**Final Business Report**

**Table of Contents:**

1. Introduction --------------------------------------------------------------------------------------------------2

Brief introduction about the problem statement and the need of solving it.

2. EDA and Business Implication ----------------------------------------------------------------------3 to 18

- Both visual and non-visual understanding of the data.

- Uni-variate / Bi-variate / multi-variate analysis to understand relationship b/w variables. How your analysis is impacting the business

3. Data Cleaning and Pre-processing ---------------------------------------------------------------19 to 23

- Approach used for identifying and treating missing values and outlier treatment (and why)

- Need for variable transformation (if any)

- Variables removed or added and why (if any)

4. Model building ---------------------------------------------------------------------------------------23 to 29

- Clear on why was a particular model(s) chosen.

- Effort to improve model performance.

5. Model validation --------------------------------------------------------------------------------------------29

- How was the model validated? Just accuracy, or anything else too?

6. Final interpretation / recommendation----------------------------------------------------------------30

- Detailed recommendations for the management/client based on the analysis done.

**Solution: -**

**1. Introduction**

**Brief introduction about the problem statement and the need of solving it.**

Problem Understanding:

a) Defining problem statement

The dataset belongs to a leading life insurance company. The company wants to predict the bonus for its agents so that it may design appropriate engagement activity for their high performing agents and upskill programs for low performing agents

b) Need of the study/project

The need of the project is to identify high performing agents and design appropriate engagement activity and design learning provide for low performing agents so that they can upskill.

c) Understanding business/social opportunity

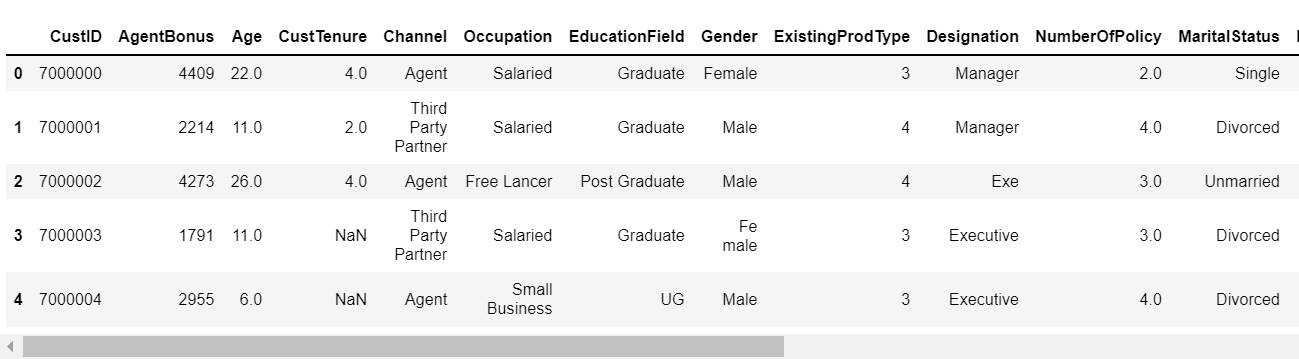
This business opportunity of this project is to identify agent performance, upskill their customer engagement strategy, increase their sales, for customer analytics, for campaign management, for identifying their best customers etc.

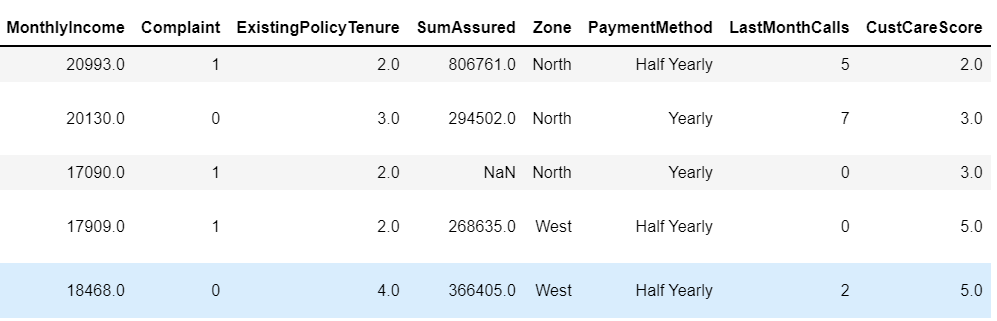
**2. EDA and Business Implication**

a) Understanding how data was collected in terms of time, frequency and methodology

1. Data is collected over a period of year or more in monthly, quarterly, half yearly and yearly frequency.
2. Data is also collected of customer from all four parts of India i.e. North, South, East and West
3. Targeted customer features are age, occupation, designation, monthly income.
4. Policies sold are from three channels agent, online and third party.

b) Visual inspection of data (rows, columns, descriptive details)





Shape of the dataset –

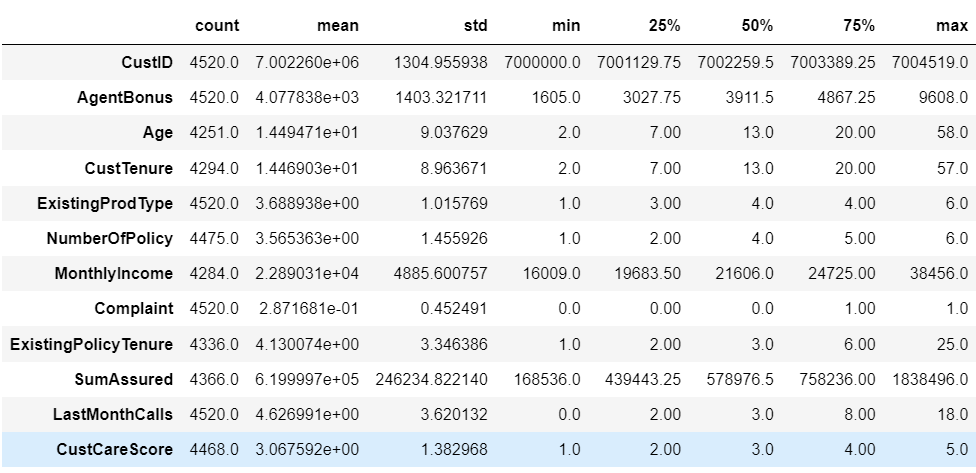
No. of rows: 4520

No. of columns: 20

Below are the Numerical columns

'CustID', 'AgentBonus', 'Age', 'CustTenure', 'ExistingProdType', 'NumberOfPolicy', 'MonthlyIncome', 'Complaint', 'ExistingPolicyTenure', 'SumAssured', 'LastMonthCalls', 'CustCareScore’

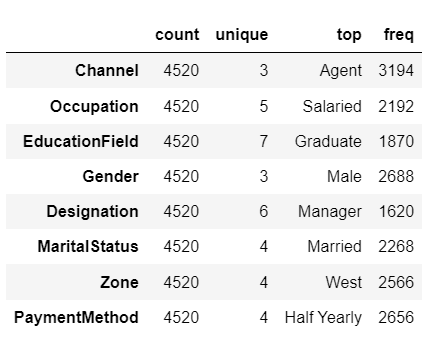
Descriptive details of Numerical columns –



