriate action.  For example, if the model shows that users who see a premium that is more than 50% higher at approval are much less likely to buy a policy, then the software can be configured to try to make those premiums closer in price.

**Assess any risks with your proposal**

The main risk with this proposal is that the results are inconclusive.  Care will need to be taken to identify the limits of what can be learned from the model, especially regarding the importance of the feature that quantifies the difference between the quoted and approved premiums.  For example, it could be the case that users who see a larger difference between the quoted and approved premiums are also the users who are seeing the highest premiums. Thus, it could be that the reason that they are not purchasing a policy is due to the price of the premium and not necessarily due to the difference between the two premiums.  Thus, we may need to build two models, one with and one without this feature in order to understand its importance.

Another risk we will need to monitor is whether our model tends to be biased toward a particular demographic. At the core of our company’s mission is to provide affordable life insurance to people in traditionally marginalize demographics. As life insurance has traditionally been agent driven, and agents work on commission, higher SES demographics have had greater access to life insurance. We will want to monitor our results to see if our model replicates this bias.

**Identify a contingency plan if your original project plan does not work out**

We can certainly build a model using the features that we have identified, so in that sense we don’t believe that there is a scenario where it doesn’t work out.  What is in question is whether the model will be accurate enough to answer the questions we are asking. Given that we are asking two questions with this project, if we end up with a model that is not accurate enough to deploy, we can focus on understanding the importance of the premium difference feature as it relates to users buying a policy or not. Answering this question will provide insight regardless of what we find.

**Data**

Below are the features in the dataset that Sherman has access to from the Data Science team at his company. As we build our model, we will evaluate if any other features would be useful and/or if some of the features included can be eliminated in favor of a more parsimonious model. There are a few features we will need to follow up on with the Data Science team to understand what they represent.

* Total\_seconds - the amount of time spent by the user on the site
* Avg seconds per sessions - if the user had multiple sessions, this is average amount of time spent
* Face value: the value of the policy if the insured dies
* Income - of the user applying
* Lrc\_score - I think this is a measure of risk, where a high score means the user is less risky to insure and therefore will likely get a better price
* Bmi - body mass index, uses height and weight to give a single number that is used to assess risk, i.e., someone with a high BMI is less healthy, higher risk
* Open\_to\_submit\_min: the number of minutes it took the user to complete and submit the application
* Submit to approve min: the number of minutes it took for the application to get approved via the automated underwriting process
* Pre-approval interactions: not sure about this exactly. Bind means they bought a policy, so this is tracking what the user did before they were approved for a policy. We need to check what kinds of interactions are tracked here.
* Term length year: how many years the policy is for.  For example, if you buy a 20-year policy, the beneficiary only collects the face value of the policy if you die within 20 years
* Age: age of the user
* Post-approval pre-bind interactions: not sure, exactly.  Bind means they bought a policy, so this is tracking what the user did after they were approved for a policy but before they bought it. We need to check what kinds of interactions are tracked here.
* Anonid\_unique: unique identifier of the user
* Sessions\_uniq: number of unique sessions of the user
* Gclid\_count: not sure
* Event total: total number of events that the user engaged in.  Events are things like visiting the site, getting a quote, completing the application, etc.
* Total\_seconds: not sure how this differs from the first feature
* Gender: gender of user
* Channel\_combined: channel is what marketing channel the user came to the site from, but I’m not sure what channel\_combined represents
* Device: the device the session was completed on
* Device name: name of device
* Current ins; if the user currently has life insurance
* Employment status: employed or not
* Business line: we have four lines of business that users can purchase a policy through
* Rx result: I think this is the result of the medical background check done during the automated underwriting process
* Premium delta: difference in the premium that the user sees from the quote they got to the premium they see after they are approved.

**Milestone 3**

**10/9/22**

**Will I be able to answer the questions I want to answer with the data I have?**

* We will definitely be able to build a classification model with this data to predict which users approved for a policy are likely to bind
* We need to do some work analyzing the quote to approved premiums
  + - How does the premium change from quote to approved in this dataset?
      * About 60% of users saw an increase in premium from quote to approved, which is premium difference (quote minus approved).
      * On average, the raw difference was about $30 more at approved, or 56%
      * These numbers are close to what Sherman observed in the larger dataset, indicating that this is a representative dataset with regard to the primary variable of interest.
    - What is the distribution of the raw difference and the percent difference?

