Life Insurance Sales

(Agent Bonus Prediction)

Capstone Project

Project Notes – 1 & 2

Yedupati Venkata Yamini

5th November 2023

Table of Contents

List of figures

List of tables

1) Understanding the problem statement

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.
- This is a supervised regression problem in the Insurance domain particularly on Agent bonus amount prediction.

b) Need of the study/project

- Predicting the bonus given to agents helps them assess the number of candidates that need engagement activities and the number of candidates that need upskill programs based on various characteristics like zones, customer characteristics etc.
- This is important as most of the insurance business runs through word of mouth with the help of agents.
- They can either be trained well about insurance product types, benefits that customers would get if they purchase insurance and how it comes handy for them in emergency situations.

c) Understanding business/social opportunity

- The project/study aims to evaluate the performance of the insurance agents in terms of the insurances they register with the company
- It also aims to predict the bonus for agents so that they can know the high and low performing agents well ahead based on the given characteristics.
- The higher the insurance policy they sell to the customer, the higher is the revenue generated for the insurance company.
- In order to reward the agents in return, they are offered bonuses.
- This will motivate them to efficiently communicate and sell insurance policies to the customers by explaining the benefits effectively.

2) **Data Report**

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

- The data was collected per each customer including the characteristics of the customer related to insurance such as tenure of the customer with the company, basic educational and occupational details of the customer, existing policy, sum assured and tenure details of the customer.
- Along with these the important and target variable 'Agent Bonus' is also recorded which is the bonus received by the agent for acquiring the customer with an insurance policy.

b) Visual inspection of data

First 5 rows of the dataset:

	CustID	AgentBonus	Age	CustTenure	Channel	Occupation	EducationField	Gender	ExistingProdType	Designation	NumberOfPolicy	MaritalStatus
0	7000000	4409	22.0	4.0	Agent	Salaried	Graduate	Female	3	Manager	2.0	Single
1	7000001	2214	11.0	2.0	Third Party Partner	Salaried	Graduate	Male	4	Manager	4.0	Divorced
2	7000002	4273	26.0	4.0	Agent	Free Lancer	Post Graduate	Male	4	Exe	3.0	Unmarried
3	7000003	1791	11.0	NaN	Third Party Partner	Salaried	Graduate	Female	3	Executive	3.0	Divorced
4	7000004	2955	6.0	NaN	Agent	Small Business	UG	Male	3	Executive	4.0	Divorced

MonthlyIncome	Complaint	ExistingPolicyTenure	SumAssured	Zone	PaymentMethod	LastMonthCalls	CustCareScore
20993.0	1	2.0	806761.0	North	Half Yearly	5	2.0
20130.0	0	3.0	294502.0	North	Yearly	7	3.0
17090.0	1	2.0	NaN	North	Yearly	0	3.0
17909.0	1	2.0	268635.0	West	Half Yearly	0	5.0
18468.0	0	4.0	366405.0	West	Half Yearly	2	5.0

Table 1: Viewing the first rows of the dataset

Last 5 rows of the dataset:

	CustID	AgentBonus	Age	CustTenure	Channel	Occupation	EducationField	Gender	ExistingProdType	Designation	NumberOfPolicy	MaritalStatus
4515	7004515	3953	4.0	8.0	Agent	Small Business	Graduate	Male	4	Senior Manager	2.0	Single
4516	7004516	2939	9.0	9.0	Agent	Salaried	Under Graduate	Female	2	Executive	2.0	Married
4517	7004517	3792	23.0	23.0	Agent	Salaried	Engineer	Female	5	AVP	5.0	Single
4518	7004518	4816	10.0	10.0	Online	Small Business	Graduate	Female	4	Executive	2.0	Single
4519	7004519	4764	14.0	10.0	Agent	Salaried	Under Graduate	Female	5	Manager	2.0	Married

MonthlyIncome	Complaint	ExistingPolicyTenure	SumAssured	Zone	PaymentMethod	LastMonthCalls	CustCareScore
26355.0	0	2.0	636473.0	West	Yearly	9	1.0
20991.0	0	3.0	296813.0	North	Yearly	1	3.0
NaN	0	2.0	667371.0	North	Half Yearly	4	1.0
20068.0	0	6.0	943999.0	West	Half Yearly	1	5.0
23820.0	0	3.0	700308.0	North	Half Yearly	1	3.0

Table 2: Viewing the last rows of the dataset

- Shown above is the glimpse of the received data with the characteristics of a customer and the bonus received by the agent.
- The dataset belongs to 4520 customers and respective bonuses received by the agents.
- There are 20 columns/attributes in the dataset with target column being ‘AgentBonus’
- Basic information of the columns and values in the dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 20 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   CustID                4520 non-null   int64
 1   AgentBonus            4520 non-null   int64
 2   Age                   4251 non-null   float64
 3   CustTenure            4294 non-null   float64
 4   Channel               4520 non-null   object
 5   Occupation            4520 non-null   object
 6   EducationField        4520 non-null   object
 7   Gender                4520 non-null   object
 8   ExistingProdType      4520 non-null   int64
 9   Designation           4520 non-null   object
 10  NumberOfPolicy        4475 non-null   float64
 11  MaritalStatus         4520 non-null   object
 12  MonthlyIncome         4284 non-null   float64
 13  Complaint             4520 non-null   int64
 14  ExistingPolicyTenure  4336 non-null   float64
 15  SumAssured            4366 non-null   float64
 16  Zone                  4520 non-null   object
 17  PaymentMethod         4520 non-null   object
 18  LastMonthCalls        4520 non-null   int64
 19  CustCareScore         4468 non-null   float64
dtypes: float64(7), int64(5), object(8)
memory usage: 706.4+ KB
```

Table 3: Basic information of the initial dataset

- There are few columns with null values in the dataset and some columns like ‘Age’ are good to be represented as integers instead of decimals. Such cleaning will be done in further sections of the report.

- Summary of the values for the columns in the dataset

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
AgentBonus	4520.0	NaN	NaN	NaN	4077.838274	1403.321711	1605.0	3027.75	3911.5	4867.25	9608.0
Age	4251.0	NaN	NaN	NaN	14.494707	9.037629	2.0	7.0	13.0	20.0	58.0
CustTenure	4294.0	NaN	NaN	NaN	14.469027	8.963671	2.0	7.0	13.0	20.0	57.0
Channel	4520	3	Agent	3194	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Occupation	4520	5	Salaried	2192	NaN	NaN	NaN	NaN	NaN	NaN	NaN
EducationField	4520	7	Graduate	1870	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Gender	4520	3	Male	2688	NaN	NaN	NaN	NaN	NaN	NaN	NaN
ExistingProdType	4520.0	NaN	NaN	NaN	3.688938	1.015769	1.0	3.0	4.0	4.0	6.0
Designation	4520	6	Manager	1620	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NumberOfPolicy	4475.0	NaN	NaN	NaN	3.565363	1.455926	1.0	2.0	4.0	5.0	6.0
MaritalStatus	4520	4	Married	2268	NaN	NaN	NaN	NaN	NaN	NaN	NaN
MonthlyIncome	4284.0	NaN	NaN	NaN	22890.309991	4885.600757	16009.0	19683.5	21606.0	24725.0	38456.0
Complaint	4520.0	NaN	NaN	NaN	0.287168	0.452491	0.0	0.0	0.0	1.0	1.0
ExistingPolicyTenure	4336.0	NaN	NaN	NaN	4.130074	3.346386	1.0	2.0	3.0	6.0	25.0
SumAssured	4366.0	NaN	NaN	NaN	619999.699267	246234.82214	168536.0	439443.25	578976.5	758236.0	1838496.0
Zone	4520	4	West	2566	NaN	NaN	NaN	NaN	NaN	NaN	NaN
PaymentMethod	4520	4	Half Yearly	2656	NaN	NaN	NaN	NaN	NaN	NaN	NaN
LastMonthCalls	4520.0	NaN	NaN	NaN	4.626991	3.620132	0.0	2.0	3.0	8.0	18.0
CustCareScore	4468.0	NaN	NaN	NaN	3.067592	1.382968	1.0	2.0	3.0	4.0	5.0

-

Table 4: Statistical summary of the initial dataset

- The above table shows a basic statistical summary of the columns in the dataset.
- Insights:
  - Around 29% have registered complaints among existing customers.
  - Most of the customers are in the North and West zones. There are hardly any/no customers in East and South zones.
  - Most of the customers opt for yearly and half yearly payment methods.
  - Customer care score is given from 1 to 5. Surprisingly the number of customers who gave ratings of 1 and 5 are almost the same.
  - Most of the customers have existing insurance product types 3 and 4. Least 1 and 6.
  - Agent bonus ranges from 1600Rs to 9600Rs.
  - 75% of the agents received less than or equal to 4800 Rs.
  - Average customer tenure with the company is 14 years.

c) Understanding of attributes

Attributes/Features in the dataset:

Data	Variable	Discerption
Sales	CustID	Unique customer ID
Sales	AgentBonus	Bonus amount given to each agents in last month
Sales	Age	Age of customer
Sales	CustTenure	Tenure of customer in organization
Sales	Channel	Channel through which acquisition of customer is done
Sales	Occupation	Occupation of customer
Sales	EducationField	Field of education of customer
Sales	Gender	Gender of customer
Sales	ExistingProdType	Existing product type of customer
Sales	Designation	Designation of customer in their organization
Sales	NumberOfPolicy	Total number of existing policy of a customer
Sales	MaritalStatus	Marital status of customer
Sales	MonthlyIncome	Gross monthly income of customer
Sales	Complaint	Indicator of complaint registered in last one month by customer
Sales	ExistingPolicyTenure	Max tenure in all existing policies of customer
Sales	SumAssured	Max of sum assured in all existing policies of customer
Sales	Zone	Customer belongs to which zone in India. Like East, West, North and South
Sales	PaymentMethod	Frequency of payment selected by customer like Monthly, quarterly, half yearly and yearly
Sales	LastMonthCalls	Total calls attempted by company to a customer for cross sell
Sales	CustCareScore	Customer satisfaction score given by customer in previous service call

Table 5: Features in the dataset

Insights:

- CustId is the unique customer ID
- AgentBonus is the bonus earned by an agent per customer.
- Channels mentioned in the dataset are ‘Online’, ‘Agent’, ‘ThirdPartyPartner’.
- The Occupations of the customers include ‘Salaried’, ‘Business’, ‘Freelancer’ etc.
- Education fields of the customers include ‘Graduate’, ‘UnderGraduate’, ‘Diploma’, ‘MBA’ etc.
- There are 6 existing prod types mentioned in the dataset.
- CustCareScore is recorded ranging from 1 to 5.
- The existing column names can be used as given. No modification is required.

3) Exploratory Data Analysis

Note: The data received needs to be further cleaned inorder to perform univariate and bivariate analysis. Hence, performing the cleaning steps first.

a) Removal of unwanted variables

- Variable 'CustID' is the unique customer ID in numbers. It does not inform much about the customers and is not required for analysis and model building. Hence, it is removed.

b) Missing Value treatment

- Below are the columns having NULL values along with their counts out of 4520 values needed.

```
: Age                    269
  MonthlyIncome          236
  CustTenure             226
  ExistingPolicyTenure   184
  SumAssured             154
  CustCareScore           52
  NumberOfPolicy          45
```

Table 6: Features having missing values

- Used KNNImputer to impute NULL values in the variables 'Age', 'MonthlyIncome', 'CustTenure', 'ExistingPolicyTenure', 'SumAssured', 'NumberOfPolicy'
- K-NearestNeighbors imputer works by filling the missing values of each sample with the mean value from *n_neighbors* nearest neighbors found in the training set. Two samples are close if the features that neither is missing are close.
- The number of neighbors considered in the KNN Imputation performed for imputation of this dataset are 5.
- Missing values in the 'CustCareScore' column are filled with the median value of the existing values in the column i.e., 3.
- Viewing the information after imputing the null values in the dataset:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 19 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Channel               4520 non-null   object
 1   Occupation            4520 non-null   object
 2   EducationField        4520 non-null   object
 3   Gender                4520 non-null   object
 4   Designation           4520 non-null   object
 5   MaritalStatus         4520 non-null   object
 6   Complaint             4520 non-null   object
 7   Zone                  4520 non-null   object
 8   PaymentMethod         4520 non-null   object
 9   CustCareScore         4520 non-null   float64
 10  AgentBonus            4520 non-null   float64
 11  Age                   4520 non-null   float64
 12  CustTenure            4520 non-null   float64
 13  ExistingProdType      4520 non-null   float64
 14  NumberOfPolicy        4520 non-null   float64
 15  MonthlyIncome         4520 non-null   float64
 16  ExistingPolicyTenure  4520 non-null   float64
 17  SumAssured            4520 non-null   float64
 18  LastMonthCalls        4520 non-null   float64
dtypes: float64(10), object(9)
memory usage: 671.1+ KB
```

Table 7: Basic information of the dataset after missing value treatment

- The null/missing values have been imputed in the dataset.

c) Addition of new variables/Correction of existing variables

- No new variables are required to be added in the dataset.
- However, there is correction required to the existing data types of existing columns
- Data type of ‘Age’ is changed to ‘integer’
- Data type of ‘CustTenure’ is changed to ‘integer’
- Data type of ‘NumberOfPolicy" is changed to ‘integer’
- Data type of ‘CustCareScore’ is changed to ‘integer’

- Datatype of ‘ExistingProdType’ is changed to ‘category’
- Data type of ‘Complaint’ is changed to ‘category’
- Data type of ‘LastMonthCalls’ is changed to ‘integer’
- Viewing the basic info after changing the columns into required data types

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 19 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Channel               4520 non-null   object
 1   Occupation            4520 non-null   object
 2   EducationField        4520 non-null   object
 3   Gender                4520 non-null   object
 4   Designation           4520 non-null   object
 5   MaritalStatus         4520 non-null   object
 6   Complaint             4520 non-null   object
 7   Zone                  4520 non-null   object
 8   PaymentMethod         4520 non-null   object
 9   CustCareScore         4520 non-null   int64
 10  AgentBonus            4520 non-null   float64
 11  Age                   4520 non-null   int64
 12  CustTenure            4520 non-null   int64
 13  ExistingProdType      4520 non-null   object
 14  NumberOfPolicy        4520 non-null   int64
 15  MonthlyIncome         4520 non-null   float64
 16  ExistingPolicyTenure  4520 non-null   int64
 17  SumAssured            4520 non-null   float64
 18  LastMonthCalls        4520 non-null   int64
dtypes: float64(3), int64(6), object(10)
memory usage: 671.1+ KB
```

Table 8: Basic information of the dataset after correcting data types

d) Data cleaning

- Some columns have that is inconsistent and needs to be corrected
- ‘Laarge Business’ has been corrected to ‘Large Business’

```
Salaried           2192
Small Business     1918
Large Business      255
Laarge Business     153
Free Lancer           2
Name: Occupation, dtype: int64
```

```
Salaried           2192
Small Business     1918
Large Business      408
Free Lancer           2
Name: Occupation, dtype: int64
```

- Values named ‘UG’ have been renamed to ‘Under Graduate’ as both mean the same.

```
Graduate          1870
Under Graduate    1190
Diploma            496
Engineer           408
Post Graduate      252
UG                 230
MBA                 74
Name: EducationField, dtype: int64
```

```
Graduate          1870
Under Graduate    1420
Diploma            496
Engineer           408
Post Graduate      252
MBA                 74
Name: EducationField, dtype: int64
```

- ‘Fe male’ has been corrected to ‘Female’

```
: Male       2688
  Female     1507
  Fe male     325
  Name: Gender, dtype: int64
```

```
Male       2688
Female     1832
Name: Gender, dtype: int64
```

- ‘Exe’ has been corrected to ‘Executive’

```
Manager           1620
Executive         1535
Senior Manager     676
AVP                336
VP                 226
Exe                127
Name: Designation, dtype: int64
```

