Humana-Mays Healthcare Analytics 2022 Case Competition

Medicare Housing Insecurity Prediction
And Potential Solutions

Chuheng Yu
Tyler Christoforo
Ziyuan Li

October 2022

Executive summary

As increasing numbers of people are suffering from housing insecurities, Humana decided to set this year's goal on identifying and predicting MAPD((medicare advantage prescription drug) members on whether they will face a housing insecurity situation. With a dataset of over 800 medical claims and condition features, we intend to implement data-cleaning methods prior to our modeling and analysis. We dropped columns with more than 60% of missing data, then replaced other missing categorical values with the mode and missing numerical values with the median. Then we implement resampling methods due to the imbalance data distribution in the hi_flag column. RandomUnderSample and fit_resample are two functions we used to achieve our goal of balancing the hi_flag column.

In order to achieve our classification training goal, our team decide to use the Random Forest Model, the Decision Tree Model, and the XGBoost Model as our model pool. The Random Forest Model has an AUC score of 0.72. As for the Decision Tree Model, we did some cross-validation and tuning prior to the training process, and the most optimal depth is 3. Then we implemented the tuned model on our training dataset and we achieved an 0.639 AUC score. Lastly, for the XGBoost Model, we tuned the model with the Grid Search Cross Validation methods in order to find the best parameters for learning rate, max depth, and n estimators. After we implemented the model, it achieved an 0.765 AUC score which outperforms the other two models. As a result, we decided to go with the XGBoost Model and generated predictions with it.

Then, we visualized the top twenty influencing variables based on their coefficient and provided a summary with the top six factors: CONS_STLINDEX, EST_AGE, Cms_low_income_ind, Cons_hxmioc, Total_physician_office_allowed_pmpm_cost, and CMS_ORIG_REAS_ENTITLE_CD. And based on the training dataset, we provided seven recommendations that covered a more user-friendly UI, a base plan for ESRD members, medicare coverage expansion, more frequent physical check-ups, money-saving events for younger generations, and sent out notifications in advance based on predicted results to alert them for emergency shelters. We believe all of the recommendations could reduce the housing insecurity possibilities.

Table Of Contents:

1. Case Description

1.1 Background

One hundred and twenty percent of a paycheck is used for housing, twenty-five percent less buying power than 10 years ago, unthinkable numbers of change are affecting the economy and in turn affecting our lives. What was once a reachable goal to acquire a safe housing condition, can now be difficult to obtain. To make matters worse, the purchasing possibility decreases a lot more for major cities. The term housing insecurity has long been around, but only in recent years has it become a major societal issue for many. It encompasses a dimension of housing challenges, including housing instability, safety, affordability and quality. Housing insecurity can easily have an impact on physical and mental health, which can result in deeper issues. It also belongs to a broader scope, defined as Health-related social needs, these are immediate health-harming conditions that can heavily affect a specific individual.

Humana believes in the equality of opportunity in accessing support and tools to be as healthy as one can be. They are committed to improving health outcomes for all, establishing key metrics, and using quantitative metrics to help them achieve such a goal. With reliable plans and impactful goals, Humana has been the third largest health insurance provider in the nation, founding military specialized healthcare services, and speaking up against drug price fixing and other unjust in society. For this reason, Humana can continue to expand its impact in alleviating housing insecurity issues, through utilizing data and better predicting members that are more likely to face housing insecurity and provide adequate assistance in time.

1.2 Objective

Humana set this year's competition stage as housing insecurity prediction, using data to create a model to identify members most likely to be struggling with housing insecurity. The data originates from Humana MAPD(medicare advantage prescription drug) members, with the time of data collection tracing back to a one-year look back. The data has over 800 features, roughly categorized into medical claims and conditions, pharmacy claims, consumer features, and other features. Thus the main objective of the competition is to utilize this data, create a predictive model, and use the results to provide recommendations and potential solutions. In other terms, a combination of technical accuracy and practical solution in the form of a comprehensive report.

1.3 Data Description

In this competition, we have a training dataset and a holdout dataset. Inside the training dataset, there are a total of 48,300 records with 96% of housing secure unique members and 4% of housing insecure members. Overall, there are 60% of females and 40% of males. The race distribution in the training dataset has 78% of White (non-Hispanic), 16% of Black(non-Hispanic), and Unknown, Other, and Hispanic all around 2% of the training population. As for the holdout dataset, there are 12,220 total records with 60% of female and 40% of male which is the same as the training dataset. The holdout dataset has a similar race distribution as the training dataset which contains 78% of White(non-Hispanic), 16% of Black, 2% Unknown, and 2% of Hispanic. As conclusion, both datasets have similar race distribution and sexual distribution. However, the training dataset is extremely unbalanced with a 92% of sample population difference. Therefore, we applied methods such as undersampling in our following data processing and modeling sections.

2. Data Processing and Modeling

2.1 Data Cleaning

Before cleaning, the dataset has 48300 instances and 881 columns. In preparing the dataset for further treatments, we need to split the dataset into numeric and categorical data.