The distribution of the raw difference and the percent difference are both negatively skewed (skewed to the left).

* What proportion of users see an increase?

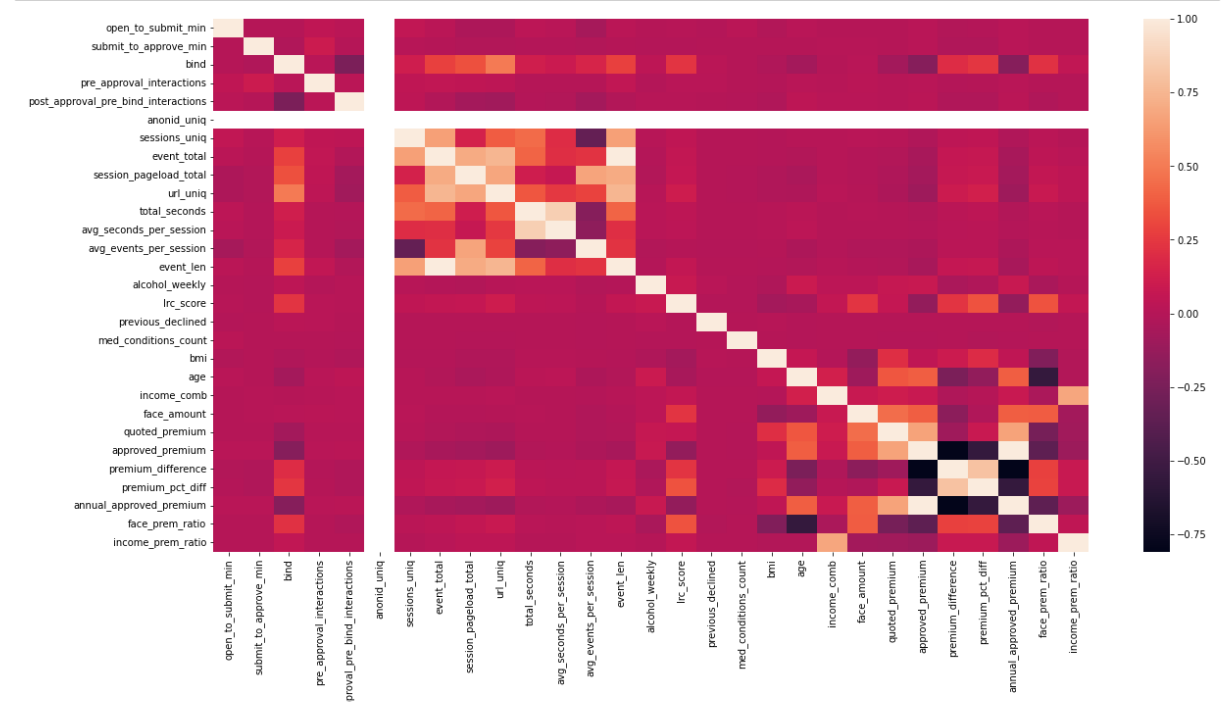
Based on our analysis, about 60% of users saw an increase in premium from quote to approved, which is premium difference (quote minus approved).

* How is this variable related to other variables in the dataset, especially the target variable?

When users shop for a long-term insurance, users are shown the quote. But after having their application approved for the same product and face value, a previous analysis showed that more than half the time, the premium the users ended up purchasing is higher than the one shown when they got a quote.  Thus, we are interested in finding out if this variable, premium difference (quote minus approved), is shown along with a quote, what impact it would have on the target variable (i.e. users’ purchasing decision).  Specifically, we’d like to learn more if our models show the ‘premium difference’ will lead users to more likely buy a premium or not.

**What visualizations are especially useful for explaining my data?**

* We created a heatmap of a correlation matrix for our numerical features (shown below)
  + This allowed us to identify:
    - A feature with no variation (the white line though the middle of the heatmap).
    - Perfectly or highly correlated features that are redundant and thus one of them can be removed
    - These are discussed in greater detail in the next section



* + Another useful visualization is a barplot of the distribution of the target variable, bind, which indicates whether a given user bound a policy or not. This is important to understand as an imbalanced target would require that we make some adjustments prior to model building, such upsampling or downsampling. The plot below show that more users did not bind, but that our target is not so imbalanced as to require some correction before modeling

Chart, bar chart

Description automatically generated

* + Scatterplot of premium difference with other numerical variables of interest (which are those?)

