



# Great Learning & UT Austin



## Predictive Modeling

### Assignment

Gunjar Fuley  
Batch- PGPDSBA Online Nov\_A 2020  
Email- [gforgunjaar@gmail.com](mailto:gforgunjaar@gmail.com)



## Problem 1: Linear Regression

You are a part of an investment firm and your work is to do research about these 759 firms. You are provided with the dataset containing the sales and other attributes of these 759 firms. Predict the sales of these firms on the bases of the details given in the dataset so as to help your company in investing consciously. Also, provide them with 5 attributes that are most important.

Data Dictionary for Firm\_level\_data:

1. sales: Sales (in millions of dollars).
2. capital: Net stock of property, plant, and equipment.
3. patents: Granted patents.
4. randd: R&D stock (in millions of dollars).
5. employment: Employment (in 1000s).
6. sp500: Membership of firms in the S&P 500 index. S&P is a stock market index that measures the stock performance of 500 large companies listed on stock exchanges in the United States
7. tobing: Tobin's q (also known as q ratio and Kaldor's v) is the ratio between a physical asset's market value and its replacement value.
8. value: Stock market value.
9. institutions: Proportion of stock owned by institutions.

Questions for Problem 1:

**1.1) Read the data and do exploratory data analysis. Describe the data briefly. (Check the null values, data types, shape, EDA). Perform Univariate and Bivariate Analysis.**

We received the data in the CSV format with file named as Firm\_level\_data.csv.

The data was uploaded using standard pandas library.

```
Index(['Unnamed: 0', 'sales', 'capital', 'patents', 'randd', 'employment',  
      'sp500', 'tobinq', 'value', 'institutions'],  
      dtype='object')
```

Above are the columns present in the dataset.

	Unnamed: 0	sales	capital	patents	randd	employment	sp500	tobinq	value	institutions
0	0	826.995050	161.603986	10	382.078247	2.306000	no	11.049511	1625.453755	80.27
1	1	407.753973	122.101012	2	0.000000	1.860000	no	0.844187	243.117082	59.02
2	2	8407.845588	6221.144614	138	3296.700439	49.659005	yes	5.205257	25865.233800	47.70
3	3	451.000010	266.899987	1	83.540161	3.071000	no	0.305221	63.024630	26.88
4	4	174.927981	140.124004	2	14.233637	1.947000	no	1.063300	67.406408	49.46

Above picture shows the initial 5 data points in the dataset. Here, we can see that the column 'Unnamed: 0' is useless and doesn't have any relevant information. Therefore, it was dropped.

	sales	capital	patents	randd	employment	sp500	tobinq	value	institutions
754	1253.900196	708.299935	32	412.936157	22.100002	yes	0.697454	267.119487	33.50
755	171.821025	73.666008	1	0.037735	1.684000	no	NaN	228.475701	46.41
756	202.726967	123.926991	13	74.861099	1.460000	no	5.229723	580.430741	42.25
757	785.687944	138.780992	6	0.621750	2.900000	yes	1.625398	309.938651	61.39
758	22.701999	14.244999	5	18.574360	0.197000	no	2.213070	18.940140	7.50

Above picture represents the last 5 data points of dataset after removing the column 'Unnamed: 0'.

We can see that the remaining columns are good to go for further analysis.

(759, 9)