```
Executive         1662
Manager           1620
Senior Manager     676
AVP                336
VP                 226
Name: Designation, dtype: int64
```

- Viewing the first few rows after the data has been corrected

:

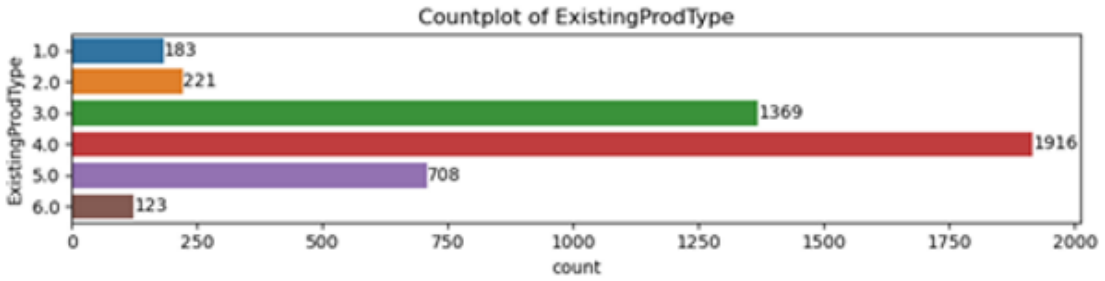
	Channel	Occupation	EducationField	Gender	Designation	MaritalStatus	Complaint	Zone	PaymentMethod	CustCareScore	AgentBonus	Age	CustTenure
0	Agent	Salaried	Graduate	Female	Manager	Single	1	North	Half Yearly	2	4409.0	22	4
1	Third Party Partner	Salaried	Graduate	Male	Manager	Divorced	0	North	Yearly	3	2214.0	11	2
2	Agent	Free Lancer	Post Graduate	Male	Exe	Unmarried	1	North	Yearly	3	4273.0	26	4
3	Third Party Partner	Salaried	Graduate	Female	Executive	Divorced	1	West	Half Yearly	5	1791.0	11	6
4	Agent	Small Business	Under Graduate	Male	Executive	Divorced	0	West	Half Yearly	5	2955.0	6	11

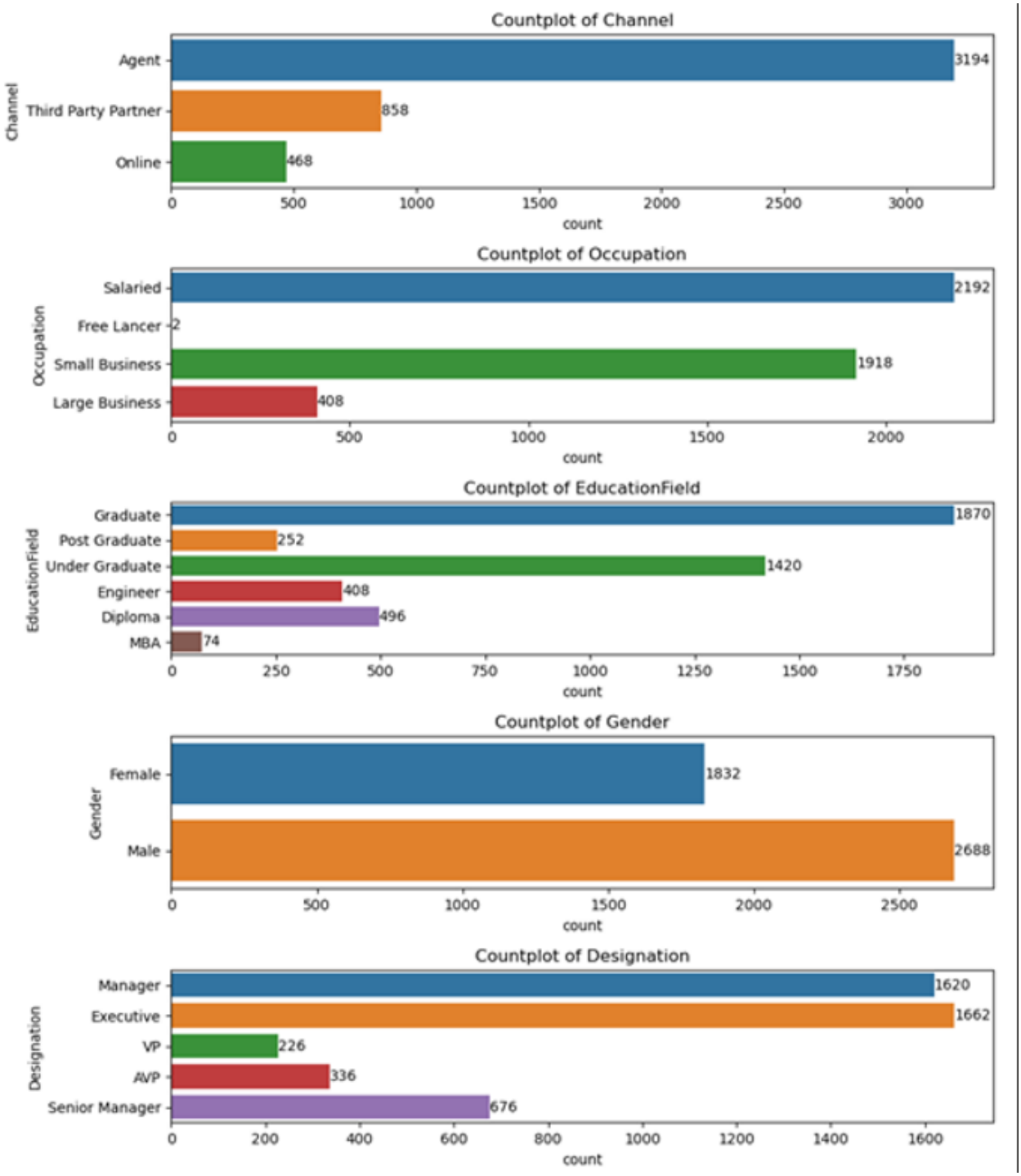
-

Table 9: Viewing the dataset after correcting data inconsistencies

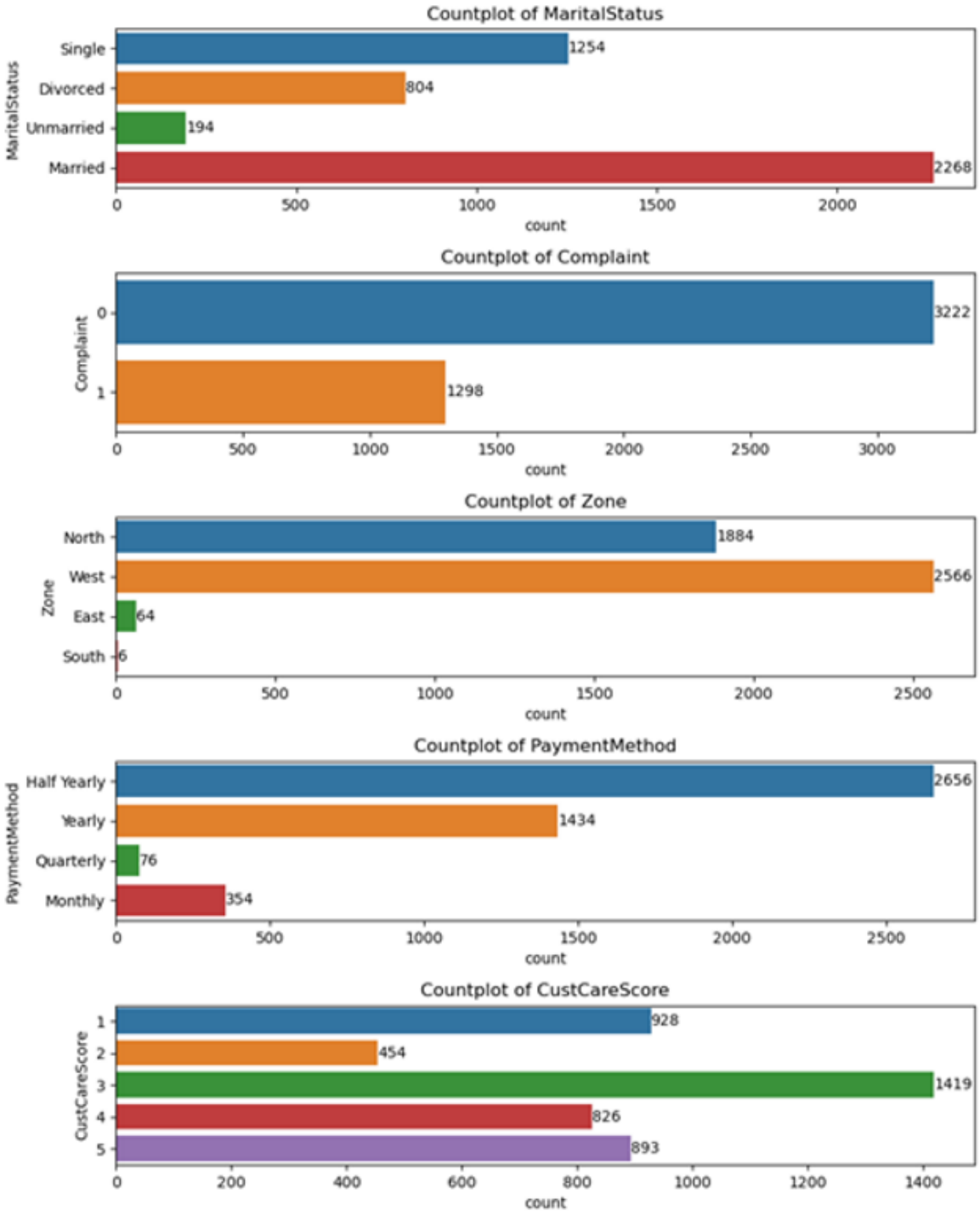
e) Univariate analysis

Univariate analysis of categorical variables:




Countplot of Channel
Channel
Agent
Third Party Partner
Online
3194
858
468
0
500
1000
1500
2000
2500
3000
count
Countplot of Occupation
Occupation
Salaried
Free Lancer
Small Business
Large Business
2192
2
1918
408
0
500
1000
1500
2000
count
Countplot of EducationField
EducationField
Graduate
Post Graduate
Under Graduate
Engineer
Diploma
MBA
1870
252
1420
408
496
74
0
250
500
750
1000
1250
1500
1750
count
Countplot of Gender
Gender
Female
Male
1832
2688
0
500
1000
1500
2000
2500
count
Countplot of Designation
Designation
Manager
Executive
VP
AVP
Senior Manager
1620
1662
226
336
676
0
200
400
600
800
1000
1200
1400
1600
count

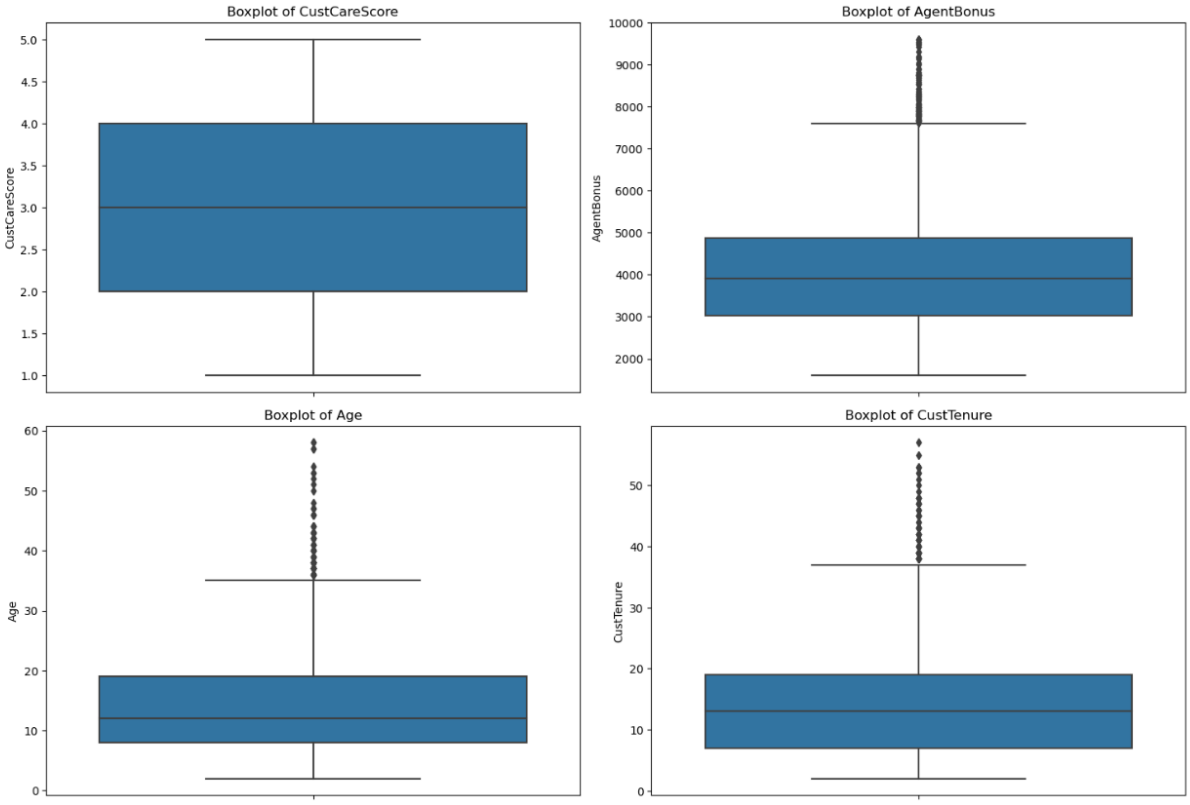
-



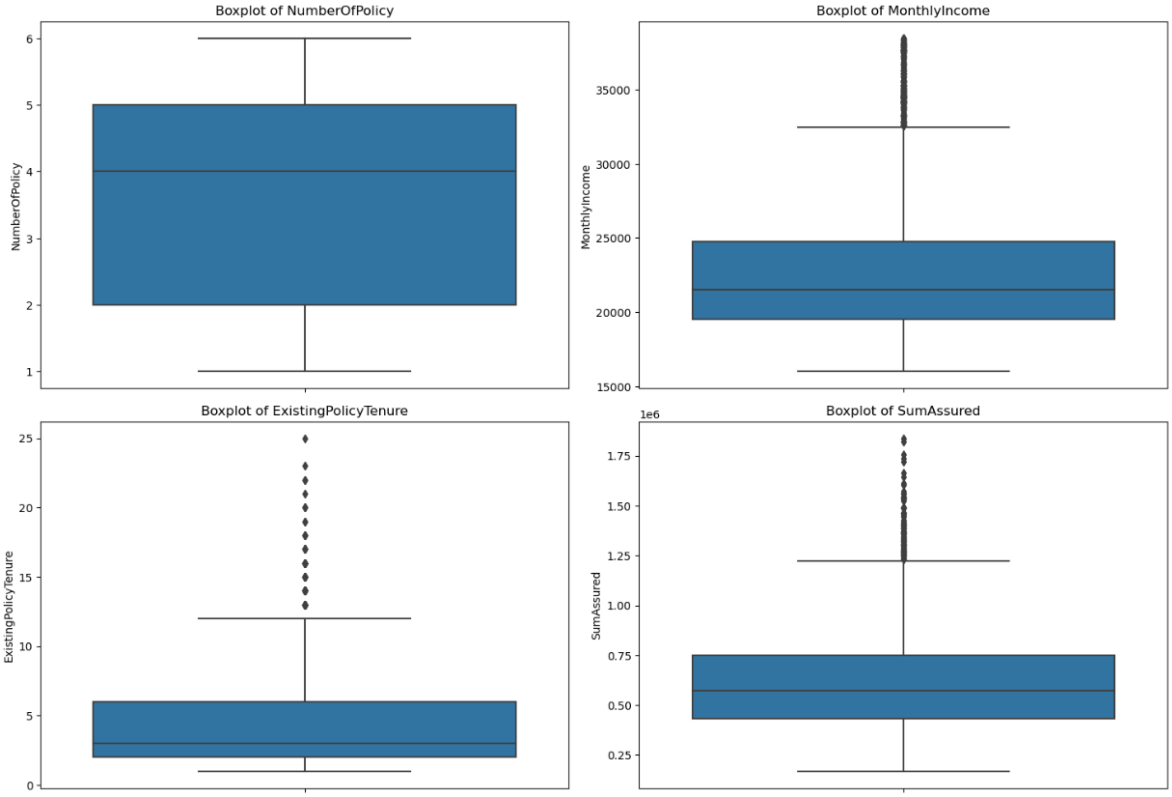
-

Figure 1: Univariate analysis of categorical features

- **Univariate analysis of numeric variables**



-



-

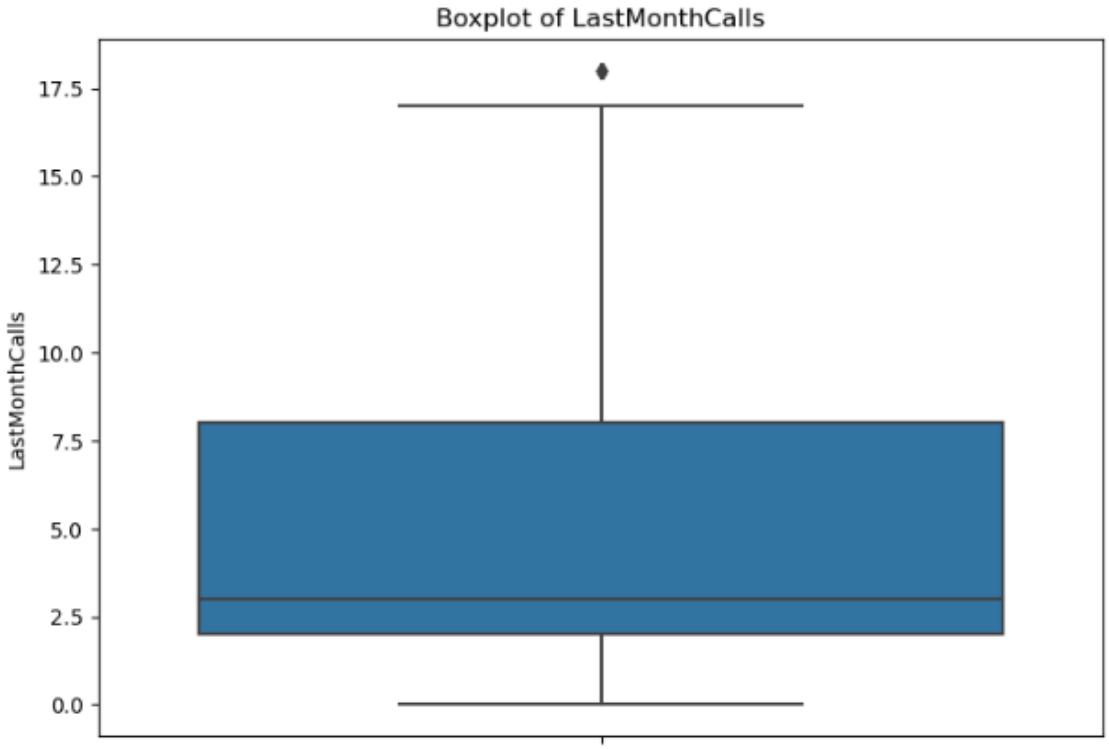


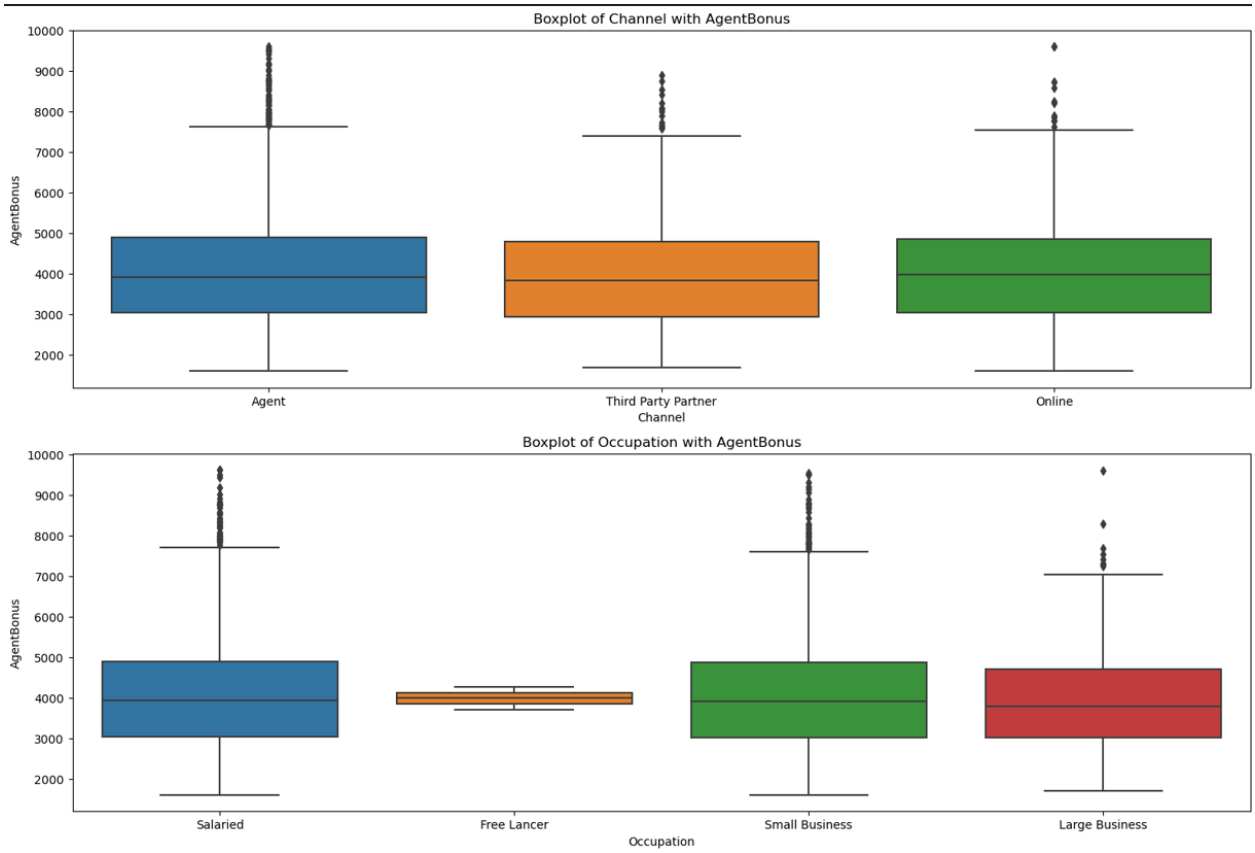
Figure 2: Univariate analysis of numeric features

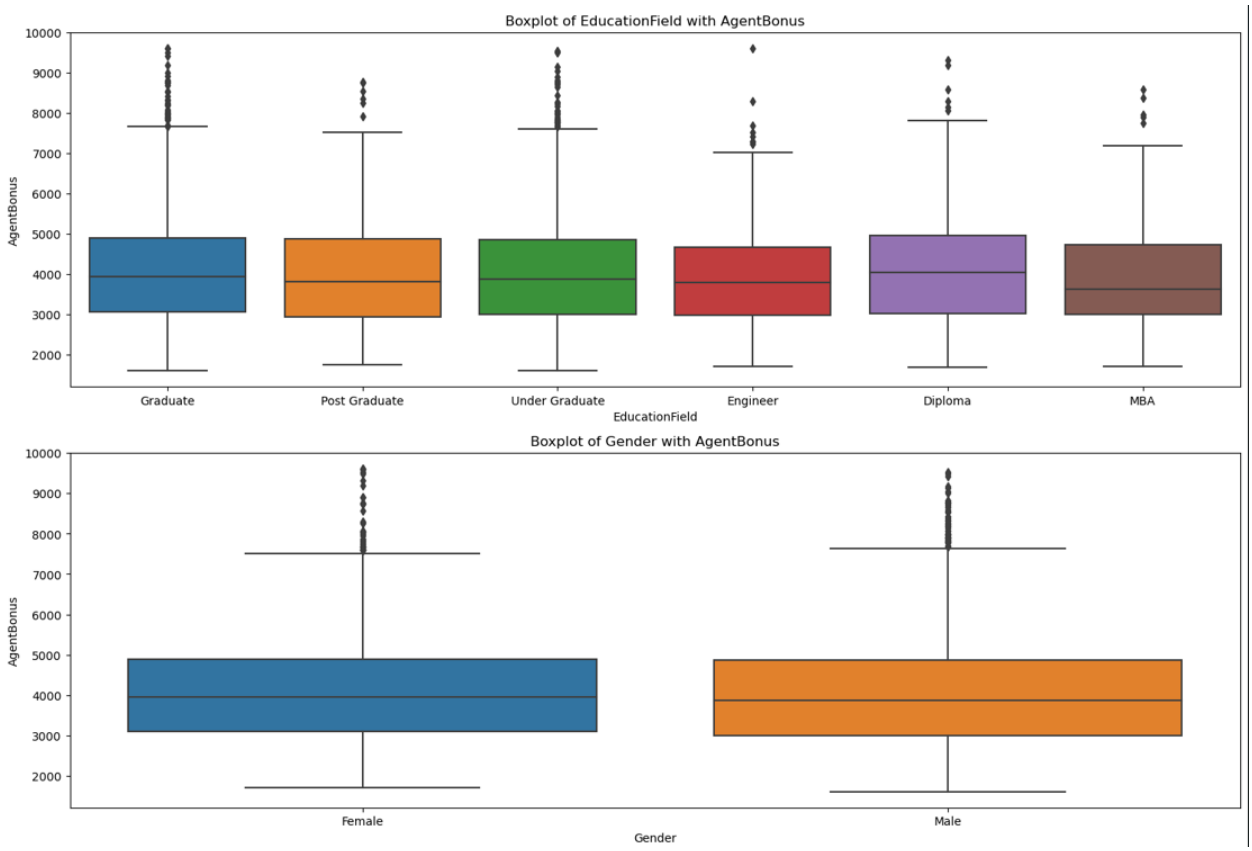
- **Insights from univariate analysis:**
  - Most of the customers are acquired through Agents.
  - Most of the customers are either salaried or running small businesses.
  - Managers and executives often opt for insurances more than VP and AVP.
  - Married People opt more for insurance.
  - Around 29% have registered complaints among existing customers.
  - Most of the customers are in the North and West zones. There are hardly any/no customers in East and South zones.
  - Most of the customers opt for yearly and half yearly payment methods.
  - Customer care score is given from 1 to 5. Surprisingly the number of customers who gave ratings of 1 and 5 are almost the same.
  - Most of the customers have existing insurance product types 3 and 4. Least 1 and 6.
  - Agent bonus ranges from 1600Rs to 9600Rs.
  - 75% of the agents received less than or equal to 4800 Rs.
  - Average customer tenure with the company is 14 years.

- On an average, customers have taken 3 policies, there are customers who have 6 policies as well.
- Average income of the customers is 22,890Rs. Income ranges from 16000 Rs to 38000 Rs.
- Average sum assured is around 6.1lakh Rs. Highest sum assured is around 18 lakh Rs.
- Half of the customers needed only 3 calls to cross sell but the other half of the customers required even 18 calls for attempting to cross sell.

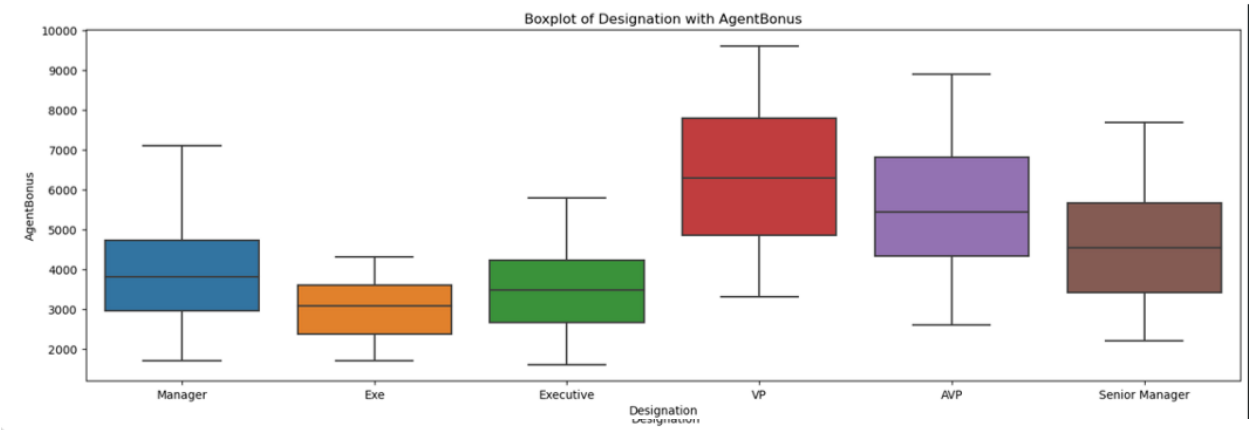
f) Bivariate analysis

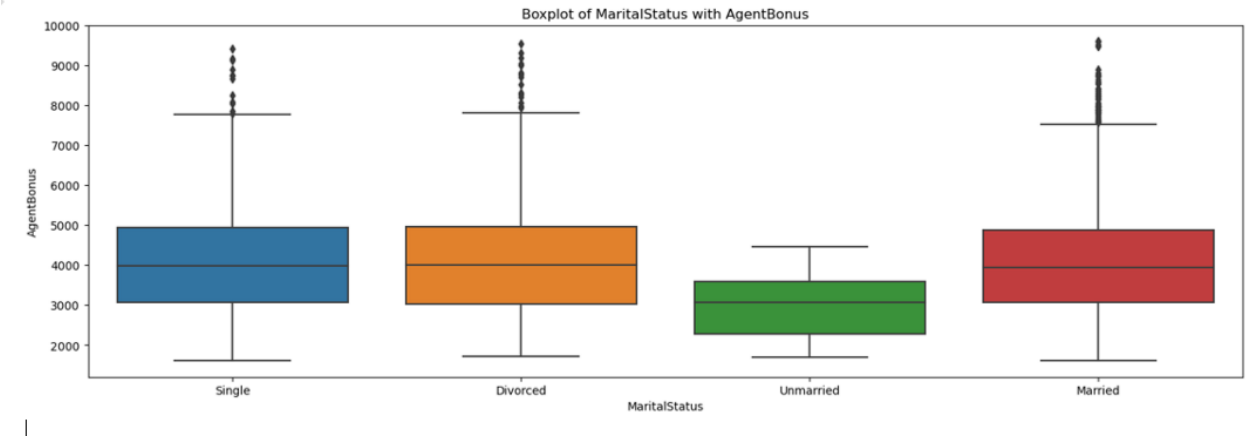
Analysis of categorical features with target variable:

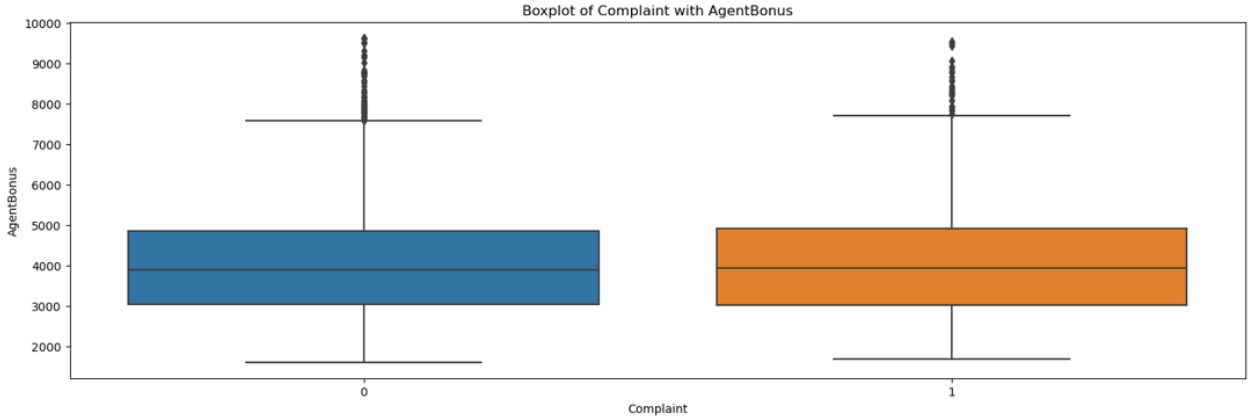


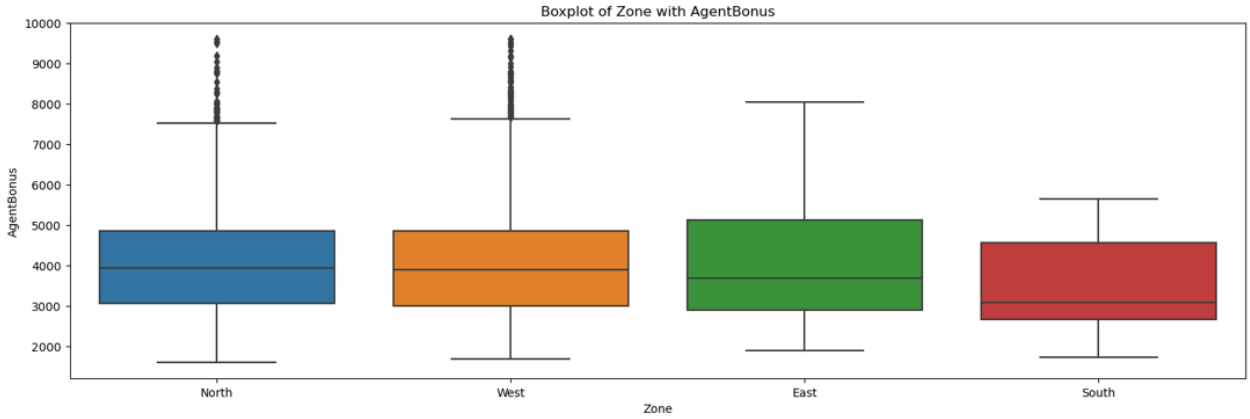

Boxplot of EducationField with AgentBonus
AgentBonus
EducationField
Graduate
Post Graduate
Under Graduate
Engineer
Diploma
MBA

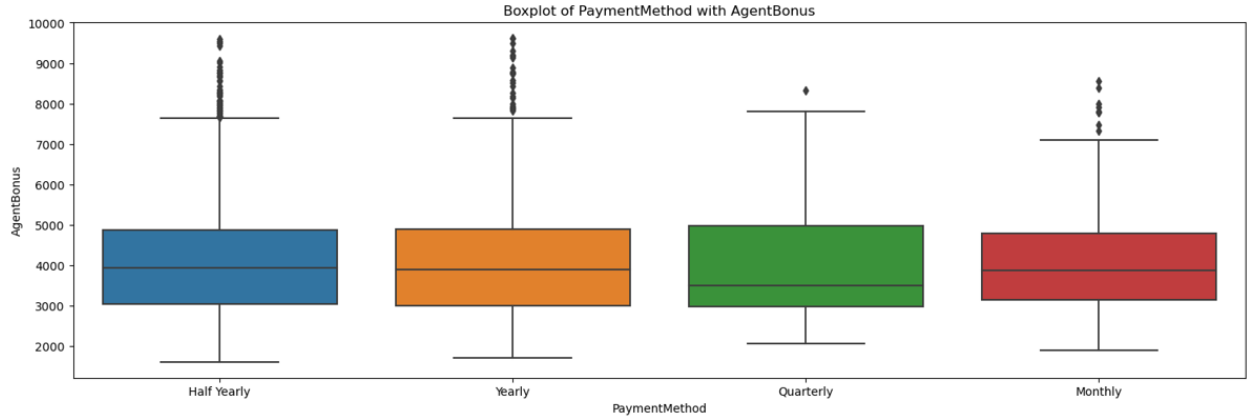
Boxplot of Gender with AgentBonus
AgentBonus
Gender
Female
Male

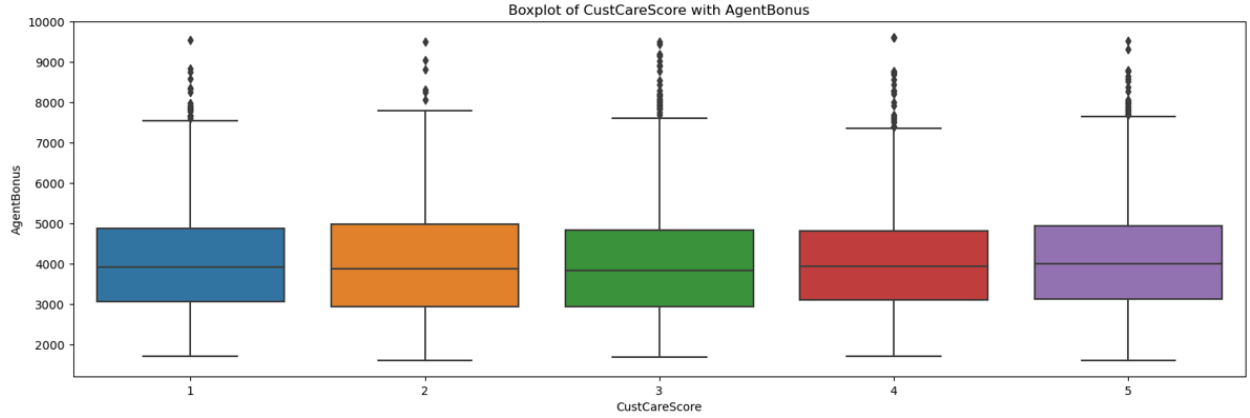

Boxplot of Designation with AgentBonus
AgentBonus
Designation
Manager
Exe
Executive
VP
AVP
Senior Manager


Boxplot of MaritalStatus with AgentBonus
AgentBonus
MaritalStatus
Single
Divorced
Unmarried
Married


Boxplot of Complaint with AgentBonus
AgentBonus
10000
9000
8000
7000
6000
5000
4000
3000
2000
0
1
Complaint


Boxplot of Zone with AgentBonus
AgentBonus
10000
9000
8000
7000
6000
5000
4000
3000
2000
North
West
East
South
Zone


Boxplot of PaymentMethod with AgentBonus
AgentBonus
10000
9000
8000
7000
6000
5000
4000
3000
2000
Half Yearly
Yearly
Quarterly
Monthly
PaymentMethod


Boxplot of CustCareScore with AgentBonus
AgentBonus
10000
9000
8000
7000
6000
5000
4000
3000
2000
1
2
3
4
5
CustCareScore

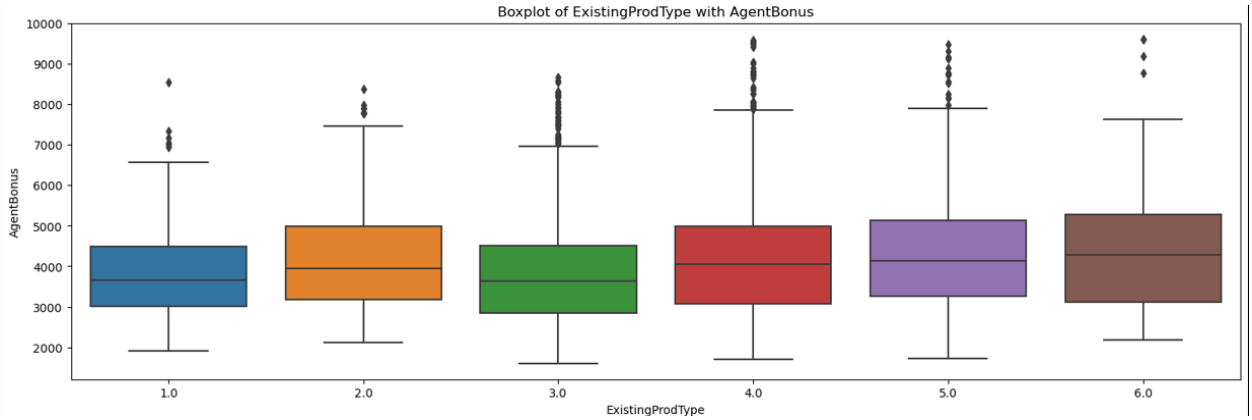
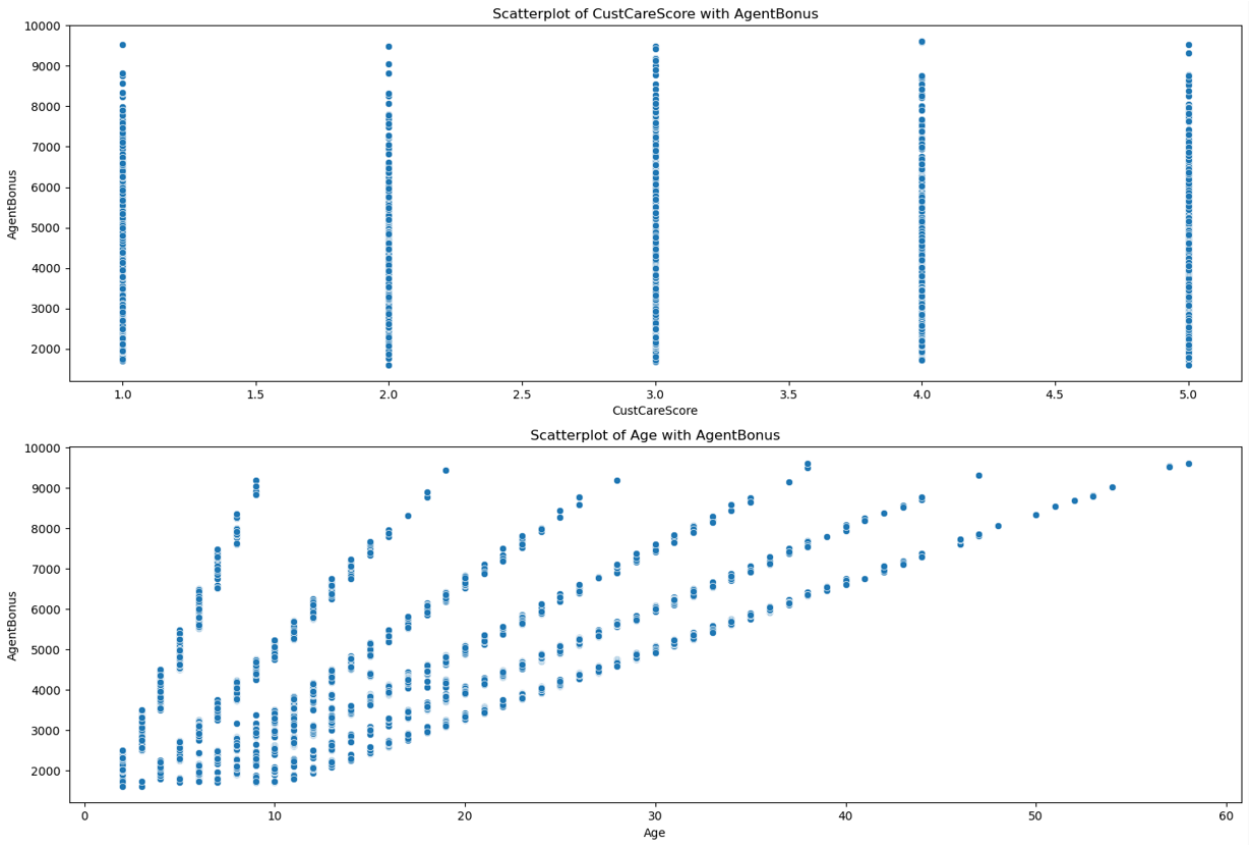
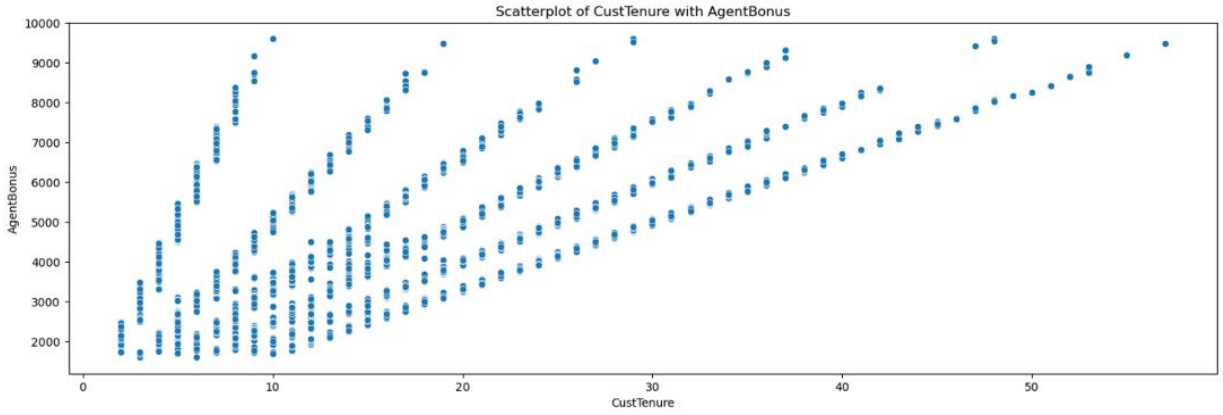
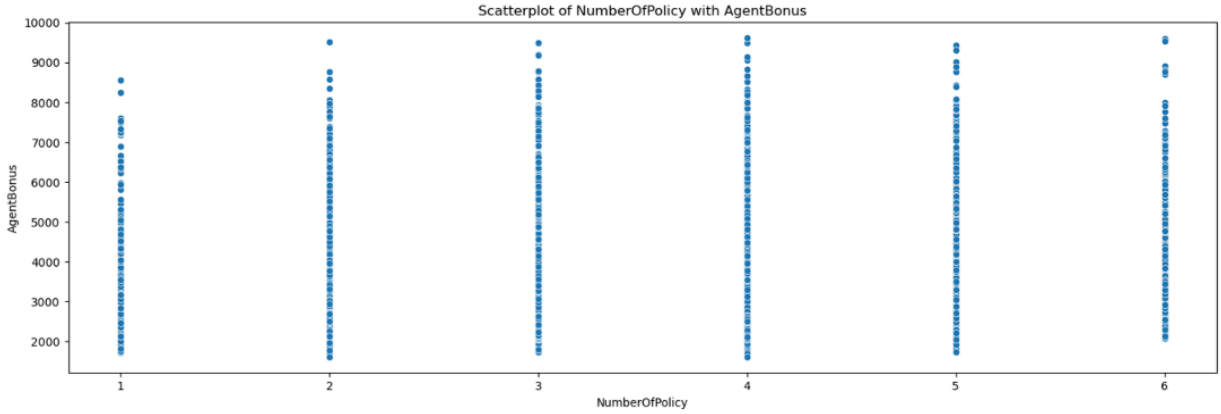


Figure 3: Bivariate analysis of categorical features with target variable

Bivariate analysis of numeric features with target variable:




Scatterplot of CustTenure with AgentBonus
AgentBonus
CustTenure
10000
9000
8000
7000
6000
5000
4000
3000
2000
0
10
20
30
40
50


Scatterplot of NumberOfPolicy with AgentBonus
AgentBonus
NumberOfPolicy
10000
9000
8000
7000
6000
5000
4000
3000
2000
1
2
3
4
5
