FUNDAMENTAL OF DATA SCIENCE

Reg No: 230701024

Name: Akshaya sri S

Dept: CSE _ A (II nd yr)

Exp-1_Matplotseaborn

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

%matplotlib inline

data=pd.read_csv('Iris.csv')

data

	ld	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

data.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
    Column
                   Non-Null Count
                                  Dtype
    Ιd
                   150 non-null
                                  int64
 0
    SepalLengthCm 150 non-null
                                  float64
 1
    SepalWidthCm 150 non-null float64
 2
    PetalLengthCm 150 non-null
                                 float64
 3
    PetalWidthCm
                   150 non-null
                                  float64
 4
    Species
                   150 non-null
                                  object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
```

data.value_counts('Species')

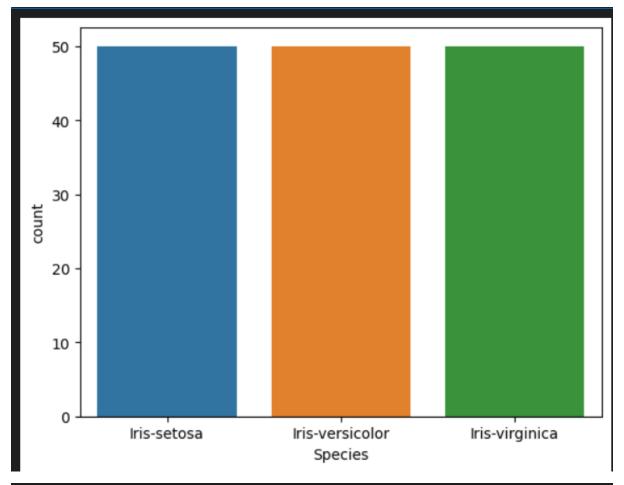
```
Species
Iris-setosa 50
Iris-versicolor 50
Iris-virginica 50
dtype: int64
```

plt.show()

dummies=pd.get_dummies(data.Species)

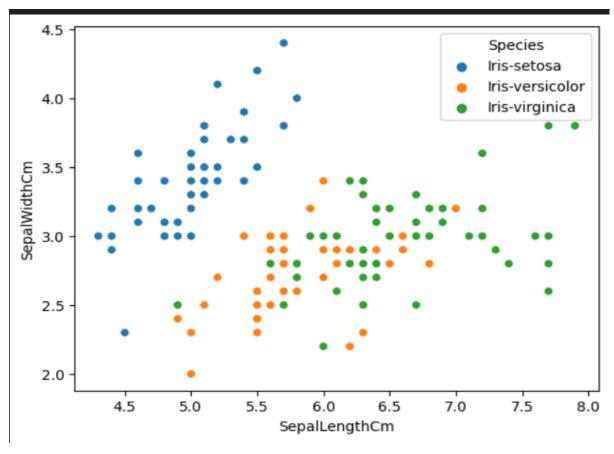
 $Final Dataset = pd.concat([pd.get_dummies(data.Species), data.iloc[:,[0,1,2,3]]], axis = 1)$

FinalDataset.head()

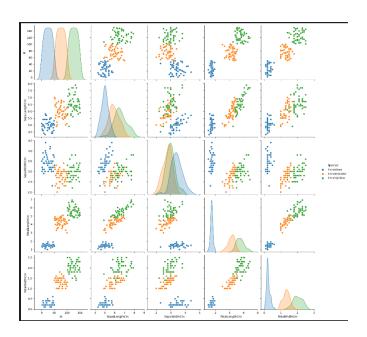


	Iris-setosa	Iris-versicolor	Iris-virginica	ld	SepalLengthCm	SepalWidthCm	PetalLengthCm
0	1	0	0	1	5.1	3.5	1.4
1	1	0	0	2	4.9	3.0	1.4
2	1	0	0	3	4.7	3.2	1.3
3	1	0	0	4	4.6	3.1	1.5
4	1	0	0	5	5.0	3.6	1.4

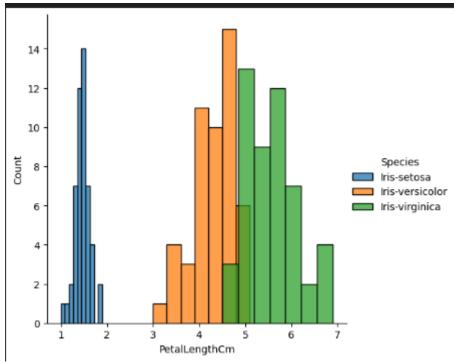
sns. scatterplot (x='SepalLengthCm', y='SepalWidthCm', hue='Species', data=data)



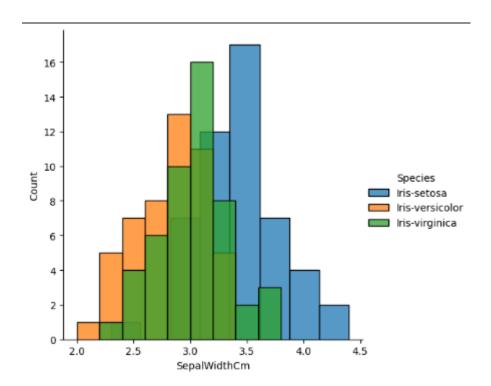
sns.pairplot(data,hue='Species',height=3);
sns.FacetGrid(data,hue='Species',height=5).map(sns.histplot,'PetalLengthCm').add_legend(); plt.sho



 $sns. Facet Grid (data, hue='Species', height=5). map (sns. histplot, 'PetalLength Cm'). add_legend (); plt. shown a simple of the control o$



sns.FacetGrid(data,hue='Species',height=5).map(sns.histplot,'SepalWidthCm').add_legend(); plt.show();



Exp-2_Numpy buildin functions