The dataset has 759 rows and 9 columns.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 759 entries, 0 to 758
Data columns (total 9 columns):
#   Column          Non-Null Count  Dtype
---  -
0   sales           759 non-null    float64
1   capital         759 non-null    float64
2   patents         759 non-null    int64
3   randd           759 non-null    float64
4   employment      759 non-null    float64
5   sp500           759 non-null    object
6   tobinq          738 non-null    float64
7   value           759 non-null    float64
8   institutions    759 non-null    float64
dtypes: float64(7), int64(1), object(1)
memory usage: 53.5+ KB
```

Above picture shows the information of the dataset. Here, we can see that the all the variables have data type 'float64' except patents which is 'int64' and sp500 which is a string.

The data in the column 'sp500' has values 'yes/no' which were converted to 0 and 1.

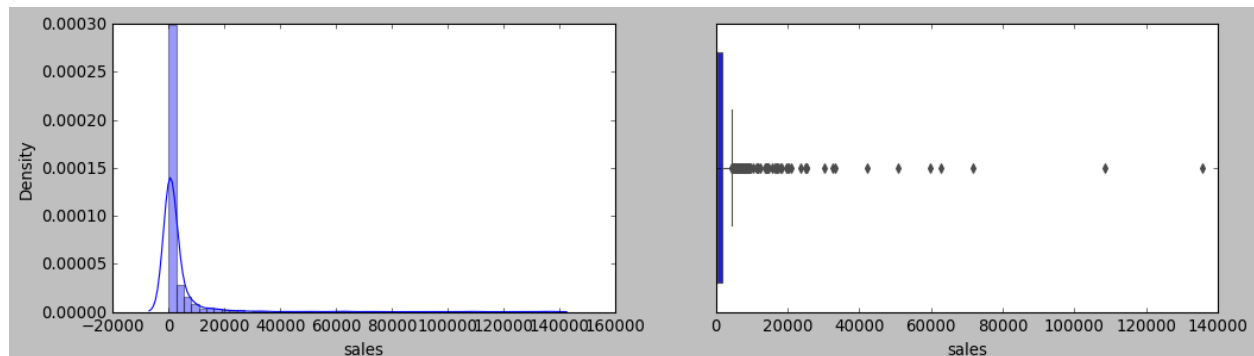
Here, 'yes' is replaced with 1 and 'no' with 0.

	count	mean	std	min	25%	50%	75%	max
sales	759.0	2689.705158	8722.060124	0.138000	122.920000	448.577082	1822.547366	135696.788200
capital	759.0	1977.747498	6466.704896	0.057000	52.650501	202.179023	1075.790020	93625.200560
patents	759.0	25.831357	97.259577	0.000000	1.000000	3.000000	11.500000	1220.000000
randd	759.0	439.938074	2007.397588	0.000000	4.628262	36.864136	143.253403	30425.255860
employment	759.0	14.164519	43.321443	0.006000	0.927500	2.924000	10.050001	710.799925
sp500	759.0	0.285903	0.452141	0.000000	0.000000	0.000000	1.000000	1.000000
tobinq	738.0	2.794910	3.366591	0.119001	1.018783	1.680303	3.139309	20.000000
value	759.0	2732.734750	7071.072362	1.971053	103.593946	410.793529	2054.160386	95191.591160
institutions	759.0	43.020540	21.685586	0.000000	25.395000	44.110000	60.510000	90.150000

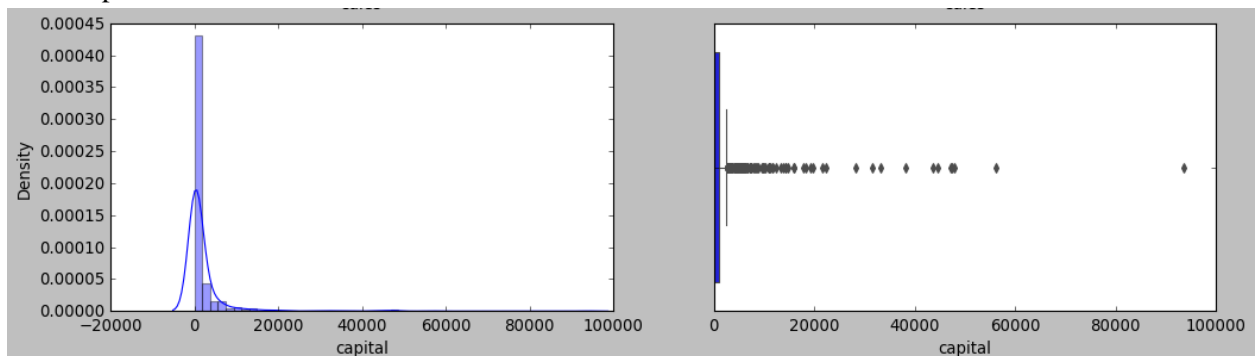
Above picture represents the descriptive statistics of all the variables present in the dataset. Mean of the target column 'sales' is 2689.70. Here the minimum value is very less i.e. 0.138 which is an outlier. In column 'patents' also we can see that mean is 25.83 but the minimum number is 0 and maximum is 1220. We can see that the treatment of the data is important before building the model.

### Univariate Analysis

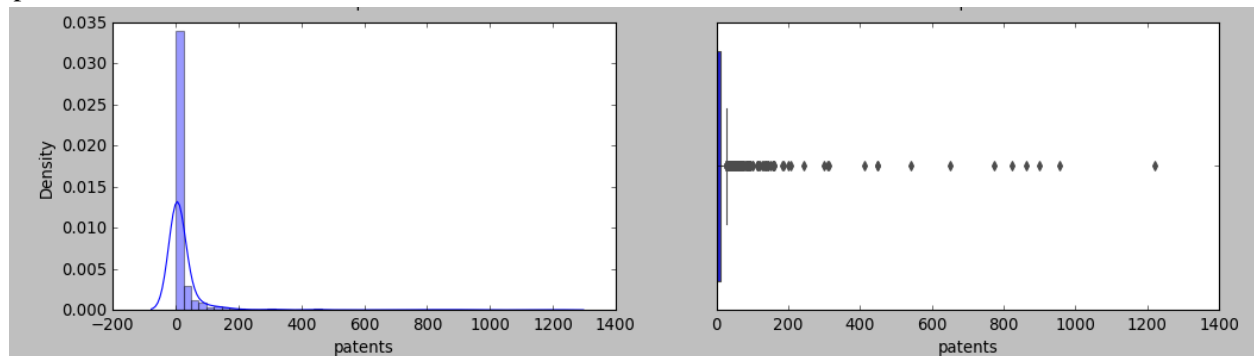
- 'sales'



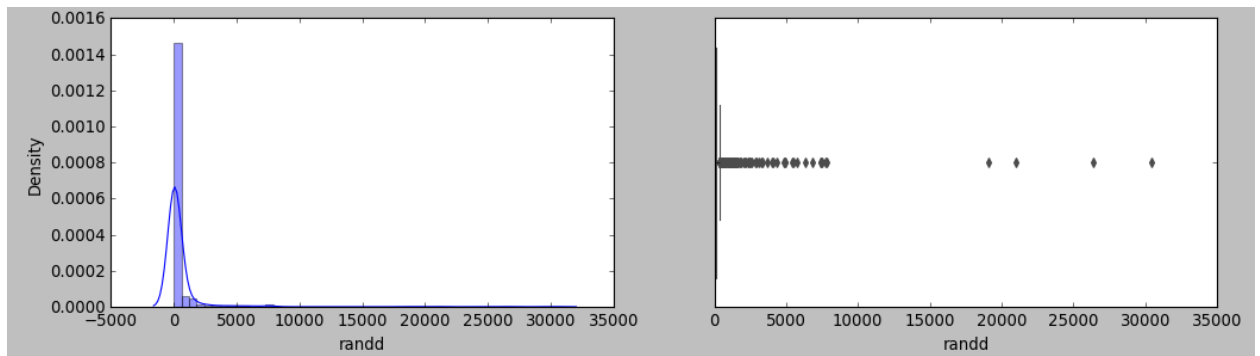
- 'capital'



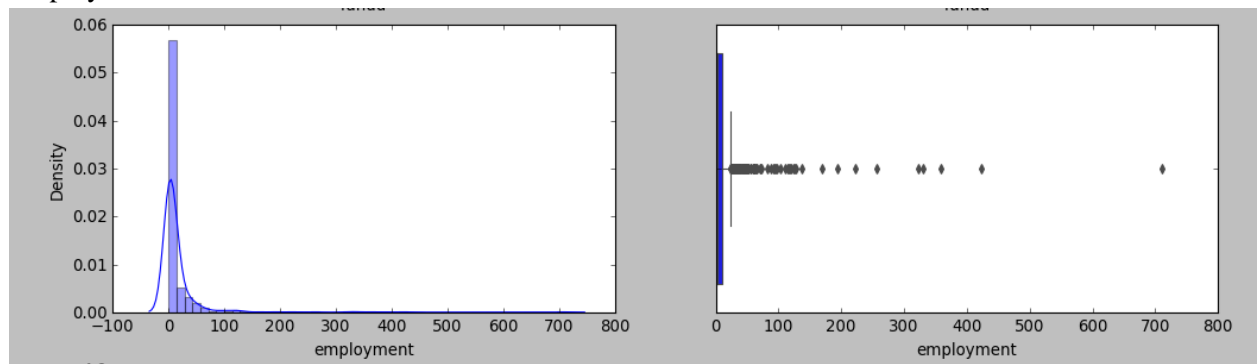
- 'patents'



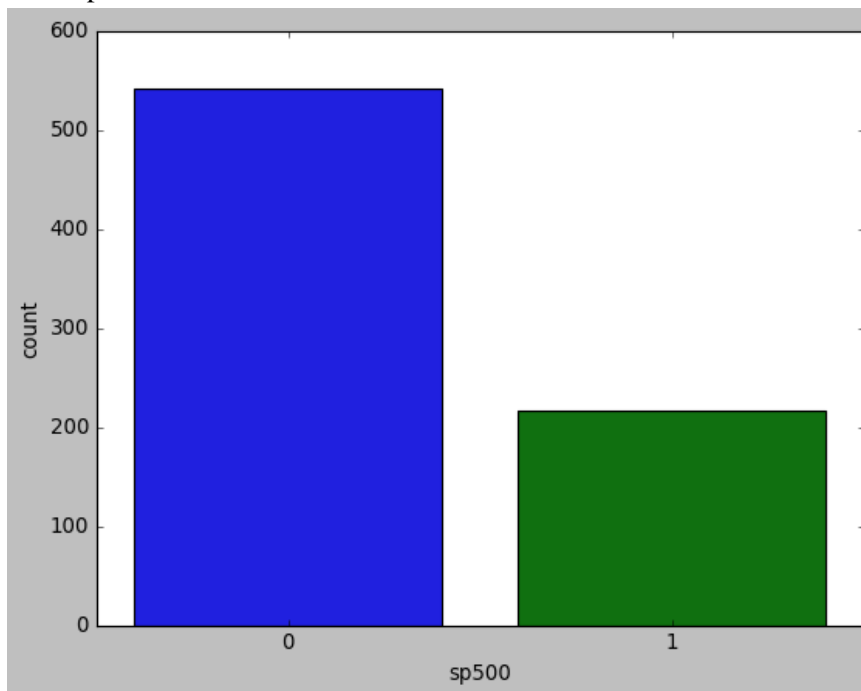
- 'randd'



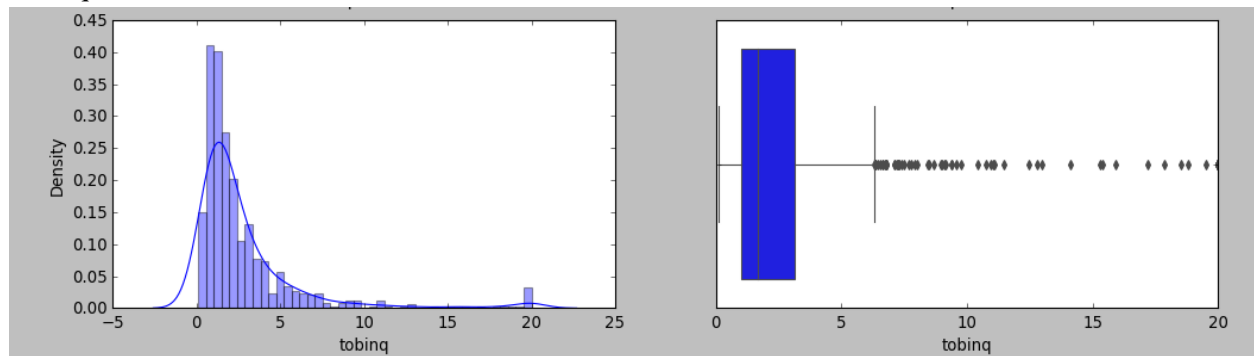
- 'employment'



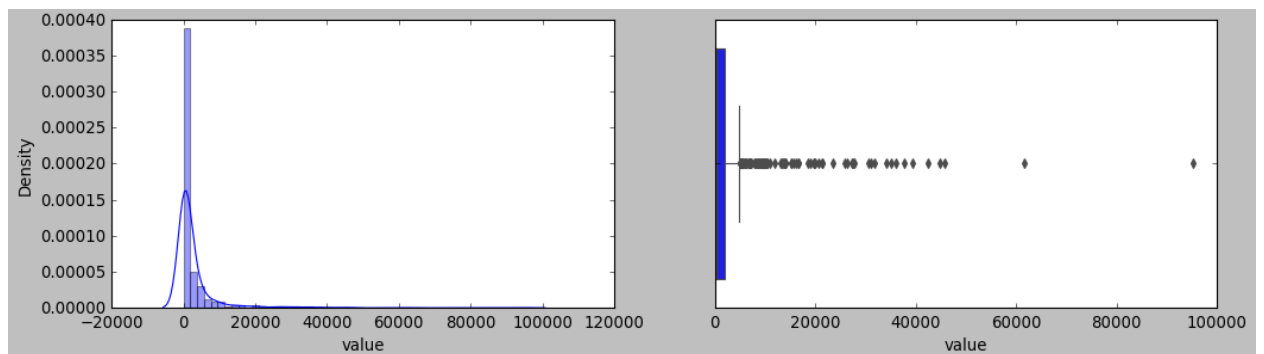
- 'sp500'



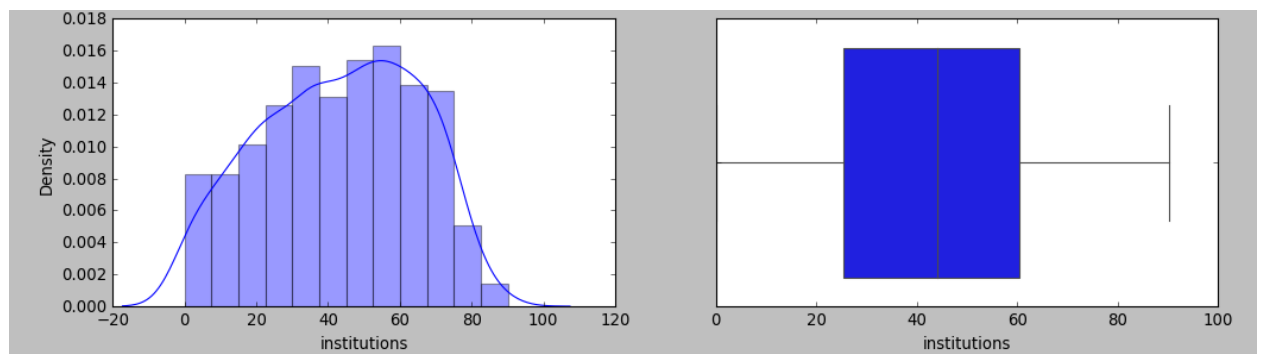
- 'tobinq'



- 'value'

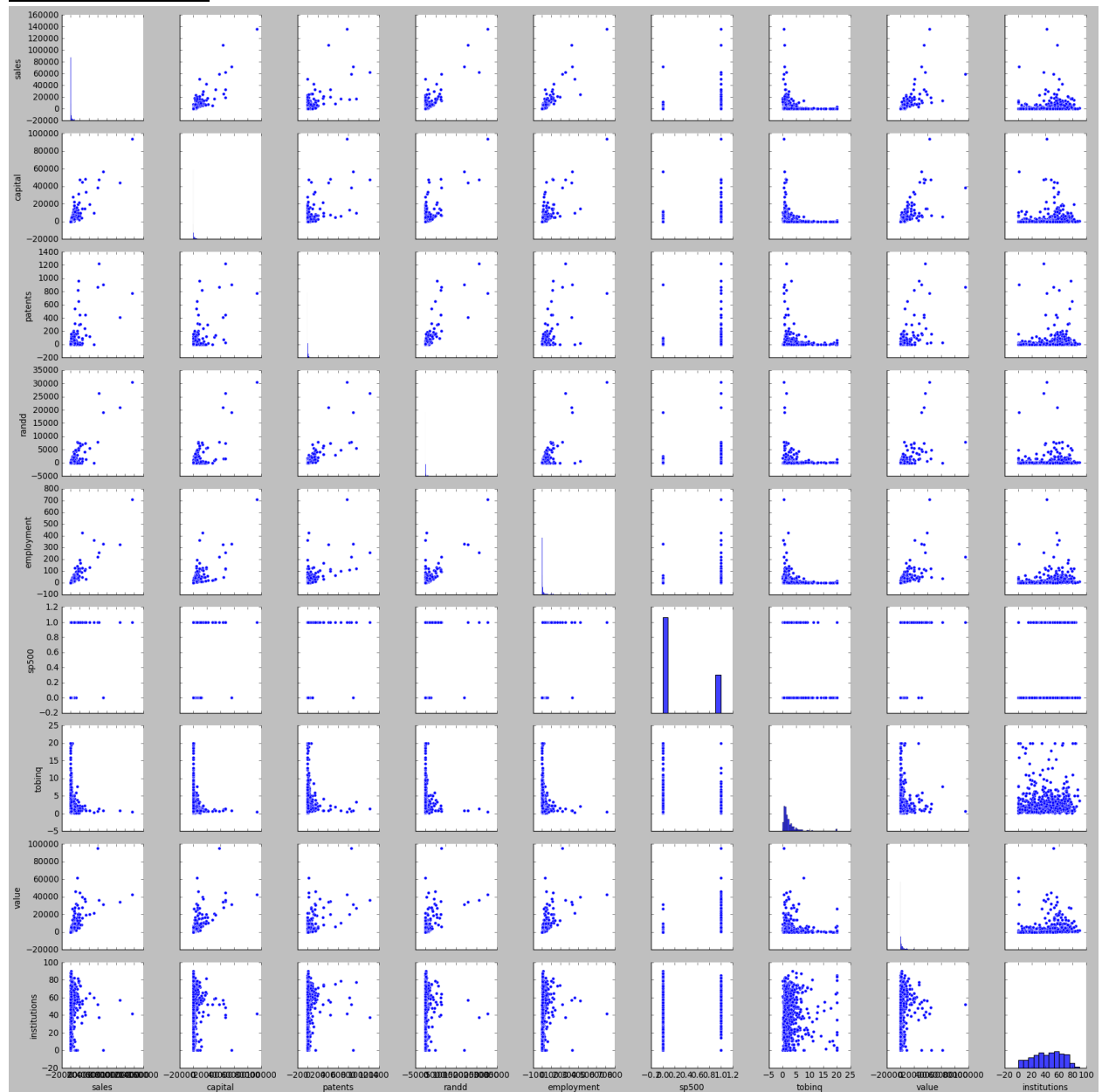


- 'institutions'



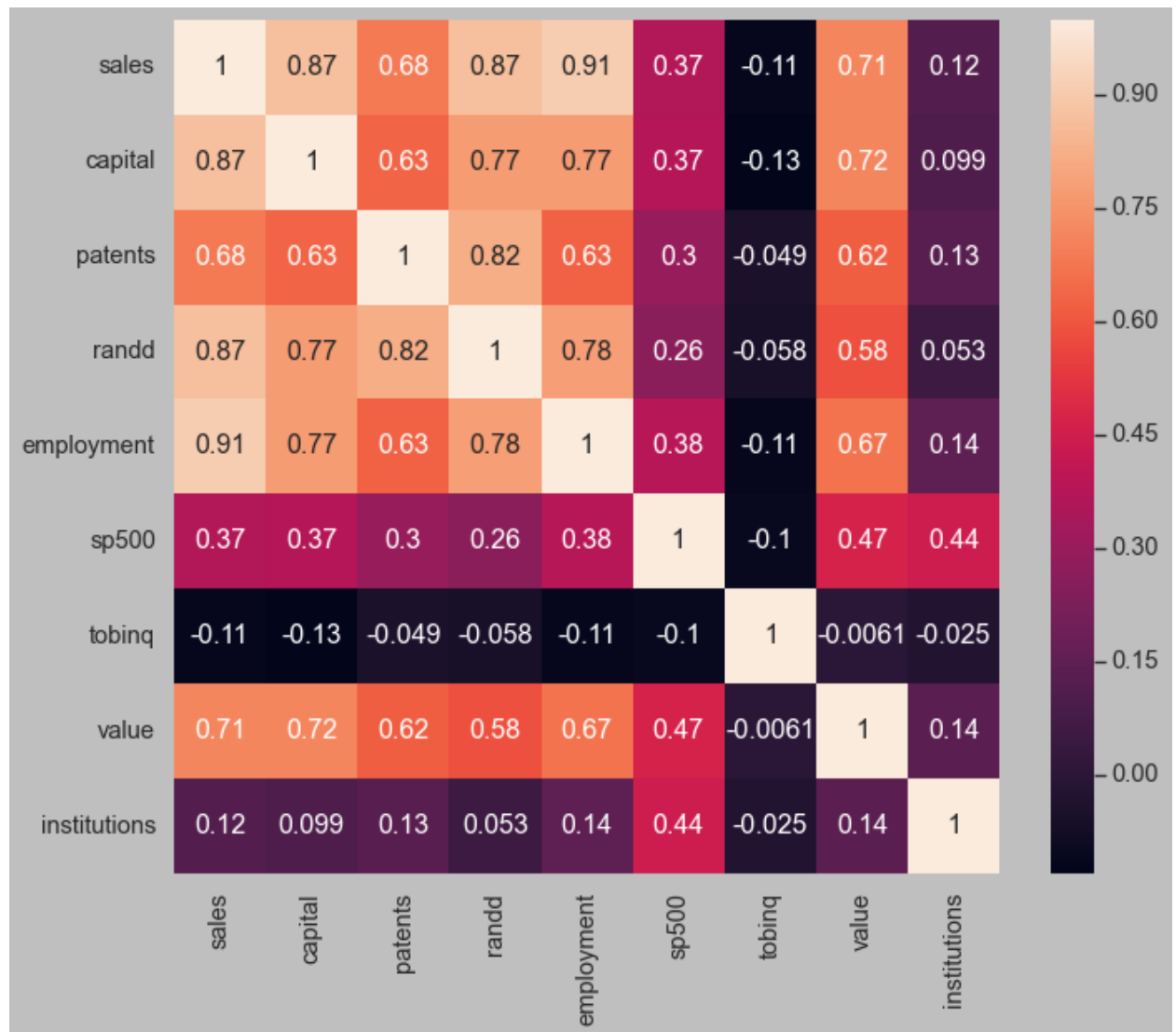
- From the above diagrams of all the variables, we can see that 'sales', 'capital', 'patents', 'randd', 'employment', 'tobinq', 'value', 'institutions'. These variables also have outliers present.
- 'sp500' is categorical in nature, where 'no' has higher weightage than 'yes'.
- For variable 'institutions', the data is uniformly distributed.

## Bivariate Analysis



- Here we can see that with the increase in capital the sales are increasing
- With the increase in patents the sales is increasing
- Randd, employment and stock value also shows the trend where increase in sales shows increase in the variable
- For institutions, the data points are randomly distributed therefore not showing any trend
- Sp500 is categorical variable and shows that companies with 'yes' has more sales than 'no'

## Correlation Matrix



- There is a high correlation of 0.91 between 'sales' and 'employment'.
- There is a high correlation of 0.87 between 'sales' and 'capital'.
- There is a high correlation of 0.87 between 'sales' and 'randd'.
- There is a high correlation of 0.82 between 'patents' and 'randd'.



## 1.2) Impute null values if present? Do you think scaling is necessary in this case?

```
sales      0
capital    0
patents    0
randd      0
employment 0
sp500      0
tobinq     21
value      0
institutions 0
dtype: int64
```

Variable 'tobinq' has 21 NULL values present in it.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 759 entries, 0 to 758
Data columns (total 9 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   sales           759 non-null   float64
1   capital         759 non-null   float64
2   patents         759 non-null   int64
3   randd           759 non-null   float64
4   employment      759 non-null   float64
5   sp500           759 non-null   int64
6   tobinq          759 non-null   float64
7   value           759 non-null   float64
8   institutions    759 non-null   float64
dtypes: float64(7), int64(2)
memory usage: 53.5 KB
```

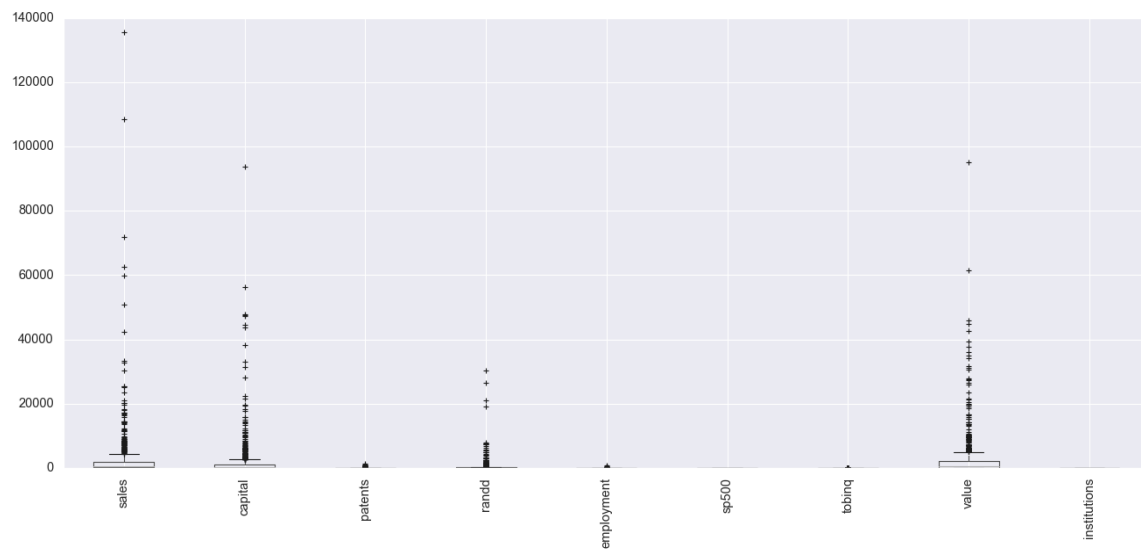
The NULL values present in the variable 'tobinq' were imputed with the median. As median is the best value for imputation of NULL values.

	sales	capital	patents	randd	employment	sp500	tobinq	value	institutions
0	826.995050	161.603986	10	382.078247	2.306000	0	11.049511	1625.453755	80.27
1	407.753973	122.101012	2	0.000000	1.860000	0	0.844187	243.117082	59.02
2	8407.845588	6221.144614	138	3296.700439	49.659005	1	5.205257	25865.233800	47.70
3	451.000010	266.899987	1	83.540161	3.071000	0	0.305221	63.024630	26.88
4	174.927981	140.124004	2	14.233637	1.947000	0	1.063300	67.406408	49.46