6

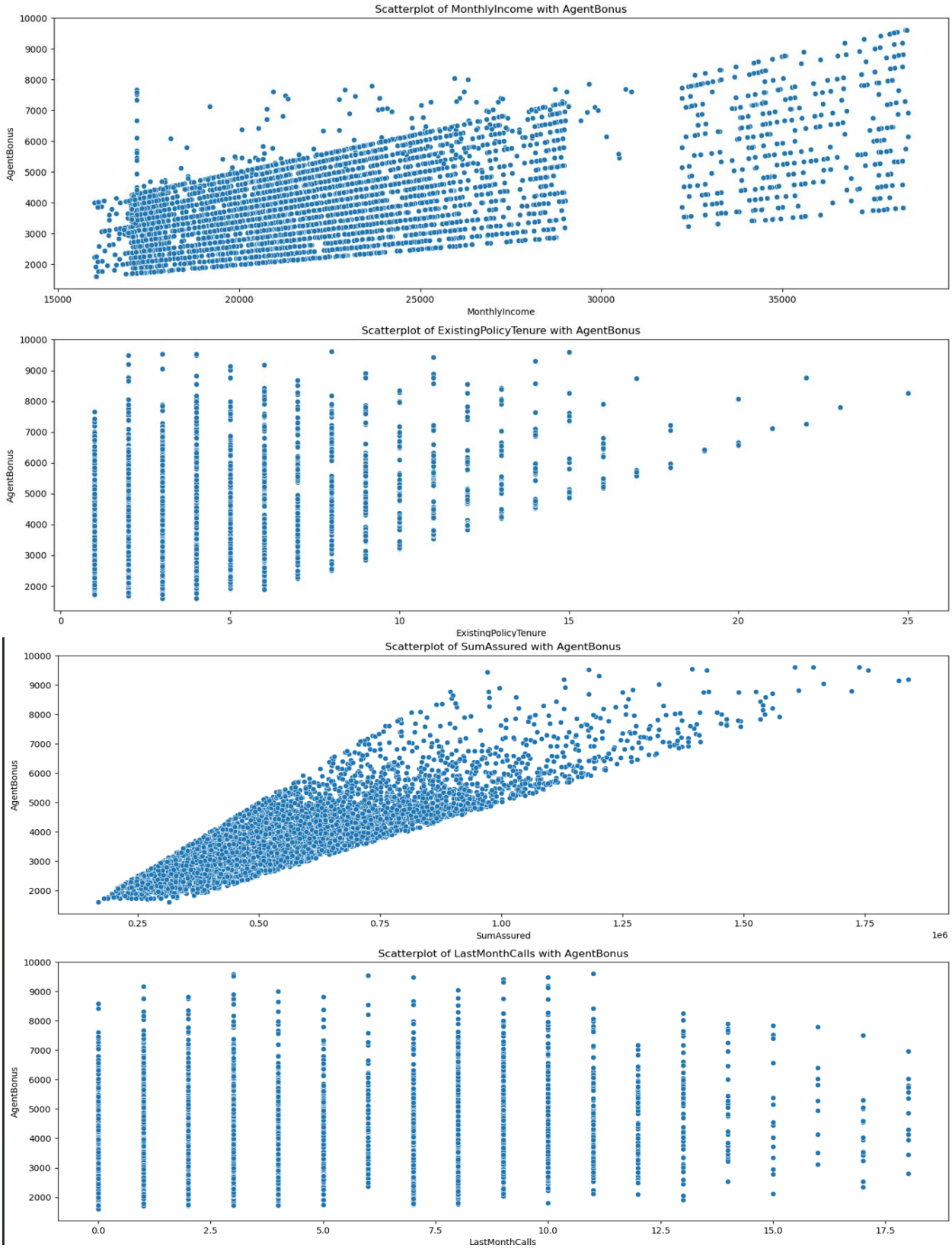


Figure 4: Bivariate analysis of numeric features with target variable

Insights from bivariate analysis:

- Agents selling insurances to VP and AVP record high bonuses.
- Agents who sold 2,4,5,6 product types have received little high bonuses compared to others.
- Agents who dealt with customers who have raised complaints have also got high bonuses as those who dealt with customers who did not raise complaints.
- Agents selling insurance to people belonging to North and West zones received high bonus
- As the age of the customer increases, the premium they pay also gets higher so the agents who sold insurance policies to older customers received high bonuses.
- Agents who were able to sell insurance policies to customers existing for a long time have received high bonuses.
- Agents who were able to sell more than 1 insurance policy received a high bonus.
- Customers having high monthly income should have opted for high premium insurances and agents who sold such insurances have received high bonuses.
- Sum assured has straightforward positive linear correlation with Agent Bonus, as the sum assured of the customer increases, AgentBonus increases.

Correlation plot of numeric variables:

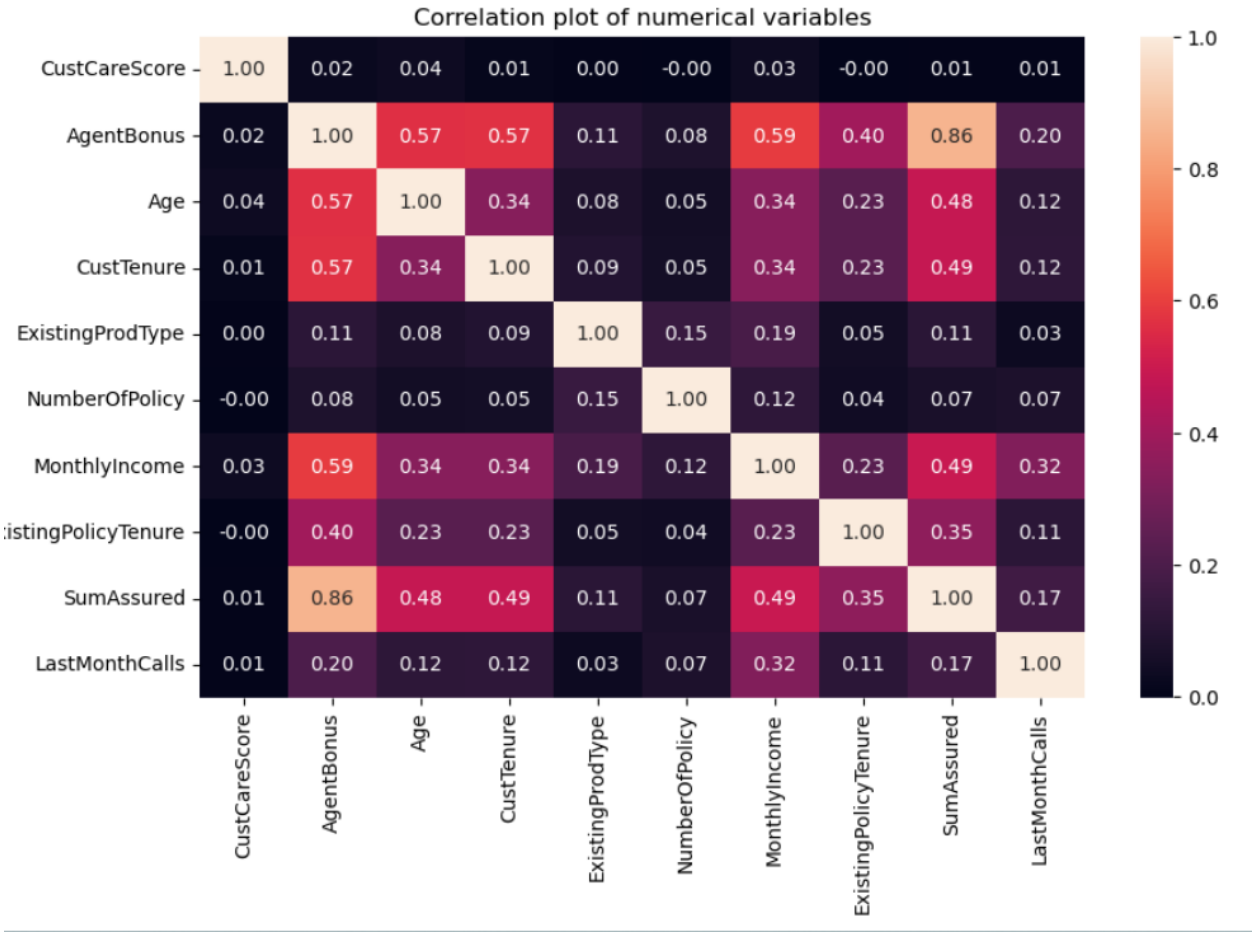


Figure 5: Correlation plot of numeric variables

Positive correlations observed from correlation plot:

- Agent bonus with Age, CustTenure, MonthlyIncome and highest with SumAssured
- CustTenure with SumAssured
- MonthlyIncome with SumAssured

g) Multivariate analysis

- **More insights by framing analytical questions and supporting plots**

1. How is MonthlyIncome and SumAssured affecting AgentBonus?
    a. Agent Bonus increases with increase in MonthlyIncome and SumAssured.
    b. Supporting graph:

Figure 6: AgentBonus distribution for people with different incomes and SumAssured

2. Do agents who did not record a complaint or have a high customer care score receive a high Agent Bonus?
    a. Agents with records of complaint and also low customer care score have also recorded high bonuses. This might indicate that the sum assured or the premium paid by such customers is high but they might have had concerns during their tenure. The company can also look into this area and also upskill the agents with after onboard support for customers.
    b. Supporting graph:

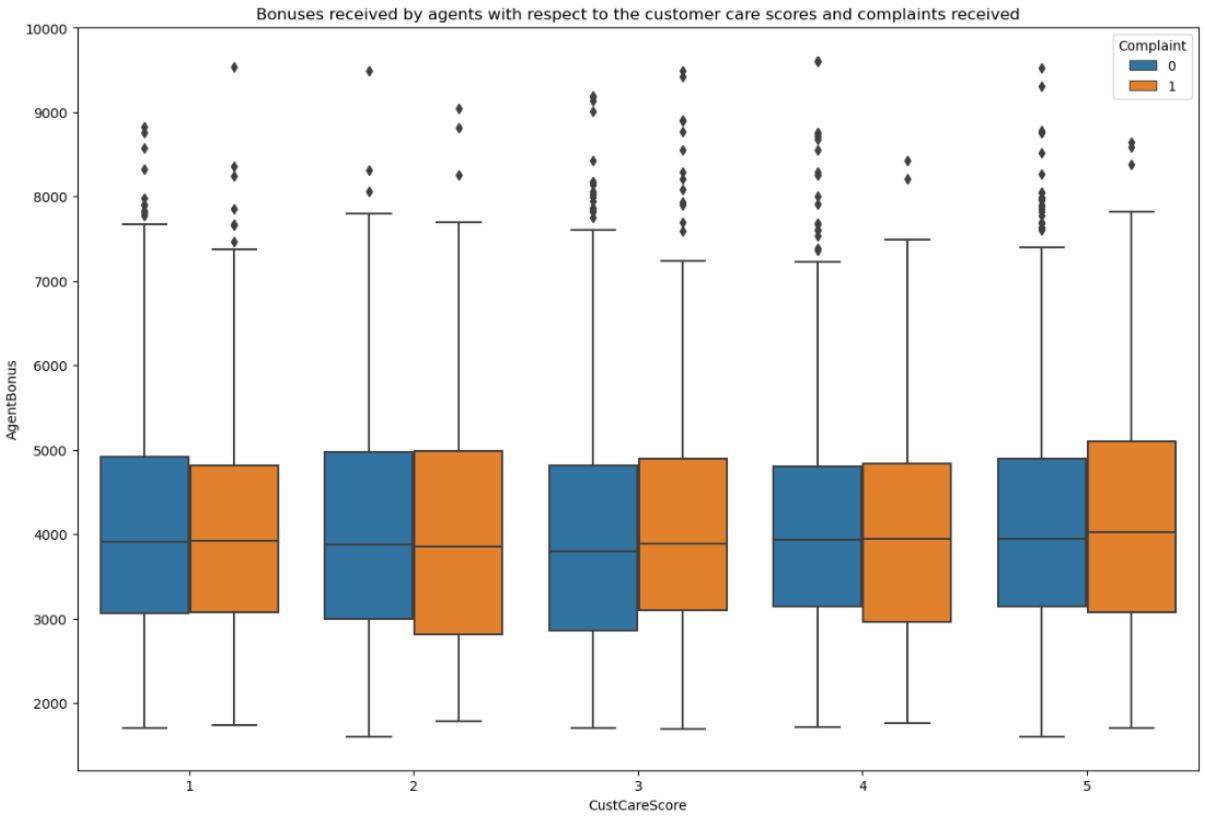


Figure 7: Customer Complaint and scores trend with AgentBonus

3. Why are only two zones performing well? What about sum assured and last month calls of people in the zones high performing vs low performing?
    a. People from South and East have taken very less policies and for very less SumAssured amounts. Last month calls were mostly less than 7 for the customers in these zones.
    b. Whereas, agents are working effectively in North and West zones in terms of SumAssured and LastMonthCalls.
    c. Supporting graph:

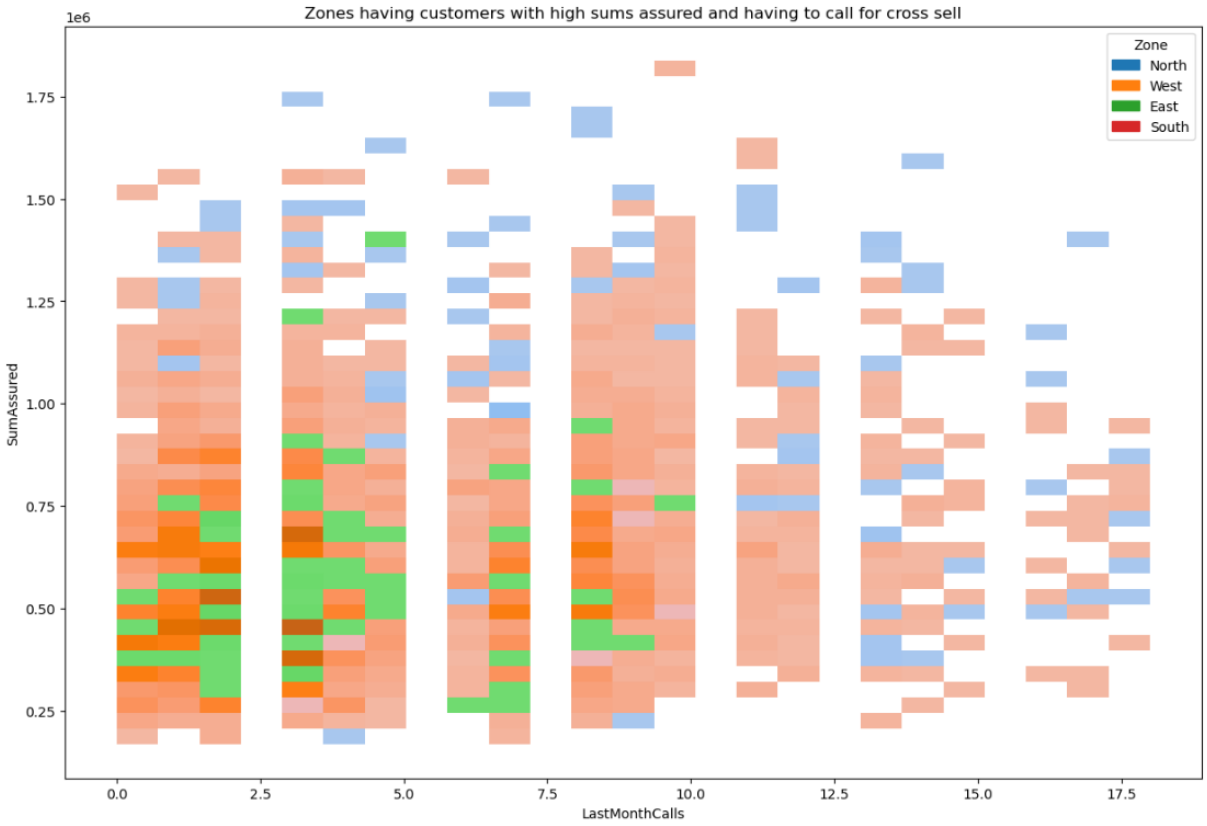
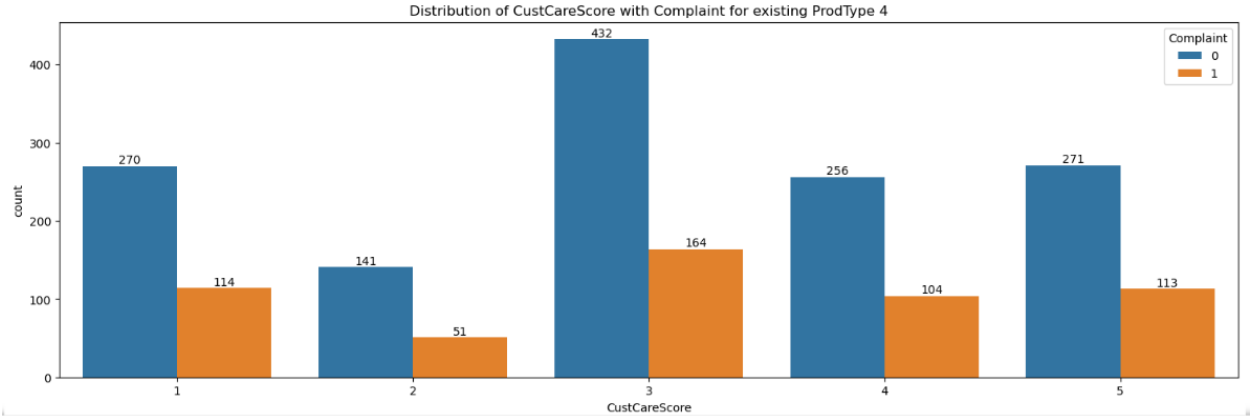


Figure 8: Zone wise sums assured and LastMonthCalls trend

4. Why is cust care score 1,2 also high? Which existing prod type people are giving which scores more?
    a. Customers who have taken product types 4 and 6 have lodged more complaints.
    b. Product type 6 is less existent among customers.
    c. 114 customers who have taken product type 4 have given a customer care score 1.
    d. Supporting graphs:



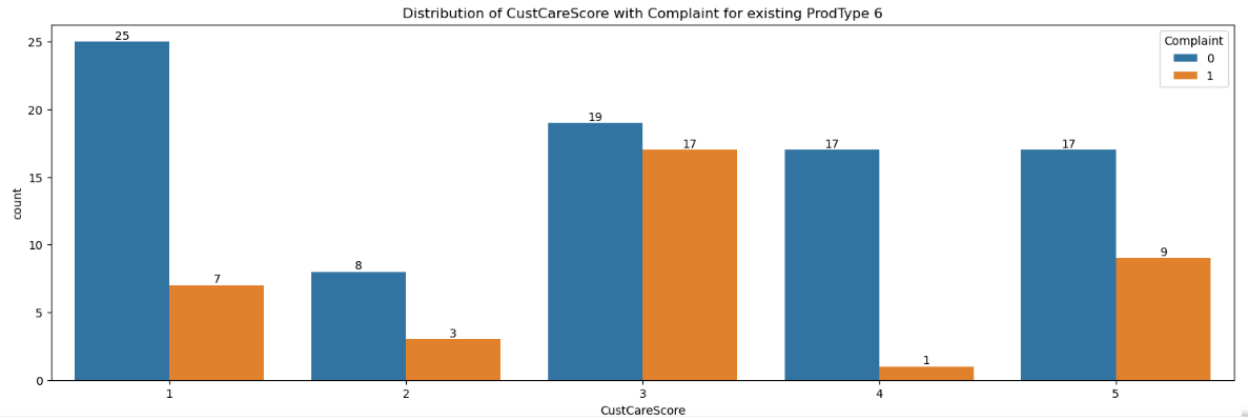


Figure 9: Customer care scores and complaint of product types 4 and 6

5. What is the age and number of policy distributions of the agents who received a high agent bonus?
    a. As the age increases, agent bonus increases with customers taking any number of policies.
    b. Supporting graph:

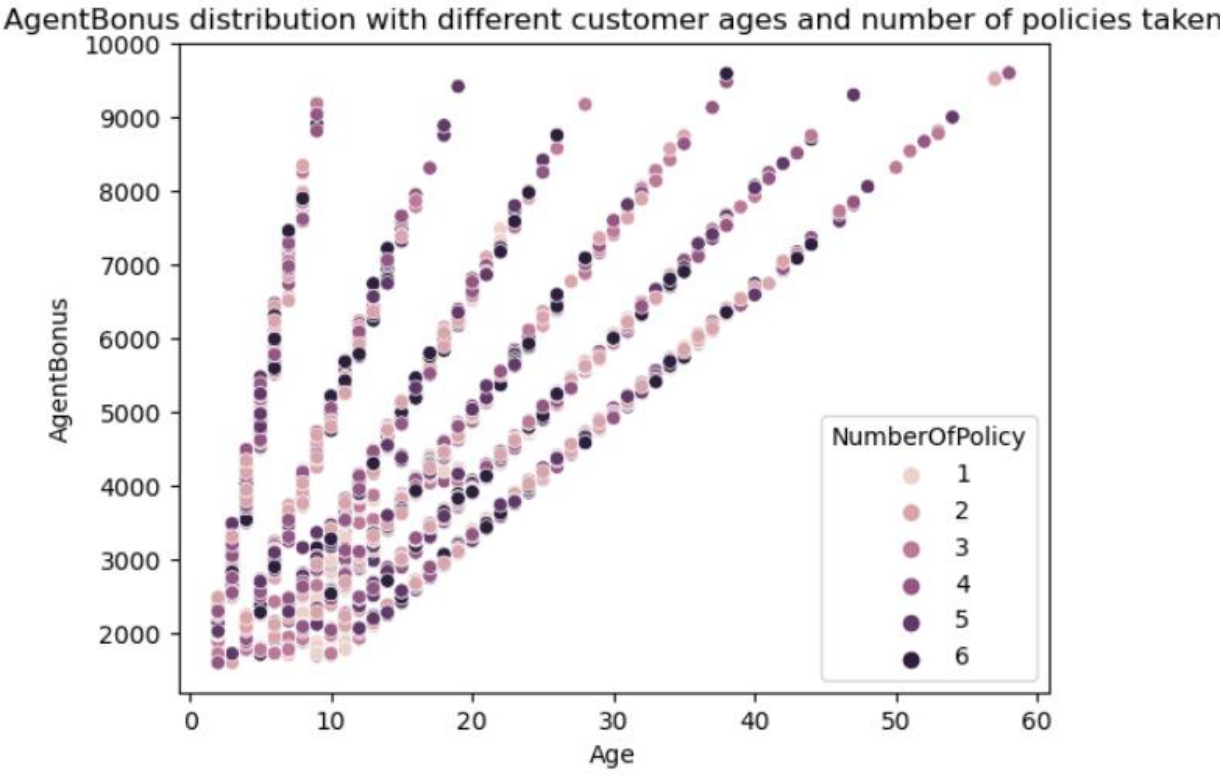


Figure 10: AgentBonus distribution with Age and NumberOfPolicies

6. What is the existing policy tenure and existing prod type for agents who received a high bonus?
    a. No matter what the existing policy tenure is, an agent receives a high bonus for any product type when they sell high SumAssured policies to customers.
    b. Supporting graph:

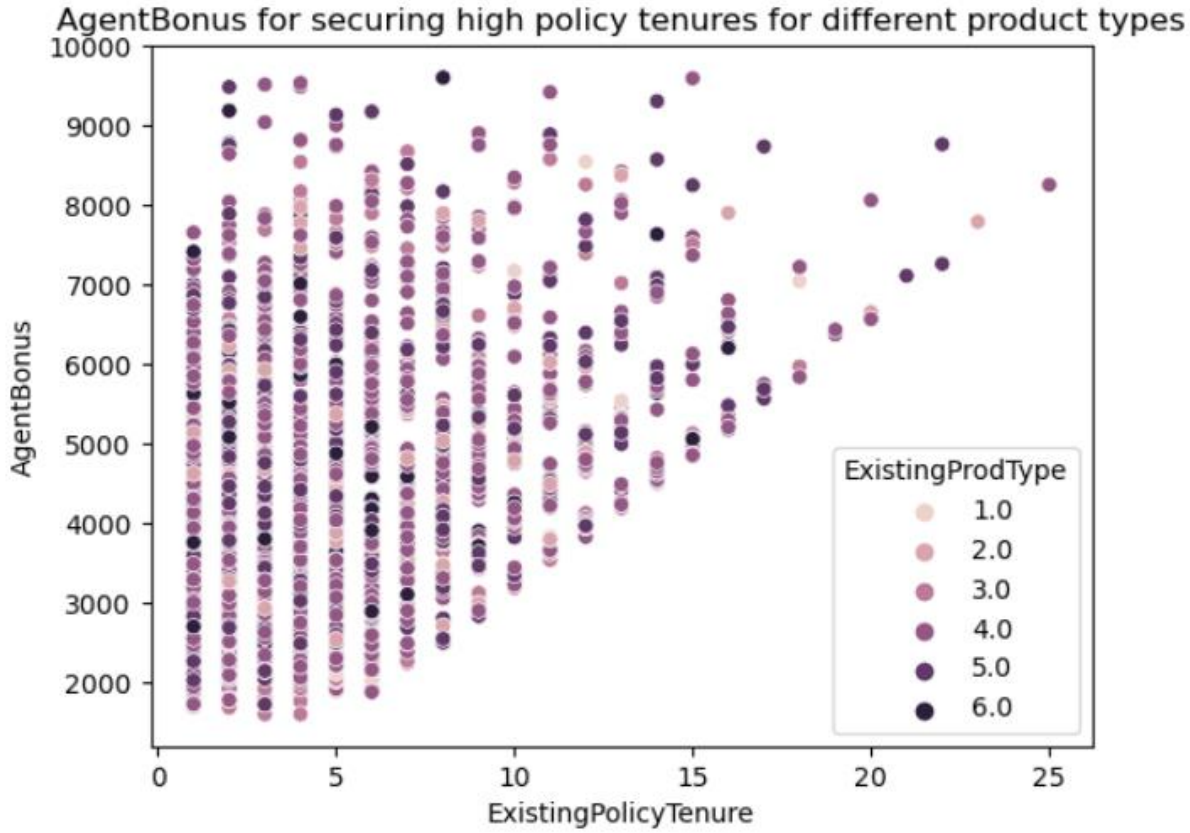


Figure 11: AgentBonus for existing product types and tenures of customers

7. How much bonuses do agents receive for selling insurance to customers belonging to different designations through different channels?
    a. Agents receive a high bonus for selling insurance to AVP and VP designates. This could be because they pay insurance with high premiums.
    b. Supporting graph:

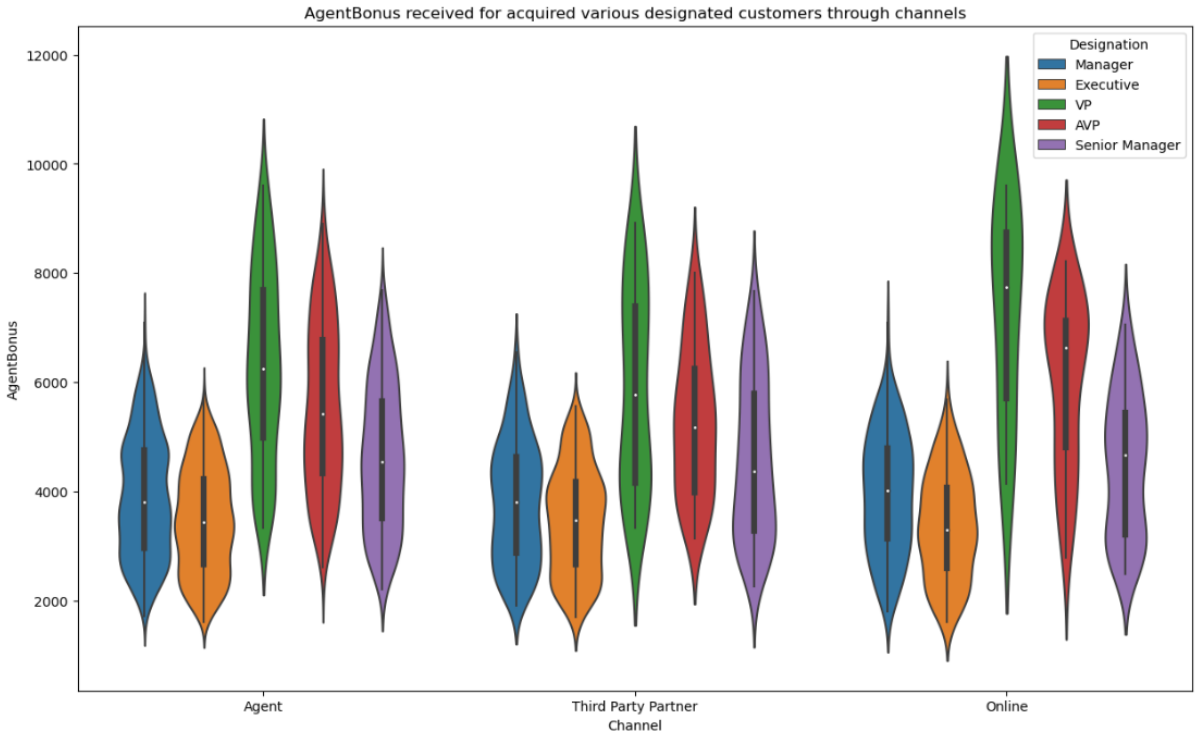


Figure 12: AgentBonus for various designated customers acquired through channels

h) Outlier treatment

- As observed in the boxplots of numerical variables individually, there are outliers present for the variables - AgentBonus, Age, CustTenure, MonthlyIncome, ExistingPolicyTenure, SumAssured, LastMonthCalls
- The above outliers do not make any deviations from the expected values according to the variables and there is no solid proof that they are erroneous. Hence we will attempt to apply the model with and without treating the outliers.
- Applied robust scaling on the variables having outliers.