```
import numpy as np
array=np.random.randint(1,100,9)
array
 array([69, 82, 26, 46, 64, 44, 96, 48, 81])
np.sqrt(array)
 array([8.30662386, 9.05538514, 5.09901951, 6.78232998, 8.
         6.63324958, 9.79795897, 6.92820323, 9.
                                                           ])
array.ndim
new_array=array.reshape(3,3)
new_array
  array([[69, 82, 26],
         [46, 64, 44],
         [96, 48, 81]])
new_array.ravel()
  array([69, 82, 26, 46, 64, 44, 96, 48, 81])
newm=new_array.reshape(3,3)
newm
 array([[69, 82, 26],
         [46, 64, 44],
         [96, 48, 81]])
```

```
newm[1:2,1:3]
new_array[0:2,0:1]
new_array[1:3]
```

Exp-3_Pandas

```
0 1 20 1 Smith 500001 2 Jones 60000
```

Empd Name Salary

0 1 Smith 50000 **1** 2 Jones 60000

R&D Spend Administration Marketing Spend State Profit

```
0 165349.20 136897.80 471784.10 New York 192261.83
1 162597.70 151377.59 443898.53 California 191792.06 2
153441.51 101145.55 407934.54 Florida 191050.39
3 144372.41 118671.85 383199.62 New York 182901.99
4 142107 34 91391 77 366168 42 Florida 166187 94
```

R&D Spend Administration Marketing Spend State Profit

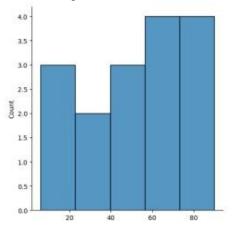
45 1000.23 124153.04 1903.93 New York 64926.08

