Capstone Project Report -

Problem Statement

Snap-Shot LLC. is a startup which specializes in building mirrorless digital cameras. The company is planning to introduce a new line of above full frame digital cameras.As the go to market strategy for new line of cameras, company decides to launch a targeted digital advertising campaign, which is a type of digital marketing that targets potential customers based on a wide set of variables, such as demographics, online habits and interests. These variables vary across geographical region , zip codes etc. Senior management of Snap-Shot has come up with a hypothesis based on the historical data the company had and consumer electronics industry, to target users with sound financial background. It was decided to target advertisement to users who have annual income more than certain amount. Getting access to financial data of online user is difficult, so company decide to ask online users few general questions through surveys/questionnaire by providing some online incentives such as free pass or coupon codes etc. Markets department needs a list of these parameters based on the zip code to add in the targeted surveys /questionnaire.

Solution:

Since financial data about online user is difficult to access, It has to be inferred /predict based on information we have . The IP address of online user can be used to lookup an approximate zip code using geo-ip mapping databases/Services which are freely available. Depending upon the demographics of the zip-code, user can be presented with a survey or questionnaire with appropriate question , which will help the team to infer important parameters such as age, education, profession etc.

The American Community Survey (ACS) helps local officials, community leaders, and businesses understand the changes taking place in their communities. It is the premier source for detailed population and housing information about our nation. The data gathered in the community survey is used to conduct various statistical tests and draw inferences which is used to run various community programs related to educations, health and infrastructure. Data is split into training and test dataset in separate csv files, with income as label. Labels are categorized into as +50000 or -50000, representing persons income greater or less than $50000 per year. Each entry in the data set is an independent with multiple categorical features. Dataset represents individuals from one zipcode.

Exploring Data:

1. Dataset Set

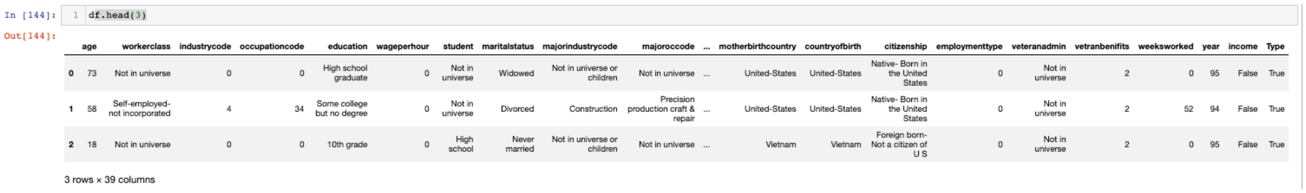
Training set is stored in 'census-income.data' and testing set is stored in 'census-income.test'. The data is set in csv format. Following is the shape of of the data

Shape

```
print(df.shape)
print(df_test.shape)
(152101, 39)
(76320, 39)
```

2. Sample Data:

Each data sample provides details of individual such as age, marital status, education, employment etc. Altogether there are 39 different features which provides multiple options



```
In [144]: df.head(3)
Out[144]:
```

	age	workerclass	industrycode	occupationcode	education	wageperhour	student	maritalstatus	majorindustrycode	majoroccode	...	motherbirthcountry	countryofbirth	citizenship	employmenttype	veteranadmin	vetranbenifits	weeksworked	year	income	Type
0	73	Not in universe	0	0	High school graduate	0	Not in universe	Widowed	Not in universe or children	Not in universe	...	United-States	United-States	Native- Born in the United States	0	Not in universe	2	0	95	False	True
1	58	Self-employed-not incorporated	4	34	Some college but no degree	0	Not in universe	Divorced	Construction	Precision production craft & repair	...	United-States	United-States	Native- Born in the United States	0	Not in universe	2	52	94	False	True
2	18	Not in universe	0	0	10th grade	0	High school	Never married	Not in universe or children	Not in universe	...	Vietnam	Vietnam	Foreign born- Not a citizen of U S	0	Not in universe	2	0	95	False	True

