AUTO INSURANCE MARKETING STRATEGY

Christopher Angeles

TABLE OF CONTENTS

Allstate
You're in good hands

OUR OBJECTIVES

Allstate is a auto insurance company located in the United States looking to expand on their marketing strategy

Allstate will be rolling out a strategy starting summer 2020 and wish to optimize their targeting algorithm

Optimize the marketing campaign by categorizing its clientele

Construct a classification algorithm that can be used for the new marketing strategy

Accurately target clients for specific needs, increasing the strength of its marketing campaign

DEMOGRAPHICS

1. Founded in 1931, Allstate is one of the largest insurance companies in the US
2. Allstate offers insurance to over 16 million households
3. Offers home, auto, life, health and retirement plans
4. Majority of its sales are through call centers and the internet
5. Advocate of car safety through reforms including teen driver education

PROCESS & TOOLS

DATA ANALYSIS

Utilizing python, pandas, and dataset provided by kaggle.com

DATA PROCESSING

Data beautifying through data reconfiguration and scaling

MODEL BUILDING

Create an unsupervised model using the sklearn library to cluster our data

METRICS

Put our model through various optimization parameters

DATA EXPLORATION

Customer	Education	Total Claim Amount	Income	Coverage	EmploymentStatus	Monthly Premium Auto	Number of Policies	Policy Type
BU79786	Bachelor	384.811147	56274	Basic	Employed	69	1	Corporate Auto
QZ44356	Bachelor	1131.464935	0	Extended	Unemployed	94	8	Personal Auto
AI49188	Bachelor	566.472247	48767	Premium	Employed	108	2	Personal Auto
WW63253	Bachelor	529.881344	0	Basic	Unemployed	106	7	Corporate Auto
HB64268	Bachelor	138.130879	43836	Basic	Employed	73	1	Personal Auto
...	...	...	...	...	...	...	...	...
LA72316	Bachelor	198.234764	71941	Basic	Employed	73	2	Personal Auto
PK87824	College	379.200000	21604	Extended	Employed	79	1	Corporate Auto
TD14365	Bachelor	790.784983	0	Extended	Unemployed	85	2	Corporate Auto
UP19263	College	691.200000	21941	Extended	Employed	96	3	Personal Auto
Y167826	College	369.600000	0	Extended	Unemployed	77	1	Corporate Auto

Identifying which factors are most important, helps build a model to accurately classify customers

Important Factors

Numerical and categorical factors including:

- Education
- Coverage
- Monthly Premium Payments
- Policy Type

DATASET

DATA PROCESSING

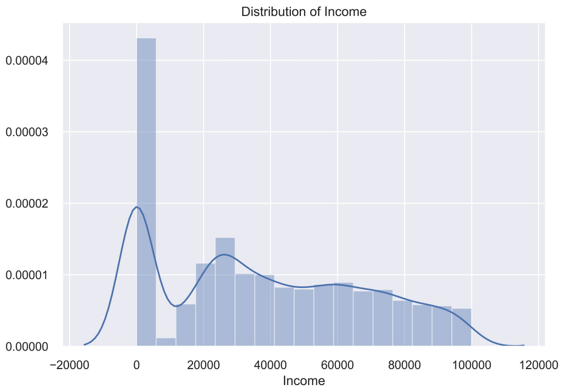
	Total Claim Amount	Customer Lifetime Value	Income	Number of Open Complaints	Monthly Premium Auto
count	9134.000000	9134.000000	9134.000000	9134.000000	9134.000000
mean	434.088794	8004.940475	37657.380009	0.384388	93.219291
std	290.500092	6870.967608	30379.904734	0.910384	34.407967
min	0.099007	1898.007675	0.000000	0.000000	61.000000
25%	272.258244	3994.251794	0.000000	0.000000	68.000000
50%	383.945434	5780.182197	33889.500000	0.000000	83.000000
75%	547.514839	8962.167041	62320.000000	0.000000	109.000000
max	2893.239678	83325.381190	99981.000000	5.000000	298.000000