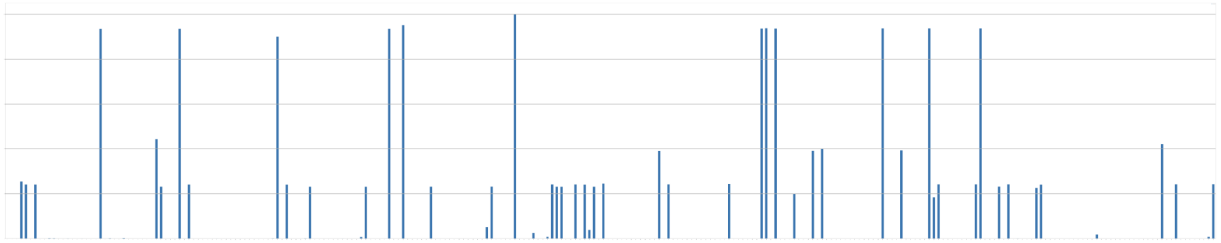
```
# select numerical columns
df_numeric = df.select_dtypes(include=[np.number])
numeric_cols = df_numeric.columns.values
# select non-numeric columns
df_non_numeric = df.select_dtypes(exclude=[np.number])
non_numeric_cols = df_non_numeric.columns.values
```

2.1.1 Missing Value Treatment

The first step is to find out how many values are missing in each column.

```
# % of values missing in each column
values_list = list()
cols_list = list()
for col in df.columns:
    pct_missing = np.mean(df[col].isnull())*100
    cols_list.append(col)
    values_list.append(pct_missing)
pct_missing_df = pd.DataFrame()
pct_missing_df['col'] = cols_list
pct_missing_df['pct_missing'] = values_list
```

```
pct_missing_df.loc[pct_missing_df.pct_missing > 0].plot(kind='bar', figsize=(500,100))
plt.grid(axis = 'y',linewidth = 10)
plt.show()
```



The graph above shows that some columns are missing a large portion of data, which should be dropped from the dataset first. We decided to manually set the threshold of 60% as the missing percent of missing values that justifies the column being dropped.

```
# dropping columns with more than 60% null values
_60_pct_missing_cols_list = list(pct_missing_df.loc[pct_missing_df.pct_missing > 60, 'col'].values)
df.drop(columns=_60_pct_missing_cols_list, inplace=True)
```

For the remaining null values, we decided to replace them with appropriate parameters. Specifically using mode for categorical data, and median for numerical data.

```
# dropping columns with more than 60% null values
_60_pct_missing_cols_list = list(pct_missing_df.loc[pct_missing_df.pct_missing > 60, 'col'].values)
df.drop(columns=_60_pct_missing_cols_list, inplace=True)
```

```
df_numeric = df.select_dtypes(include=[np.number])
numeric_cols = df_numeric.columns.values
for col in numeric_cols:
    missing = df[col].isnull()
    num_missing = np.sum(missing)
    if num_missing > 0:  # impute values only for columns that have missing values
        med = df[col].median() #impute with the median
        df[col] = df[col].fillna(med)
```

```
df_non_numeric = df.select_dtypes(exclude=[np.number])
non_numeric_cols = df_non_numeric.columns.values
for col in non_numeric_cols:
    missing = df[col].isnull()
    num_missing = np.sum(missing)
    if num_missing > 0:  # impute values only for columns that have missing values
        mod = df[col].describe()['top'] # impute with the most frequently occuring value
        df[col] = df[col].fillna(mod)
```

2.1.2 Categorical and Numeric Data

The next step would be to transfer categorical data into numerical data.

```
df = pd.get_dummies(df)
df.shape
```

```
(48300, 908)
```

After applying get_dummies and transforming categorical data, we now have 48300 rows and 908 columns.

2.2 Data Modeling

2.2.1 Resampling

Knowing the dependent variable is ‘hi_flag’, we decided to check whether the dataset is balanced or not.

```
df['hi_flag'].value_counts()
```

```
0    46182
1     2118
Name: hi_flag, dtype: int64
```

The outcome was that the dataset was extremely imbalanced with 46182 rows of 0 and only 2118 rows of 1.

After performing the regular test train split, we decided to use RandomUnderSample and fit_resample to achieve our goal of balancing the hi_flag column.

```
y = df.hi_flag
X = df.drop(['hi_flag','id'], axis=1).select_dtypes(exclude=['object'])
train_X, test_X, train_y, test_y = train_test_split(X.values, y.values, test_size=0.1)
```

```
from imblearn.under_sampling import RandomUnderSampler
undersample = RandomUnderSampler(sampling_strategy='majority')
X_train_under, y_train_under = undersample.fit_resample(train_X, train_y)
```

Now we can check the count of 0 and 1 with the following function, and the resampled training data set is even now.

```
np.unique(y_train_under, return_counts=True)
```

```
(array([0, 1]), array([1895, 1895]))
```