3 rows × 39 columns

3. Datatype of each attribute

Fields and Attributes Types

```
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 199523 entries, 0 to 199522
Data columns (total 42 columns):
age                       199523 non-null int64
workerclass               199523 non-null object
industrycode              199523 non-null int64
occupationcode            199523 non-null int64
education                 199523 non-null object
wageperhour               199523 non-null int64
student                   199523 non-null object
maritalstatus             199523 non-null object
majorindustrycode         199523 non-null object
majoroccode               199523 non-null object
race                      199523 non-null object
hispanicorigin            199523 non-null object
sex                       199523 non-null object
memberofl                 199523 non-null object
unemploymentreason        199523 non-null object
employmentstat            199523 non-null object
capttaingain              199523 non-null int64
capitalloss               199523 non-null int64
stockdivdend              199523 non-null int64
taxfilerstat              199523 non-null object
previousresidence         199523 non-null object
previousstate             199523 non-null object
householdstat             199523 non-null object
householdsummary          199523 non-null object
instanceweight            199523 non-null float64
migrationcodemsa          199523 non-null object
migrationcodechangereg    199523 non-null object
migrationcodemovereg      199523 non-null object
samehouse                 199523 non-null object
migrationsunbelt          199523 non-null object
numberofemployee          199523 non-null int64
familymemberunder18       199523 non-null object
fatherbirthcountry        199523 non-null object
motherbirthcountry        199523 non-null object
countryofbirth            199523 non-null object
citizenship               199523 non-null object
employmenttype            199523 non-null int64
veteranadmin              199523 non-null object
vetranbenifits            199523 non-null int64
weeksworked               199523 non-null int64
year                      199523 non-null int64
income                    199523 non-null object
dtypes: float64(1), int64(12), object(29)
memory usage: 63.9+ MB
```

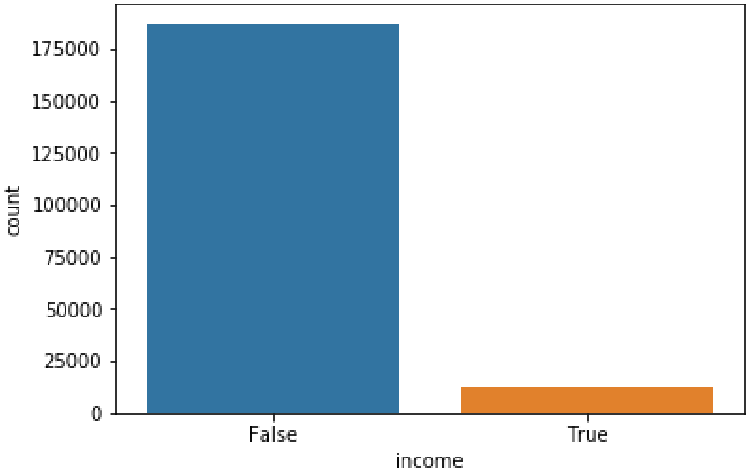
We can see that the input variables are a mixture of numerical and categorical or ordinal data types, where the non-numerical columns are represented using strings. At a minimum, the categorical variables will need to be ordinal or one-hot encoded.

Following transformation are done on the training data set

1. Convert all the fields to categorical type. One-Hot encode categorical variables.
2. Changed the target/labels as True or False , for greater than 50k and less than 50k respectively.
3. There are no missing values in the dataset.
4. Remove financial variables from the dataset such as capital gain/loss, dividends , Race, since they cannot be used in the questionnaire.