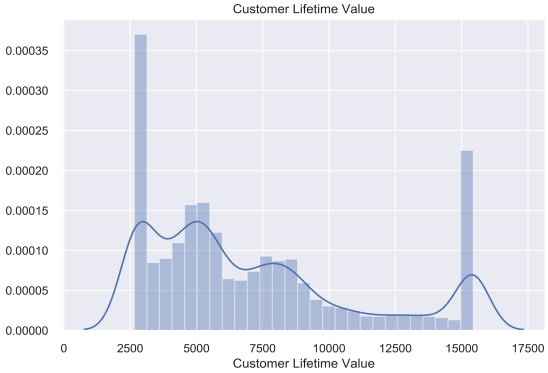
Preparing the dataset to reflect what insights we want to extract from it

Procedure

- Dropped date column
- Changed columns from numerical to categorical - Number of open complaints
- Windorized and standardized numerical columns

DATA SCALING






```
norm_cols = ['Customer Lifetime Value', 'Income',
             'Monthly Premium Auto',
             'Months Since Policy Inception',
             'Number of Policies', 'Months Since Last Claim']

```

```
scaler = StandardScaler()
for i in norm_cols:
    df1[i] = scaler.fit_transform(df1[i].values.reshape(-1, 1))
```

Scaling our data allows us to keep the same distribution of our dataset while allowing us to compare two variables

SCALING

BUILDING THE MODEL

MODEL PIPELINE

BUILD

Create an unsupervised machine model that can cluster our dataset customers

VISUALIZE

Visualize these clusters and analyze similarities within clusters

METRICS

Calculate how well our model is clustering our dataset

TRAIN & TEST

Test how well our model can predict new data points

CLUSTERING METHODS

CLUSTERING

K MEANS CLUSTERING

Aims to partition observations into clusters in which each observation belongs to the cluster with the nearest mean

DB SCAN CLUSTERING

Density based, spatial clustering method that groups together points that are closely packed together

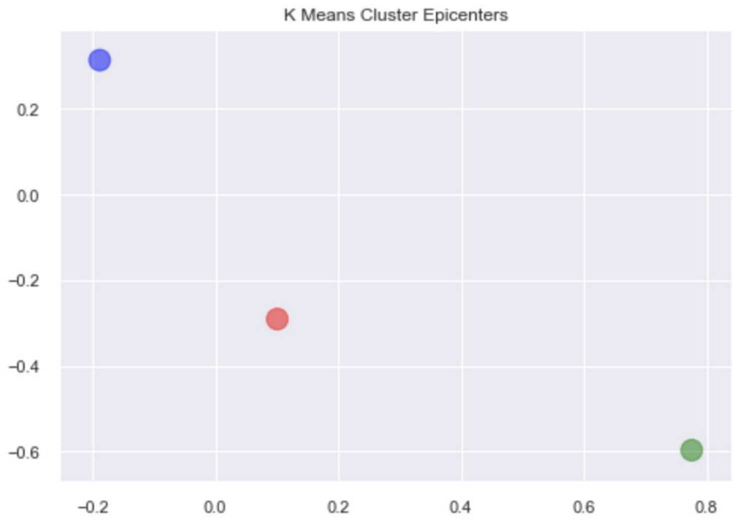
AGGLOMERATIVE CLUSTERING

Also called hierarchical clustering, a bottom-up clustering approach where each observation is assigned its own cluster and by calculating the similarity between clusters, merges them

K MEANS MODEL

```python
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score, davies_bouldin_score
# KMeans Model
X = df
model = KMeans(n_clusters=3)
model.fit(X)
y_pred = model.predict(X)
```

```python
centers = model.cluster_centers_
plt.scatter(centers[:, 0], centers[:, 1],
                    c=['red','blue', 'green'], s=200, alpha=0.5)
plt.title('K Means Cluster Epicenters')
```



