Business Report

Yuthika Khedwal

Content :

1) Introduction of the business problem

a) Problem statement

The major objective of this data set is to extract actionable insights from the leading life insurance company data and make strategic changes to make the company grow. Primary objective is to create Machine Learning models which correctly predicts the bonus for its agents so that it may provide information regarding high performing agents and low performing agents. Once a model is developed then it can extract actionable insights and recommendation, so based of which the company may design appropriate engagement activity and up skill programs for their agents as required.

b) Need of the study/project

Based on their agents to sell the policies, the insurance companies are heavily dependent on their success. So, it becomes very crucial to find and design engagement activity for their high performing agents giving them more and more incentives to keep up their performance and achieve more and also, up skill programs for their low performing agents to get better and perform better, and such that all together their agents are more able to sell the quality insurance to their customers and add more greater value to the company. And through this project with the help of data and its analysis help the insurance company to make data-driven business decisions. It empowers companies with high-level data and information that is leveraged into improved insurance processes and new opportunities.

c) Understanding business / social opportunity

Usually businesses benefit to the extent that they stay close to customers. Traditionally, the insurance company has relied on strong networking and trusted relationships. By transforming into social businesses, insurers can tap significant opportunities that enable them to generate more demand, win customer loyalty and maximize returns.

2) Data Report

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

The data belongs to a leading life insurance company. The agent's different sales data based on the customers' varied attributes like age, tenure in organization, channel through which acquisition is done, their occupation, education, Designation Marital status, Gender, their location, complaint registered, income, customer satisfaction score, all collected in the course of time they were with the company. Certain attributes leading to the Agent's bonus are also captured.

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

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

```
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)
```

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
CustID	4520.00	NaN	NaN	NaN	7002259.50	1304.96	7000000.00	7001129.75	7002259.50	7003389.25	7004519.00
AgentBonus	4520.00	NaN	NaN	NaN	4077.84	1403.32	1605.00	3027.75	3911.50	4867.25	9608.00
Age	4251.00	NaN	NaN	NaN	14.49	9.04	2.00	7.00	13.00	20.00	58.00
CustTenure	4294.00	NaN	NaN	NaN	14.47	8.96	2.00	7.00	13.00	20.00	57.00
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	4520.00	NaN	NaN	NaN	3.69	1.02	1.00	3.00	4.00	4.00	6.00
Designation	4520	6	Manager	1620	NaN	NaN	NaN	NaN	NaN	NaN	NaN
NumberOfPolicy	4475.00	NaN	NaN	NaN	3.57	1.46	1.00	2.00	4.00	5.00	6.00
MaritalStatus	4520	4	Married	2268	NaN	NaN	NaN	NaN	NaN	NaN	NaN
MonthlyIncome	4284.00	NaN	NaN	NaN	22890.31	4885.60	16009.00	19683.50	21606.00	24725.00	38456.00
Complaint	4520.00	NaN	NaN	NaN	0.29	0.45	0.00	0.00	0.00	1.00	1.00
ExistingPolicyTenure	4336.00	NaN	NaN	NaN	4.13	3.35	1.00	2.00	3.00	6.00	25.00
SumAssured	4366.00	NaN	NaN	NaN	619999.70	246234.82	168536.00	439443.25	578976.50	758236.00	1838496.00
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	4520.00	NaN	NaN	NaN	4.63	3.62	0.00	2.00	3.00	8.00	18.00
CustCareScore	4468.00	NaN	NaN	NaN	3.07	1.38	1.00	2.00	3.00	4.00	5.00

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

The name of the columns seems to be fine with no special characters or spaces between them.

Unique values of various Categories :

Channel : 3

Online 468
Third Party Partner 858
Agent 3194
Name: Channel, dtype: int64

Occupation : 5
Free Lancer 2
Laarge Business 153
Large Business 255
Small Business 1918
Salaried 2192
Name: Occupation, dtype: int64

EducationField : 7
MBA 74
UG 230
Post Graduate 252
Engineer 408
Diploma 496
Under Graduate 1190
Graduate 1870
Name: EducationField, dtype: int64

Gender : 3
Fe male 325
Female 1507
Male 2688
Name: Gender, dtype: int64

Designation : 6
Exe 127
VP 226
AVP 336
Senior Manager 676
Executive 1535
Manager 1620
Name: Designation, dtype: int64

MaritalStatus : 4
Unmarried 194
Divorced 804

Single 1254
Married 2268
Name: MaritalStatus, dtype: int64

Zone : 4
South 6
East 64
North 1884
West 2566
Name: Zone, dtype: int64

PaymentMethod : 4
Quarterly 76
Monthly 354
Yearly 1434
Half Yearly 2656
Name: PaymentMethod, dtype: int64

The highlighted data seems to be recorded incorrectly and required replacement and this was done to ensure the right categories are picked up by the model

```
df['Occupation']=df['Occupation'].replace(to_replace='Laarge Business',value='Large Business')
```

```
df['Gender']=df['Gender'].replace(to_replace='Fe male',value='Female')
```

```
df['Designation']=df['Designation'].replace(to_replace='Exe',value='Executive')
```

```
df['EducationField']=df['EducationField'].replace(to_replace='UG',value='Under Graduate')
```

```
df['MaritalStatus']=df['MaritalStatus'].replace(to_replace='Unmarried',value='Single')
```

Post fixing of the data :

Channel : 3
Online 468
Third Party Partner 858
Agent 3194
Name: Channel, dtype: int64

Occupation : 4

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

EducationField : 6
MBA 74
Post Graduate 252
Engineer 408
Diploma 496
Under Graduate 1420
Graduate 1870
Name: EducationField, dtype: int64

Gender : 2
Female 1832
Male 2688
Name: Gender, dtype: int64

Designation : 5
VP 226
AVP 336
Senior Manager 676
Manager 1620
Executive 1662
Name: Designation, dtype: int64

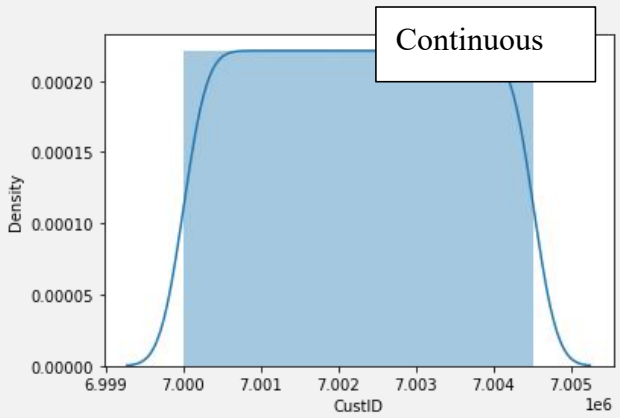
MaritalStatus : 3
Divorced 804
Single 1448
Married 2268
Name: MaritalStatus, dtype: int64

Zone : 4
South 6
East 64
North 1884
West 2566
Name: Zone, dtype: int64

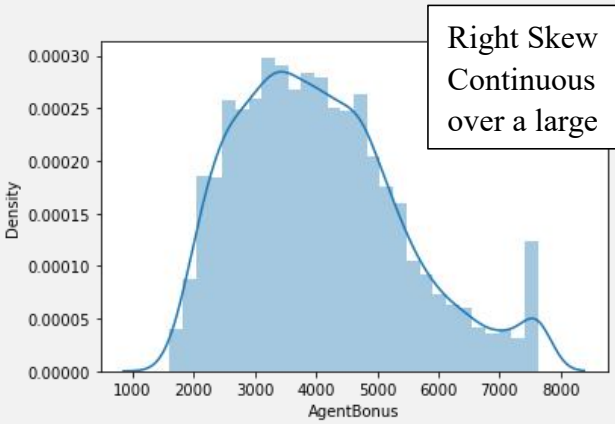
PaymentMethod : 4
Quarterly 76
Monthly 354
Yearly 1434
Half Yearly 2656
Name: PaymentMethod, dtype: int64

3) Exploratory data analysis

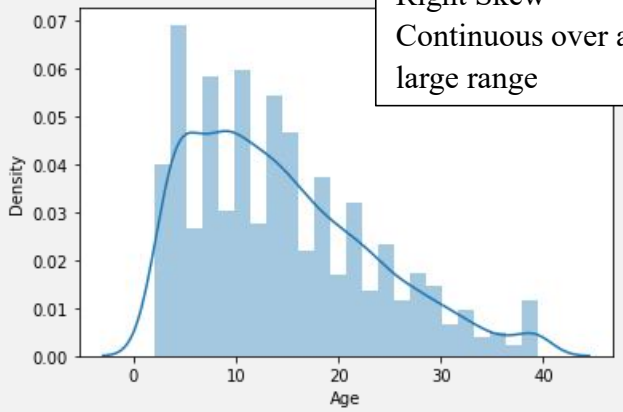
a) Univariate analysis (distribution and spread for every continuous attribute, distribution of data in categories for categorical ones)