2.2.2 Building Model Pool

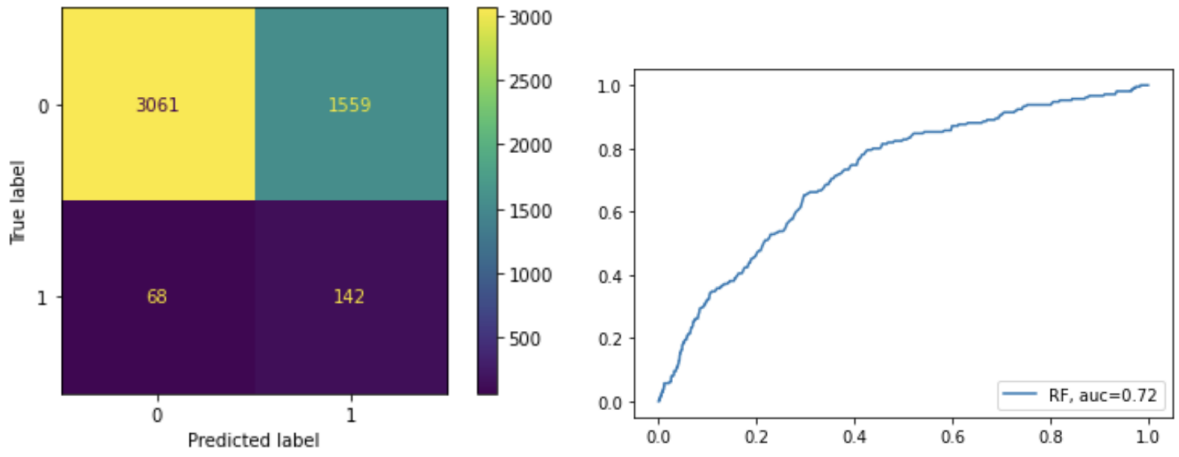
The problem we are facing in this project is a binary classification that we are predicting the value of 'hi_flag'. We chose Random Forest Model, Decision Tree Model, and XGBoost Model as our classification models.

2.2.3 Random Forest Model

```
rf = RandomForestClassifier(n_estimators=1000, criterion='gini', max_depth=None, n_jobs=-1, random_state=0)
```

```
rf.fit(X_train_under, y_train_under)
rf_ypr = rf.predict(test_X)

rf_yprob = rf.predict_proba(test_X)

# Let's evaluate our RF
conf = confusion_matrix(test_y,rf_ypr)
ConfusionMatrixDisplay(conf).plot()
plt.show()

# Calculate the ROC curve points
fpr, tpr, _ = roc_curve(test_y, rf_yprob[:,1]) #just take yprob of positive class

# Save the AUC in a variable to display it. Round it first
auc = np.round(roc_auc_score(y_true = test_y, y_score = rf_yprob[:,1]), decimals = 3)

# Create and show the plot
plt.plot(fpr,tpr,label=f"RF, auc={auc}")
plt.legend(loc=4)
plt.show()
```

The results are displayed below.

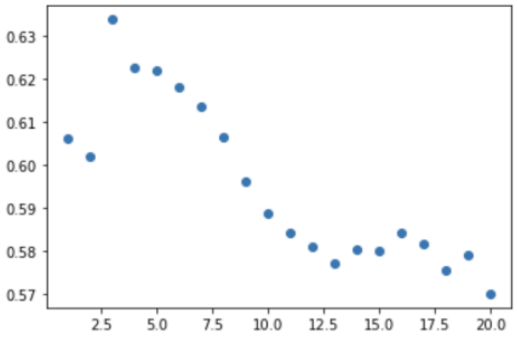


2.2.4 Decision Tree Model

Compared to Random Forest Model, most other models require extra tuning. For instance, in the decision tree model, we need to adjust its depth in order to increase the model's accuracy.

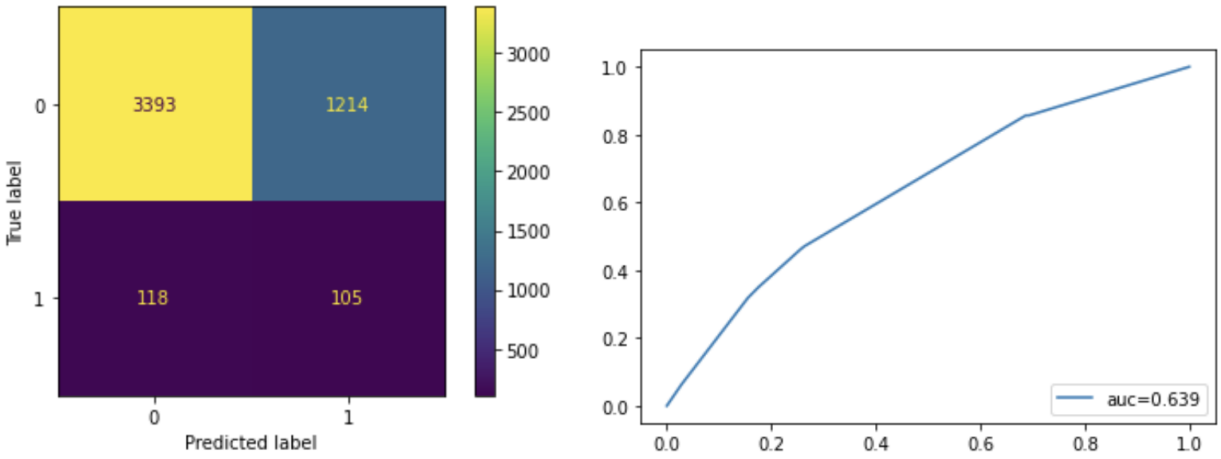
```
# We now create and fit trees from max depth 1 to max depth 20.
# The purpose is to compute accuracy score and the scoring method is cross validation.

# #train_X, test_X, train_y, test_y
from sklearn.pipeline import Pipeline
a=[]
b=[]
for i in range(1,21):
    M = Pipeline([('dt',DecisionTreeClassifier(max_depth = i))])
    acc = cross_val_score(M,X_train_under,y_train_under,cv=5).mean()
    a.append(i)
    b.append(acc)
    print(f'CV classification accuracy with depth {i}:{acc}')

# Plot the CV score and maximum depth
plt.scatter(a,b)

# Print the optimal max depth for the decision tree
print(f'The optimal depth is {b.index(max(b))+1}')
```

```
CV classification accuracy with depth 1:0.6060846560846562
CV classification accuracy with depth 2:0.6018518518518519
CV classification accuracy with depth 3:0.6338624338624339
CV classification accuracy with depth 4:0.6227513227513227
CV classification accuracy with depth 5:0.621957671957672
CV classification accuracy with depth 6:0.6182539682539682
CV classification accuracy with depth 7:0.6137566137566137
CV classification accuracy with depth 8:0.6066137566137566
CV classification accuracy with depth 9:0.5962962962962962
CV classification accuracy with depth 10:0.5888888888888889
CV classification accuracy with depth 11:0.5841269841269842
CV classification accuracy with depth 12:0.5809523809523809
CV classification accuracy with depth 13:0.5772486772486772
CV classification accuracy with depth 14:0.5804232804232805
CV classification accuracy with depth 15:0.5801587301587301
CV classification accuracy with depth 16:0.5843915343915345
CV classification accuracy with depth 17:0.5817460317460317
CV classification accuracy with depth 18:0.5756613756613758
CV classification accuracy with depth 19:0.5791005291005291
CV classification accuracy with depth 20:0.5701058201058202
The optimal depth is 3
```



```
DTCtree = DecisionTreeClassifier(max_depth=3)
DTCtree.fit(X_train_under, y_train_under)
ypr = DTCtree.predict(test_X)
yprob = DTCtree.predict_proba(test_X)
# Plot confusion and ROC, compare with RF
conf = confusion_matrix(test_y,ypr)
ConfusionMatrixDisplay(conf).plot()
plt.show()
# Calculate the ROC curve points
fpr2, tpr2, _ = roc_curve(test_y, yprob[:,1]) #just take yprob of positive class

# Save the AUC in a variable to display it. Round it first
auc2 = np.round(roc_auc_score(y_true = test_y, y_score = yprob[:,1]), decimals = 3)

# Create and show the plot
plt.plot(fpr2,tpr2,label=f"auc={auc2}")
plt.legend(loc=4)
plt.show()
```