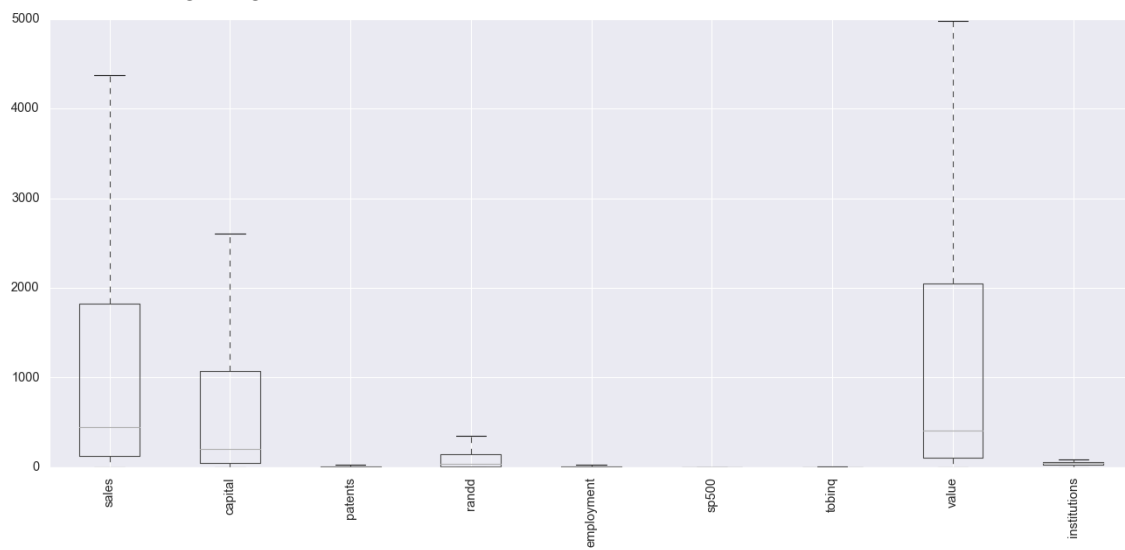
The scaling is necessary in this case because the dataset has different attributes with different magnitudes. Also, after model building it was observed that the scaled data has performed better than non-scaled data.

## Outlier Treatment

```
sales      100
capital    121
patents    122
randd      114
employment 103
sp500      0
tobinq     67
value      96
institutions 0
dtype: int64
```



We shall be treating the outliers by imputing them with the standard technique of imputing with upper quantile and lower quantile limits. The upper value is calculated by  $Q3 + (1.5 * IQR)$  & lower value is calculated by  $Q1 - (1.5 * IQR)$ . After imputation the data looks like the following image.



**1.3) Encode the data (having string values) for Modelling. Data Split: Split the data into test and train (70:30). Apply Linear regression. Performance Metrics: Check the performance of Predictions on Train and Test sets using R-square, RMSE.**

**Iteration-1**

The data was split into train and test with ratio 70:30 using scikit learn library `train_test_split` function.

```
LinearRegression()
```

Linear Regression was applied to the training data.

```
The coefficient for capital is 0.40695504390920023
The coefficient for patents is -4.672408588719046
The coefficient for randd is 0.6568588577973116
The coefficient for employment is 73.16442024702089
The coefficient for sp500 is 182.39778737538043
The coefficient for tobing is -41.79421805956599
The coefficient for value is 0.2442051292102374
The coefficient for institutions is 0.5694371405664665
```

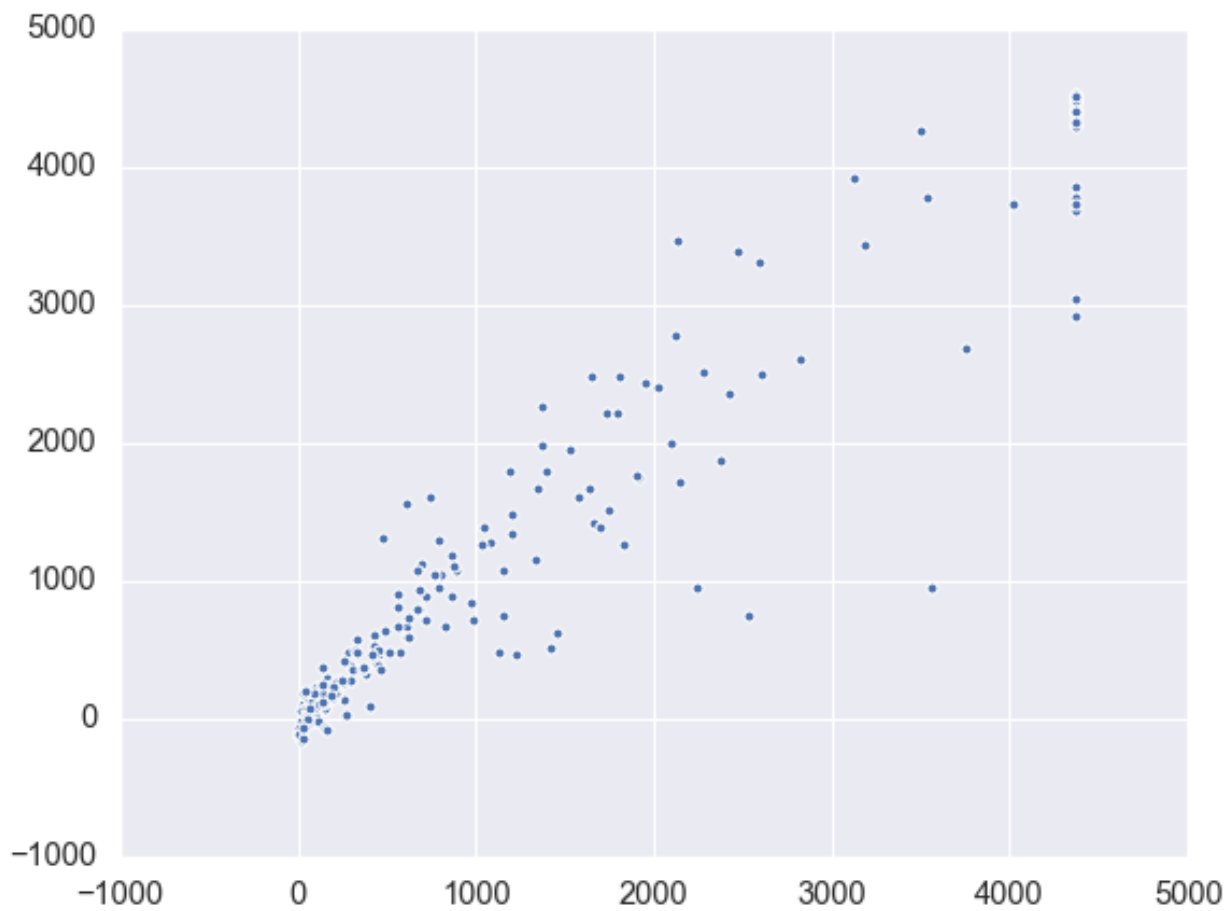
Above are coefficients for all the variables.

```
The intercept for our model is 89.59971563179124
```

We can see that the intercept for the model is very high.

The R square and RMSE scores were calculated on the train and test data.

Parameter	Train	Test
R Square	0.9357156436246878	0.9244167412828321
RMSE	395.10352669386776	398.96673210964946



```

=====
                        OLS Regression Results
=====
Dep. Variable:          sales      R-squared:          0.936
Model:                  OLS        Adj. R-squared:       0.935
Method:                 Least Squares  F-statistic:        949.8
Date:                   Fri, 14 Jan 2022  Prob (F-statistic):    2.05e-305
Time:                   14:04:19    Log-Likelihood:     -3928.4
No. Observations:       531        AIC:                7875.
Df Residuals:           522        BIC:                7913.
Df Model:                8
Covariance Type:        nonrobust
=====
                        coef      std err      t      P>|t|      [0.025      0.975]
-----
Intercept              89.5997      44.391      2.018      0.044      2.394      176.806
capital                 0.4070       0.042      9.733      0.000      0.325      0.489
patents                -4.6724       2.788     -1.676      0.094     -10.149      0.804
randd                  0.6569       0.233      2.819      0.005       0.199      1.115
employment             73.1644       4.463     16.394      0.000     64.397     81.932
sp500                 182.3978      66.372      2.748      0.006     52.009     312.786
tobinq                -41.7942      11.343     -3.685      0.000     -64.077     -19.512
value                  0.2442       0.025      9.597      0.000       0.194      0.294
institutions           0.5694       0.902      0.632      0.528     -1.202      2.341
=====
Omnibus:                203.580    Durbin-Watson:       1.950
Prob(Omnibus):          0.000    Jarque-Bera (JB):    1444.298
Skew:                   1.503    Prob(JB):            0.00
Kurtosis:               10.500    Cond. No.            9.77e+03
=====

```

Notes:

- [1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
- [2] The condition number is large, 9.77e+03. This might indicate that there are strong multicollinearity or other numerical problems.

## Iteration-2

In iteration-2 the train and test data was scaled using zscore library.

`LinearRegression()`

Linear Regression was applied to the training data.

The coefficient for capital is 0.25562996533467913

The coefficient for patents is -0.030288836518586663

The coefficient for randd is 0.05366356278813676

The coefficient for employment is 0.413206639209281

The coefficient for sp500 is 0.053112632061698334

The coefficient for tobing is -0.048317963335499745

The coefficient for value is 0.2752477926572372

The coefficient for institutions is 0.007933921954478595

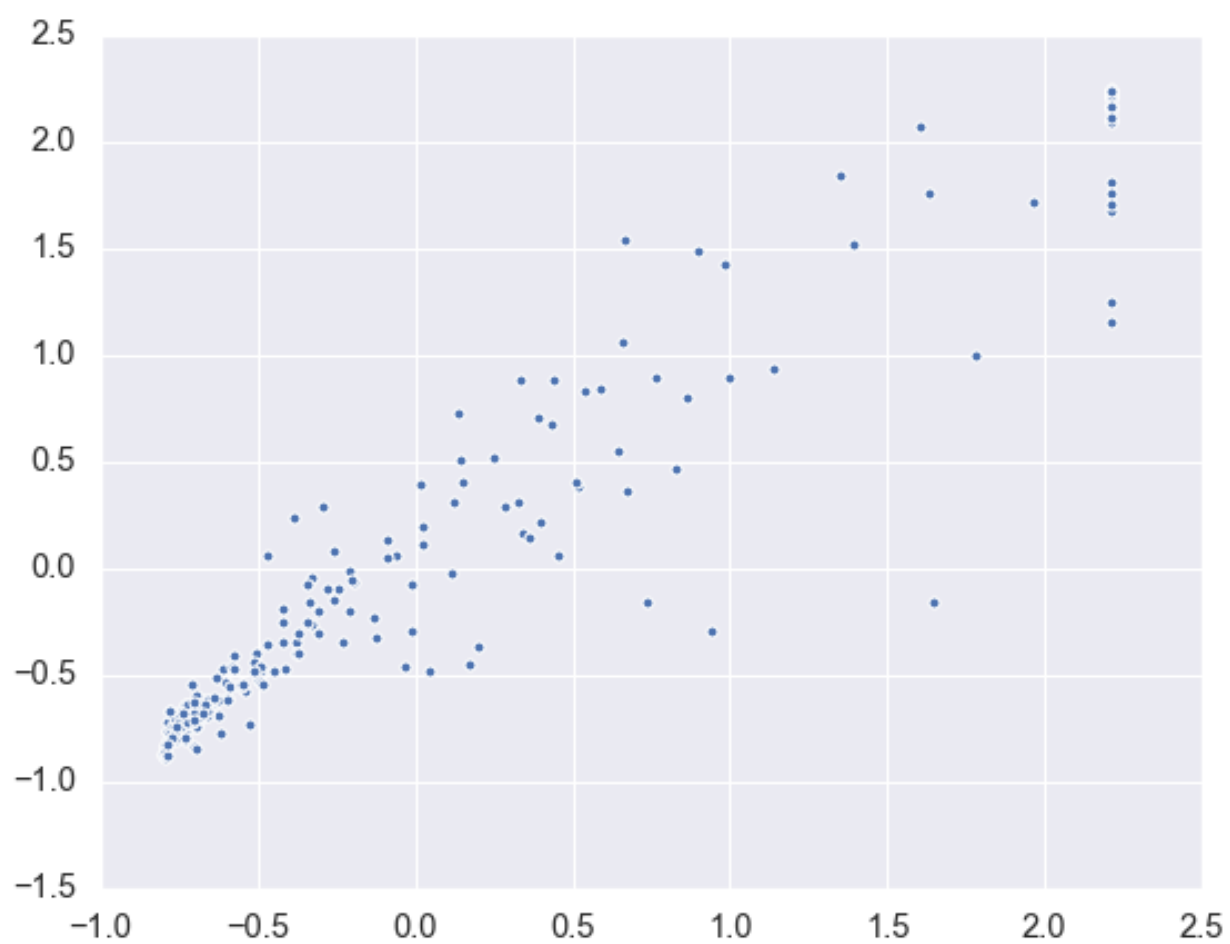
Above are coefficients for all the variables.

The intercept for our model is 4.0119512621673827e-17

We can see that the intercept for the model is very low almost close to 0.

The R square and RMSE scores were calculated on the train and test data.

Parameter	Train	Test
R Square	0.9357156436246878	0.9256872194275775
RMSE	0.2535435985689881	0.2726037060871009



#### **1.4) Inference: Based on these predictions, what are the business insights and recommendations?**

We have built the Linear Regression twice in this case study. First model was built without scaling the data and second model was built after scaling the data.

We can see that the RMSE values for train and test are less in the second case. Also, the intercept of the model in the first case is very high i.e. 86.59, which shows it has high influence on the model.

However, in the second case the intercept of the model is very less i.e. close to zero.

Based on the analysis, following are the business insights-

- Companies with higher number of employees have higher sales
- Companies with higher capital and R&D stock have better sales revenue
- Companies having prestigious Membership of firms in the S&P 500 index, have higher sales compared to those who aren't the member.

Therefore, the recommendation to management shall be:

- Go ahead with the second method of building the Linear Regression model.
- The management should prefer investing in companies which have
  - Higher employees number
  - R & D stock
  - Membership of S&P 500 index.
- The management shall have less risk with these companies as the prediction on the historical data suggests good sales revenue for them in future.

## **Problem 2: Logistic Regression and Linear Discriminant Analysis**

You are hired by the Government to do an analysis of car crashes. You are provided details of car crashes, among which some people survived and some didn't. You have to help the government in predicting whether a person will survive or not on the basis of the information given in the data set so as to provide insights that will help the government to make stronger laws for car manufacturers to ensure safety measures.

Also, find out the important factors on the basis of which you made your predictions.