4. Distribution of labels

The class distribution is then summarized, confirming a high class imbalance with approximately 94 percent for the majority class (<=50K i.e. False) and approximately 6 percent for the minority class (>50K i.e. True)



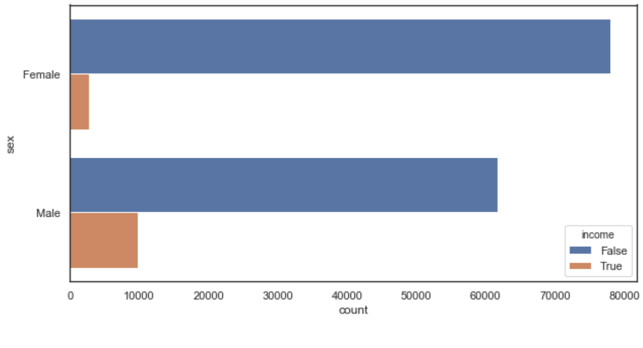
5. Exploring distribution of classes across various categorical variables.

Imbalance in the dataset is clearly seen in almost all the fields in the dataset. From following infographics, we can see that there is no dominant categorical variable which influences the target/class label.

Sex: Looking at the distribution classes over males and females, we have both the sexes

having more and less income from our threshold level.

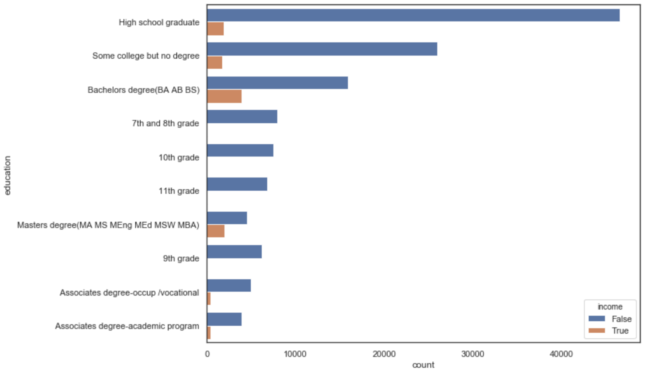
Though males have slightly higher ratio. (True: Above 50k, False: Below 50k).



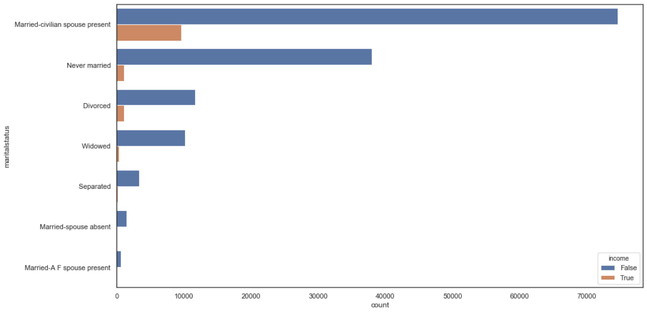
Education: Looking at the distribution classes over various levels of educations, we have spread

of both classes on various education levels, seems that people with higher education have higher

income. Children below 8th grade will be excluded for training the model.

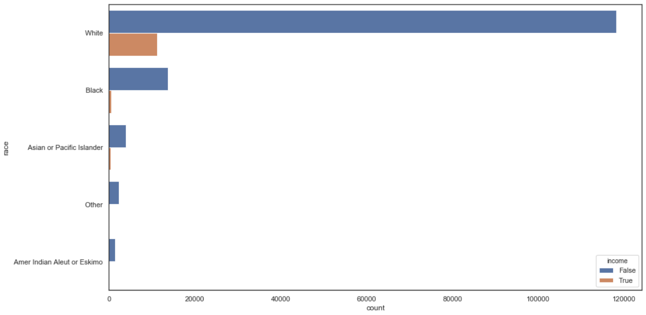


Marital Status: People with some marital status have incomes greater than 50k.