Continuous

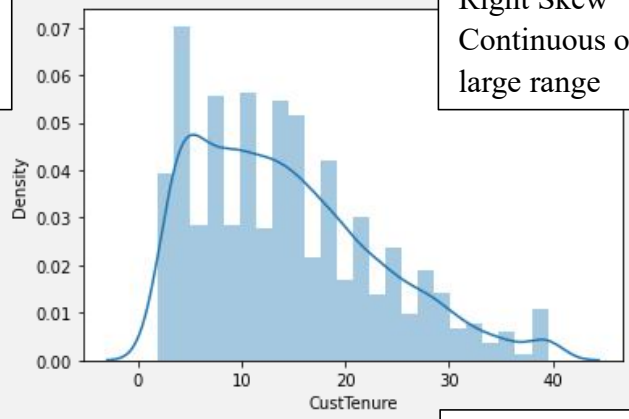


Right Skew Continuous over a large



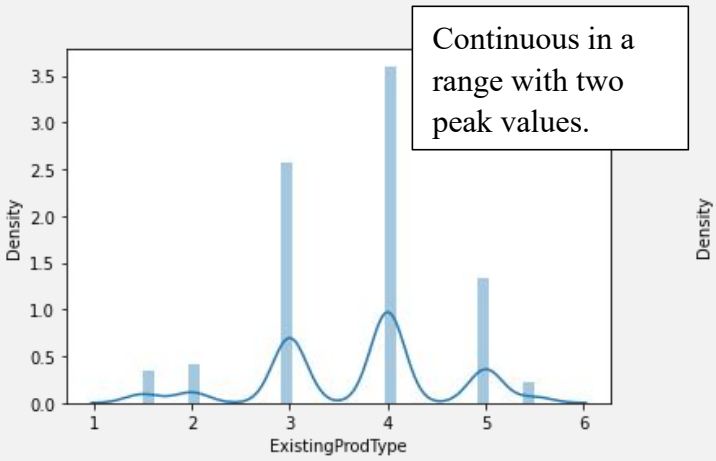
Right Skew Continuous over a large range

Continuous Right Skewed

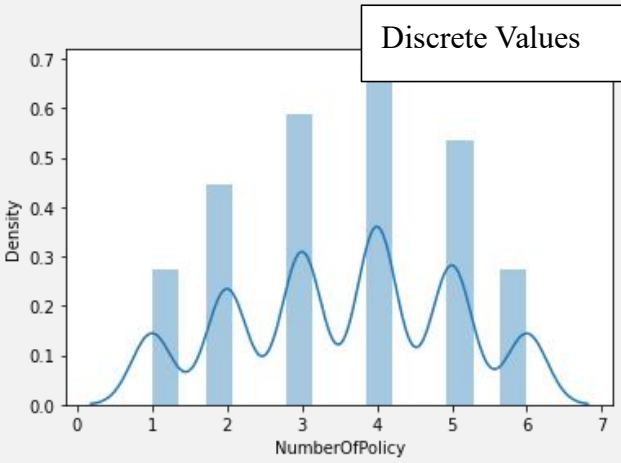


Right Skew Continuous over a large range

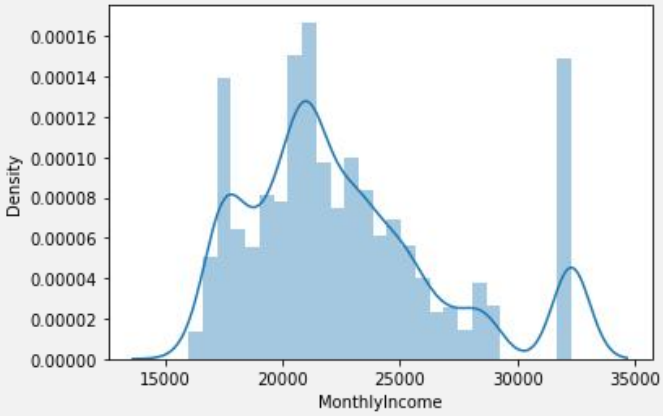
More Discrete Kind of data 4 is the most frequent observation

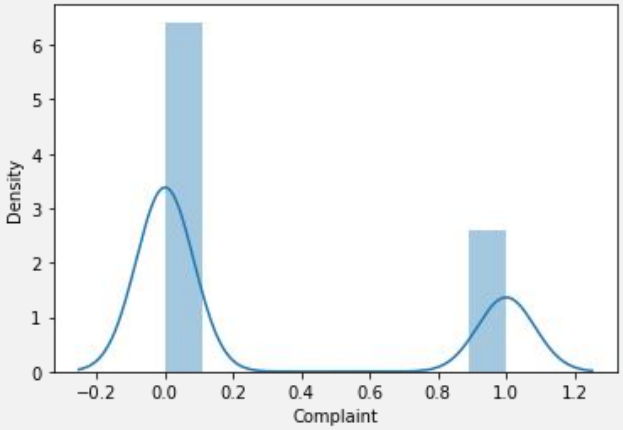


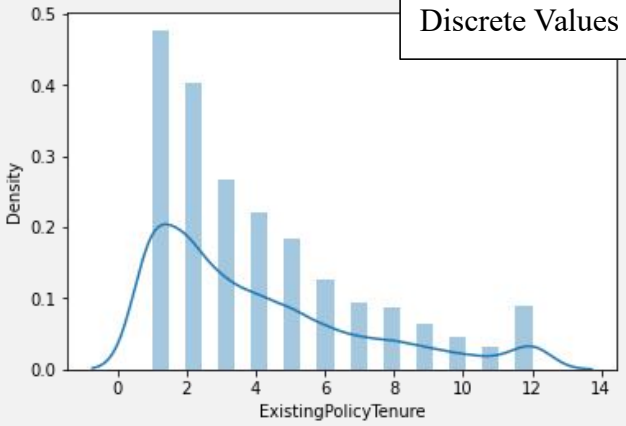
Continuous in a range with two peak values.