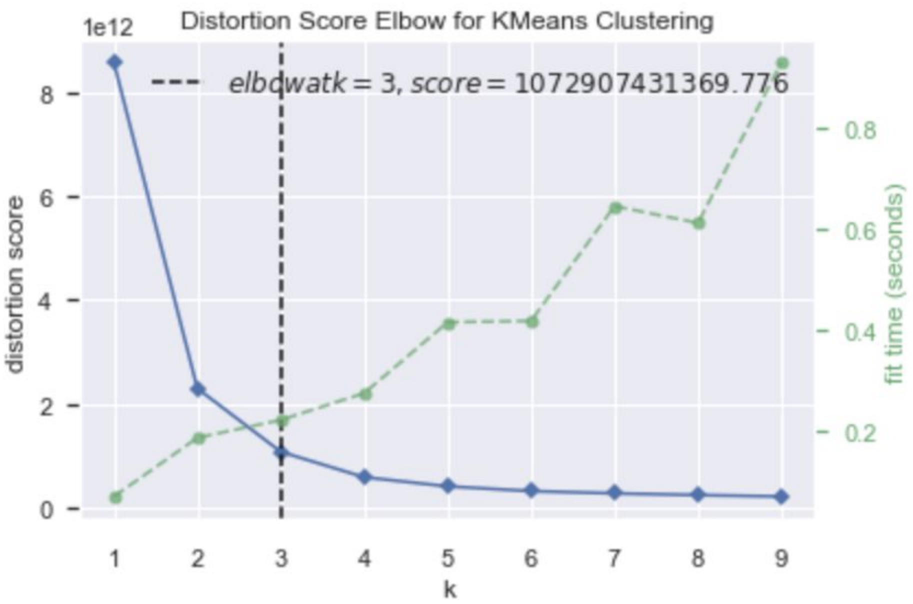
K MEANS MODEL

K Means is an algorithm that tries to partition the dataset into distinct, non-overlapping groups or clusters

K MEANS OPTIMIZATION

Optimization allows our model to perform at its maximum to classify clusters

K Means Elbow is a visualization method that computes the optimal number of clusters for a K Means model within the ranges specified



K MEANS OPTIMIZATION

DB SCAN & AGGLOMERATIVE OPTIMIZATION

Looping through the metrics for DB Scan and Agglomerative clustering models, we can find the optimal parameters for each model

```
for i in range(1,10):
    clustering = DBSCAN(eps=i).fit(X)
    y_pred = clustering.fit_predict(X)
    print('Eps of ', i)
    print('Davies Bouldin Score ', davies_bouldin_score(X,y_pred))
    print('Silhouette Score ', silhouette_score(X,y_pred))
```

...

```
metric_list = ['euclidean','manhattan']
for i in metric_list:
    clustering = DBSCAN(eps=6, metric=i).fit(X)
    y_pred = clustering.fit_predict(X)
    print('Metric ', i)
    print('Davies Bouldin Score ', davies_bouldin_score(X,y_pred))
    print('Silhouette Score ', silhouette_score(X,y_pred))
```

...

```
for i in range(1,10):
    clustering = DBSCAN(eps=6, metric='euclidean', min_samples=i ).fit(X)
    y_pred = clustering.fit_predict(X)
    print('Min Samples ', i)
    print('Davies Bouldin Score ', davies_bouldin_score(X,y_pred))
    print('Silhouette Score ', silhouette_score(X,y_pred))
```

COMPARISON

K Means

```
# KMeans
model = KMeans(n_clusters=3)
model.fit(X)
y_pred = model.predict(X)
print(davies_bouldin_score(X,y_pred))
print(silhouette_score(X, y_pred))

df['Cluster'] = model.labels_

df.Cluster.value_counts()
```

...

DB Scan

```
# DBSCAN
model = DBSCAN(eps=20, metric='euclidean', min_samples=6).fit(X)
y_pred = model.fit_predict(X)
print(davies_bouldin_score(X,y_pred))
print(silhouette_score(X, y_pred))

df['Cluster'] = model.labels_

df.Cluster.value_counts()
```

...