Data Dictionary:

1. dvcat: factor with levels (estimated impact speeds) 1-9km/h, 10-24, 25-39, 40-54, 55+
2. weight: Observation weights, albeit of uncertain accuracy, designed to account for varying sampling probabilities. (The inverse probability weighting estimator can be used to demonstrate causality when the researcher cannot conduct a controlled experiment but has observed data to model)
3. Survived: factor with levels Survived or not\_survived
4. airbag: a factor with levels none or airbag
5. seatbelt: a factor with levels none or belted
6. frontal: a numeric vector; 0 = non-frontal, 1=frontal impact
7. sex: a factor with levels f: Female or m: Male
8. ageOFocc: age of occupant in years
9. yearacc: year of accident
10. yearVeh: Year of model of vehicle; a numeric vector
11. abcat: Did one or more (driver or passenger) airbag(s) deploy? This factor has levels deploy, nodeploy and unavail
12. occRole: a factor with levels driver or pass: passenger
13. deploy: a numeric vector: 0 if an airbag was unavailable or did not deploy; 1 if one or more bags deployed.
14. injSeverity: a numeric vector; 0: none, 1: possible injury, 2: no incapacity, 3: incapacity, 4: killed; 5: unknown, 6: prior death
15. caseid: character, created by pasting together the populations sampling unit, the case number, and the vehicle number. Within each year, use this to uniquely identify the vehicle.

Questions for Problem 2:

- a) Data Ingestion: Read the dataset. Do the descriptive statistics and do null value condition check, write an inference on it. Perform Univariate and Bivariate Analysis. Do exploratory data analysis.**

We received the data in the CSV format with file named as Car\_Crash.csv.

The data was uploaded using standard pandas library.

```
Index(['Unnamed: 0', 'dvcat', 'weight', 'Survived', 'airbag', 'seatbelt',  
      'frontal', 'sex', 'ageOFocc', 'yearacc', 'yearVeh', 'abcat', 'occRole',  
      'deploy', 'injSeverity', 'caseid'],  
      dtype='object')
```

Above are the columns present in the dataset.



	Unnamed: 0	dvcat	weight	Survived	airbag	seatbelt	frontal	sex	ageOFocc	yearacc	yearVeh	abcat	occRole	deploy	injSeverity	caseid
0	0	55+	27.078	Not_Survived	none	none	1	m	32	1997	1987.0	unavail	driver	0	4.0	2:13:2
1	1	25-39	89.627	Not_Survived	airbag	belted	0	f	54	1997	1994.0	nodeploy	driver	0	4.0	2:17:1
2	2	55+	27.078	Not_Survived	none	belted	1	m	67	1997	1992.0	unavail	driver	0	4.0	2:79:1
3	3	55+	27.078	Not_Survived	none	belted	1	f	64	1997	1992.0	unavail	pass	0	4.0	2:79:1
4	4	55+	13.374	Not_Survived	none	none	1	m	23	1997	1986.0	unavail	driver	0	4.0	4:58:1

Above picture shows the initial 5 data points in the dataset. Here, we can see that the column 'Unnamed: 0' is useless and doesn't have any relevant information. Also, column 'caseid' has unique values which are used to uniquely identify the vehicle Therefore, it was dropped.

	dvcat	weight	Survived	airbag	seatbelt	frontal	sex	ageOFocc	yearacc	yearVeh	abcat	occRole	deploy	injSeverity
0	55+	27.078	Not_Survived	none	none	1	m	32	1997	1987.0	unavail	driver	0	4.0
1	25-39	89.627	Not_Survived	airbag	belted	0	f	54	1997	1994.0	nodeploy	driver	0	4.0
2	55+	27.078	Not_Survived	none	belted	1	m	67	1997	1992.0	unavail	driver	0	4.0
3	55+	27.078	Not_Survived	none	belted	1	f	64	1997	1992.0	unavail	pass	0	4.0
4	55+	13.374	Not_Survived	none	none	1	m	23	1997	1986.0	unavail	driver	0	4.0

Above table shows the first 5 data points of the dataset after dropping the columns.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11217 entries, 0 to 11216
Data columns (total 14 columns):
#   Column                Non-Null Count  Dtype
---  -
0   dvcat                  11217 non-null  object
1   weight                 11217 non-null  float64
2   Survived               11217 non-null  object
3   airbag                 11217 non-null  object
4   seatbelt               11217 non-null  object
5   frontal                11217 non-null  int64
6   sex                   11217 non-null  object
7   ageOFocc               11217 non-null  int64
8   yearacc                11217 non-null  int64
9   yearVeh                11217 non-null  float64
10  abcat                  11217 non-null  object
11  occRole                11217 non-null  object
12  deploy                 11217 non-null  int64
13  injSeverity            11140 non-null  float64
dtypes: float64(3), int64(4), object(7)
memory usage: 1.2+ MB
```

From the above figure we can see that the datatypes of all the columns. Most of the variables doesn't have outliers present except column 'injSeverity'.