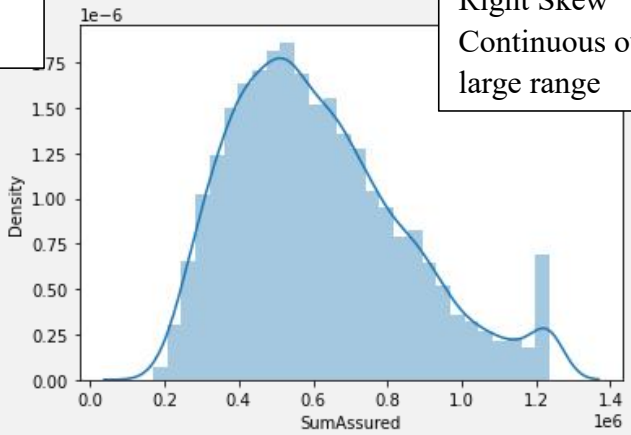
Discrete Values



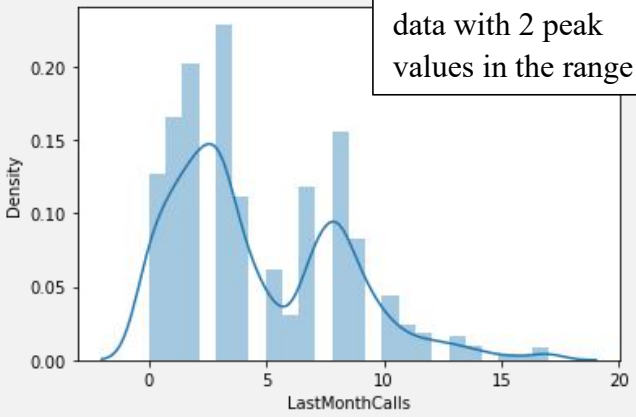




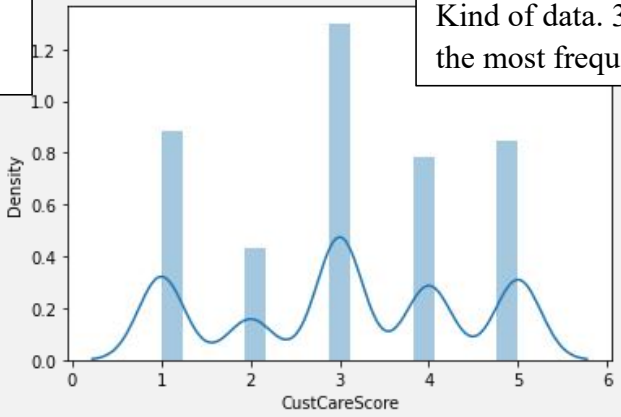
Right Skew Discrete Values



Right Skew Continuous over a large range

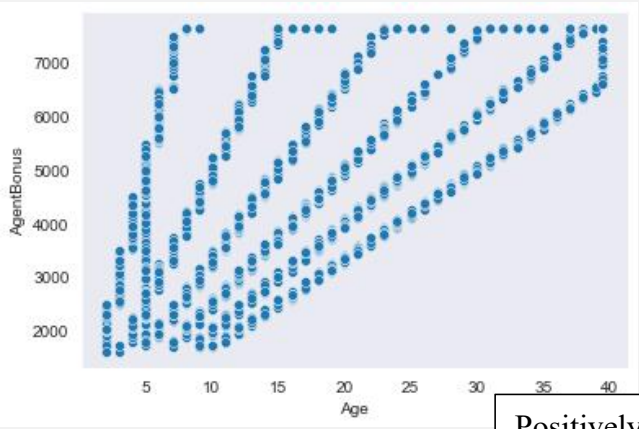


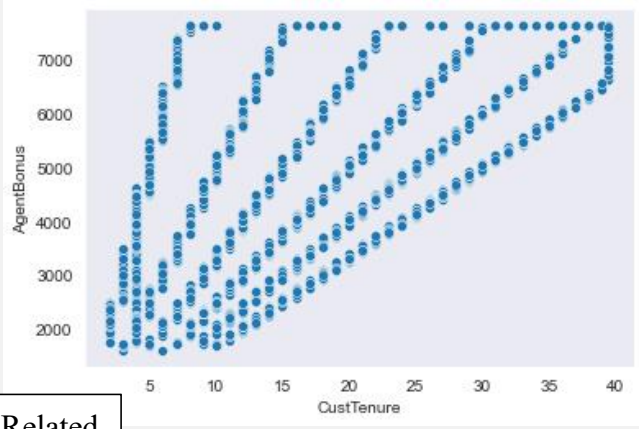
Discrete kind of data with 2 peak values in the range



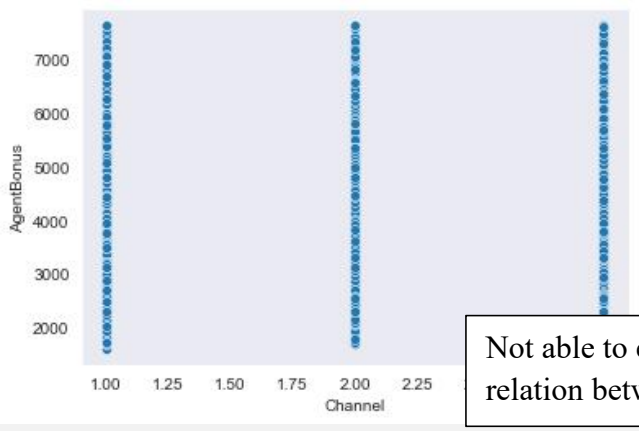
More Discrete Kind of data. 3 is the most frequent

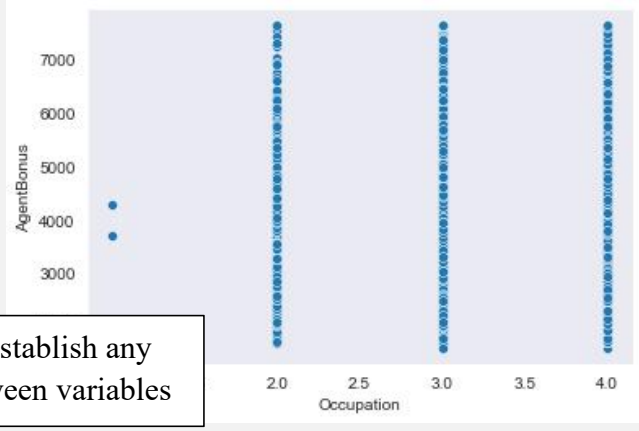
b) Bivariate analysis (relationship between different variables , correlations)



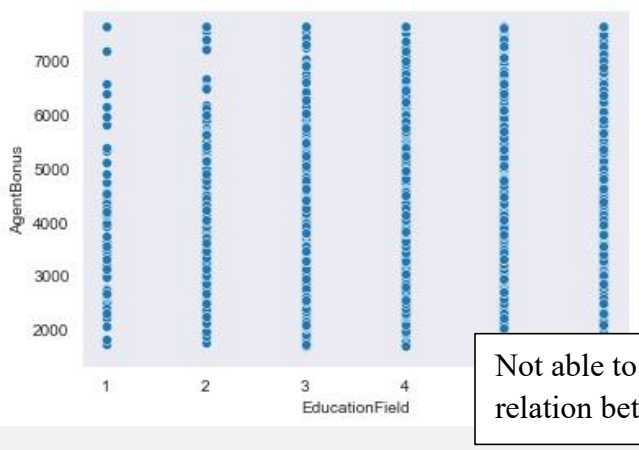


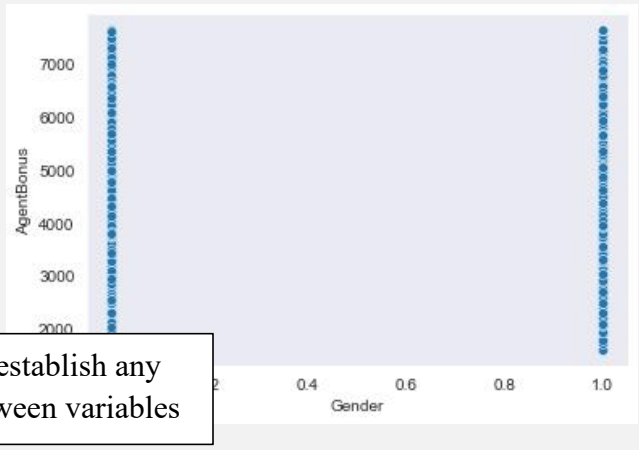
Positively Related





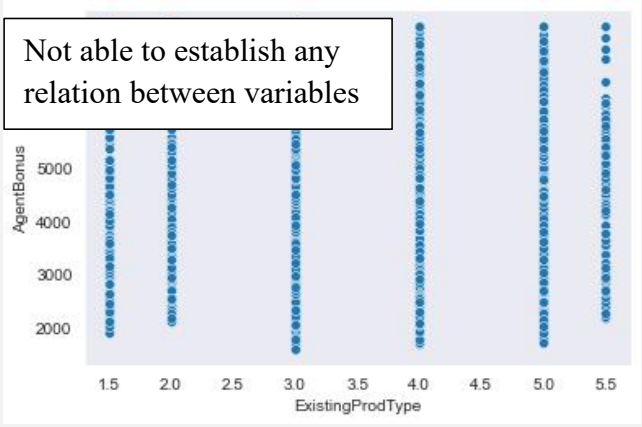
Not able to establish any relation between variables



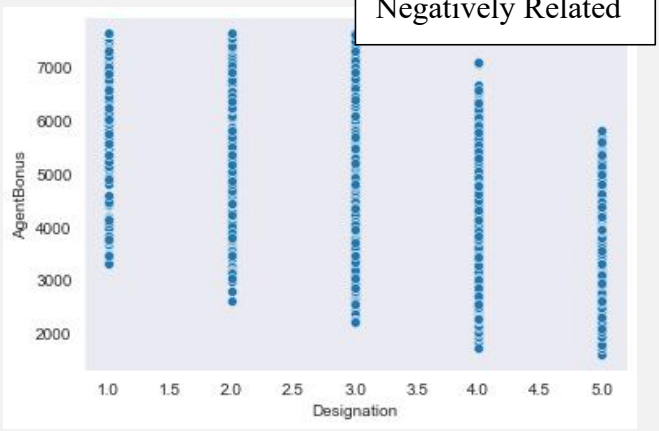


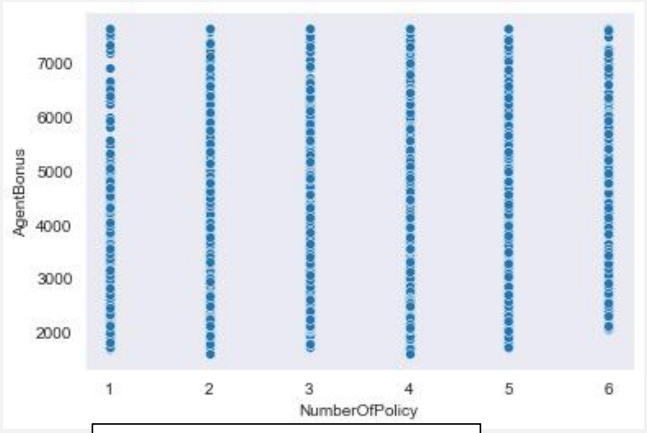
Not able to establish any relation between variables