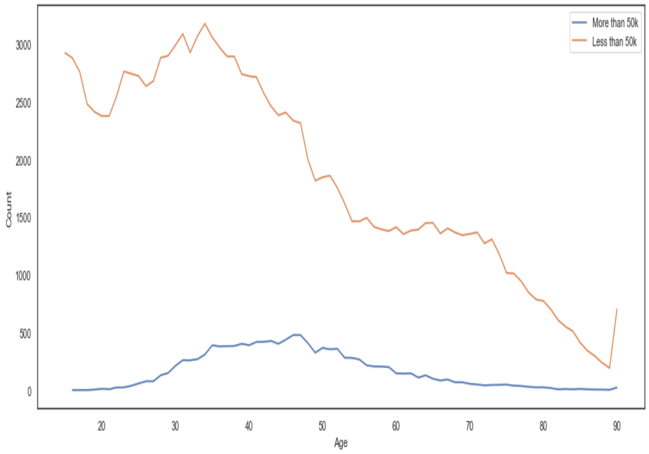


Race: We have a non-equal distribution of classes in people of different race. We will not be using

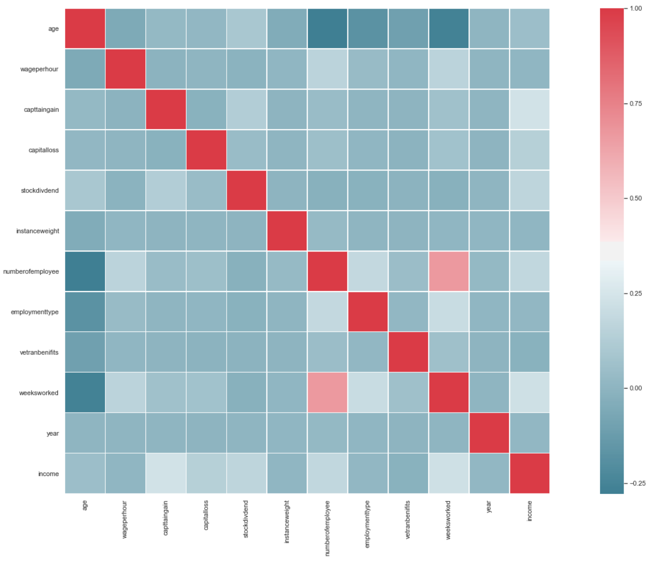
this parameter for our prediction as company has decided not to use this parameter in the questionnaire.



Age:Distribution of people's age in both the classes.



Numerical Features: Correlation between various numerical features:



Training Binary Classifier and Baseline results

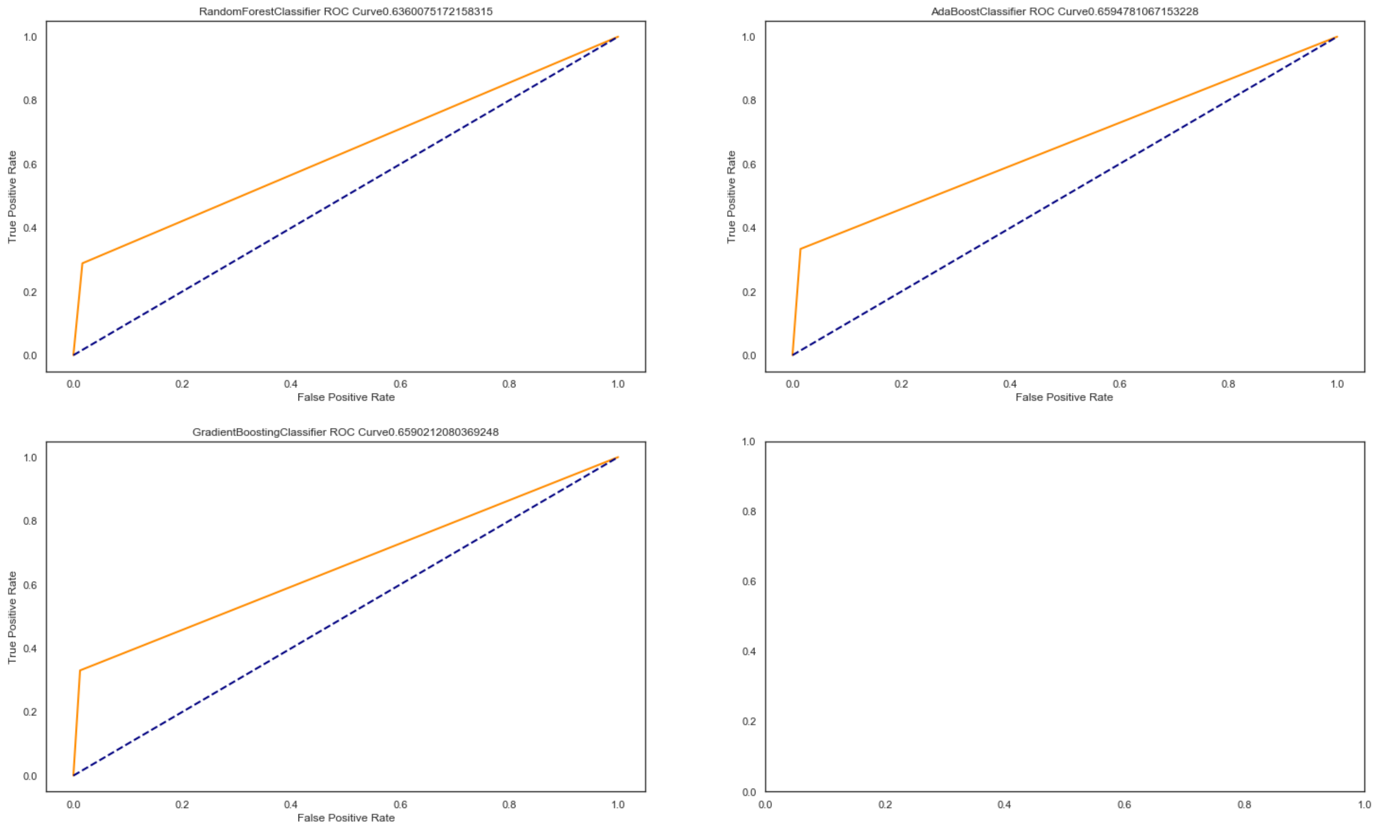
The goal of this exercise is to build a model using the dataset available for a zip code to predict income and also identify most important features of the dataset which influence prediction. Following is high level approach for training a model after the data is cleaned and transformed.

1. Try out multiple ensemble learning, classification algorithm and try out the best performing algorithm with out any hyper parameter tuning.
    1. **RandomForestClassifier**: Random forest is an ensemble model using bagging as the ensemble method and decision tree as the individual model.
    2. **AdaBoostClassifier**: AdaBoost is a boosting ensemble model and works especially well with the decision tree. Boosting model’s key is learning from the previous mistakes, e.g. misclassification data points.AdaBoost learns from the mistakes by increasing the weight of misclassified data points.
    3. **GradientBoostingClassifier:**Gradient Boosting learns from the mistake, residual error directly, rather than update the weights of data points.
2. With Grid-search find best hyper parameter for the algorithm.
3. Choose the final hyper parameters and use the model for prediction.
4. Following metrics are used to evaluate performance of the model.
    1. **Area under ROC curve:** In a ROC curve the true positive rate (Sensitivity) is plotted in function of the false positive rate (100-Specificity) for different cut-off points of a parameter.The area under the **ROC curve** (AUC) is a measure of how well a parameter can distinguish between two diagnostic groups (diseased/normal).