Agglomerative

```
# Agglomeric
model = AgglomerativeClustering(n_clusters=3,linkage='ward', affinity='euclidean')
y_pred = model.fit_predict(X)
#y_pred = single.labels_.astype(np.int)

print(davies_bouldin_score(X,y_pred))
print(silhouette_score(X,y_pred))


df['Cluster'] = model.labels_

df.Cluster.value_counts()
```

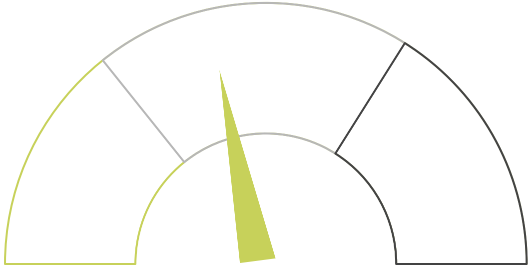
COMPARISON

METRICS

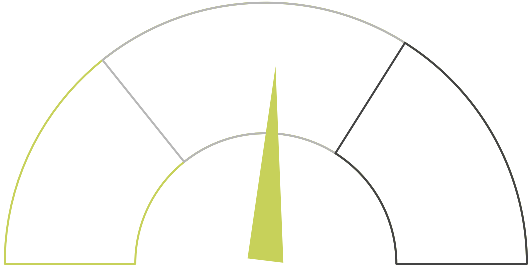
Silhouette Score is a classification metrics that measures how similar a datapoint is to its own cluster compared to all clusters

Dunn Index uses cluster size and intercluster distances to evaluate clustering algorithms

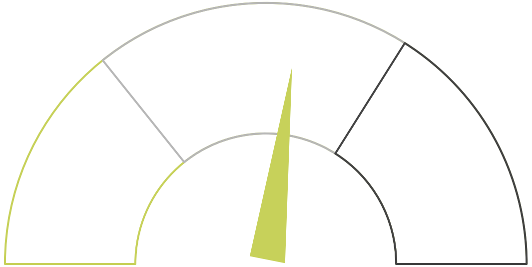
Davies-Bouldin Index uses the ratio of parameters within the cluster to the parameters between clusters to optimize clustering algorithms



Silhouette Score



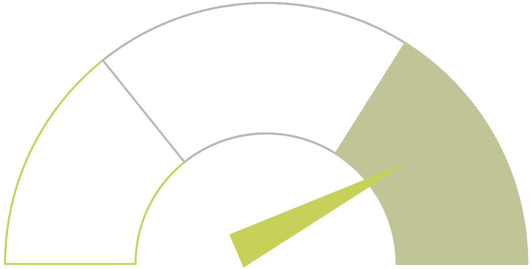
Dunn Index



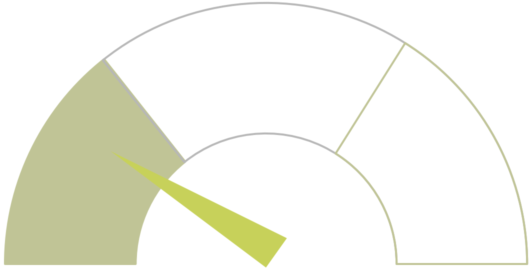
Davies-Bouldin Index

COMPARISON

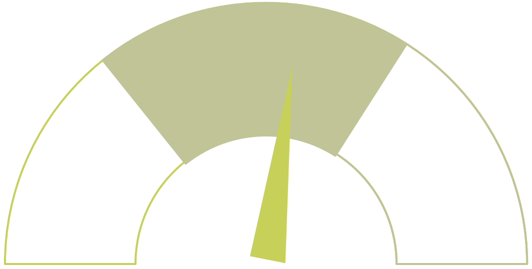
	K Means	DB Scan	Agglomerative
Silhouette Score	0.54	0.68	0.51
Davies - Bouldin Index	0.58	1.64	0.60
Number of Clusters	3	6	3