```
46 1315.46 115816.21 297114.46 Florida 49490.75
      47 0.00 135426.92 0.00 California 42559.73
      48 542.05 51743.15 0.00 New York 35673.41
      49 0 00 116983 80 45173 06 California 14681 40
         emp id name salary
      0 1 SREE VARSSINI K S 5000
      1 2 SREEMATHI B 6000
      2 3 SREYA G 7000
      3 4 SREYASKARI MULLAPUDI 5000
      4 5 SRI AKASH U G 8000
         emp id name salary
      2 3 SREYA G 7000
      3 4 SREYASKARI MULLAPUDI 5000
      4 5 SRI AKASH U G 8000
      5 6 SRI HARSHAVARDHANAN R 3000
      6 7 SRI HARSHAVARDHANAN R 6000
     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 7 entries, 0 to 6 Data
     columns (total 3 columns):
     # Column Non-Null Count Dtype
     0 emp id 7 non-null int64
     1 name 7 non-null object 2 salary 7 non-null int64 dtypes:
       int64(2), object(1) memory usage: 296.0+ bytes salary
      0 5000
      1 6000
     2 7000
      3 5000
      4 8000
      5 3000
      6 6000
     5714.285714285715
df.salary.median()
     6000.0
df.salary.mode()
         salary
      0 5000
      1 6000
df.salary.var()
     2571428.5714285714
df.salary.std()
     1603.5674514745463
```

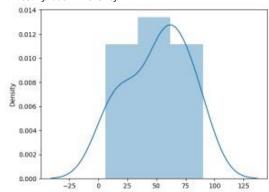
```
df.describe()
                   emp id salary
          count 7.000000 7.000000 mean
          4.000000 5714.285714 std
          2.160247 1603.567451
          min 1.000000 3000.000000 25%
          2.500000 5000.000000 50%
          4.000000 6000.000000
          75% 5.500000 6500.000000 max
          7 000000 8000 000000
    df.describe(include='all')
                    emp id name salary
          count 7.000000 7 7.000000
          unique NaN 6 NaN top NaN SRI
           HARSHAVARDHANAN R NaN
           freq NaN 2 NaN
            mean 4.000000 NaN 5714.285714
             std 2.160247 NaN 1603.567451
          min 1.000000 NaN 3000.000000 25%
            2.500000 NaN 5000.000000 50%
            4.000000 NaN 6000.000000 75%
           5.500000 NaN 6500.000000 max 7
               000000 NaN 8000 000000
          array([[1, 'SREE VARSSINI K S',
                       5000],
          [2, 'SREEMATHI B', 6000],
          [3, 'SREYA G', 7000],
                  'SREYASKARI
                                   MULLAPUDI',
                                                    5000],
                                                                                  'SRI
                                                                                           AKASH
                                                                                                            G'.
                                                                         ſ5,
https://colab.research.google.com/drive/1TNEzkVEMxSI_3eUDFZrcEeJH-g7BNg2j#scrollTo=IDn_tbKJiBVI&printMode=true 3/4
10/14/24, 12:15 PM pandasclass.ipynb - Colab