- Scaling the data using a robust scaler to bring all the numeric variables to a comparable scale.
- Robust scaler scales the data by centerning the median and setting the IQR to 1. This can come handy in some cases when outliers are causing a problem.
- Capped the data to lower and upper whiskers after applying robust scaling
- Below is the analysis of numeric variables after outlier treatment

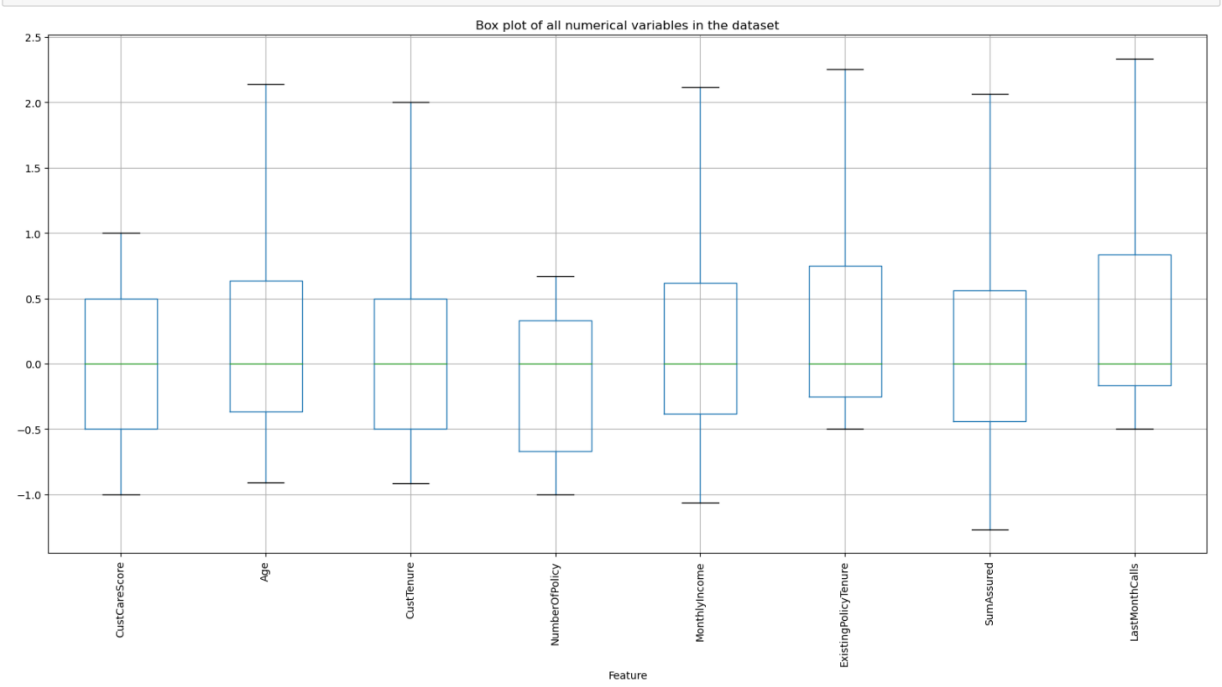


Figure 13: Outlier analysis after robust scaling and capping treatment

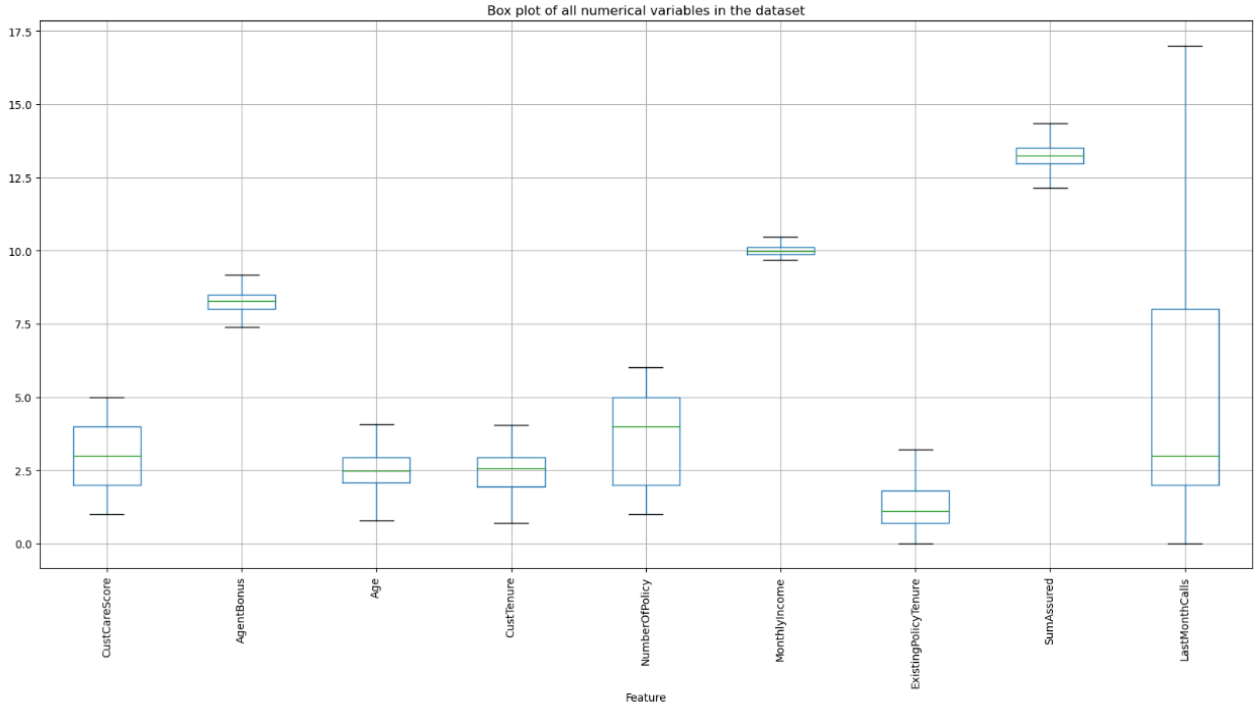
- Below is the summary of the variables after applying robust scaling and capping.
- Robust scaling has brought the median to 0 and IQR (75% - 25%) value to 1.

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
CustCareScore	4520.0	NaN	NaN	NaN	0.033407	0.687504	-1.0	-0.5	0.0	0.5	1.0
Age	4520.0	NaN	NaN	NaN	0.199692	0.767383	-0.909	-0.364	0.0	0.636	2.136
CustTenure	4520.0	NaN	NaN	NaN	0.097148	0.706875	-0.917	-0.5	0.0	0.5	2.0
NumberOfPolicy	4520.0	NaN	NaN	NaN	-0.146103	0.483343	-1.0	-0.667	0.0	0.333	0.667
MonthlyIncome	4520.0	NaN	NaN	NaN	0.198424	0.823	-1.065	-0.384	-0.0	0.61625	2.116625
ExistingPolicyTenure	4520.0	NaN	NaN	NaN	0.272843	0.760859	-0.5	-0.25	0.0	0.75	2.25
SumAssured	4520.0	NaN	NaN	NaN	0.122081	0.735652	-1.268	-0.438	0.0	0.56225	2.062625
LastMonthCalls	4520.0	NaN	NaN	NaN	0.270722	0.601757	-0.5	-0.167	0.0	0.833	2.333
Channel	4520	3	Agent	3194	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Occupation	4520	4	Salaried	2192	NaN	NaN	NaN	NaN	NaN	NaN	NaN
EducationField	4520	6	Graduate	1870	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Gender	4520	2	Male	2688	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Designation	4520	6	Manager	1620	NaN	NaN	NaN	NaN	NaN	NaN	NaN
MaritalStatus	4520	4	Married	2268	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Complaint	4520.0	2.0	0.0	3222.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Zone	4520	4	West	2566	NaN	NaN	NaN	NaN	NaN	NaN	NaN
PaymentMethod	4520	4	Half Yearly	2656	NaN	NaN	NaN	NaN	NaN	NaN	NaN
CustCareScore	4520.0	NaN	NaN	NaN	3.066814	1.375007	1.0	2.0	3.0	4.0	5.0
ExistingProdType	4520.0	6.0	4.0	1916.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Figure 14: Summary of the dataset after robust scaling and capping treatment

i) Variable transformation

- We will also use a log transformed version of the numerical variables in the dataset and train the model based on this data also.
- The idea is to use the dataset that performs better while modeling.
- For treating outliers in numeric variables in an alternate way, applied log transformations on the numeric variables that have outliers.
- Log transformation has removed outliers in some numeric variables but for some variables, the outliers are still present.
- Capped the remaining outliers to lower and upper whisker values i.e., values lower than 25% quartile to 25% quartile value. Values less than 75% quartile value to 75% quartile value.
- Outlier analysis of variables after applying log transformation and outlier treatment



-

Figure 15: Outlier analysis after log transformation and capping treatment

- Glimpse of the above treated data

	Channel	Occupation	EducationField	Gender	Designation	MaritalStatus	Complaint	Zone	PaymentMethod	CustCareScore	AgentBonus	Age	CustTen
0	Agent	Salaried	Graduate	Female	Manager	Single	1	North	Half Yearly	2.0	8.391403	3.091042	1.386
1	Third Party Partner	Salaried	Graduate	Male	Manager	Divorced	0	North	Yearly	3.0	7.702556	2.397895	0.693
2	Agent	Free Lancer	Post Graduate	Male	Exe	Unmarried	1	North	Yearly	3.0	8.360071	3.258097	1.386
3	Third Party Partner	Salaried	Graduate	Female	Executive	Divorced	1	West	Half Yearly	5.0	7.490529	2.397895	1.791
4	Agent	Small Business	Under Graduate	Male	Executive	Divorced	0	West	Half Yearly	5.0	7.991254	1.791759	2.397

ExistingProdType	NumberOfPolicy	MonthlyIncome	ExistingPolicyTenure	SumAssured	LastMonthCalls
3.0	2.0	9.951944	0.693147	13.600783	5.0
4.0	4.0	9.909967	1.098612	12.593041	7.0
4.0	3.0	9.746249	0.693147	13.443552	0.0
3.0	3.0	9.793059	0.693147	12.501109	0.0
3.0	4.0	9.823795	1.386294	12.811495	2.0

Figure 16: Viewing the dataset after log transformation and capping treatment

- Summary of the above treated data

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Channel	4520	3	Agent	3194	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Occupation	4520	4	Salaried	2192	NaN	NaN	NaN	NaN	NaN	NaN	NaN
EducationField	4520	6	Graduate	1870	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Gender	4520	2	Male	2688	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Designation	4520	6	Manager	1620	NaN	NaN	NaN	NaN	NaN	NaN	NaN
MaritalStatus	4520	4	Married	2268	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Complaint	4520.0	2.0	0.0	3222.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Zone	4520	4	West	2566	NaN	NaN	NaN	NaN	NaN	NaN	NaN
PaymentMethod	4520	4	Half Yearly	2656	NaN	NaN	NaN	NaN	NaN	NaN	NaN
CustCareScore	4520.0	NaN	NaN	NaN	3.066814	1.375007	1.0	2.0	3.0	4.0	5.0
AgentBonus	4520.0	NaN	NaN	NaN	8.25577	0.340535	7.380879	8.015575	8.271676	8.490284	9.170351
Age	4520.0	NaN	NaN	NaN	2.456335	0.677008	0.781945	2.079442	2.484907	2.944439	4.060443
CustTenure	4520.0	NaN	NaN	NaN	2.449945	0.681553	0.693147	1.94591	2.564949	2.944439	4.043051