**Do I need to adjust the data and/or driving questions?**

* Our dataset consists of combining two data sources
  + User (127k observations): A dataset of users who were approved for a life insurance product (this represents about 30-40% of users who apply).  Some of these users bought a policy (referred to as ‘binding’) and some did not.  Each observation includes a binary variable for bind, which is our target
  + Quote (101k observations): The second dataset was generated by Sherman as part of an analysis he did that examines the premium that users are quoted when they first get to the site based on a few inputs: including gender, age, height, weight, and zip code, and the premium they are shown after being approved for the same product.  This dataset includes the premium at quote, the premium at approved, the raw difference, and the difference as a percent of the quoted premium.  The company is particularly interested in the importance of this difference in the model
  + After the joining the dataset based on the unique identifier application id, we have a dataset of about 45k observations
* In analyzing the categorical variables, it was discovered that there are only 41.7k unique application ids in our dataset of 45k. As application id is the unique identifier for our data, this implies that we have some duplicates, which we removed. Unfortunately, this further reduces the amount of data we have to train our model on.
* This leads to the following question: **will including the premium difference between the quoted and approved premiums improve our model more than including more observations?** A model could be built from the users dataset alone, which does not include that feature but would have three times as many observations. To answer this question, we return to our driving questions:
  + - **Can we accurately predict whether a user who is approved for a policy will bind?** For this question it makes sense to create models using both datasets to see which data provides better predictions.
    - **How does the difference between quoted and approved premiums influence users’ decision to bind or not?**  One way to answer that question is to create models with and without this feature, and assess how the inclusion of this feature affects the models accuracy. For this analysis, it makes sense to use the same dataset but with this feature removed, i.e. hold everything else constant to assess the importance of this feature.

**Data Dictionary**

After meeting with a member of the Data Science team this is the updated data dictionary

**From the user dataset**

* application\_id: categorical, unique identifier of the application
* bind\_date: date, the date of when the applicant bound a policy (needs to be removed from model to avoid leakage)
* open\_to\_submit\_min: integer, number of minutes to complete and submit the application
* submit\_to\_approve\_min: integer, number of minutes for the application to get approved after it is submitted
* channel\_combined: categorical, marketing channel(s) responsible for the application start ? or the quote? or something else?
* business\_line: categorical, direct, protect, core, advisors
* open\_date: date, the date the application was started
* bind: binary, indicating if the applicant bound a policy (target variable)
* anonid\_uniq: number of anonymous id's from segment
* sessions\_uniq: number of unique sessions
* event\_total: number of events
* session\_pageload\_total: number of session page loads
* url\_uniq: number of unique url's accessed by the user
* total\_seconds: how long on the web site
* gclid\_count: ?
* customer\_portal\_loads: ?
* device: device type, i.e., phone, tablet, desktop
* device\_name: device type + OS
* avg\_seconds\_per\_session: average time per session in second
* avg\_events\_per\_session: average number of events per session
* event\_len: average number of seconds per event
* gender: gender of applicant
* tb: binary, tobacco user or not
* current\_ins: binary, does the applicant currently have life insurance
* rx\_result: (remove)
* replacement\_ins: binary, does the applicant currently have replacement insurance (what is replacement insurance?)
* legal\_resident: resident or visa
* employment\_status: 'fulltime', 'parttime', 'fulltime\_student', 'retired','stay\_at\_home', 'parttime\_student', 'unemployed'
* postal\_code\_address: zip code of home address
* income\_payment\_period: how frequently the applicant gets paid
* designate\_secondary\_contact: no non-null values, so this can be removed
* alcohol\_weekly: number of drinks per week as reported by the applicant
* disability\_pmts: binary, does (or has?) the applicant receive(d) diability payments (history)
* term\_length\_years: length of term (this will likely be removed, but compare to term length from quote)
* household\_income: reported household income (verify)
* lrc\_score: brought in from vendor risk score from lexus nexes, mortality risk score. This generally highly correlated with being approved
* face\_value: face value of the policy (this will likely be removed, but compare to face value from quote)
* previous\_declined: binary, has the applicant been previously declined for life insurance
* travel\_countries\_count: number of countries the applicant has visited
* med\_conditions\_count: number of medical conditions the applicant has history
* bmi: body mass index of applicant
* age: age of applicant
* income: income of applicant
* annual\_premium\_estimated: the annual premium shown to the applicant after approval (this will likely be removed, but compare to premium from quote)
* income\_comb: annual income
* income\_prem\_ratio: rato of income to premium (annual)