Not able to establish any relation between variables

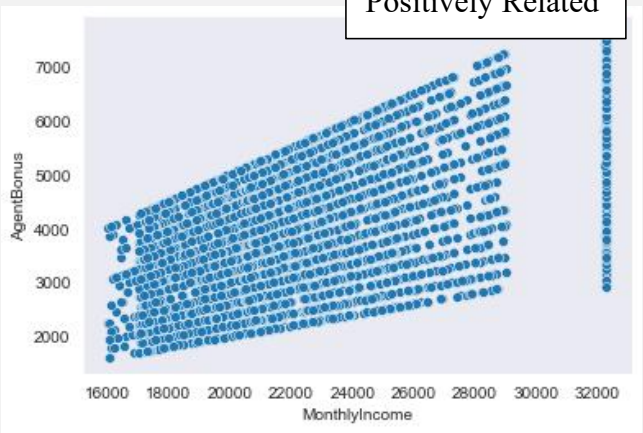


Negatively Related

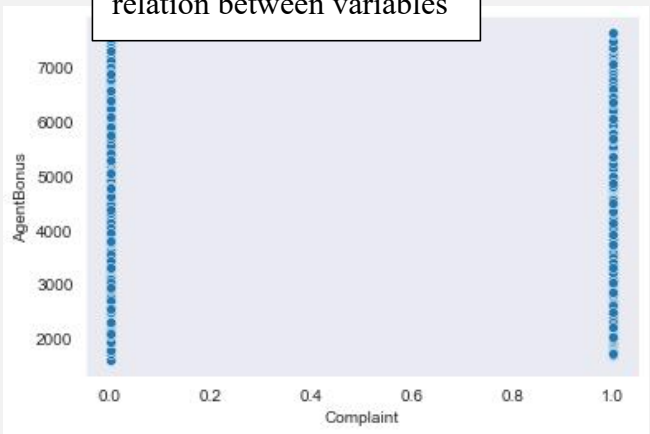




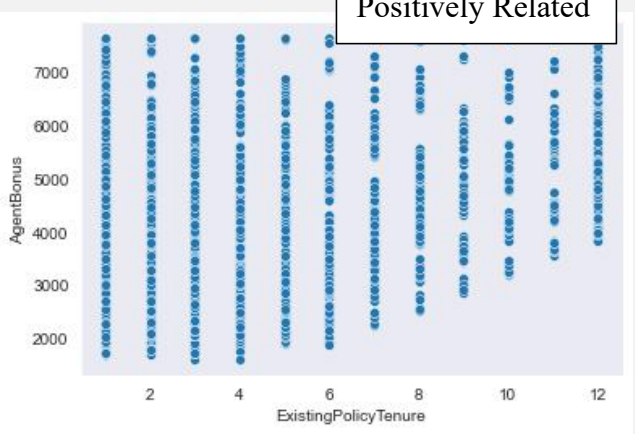
Positively Related

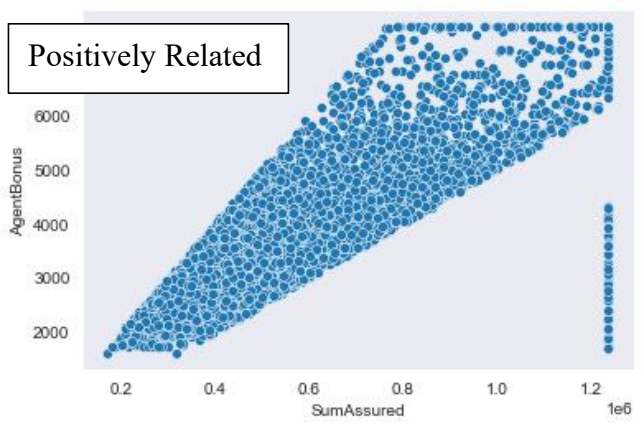


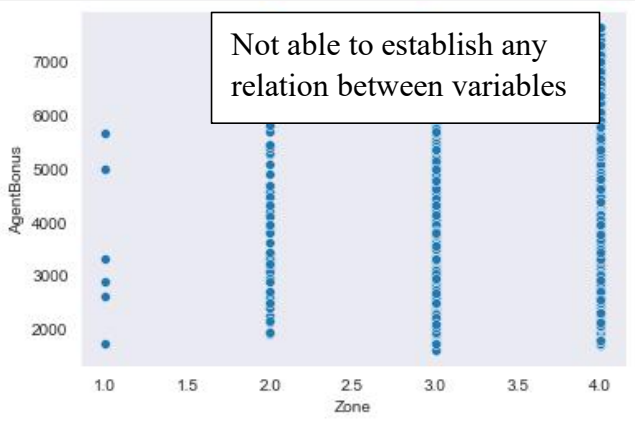
Not able to establish any relation between variables