ExistingProdType	4520.0	6.0	4.0	1916.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NumberOfPolicy	4520.0	NaN	NaN	NaN	3.561726	1.449968	1.0	2.0	4.0	5.0	6.0
MonthlyIncome	4520.0	NaN	NaN	NaN	10.012917	0.191406	9.680906	9.880789	9.977992	10.116672	10.470495
ExistingPolicyTenure	4520.0	NaN	NaN	NaN	1.128817	0.791606	0.0	0.693147	1.098612	1.791759	3.218876
SumAssured	4520.0	NaN	NaN	NaN	13.250766	0.393799	12.150953	12.977087	13.255825	13.527842	14.353976
LastMonthCalls	4520.0	NaN	NaN	NaN	4.624336	3.610676	0.0	2.0	3.0	8.0	17.0

Figure 17: Summary of the dataset after log transformation and capping treatment

4) Business insights using EDA

a) Is the data unbalanced? If so, what can be done?

- Since the target variable is a linear variable and most of the variables in the dataset are distributed in a balanced manner, the data is not unbalanced.

b) Business insights using clustering:

Below is the elbow plot of the inertia values for the 10 clusters.

Inertia values:
[271928941532129.7, 97475437843043.8, 50580540716721.25, 29948318161952.61, 19782755415811.637, 14493342931590.48, 10678872061946.232, 8245274704799.073, 6799968631945.648, 5538430055932.435]

Elbow plot:

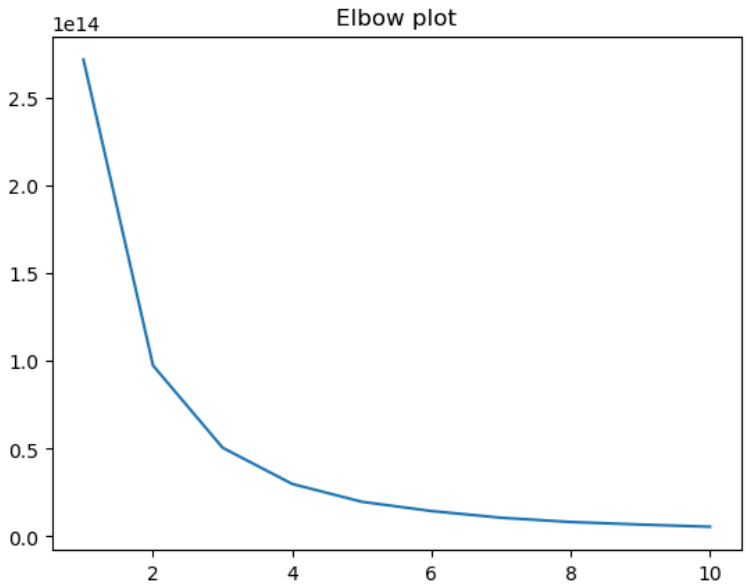


Figure 18: Elbow plot for inertia values of various number of clusters

As we can see, sharpest drop is seen at number of clusters = 2. Hence, the data has been profiled into 2 clusters.

Business insights from the clusters formed:

- Almost 66% of the data lies in cluster 0, remainder of the data lies in cluster 1.
- AgentBonus statistics:
    - Average AgentBonus for cluster 0 is Rs 3400 whereas for cluster 1 it is Rs 5446.

    - SumAssured for cluster 0 is around Rs 4.7 lakh and that of cluster 1 is Rs around Rs 9 lakh.
- Cluster 1 is the cluster comprising of agents having high AgentBonus and SumAssured.
- Cluster 0 is the cluster comprising of agents having low AgentBonus and SumAssured.
- Below are the supporting graphs:

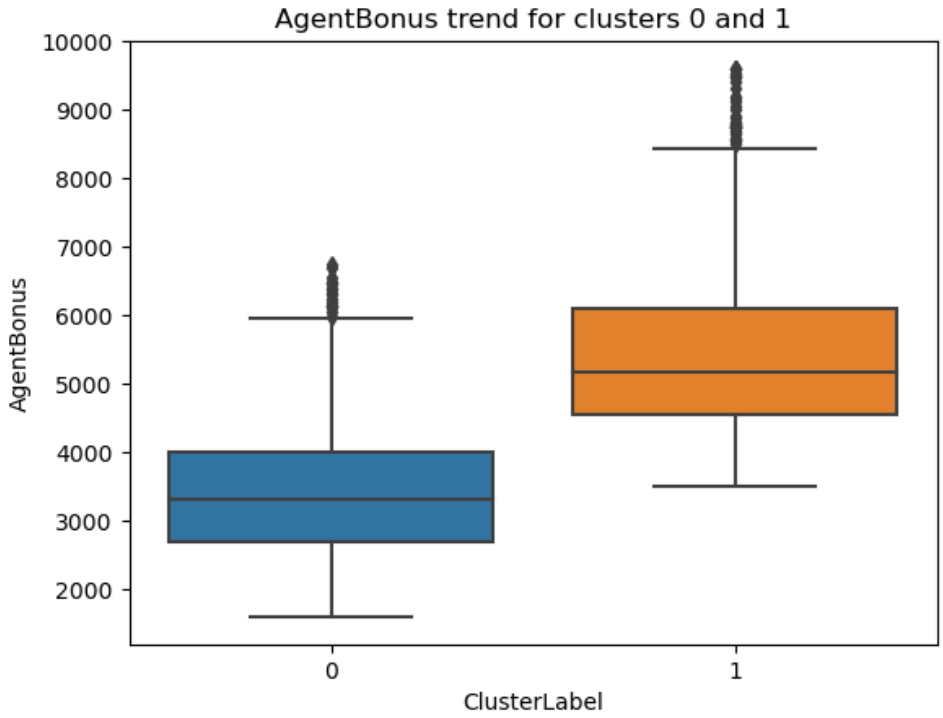


Figure 19: Agent Bonus trend for identified clusters

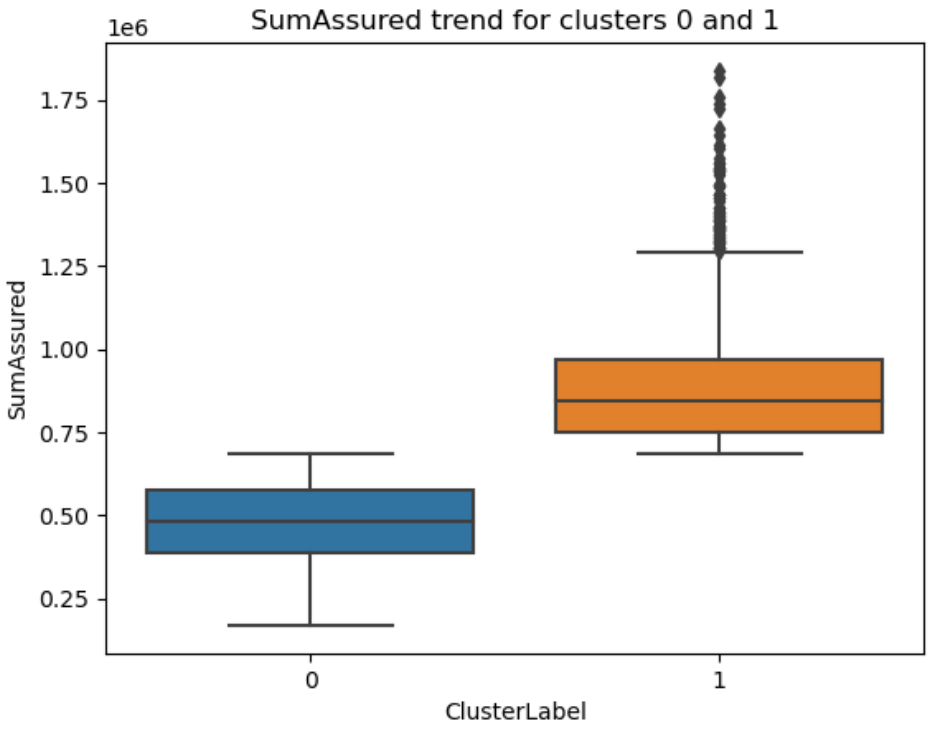


Figure 20: Sum Assured trend for identified clusters

- As the monthly income increases, the AgentBonus increases for the Agent belonging to clusters 0 and 1.
- Average tenure of the customers dealt by agents having high bonus is 19 years where as for the agents earning low bonus is around 11 years.

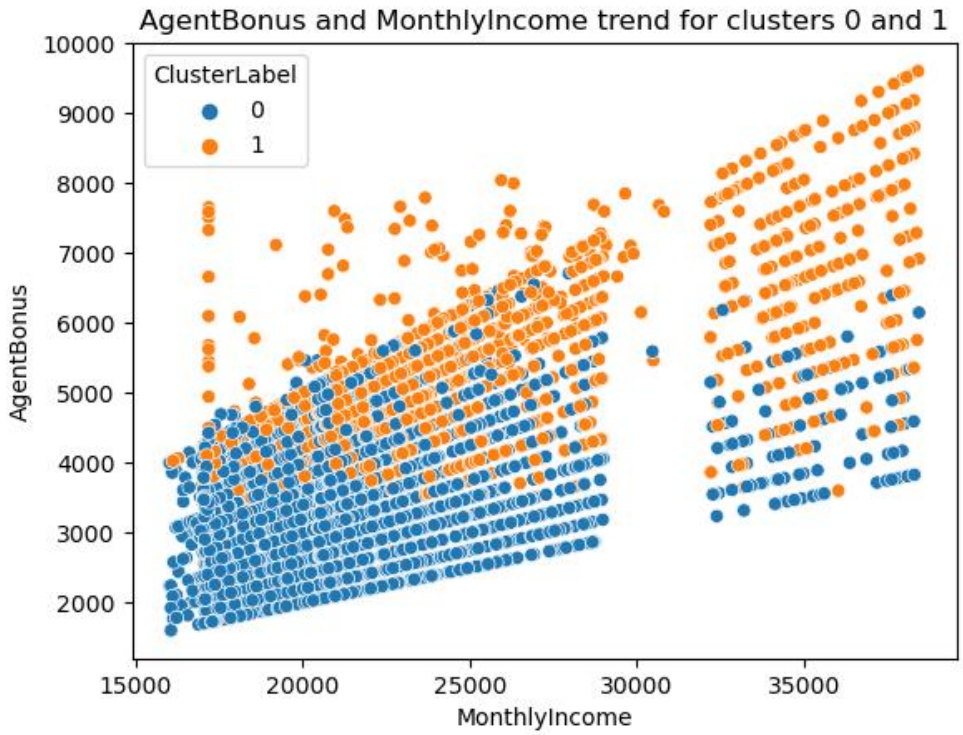


Figure 21: Agent Bonus and Monthly Income trend for identified clusters

c) Summary of business insights using EDA

- Most of the customers are acquired through agents and either salaried or running small businesses and are from North and West zones. If the company chooses to expand their presence in South and East zones, then a good amount of training and skilled agents need to be deployed in those zones.
- Agent bonus ranges from 1600Rs to 9600Rs. 75% of the agents received less than or equal to 4800 Rs.
- Agents who sold 2,4,5,6 product types have received little high bonuses compared to others. It indicates that agents selling other product types need to be involved in upskill programs.
- Sum assured has straightforward positive linear correlation with Agent Bonus, as the sum assured of the customer increases, AgentBonus increases. Agents receiving high bonuses can be involved in selling less existing or prevalent product types.
- Customers paying high premiums might have had concerns during their tenure. The company can also look into this area and also upskill the agents with after onboard support for customers.

5. Model building and interpretation

a. **Predictive, descriptive, prescriptive models**

b. **Testing models against test data**

c. **Interpretation of the model(s)**

- Each model is described, tested against the same and interpreted sequentially.
- Before applying the models, the data needs to be encoded.

Encoding the data:

- The three datasets that were considered earlier are used for modelling. Hence, they need to be encoded for categorical variables.
- Basic info of the original dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 19 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Channel               4520 non-null   object
 1   Occupation            4520 non-null   object
 2   EducationField        4520 non-null   object
 3   Gender                4520 non-null   object
 4   Designation           4520 non-null   object
 5   MaritalStatus         4520 non-null   object
 6   Complaint             4520 non-null   object
 7   Zone                  4520 non-null   object
 8   PaymentMethod         4520 non-null   object
 9   CustCareScore         4520 non-null   int64
 10  AgentBonus            4520 non-null   float64
 11  Age                   4520 non-null   int64
 12  CustTenure            4520 non-null   int64
 13  ExistingProdType      4520 non-null   object
 14  NumberOfPolicy        4520 non-null   int64
 15  MonthlyIncome         4520 non-null   float64
 16  ExistingPolicyTenure  4520 non-null   int64
 17  SumAssured            4520 non-null   float64
 18  LastMonthCalls        4520 non-null   int64
dtypes: float64(3), int64(6), object(10)
memory usage: 671.1+ KB
```