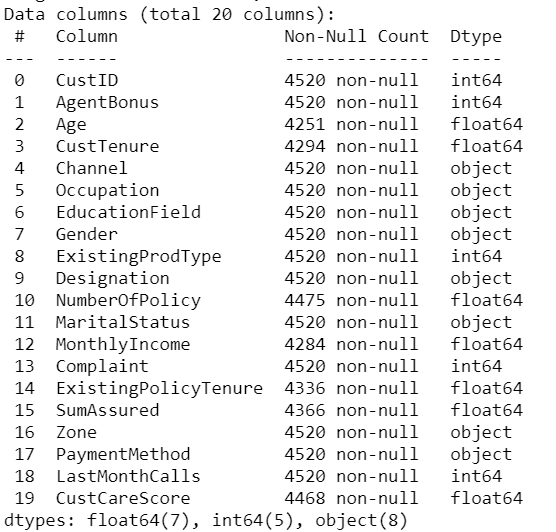
Below are the Categorical columns

'Channel', 'Occupation', 'EducationField', 'Gender', 'Designation', 'MaritalStatus', 'Zone', 'PaymentMethod'

Descriptive details of Categorical columns –



c) Understanding of attributes (variable info, renaming if required)



* There are 20 columns
* 8 columns having object datatypes
* 12 columns having int and float datatypes
* Below of the columns have null values

Age 269

MonthlyIncome 236

CustTenure 226

ExistingPolicyTenure 184

SumAssured 154

CustCareScore 52

NumberOfPolicy 45

* Below are the unique values and their count in each categorical columns-

CHANNEL: 3

Online 468

Third Party Partner 858

Agent 3194

OCCUPATION: 4

Free Lancer 2

Large Business 408

Small Business 1918

Salaried 2192

EDUCATIONFIELD: 6

MBA 74

Post Graduate 252

Engineer 408

Diploma 496

Under Graduate 1420

Graduate 1870

GENDER: 2

Female 1832

Male 2688

DESIGNATION: 5

VP 226

AVP 336

Senior Manager 676

Manager 1620

Executive 1662

MARITALSTATUS: 3

Divorced 804

Single 1448

Married 2268

ZONE: 4

South 6

East 64

North 1884

West 2566

PAYMENTMETHOD: 4

Quarterly 76

Monthly 354

Yearly 1434

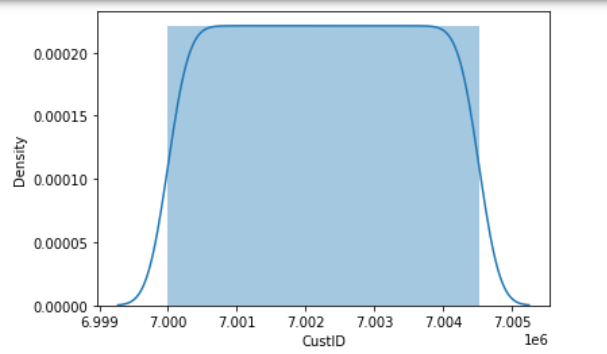
Half Yearly 2656

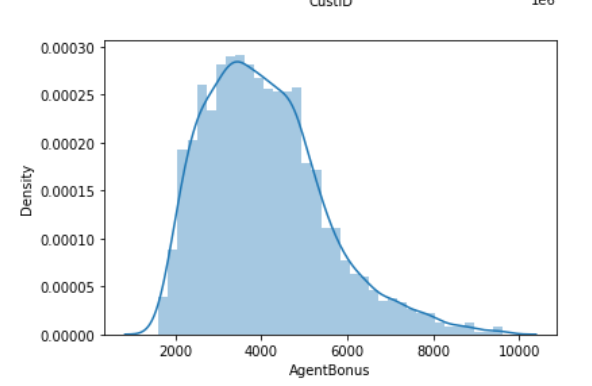
* There are no duplicate rows.

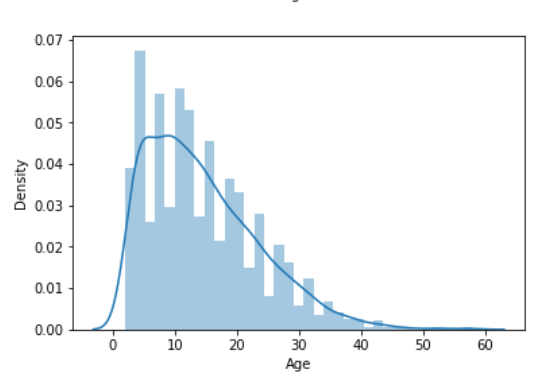
**3. Exploratory Data Analysis**

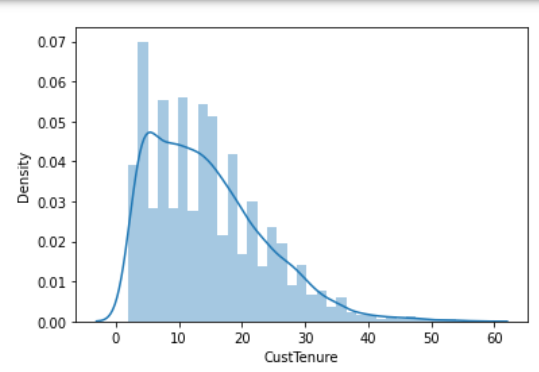
a) Univariate analysis (distribution and spread for every continuous attribute, distribution of data in categories for categorical ones)