```
survived          10037
Not_Survived      1180
Name: Survived, dtype: int64
```

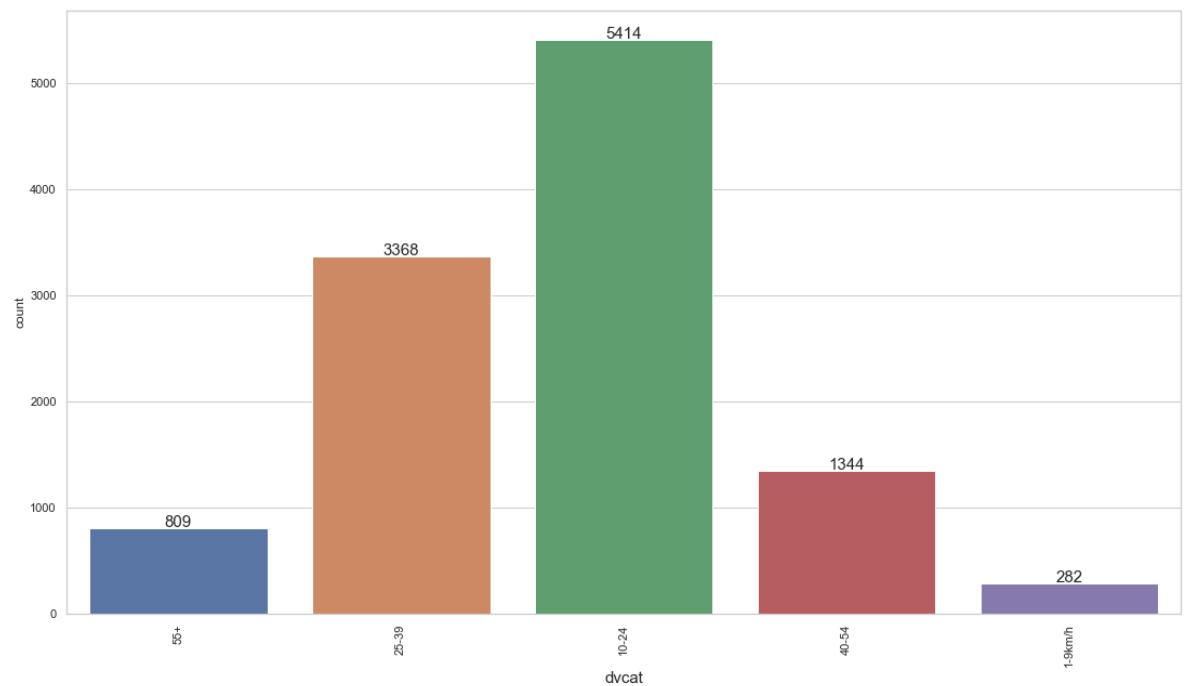
The percentage of people surviving is 89.48  
The percentage of people not surviving is 10.52

‘Survived’ is the target column here which is a dependent variable. The percentage of the people not surviving is very less as compared to the survived. Therefore, we can say that the data is imbalanced.

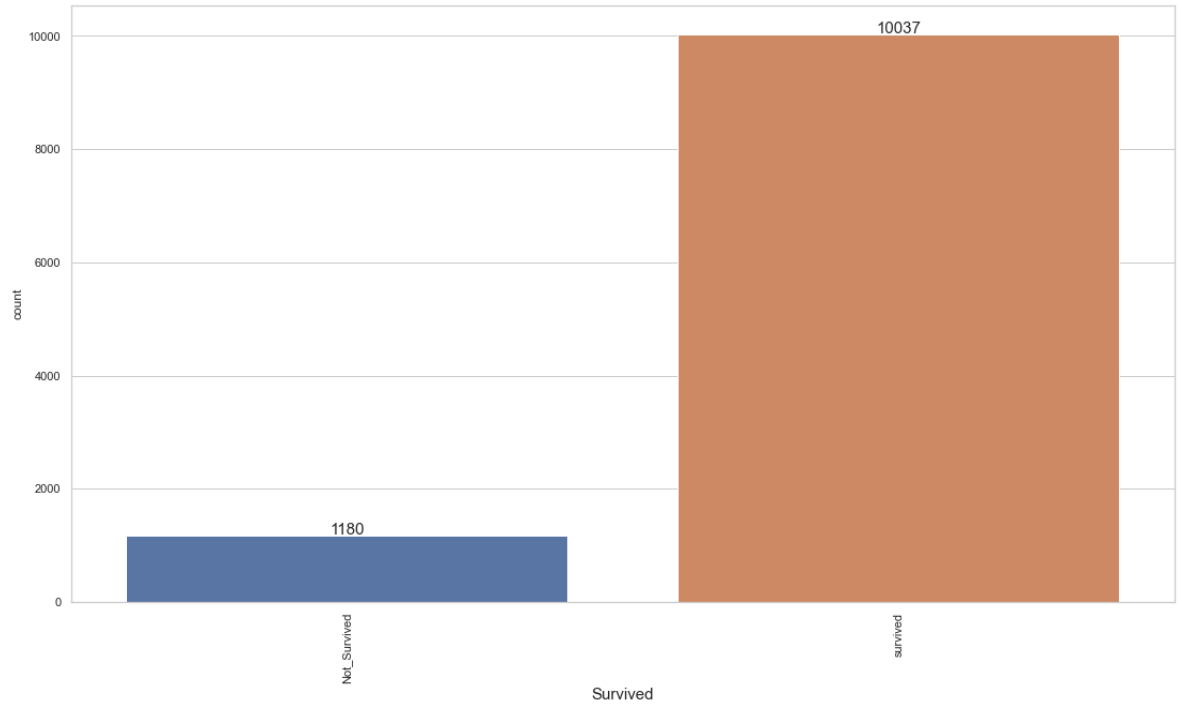
	count	mean	std	min	25%	50%	75%	max
weight	11217.0	431.405309	1406.202941	0.0	28.292	82.195	324.056	31694.04
frontal	11217.0	0.644022	0.478830	0.0	0.000	1.000	1.000	1.00
ageOfOcc	11217.0	37.427654	18.192429	16.0	22.000	33.000	48.000	97.00
yearacc	11217.0	2001.103236	1.056805	1997.0	2001.000	2001.000	2002.000	2002.00
yearVeh	11217.0	1994.177944	5.658704	1953.0	1991.000	1995.000	1999.000	2003.00
deploy	11217.0	0.389141	0.487577	0.0	0.000	0.000	1.000	1.00
injSeverity	11140.0	1.825583	1.378535	0.0	1.000	2.000	3.000	5.00

Above is the descriptive statistics for the dataset. The average weight for the cars is 431.40. The average age of occupant 37.42.

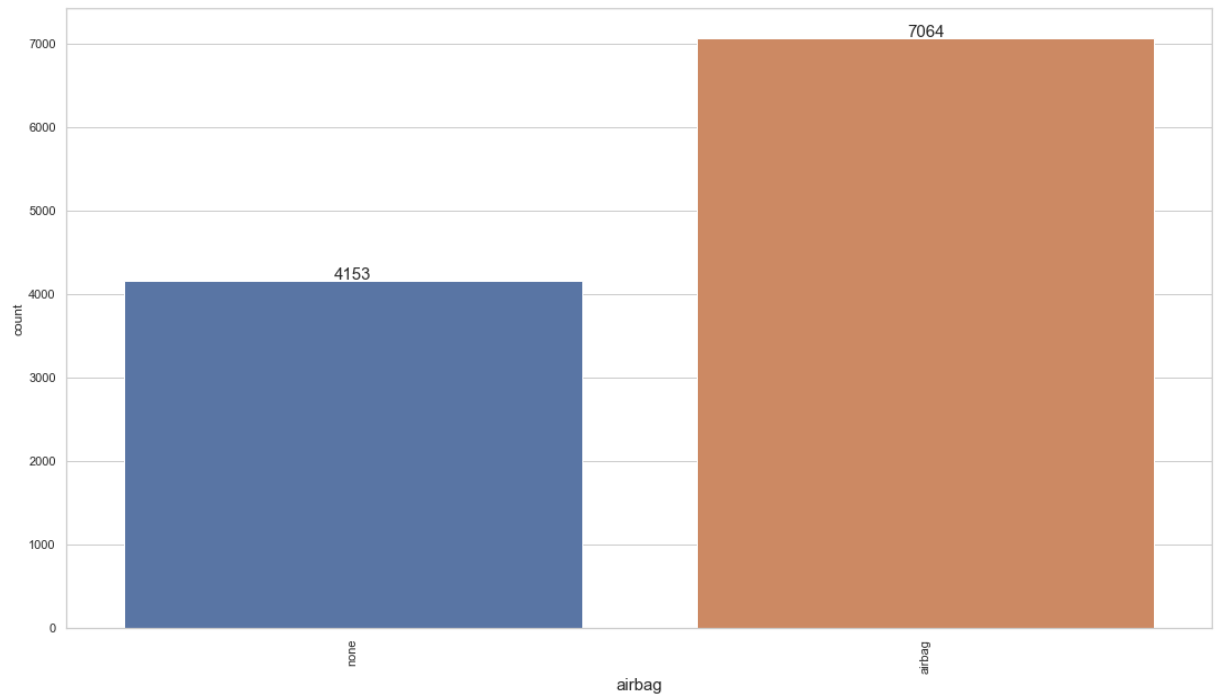
### Univariate Analysis



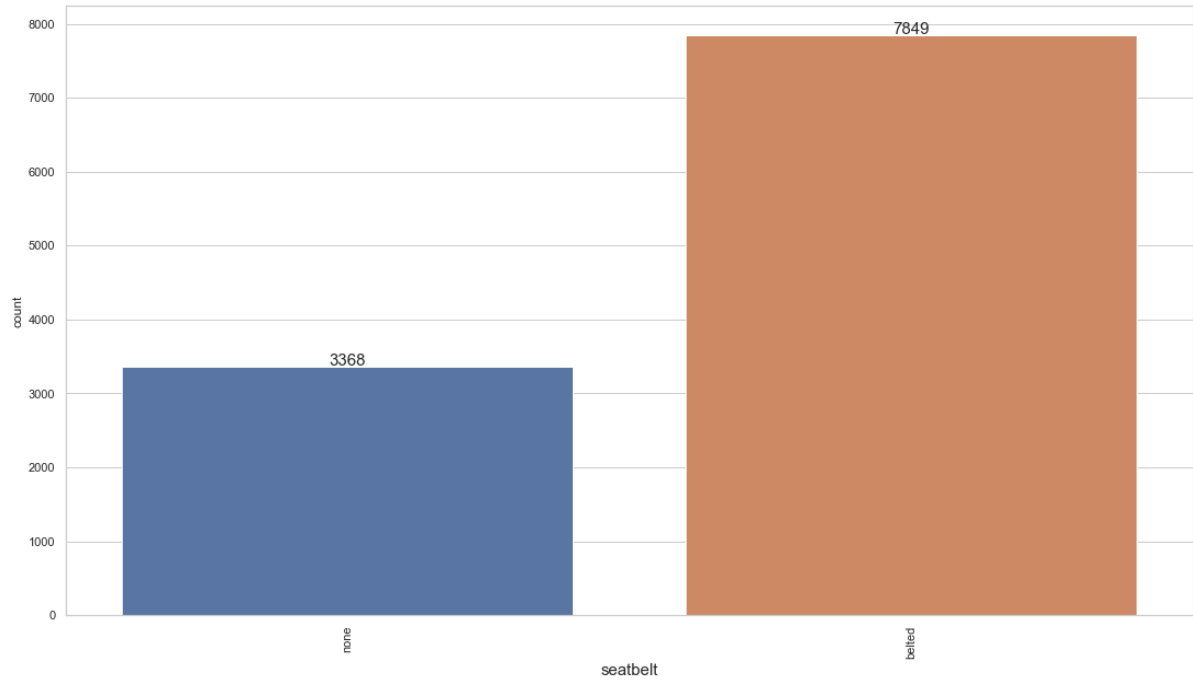
From the above graph we can understand that most of accidents have impact speed 10-24 km/ hour followed by 25-39 km/ hour.



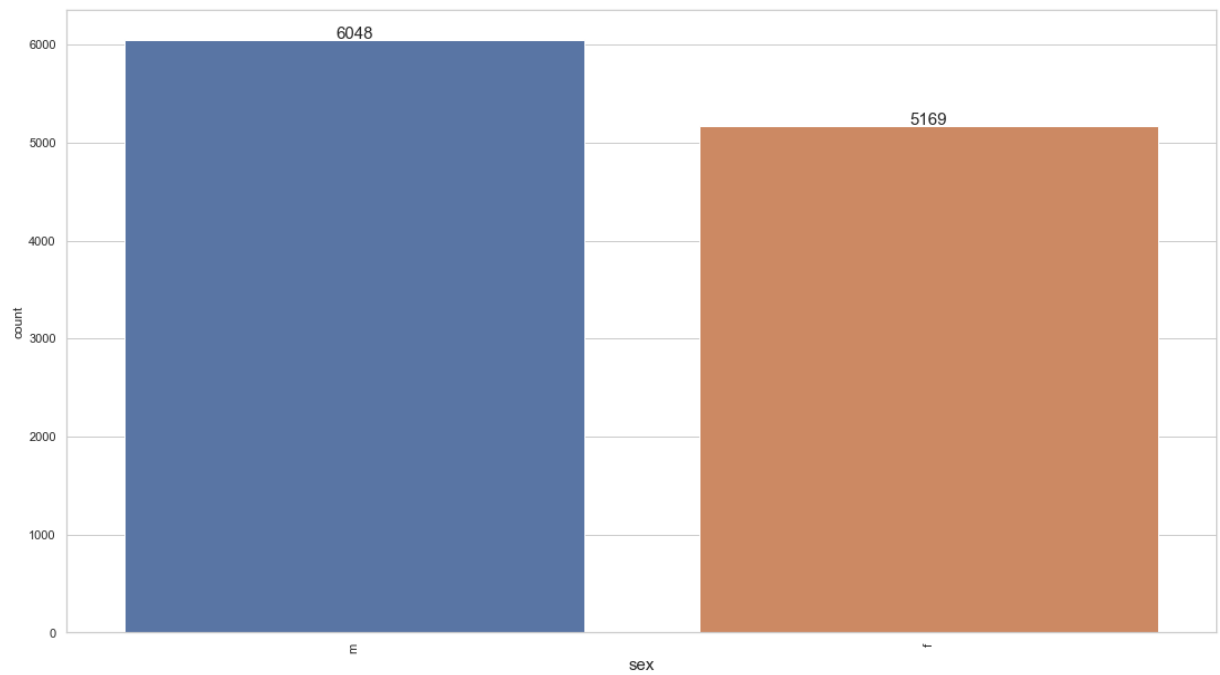
From the above graph we can understand that in 10.51% of the cases the driver/passenger couldn't survive. Also, this is our target dependent variable. So we can say that the data is imbalanced.



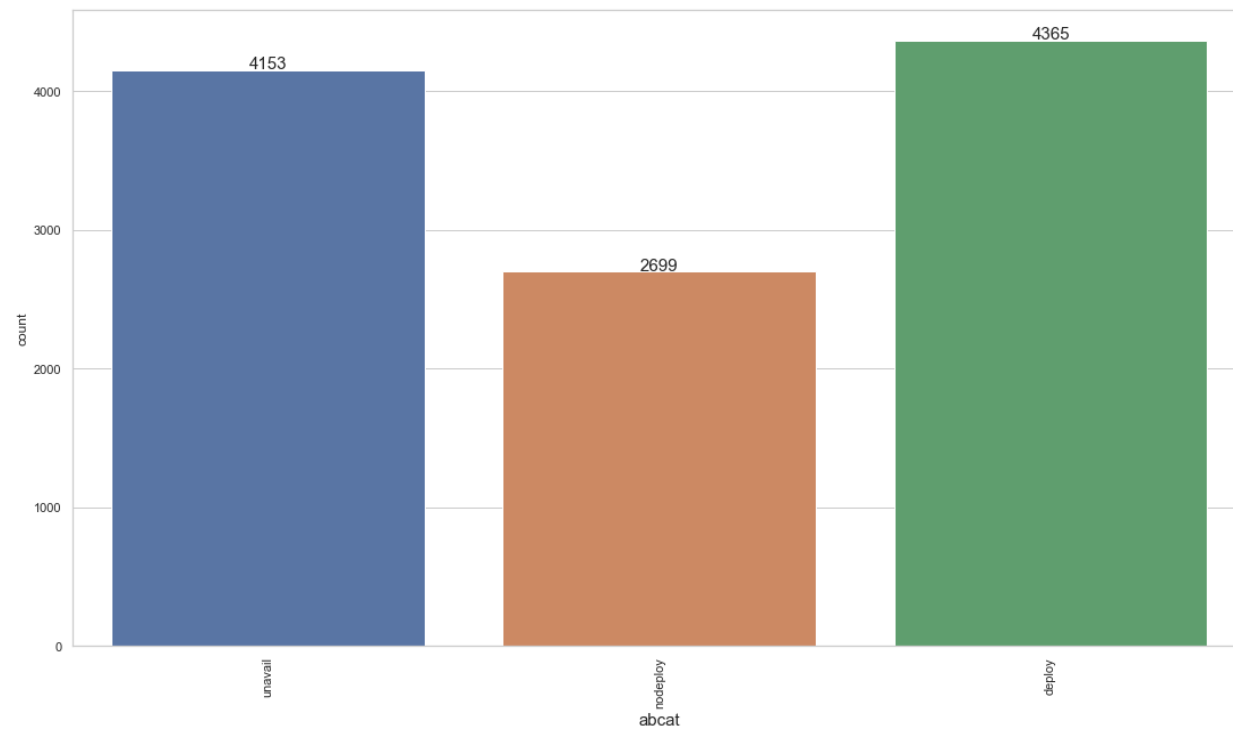
The above graph represents that the vehicles with airbags are more than the vehicles without airbag.



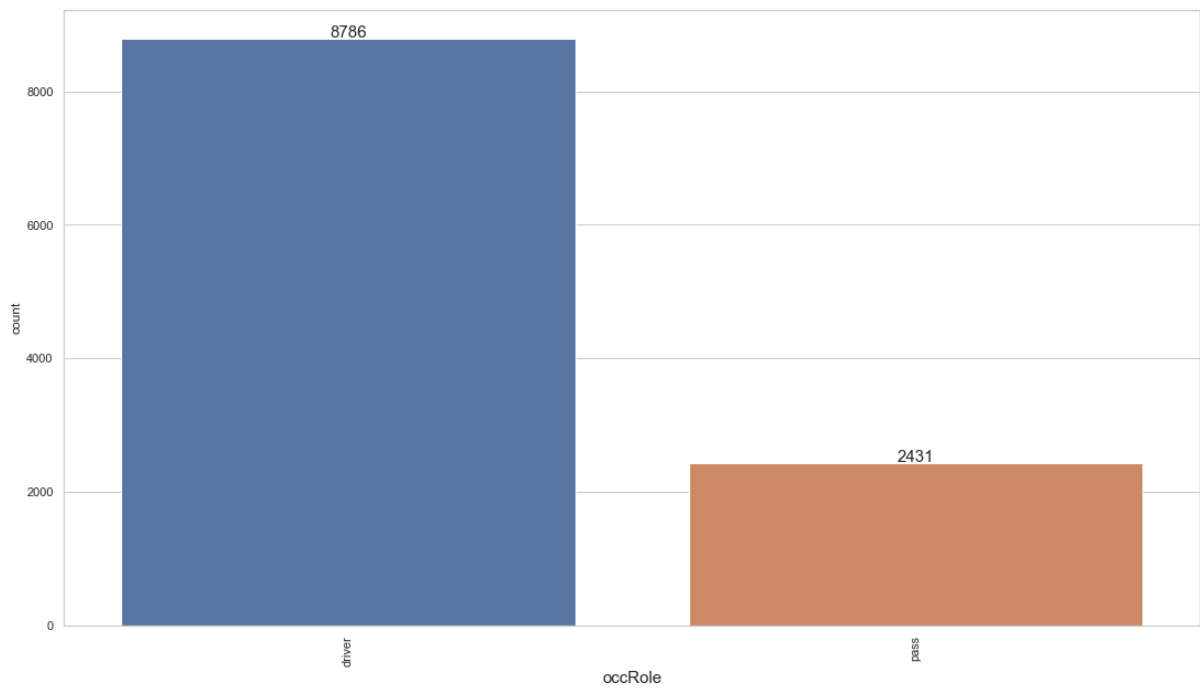
According to above graph, the number of people not wearing the seatbelt is very high. However, it is less than the people wearing the seatbelt.



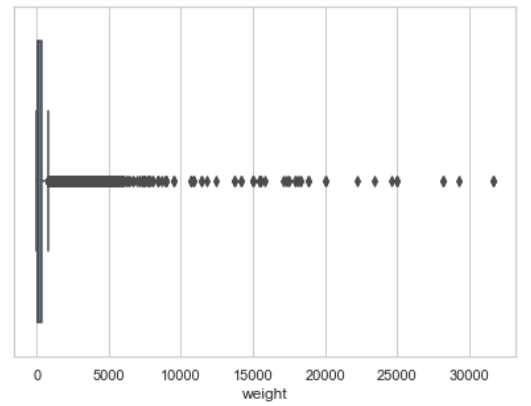
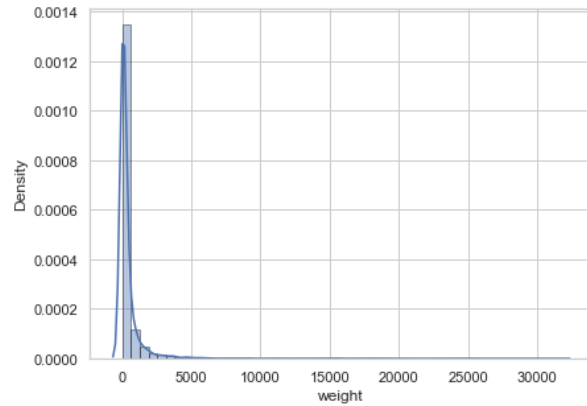
According to above graph, the males are slightly higher in number than females who came across these car crash accidents.



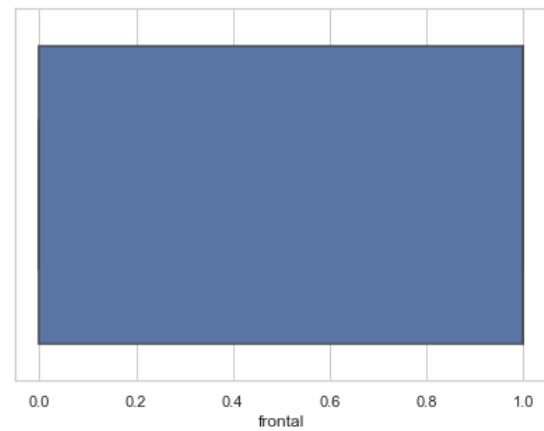
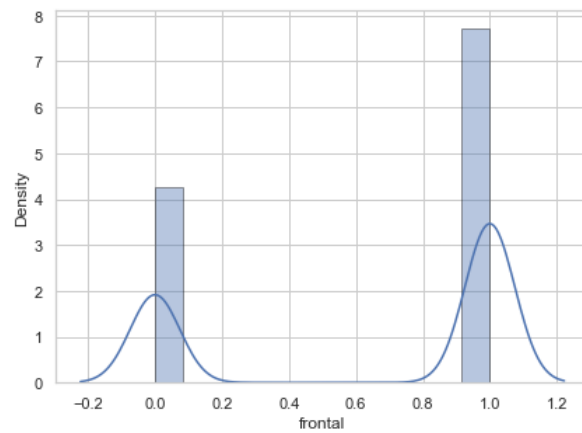
According to above graph, among the passengers who had already airbag safety feature present in their car, there is a significant number of cases where airbag did not deploy. In almost 38% of the cases, the airbag was present but didn't deploy.



According to above graph, the accidental impact happened mostly on the driver. In 21% cases the impact happened on the passenger.

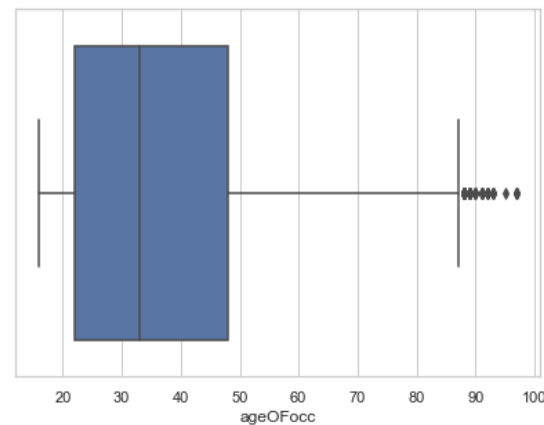
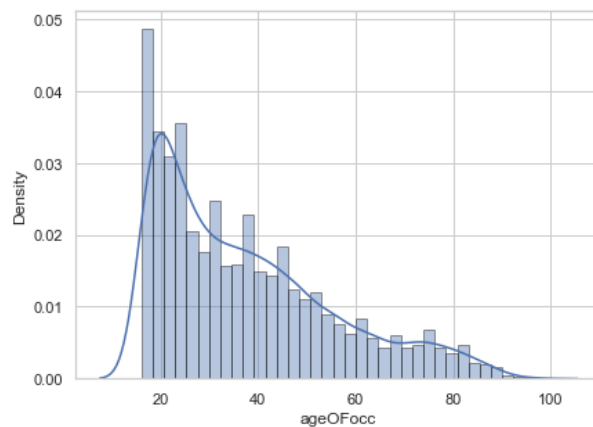


From above graph, we can see that the variable weight has high number of outliers present. Also, the data is somehow normally distributed.

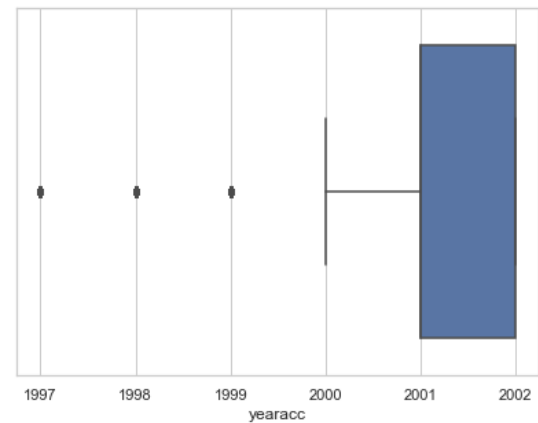
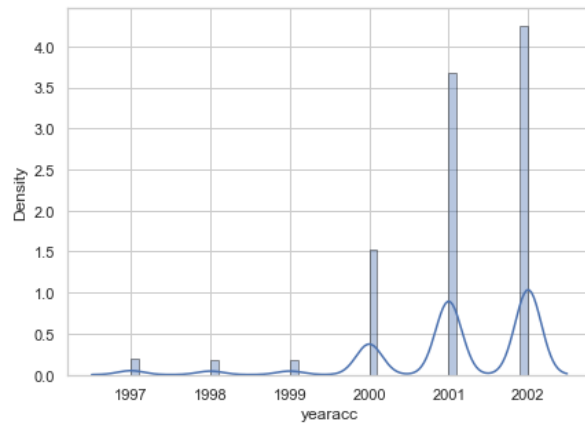