Here is the result.



The AUC score is 0.639.

2.2.5 XGBoost Model

Finally, our last model is the XGBoost Model. The first step is to tune the model using GridSearchCV.

```
param_grid = dict({'n_estimators': [50, 1000],'max_depth': [1,10],'learning_rate' : [0.01, 0.1]})
gb = XGBClassifier()
grid = GridSearchCV(gb, param_grid, cv = 5, scoring = 'roc_auc', n_jobs = -1)
grid.fit(X_train_under,y_train_under)
```

GridSearchCV
estimator: XGBClassifier
XGBClassifier

```
display(grid.best_params_)
```

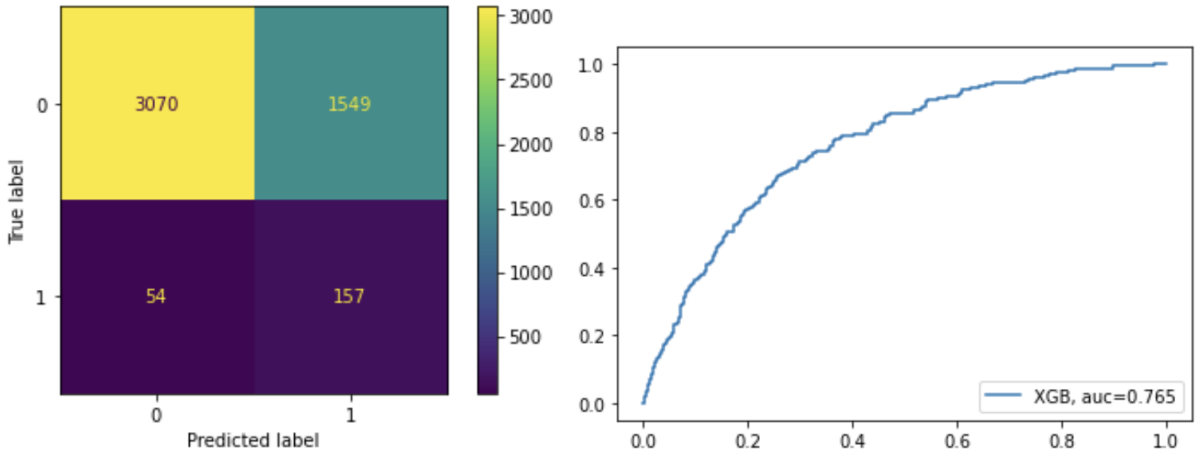
```
{'learning_rate': 0.1, 'max_depth': 1, 'n_estimators': 1000}
```

Then, we implement the model.

```
gbF = XGBClassifier(max_depth=1, learning_rate=0.1,
                    n_estimators=1000, booster='gbtree',n_jobs=1, gamma=0.001, seed = 166)
```

```
gbF = gbF.fit(X_train_under,y_train_under)
ypr = gbF.predict(test_X)
yprob = gbF.predict_proba(test_X)

# Plot confusion and ROC, compare with RF
conf = confusion_matrix(test_y,ypr)
ConfusionMatrixDisplay(conf).plot()
plt.show()

# Calculate the ROC curve points
fpr2, tpr2, _ = roc_curve(test_y, yprob[:,1]) #just take yprob of positive class

# Save the AUC in a variable to display it. Round it first
auc2 = np.round(roc_auc_score(y_true = test_y, y_score = yprob[:,1]), decimals = 3)

# Create and show the plot
plt.plot(fpr2,tpr2,label=f"XGB, auc={auc2}")
plt.legend(loc=4)

plt.show()
```

Here is the result.