K MEANS



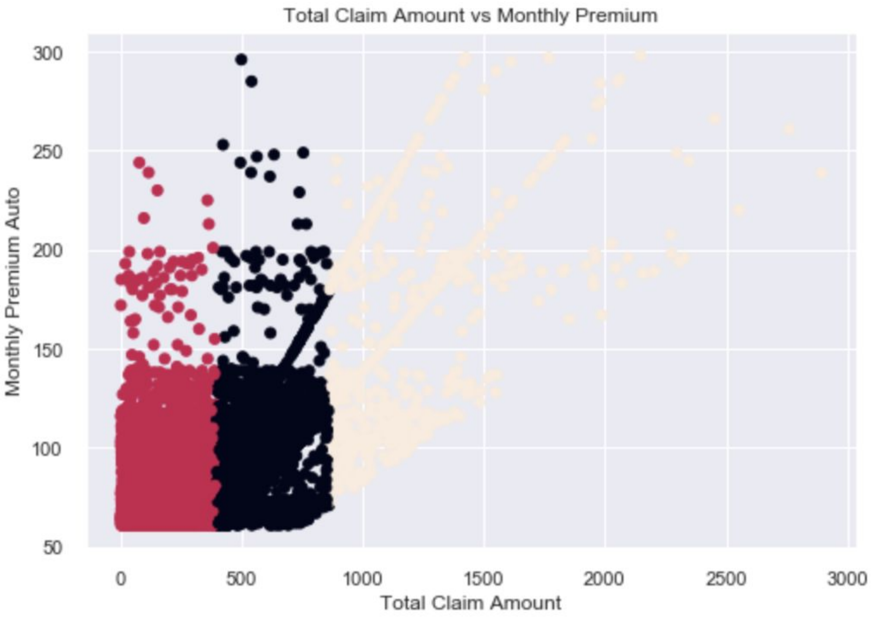
DB SCAN



AGGLOMERATIVE

COMPARISON

K MEANS VISUALIZATION



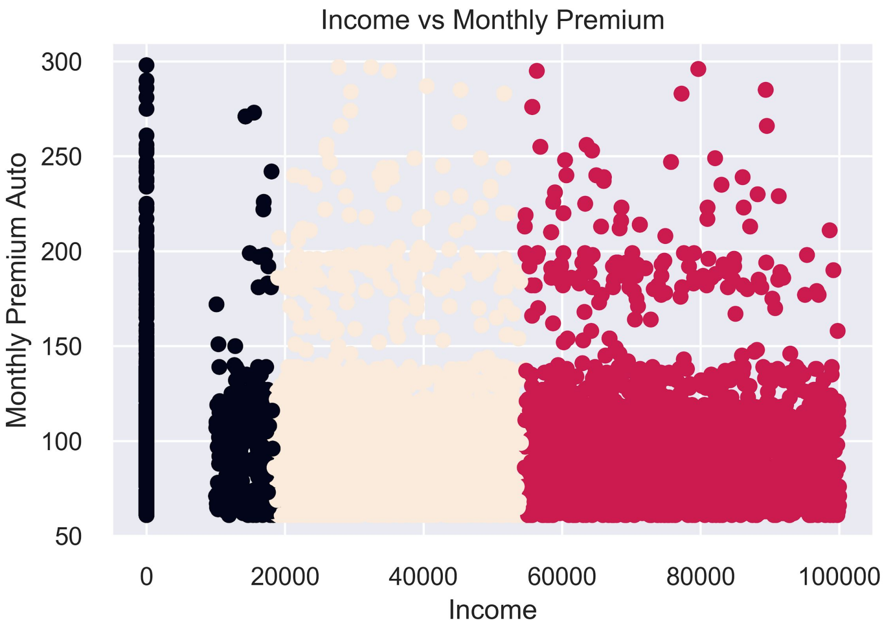
Visualizing our clusters allow us to see how well the model is at creating distinct groupings

Distinguishing different clusters between customers monthly payments and their total claim payments helps us categorize customers

K MEANS VISUALIZATION

Income plays a large role in how much one can afford in auto insurance

Classifying customers who can afford larger monthly premium payments will help us target them in our marketing campaign



CLUSTERING PREDICTION

Hollie Duncan

Cluster 1

Potential Customer

John Patterson

Cluster 2

CLUSTERING PREDICTION

PREDICTION ROADMAP

1 ADD CLUSTERS BACK TO DATA

Add our clustering algorithm back to our data and group customers by our new clusters

2 TRAIN/TEST SPLIT

Split our data into a training set and test set to create new model

3 TEST MULTIPLE MODELS

Test various model predictions

4 RUN & REPORT

Run the best model and report on the accuracy of the model to predict new customers