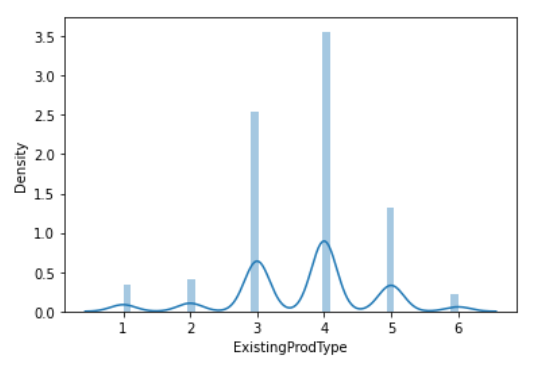
Numerical Variables –

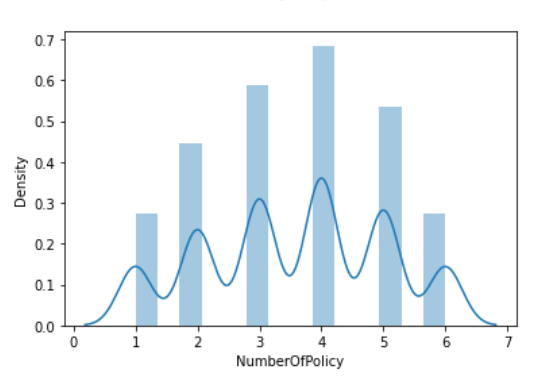


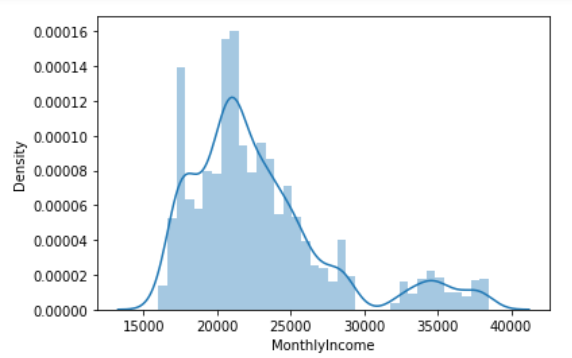


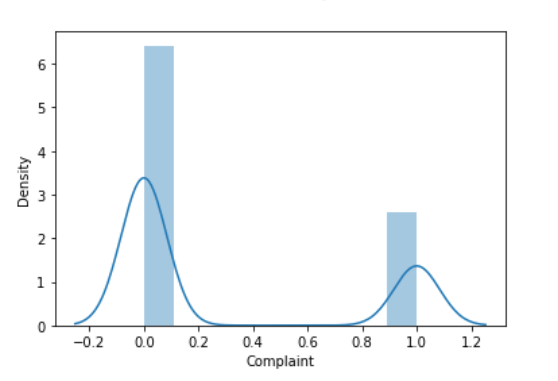


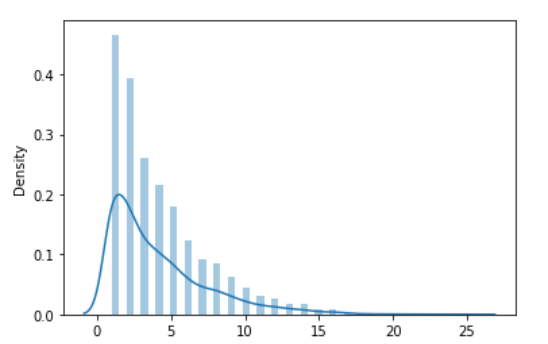


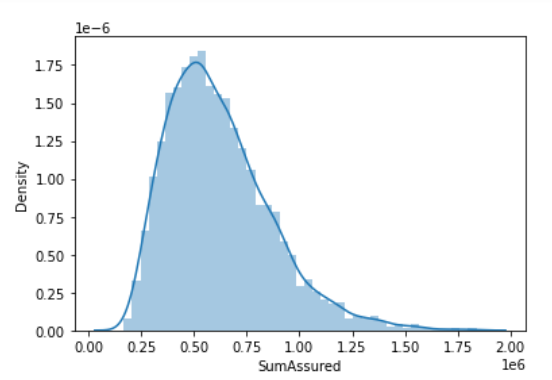


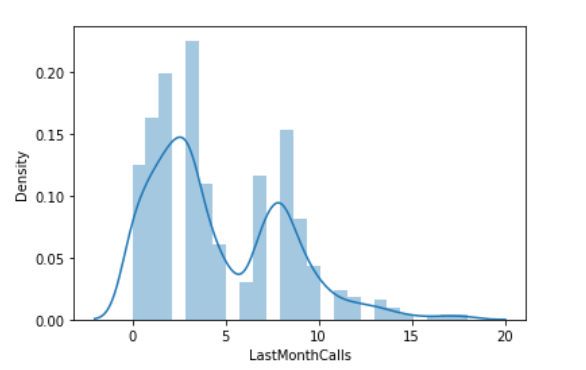


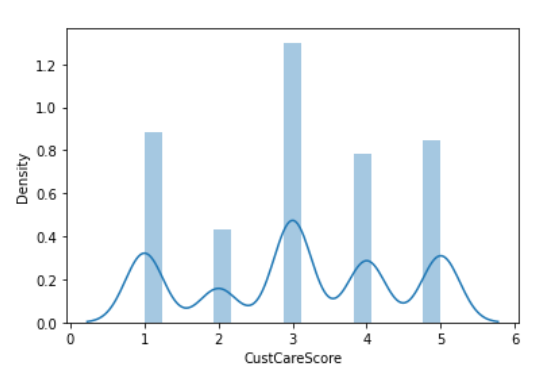




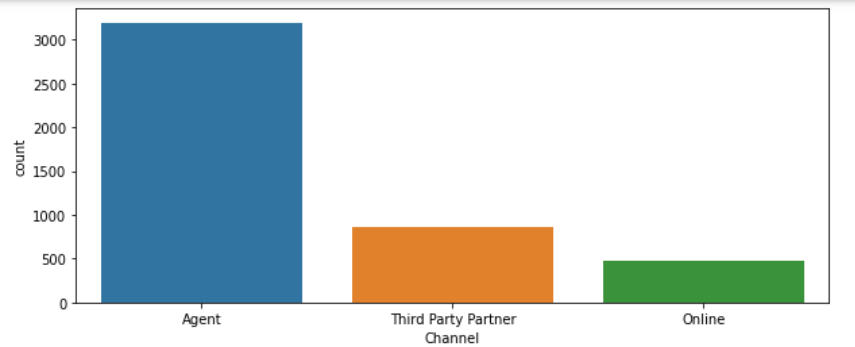


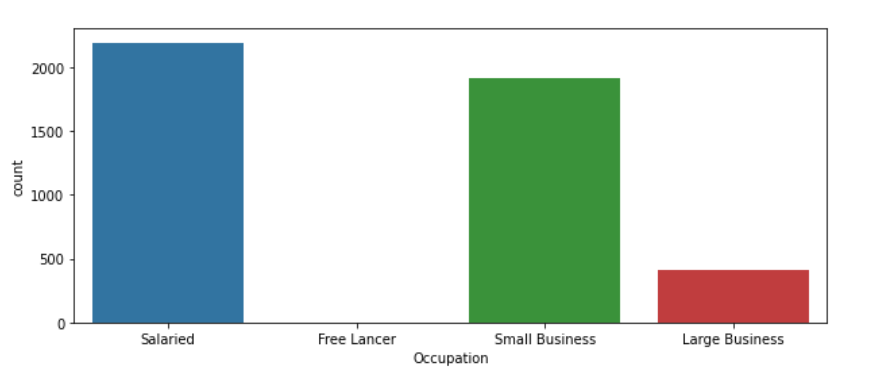


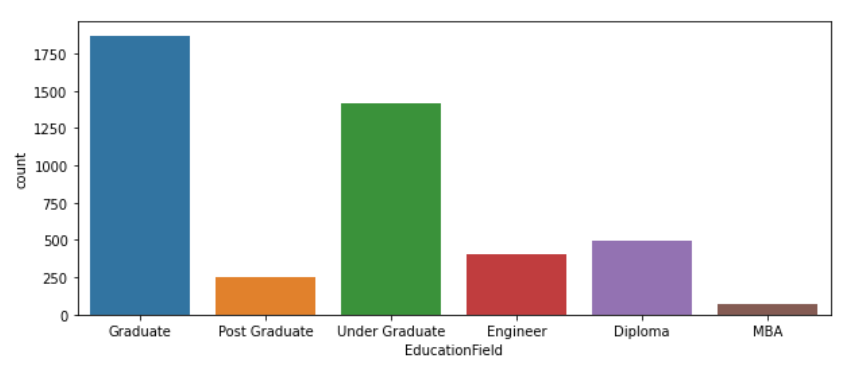