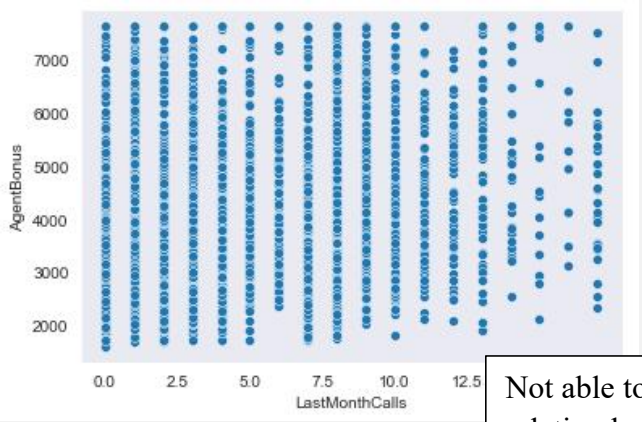


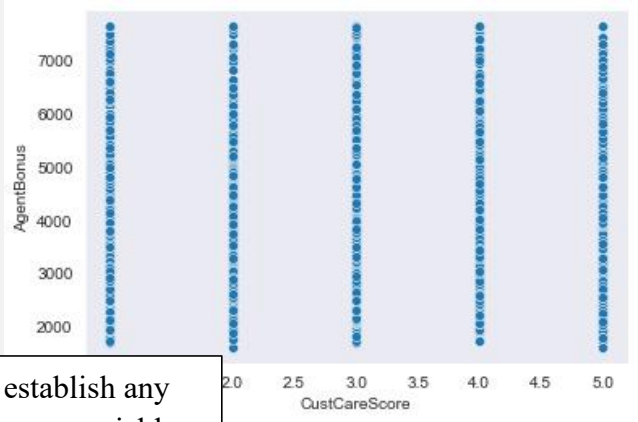
Positively Related







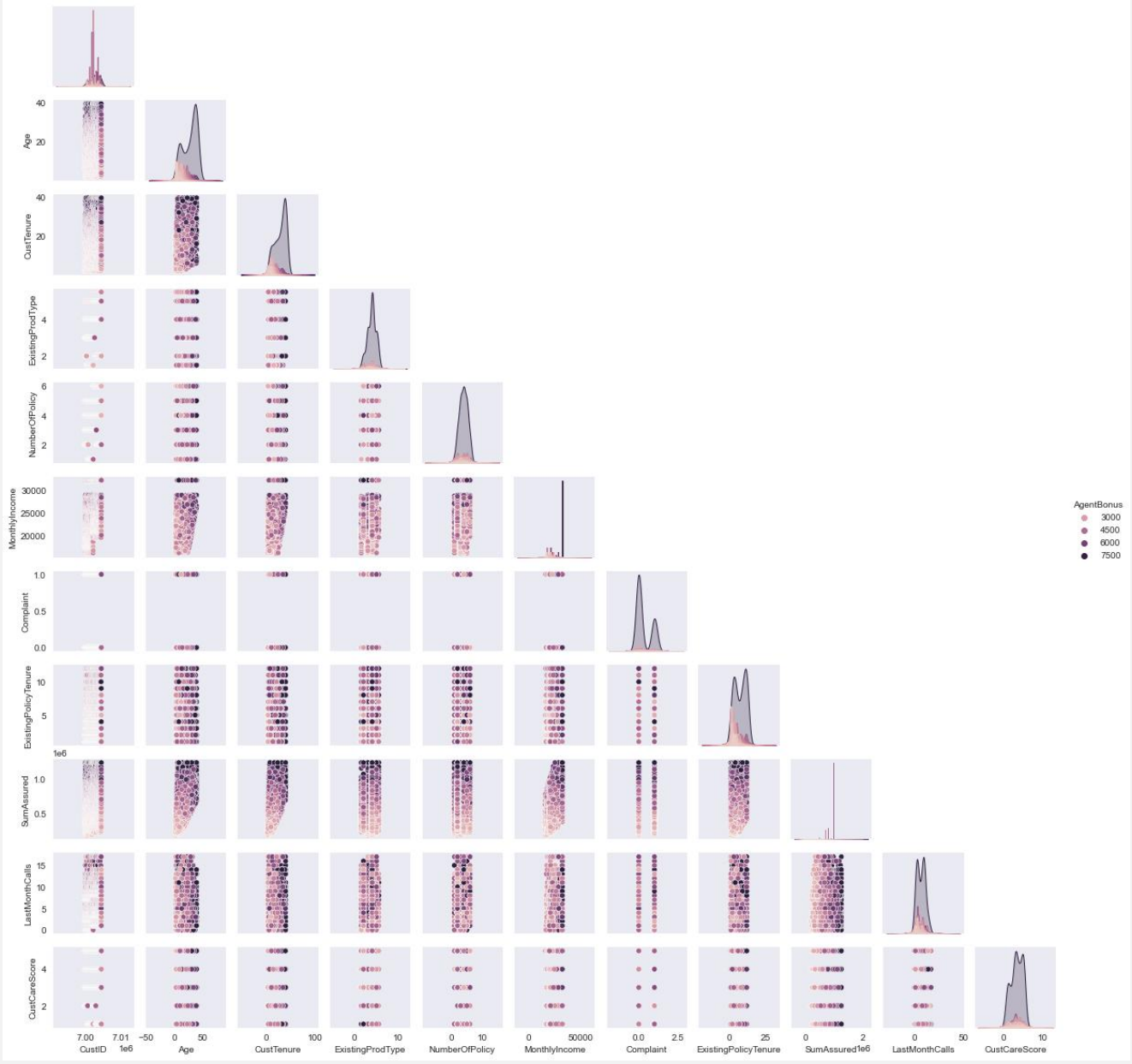




Not able to establish any relation between variables

Most of the variables don't seem to be related closely to each other which means there is low multi-collinearity in the data and each feature would have its importance in building the right model . because of this we have not dropped any columns and would want to build the model to see the variable importance.

The pair plot also seems to suggest the same thing . But due to the huge number of columns pair plot was not providing very clear insight and hence resorted to bi variate plots with every combination possible.


AgentBonus
3000
4500
6000
7500
CustID
Age
CustTenure
ExistingProdType
NumberOfPolicy
MonthlyIncome
Complaint
ExistingPolicyTenure
SumAssured
LastMonthCalls
CustCareScore
1e6

c) Removal of unwanted variables

In the dataset CustID, MaritalStatus and PaymentMethod are all redundant columns and thus have been removed. Chose not to remove any other columns and left to the model phase where the variable importance would be judged.

```
df.drop(['CustID','MaritalStatus','PaymentMethod'],axis=1,inplace=True)
```

d) Missing Value treatment

There are 1166 missing values in the dataset :

Age	*269*
MonthlyIncome	*236*
CustTenure	*226*
ExistingPolicyTenure	*184*
SumAssured	*154*
CustCareScore	*52*
NumberOfPolicy	*45*
LastMonthCalls	*0*
Zone	*0*
Complaint	*0*
AgentBonus	*0*
ExistingProdType	*0*
Gender	*0*
EducationField	*0*
Occupation	*0*
Channel	*0*
Designation	*0*

dtype: int64

The missing values have been treated with most frequent values than median for numeric data including categorical data . The main reason of choosing mode or most frequent entry was it was making more sense considering the sports domain to which the problem belongs . More so as we have been in the various plots as well the numeric data has discrete pattern due to which we treated them as categorical data.

```
null_rows=0
for i in (df.isnull().sum(axis=1)):
    if i>0 :
        null_rows=null_rows+1
print (" Total Missing Rows ",null_rows)
```

```
Total Missing Rows  1073
```

- There can be two options for Missing Value Treatment
  - Either impute the missing values with median or median values for Numeric columns while mode values for categorical columns
  - Drop the rows with missing values but then we are looking at almost 23.74% reduction in data (3447 out of 4520 rows) and hence ruled out

So, Impute the missing values with the mode value of the column.

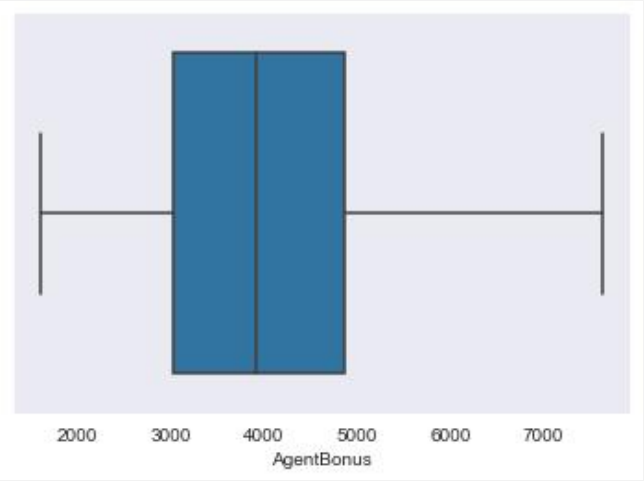
```
from sklearn.impute import SimpleImputer
imputer = SimpleImputer(strategy='most_frequent',missing_values=np.nan)
```

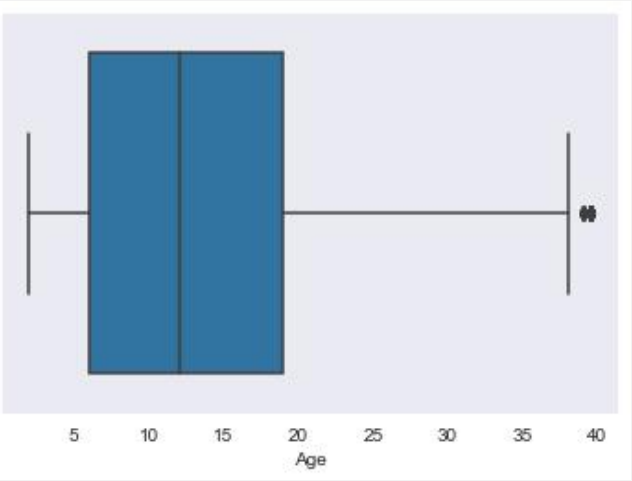
```
for i,col_val in enumerate(list(df.columns)):
    if df[col_val].isnull().sum()>0 :
        df[col_val]=imputer.fit_transform(df[col_val].values.reshape(-1,1))[:,0]
```

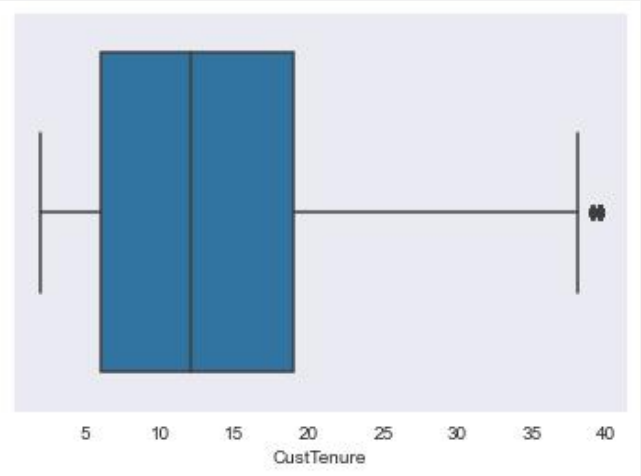
After Treatment of missing value :

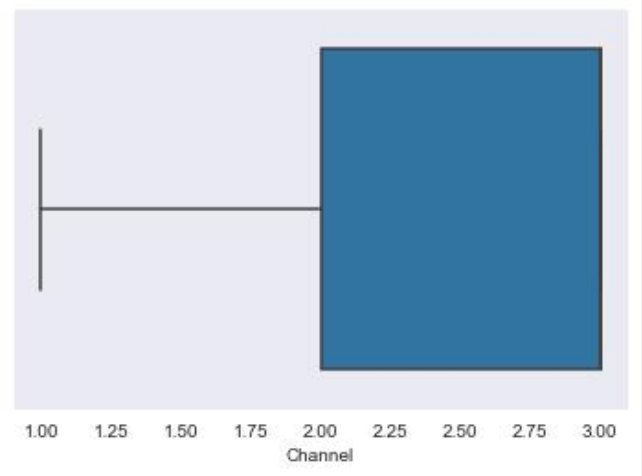
AgentBonus 0
Age 0
CustTenure 0
Channel 0
Occupation 0
EducationField 0
Gender 0
ExistingProdType 0
Designation 0
NumberOfPolicy 0
MonthlyIncome 0
Complaint 0
ExistingPolicyTenure 0
SumAssured 0
Zone 0
LastMonthCalls 0
CustCareScore 0
dtype: int64