2.3 Result

Comparing the AUC scores from three models, we decided to go with the XGBoost Model. We first read the holdout dataset and implement necessary treatments to make it in the same format as the trained dataset. Next, we made our prediction.

```
df_holdout = pd.read_csv('2022_Competition_Holdout.csv')
df_holdout.drop(columns=_60_pct_missing_cols_list, inplace=True)
df_holdout = pd.get_dummies(df_holdout)
id = df_holdout.id
X_H = df_holdout.drop(['id'], axis=1).select_dtypes(exclude=['object']).values
```

```
/Users/ryanli/opt/miniconda3/lib/python3.9/site-packages/IPython/core/interactiveshell.py:3457: DtypeWarning: Columns
(760) have mixed types.Specify dtype option on import or set low_memory=False.
  exec(code_obj, self.user_global_ns, self.user_ns)
```

```
(12220, 906)
```

```
hold_out_pre_prob = gbF.predict_proba(X_H)
```

Next, we compute the attribute importance.

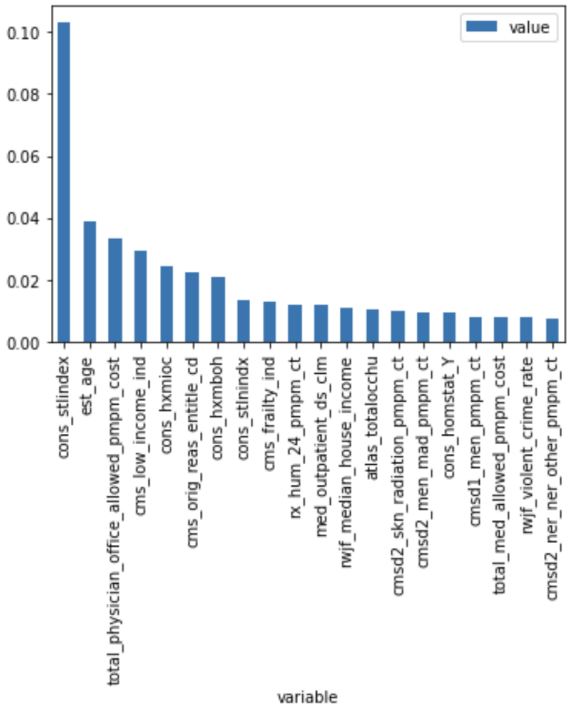
```
def get_xgb_imp(xgb, var_n):
    from numpy import array
    imp_vals = xgb.booster().get_fscore()
    imp_dict = {var_n[i]:float(imp_vals.get('f'+str(i),0.)) for i in range(len(var_n))}
    total = array(imp_dict.values()).sum()
    return {k:v/total for k,v in imp_dict.items()}

attributes = list(X.columns)
gbF.feature_importances_
```

```
array([0.        , 0.00214858, 0.        , 0.        , 0.00276989,
       0.        , 0.        , 0.        , 0.        , 0.00255138,
       0.00269761, 0.        , 0.        , 0.        , 0.00194141,
       0.        , 0.00607338, 0.00280546, 0.        , 0.        ,
       0.        , 0.00204023, 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.0022987 ,
       0.00424141, 0.00414258, 0.        , 0.        , 0.        ,
       0.0021239 , 0.        , 0.        , 0.        , 0.00252758,
       0.00272304, 0.        , 0.00234424, 0.00242476, 0.00264147,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.00270261, 0.        , 0.        , 0.00243505, 0.        ,
       0.        , 0.        , 0.00289909, 0.00349762, 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.00258933,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.00203431, 0.0030494 , 0.        , 0.00212213, 0.        ,
```

Print and plot the top 20 influencing variables.

```
df_imp = pd.DataFrame([gbF.feature_importances_], columns = attributes)
df_imp = pd.melt(df_imp, value_vars=attributes)
df_imp = df_imp.sort_values(by=['value'],ascending=False)
df_imp = df_imp[:20]
print(df_imp[:20])
plt = df_imp.plot.bar(x='variable')
```

	variable	value
393	cons_stlindex	0.103258
212	est_age	0.039068
678	total_physician_office_allowed_pmpm_cost	0.033453
526	cms_low_income_ind	0.029186
100	cons_hxmioc	0.024354
390	cms_orig_reas_entitle_cd	0.022474
197	cons_hxmboh	0.020895
281	cons_stlnindx	0.013378
715	cms_frailty_ind	0.012898
413	rx_hum_24_pmpm_ct	0.012174
399	med_outpatient_ds_clm	0.011828
741	rwjf_median_house_income	0.010883
392	atlas_totalocchu	0.010444
719	cmsd2_skn_radiation_pmpm_ct	0.009751
767	cmsd2_men_mad_pmpm_ct	0.009555
876	cons_homstat_Y	0.009464
760	cmsd1_men_pmpm_ct	0.008097
401	total_med_allowed_pmpm_cost	0.008085
402	rwjf_violent_crime_rate	0.007794
604	cmsd2_ner_ner_other_pmpm_ct	0.007343



Finally, we print out the predictability of the instances in the holdout dataset becoming house insecure and convert it into an excel file.

```
Prob_df = pd.DataFrame(hold_out_pre_prob)
Index_Prob_df = pd.concat([id, Prob_df], axis=1)
Index_Prob = Index_Prob_df.drop(0,axis=1)
Index_Prob
```

	id	1
0	100093066.0	0.397038
1	100313000.0	0.511809
2	100330875.0	0.454670
3	100346385.0	0.301110
4	100443164.0	0.285797
...	...	...
12215	999896529.0	0.475475
12216	999899065.0	0.650275
12217	999946210.0	0.408426
12218	999984658.0	0.470805
12219	999995805.0	0.616485