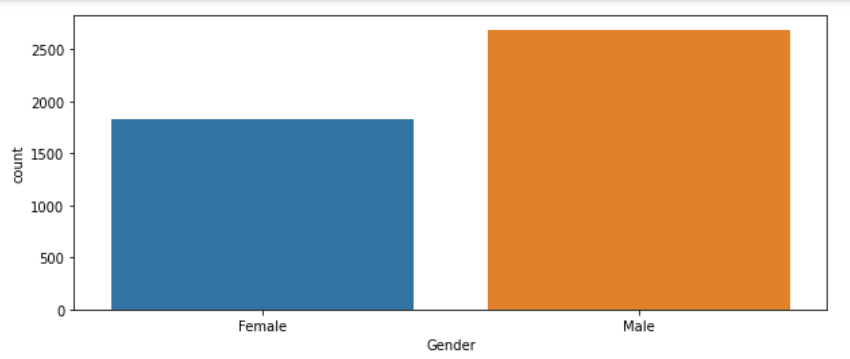


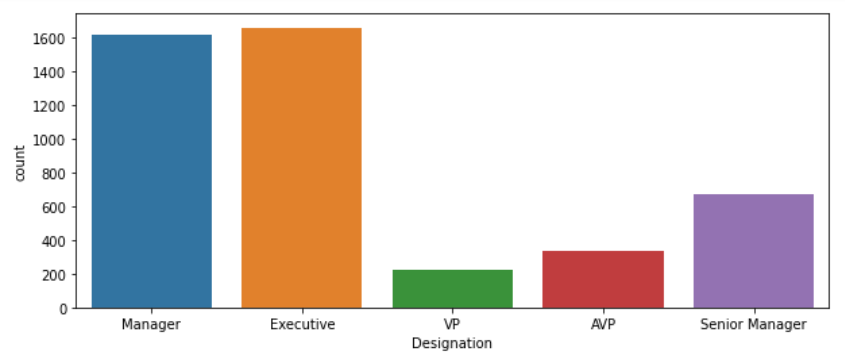
Categorical Variables-

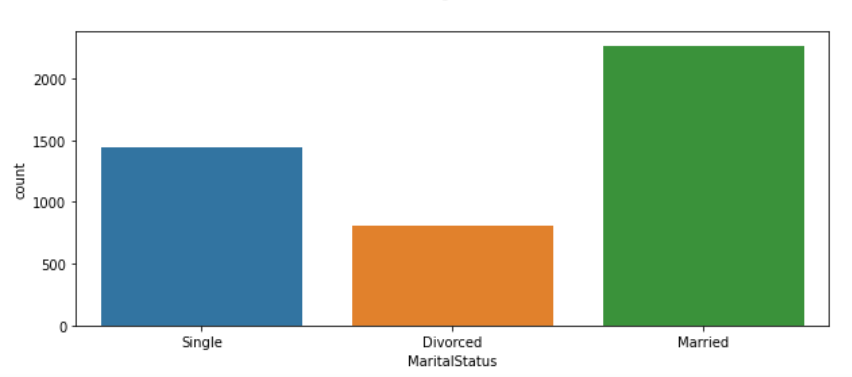


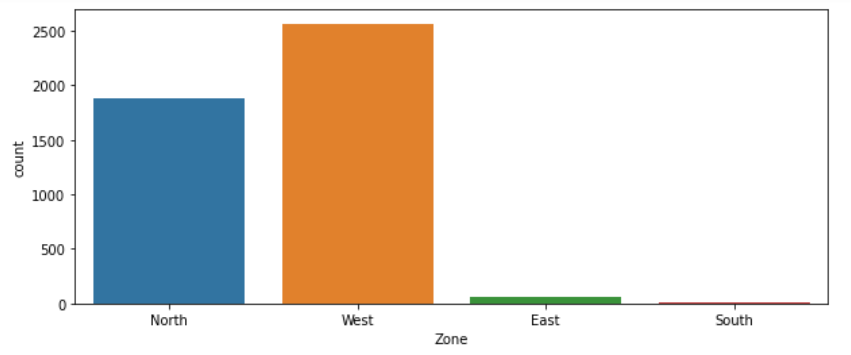


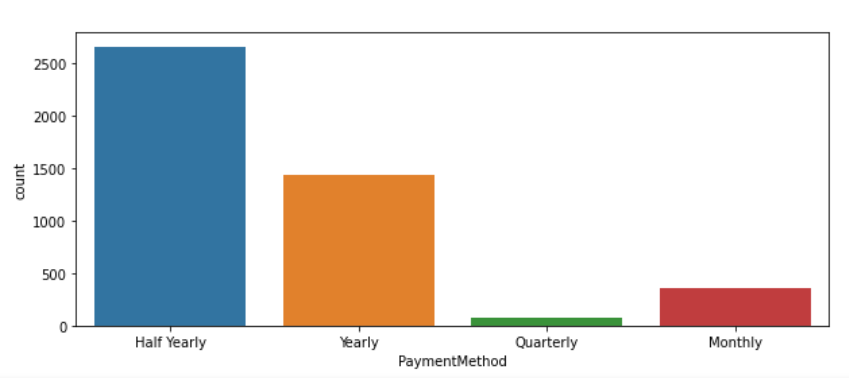




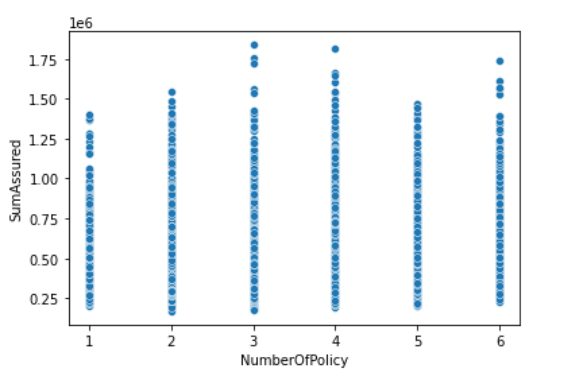


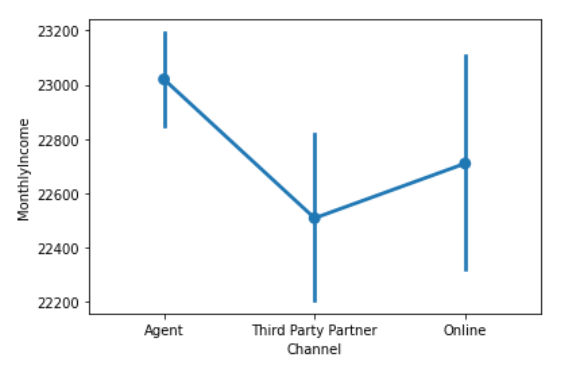


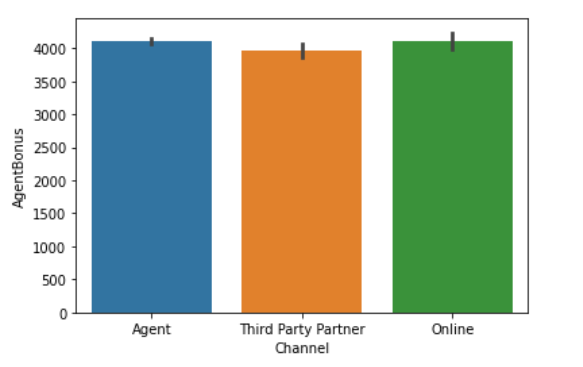


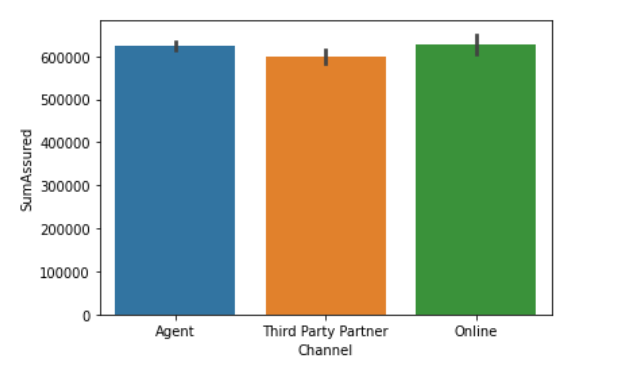


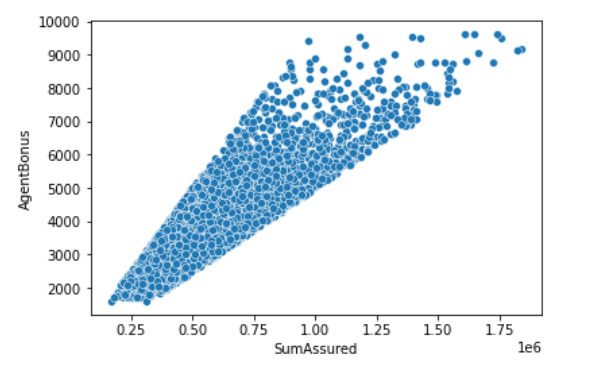
b) Bivariate analysis (relationship between different variables, correlations)