          [6, 'SRI HARSHAVARDHANAN R', 3000],
[7, 'SRI HARSHAVARDHANAN R', 6000]], dtype=object)
             emp id name salary
          0 1 SREE VARSSINI K S 5000
          1 2 SREEMATHI B 6000
          2 3 SREYA G 7000
          3 4 SREYASKARI MULLAPUDI 5000
          4 5 SRI AKASH U G 8000
          5 6 SRI HARSHAVARDHANAN R 3000
          6 7 SRI HARSHAVARDHANAN R 6000
```

EXP-4_Outlier Detection

<seaborn.axisgrid.FacetGrid a0x78f3291c2710>



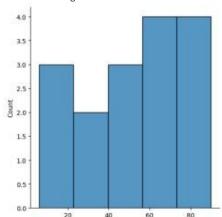
<Axes: ylabel='Density'>



new_array

array([27, 50, 44, 6, 58, 61, 23, 86, 67, 20, 75, 7, 79, 61, 90, 54])

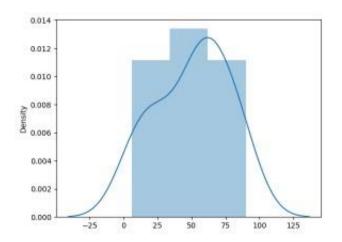
<seaborn.axisgrid.FacetGrid at 0x78f2e09bb580>



lr1,ur1

(-38.5, 133.5) final_array array([27, 50, 44, 6, 58, 61, 23, 86, 67, 20, 75, 7, 79, 61, 90, 54])

<Axes: ylabel='Density'>



Exp-5_Missing and inappropriate data

Datase⁻

Hotel.csv

CustomerID	Age_Group	Rating(1- 5)	Hotel	FoodPreference	Bill	NoOfPax	EstimatedSalary	Age_Group
1	20-25	4	Ibis	veg	1300	2	40000	20-25
2	30-35	5	LemonTree	Non-Veg	2000	3	59000	30-35
3	25-30	6	RedFox	Veg	1322	2	30000	25-30
4	20-25	-1	LemonTree	Veg	1234	2	120000	20-25
5	35+	3	Ibis	Vegetarian	989	2	45000	35+
6	35+	3	Ibys	Non-Veg	1909	2	122220	35+
7	35+	4	RedFox	Vegetarian	1000	-1	21122	35+
8	20-25	7	LemonTree	Veg	2999	-10	345673	20-25
9	25-30	2	Ibis	Non-Veg	3456	3	-99999	25-30
9	25-30	2	Ibis	Non-Veg	3456	3	-99999	25-30
10	30-35	5	RedFox	non-Veg	- 6755	4	87777	30-35

	Customeri	ID Mge_G	oup raus	ig(1-5)	Hotel	FoodPrefere	ence Bill	NOOIPax	Estimated	2 Solory	Age_Group.1	
0		1 2	0-25	4	Ibis		veg 1300	2		40000	20-25	
1		2 3	0-35	5 Le	monTree	Non	-Veg 2000	3		59000	30-35	
2		3 2	5-30	6	RedFax		Veg 1322	2		30000	25-30	
3		4 2	0-25	-1 Le	monTree		Veg 1234	2		120000	20-25	
4		5	35+	3	Ibis	Veget	arian 989	2		45000	35+	
5		6	35+	3	Ibys	Non	Veg 1909	2		122220	35+	
6		7	35+	4	RedFax	Veget	arian 1000	-1		21122	35+	
7		8 2	0-25	7 Le	monTree		Veg 2999	-10		345673	20-25	
8			5-30	2	Ibis	Non	-Veg 3456	3		-99999	25-30	
9			5-30	2	ibis		Veg 3456	3		-99999	25-30	
10			0-35	5	RedFax		-Veg -6755	4		87777	30-35	
0		Fals	e									
1		Fals		<	lass	'pan	das.c	ore.f	rame	. Dat	aFrame'>	
			533			ndex:						
2		Fals	9	Da	ata c	olumn	s (to	tal 9	col	umns	:):	
3		Fals	e			olumn			Non	-Nul	1 Count	Dtyp
4		Fals	e			uetem	ONTO		44		nu11	int6
5		Fals		9		ustom age Gr					null	obje
		YEAR TO	770			ating	100				null	int6
6		Fals	e			lotel	/				null	obje
7		Fals	e	4	1 F	oodPr	efere	nce	11	non-	null	obje
1		1-			5 E	Bill			11	non-	null	int6
8		rais	P								11022	Tite
8		Fals	8	•		loOfPa		~~	11 1	non-	null	inte
9)	Tru	e		7 E	loOfPa stima lge_Gr	tedSa	\$6850S	11 i	non- non- non-	A December 1997	int6
9 10 dt	ype:	Tru Fals boo	e e 1	dt me	ypes	loOfPa stima kge_Gr s: int usag	tedSa oup.1 64(5) e: 92	, obj 4.0+	11 1 11 1 11 ect(4 byte:	non- non- non- 4)	null null	int6