SUPERVISED MACHINE MODEL

K MEANS OPTIMIZATION

Train/Test Split separates our data into training and test sets so we can build our prediction model

```
X_train, X_test, y_train, y_test = train_test_split(df2.drop('Cluster', axis=1),
                                                    df2.Cluster, test_size=1/3)
```

```
model_list=[KNeighborsClassifier(), GaussianNB(),
            DecisionTreeClassifier(),
            RandomForestClassifier(),
            CatBoostClassifier()]
```

```
l_acc = []
l_cm = []
for model in model_list:
    model2=model.fit(X=X_train, y=y_train)
    y_pred2 = model2.predict(X_test)
    l_acc.append(accuracy_score(y_test,y_pred2))
    l_cm.append(confusion_matrix(y_test,y_pred2))
    print(type(model2).__name__, ' is done')
```

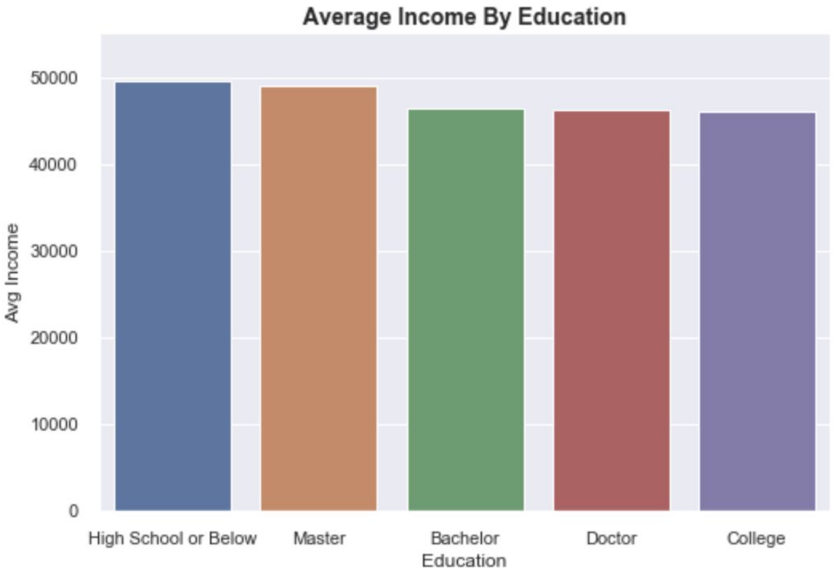
	0	1
1	GaussianNB	0.829885
3	RandomForestClassifier	0.99803
0	KNeighborsClassifier	0.999343
4	CatBoostClassifier	0.999672
2	DecisionTreeClassifier	1

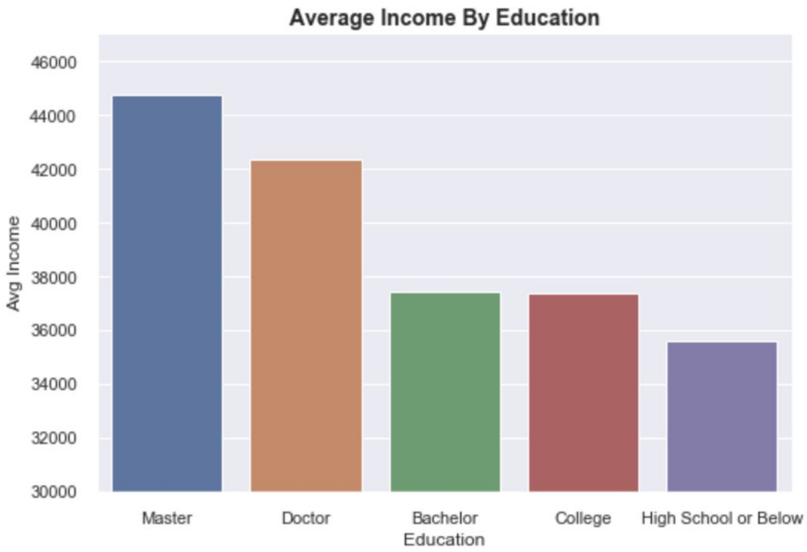
Accuracy of our models to predict and classify new customers is 99%, allowing us to configured this model to new data

