……………………………………………...……………………………………………...……………………………………………...……………………………………………………………...………………

**Name:** ABHIMANYU YADAV

**Email address:** abhimanyuyadavnsit@gmail.com

**Contact number:** 8860088322

**Anydesk address:**

**Years of Work Experience: 1**

**Date: 10th July 2022**

……………………………………………...……………………………………………...……………………………………………...……………………………………………………………...………………

**Self Case Study -1:** HOMESITE QUOTE CONVERSION

“After you have completed the document, please submit it in the classroom in the pdf format.”

Please check this video before you get started: <https://www.youtube.com/watch?time_continue=1&v=LBGU1_JO3kg>

# **Overview**

1. Every organization that we come across in our day-to-day life works on a limited resources and **each of the processes that are carried out in an organization consumes resources** and for the long-term survival organization it is important that these processes that are carried out in an organization are carried out in the most efficient way possible.
2. Homesite is looking out for **optimizing one such process of quoting home insurance prices to potential customers** where they want to build a model which can achieve the best possible conversion rate for them.
3. For this task they have provided us with a dataset which represents the activity of large number of customers who are interested in buying policies from there website. The **provided features for each of these activities are anonymized and provide a rich representation of perspective customer and policy.**
4. So, our task as a ML Engineer is to **design a model that could perform this task in the most optimized way possible.**

# 

# **Research-Papers/Solutions/Architectures/Kernels**

\*\*\* Mention the urls of existing research-papers/solutions/kernels on your problem statement and in your own words write a detailed summary for each one of them. If needed you can include images or explain with your own diagrams. it is mandatory to write a brief description about that paper. Without understanding of the resource please don’t mention it\*\*\*

EXISTING SOLUTIONS

1. <https://towardsdatascience.com/a-classification-problem-with-python-homesite-quote-conversion-15174bca09b8>
   1. They have used the representation of data provided without any pre-processing on it and used various model like Decision Tree, Random Forest, KNN , Gradient Boosting & Neural Network out which the Gradient Boosting model has performed the best.
2. <https://www.kaggle.com/competitions/homesite-quote-conversion/discussion/18837>
   1. Using different feature engineering improved the auc score to: Public = 0.96910, Private = 0.96855.
      1. Removed features that had zero or near zero variance 97% taken as a threshold for near zero variance.
      2. Replace NAs in 2 columns with a new constant value (100).
      3. Centre, Scale, and Box-Cox transformation on numerical.
      4. Extract date features (day, month, day of week, year, quarter, week of year, day since epoch, etc).
      5. One-hot encoding for categorical and date features.
      6. Count -1s, NAs, ""s in each row and add the counts as new features.
      7. Used xgb.importance and correlation to target to select most important features for the coming step.
      8. Add features: log(x), sqrt(x), 1/x (where x is in the top N important features).
      9. Add features: x+y, x-y, x\*y, x/y (where x and y are in the top N important features, or x and y are highly correlated with each other).
      10. Repeated 9 and 10 to perform up to 4 operations (e.g.: x+((1/w)\*(y-z))).

# 

# **First Cut Approach**

\*\*\* Explain in steps about how you want to approach this problem and the initial experiments that you want to do. ***(MINIMUM 200 words)*** \*\*\*

\*\*\* When you are doing the basic EDA and building the First Cut Approach you should not refer any blogs or papers \*\*\*

1. Will start with defining the problem statement for a given business problem description followed by defining objectives that we wish to achieve but these objectives must be fulfilled such that certain constraints are taken care of while achieving the objectives. The constraints for the given problem are as follows:
   1. High cost of misclassification if company working on limited resources.
   2. Working with probability scores to give the freedom to tweak the thresholds based on the business scenario.
   3. No strict latency requirements.
   4. High Interpretability of the model is desired to ensure informed decision making.
2. The second task involves shaping up the business problem into a ML problem. The steps to achieve it are as follows:
   1. Take a basic overview of data which involves what all data have we been provided with, what all files have we been provided with to perform the task, what all information have been provided in those files.
   2. Identifying the type of ML problem here in our case it is a BINARY CLASSIFICATION problem.
   3. Deciding the performance metrics we will be working with to compare the performance of models we have built and choosing the one which performs the best on the data provided.
   4. Strategy for construction of train, cross validation and test need to be finalized based on the nature of data. Since the data in our case if of NON-TEMPORAL nature we would be sticking to RANDOM SPLITTING.
3. EDA – In our case the data provided to us has been already featurized and anonymized which makes it difficult to make sense out of it but have followed the following steps to extract information out of the data.
   1. Reading basic statistics of data
   2. The dataset was checked for imbalance, and it was found that the successful conversion datapoints make only 18% of the dataset rest all the datapoints belong to unsuccessful conversion category.
   3. Checked dataset for null values and it was found that two columns **PersonalField84 & PropertyField29 had more than 50% of their values as null values so dropped both the columns from the dataset and for rest of the nan values in the dataset replaced them with empty string.**
   4. Identified PropertyField6 & GeographicField10A as features that had only one value for all the datapoints so dropped them.
   5. Tried processing the Orignal\_Quote\_Date feature to make three new features month , day , year but the model performance dropped so did not include them in the final feature list.
   6. Encoded categorical features using one hot encoding technique.
4. BUILT MACHINE LEARNING MODELS
   1. Trained Random Forest and XGBoost model on the given dataset computed the ROC score for them to compare the performance of both the models.
   2. Didn’t try using KNN because we wish to have a model that is highly interpretable which is not possible with KNN as KNN can only be interpreted for low dimensional data and RF and XGBoost are anyways powerful alternatives to it.

**Notes when you build your final notebook**:

1. You should not train any model either it can be a ML model or DL model or Countvectorizer or even simple StandardScalar
2. You should not read train data files
3. The function1 takes only one argument “X” (a single data points i.e 1\*d feature) and the inside the function you will preprocess data point similar to the process you did while you featurize your train data
   1. Ex: consider you are doing taxi demand prediction case study (problem definition: given a time and location predict the number of pickups that can happen)
   2. so in your final notebook, you need to pass only those two values
   3. def final(X):

preprocess data i.e data cleaning, filling missing values etc

compute features based on this X

use pre trained model

return predicted outputs

final([time, location])

* 1. in the instructions, we have mentioned two functions one with original values and one without it
  2. final([time, location]) # in this function you need to return the predictions, no need to compute the metric
  3. final(set of [time, location] values, corresponding Y values) # when you pass the Y values, we can compute the error metric(Y, y\_predict)

1. After you have preprocessed the data point you will featurize it, with the help of trained vectorizers or methods you have followed for your train data
2. Assume this function is like you are productionizing the best model you have built, you need to measure the time for predicting and report the time. Make sure you keep the time as low as possible
3. Check this live session: <https://www.appliedaicourse.com/lecture/11/applied-machine-learning-online-course/4148/hands-on-live-session-deploy-an-ml-model-using-apis-on-aws/5/module-5-feature-engineering-productionization-and-deployment-of-ml-models>