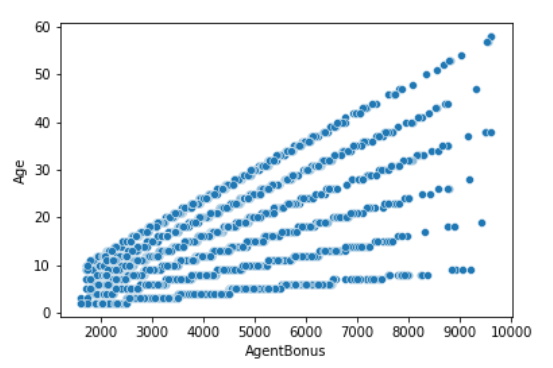


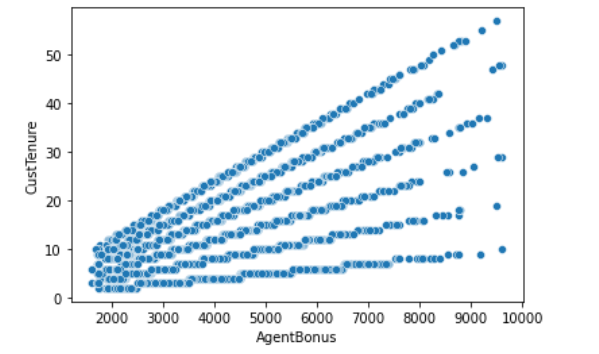


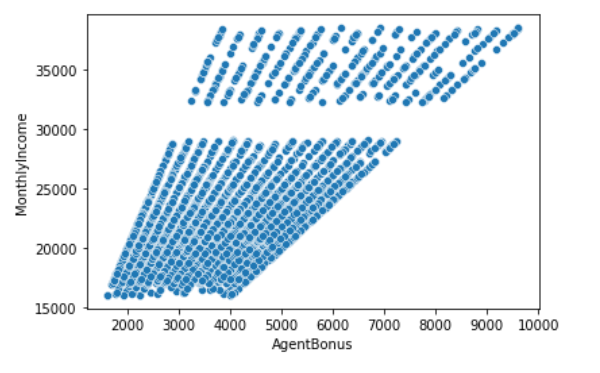




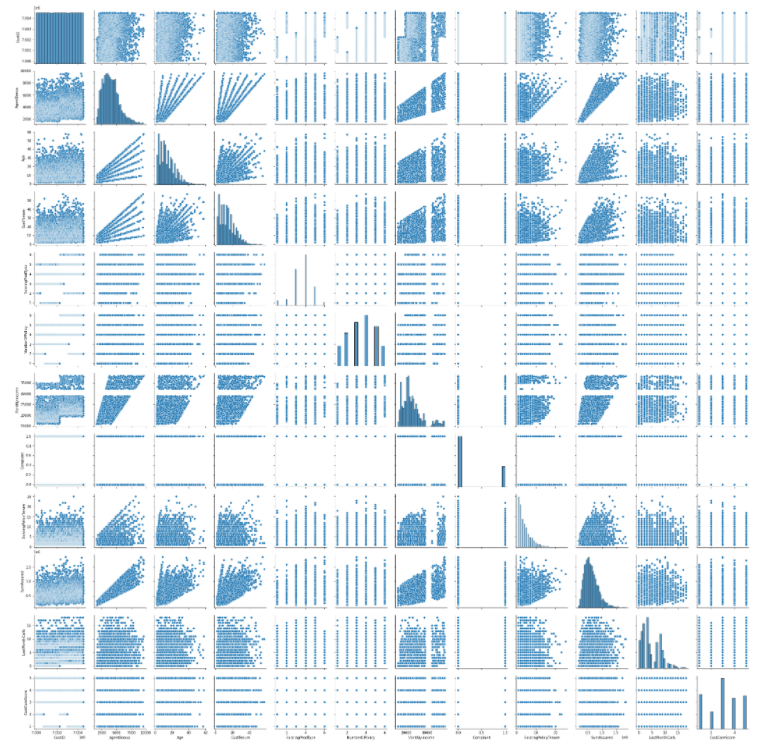




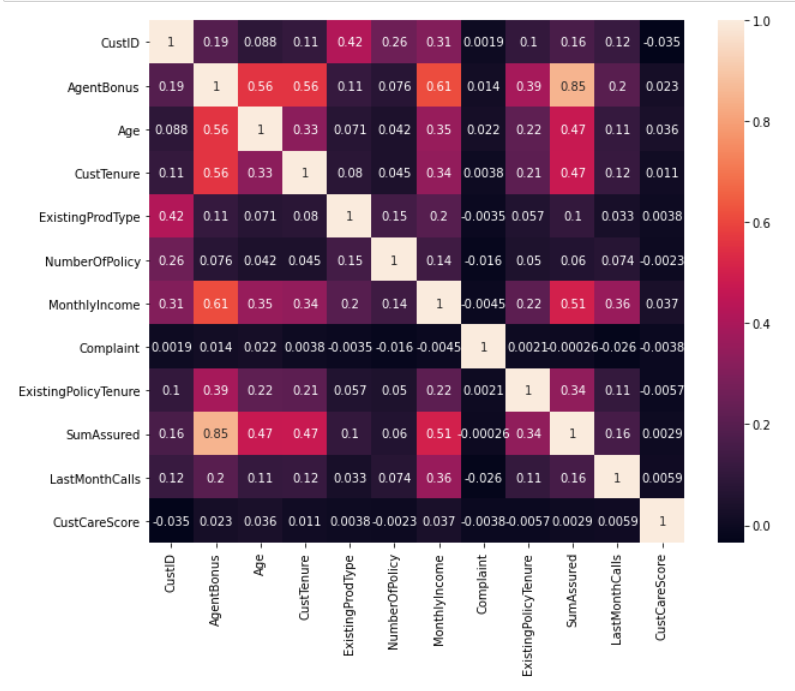




Pairplot –



HeatMap –



**3. Data Cleaning and Pre-processing**

a) Removal of unwanted variables (if applicable)

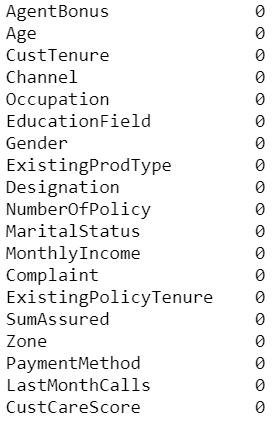
As agent performance is to be identified so Customer ID is irrelevant for this and can be dropped.

b) Missing Value treatment (if applicable)

Below columns have missing values –

1. Age 269
2. MonthlyIncome 236
3. CustTenure 226
4. ExistingPolicyTenure 184
5. SumAssured 154
6. CustCareScore 52
7. NumberOfPolicy 45

After missing values treatment (Imputing with median)

