Project on Sales prediction using Machine Learning

- **Aim:-**To create a Data science Project, where we will be predicting Sales prediction means predicting how much of a product people will buy based on factors such as the amount you spend to advertise your product, the segment of people you advertise for, or the platform you are advertising on about your product.

  Steps to be taken in the project is sub-divided into the following sections. These are:
  - Importing the libraries such as 'numpy', 'pandas','sklearn. model' etc.
  - Loading Dataset as a CSV file for training & testing the models.
  - Splitting the data set into independent & dependent sets.
  - Checking if still any null values or any other data types other than float and integers are present into the dataset or not.
  - Importing the train_test_split model from sklearn.model for splitting data into train & test sets.
  - Applying the different kinds of ML Algorithms .which gives Best accuracy of model.
  - Also checking with new data set for predicting the values.
- Steps of creating ML model:-
- Importing numpy as np & pandas as pd for loading and reading the data-set & using matplotlib.pyplot and Seaborn for visualization of data.

```
[1]
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as snr
```

- Loading the csv-dataset in the variable name 'data' Then viewing the data with data.head()

```
data=pd.read_csv("/content/Advertising.csv")

[4] data.head()
```

	Unnamed: 0	TV	Radio	Newspaper	Sales
0	1	230.1	37.8	69.2	22.1
1	2	44.5	39.3	45.1	10.4
2	3	17.2	45.9	69.3	9.3
3	4	151.5	41.3	58.5	18.5
4	5	180.8	10.8	58.4	12.9

- Checking the data such as number of columns, rows and type of data(float,integer) with help of data.info()

```
[6] data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  200 non-null    int64
 1   TV          200 non-null    float64
 2   Radio       200 non-null    float64
 3   Newspaper   200 non-null    float64
 4   Sales       200 non-null    float64
dtypes: float64(4), int64(1)
memory usage: 7.9 KB
```

We observe that the above data have integer and float.

```
[8] data.shape

(200, 5)
```

Train data have 200 Rows and 5 columns

- Now checking data have Nan value or not.

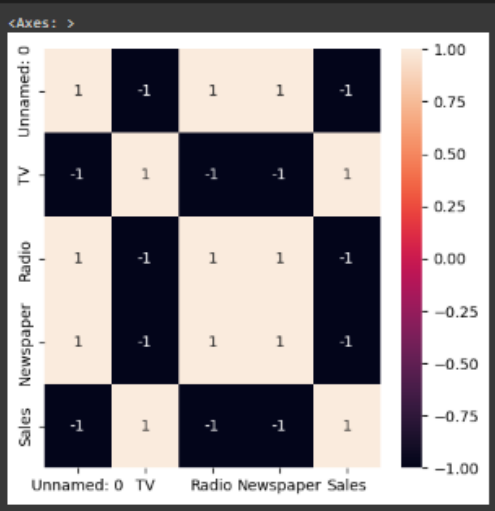
```
# total no of NAN values in dataset
data.isnull().sum().sum()

0

#missing values columns wise
data.isnull().sum(axis=0).sort_values()

                0
Unnamed: 0      0
TV              0
Radio           0
Newspaper       0
Sales           0
dtype: int64
```

We observe that the above data have not Nan value.

- Now we check the data dependency.

visualization of dataset

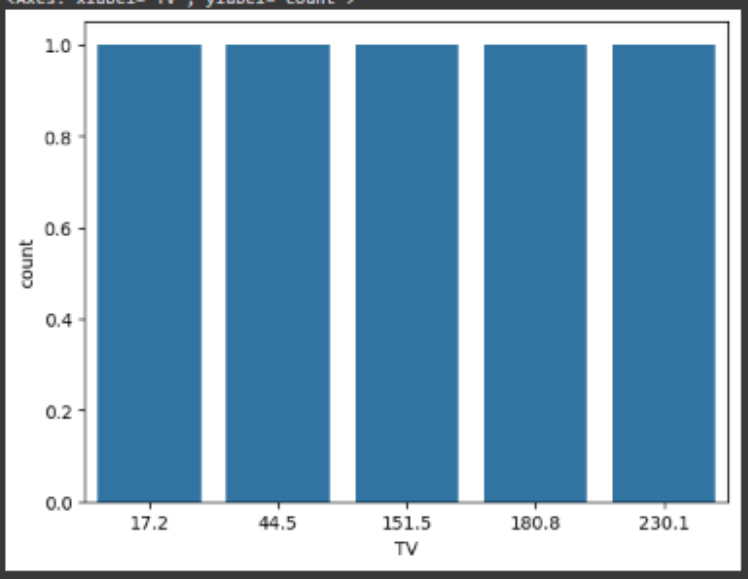
```
plt.figure(figsize=(5,5))
snr.heatmap(data[4:6].corr(),annot=True)
```



We see that data dependent each other.