```
1    7224
0    3993
Name: frontal, dtype: int64
```

Frontal is a categorical variable. We can see that in the majority of the cases the impact happened from front side.



According to the graph above, we can see that the age of occupant variable has data normally distributed with right skewness. Also, there is presence of significant number of outliers in the data.

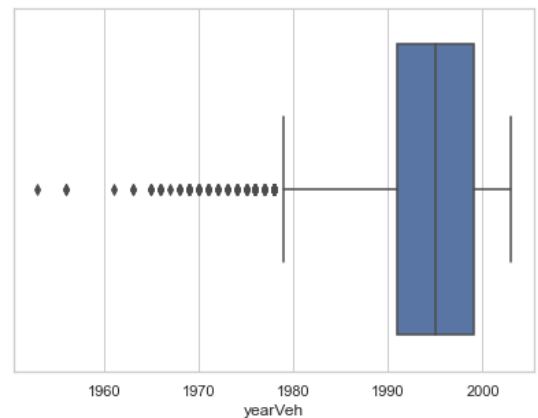
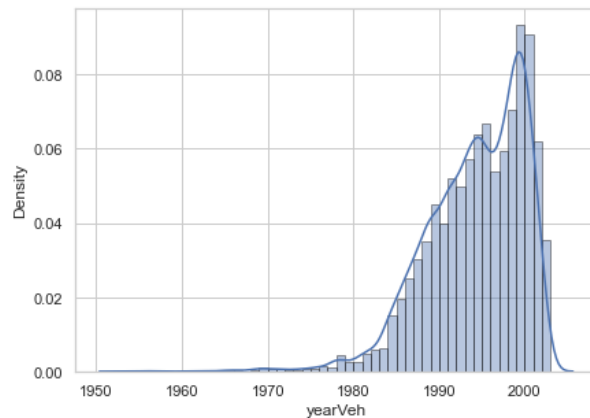


```

2002    4764
2001    4115
2000    1716
1997     224
1999     200
1998     198
Name: yearacc, dtype: int64

```

Year of accident is actually a categorical variable. We can see that the year 2002 has highest number of accidents reported which is followed by 2001, 2000, 1997, 1999 and 1998 respectively.



According to the above graph, the year of model of vehicle data is uniformly distributed with left skewness. Also, there is significant number of outliers also present.

```

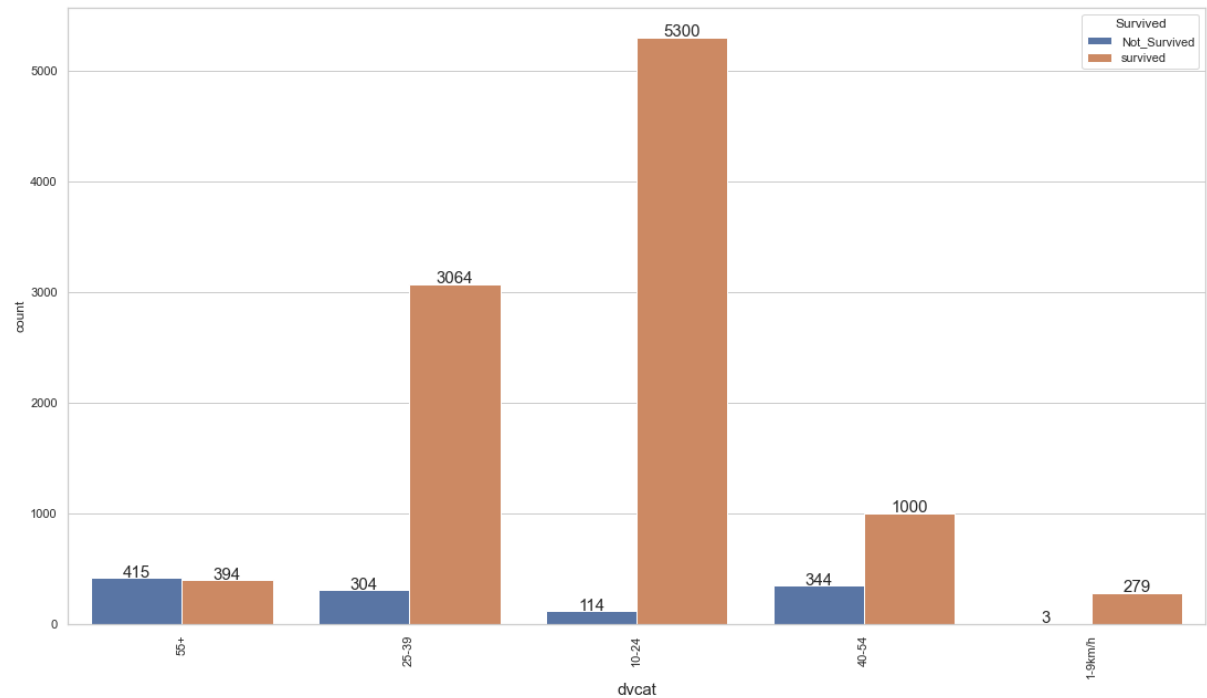
0    6852
1    4365
Name: deploy, dtype: int64

```

In most number of accidents, the airbag didn't deploy. In 61% of the cases the airbag was not present or didn't work during the accident.

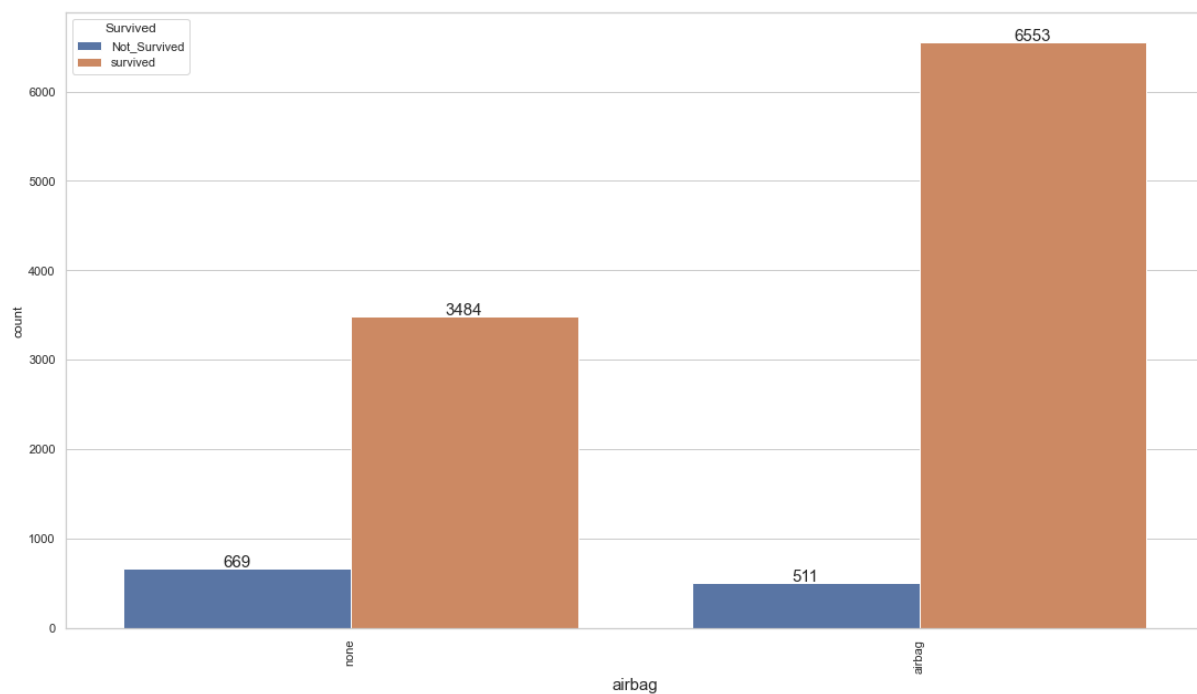
```
3.0    3337
0.0    2734
1.0    2218
2.0    1682
4.0    1101
5.0      68
Name: injSeverity, dtype: int64
```

From the above figure, we can understand that the injury severity. Most number of cases reported incapacity.

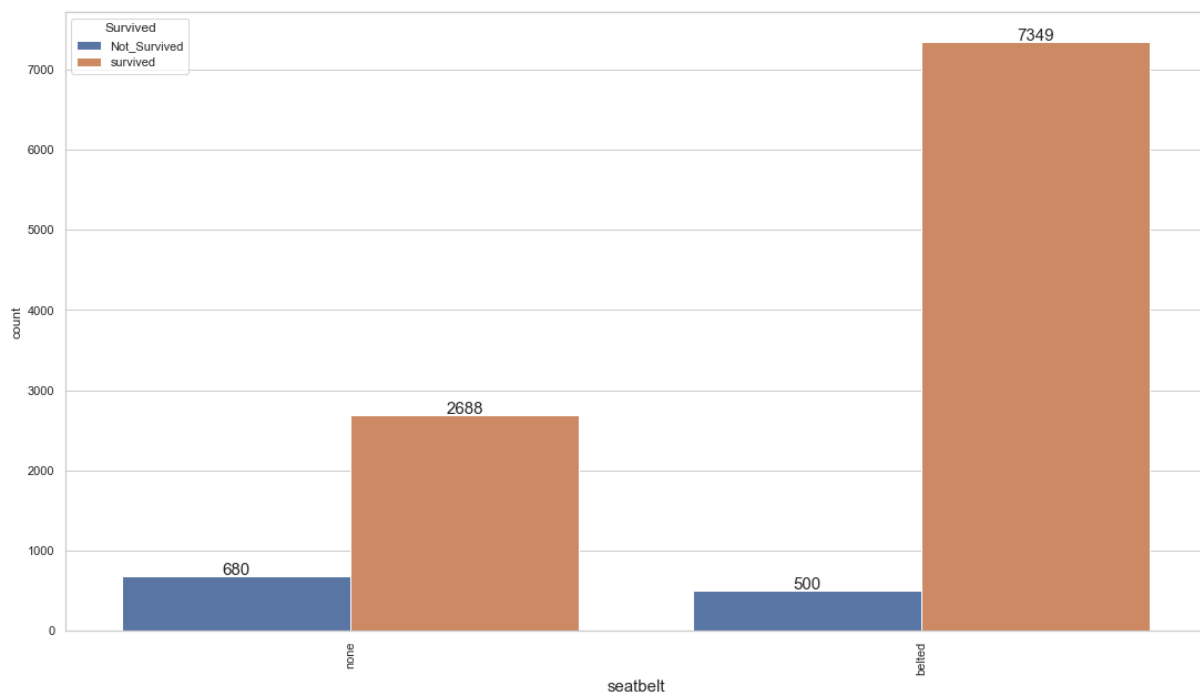


From the above graph we can clearly understand that most number of accidental deaths happened due to vehicle's high speed.

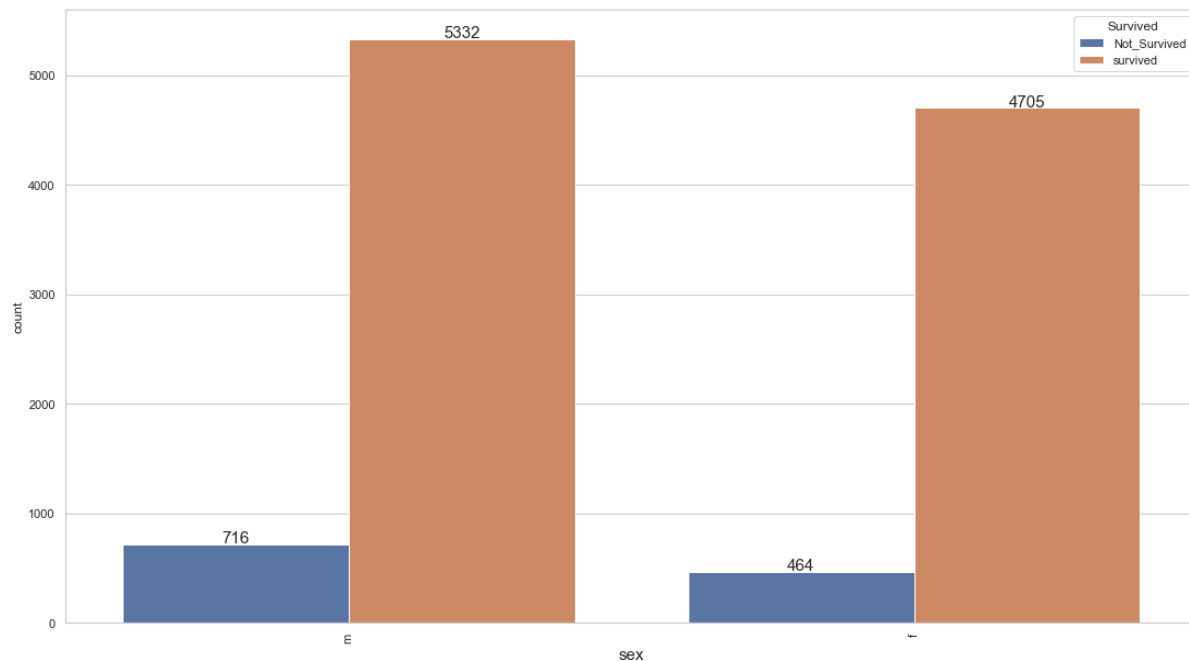




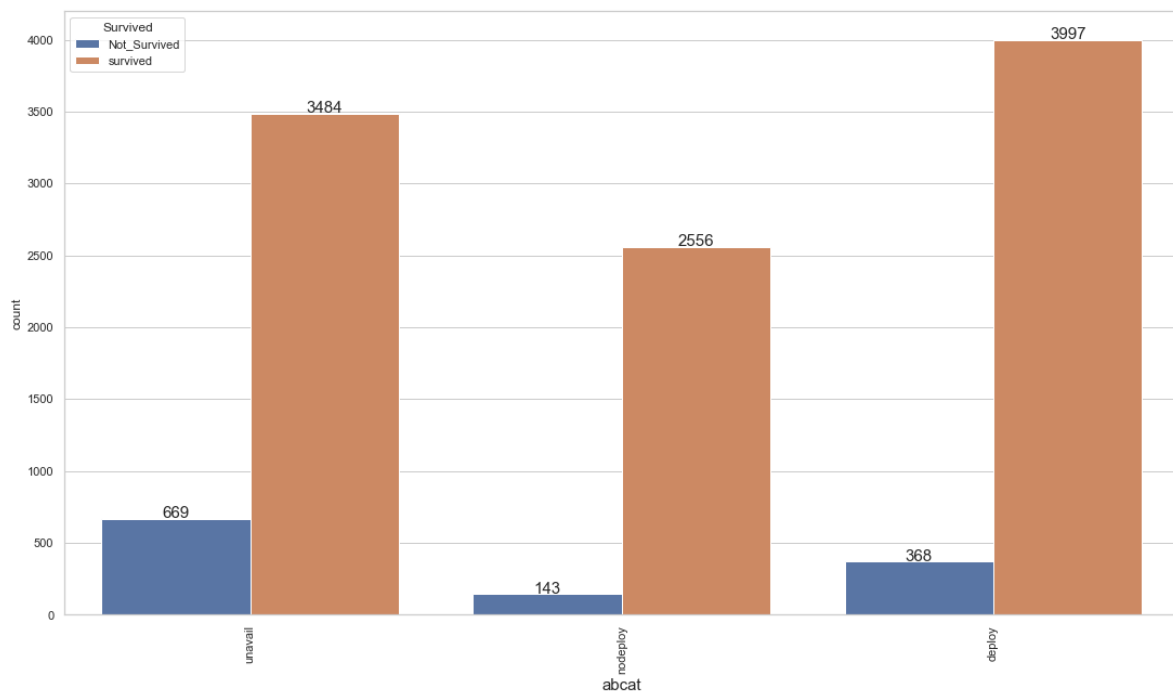
From the above graph, we can see that the most number of people have survived in case where airbag was present.



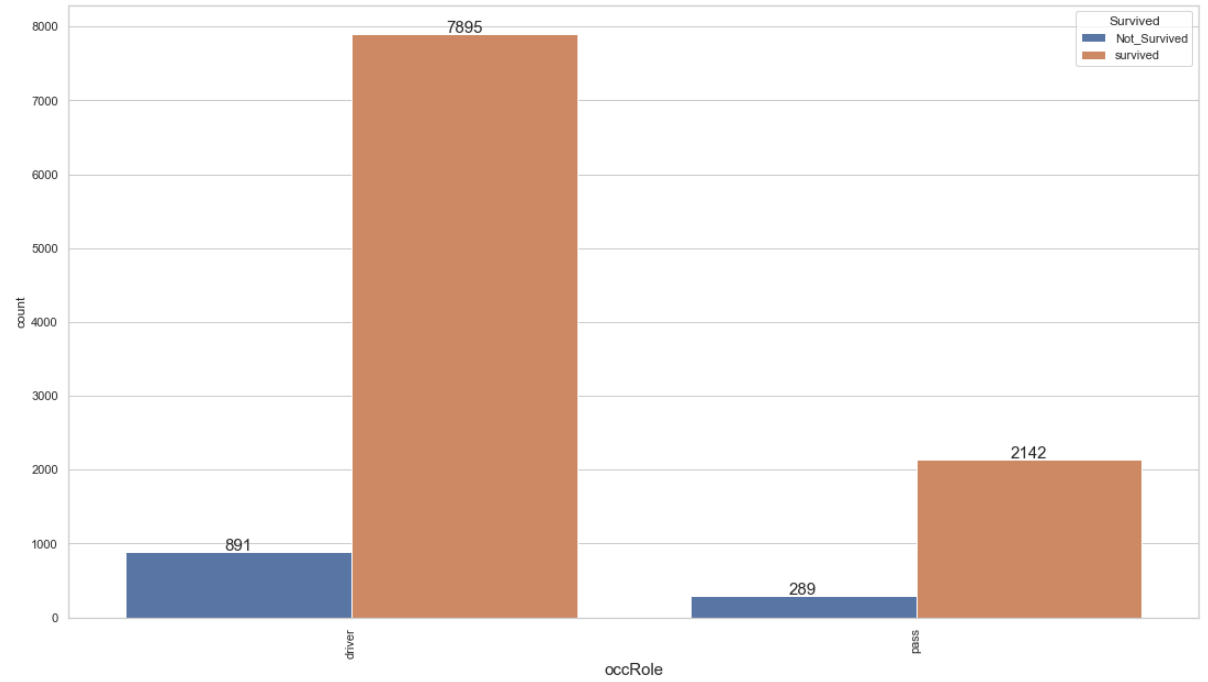
From the above graph, we can see that the most number of people have survived in case where seat belt was used by the driver while driving.



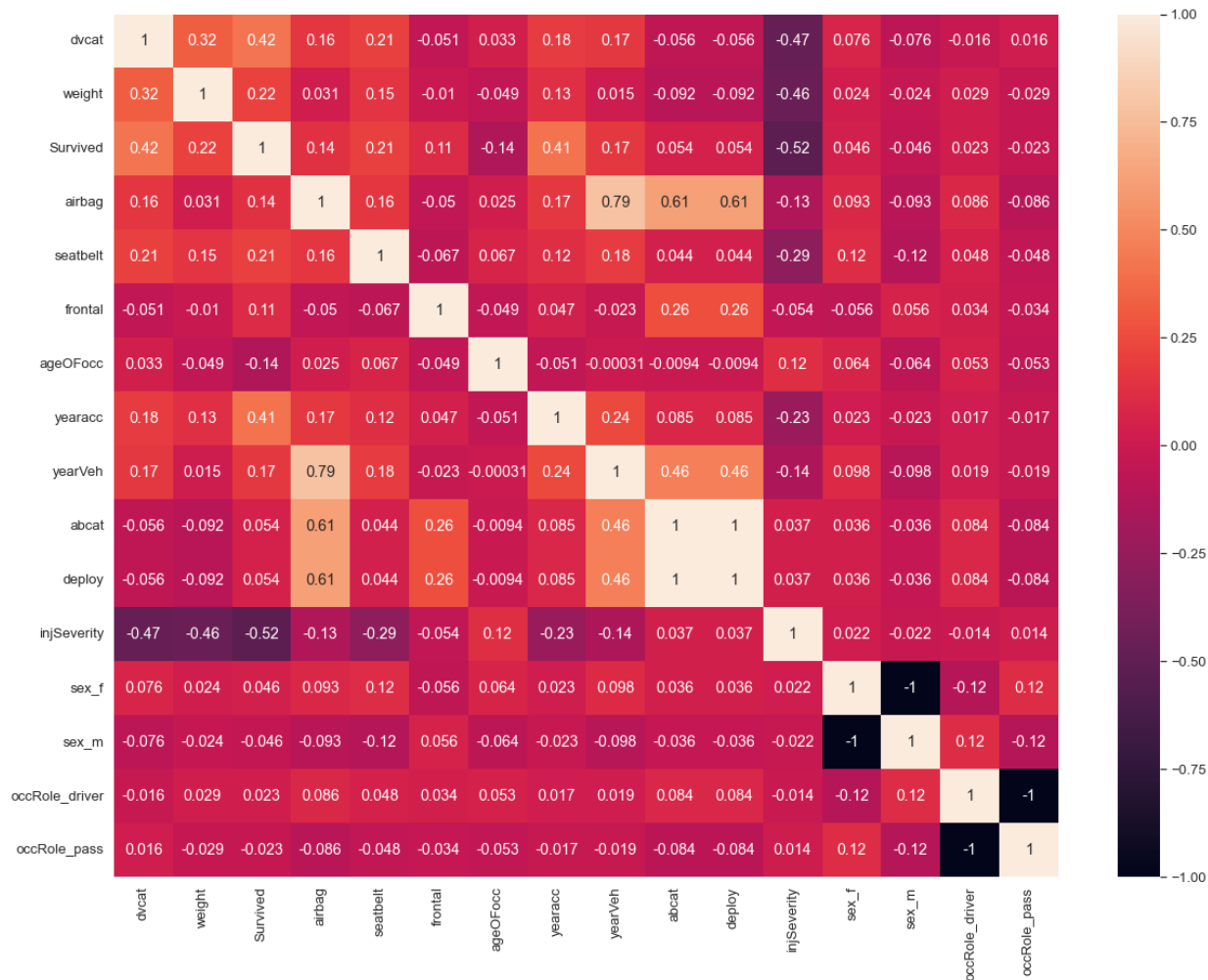
The number of males who couldn't survive the accident is more than the number of females.



Above graph shows that the higher number of driver/ passengers have survived the accident where the airbag deployed.



Among the people who didn't survive the accident most were on the driver's seat.



- We can a correlation of 0.79 between year of vehicle and the airbag.

### Null Value Treatment