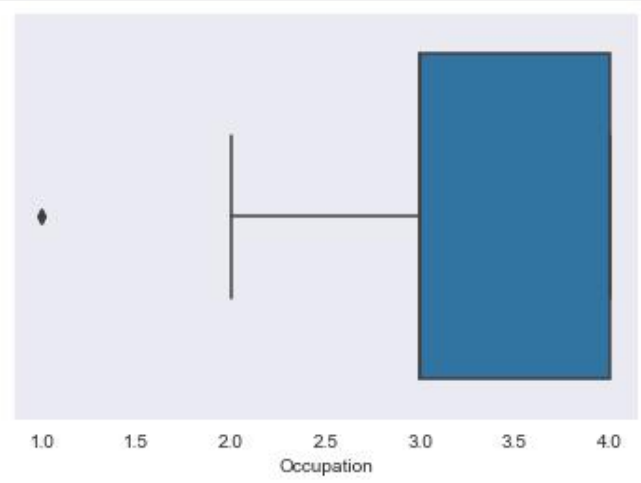
e) Outlier treatment

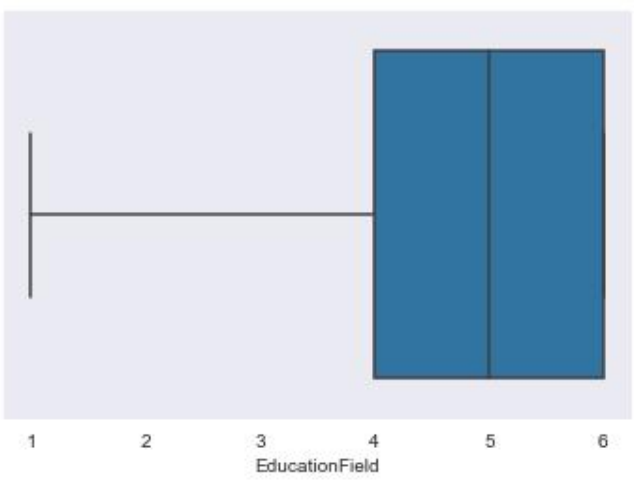


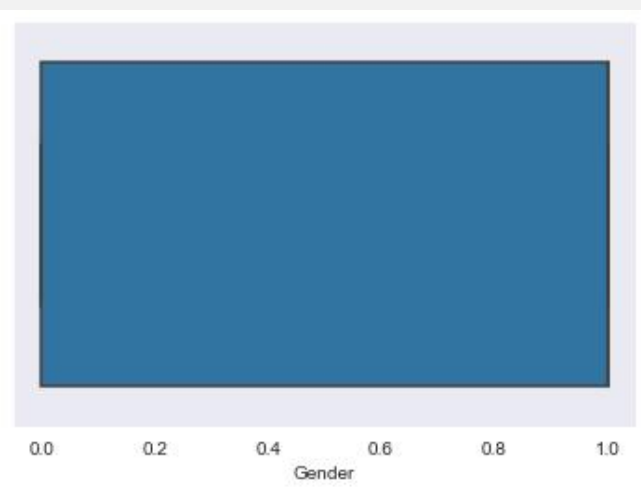


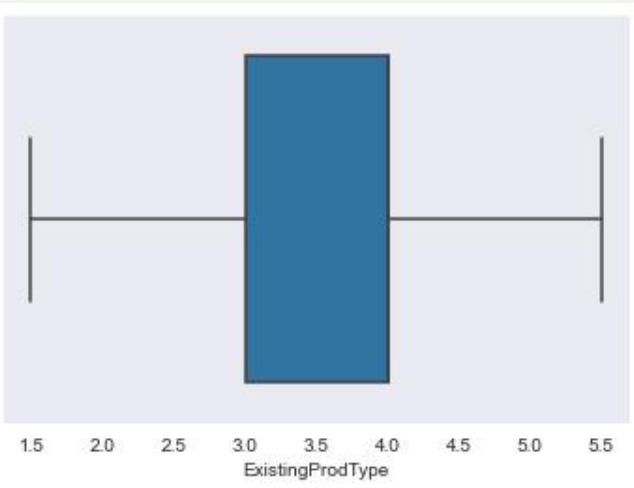

5
10
15
20
25
30
35
40
CustTenure

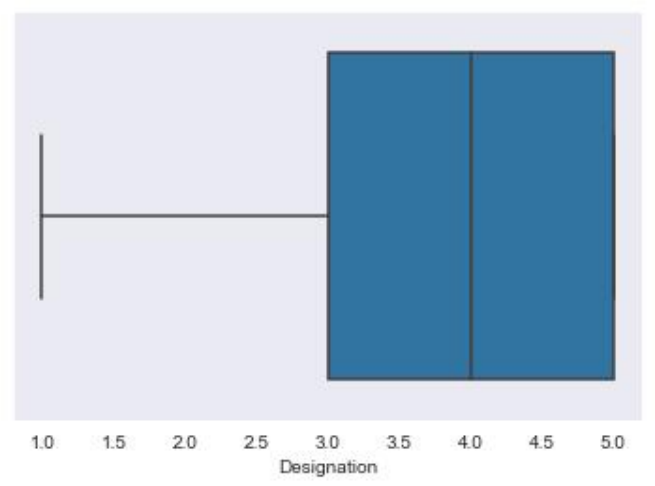

1.00
1.25
1.50
1.75
2.00
2.25
2.50
2.75
3.00
Channel

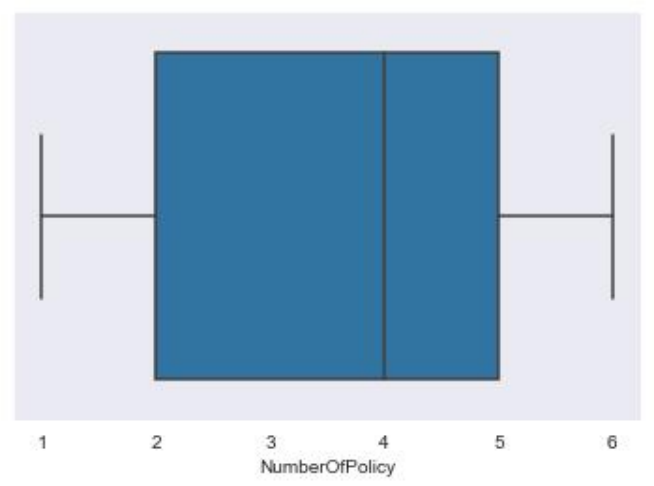

1.0
1.5
2.0
2.5
3.0
3.5
4.0
Occupation

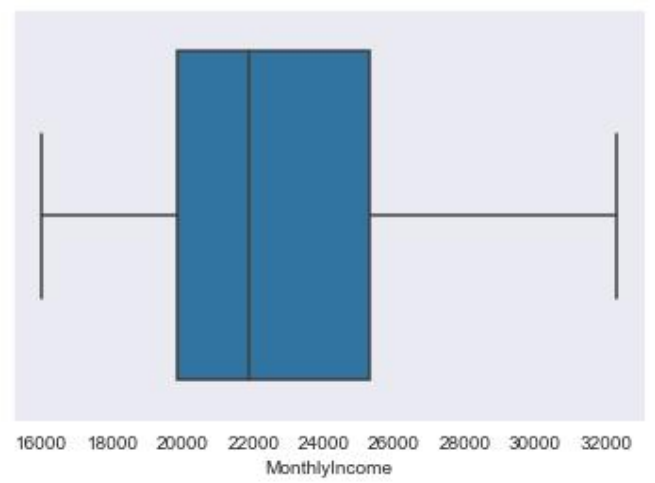

1
2
3
4
5
6
EducationField

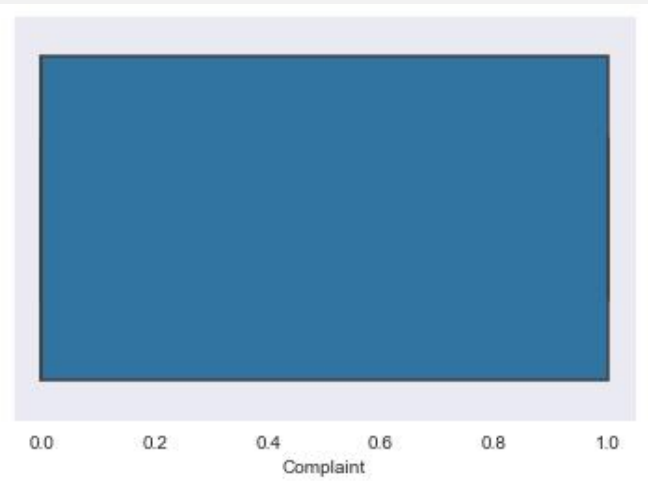

0.0
0.2
0.4
0.6
0.8
1.0
Gender

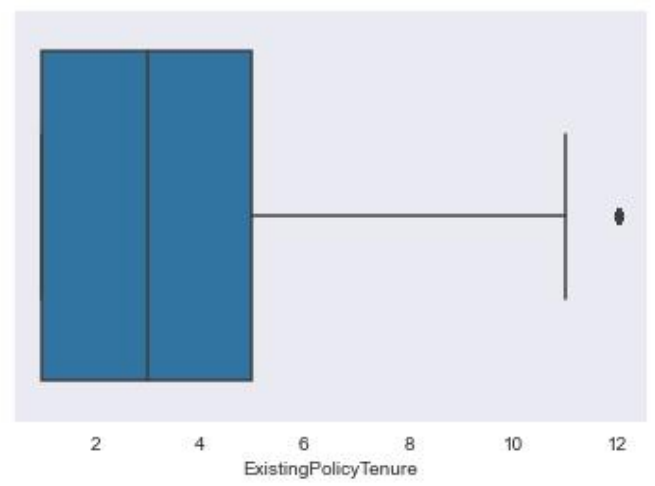

1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
ExistingProdType











Even though most of the numeric data here is discrete but few of the variables here are playing an important role in predicting the required value for the model which might get affected because of the outlying values, hence the outliers might reduce the value to the model. Like the age and customer tenure with the company which stands out while most of the others are in the right range.

So, in favour of doing the outlier treatment :
Detecting the Outliers

```
col_names = list(df.select_dtypes(exclude=['object']).columns)
fig, ax = plt.subplots(len(col_names), figsize=(5,50)).
for i,col_val in enumerate(col_names):
    sns.boxplot(df[col_val])
    ax[i].set_title('{}'.format(col_val), fontsize=8)
    plt.show()
```

Outlier Treatment :

```
def remove_outlier(col):
    sorted(col)
    Q1,Q3=col.quantile([0.25,0.75])
    IQR=Q3-Q1
    lower_range= Q1-(1.5 * IQR)
    upper_range= Q3+(1.5 * IQR)
    return lower_range, upper_range
```

```
for i,col_val in enumerate(col_names):
    lwr,upr=remove_outlier(df[col_val])
    df[col_val]=np.where(df[col_val]>upr,upr,df[col_val])
    df[col_val]=np.where(df[col_val]<lwr,lwr,df[col_val])
    print("Outlier fixed for ", col_val)
```

Outlier fixed for CustID
Outlier fixed for AgentBonus
Outlier fixed for Age
Outlier fixed for CustTenure
Outlier fixed for ExistingProdType
Outlier fixed for NumberOfPolicy
Outlier fixed for MonthlyIncome
Outlier fixed for Complaint
Outlier fixed for ExistingPolicyTenure
Outlier fixed for SumAssured
Outlier fixed for LastMonthCalls
Outlier fixed for CustCareScore

f) Variable transformation

The variables has been encoded to numeric values for the following variables :

```