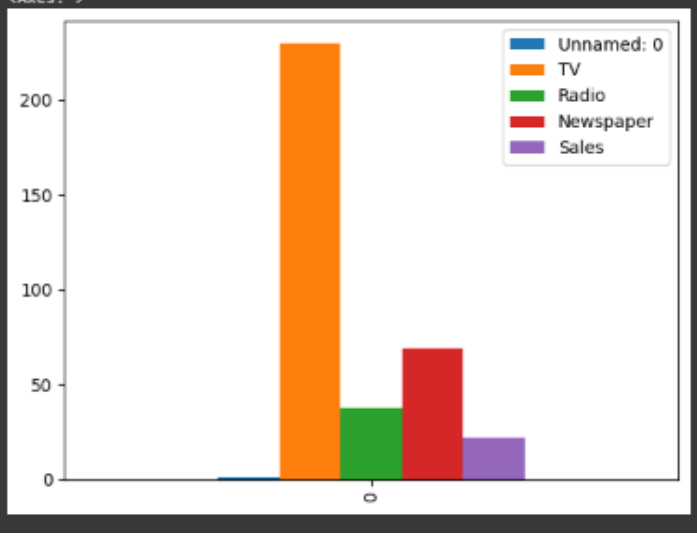
- Visualizing the sales dependent upon on like TV, Radio , Newspaper.

```
[14] snr.countplot(x='TV',data=data.iloc[0:5])
```



```
[28] data.iloc[0:1].plot(kind='bar')
```

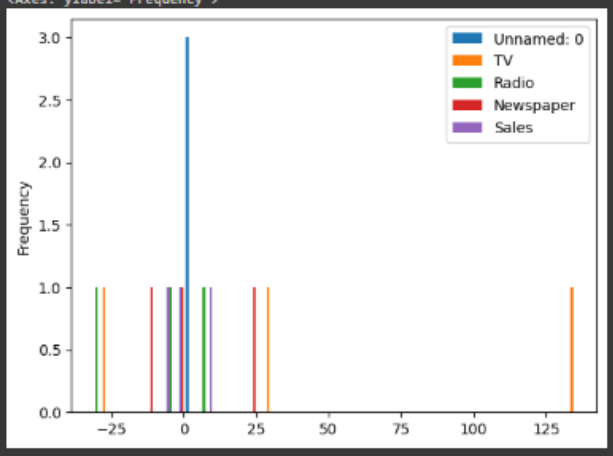
```
<Axes: >
```



As per Visualizing the above graph, Tv advertisement is more than other .

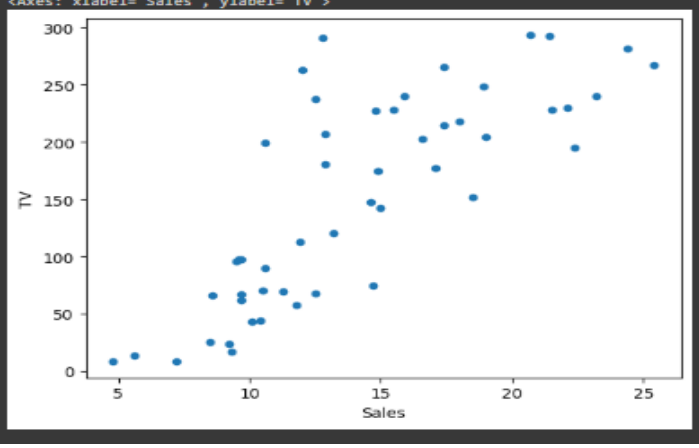
```
[21] data[1:5].diff().plot(kind='hist',bins=200)
```

```
<Axes: ylabel='Frequency'>
```



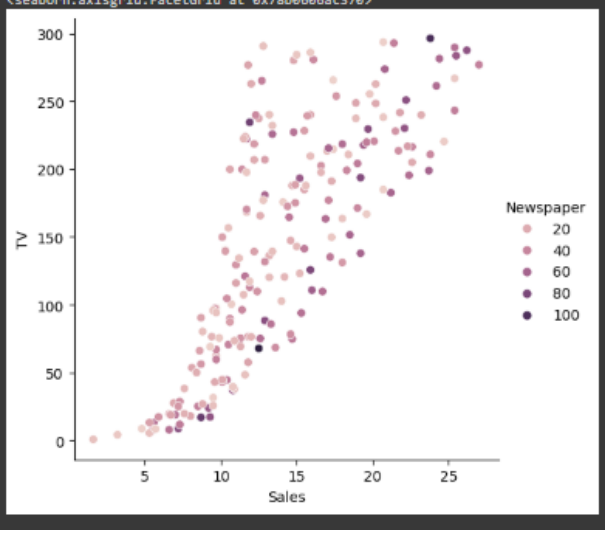
```
data.iloc[0:50].plot.scatter(x='Sales',y='TV')
```

```
<Axes: xlabel='Sales', ylabel='TV'>
```




```
[25] snr.relplot(x='Sales',y='TV',hue='Newspaper',data=data)
```

```
<seaborn.axisgrid.FacetGrid at 0x78b0606ac370>
```



After visualization of data, we predict sales order by using Machine Learning .

- Splitting the dataset into dependent(y) & independent(x) sets

```
[27] #Divide the data into dependent and independent set
     x=data.drop(columns=['Sales'])
     y=data['Sales']
```

- Importing train_test_split from sklearn.model library for splitting the data into train and test sets. (we consider train dataset).

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.15)
```

- Importing logistic regression from sklearn Libaray & then activating the Machine learning Model .Then used regression.fit() to training the model by providing train & test sets as x & y. And then predicted the trained model with help of MLM & the checked score as regression.score(x,y)

In the above model we can see that the accuracy obtained is 89%

- Now applying new algorithm Decision Tree, then checked score.

we can see that the accuracy obtained is 99.5%

- Now applying new algorithm random forest , then checked score.

we can see that the accuracy obtained is 99.3%

- Now we compare all algorithms with accuracy

Algorithms	accuracy
Liner regression	89%
Random Forest	99.3%
Decision Tree	99.5%

Decision Tree algorithms is better than Linear ,random Forest regression.

Applying Decision Tree algorithms for new predictions value.

Conclusion:- In this new data set we analysed the data we found the sales order dependent on all advertisement.

Thank you