```

dvcat      0
weight     0
Survived   0
airbag     0
seatbelt   0
frontal    0
ageOfOcc   0
yearacc    0
yearVeh    0
abcat      0
deploy     0
injSeverity 77
sex_f      0
sex_m      0
occRole_driver 0
occRole_pass 0
dtype: int64

```

We can see that the variable 'injSeverity' has outliers present in it.

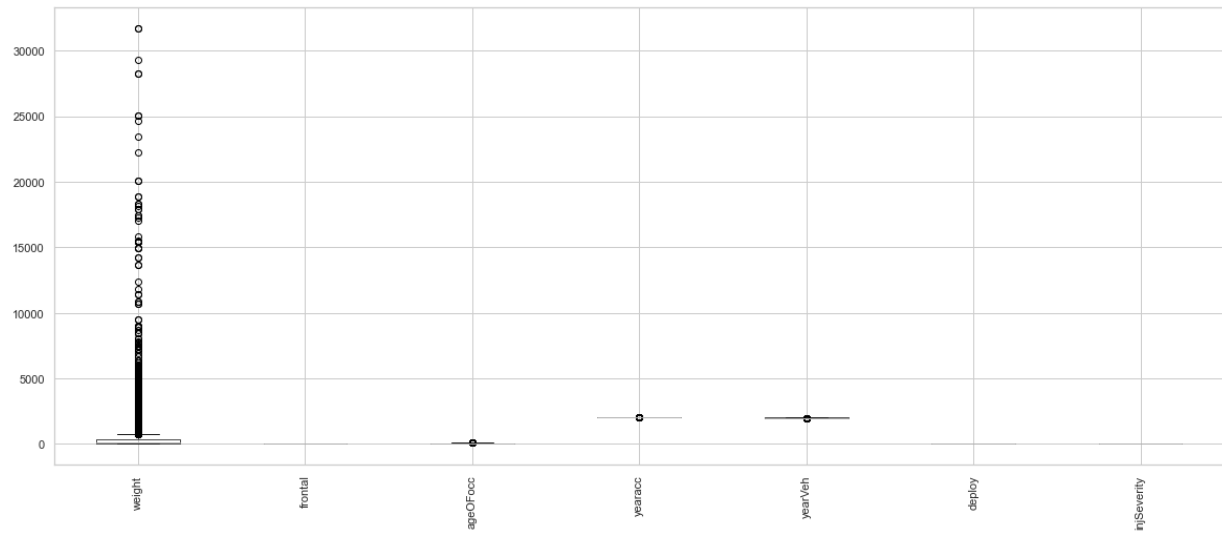
As 'injSeverity' is a categorical variable therefore it was imputed using the median.

```
dvcat      0
weight     0
Survived   0
airbag     0
seatbelt   0
frontal    0
ageOFocc   0
yearacc    0
yearVeh    0
abcat      0
deploy     0
injSeverity 0
sex_f      0
sex_m      0
occRole_driver 0
occRole_pass 0
dtype: int64
```

After NULL value treatment, we can see that there is no presence of outlier in the dataset.

### **Outlier Treatment**

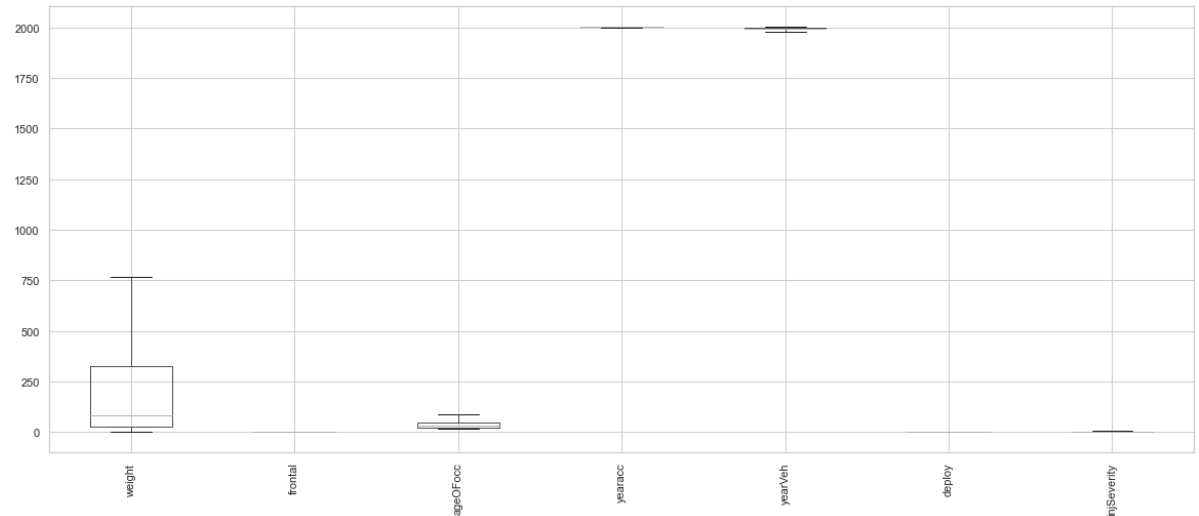
```
Survived      0
abcat          0
ageOFocc      68
airbag        0
deploy        0
dvcat         0
frontal       0
injSeverity   0
occRole       0
seatbelt      0
sex           0
weight       1412
yearVeh      147
yearacc      622
dtype: int64
```



We can see the presence of outliers in the dataset.

We shall be treating the outliers by imputing them with the standard technique of imputing with upper quantile and lower quantile limits. The upper value is calculated by  $Q3 + (1.5 * IQR)$  & lower value is calculated by  $Q1 - (1.5 * IQR)$ . After imputation the data looks like the following image.

```
Survived      0
abcat         0
ageOfOcc      0
airbag        0
deploy        0
dvcat         0
frontal       0
injSeverity   0
occRole       0
seatbelt      0
sex           0
weight        0
yearVeh       0
yearacc       0
dtype: int64
```



The outlier treatment was done & we can see that there's no presence of outlier in the dataset.

**b) Encode the data (having string values) for Modelling. Data Split: Split the data into train and test (70:30). Apply Logistic Regression and LDA (linear discriminant analysis).**

	dvcat	weight	Survived	airbag	seatbelt	frontal	sex	ageOfOcc	yearacc	yearVeh	abcat	occRole	deploy	injSeverity
0	55+	27.078	Not_Survived	none	none	1.0	m	32.0	1999.5	1987.0	unavail	driver	0.0	4.0
1	25-39	89.627	Not_Survived	airbag	belted	0.0	f	54.0	1999.5	1994.0	nodeploy	driver	0.0	4.0
2	55+	27.078	Not_Survived	none	belted	1.0	m	67.0	1999.5	1992.0	unavail	driver	0.0	4.0
3	55+	27.078	Not_Survived	none	belted	1.0	f	64.0	1999.5	1992.0	unavail	pass	0.0	4.0
4	55+	13.374	Not_Survived	none	none	1.0	m	23.0	1999.5	1986.0	unavail	driver	0.0	4.0

Following variables require data encoding-

a. Dvcat

```

10-24      5414
25-39      3368
40-54      1344
55+         809
1-9km/h    282
Name: dvcat, dtype: int64

```

Before encoding the data points were representing the speed limits. They were encoded from 4 to 0 in the decreasing order of vehicle speed. The probability of survival is high with less speed and low with more speed.