1. Following are the metrics evaluation for selected algorithms

Metrics

```
RandomForestClassifier
Validation Metrics :
F1 Score          :  0.39222807336117665
Precission Score :  0.28931154567372086
Recall Score      :  0.608793686583991
roc_auc_score     :  0.7741508164319731
Confusion Matrix :  [[41204  2653]
 [  694  1080]]
Test Metrics  :
F1 Score          :  0.3909757967363925
Precission Score :  0.2885548011639185
Recall Score      :  0.6061120543293718
roc_auc_score     :  0.7730662486297625
Confusion Matrix :  [[68974  4401]
 [ 1160  1785]]

AdaBoostClassifier
Validation Metrics :
F1 Score          :  0.44396474186004675
Precission Score :  0.330565229038307
Recall Score      :  0.6757940854326396
roc_auc_score     :  0.8093729016365344
Confusion Matrix :  [[41306  2499]
 [  592  1234]]
Test Metrics  :
F1 Score          :  0.44477965736450814
Precission Score :  0.33365664403491757
Recall Score      :  0.6668820678513732
roc_auc_score     :  0.8052949089683632
Confusion Matrix :  [[69103  4122]
 [ 1031  2064]]

GradientBoostingClassifier
Validation Metrics :
F1 Score          :  0.45164815152067017
Precission Score :  0.332172515403161
Recall Score      :  0.7053469852104665
roc_auc_score     :  0.8242619410815172
Confusion Matrix :  [[41380  2493]
 [  518  1240]]
Test Metrics  :
F1 Score          :  0.44970283953334805
Precission Score :  0.33026188166828324
Recall Score      :  0.7044827586206897
roc_auc_score     :  0.824026996308438
Confusion Matrix :  [[69277  4143]
 [  857  2043]]
```



2. From the performance evaluation, GradientBoostingClassifier is selected as the classification algorithm. Following are most important features from the dataset

Model
weeksworked
age
majoroccode__ Executive admin and managerial
majoroccode__ Professional specialty
sex__ Female
sex__ Male
education__ Masters degree(MA MS MEng MEd MSW MBA)
education__ Prof school degree (MD DDS DVM LLB JD)
education__ Bachelors degree(BA AB BS)
education__ Doctorate degree(PhD EdD)

Model Performance Tuning

A. Handling of imbalance data

Most machine learning algorithms work best when the number of samples in each class are about equal. This is because most algorithms are designed to maximize accuracy and reduce error. Following methods can be used to deal with imbalanced dataset

1. **Performance Metrics:**

   Accuracy score in imbalance dataset suffers accuracy paradox and have misleading results. Metrics that can provide better insight include:

   - **Confusion Matrix**: a table showing correct predictions and types of incorrect predictions.
   - **Precision**: the number of true positives divided by all positive predictions. Precision is also called Positive Predictive Value. It is a measure of a classifier's exactness. Low precision indicates a high number of false positives.
   - **Recal**l: the number of true positives divided by the number of positive values in the test data. Recall is also called Sensitivity or the True Positive Rate. It is a measure of a classifier's completeness. Low recall indicates a high number of false negatives.
   - **F1**: Score: the weighted average of precision and recall.

2. **Change the algorithm:**

   while in every machine learning problem, it's a good rule of thumb to try a variety of algorithms, it can be especially beneficial with imbalanced datasets. Decision trees frequently perform well on imbalanced data. They work by learning a hierarchy of if/else questions and this can force both classes to be addressed.

3. **Resampling Technique:**

Resampling (oversampling or under sampling) can be defined as adding more copies of the minority class. Oversampling of data points from minority class. In this example we have oversampled data points with True sample. A slight oversampling of True class can help in getting better results. In our case we have increased the True class sample to 11.3% from 6.0%.

4.**Generate synthetic samples:**

A technique similar to upsampling is to create synthetic samples. One can use imblearn's SMOTE or Synthetic Minority Oversampling Technique. SMOTE uses a nearest neighbors algorithm to generate new and synthetic data we can use for training our model. We will not be creating synthetic samples.

B. HyperParameter Tuning using GridSearchCV

Machine learning models are parameterized so that their behavior can be tuned for a given problem. Models can have many parameters and finding the best combination of parameters can be treated as a search problem. Grid search is an approach to parameter tuning that will methodically build and evaluate a model for each combination of algorithm parameters specified in a grid.