12220 rows × 2 columns

```
Index_Prob['rank'] = Index_Prob[1].rank(ascending = False)
```

```
Index_Prob.to_excel("CC_output.xlsx")
```

3. Analysis Summary

3.1 Impacting Factors

3.1.1 CONS_STLINDEX

Our first most impactful factor is CONS_STLINDEX which represents the short-term loan index. It is a demographic-based analytical model which predicts the likelihood someone in the household has applied for a short-term loan. The higher index, the higher likelihood. This attribute has a coefficient of 0.103258, and it is the only coefficient above 0.10. In our training dataset, the max value is 9 and the minimum value is 0. With an average of 6.65 and 78% of members having a predicted value above 5, most members are predicted to have a high possibility of applying for a short-term loan. Especially the average interest rate for a short-term loan is around 8% to 13% which is higher than the 7.21% 30-year average fixed mortgage annual percentage rate and 6.48% average 15-year fixed mortgage annual percentage rate. With a high-interest rate and relatively low income (one-quarter of all Medicare beneficiaries live on incomes below $17,000 per person in 2019), it is less likely that they would pay off their debt which might cause them to result in insecure housing issues.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Number of Members	35	190	309	451	672	740	1342	1640	2029	2107
Numer of Insecured members	16	47	93	108	137	168	226	207	161	130
Ratio %	45.71	24.74	30.10	23.95	20.39	22.70	13.84	12.62	7.93	6.17

Table 1 Summary statistics for the short-term loan index column

3.1.2 EST_AGE

Estimated age of the MAPD members ranked second in terms of impacting score. This column was calculated using estimated birthdays relative to the score/index date. The data has a range of 25-100, averaging 62.5, with age 78 having the highest count of 2887. Strategically dividing the age into working age and non-working age, using the retirement age of 64 as the benchmark. 85.6 percent of the members are above 64, and the other 14.4 percent is attributed to ages 25-64. At first glance, it is easy to notice the dominating majority belongs to the legal retirement age group. The reason can be due to diminishing working ability and a ruthless job market that prioritized younger and "healthier" workers. Moreover, given the higher probability of an older individual having illness and other health-harming conditions, it further jeopardizes the income level of that individual, thereby increasing the odds of housing insecurity. Since elders have more health conditions, more doctor visits, and routine check up are needed to provide supplements and medication to ease those conditions. This naturally takes up time and energy, but when these elders are faced with housing insecurity, it can be easy to neglect these actions and instead focus on a safe stay. Referencing Maslow's pyramid theory, the priority of mental and physical health lies below that of environmental safety, further proving how housing insecurity can drastically affect one's wellbeing. A study from Unity Health care(Jabfm 2019 04 180374) conducted research on the effect of housing insecurity on access to preventative and primary care, with results showing an association between the two. For groups in the 25-64 age bracket, the reasons can be largely similar. Reasons like unemployment, investment mistakes, high-risk behaviors (substance abuse), and other factors lead to bankruptcy. As the 20s and 30s are normally the ages to build wealth and support, if this process is struck with stagnation, it can be hard to reach housing security. Younger age housing insecurity can also become a causative factor for young children. Recent articles suggest housing insecurity is affecting young children, sometimes even really young children. And unfortunately, children's housing insecurity is directly influenced by their families' income situation and background, this can be more of an issue for bigger families. (https://ajph.aphapublications.org/doi/full/10.2105/AJPH.2011.300139)

Age bracket	**<20**	**20-30**	**30-40**	**40-50**	**50-60**	**60-70**	**70-80**	**80-90**	**90-100**
Number of	0	31	109	444	2184	14735	21764	8169	864

Members									
Percentage %	0	0.06	0.23	0.92	4.52	30.51	45.06	16.91	1.79
Number of housing Insecurity	0	1	9	47	178	883	757	221	22
Ration %	0	3.23	8.26	10.59	8.15	5.99	3.48	2.7	2.55

Table 2 Summary statistics for the age column

3.1.3 Total_physician_office_allowed_pmpm_cost

Total allowed pmpm cost was denoted as allowed cost per month for overall claims, in this column, specifically related to physician visits. The column of data had an average number of 55.157, with a minimum of 0 and a maximum of **number.**Surprisingly, the minimum number 0 had 9487 counts, which make up 19.64% of members. This demonstrates there is a good chance 19.64 percent of people don't have allowed costs, most likely pointing to an absence of insurance. To understand this further, this can be categorized as actively choosing not to be insured and not having the choice of being insured. The difference lies in having the ability to make that choice. However, it seems unlikely one would choose not to be insured since usually it is offered with employment. Thus, having one-fifth of members not available insurance can strongly discourage them from seeking medical advice. Also, out of the housing insecurity members, 64.92% of them had total allowed costs that were below average. Showing strong evidence of how having more costs allowed for claiming can make a difference. A discovery of 58.89 percent of members having less than the average total allowed cost is also important to observe, this implies there are rather large numbers within these members, most noticeably 528.02 dollars. This can be proven by the large variance of 3204.70 and a high standard deviation of 56.61. To put all these numbers into context, a reference from the Agency for Healthcare Research and Quality had shown two charts comparing average costs for doctor visits with and without insurance, categorized by types, illustrated by **the two tables below.** It can be inferred that without insurance, there can be almost a 60 - 100 dollar difference. With the average total allowed cost from the MAPD members, even 55 dollars allowed cost can help out a lot. With this it can be reiterated that physician visits can be discouraging, if the individual is experiencing housing insecurity, it can further discourage them from it. Simply because extra income has to be allocated to another part, whereas environmental safety hasn't even settled for them. Additionally, only 17.15 percent had total allowed costs of over 100 dollars in claims, which gave these members a good reason to have physician visits and focus on health-related issues.