**From the quote dataset**

* created\_at: when the application was started
* business\_line\_y: business line, join on this
* quote\_id: quote id associated with the approved app
* face\_amount: face amount of the approved policy
* product: procuct code, including term length
* quoted\_risk\_class: the risk class that the applicant was put it at quote
* approved\_risk\_class: the risk class the applicant was put it after approval
* quoted\_premium: the monthly premium shown at quote
* approved\_premium: the monthly premium shown after approval
* premium\_difference: the difference between the two premiums (quote - approved)
* premium\_pct\_diff: the percent difference between the two premiums (quote - approved)/quote

**After discussing with the data science team, the following fields can be removed**

* The following two field should be removed to avoid leakage, as these would only be known after the user binds
  + post\_bind\_interactions:
  + total\_interactions: sum of the previous three fields (including post bind interactions)
* utm\_source\_len: not sure how the length of the various parts of the utm is relevant to predicting bind
* utm\_medium\_len
* utm\_campaign\_len
* utm\_term\_len
* referrer\_host\_len
* record\_created\_at: when the data was pulled
* mvr\_points
* mvr\_order\_status
* zip\_d1
* income\_grp\_v1: binned income group: under 50k, 50k-100k, 100k+
* income\_grp\_v2: binned income group; under 50k, 50k+

**Data cleaning**

We had significant work to do in understanding our dataset and cleaning it up

* We started with over 70 features, so it was important that we understood each one, including missing values, distribution, correlation, etc. in order to get to a set of features that provides the most accurate model by selecting those features and combination of features that are most informative regarding whether or not an approved user will purchase a policy

Data cleaning steps:

* Verified how income to premium ratio is defined in the data, and used the approved premium from the quote dataset to compute it as the same variable in the user dataset had a many missing values
  + Removed estimated\_premium
* Compared other variables in both datasets: term length and face value.  Term length is how long the life insurance policy is good for, and the face value is the amount the policy pays if the insured dies.  After comparing these values in both datasets, we are retaining those from the quote dataset as the data science team believes that they are more accurate, and those variables had multiple missing values in the user dataset.
  + Removed term length and face value from the user dataset
* Business line existed in both datasets and appeared to have 93% agreement
  + Removed business line field from the user dataset
* Removed fields gclid\_count, customer\_portal\_loads until we can verify what they represent (Data Science did not know)
* There were three fields that had user income data, two were removed after exploration
  + One income field from user had a large number of missing values and didn’t align with the other two
  + The other two income fields seemed to agree, but one of them had more missing values so it was removed
* Three other variables (replacement\_ins, travel\_countries\_count, legal\_resident) had missing values for at least 80% of observations
  + None of these variables seemed to be very correlated with the target (based on distribution with respect to bind), and including variables with so many missing values could negatively impact the ability of the model to make accurate predictions on new data, so we removed them for now.
  + As a Random Forest model can deal better with missing values, we may include these features for that model
* After creating a heatmap of correlations in the numerical features, a few features were found to be perfectly or highly correlated
  + Anonid\_uniq was removed as there was no variation in this feature
  + event\_total and event\_len are perfectly correlated, so we removed event\_len
  + total\_seconds and avg\_seconds\_per\_session are also highly correlated (0.86).  As the correlations of avg\_seconds\_per\_session is closer to 0 for the variables in this group, we retained that variable
  + Approved\_premium (monthly) and annual\_approved\_premium were also perfectly correlated.  We retained the annual version since other features in the set are annual
* We created face amount to premium ratio, which represents the ‘value’ of a policy in the sense that it captures the value of the policy relative to what the insured pays for it

**Do I need to adjust my model/evaluation choices?**

* I think a logistic regression model with regularization and a random forest model are still good choices to start with
  + Logistics regression with regularization will help us to further reduce the number of features by applying a penalty to those features that are redundant or noisy
  + Random Forest will allow for us to assess feature importance, which has important implications for both questions that we are asking:
    - To accurately predict which users will bind, and therefore better target new users, we need to be able to identify the optimal feature set to optimize accuracy. Knowing which features are most important will help with that process
    - The other question we are asking is regarding the importance of the difference between the quoted and approved premiums in influencing whether a user binds. Using Random Forest will allow us to assess the importance of this feature.

**Are my original expectations still reasonable?**