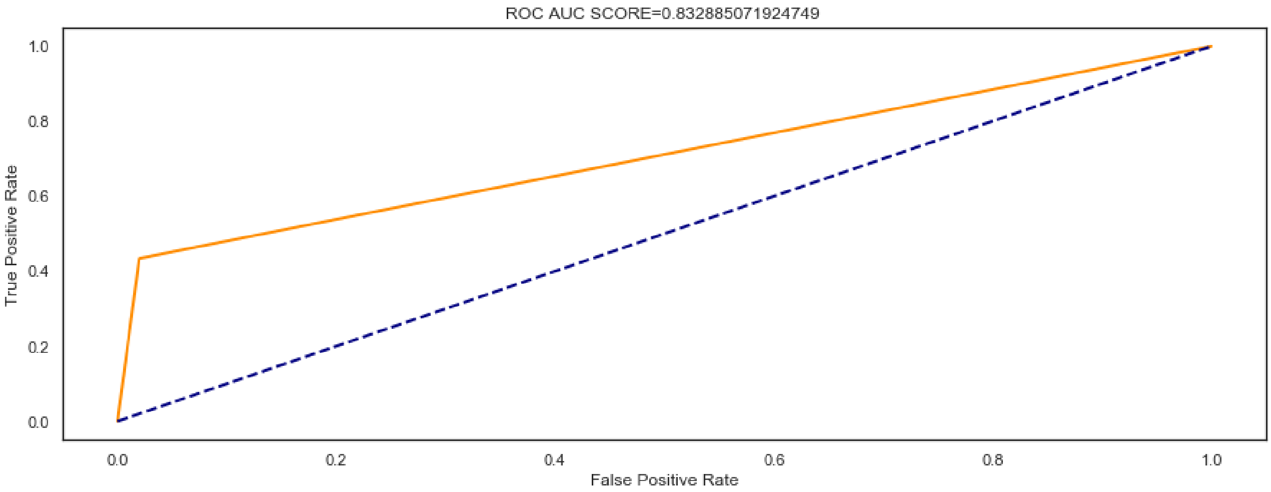
GridSearchCV

```
# Create the parameter grid based on the results of random search
param_grid = {
    'max_features': [2, 3,4],
    'min_samples_leaf': [2,3,4],
    'min_samples_split': [2,3,5],
    'n_estimators': [150,300,1000 ]
}

# Create a based model
rf = GradientBoostingClassifier()
#Scoring
scorings = {'AUC': 'roc_auc', 'F1': 'f1'}


# Instantiate the grid search model
grid_search = GridSearchCV(estimator = rf, param_grid = param_grid, cv =
3, n_jobs = 1,
                               verbose = 2, scoring=scorings, refit='AUC',
return_train_score=True)
```

Grid Search CV finds following best hyper tuning parameters

GridSearchCV

```
F1 Score          :  0.5452583623066779
Precission Score :  0.43405230005726286
Recall Score      :  0.7330754352030948
roc_auc_score     :  0.832885071924749
Confusion Matrix :  [[41088  2965]
 [  828  2274]]
0.5452583623066779
```



Conclusion:

From census data, it can be learned that every zip code is unique and different parameters dominating the social and economic levels of residents. For better results, Snapshot LLC can generate a generic model across the state or group of zip codes, to identifying important parameters affecting residents income level. Imbalance dataset adds to the complexity of the problem and needs some more processing to get unbiased results. We have seen by resampling method we can increase efficiency of the algorithm from _ to _.

Most important features for the current zip code are following, marketing department can use these parameters can try to get this information from the survey or other mechanism.

Further Reading and References:

- Learning from Imbalanced Data - Literature Review
- Learning from Imbalanced Classes
- Learning from imbalanced data: open challenges and future directions
- Handling imbalanced datasets in machine learning
- https://towardsdatascience.com/methods-for-dealing-with-imbalanced-data-5b761be45a18
- https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/
- https://towardsdatascience.com/basic-ensemble-learning-random-forest-adaboost-gradient-boosting-step-by-step-explained-95d49d1e2725