Specialty type	Medicaid	Medicare	Private Insurance
Overall	$83	$112	$130
Primary care	$79	$104	$119
Pediatric	$82	-	$125

Table 3 Average estimated cost based on specialty type with insurance

Specialty type	Cost without insurance
Average	$265
Primary care	$186
Pediatric	$169

Table 4 Average estimated cost based on specialty type without insurance

3.1.4 cms_low_income_ind

CMS stands for the Centers for Medicare/Medicaid Services, it is an operating division within the Department of Health and Human Services. CMS offers low-income subsidy(LIS), regarding this CMS has written documents on Resource and Cost-Sharing Limits for LIS and Guidance to states on the LIS, which are both referenced for this part of the analysis[]. The data of this column consists of 0 and 1 indicating whether the members receive a subsidy, 7839 members that are receiving LIS, making up 0.1623 percent of all the members. The subsidy provides support for prescribed drugs and health care coverage. The eligibility criteria compose of four components and lengthy reading articles. Most revolve around the Supplemental Security Income program to calculate the income status of the individual. Now, what do these mean in a housing insecurity matter? By definition, the members receiving the subsidy are low-income individuals or families. And low income can lead to a tight spending budget, which is more likely to lead to insufficient allocations toward housing. However, as mentioned the qualifications and lengthy readings have a small chance of miscategorizing or accidentally denying subsidy for some. Although this scenario is unlikely, this variable's column can be somewhat affected and misjudged. Extending this thought, the CMS does have resource limits and can't fully subsidize some people, which depends on that person's income, which may

raise the likelihood of housing insecurity. Overall, the data does support having subsidized prescriptions can help ease the housing insecurity issue, as nearly 70 percent of the housing insecure members are not receiving CMS subsidies.

	0(Non-subsidized)	**1(Subsidized)**
Number of Members	33,359	13,030
Number of housing insecurities	1478	640
Percent % (Insecure/ total insecure)	69.78	30.22

Table 5 Summary statistics for the cms subsidy index column

3.1.5 Cons_hxmioc

Managing Illness or condition index gives information on the likelihood of the member self-monitoring one's health through different methods. This Index consists of a range of null to 9, these nulls are switched to 0 for ease of analysis, and the average is 5.59. And 33.0 percent of the members have an index of below average, within that 23.6 percent are null values. On the other end, the 8 and 9 indexes make up 44.3 percent of the members. In a sense the data is skewed towards a higher value of the index, rendering this column with a certain level of bias. Although this appears biased, the lower index values do have higher percentages of housing insecurity members, with 40 percent of them having an index of null, further showing evidence towards a low monitor level can indicate a higher housing insecurity possibility.Being able to self-monitor health includes time availability, internet accessibility, and health awareness. A high index value could show a higher possibility of the member having these conditions satisfied. However, with a null or low index value, although not all of them are a result of these conditions, it does have an association. Thus showing a higher possibility of not satisfying these conditions. When time availability is not satisfied, this signals to busy schedule, inflexible work time or time allocated elsewhere; Internet accessibility can elude to a living environment that lacks some functionality, and in extreme cases, not having full access to basic living necessities; As for health awareness, not being aware can be related to it being less of a priority, hence showing a neglecting attitude. Overall, the likelihood of managing health can't be a strong indication of housing insecurity, but when it is combined with other variables like age, demographic it could appear to be more complimentary.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Number of Members	11401	268	617	1623	1252	779	5401	5548	9035	12376

Numer of Insecured members	848	31	49	100	84	43	251	210	215	287
Ratio %	7.44	11.57	7.94	6.16	6.71	5.52	4.65	3.79	2.38	2.32

Table 6 Summary statistics for the managing illness index column

3.1.6 CMS_ORIG_REAS_ENTITLE_CD

As our sixth most impactful factor, CMS_ORIG_REAS_ENTITLE_CD stands for the code which indicates the original reason for entry into Medicare. Our XGBoost model has a coefficient of 0.022474. There is a total of four different values in this attribute to represent the reasons: 0 stands for NULL, 1 stands for Disable, 2 stands for End Stage Renal Disease (ESRD), and 3 stands for both Disable and ESRD. As for a more detailed explanation, ESRD is when someone has permanent kidney failure that requires a regular course of dialysis or a kidney transplant. The average amount charged for a kidney transplant in the U.S. in 2020 is around $442,500, and the average wait time frame for waiting on the waitlist for a kidney transplant is 3-5 years at most centers which could be longer in some parts of the country. Although a member might have the time and money for the kidney transplant process, there are other unpredictable factors such as how well the patient and the kidney are, the blood group, if a member is sensitized with high antibody levels, and the available donors. So members enter into the Medicare program, they would shift more of their resources and attention to their condition. Therefore, the cost of time and money is extremely high for a Medicare member which could lead to an insecure housing situation. From our findings, we could identify a positive correlation between the classification result and the reason for entering into Medicare. In the training dataset, 71.84% of members do not use any Disable or ESRD to enter the Medicare program. There are 28.06% of members entered the Medicare program because of disabilities which is reasonable due to the 61 million disabled adults in the US. Categories 2 and 3 only comprise around 0.1% of the training sample population. As a result, we conclude that even though not major of the portion members have disabled, ESRD or both situations, the positive effect on our result is still significant.