Table 10: Basic info of the original corrected dataset

- Performed one hot encoding on the data. Created dummy variables for the categories in each categorical column.
- Basic information of the above dataset after performing one hot encoding

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 39 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Complaint                       4520 non-null   int64
 1   CustCareScore                   4520 non-null   int64
 2   AgentBonus                      4520 non-null   float64
 3   Age                             4520 non-null   int64
 4   CustTenure                      4520 non-null   int64
 5   NumberOfPolicy                  4520 non-null   int64
 6   MonthlyIncome                   4520 non-null   float64
 7   ExistingPolicyTenure            4520 non-null   int64
 8   SumAssured                      4520 non-null   float64
 9   LastMonthCalls                  4520 non-null   int64
 10  Channel_Online                  4520 non-null   uint8
 11  Channel_Third Party Partner     4520 non-null   uint8
 12  Occupation_Large Business       4520 non-null   uint8
 13  Occupation_Salaried             4520 non-null   uint8
 14  Occupation_Small Business       4520 non-null   uint8
 15  EducationField_Engineer         4520 non-null   uint8
 16  EducationField_Graduate         4520 non-null   uint8
 17  EducationField_MBA              4520 non-null   uint8
 18  EducationField_Post Graduate    4520 non-null   uint8
 19  EducationField_Under Graduate   4520 non-null   uint8
 20  Gender_Male                     4520 non-null   uint8
 21  Designation_Executive           4520 non-null   uint8
 22  Designation_Manager             4520 non-null   uint8
 23  Designation_Senior Manager      4520 non-null   uint8
 24  Designation_VP                  4520 non-null   uint8
 25  MaritalStatus_Married           4520 non-null   uint8
 26  MaritalStatus_Single            4520 non-null   uint8
 27  MaritalStatus_Unmarried         4520 non-null   uint8
 28  Zone_North                      4520 non-null   uint8
 29  Zone_South                      4520 non-null   uint8
 30  Zone_West                       4520 non-null   uint8
 31  PaymentMethod_Monthly           4520 non-null   uint8
 32  PaymentMethod_Quarterly         4520 non-null   uint8
 33  PaymentMethod_Yearly            4520 non-null   uint8
 34  ExistingProdType_2.0            4520 non-null   uint8
 35  ExistingProdType_3.0            4520 non-null   uint8
 36  ExistingProdType_4.0            4520 non-null   uint8
 37  ExistingProdType_5.0            4520 non-null   uint8
 38  ExistingProdType_6.0            4520 non-null   uint8
dtypes: float64(3), int64(7), uint8(29)
memory usage: 481.3 KB
```

Table 11: Basic info of the original corrected encoded dataset

Encoding the scaled data:

- Basic info of the original dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   CustCareScore         4520 non-null   float64
 1   Age                   4520 non-null   float64
 2   CustTenure            4520 non-null   float64
 3   NumberOfPolicy        4520 non-null   float64
 4   MonthlyIncome         4520 non-null   float64
 5   ExistingPolicyTenure  4520 non-null   float64
 6   SumAssured            4520 non-null   float64
 7   LastMonthCalls        4520 non-null   float64
 8   AgentBonus            4520 non-null   float64
 9   Channel               4520 non-null   object
 10  Occupation            4520 non-null   object
 11  EducationField        4520 non-null   object
 12  Gender                4520 non-null   object
 13  Designation           4520 non-null   object
 14  MaritalStatus         4520 non-null   object
 15  Complaint             4520 non-null   object
 16  Zone                  4520 non-null   object
 17  PaymentMethod         4520 non-null   object
 18  CustCareScore         4520 non-null   int64
 19  ExistingProdType      4520 non-null   object
 20  AgentBonus            4520 non-null   float64
dtypes: float64(10), int64(1), object(10)
memory usage: 741.7+ KB
```

-

Table 12: Basic info of the scaled dataset

- Performed one hot encoding on the data. Created dummy variables for the categories in each categorical column.

- Basic information of the above dataset after performing one hot encoding.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 41 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   CustCareScore                   4520 non-null   float64
 1   Age                             4520 non-null   float64
 2   CustTenure                      4520 non-null   float64
 3   NumberOfPolicy                  4520 non-null   float64
 4   MonthlyIncome                   4520 non-null   float64
 5   ExistingPolicyTenure            4520 non-null   float64
 6   SumAssured                      4520 non-null   float64
 7   LastMonthCalls                  4520 non-null   float64
 8   AgentBonus                      4520 non-null   float64
 9   Complaint                       4520 non-null   int64
 10  CustCareScore                   4520 non-null   int64
 11  AgentBonus                      4520 non-null   float64
 12  Channel_Online                  4520 non-null   uint8
 13  Channel_Third Party Partner     4520 non-null   uint8
 14  Occupation_Large Business       4520 non-null   uint8
 15  Occupation_Salaried             4520 non-null   uint8
 16  Occupation_Small Business       4520 non-null   uint8
 17  EducationField_Engineer         4520 non-null   uint8
 18  EducationField_Graduate         4520 non-null   uint8
 19  EducationField_MBA              4520 non-null   uint8
 20  EducationField_Post Graduate    4520 non-null   uint8
 21  EducationField_Under Graduate   4520 non-null   uint8
 22  Gender_Male                     4520 non-null   uint8
 23  Designation_Executive           4520 non-null   uint8
 24  Designation_Manager             4520 non-null   uint8
 25  Designation_Senior Manager      4520 non-null   uint8
 26  Designation_VP                  4520 non-null   uint8
 27  MaritalStatus_Married           4520 non-null   uint8
 28  MaritalStatus_Single            4520 non-null   uint8
 29  MaritalStatus_Unmarried         4520 non-null   uint8
 30  Zone_North                      4520 non-null   uint8
 31  Zone_South                      4520 non-null   uint8
 32  Zone_West                       4520 non-null   uint8
 33  PaymentMethod_Monthly           4520 non-null   uint8
 34  PaymentMethod_Quarterly         4520 non-null   uint8
 35  PaymentMethod_Yearly            4520 non-null   uint8
 36  ExistingProdType_2.0            4520 non-null   uint8
 37  ExistingProdType_3.0            4520 non-null   uint8
 38  ExistingProdType_4.0            4520 non-null   uint8
 39  ExistingProdType_5.0            4520 non-null   uint8
 40  ExistingProdType_6.0            4520 non-null   uint8
dtypes: float64(10), int64(2), uint8(29)
memory usage: 551.9 KB
```

Table 13: Basic info of the scaled encoded dataset

Encoding the log transformed data:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 19 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Channel               4520 non-null   object
 1   Occupation            4520 non-null   object
 2   EducationField        4520 non-null   object
 3   Gender                4520 non-null   object
 4   Designation           4520 non-null   object
 5   MaritalStatus         4520 non-null   object
 6   Complaint             4520 non-null   object
 7   Zone                  4520 non-null   object
 8   PaymentMethod         4520 non-null   object
 9   CustCareScore         4520 non-null   float64
 10  AgentBonus            4520 non-null   float64
 11  Age                   4520 non-null   float64
 12  CustTenure            4520 non-null   float64
 13  ExistingProdType      4520 non-null   object
 14  NumberOfPolicy        4520 non-null   float64
 15  MonthlyIncome         4520 non-null   float64
 16  ExistingPolicyTenure  4520 non-null   float64
 17  SumAssured            4520 non-null   float64
 18  LastMonthCalls        4520 non-null   float64
dtypes: float64(9), object(10)
memory usage: 671.1+ KB
```

Table 14: Basic info of the log transformed dataset

- Performed one hot encoding on the data. Created dummy variables for the categories in each categorical column.

- Basic information of the above dataset after performing one hot encoding.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4520 entries, 0 to 4519
Data columns (total 39 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Complaint                       4520 non-null   int64
 1   CustCareScore                   4520 non-null   float64
 2   AgentBonus                      4520 non-null   float64
 3   Age                             4520 non-null   float64
 4   CustTenure                      4520 non-null   float64
 5   NumberOfPolicy                  4520 non-null   float64
 6   MonthlyIncome                   4520 non-null   float64
 7   ExistingPolicyTenure            4520 non-null   float64
 8   SumAssured                      4520 non-null   float64
 9   LastMonthCalls                  4520 non-null   float64
 10  Channel_Online                  4520 non-null   uint8
 11  Channel_Third Party Partner     4520 non-null   uint8
 12  Occupation_Large Business       4520 non-null   uint8
 13  Occupation_Salaried             4520 non-null   uint8
 14  Occupation_Small Business       4520 non-null   uint8
 15  EducationField_Engineer         4520 non-null   uint8
 16  EducationField_Graduate         4520 non-null   uint8
 17  EducationField_MBA              4520 non-null   uint8
 18  EducationField_Post Graduate    4520 non-null   uint8
 19  EducationField_Under Graduate   4520 non-null   uint8
 20  Gender_Male                     4520 non-null   uint8
 21  Designation_Executive           4520 non-null   uint8
 22  Designation_Manager             4520 non-null   uint8
 23  Designation_Senior Manager      4520 non-null   uint8
 24  Designation_VP                  4520 non-null   uint8
 25  MaritalStatus_Married           4520 non-null   uint8
 26  MaritalStatus_Single            4520 non-null   uint8
 27  MaritalStatus_Unmarried         4520 non-null   uint8
 28  Zone_North                      4520 non-null   uint8
 29  Zone_South                      4520 non-null   uint8
 30  Zone_West                       4520 non-null   uint8
 31  PaymentMethod_Monthly           4520 non-null   uint8
 32  PaymentMethod_Quarterly         4520 non-null   uint8
 33  PaymentMethod_Yearly            4520 non-null   uint8
 34  ExistingProdType_2.0            4520 non-null   uint8
 35  ExistingProdType_3.0            4520 non-null   uint8
 36  ExistingProdType_4.0            4520 non-null   uint8
 37  ExistingProdType_5.0            4520 non-null   uint8
 38  ExistingProdType_6.0            4520 non-null   uint8
dtypes: float64(9), int64(1), uint8(29)
memory usage: 481.3 KB
```

Table 15: Basic info of the log transformed encoded dataset

Brief overview of the metrics considered for building the models:

- **R-squared**: Explains the percent of the variance explained by the model on the training and test datasets respectively. Higher R-sqaured indicates more of the data in the dataset captured by the model.
- **RMSE**(RootMeanSquaredError): - Root mean square error or root mean square deviation is one of the most used measures for evaluating the quality of predictions. It shows how far predictions fall from measured true values using Euclidean distance. It is the square root of sums of error terms.
- **MAE**: It shows how far predictions fall from measured true values using Euclidean distance. It is the absolute value of the sums of the error terms.
- **Coefficients**: The coefficients and the linear equation built by the model. This equation will be used to predict the values of the target variable in the test dataset.
- **Intercept**: Slope of the equation built by the linear regression model.

Model 1: Linear Regression on original outlier present encoded dataset

Linear Regression Model:

- Linear Regression fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation.

Steps performed:

- The original encoded data has been split into training and test datasets with 70 : 30 ratio.
- Linear Regression model has been applied with all default parameters.
    - Fit_intercept = true (calculates intercept for the linear equation formed)
    - Normalize = false (does not normalize the data before applying model)
    - Positive = false (does not force all coefficients of the equation to be False)
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the linear regression model:

R-squared	Train	0.8254454899379167
	Test	0.8106555446763404
RMSE	Train	587.2937848801013
	Test	607.9724350233483
MAE	Train	461.2067534401857
	Test	480.06573546745017
Equation	(45.45) * Complaint + (7.094) * CustCareScore + (21.471) * Age + (22.55) * CustTenure + (5.186) * NumberOfPolicy + (0.036) * MonthlyIncome + (33.734) * ExistingPolicyTenure + (0.003) * SumAssured + (-4.465) * LastMonthCalls + (41.414) * Channel_Online + (13.501) * Channel_Third Party Partner + (-247.787) * Occupation_Large Business + (-154.668) * Occupation_Salaried + (-273.224) * Occupation_Small Business + (-19.209) * EducationField_Engineer + (-104.356) * EducationField_Graduate + (-161.448) * EducationField_MBA + (-71.789) * EducationField_Post Graduate + (-2.274) * EducationField_Under Graduate + (22.914) * Gender_Male + (-320.621) * Designation_Executive + (-351.371) *	

	Designation_Manager + (-165.795) * Designation_Senior Manager + (-69.794) * Designation_VP + (-45.743) * MaritalStatus_Married + (26.493) * MaritalStatus_Single + (-67.565) * MaritalStatus_Unmarried + (66.093) * Zone_North + (213.306) * Zone_South + (72.486) * Zone_West + (-38.507) * PaymentMethod_Monthly + (-10.389) * PaymentMethod_Quarterly + (-45.345) * PaymentMethod_Yearly + (73.562) * ExistingProdType_2.0 + (-95.425) * ExistingProdType_3.0 + (-53.263) * ExistingProdType_4.0 + (-62.889) * ExistingProdType_5.0 + (-48.188) * ExistingProdType_6.0
Intercept	[885.1870254]

Table 16: Metrics: Linear Regression on outlier present encoded dataset

Interpretation:

- The model was able to explain 82% and 81% of the variance in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 587 and test set is around 607.
- The RMSE values and MAE values are high which indicates that the data is not very suitable for applying linear regression on.
- From the equation built, agent bonus increases when values for the attributes like Complaint, CustCareScore, SumAssured increases.
- AgentBonus decreases when the values for the attributes like LastMonthCalls, Occupation_LargeBusiness, Occupation_SmallBusiness, EducationField_Graduate, Designation_Manager, ExistingProdType_3.0, ExistingProdType_4.0.

Model 2: Linear Regression on scaled encoded dataset

Linear Regression Model:

- Linear Regression fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation.

Steps performed:

- The robust scaled and encoded data has been split into training and test datasets with 70 : 30 ratio.
- Linear Regression model has been applied with all default parameters.
    - Fit_intercept = true: calculates intercept for the linear equation formed.
    - Normalize = false: does not normalize the data before applying model.
    - Positive = false: does not force all coefficients of the equation to be False.
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the linear regression model:

R-squared	Train	0.8207060750319559
	Test	0.8035902375130146
RMSE	Train	0.3126778308009761
	Test	0.32697983989619184
MAE	Train	0.24730516405262326
	Test	0.25800052429019177
Equation	((-4418167349938.303) * CustCareScore + (0.122) * Age + (0.139) * CustTenure + (0.01) * NumberOfPolicy + (0.109) * MonthlyIncome + (0.073) * ExistingPolicyTenure + (0.612) * SumAssured + (-0.009) * LastMonthCalls + (0.019) * Complaint + (2209083674969.152) * CustCareScore + (0.022) * Channel_Online + (0.005) * Channel_Third Party Partner + (-0.129) * Occupation_Large Business + (-0.085) * Occupation_Salaried + (-0.148) * Occupation_Small Business + (-0.019) * EducationField_Engineer + (-0.058) *		

	EducationField_Graduate + (-0.092) * EducationField_MBA + (-0.046) * EducationField_Post Graduate + (-0.001) * EducationField_Under Graduate + (0.013) * Gender_Male + (-0.197) * Designation_Executive + (-0.219) * Designation_Manager + (-0.12) * Designation_Senior Manager + (-0.043) * Designation_VP + (-0.03) * MaritalStatus_Married + (0.009) * MaritalStatus_Single + (-0.035) * MaritalStatus_Unmarried + (0.033) * Zone_North + (0.106) * Zone_South + (0.03) * Zone_West + (-0.051) * PaymentMethod_Monthly + (-0.01) * PaymentMethod_Quarterly + (-0.019) * PaymentMethod_Yearly + (0.034) * ExistingProdType_2.0 + (-0.083) * ExistingProdType_3.0 + (-0.061) * ExistingProdType_4.0 + (-0.076) * ExistingProdType_5.0 + (-0.04) * ExistingProdType_6.0
Intercept	[-6.62725102e+12]

Table 17: Metrics: Linear Regression on scaled encoded dataset

Interpretation:

- The model was able to explain 82% and 80% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.31 and test set is around 0.32.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original dataset.
- From the equation built, agent bonus increases when values for the attributes like Complaint, CustCareScore, SumAssured increases.
- AgentBonus decreases when the values for the attributes like LastMonthCalls, Occupation_LargeBusiness, Occupation_SmallBusiness, EducationField_Graduate, Designation_Manager, ExistingProdType_3.0, ExistingProdType_4.0.

Model 3: Linear Regression on log transformed and encoded data set

Linear Regression Model:

- Linear Regression fits a linear model with coefficients w = (w1, …, wp) to minimize the residual sum of squares between the observed targets in the dataset, and the targets predicted by the linear approximation.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- Linear Regression model has been applied with all default parameters.
    - Fit_intercept = true: calculates intercept for the linear equation formed.
    - Normalize = false: does not normalize the data before applying model.
    - Positive = false: does not force all coefficients of the equation to be False.
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the linear regression model:

R-squared	Train	0.817538422918356
	Test	0.79431228196901
RMSE	Train	0.14564174905328822
	Test	0.15393147953692923
MAE	Train	0.11849763555061978
	Test	0.12414303919785241
Equation	(0.009) * Complaint + (0.002) * CustCareScore + (0.059) * Age + (0.059) * CustTenure + (0.002) * NumberOfPolicy + (0.18) * MonthlyIncome + (0.032) * ExistingPolicyTenure + (0.566) * SumAssured + (-0.0) * LastMonthCalls + (0.009) * Channel_Online + (0.001) * Channel_Third Party Partner + (-0.08) * Occupation_Large Business + (-0.053) * Occupation_Salaried + (-0.077) * Occupation_Small Business + (0.003) * EducationField_Engineer + (-0.023) * EducationField_Graduate + (-0.034) * EducationField_MBA + (-0.019) * E	

	ducationField_Post Graduate + (-0.0) * EducationField_Under Graduate + (0.006) * Gender_Male + (-0.077) * Designation_Executive + (-0.08) * Designation_Manager + (-0.037) * Designation_Senior Manager + (-0.011) * Designation_VP + (-0.014) * MaritalStatus_Married + (0.005) * MaritalStatus_Single + (-0.028) * MaritalStatus_Unmarried + (0.011) * Zone_North + (0.049) * Zone_South + (0.015) * Zone_West + (-0.014) * PaymentMethod_Monthly + (-0.002) * PaymentMethod_Quarterly + (-0.008) * PaymentMethod_Yearly + (0.018) * ExistingProdType_2.0 + (-0.031) * ExistingProdType_3.0 + (-0.021) * ExistingProdType_4.0 + (-0.022) * ExistingProdType_5.0 + (-0.008) * ExistingProdType_6.0
Intercept	[-1.23671641]

Table 18: Metrics: Linear Regression on log transformed encoded dataset

Interpretation:

- The model was able to explain 81% and 79% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.14 and test set is around 0.15.
- The RMSE values and MAE values are significantly decreased and are better than the model built with scaled data, which indicates that the data is most suitable for applying linear regression on.
- From the equation built, agent bonus increases when values for the attributes like Complaint, CustCareScore, MonthlyIncome increases.
- AgentBonus decreases when values for the attributes like Zone_West, Designation_Manager etc. increase.

Model 4: Linear Regression on log transformed encoded dataset using Ordinal Least Squares method from Statsmodels library.

- As we can see in the above three models, the log transformed and encoded dataset has the least RMSE value with good R-squared value.
- Hence, we will consider the log transformed dataset for our further modelling.
- In this model, Ordinal Least Squares method from Statsmodels library has been used.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- Linear Regression using Ordinal Least Squares method has been applied with default parameters listed below.
    - Endog: response variable of trained dataset
    - Exog: independent variable of the trained dataset
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

- Summary of the regression model:

OLS Regression Results

Dep. Variable:	AgentBonus	**R-squared:**	0.818
Model:	OLS	**Adj. R-squared:**	0.815
Method:	Least Squares	**F-statistic:**	368.5
Date:	Sat, 04 Nov 2023	**Prob (F-statistic):**	0.00
Time:	13:12:56	**Log-Likelihood:**	1606.3
No. Observations:	3164	**AIC:**	-3135.
Df Residuals:	3125	**BIC:**	-2898.
Df Model:	38		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-1.2367	0.286	-4.324	0.000	-1.797	-0.676
Complaint	0.0091	0.006	1.567	0.117	-0.002	0.021
CustCareScore	0.0023	0.002	1.188	0.235	-0.001	0.006
Age	0.0591	0.004	13.770	0.000	0.051	0.067
CustTenure	0.0585	0.004	13.594	0.000	0.050	0.067
NumberOfPolicy	0.0019	0.002	1.023	0.306	-0.002	0.006
MonthlyIncome	0.1798	0.025	7.059	0.000	0.130	0.230
ExistingPolicyTenure	0.0322	0.004	9.044	0.000	0.025	0.039
SumAssured	0.5664	0.009	65.084	0.000	0.549	0.583
LastMonthCalls	-0.0002	0.001	-0.291	0.771	-0.002	0.001
Channel_Online	0.0088	0.009	1.020	0.308	-0.008	0.026
Channel_Third Party Partner	0.0010	0.007	0.150	0.881	-0.012	0.014
Occupation_Large Business	-0.0803	0.111	-0.724	0.469	-0.298	0.137
Occupation_Salaried	-0.0533	0.105	-0.509	0.611	-0.259	0.152
Occupation_Small Business	-0.0767	0.107	-0.718	0.473	-0.286	0.133
EducationField_Engineer	0.0031	0.038	0.081	0.935	-0.071	0.078
EducationField_Graduate	-0.0228	0.022	-1.023	0.306	-0.066	0.021
EducationField_MBA	-0.0344	0.032	-1.089	0.276	-0.096	0.028
EducationField_Post Graduate	-0.0194	0.025	-0.771	0.441	-0.069	0.030
EducationField_Under Graduate	-0.0005	0.009	-0.050	0.960	-0.018	0.018

-

Gender_Male	0.0060	0.005	1.119	0.263	-0.005	0.017
Designation_Executive	-0.0765	0.014	-5.544	0.000	-0.104	-0.049
Designation_Manager	-0.0799	0.012	-6.721	0.000	-0.103	-0.057
Designation_Senior Manager	-0.0365	0.012	-3.056	0.002	-0.060	-0.013
Designation_VP	-0.0112	0.016	-0.686	0.493	-0.043	0.021
MaritalStatus_Married	-0.0143	0.007	-1.978	0.048	-0.028	-0.000
MaritalStatus_Single	0.0048	0.008	0.597	0.551	-0.011	0.020
MaritalStatus_Unmarried	-0.0285	0.015	-1.898	0.058	-0.058	0.001
Zone_North	0.0114	0.023	0.503	0.615	-0.033	0.056
Zone_South	0.0490	0.069	0.705	0.481	-0.087	0.185
Zone_West	0.0149	0.023	0.661	0.508	-0.029	0.059
PaymentMethod_Monthly	-0.0136	0.030	-0.459	0.646	-0.072	0.045
PaymentMethod_Quarterly	-0.0020	0.024	-0.084	0.933	-0.049	0.045
PaymentMethod_Yearly	-0.0077	0.009	-0.856	0.392	-0.025	0.010
ExistingProdType_2.0	0.0177	0.018	0.975	0.329	-0.018	0.053
ExistingProdType_3.0	-0.0309	0.031	-0.992	0.321	-0.092	0.030
ExistingProdType_4.0	-0.0209	0.032	-0.659	0.510	-0.083	0.041
ExistingProdType_5.0	-0.0224	0.033	-0.676	0.499	-0.087	0.043
ExistingProdType_6.0	-0.0081	0.036	-0.224	0.823	-0.079	0.063

Omnibus:	37.224	Durbin-Watson:	1.993
Prob(Omnibus):	0.000	Jarque-Bera (JB):	24.250
Skew:	0.063	Prob(JB):	5.42e-06
Kurtosis:	2.590	Cond. No.	2.11e+03

-

Table 19: Summary of OLS model

Interpretation:

- R-squared from the above table is obtained as: 0.818
- This indicates that the linear regression model built using OLS was able to explain 82% of the variance in the data. In other words, accuracy is 82%
- Adjusted R-squared: The intention of this metric is like r-squared with the exception that the adjusted r-squared value gets adjusted with the multi-collinear independent variables.
- F-statistic is calculated using the formula: (SSR/DFssr)/(SSE/DFsse)
- The following represents the hypothesis test for the linear regression model:

- H0: $\beta 1 = \beta 2 = \beta 3 = 0$
- Ha: At least one of the coefficients is not equal to zero.
- The probability of f-statistic is 0 which is less than 0.05, with 95% confidence we can say reject the null hypothesis. That means, we can say that at least one of the coefficients is not equal to 0.

- Lower AIC, BIC scores indicate a better model. These scores indicate the loss in data while training the model.
- The coefficients of the linear regression equation obtained from the model can be interpreted as below:

```
(-1.237) * const + (0.009) * Complaint + (0.002) * CustCareScore + (0.059) * Age + (0.059) * CustTenure + (0.002) * NumberOfPolicy + (0.18) * MonthlyIncome + (0.032) * ExistingPolicyTenure + (0.566) * SumAssured + (-0.0) * LastMonthCalls + (0.009) * Channel_Online + (0.001) * Channel_Third Party Partner + (-0.08) * Occupation_Large Business + (-0.053) * Occupation_Salaried + (-0.077) * Occupation_Small Business + (0.003) * EducationField_Engineer + (-0.023) * EducationField_Graduate + (-0.034) * EducationField_MBA + (-0.019) * EducationField_Post Graduate + (-0.0) * EducationField_Under Graduate + (0.006) * Gender_Male + (-0.077) * Designation_Executive + (-0.08) * Designation_Manager + (-0.037) * Designation_Senior Manager + (-0.011) * Designation_VP + (-0.014) * MaritalStatus_Married + (0.005) * MaritalStatus_Single + (-0.028) * MaritalStatus_Unmarried + (0.011) * Zone_North + (0.049) * Zone_South + (0.015) * Zone_West + (-0.014) * PaymentMethod_Monthly + (-0.002) * PaymentMethod_Quarterly + (-0.008) * PaymentMethod_Yearly + (0.018) * ExistingProdType_2.0 + (-0.031) * ExistingProdType_3.0 + (-0.021) * ExistingProdType_4.0 + (-0.022) * ExistingProdType_5.0 + (-0.008) * ExistingProdType_6.0
```

- The values in the columns [0.025] and [0.975] indicate that the actual values of coefficients lie between the values in the respective columns with 95% confidence.

For example:

- The coefficient of **Complaint** will lie in the interval [-0.002, -0.021] with 95% confidence.

Interpreting significant coefficients using the model:

- The coefficients tell us how one unit change in X can affect y.
- The sign of the coefficient indicates if the relationship is positive or negative.

In this data set, for example

- An increase in 1 unit of Complaint is going to increase the value of AgentBonus by 0.0091.
- However, we observe that not all variables have very significant influence on the predictor variable.

- Value of condition number from the above summary is high which indicates that there are also signs of multi collinearity.
- Multicollinearity occurs when predictor variables in a regression model are correlated.
- This correlation is a problem because predictor variables should be independent. If the collinearity between variables is high, we might not be able to trust the p-values to identify independent variables that are statistically significant.
- When we have multicollinearity in the linear model, the coefficients that the model
- suggests are unreliable.

Interpreting significant coefficients:

- Null hypothesis: Predictor variable is not significant
- Alternate hypothesis: Predictor variable is significant

- (P > |t|) gives the p-value for each predictor variable to check the null hypothesis.
- If the level of significance is set to 5% (0.05), the p-values greater than 0.05 would indicate that the corresponding predictor variables are not significant.
- However, due to the presence of multicollinearity in our data, the p-values will also change.
- We need to ensure that there is no multicollinearity to interpret the p-values.

Determining if each variable is significant:

- H0:coefficient of Complaint = 0, probability of Complaint being 0.0091 is 0.117 – failed to reject null hypothesis – variable not significant.
- H0: CustCareScore = 0, probability of CustCareScore being 0.0023 is 0.235 - fail to reject null hypothesis - variable not significant.
- H0: Age = 0, probability of Age being 0.00591 is 0.0 - reject null hypothesis – variable significant.
- Significant variables from the model are:
    - Age, CustTenure, MonthlyIncome, ExistingPolicyTenure, SumAssured, Designation_Executive, Designation_Manager, Designation_SeniorManager, MaritalStatus_Married.

Model 5: Eliminating the highly multi-collinear independent features using VIF and applying Linear Regression

- A variance inflation factor (VIF) is a measure of the amount of multicollinearity in regression analysis. Multicollinearity exists when there is a correlation between multiple independent variables in a multiple regression model.
- Higher the variance inflation factor value, higher is the collinearity of the independent variable with the other independent variables.
- Variance inflation factor has been applied on the training and test data sets.
- Below are the variance inflation factor values in descending order in the training data.

	Columns	VIF
5	MonthlyIncome	2943.434352
7	SumAssured	1837.326376
12	Occupation_Salaried	667.979447
13	Occupation_Small Business	624.815092
11	Occupation_Large Business	137.401518
35	ExistingProdType_4.0	63.037918
34	ExistingProdType_3.0	42.976929
29	Zone_West	41.659957
27	Zone_North	31.736493
15	EducationField_Graduate	30.196378

Table 20: VIF values of independent attributes

- Removed the variables 'MonthlyIncome', 'SumAssured', 'Occupation_Salaried', 'Occupation_Small Business', 'Occupation_Large Business','ExistingProdType_4.0','ExistingProdType_3.0' as they are highly collinear with other variables and applied linear regression model on the remaining variables.
- Linear regression has been applied with default parameters.
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Below are the metrics of the linear regression model.

R-squared	Train	0.817409106142859
	Test	0.7943504106192938
RMSE	Train	0.1456933505624581
	Test	0.15391721161751243
MAE	Train	0.11848241607537766
	Test	0.1241405770171644
Equation	(0.009) * Complaint + (0.002) * CustCareScore + (0.06) * Age + (0.06) * CustTenure + (0.002) * NumberOfPolicy + (0.165) * MonthlyIncome + (0.033) * ExistingPolicyTenure + (0.559) * SumAssured + (-0.0) * LastMonthCalls + (0.009) * Channel_Online + (0.001) * Channel_Third Party Partner + (-0.016) * Occupation_Large Business + (0.005) * Occupation_Salaried + (-0.012) * Occupation_Small Business + (0.003) * EducationField_Engineer + (-0.016) * EducationField_Graduate + (-0.024) * EducationField_MBA + (-0.012) * EducationField_Post Graduate + (0.001) * EducationField_Under Graduate + (0.006) * Gender_Male + (-0.079) * Designation_Executive + (-0.08) * Designation_Manager + (-0.034) * Designation_Senior Manager + (-0.003) * Designation_VP + (-0.014) * MaritalStatus_Married + (0.005) * MaritalStatus_Single + (-0.028) * MaritalStatus_Unmarried + (0.007) * Zone_North + (0.022) * Zone_South + (0.01) * Zone_West + (-0.006) * PaymentMethod_Monthly + (0.002) * PaymentMethod_Quarterly + (-0.01) * PaymentMethod_Yearly + (0.02) * ExistingProdType_2.0 + (-0.023) * ExistingProdType_3.0 + (-0.011) * ExistingProdType_4.0 + (-0.01) * ExistingProdType_5.0 + (0.004) * ExistingProdType_6.0	
Intercept	[-1.06573344]	

Table 21: Metrics: Linear Regression on VIF filtered dataset

Interpretation:

- The model was able to explain only 54% and 53% of the training and test data respectively.
- The train and test variances are comparable. It indicates that the model has not been overfit.
- RMSE for the training set is around 0.1 and test set is around 0.12.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on.
- They are slightly higher than the values obtained with ensemble models.

- But the values are slightly higher than that of other ensemble regression models built.
- According to the equation, AgentBonus has positive coefficients for Complaint, CustCareScore, Age, NumberOfPolicy etc. which indicates that the target here is positively correlated with the above variables.
- AgentBonus has negative coefficients for independent attributes like Designation_Manager, Designation_Senio Manager, Designation_Executive, LastMonthCalls etc. which indicates that 'AgentBonus' is negatively correlated with these variables.

Model 6: Selecting the ideal features giving lowest AIC score using stepwise regression (forward selection and backward elimination)

- In this method, we apply both forward selection and backward elimination techniques to remove the least significant variables based on the AIC scores of the multiple models built using OrdinalLeastSquares method.
- The best set of features retrieving the least AIC score are chosen.
- This method suggests the optimal set of independent features to apply the linear regression model on.
- The suggested features are 'SumAssured', 'Age', 'CustTenure', 'MonthlyIncome', 'ExistingPolicyTenure', 'Designation_Executive', 'Designation_Manager', 'Designation_Senior Manager', 'MaritalStatus_Married', 'MaritalStatus_Unmarried', 'ExistingProdType_2.0', 'ExistingProdType_3.0', 'Complaint'
- Linear regression has been applied on the above features with default parameters.
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Below are the metrics of the linear regression model.

R-squared	Train	0.8168363162680612
	Test	0.7958150607056703
RMSE	Train	0.14592169253912177
	Test	0.15336812789638557
MAE	Train	0.11881382800809015

	Test	0.12370421925705345
Equation	(0.566) * SumAssured + (0.059) * Age + (0.058) * CustTenure + (0.178) * MonthlyIncome + (0.032) * ExistingPolicyTenure + (-0.073) * Designation_Executive + (-0.076) * Designation_Manager + (-0.034) * Designation_Senior Manager + (-0.017) * MaritalStatus_Married + (-0.033) * MaritalStatus_Unmarried + (0.03) * ExistingProdType_2.0 + (-0.009) * ExistingProdType_3.0 + (0.009) * Complaint	
Intercept	[-1.28650616]	

Table 22: Metrics: Stepwise regression

Interpretation:

- The model was able to explain only 81% and 79% of the training and test data respectively.
- The train and test variances are comparable. It indicates that the model has not been overfit.
- RMSE for the training set is around 0.14 and test set is around 0.15.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on.
- They are slightly higher than the values obtained with ensemble models.
- But the values are slightly higher than that of other ensemble regression models built.
- According to the equation, AgentBonus has positive coefficients for SumAssured, Age, CustTenure, MonthlyIncome, ExistingPolicyTenure etc. which indicates that the target here is positively correlated with the above variables.
- AgentBonus has negative coefficients for independent attributes like Designation_Manager, Designation_Senior Manager, Designation_Executive, etc. which indicates that 'AgentBonus' is negatively correlated with these variables.

2. Model Tuning

a. Ensemble models

Model 7: Bagging Regressor with Support Vector Regression and Linear Regression

- Bagging ensembling technique builds several weak models given the dataset and accumulates the results of each model to get the best and final prediction for the target variable
- In this model, bagging technique is applied to the log transformed dataset through BaggingRegressor.
- Base estimators considered to build the several models on are SupportVectorRegressor and LinearRegression models.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- GridSearchCV is applied to the Bagging Regressor algorithm with the below parameters
    - Base estimators: Support Vector Regression model, LinearRegressionModel
    - N_estimators: [40, 50, 60, 70, 80, 90, 100] (number of models of base estimator to build)
    - Number of cross validations performed: 5
- Applied Bagging Regressor algorithm with the best parameters obtained from the above grid search algorithm.
    - Base_estimator: LinearRegression()
    - N_estimators: 100
    - Random_state: 1
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the bagging regressor model:

R-squared	Train	0.817507595043120
	Test	0.7942960630248834
RMSE	Train	0.14565405201622356
	Test	0.15393754834083262
MAE	Train	0.11852662360010748
	Test	0.1241222365770391

Table 23: Metrics: Bagging regressor with base LinearRegressor

Interpretation:

- The model was able to explain 81% and 79% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.14 and test set is around 0.15.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original dataset but like the plain linear regression model applied on log transformed encoded data set.

Model 8: Random Forest Regressor

- Bagging ensembling technique builds several weak models given the dataset and accumulates the results of each model to get the best and final prediction for the target variable
- In this model, bagging technique is applied to the log transformed dataset through RandomForest where multiple decision trees are formed with specified criterion.
- The average of the predictions of all these individual models is considered the final predi tion of the target variable.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- GridSearchCV is applied to the Random Forest Regressor algorithm with the below parameters
    - Criterion: squared_error, absolute_error
    - N_estimators: [40, 50, 60, 70, 80, 90, 100] (number of models of base estimator to build)
    - Max_depth: [10,15,20]
    - Number of cross validations performed: 5
- Applied Bagging Regressor algorithm with the best parameters obtained from the above grid search algorithm.
    - Base_estimator: absolute_error
    - N_estimators: 90
    - Max_depth: 20 (maximum tree depth of each random forest tree constructed)
    - Random_state: 1
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the bagging regressor model:

R-squared	Train	0.978280814769586
	Test	0.8573722947103191
RMSE	Train	0.05024836537105073
	Test	0.12818145918646076
MAE	Train	0.03782774737840577
	Test	0.09837814616626694

Table 24: Metrics: Random Forest Regressor (Absolute error criterion)

Interpretation:

- The model was able to explain 97% and 85% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.05 and test set is around 0.12.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original dataset but like the plain linear regression model applied on log transformed encoded dataset.

Model 9: Random Forest Regressor with squared error criterion

- Bagging ensembling technique builds several weak models given the dataset and accumulates the results of each model to get the best and final prediction for the target variable
- In this model, bagging technique is applied to the log transformed dataset through RandomForest where multiple decision trees are formed with specified criterion.
- The average of the predictions of all these individual models is considered the final predi tion of the target variable.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- Applied Bagging Regressor algorithm with the below parameters
    - Base_estimator: squared_error
    - N_estimators: 90
    - Max_depth: 20 (maximum tree depth of each random forest tree constructed)
    - Random_state: 1
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the bagging regressor model:

R-squared	Train	0.980819969464432
	Test	0.8581665209337239
RMSE	Train	0.04721987366785347
	Test	0.12782407000287704
MAE	Train	0.0362255114447576
	Test	0.09852115671473491

Table 25: Metrics: Random Forest Regressor (Squared error criterion)

Interpretation:

- The model was able to explain 98% and 85% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.04 and test set is around 0.12.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original dataset but like the plain linear regression model applied on log transformed encoded dataset.

Model 10: Gradient Boosting Regressor

- Boosting is one kind of ensemble Learning method which trains the model sequentially and each new model tries to correct the previous model. It combines several weak learners into strong learners.
- Gradient Boosting is a powerful boosting algorithm that combines several weak learners into strong learners, in which each new model is trained to minimize the loss function such as mean squared error or cross-entropy of the previous model using gradient descent.
- In each iteration, the algorithm computes the gradient of the loss function with respect to the predictions of the current ensemble and then trains a new weak model to minimize this gradient.
- The predictions of the new model are then added to the ensemble, and the process is repeated until a stopping criterion is met.
- The average of the predictions of all these individual models is considered the final predi tion of the target variable.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- GridSearchCV is applied to the Random Forest Regressor algorithm with the below parameters
    - N_estimators: [50, 60, 70, 80, 90, 100] (number of models of base estimator to build)
    - Learning_rate: [0.0001, 0.001, 0.01, 0.1, 1.0] (the contribution of each tree to this sum can be weighted to slow down the learning by the algorithm. This weighting is called a shrinkage or a learning rate)
    - Number of cross validations performed: 5
    - Subsample: [0.5, 0.7, 1.0] (The fraction of samples to be used for fitting the individual base learners)
    - Max_depth: [3, 7, 9] (Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree)
- Most important features obtained from the gridsearch algorithm are:

	Importance
SumAssured	0.771653
Age	0.065828
CustTenure	0.059351
MonthlyIncome	0.053899
ExistingPolicyTenure	0.013418
LastMonthCalls	0.007583
NumberOfPolicy	0.003527
CustCareScore	0.003520
Designation_Manager	0.001327
ExistingProdType_4.0	0.001288
MaritalStatus_Married	0.001174
Channel_Third Party Partner	0.001069
Gender_Male	0.001016
Designation_Senior Manager	0.001015
Channel_Online	0.000998
PaymentMethod_Yearly	0.000982
Complaint	0.000872
ExistingProdType_3.0	0.000861
Designation_Executive	0.000838
EducationField_Under Graduate	0.000795

Table 26: Important features: Gradient Boosting regressor

- Applied Gradient Boosting Regressor algorithm on important columns from above table with the best parameters obtained from GridSearchCV algorithm
    - Max_depth: 7
    - N_estimators: 50
    - Subsample: 0.7
    - Random_state: 1
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the gradient boosting regressor model:

R-squared	Train	0.9423341268202841
	Test	0.8581370294947515
RMSE	Train	0.08187656419367166
	Test	0.12783735854323297
MAE	Train	0.06198698873567322
	Test	0.09899789010557909

Table 27: Metrics: Gradient Boosting regressor

Interpretation:

- The model was able to explain 94% and 85% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.08 and test set is around 0.12.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original dataset but like the plain linear regression model applied on log transformed encoded dataset.

Model 11: XGBoost Regressor

- XGBoost dominates structured or tabular datasets on classification and regression predictive modeling problems.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- GridSearchCV is applied to the Random Forest Regressor algorithm with the below parameters
    - Objective: reg:squarederror (regression with squared loss)
    - N_estimators: [40, 50, 60] (number of models of base estimator to build)
    - Learning_rate: [0.0001, 0.001, 0.01, 0.1, 1.0] (the contribution of each tree to this sum can be weighted to slow down the learning by the algorithm. This weighting is called a shrinkage or a learning rate)
    - Number of cross validations performed: 5
    - Subsample: [0.5, 0.7, 1.0] (The fraction of samples to be used for fitting the individual base learners)
    - Max_depth: [3, 7, 9] (Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree)
- Most important features obtained from the gridsearch algorithm are:

	Importance
SumAssured	0.593450
Age	0.046299
MonthlyIncome	0.038339
CustTenure	0.037782
PaymentMethod_Quarterly	0.021693