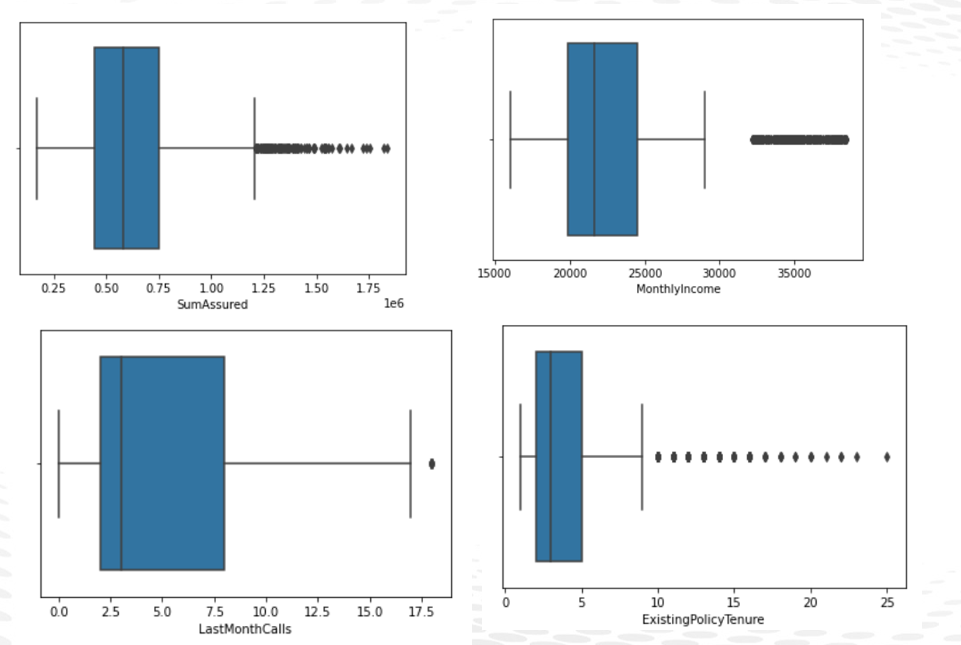


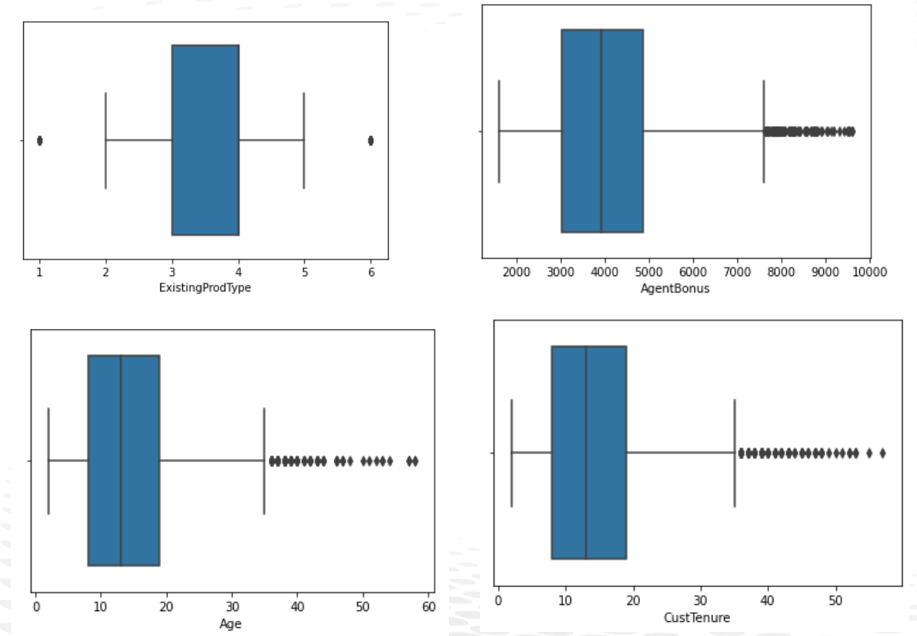
d) Outlier treatment

Below column had outliers -

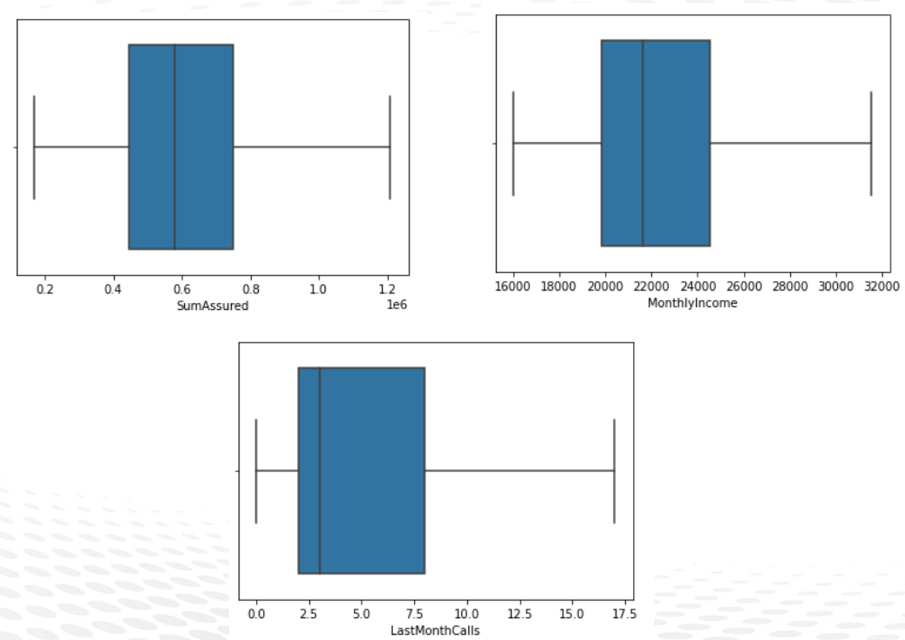
'AgentBonus','Age','CustTenure','ExistingProdType','MonthlyIncome','ExistingProdType',

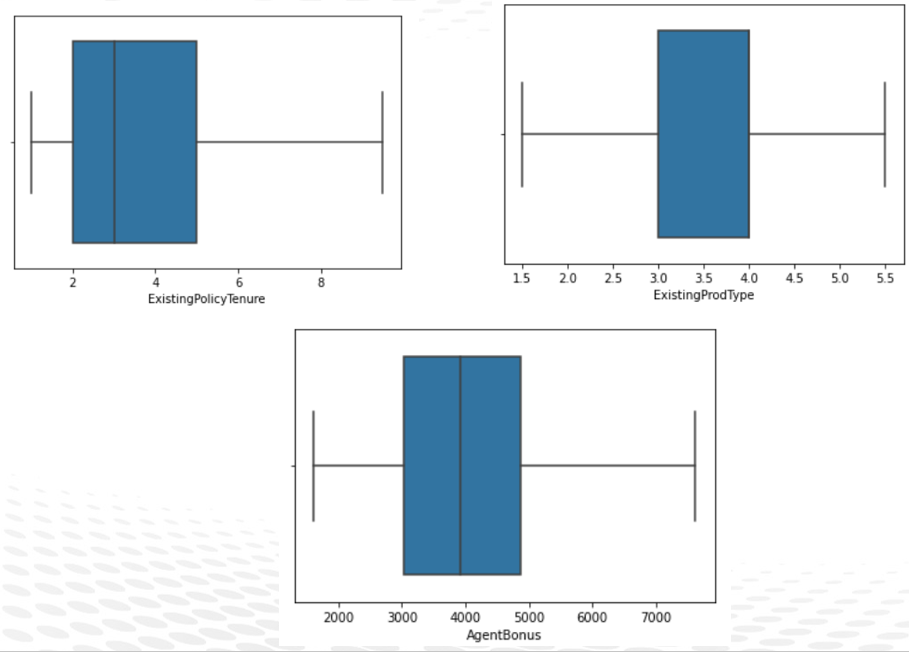
'SumAssured', 'LastMonthCalls'

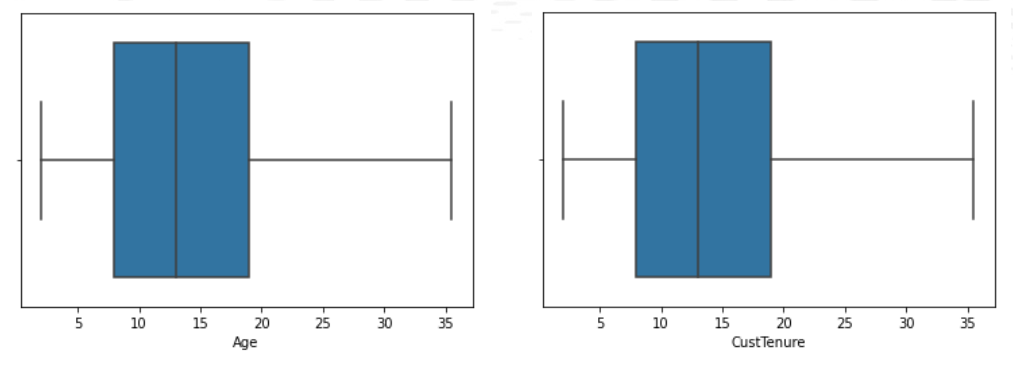




After Outlier Treatment

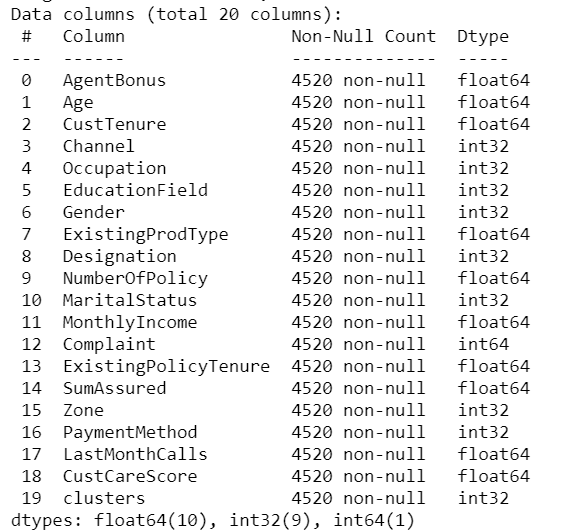






e) Variable transformation (if applicable)

Converted categorical variables to numerical variable using label encoding



f) Addition of new variables (if required)

1. Did not add new feature as of now.

**Business insights from EDA**

a) Is the data unbalanced? If so, what can be done? Please explain in the context of the business

1. Since this is and regression problem hence even if the data is imbalanced does not matter.
2. Data imbalance needs to be corrected in case of classification problem

**4. Model building**

**Models used are a combination of white box and black box techniques.**

**Linear Regression**

*from sklearn.linear\_model import LinearRegression*

*from sklearn import metrics*

*regression\_model = LinearRegression()*

*regression\_model.fit(x\_train, y\_train)*

The intercept for our model is -0.002975364383266541

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Linear Regression | 0.80 | 0.80 | 0.44 | 0.79 | 0.79 | 0.46 |

**Decision Tree Regressor**

*from sklearn.tree import DecisionTreeRegressor*

*# create a regressor object*

*DTRegModel = DecisionTreeRegressor(random\_state=0)*

*# fit the regressor with X and Y data*

*DTRegModel.fit(x\_train, y\_train)*

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Decision Tree Regressor | 0.84 | 0.84 | 0.00 | 0.80 | 0.80 | 0.53 |