df['Channel'] = df['Channel'].replace(to_replace='Online',value=1)
df['Channel'] = df['Channel'].replace(to_replace='Third Party Partner',value=2)
df['Channel'] = df['Channel'].replace(to_replace='Agent',value=3)
```

```
df['Occupation'] = df['Occupation'].replace(to_replace='Free Lancer',value=1)
df['Occupation'] = df['Occupation'].replace(to_replace='Large Business',value=2)
df['Occupation'] = df['Occupation'].replace(to_replace='Small Business',value=3)
df['Occupation'] = df['Occupation'].replace(to_replace='Salaried',value=4)
```

```
df['EducationField'] = df['EducationField'].replace(to_replace='MBA',value=1)
df['EducationField'] = df['EducationField'].replace(to_replace='Post Graduate',value=2)
df['EducationField'] = df['EducationField'].replace(to_replace='Engineer',value=3)
df['EducationField'] = df['EducationField'].replace(to_replace='Diploma',value=4)
df['EducationField'] = df['EducationField'].replace(to_replace='Under Graduate',value=5)
df['EducationField'] = df['EducationField'].replace(to_replace='Graduate',value=6)
```

```
df['Gender'] = df['Gender'].replace(to_replace='Female',value=0)
df['Gender'] = df['Gender'].replace(to_replace='Male',value=1)
```

```
df['Designation'] = df['Designation'].replace(to_replace='VP',value=1)
df['Designation'] = df['Designation'].replace(to_replace='AVP',value=2)
df['Designation'] = df['Designation'].replace(to_replace='Senior Manager',value=3)
df['Designation'] = df['Designation'].replace(to_replace='Manager',value=4)
df['Designation'] = df['Designation'].replace(to_replace='Executive',value=5)
```

```
df['Zone'] = df['Zone'].replace(to_replace='South',value=1)
df['Zone'] = df['Zone'].replace(to_replace='East',value=2)
df['Zone'] = df['Zone'].replace(to_replace='North',value=3)
df['Zone'] = df['Zone'].replace(to_replace='West',value=4)
```

g) Addition of new variables

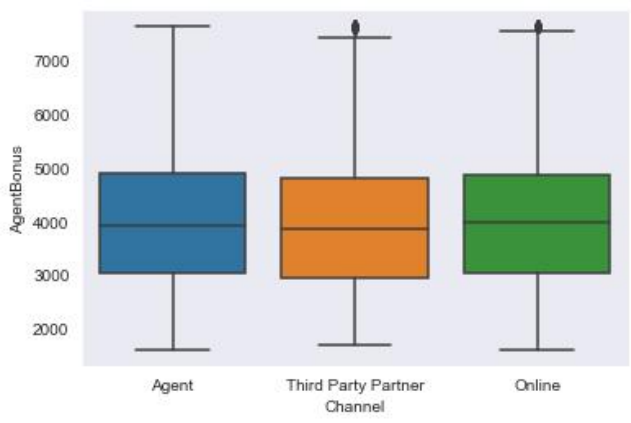
No new variables were added at this stage . But before proceeding with the model one hot encoding would be required on few categories which would increase the number of column not essentially the number of variables.

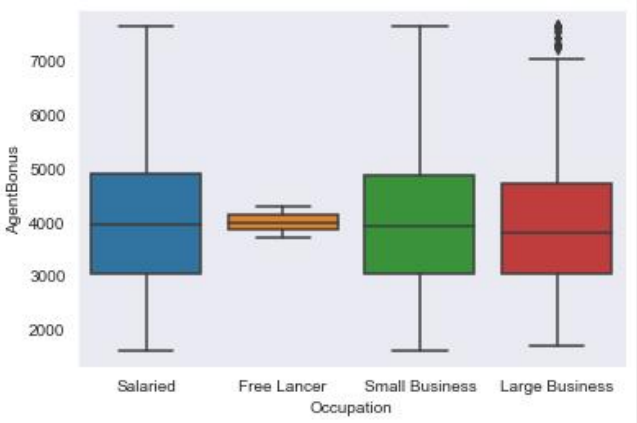
4) Business insights from EDA

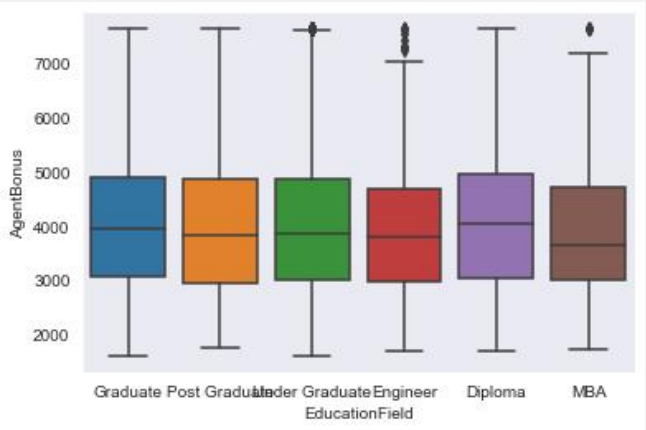
a) Is the data unbalanced? If so, what can be done? Please explain in the context of the business

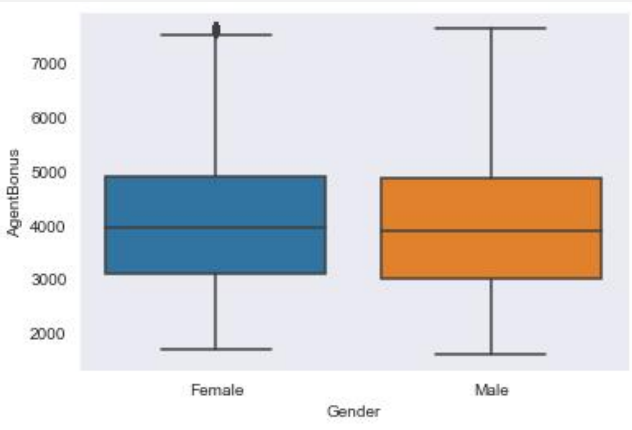
The data is balanced.