	0(None)	**1(Disable)**	**2(ESRD)**	**3(Both)**
Number of Members	33,359	13,030	15	29

Number of housing insecurities	999	910	2	3
Ratio %	3.00%	6.98%	13.33%	10.34%

Table 7 Summary statistics for the original reason for entry column

4. Recommendations

Based on our findings and analysis, we have come up with recommendations from different perspectives, categorizing them into product, service, information and user experience. Each perspective is taken into consideration of the impacting factors discussed in part 3.

Firstly, the products Humana can provide are essential, as one of the largest insurance companies in the US, ensuring the service can affect many members. Humana can take into consideration of the member's disability and/or ERSD status, helping levitate the insurance price for these members while maintaining high-quality services. Although Humana has a number of chronic kidney disease programs, more affordable plans such as medicare plan K through N should be brought up, targeted tentative programs with less cost and lower premium coordination, dialysis education, and palliative care.

Followed by services Humana can aid in, in the context of medicine coverage plans. Humana can expand its coverage to more convenience stores that offer pharmaceuticals. Currently, there is lots of collaboration between Walmart, with plans like Value Rx, and Premiere Rx plans. However, increasing the number of partners can widen the aid of Humana and reach more members. Furthermore, diversifying to other large department stores can help advertise Humana as a brand and major impactor. Potentially working with large distributor stores like Costco and Target, to aid in other medical or health related events. Humana can take its service impact to the next level, by reaching local communities and medical facilities, collaborating with them to set out voluntary physical check-ups, and psychiatrist consultation events at the fraction of the expensive prices at larger hospitals. Creating opportunities for the members who don't have the ability or condition to regularly visit a hospital, and connecting the local community tighter to increase the overall health of people.

Information is the most impactful knowledge nowadays, but even more so for reliable and accurate information. Humana can have a game-changing role in this. For this section, it will be divided into two group-specific recommendations. On the one

hand, targeted towards the younger members, Humana can hold information sessions and events on the topic of monetary investments, possibly even collaborating with schools. Encourage the younger generation to save money and invest smartly, offering them the insights and advantages of investing in the future. Through interactive and simulated events, let the younger generation realize the importance of investing in one's health and future. Moreover, introduce the topic of loans, explain the parameters and the reality of loans, and offer advice and suggestions on more regulated and reliable sources for loans. On the other hand, more targeted towards members older, with the predicting results and impacting factors, Humana can utilize this information and set thresh holds of admonitory. Sending out notifications through mail or email in advance to potential housing-insecure members, about shelters, government assistance instructions, and other related informed suggestions.

Lastly, the user experience perspective. Humana can start by improving the company website to a more user-friendly interface, showing what customers qualify for, and suggesting optimized products based on the customer's situation. An increase of easy to comprehend information can be added to the website, for example, questions on how to qualify for subsidy services. This improvement can help increase website traffic and customer satisfaction. Making sure the customer's questions and needs are addressed, without prying into lengthy documents and uncertain rules.

5. Conclusion

After carefully tailoring and preparing the data, precise model deciding and research-based analysis, much can be learned from and actions can be taken. Looking closely at the most impactful attributes, Humana can takes these into consideration for future planning. Fully taking advantage of sound predictions to identity members that are more likely to undergo housing insecurity, and be able to help alleviate their situation. More specifically, Humana can follow the perspective of product, service, information and user experience, and extend their influence to aid the housing insecurity admonitory stage and aftermath.

6. Reference

[1]https://www.in2013dollars.com/Housing/price-inflation/1967-to-2022?amount=3000
[2]https://health.gov/healthypeople/priority-areas/social-determinants-health/literature-summaries/housing-instability#cit2
[3]https://link.springer.com/article/10.1111/j.1525-1497.2005.00278.x
[4]https://www.humana.com/about/impact
[5]https://www.statista.com/statistics/1100710/organ-transplantation-costs-breakdown-us/
[6]https://www.kidney.org/atoz/content/transplant-waitlist#:~:text=What%20is%20the%20average%20wait,some%20parts%20of%20the%20country
[7]https://www.inclusivecitymaker.com/disability-statistics-in-the-us/#:~:text=How%20many%20people%20with%20disabilities,or%201%20in%204%20adults
[8]https://www.talktomira.com/post/the-cost-of-a-doctor-visit-without-insurance
[9]https://www.lendio.com/business-calculators/short-term-loan-calculator/
[10]https://www.bankrate.com/mortgages/mortgage-rates/
[11]https://www.kff.org/medicare/issue-brief/medicare-beneficiaries-financial-security-before-the-coronavirus-pandemic/#:~:text=Income%20among%20Medicare%20Beneficiaries&text=In%202019%2C%20half%20of%20all,per%20person%20(Figure%201)
[12]https://www.cms.gov/files/document/fy2022-cms-congressional-justification-estimates-appropriations-committees.pdf
[13]https://www.cms.gov/Medicare/Eligibility-and-Enrollment/LowIncSubMedicarePresCov/Downloads/StateLISGuidance021009.pdf
[14]https://www.cms.gov/files/document/2021-lis-resource-limits-memo.pdf