```

3      5414
2      3368
1      1344
0       809
4       282
Name: dvcat, dtype: int64

```

b. Survived

It is the target variable. 'Not\_Survived' was encoded to 0 and 'Survived' was encoded to 1

c. Airbag

'none' was encoded to 0 and 'airbag' was encoded to 1.

d. Seatbelt

'none' was encoded to 0 and 'belted' was encoded to 1

e. Sex

Here one hot encoding was applied. Therefore, new columns for male and female with 0 as no and 1 as yes were created and the original column was dropped.

f. Abcat

'deploy' was encoded to 1 and others were encoded to 0

g. occRole

Here one hot encoding was applied. Therefore, new columns for passenger and driver with 0 as no and 1 as yes were created and the original column was dropped.

	dvcat	weight	Survived	airbag	seatbelt	frontal	ageOFocc	yearacc	yearVeh	abcat	deploy	injSeverity	sex_f	sex_m	occRole_driver	occRole_pass
0	0	27.078	0	0	0	1.0	32.0	1999.5	1987.0	0	0.0	4.0	0	1	1	0
1	2	89.627	0	1	1	0.0	54.0	1999.5	1994.0	0	0.0	4.0	1	0	1	0
2	0	27.078	0	0	1	1.0	67.0	1999.5	1992.0	0	0.0	4.0	0	1	1	0
3	0	27.078	0	0	1	1.0	64.0	1999.5	1992.0	0	0.0	4.0	1	0	0	1
4	0	13.374	0	0	0	1.0	23.0	1999.5	1986.0	0	0.0	4.0	0	1	1	0

Above figure shows the head of the dataset after the encoding.

### **Train-Test Split & Model Building**

The data is imbalanced therefore it was first scaled using standard scalar. Then the SMOTE function was applied to balance the dataset. Then, the data was split into train and test set with 70:30 ratio.

Logistic Regression and Linear Discriminant Analysis were imported using sklearn library and was applied on the training dataset and then checked on the test data.

```
LogisticRegression()
```

```
LinearDiscriminantAnalysis()
```



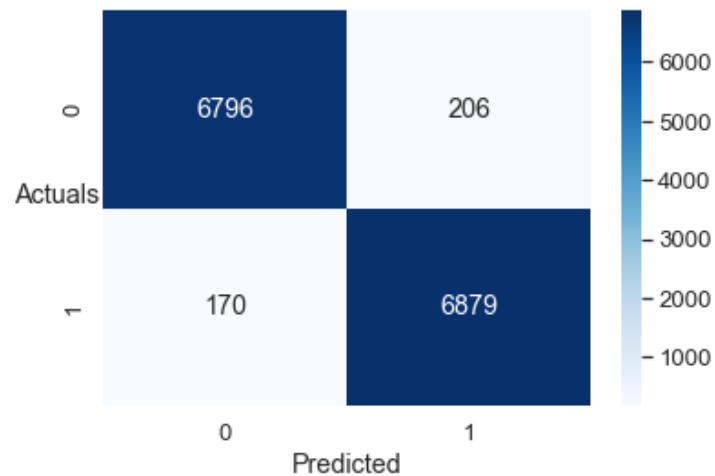
- c) **Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC\_AUC score for each model. Compare both the models and write inferences, which model is best/optimized.**

### Logistic Regression Training Set

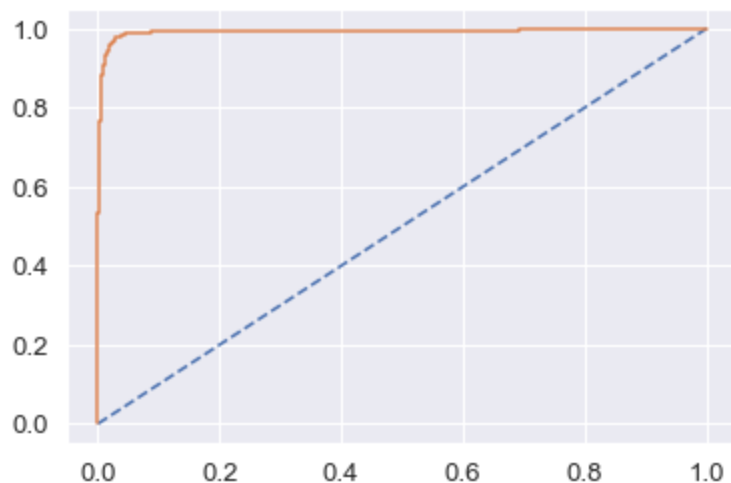
The model score for Logistic Regression Training set is 0.9732403387659241

The classification report & Confusion matrix for Logistic Regression training set is

	precision	recall	f1-score	support
0	0.98	0.97	0.97	7002
1	0.97	0.98	0.97	7049
accuracy			0.97	14051
macro avg	0.97	0.97	0.97	14051
weighted avg	0.97	0.97	0.97	14051



The AUC score for Logistic Regression Training dataset is: 0.9916

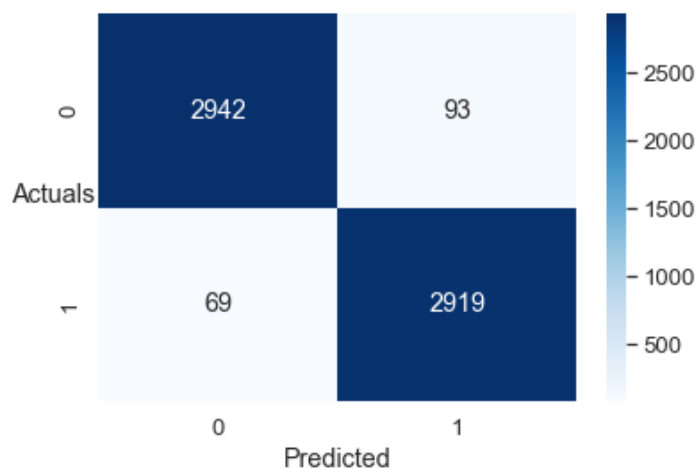


### Logistic Regression Testing Set

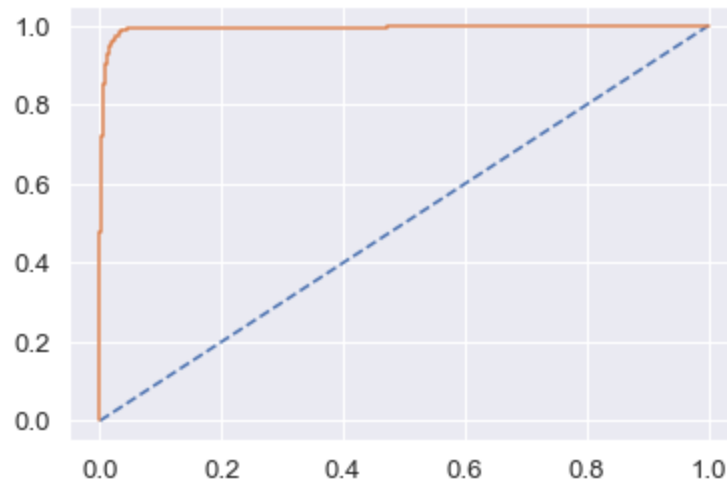
The model score for Logistic Regression Testing set is 0.9731031047650672

The Classification Report & Confusion Matrix for Logistic Regression testing set is

		precision	recall	f1-score	support
	0	0.98	0.97	0.97	3035
	1	0.97	0.98	0.97	2988
	accuracy			0.97	6023
	macro avg	0.97	0.97	0.97	6023
	weighted avg	0.97	0.97	0.97	6023



The AUC score for Logistic Regression testing set is: 0.9931

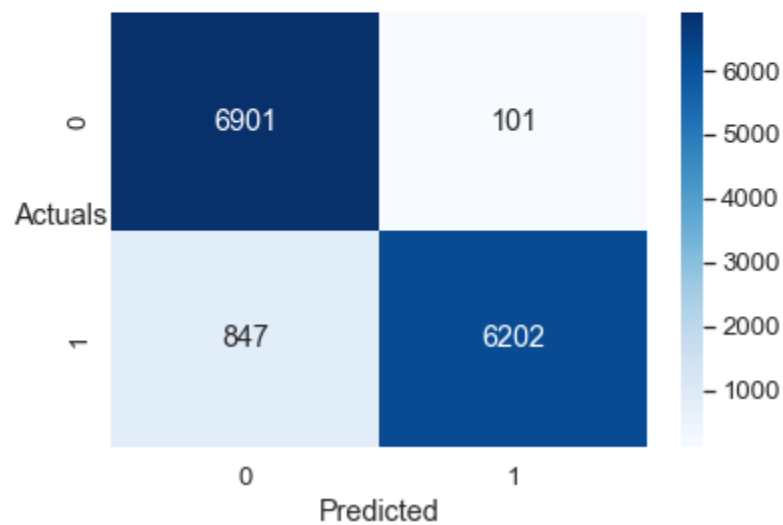


### Linear Discriminant Analysis Training Set

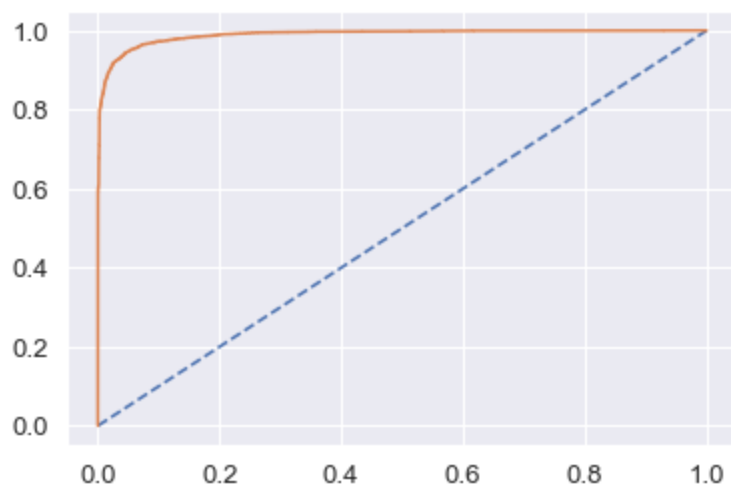
The model score for Linear Discriminant Analysis training set is 0.9325314924204683

The classification report for Linear Discriminant Analysis training set is

	precision	recall	f1-score	support
0	0.89	0.99	0.94	7002
1	0.98	0.88	0.93	7049
accuracy			0.93	14051
macro avg	0.94	0.93	0.93	14051
weighted avg	0.94	0.93	0.93	14051



The AUC score for Linear Discriminant Analysis training set is: 0.989

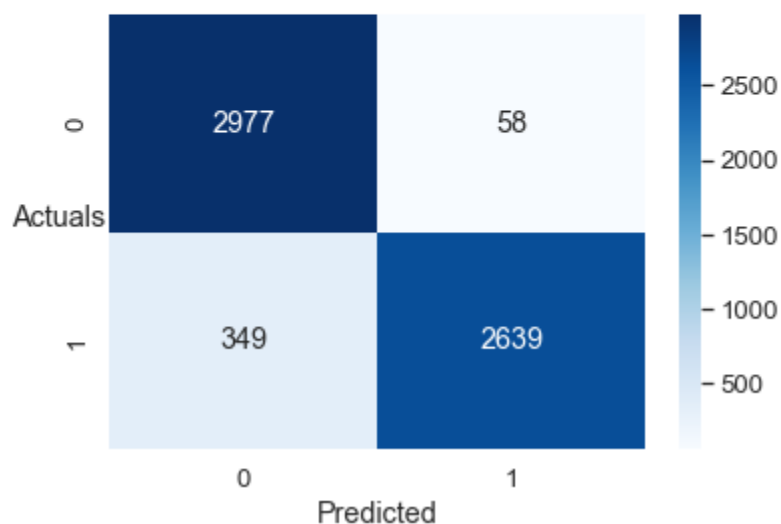


### Linear Discriminant Analysis Test Set

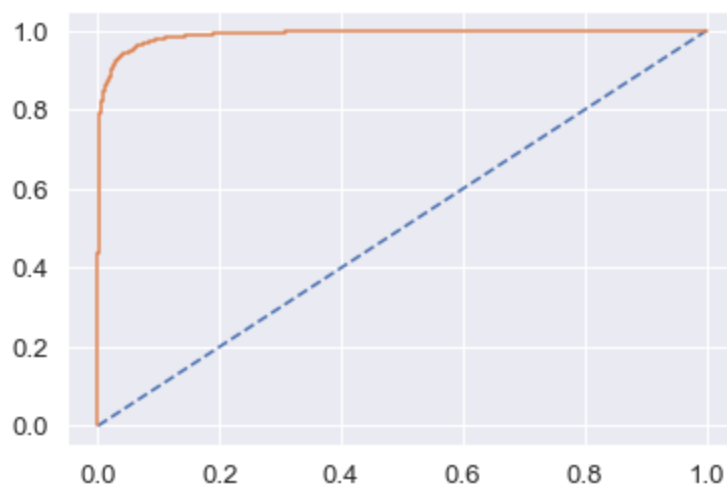
The model score for Linear Discriminant Analysis testing set is 0.9324257014776689

The classification report for Linear Discriminant Analysis testing set is

	precision	recall	f1-score	support
0	0.90	0.98	0.94	3035
1	0.98	0.88	0.93	2988
accuracy			0.93	6023
macro avg	0.94	0.93	0.93	6023
weighted avg	0.94	0.93	0.93	6023



The AUC score for Linear Discriminant Analysis testing set is: 0.990



	LR Train	LR Test	LDA Train	LDA Test
<b>Precision</b>	0.971	0.969	0.984	0.978
<b>Recall</b>	0.976	0.977	0.880	0.883
<b>F1 Score</b>	0.973	0.973	0.929	0.928
<b>Accuracy</b>	0.973	0.973	0.933	0.932
<b>AUC Score</b>	0.992	0.993	0.989	0.990

Above data frame has consolidated all the comparison parameters for both the models.

We can see considering parameters like accuracy, F1-Score on train and test data, the Logistic Regression model has better scores than the Logistic Discriminant Analysis.

In this case, we shall be comparing the models with F1-Score because the data is imbalanced.

Also, we can see that the number of False Negatives is very high in case of train dataset of Logistic Discriminant Analysis.

Therefore, the recommendation to the management shall be to consider Logistic Regression model while predicting the survived passenger/ driver in an accident.

**d) Inference: Based on these predictions, what are the insights and recommendations?**

The insights and recommendation to the Government are-

- For predicting whether the passenger or driver will survive the accident. The Government should rely on Logistic Regression model.
- It was observed that the major factors leading to death of the driver or passenger are high speed, unavailability of airbag, no use of seat belt and deployment of airbag.
- Government should start awareness program to aware public about the disadvantages of speeding. Fines for over speeding should be increased and made strict.
- Government should make every vehicle complaint with compulsory seat belt norm. If a driver does not use seat he/she should be unable to start the car.
- Government should also make airbags compulsory for all the automobile manufacturers and subsequently the buyers.
- Safety advertisements should include men more than women for awareness. Because data shows the number of more men casualties compared to women.