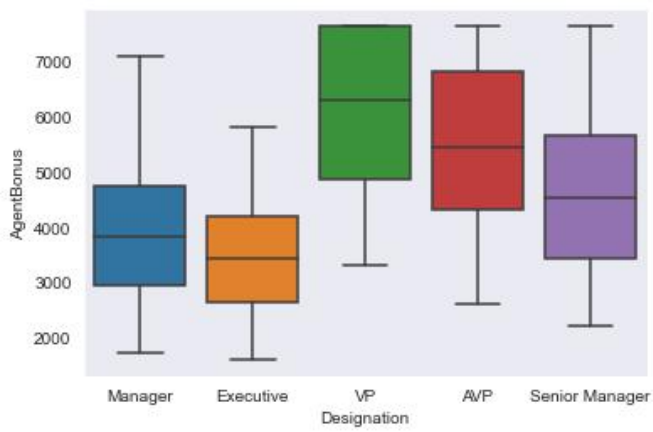
b) Any business insights using clustering (if applicable)

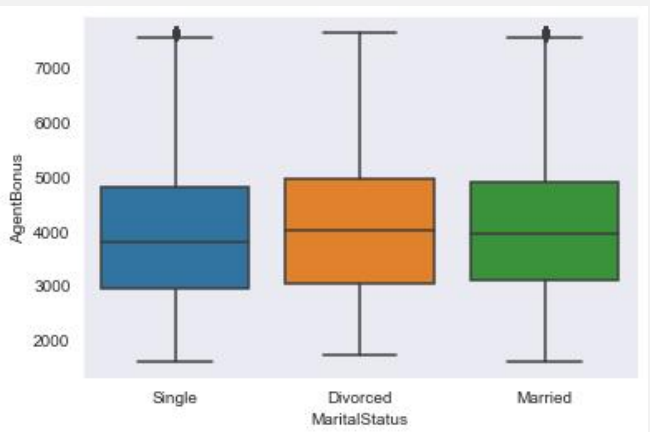


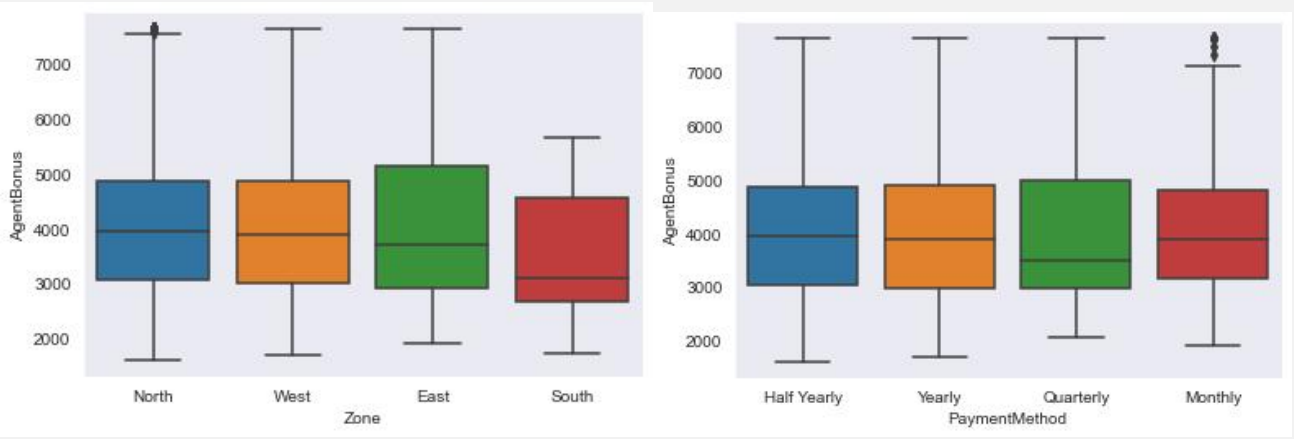






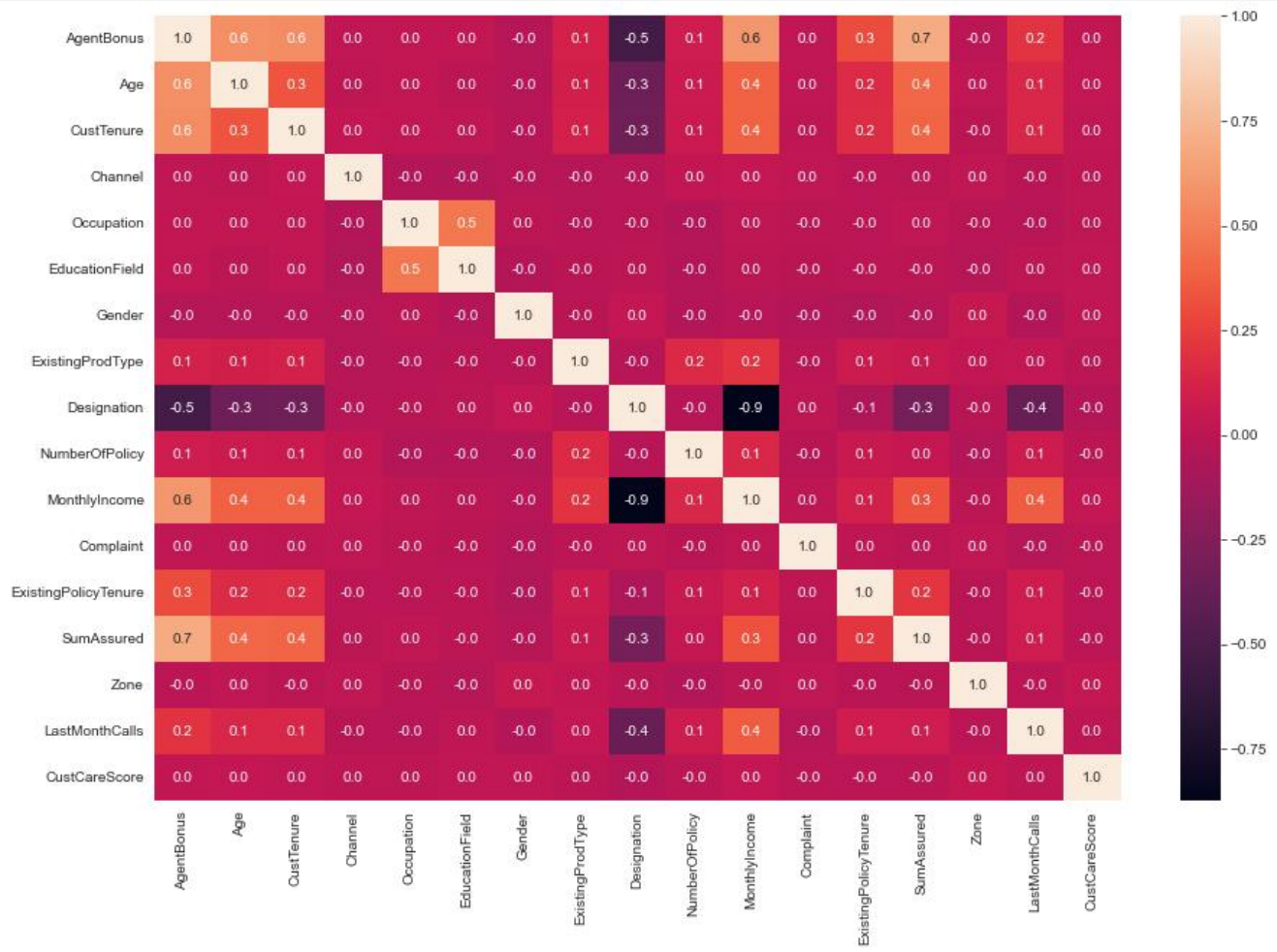






Age, CustTenure, monthlyIncome, SumAssured seems to be correlated with AgentBonus which means with increase in age and then tenure of customer also increase in sumAssured and monthlyincome brings the best performance in an Agent, but may not be true for the everyone. Designation plays an negative role on the Agents Bonus as well.

d) Any other business insights



- *Age, CustTenure, monthlyIncome, SumAssured seems to be correlated with AgentBonus which means with increase in age and then tenure of customer also increase in sumAssured and monthlyincome brings the best performance in an Agent, but may not be true for the everyone.*
- *Designation plays an negative role on the Agents Bonus as well as at monthlyIncome which means as move from VP towards Executive the Bonus and monthy income decreases.*