STRATEGY & INSIGHTS

Utilize the model to look at trends within our clusters

CLUSTERING TRENDS



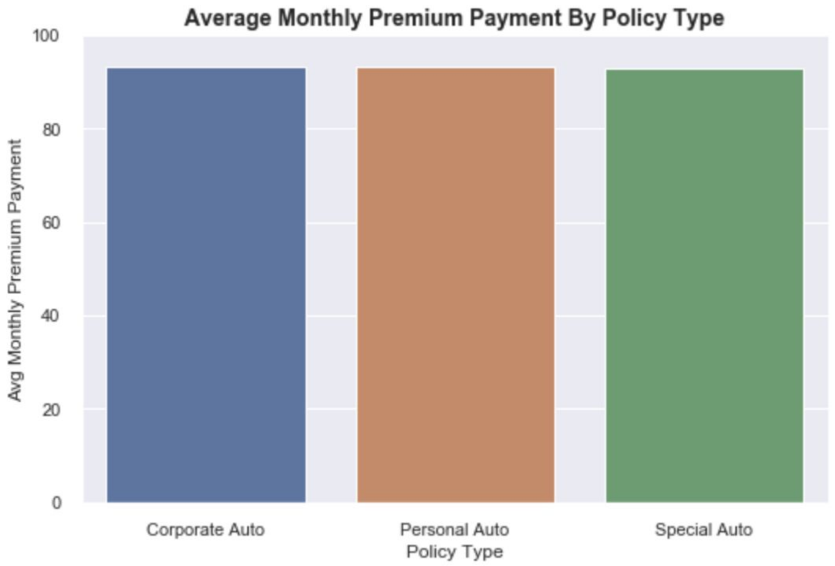


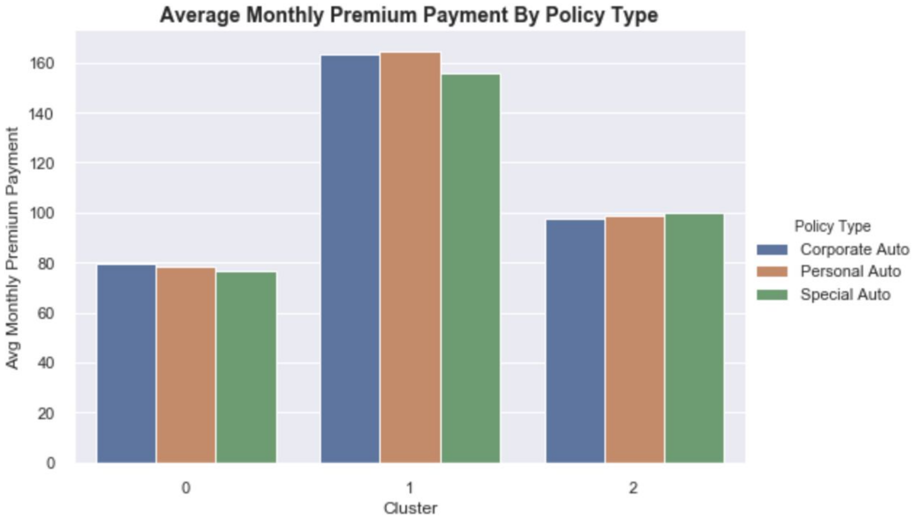
Comparing our clustering algorithm versus the average income by education, we can see our targeted audience is different

INCOME CLUSTERING

CLUSTERING TRENDS

POLICY TYPE CLUSTERING





Target audience by premium payment by what type of insurance is enhanced with our clustering algorithm, allowing specific targeting for our marketing strategy

STRATEGY

Use clustering algorithm for target audiences

MODEL

Create a clustering model to group our target audience

OPTIMIZATION

Optimize model and allow for new data to be added and clustered

INCOME

MONTHLY PREMIUM

EDUCATION

POLICY TYPE

STRATEGY

THANKS

DRIVERS EDUCATION

- Visualization really helps see your clusters
- Data processing can make the difference between a good clustering model and one that makes every datapoint a cluster
- You have to optimize every model as they are all different
- Pivot! Pivot! Pivot! It's a great way to see how your model clusters your data