ExistingPolicyTenure	0.019616
EducationField_Engineer	0.019202
Zone_West	0.013034
Designation_Senior Manager	0.011617
ExistingProdType_4.0	0.010716
Designation_Manager	0.010606
ExistingProdType_5.0	0.010355
EducationField_Graduate	0.010090
LastMonthCalls	0.009850
Channel_Online	0.009814
Designation_Executive	0.009292
EducationField_Under Graduate	0.009291
ExistingProdType_2.0	0.008270
Occupation_Salaried	0.008135
Occupation_Small Business	0.008033

Table 28: Important features: XGBoost Regressor

- Applied XGBoost Regressor algorithm on important columns from above table with the best parameters obtained from GridSearchCV algorithm
    - Objective: reg:squarederror
    - Max_depth: 7
    - n_estimators: 60
    - subsample: 1
    - learning_rate: 0.1
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the XGBoost regressor model:

R-squared	Train	0.940861907955296
	Test	0.8636691811111801
RMSE	Train	0.08291513804989126
	Test	0.12531997090190905
MAE	Train	0.06015950557997443
	Test	0.09734143222231202

Table 29:Metrics: XGBoost Regressor

Interpretation:

- The model was able to explain 94% and 86% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.08 and test set is around 0.12.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original dataset but like the plain linear regression model applied on log transformed encoded dataset.

Model 12: XGBoost Random Forest Regressor

- XGBoost dominates structured or tabular datasets on classification and regression predictive modeling problems.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- GridSearchCV is applied to the XGBoost Random Forest Regressor algorithm with the below parameters
    - Objective: reg:squarederror (regression with squared loss)
    - N_estimators: [40, 50, 60] (number of models of base estimator to build)
    - Learning_rate: [0.0001, 0.001, 0.01, 0.1, 1.0] (the contribution of each tree to this sum can be weighted to slow down the learning by the algorithm. This weighting is called a shrinkage or a learning rate)
    - Number of cross validations performed: 5
    - Subsample: [0.5, 0.7, 1.0] (The fraction of samples to be used for fitting the individual base learners)
    - Max_depth: [3, 7, 9] (Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree)
- Most important features obtained from the grid search algorithm are:

	Importance
SumAssured	0.562275
MonthlyIncome	0.044615
CustTenure	0.039583
Age	0.038359
Designation_VP	0.021759
ExistingPolicyTenure	0.018949
Designation_Executive	0.015487
ExistingProdType_2.0	0.013186
Designation_Senior Manager	0.013179
Zone_West	0.011947
EducationField_Engineer	0.011585
MaritalStatus_Unmarried	0.011130
EducationField_MBA	0.010988
ExistingProdType_4.0	0.010075
Designation_Manager	0.010036
LastMonthCalls	0.009981
EducationField_Under Graduate	0.009940
ExistingProdType_5.0	0.009896
Channel_Third Party Partner	0.009542
EducationField_Post Graduate	0.009533

Table 30:Important Features: XGBoost RandomForest Regressor

- Applied XGBoost Random Forest Regressor algorithm on important columns from above table with the best parameters obtained from GridSearchCV algorithm
    - Objective: reg:squarederror
    - Max_depth: 9
    - n_estimators: 60
    - subsample: 0.5
    - learning_rate: 1.0
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the XGBoost Random Forest regressor model:

R-squared	Train	0.9084511820798093
	Test	0.8583323158699999
RMSE	Train	0.10316361314802312
	Test	0.12774933877396918
MAE	Train	0.07841852580698634
	Test	0.10020733078965871

Table 31:Metrics: XGBoost RandomForest Regressor

Interpretation:

- The model was able to explain 90% and 85% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.1 and test set is around 0.12.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original dataset but like the plain linear regression model applied on log transformed encoded dataset.
- But the values are slightly higher than that of other ensemble regression models built.

b. Other Model tuning measures

Need of feature engineering:

- All the features in the dataset are individually explaining their relationship with target variable and among themselves are not suitable to be feature engineered.
- There is no scope observed to add new features or modify/accumulate existing features.
- Hence, other variations of linear regression models are built with various hyper parameter tuning techniques.

Model 13: Ridge Regression algorithm with GridSearch cross validation hyper parameter tuning technique

- Log transformed encoded dataset is considered for Ridge Regression model.
- GridSearchCV performs exhaustive search over specified parameter values for an estimator.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- GridSearchCV is applied to the ridge algorithm with the below parameters
    - Alpha: [0.1, 0.3, 0.9,2,5,10] (alpha is the parameter that balances the amount of emphasis given to minimizing RSS vs minimizing the sum of squares of coefficients.)
    - Random state: 1
- Best parameter obtained from GridSearchCV is 5 with a score of 81.2%
- Applied ridge regression with below parameters
    - alpha = 5
    - random_state = 1
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the ridge regression model:

R-squared	Train	0.817409106142859
	Test	0.7943504106192938
RMSE	Train	0.1456933505624581
	Test	0.15391721161751243
MAE	Train	0.11848241607537766
	Test	0.1241405770171644
Equation	(0.009) * Complaint + (0.002) * CustCareScore + (0.06) * Age + (0.06) * CustTenure + (0.002) * NumberOfPolicy + (0.165) * MonthlyIncome + (0.033) * ExistingPolicyTenure + (0.559) * SumAssured + (-0.0) * LastMonthCalls + (0.009) * Channel_Online + (0.001) * Channel_Third Party Partner + (-0.016) * Occupation_Large Business + (0.005) * Occupation_Salaried + (-0.012) * Occupation_Small Business + (0.003) * EducationField_Engineer + (-0.016) * EducationField_Graduate + (-0.024) * EducationField_MBA + (-0.012) * EducationField_Post Graduate + (0.001) * EducationField_Under Graduate + (0.006) * Gender_Male + (-0.079) * Designation_Executive + (-0.08) * Designation_Manager + (-0.034) * Designation_Senior Manager + (-0.003) * Designation_VP + (-0.014) * MaritalStatus_Married + (0.005) * MaritalStatus_Single + (-0.028) * MaritalStatus_Unmarried + (0.007) * Zone_North + (0.022) * Zone_South + (0.01) * Zone_West + (-0.006) * PaymentMethod_Monthly + (0.002) * PaymentMethod_Quarterly + (-0.01) * PaymentMethod_Yearly + (0.02) * ExistingProdType_2.0 + (-0.023) * ExistingProdType_3.0 + (-0.011) * ExistingProdType_4.0 + (-0.01) * ExistingProdType_5.0 + (0.004) * ExistingProdType_6.0	
Intercept	[-1.06573344]	

Table 32: Metrics: Ridge Regression model with GridSearchCV

Interpretation:

- The model was able to explain 81% and 79% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.14 and test set is around 0.15.
- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original

dataset but like the plain linear regression model applied on log transformed encoded dataset.

- From the equation built, agent bonus increases when values for the attributes like Complaint, CustCareScore, SumAssured increases.
- AgentBonus decreases when the values for the attributes like Occupation_LargeBusiness, Occupation_SmallBusiness, EducationField_Graduate, Designation_Manager, ExistingProdType_3.0, ExistingProdType_4.0.
- AgentBonus does not seem to affect with the change in LastMonthCalls.

Model 14: Lasso Regression algorithm with RandomizedSearch cross validation hyper parameter tuning technique

- Log transformed encoded dataset is considered for Lasso Regression model.
- In RandomizedSearchCV, a fixed number of parameter settings is sampled from the specified distributions. The number of parameter settings that are tried is given by n_iter.

Steps performed:

- The log transformed and encoded data has been split into training and test datasets with 70 : 30 ratio.
- GridSearchCV is applied to the lasso algorithm with the below parameters
    - Alpha: [0.0001, 0.001, 0.05, 0.1] (alpha is the parameter that balances the amount of emphasis given to minimizing RSS vs minimizing the sum of squares of coefficients.)
    - Random state: 1
- Best parameter obtained from GridSearchCV is 0.0001 with a score of 81.3%
- Applied lasso regression with below parameters
    - alpha = 0.0001
    - random_state = 1
- The model has been fit and trained with the training dataset.
- The obtained model is applied against the test dataset.

Metrics of the ridge regression model:

R-squared	Train	0.817363896022623
	Test	0.7948135890638899
RMSE	Train	0.14571138652821367
	Test	0.15374378233452216
MAE	Train	0.11864870418546235
	Test	0.12410862748802384
Equation	(0.009) * Complaint + (0.002) * CustCareScore + (0.059) * Age + (0.058) * CustTenure + (0.002) * NumberOfPolicy + (0.175) * MonthlyIncome + (0.032) * ExistingPolicyTenure + (0.567) * SumAssured + (-0.0) * LastMonthCalls + (0.007) * Channel_Online + (0.0) * Channel_Third Party Partner + (-0.0) * Occupation_Large Business + (0.006) * Occupation_Salaried + (-0.0) * Occupation_Small Business + (0.0) * EducationField_Engineer + (-0.004) * EducationField_Graduate + (-0.009) * EducationField_MBA + (-0.0) * EducationField_Post Graduate + (0.001) * EducationField_Under Graduate + (0.005) * Gender_Male + (-0.072) * Designation_Executive + (-0.075) * Designation_Manager + (-0.031) * Designation_Senior Manager + (-0.002) * Designation_VP + (-0.014) * MaritalStatus_Married + (0.005) * MaritalStatus_Single + (-0.026) * MaritalStatus_Unmarried + (0.0) * Zone_North + (0.0) * Zone_South + (0.003) * Zone_West + (0.0) * PaymentMethod_Monthly + (0.0) * PaymentMethod_Quarterly + (-0.007) * PaymentMethod_Yearly + (0.02) * ExistingProdType_2.0 + (-0.016) * ExistingProdType_3.0 + (-0.006) * ExistingProdType_4.0 + (-0.007) * ExistingProdType_5.0 + (0.002) * ExistingProdType_6.0	
Intercept	[-1.27623283]	

Table 33: Metrics: Lasso Regression model with RandomizedSearchCV

Interpretation:

- The model was able to explain 81% and 79% of the variances in the training and test datasets respectively.
- The train and test variances are comparable. It indicates that the model has not overfit.
- RMSE for the training set is around 0.14 and test set is around 0.15.

- The RMSE values and MAE values are low which indicates that the data well suited for applying linear regression on. They are better than the values obtained with the original dataset but like the plain linear regression model applied on log transformed encoded dataset.
- From the equation built, agent bonus increases when values for the attributes like Complaint, CustCareScore, SumAssured increases.
- AgentBonus decreases when the values for the attributes like Occupation_LargeBusiness, Occupation_SmallBusiness, EducationField_Graduate, Designation_Manager, ExistingProdType_3.0, ExistingProdType_4.0.
- AgentBonus does not seem to affect with the change in LastMonthCalls, Occupation_LargeBusiness etc.

c. Interpretation of the most optimum model and its implication on the business

Summarizing the scores and metrics of various regression models built.

Model	R-squared		Adjusted R-squared		Root Mean Squared Error		Mean Absolute Error	
	Train	Test	Train	Test	Train	Test	Train	Test
Linear Regression on Original encoded dataset	0.825	0.811	0.823	0.805	587.29	607.972	461.207	480.066
Linear Regression on scaled encoded dataset	0.82	0.80	0.818	0.798	0.313	0.327	0.247	0.258
Linear Regression on log-transformed encoded dataset	0.818	0.794	0.815	0.788	0.146	0.154	0.118	0.124
OLS	0.818	-	0.815	-	0.438	0.431	0.369	0.364
Linear Regression on VIF filtered attributes	0.546	0.534	0.541	0.523	0.23	0.23	0.186	0.187

Linear Regression using Stepwise regression involving Backward elimination and forward selection	0.816	0.796	0.816	0.784	0.146	0.153	0.119	0.124
Bagging Regressor with Base Linear Regressor	0.818	0.794	0.815	0.788	0.146	0.154	0.119	0.124
Random Forest Regressor with absolute error criterion	0.978	0.857	0.978	0.853	0.05	0.128	0.04	0.1
Random Forest Regressor with squared error criterion	0.981	0.859	0.98	0.854	0.047	0.128	0.036	0.099
Gradient Boosting Regressor	0.942	0.858	0.942	0.857	0.081	0.128	0.062	0.099
XGBoost Regressor	0.941	0.864	0.941	0.863	0.083	0.125	0.06	0.097
XGBoost Random Forest Regressor	0.908	0.858	0.91	0.857	0.103	0.128	0.784	0.100
Ridge Regression with Grid Search CV	0.817	0.794	0.815	0.788	0.146	0.154	0.118	0.124
Lasso Regression with Randomized Search CV	0.817	0.795	0.815	0.788	0.145	0.154	0.119	0.124

Table 34: Metrics summary of all models

Most optimum model:

- The most optimum model for a linear regression is chosen based on the R-squared and RMSE criteria.
- Higher the R-squared and adjusted R-squared values, higher is the implication and usability of the model.

- Low Root Mean Square Error values indicate the model has accurately predicted the value of the data as is close to the actual test data.
- From the metrics of the above models, it is evident that 3 models that are highlighted have given the best performance in terms of high R-squared values and low RMSE values.
- The next best model that has low RMSE value is the XGBoost Regressor model. This has shown best measure in all metrics of Linear Regression.
- It has covered most of the data by showing high R-squared value and has low RMSE and MAE error terms.
- Random Forest regressor with absolute error criterion and squared error criterion have also given the best RMSE score of all the models but the difference between train and test RMSE scores is quite significant.
- It indicates that the model has been overfit.
- Parameters of most optimum XGBoostRegressor model:

- {'objective': 'reg:squarederror', 'base_score': None, 'booster': None, 'callbacks': None, 'colsample_bylevel': None, 'colsample_bynode': None, 'colsample_bytree': None, 'device': None, 'early_stopping_rounds': None, 'enable_categorical': False, 'eval_metric': None, 'feature_types': None, 'gamma': None, 'grow_policy': None, 'importance_type': None, 'interaction_constraints': None, 'learning_rate': 0.1, 'max_bin': None, 'max_cat_threshold': None, 'max_cat_to_onehot': None, 'max_delta_step': None, 'max_depth': 7, 'max_leaves': None, 'min_child_weight': None, 'missing': nan, 'monotone_constraints': None, 'multi_strategy': None, 'n_estimators': 60, 'n_jobs': None, 'num_parallel_tree': None, 'random_state': 1, 'reg_alpha': None, 'reg_lambda': None, 'sampling_method': None, 'scale_pos_weight': None, 'subsample': 1.0, 'tree_method': None, 'validate_parameters': None, 'verbosity': None}

- Metrics of the most optimum model:

  - R-squared on training data set: 0.940861907955296
  - R-squared on test data set: 0.8636691811111801
  -
  - RMSE on training data set: 0.08291513804989126
  - RMSE on test data set: 0.12531997090190905
  -
  - MAE on training data set: 0.06015950557997443
  - MAE on test data set: 0.09734143222231202
  -
  - Adjusted R-squared on training data: 0.9406178459881274
  - Adjusted R-squared on test data: 0.8623485397955656

Checking if the assumptions of linear regression model are satisfied by the chosen most optimum model XGBoost Regressor:

Assumption 1: The data needs to be from a linear relationship – For this to be satisfied, in a plot for the predicted values and residuals, there should not be any pattern.
Assumption 2: Independence of residuals - For this to be satisfied, in a plot for the predicted values and residuals, there should not be any pattern.

As seen in the below graph, there is no visible pattern and is spread randomly. This indicates that the above assumptions are satisfied. The errors are not predictable.

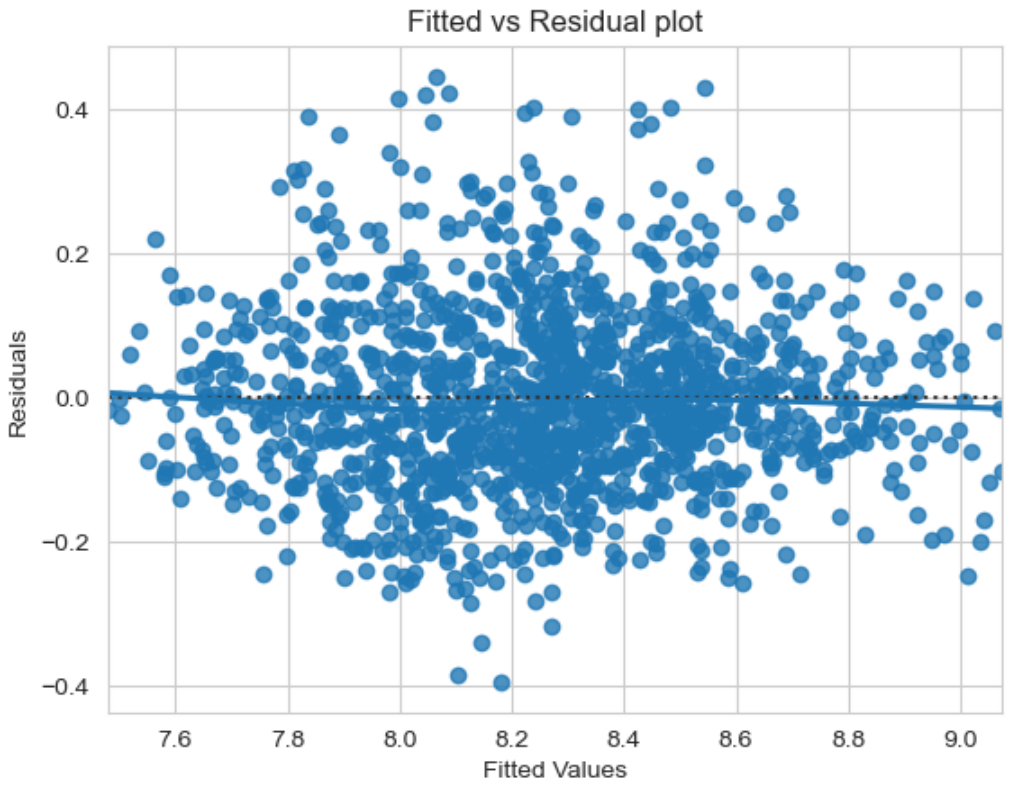


Figure 22: Independence assumption check of residuals

Assumption 3: Normality of residuals – This indicates that the residuals are random and are not belonging to a random sample. For this assumption to be satisfied, the histogram plot of the residuals should show a normal distribution

Below is the residual histogram plot for XGBoost Regressor model. As visible, the third assumption is also satisfied

Alternatively, we can plot a Quantile-quantile plot which shows the line for a perfectly normal distribution with the curve obtained by plotting the normality of residuals.

If the curved line is most inclined to the straight line, then the residuals are said to be most normally distributed.

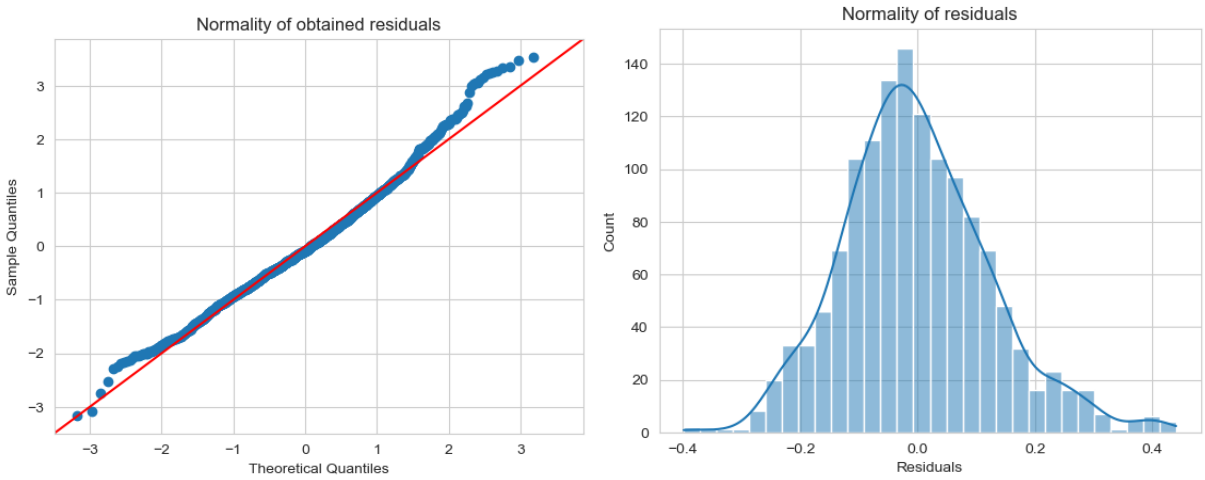


Figure 23: Q-Q plot and Histogram of residuals

With the above checks it is evident that the XGBoost Regressor is the most optimum model and most suitable for achieving the objective of the problem statement.

Implications of business by utilizing the most optimum model:

- According to the most optimum model, below are the most important features that can provide more value in predicting the AgentBonus.

	Importance
SumAssured	0.593450
Age	0.046299
MonthlyIncome	0.038339
CustTenure	0.037782
PaymentMethod_Quarterly	0.021693
ExistingPolicyTenure	0.019616
EducationField_Engineer	0.019202
Zone_West	0.013034
Designation_Senior Manager	0.011617
ExistingProdType_4.0	0.010716
Designation_Manager	0.010606
ExistingProdType_5.0	0.010355
EducationField_Graduate	0.010090
LastMonthCalls	0.009850
Channel_Online	0.009814
Designation_Executive	0.009292
EducationField_Under Graduate	0.009291
ExistingProdType_2.0	0.008270
Occupation_Salaried	0.008135
Occupation_Small Business	0.008033

Table 35: Most important features chosen by XGBoost Regressor Model.

- As suggested by most of the models and the current model, Age, Sum Assured and Monthly Income are the most important variables in predicting the Agent and they have a positive correlation with Agent Bonus.
- If agent can acquire customers who opt for insurance policies having high Sum Assured amounts, Agent will be having high Bonus.

- Indirectly, people with high Age would pay high premiums which provide good revenue to the company.
- The same applies to people having High Income. They wish to secure their health and other assets with the security provided by insurance policies.
- Applying linear regression with the most important features chosen by the model we get the equation –

```
(0.57) * SumAssured + (0.061) * Age + (0.06) * CustTenure + (0.273)
* MonthlyIncome + (0.033) * ExistingPolicyTenure + (0.013) * Payment
Method_Quarterly + (0.008) * Designation_Senior Manager + (0.005) *
Zone_West + (-0.022) * Designation_Manager + (-0.001) * EducationFie
ld_Engineer + (-0.004) * ExistingProdType_4.0 + (0.001) * EducationF
ield_Graduate + (-0.012) * ExistingProdType_5.0
```

- This equation implies that with agents securing customers for high policy tenures tend to get high bonus.
- On the other hand, agents who sell insurance policies to customers who are Managers, having an education background of Engineer and those who chose product types receive less bonuses.
- To summarize, high performing agents are the ones who acquire customers with high Sum Assured amounts, such agents can be rewarded with bonus or can be given training related to other product types if the company chooses to increase revenue generated by a particular product type.
- Also, the company identify agents who bring customers with low Sum Assured amounts and train them to effectively communicated with aged customers and high income customers in order to be able to sell the insurance policies as they expect.

---------- END OF REPORT ----------