**Random Forest Regressor**

*from sklearn.ensemble import RandomForestRegressor*

*RandomForestRegModel = RandomForestRegressor(n\_estimators = 10, random\_state = 0)*

*RandomForestRegModel.fit(x\_train, y\_train)*

*>>RandomForestRegressor(n\_estimators=10, random\_state=0)*

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Random Forest Regressor | 0.97 | 0.97 | 0.17 | 0.83 | 0.84 | 0.40 |

**Gradient Boosting**

from sklearn.ensemble import GradientBoostingRegressor

gbr = GradientBoostingRegressor(n\_estimators=3,learning\_rate=1)

gbr.fit(x\_train, y\_train)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Gradient boosting | 0.80 | 0.80 | 0.45 | 0.79 | 0.78 | 0.47 |

**Bagging**

from sklearn.ensemble import BaggingRegressor

bgcl = BaggingRegressor(base\_estimator=RandomForestRegressor(),n\_estimators=10, random\_state=0).fit(x, y)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Bagging | 0.95 | 0.95 | 0.23 | 0.84 | 0.85 | 0.39 |

**Support Vector Machine**

*from sklearn.svm import SVR*

*svr = SVR()*

*svr.fit(x\_train, y\_train)*

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Support Vector Machine | 0.89 | 0.89 | 0.34 | 0.78 | 0.78 | 0.46 |

**Bayesian Ridge**

*from sklearn.linear\_model import BayesianRidge*

*BRmodel = BayesianRidge()*

*BRmodel.fit(x\_train, y\_train)*

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Bayesian Ridge | 0.80 | 0.80 | 0.44 | 0.79 | 0.79 | 0.45 |

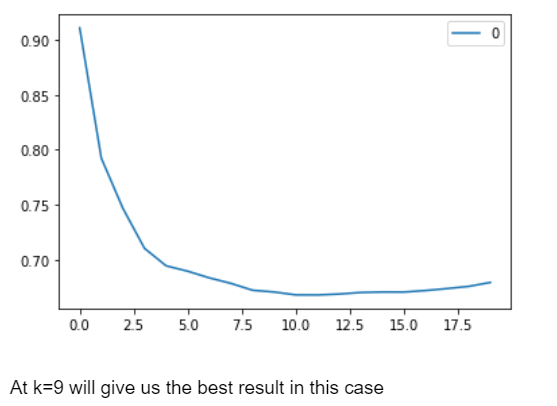
**KNN Regressor**

*from sklearn.neighbors import KNeighborsRegressor*

*model = KNeighborsRegressor(n\_neighbors = K)*

*model.fit(x\_train, y\_train)*

WSS plot:



Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| KNN Regressor | 0.64 | 0.64 | 0.60 | 0.53 | 0.55 | 0.67 |

**Artificial Neural Network**

*from sklearn.neural\_network import MLPRegressor*

*MLPReg = MLPRegressor(random\_state=1, max\_iter=500).fit(x\_train, y\_train)*

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Artificial Neural Network | 0.95 | 0.93 | 0.26 | 0.71 | 0.75 | 0.50 |

**Interpretation of the model(s):**

1) Random Forest Regressor has the highest R2 and Adj.R2 value and lowest RMSE value for both training and testing data

**2) Model Tuning**

**Random Forest Regressor**

Best params:

{'bootstrap': True,

'max\_features': 'auto',

'min\_samples\_split': 8,

'n\_estimators': 20}

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Tuned Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Random Forest Regressor | 0.98 | 0.98 | 0.14 | 0.85 | 0.85 | 0.39 |

**Support Vector Machine**

Best Params:

{'C': 100, 'gamma': 0.0001, 'kernel': 'rbf'}

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Tuned Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Support Vector Machine | 0.80 | 0.80 | 0.44 | 0.78 | 0.79 | 0.46 |

**Artificial Neural Network**

Best Params:

{'activation': 'logistic',

'alpha': 0.05,

'hidden\_layer\_sizes': (50, 50, 50),

'learning\_rate': 'adaptive',

'solver': 'adam'}

Metrics:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Tuned Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Artificial Neural Network | -0.20 | 0.82 | 0.20 | -0.24 | 0.80 | -0.24 |

**Interpretation of the tuned model(s):**

Random forest regressor has the highest R2 and Adj.R2 value

**5. Model validation**

Below metrics are used for model evaluation

1) Adjusted R-square

2) R-square

3) RMSE

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Train Data | | | Test Data | | |
| Models | Adj. R-Square | R-square | RMSE | Adj. R-Square | R-square | RMSE |
| Linear Regression | 0.80 | 0.80 | 0.44 | 0.79 | 0.79 | 0.46 |
| Decision Tree Regressor | 0.84 | 0.84 | 0.00 | 0.80 | 0.80 | 0.53 |
| Random Forest Regressor | 0.97 | 0.97 | 0.17 | 0.83 | 0.84 | 0.40 |
| Support Vector Machine | 0.89 | 0.89 | 0.34 | 0.78 | 0.78 | 0.46 |
| Bayesian Ridge | 0.80 | 0.80 | 0.44 | 0.79 | 0.79 | 0.45 |
| KNN Regressor | 0.64 | 0.64 | 0.60 | 0.53 | 0.55 | 0.67 |
| Artificial Neural Network | 0.95 | 0.93 | 0.26 | 0.71 | 0.75 | 0.50 |
| Gradient boosting | 0.80 | 0.80 | 0.45 | 0.79 | 0.78 | 0.47 |
| Bagging | 0.95 | 0.95 | 0.23 | 0.84 | 0.85 | 0.39 |

**6. Final interpretation / recommendation**

a) Any business insights using clustering (if applicable)

Cluster 1:

1. Average age of customer is 20
2. Average agent bonus is around Rs.5726
3. Average Tenure of customer in organization 21 years
4. Most common channel through which acquisition of customer is done is Agent
5. Common Payment method frequency is half yearly
6. Average customer care score is 3
7. Average last month calls are 5

Cluster 2:

1. Average age of customer is 11
2. Average agent bonus is around Rs. 2862
3. Average Tenure of customer in organization 10 years
4. Most common channel through which acquisition of customer is done is Agent
5. Common Payment method frequency is half yearly
6. Average customer care score is 3
7. Average last month calls are 3

Cluster 3:

1. Average age of customer is 13
2. Average agent bonus is around Rs. 4167
3. Average Tenure of customer in organization 14 years
4. Most common channel through which acquisition of customer is done is Agent
5. Common Payment method frequency is half yearly
6. Average customer care score is 3
7. Average last month calls are 3

b) Any other business insights

Other use cases of this project could be-

1. **Policy Lapsation**
2. **Customer attrition**
3. **Product affinity**
4. **Agent’s attrition**
5. **Customer Life Time Value**
6. **Agent Life Time Value**
7. **Customer Segmentation**