9 10 dt	ype:	Tru Fals boo	e e 1 Rating(1-5)	d1 me	ypes emory	loOfPa stima kge_Gr s: int usag	tedSa oup.1 64(5) e: 92	, obj 4.0+	11 i 11 i ect(byte:	non- non- 4) 5	null null null	int6
9 10 dt	ype:	Tru Fals boo Age_Group	e e 1 Rating(1-5)	di di me	ypes cypes emory	loOfPa stima age_Gr s: int r usag	tedSa oup.1 64(5) e: 92	, obj	11 in	non- non- non- 4) s Group.1	null null null	int6 int6 obje
9 10 dt cus	stomerID	Fals boo Age_Group 20-25 30-35	e e 1 Rating(1-5) 4	dit me	ypes emory	Non-Veg 200	tedSa oup.1 64(5) ee: 92 iii NoOfPax 00 2	, obj 4.0+	11 1 11 1 11 1 ect(4 byte:	non- non- non- 4) s _Group.1 20:25	null null null	int6
9 10 dt cus	stomerID	Tru Fals boo Age_Group 20-25 30-35 25-30	e e 1 Rating(1-5) 4 5	Hote LemonTre RedFo	ypes emory FoodPros	loOfPa stima Age_Gr s: int r usag eference B veg 13 Non-Veg 20 Veg 13	tedSa coup.1 64(5) e: 92 iii NoOfPax iii 0 2 20 3	, obj 4.0+ EstimatedS	11 11 11 ect(4 byte: salary Age	mon - non - non - 4) s Group.1 20-25 30-35 25-30	null null null	int6
9 10 dt cus	stomerID	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25	e e 1 Rating(1-5) 4 5 6	dit me Hote Bit LemonTre RadFo LemonTre	7 E 3 A types emory d FoodPres s	loOfPa stima Age_Gr s: int r usag eference B veg 13 Non-Veg 20 Veg 13 Veg 12	tedSaroup.1 64(5) e: 92 iii NoOfPax 20 2 24 2	obj 4.0+ EstimatedS	11 11 11 11 11 11 11 11	mon - non - 4) s Group.1 20-25 30-35 25-30 20-25	null null null	int6
9 10 dt cus	stomerID	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 35+	e e 1 1	dit me Hote Ibi LemonTre RadFo LemonTre	7 E 3 A types types FoodPri s 8 X 8	Age_Gr : int / usag / u	tedSa oup.1 64(5) e: 92 iii NoOIPax iii 0 NoOIPax 200 3 22 2 24 2 39 2	obj 4.0+	11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Group.1 20-25 30-35 25-30 20-25 35+	null null null	int6
9 10 dt Cus 0 1 2 3 4 5	stomerID 1 2 3 4 5	Tru Fals boo Age_Group 20-25 30-35 20-25 35+ 36+	e e 1 1 Rating(1-5) 4 5 6 6 -1 3 3 3	Hote Bit LemonTre RadFo LemonTre Ribit LemonTre Ribit Rib	7 E 3 A 5 Lypes Semory H FoodPri S 8 8 8	Stima Age_Gr : int ' usage deference B veg 13 Non-Veg 20 Veg 13 Veg 12 Non-Veg 19 Non-Veg 19	tedSa oup.1 64(5) e: 92 iii NoOiPax 200 2 24 2 39 2	Stimated S	11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Group.1 20-25 30-35 25-30 20-26 35+ 35+	null null null	int6
9 dt cus	stomerID	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 35+ 35+ 35+	e e 1 1	Hote Boi LemonTre RadFo LemonTre Boi By RedFo	7 E 3 A 4 E 5 E 5 E 6 E 7 E 8	Stima Age_Gr : int usage eference B veg 13 Non-Veg 20 Veg 13 Veg 12 Non-Veg 19 Non-Veg 19 Non-Veg 19	tedSa oup.1 64(5) e: 92 iil NoOfPax 200 2 24 2 39 2 99 2 90 -1	, obj 4.0+	11 1 1 1 1 1 1 1 1 1	Group.1 20-25 30-35 25-30 20-25 35+ 35+ 35+	null null null	int6
9 10 dt Cus 0 1 2 3 4 5	stomerID 1 2 3 4 5	Tru Fals boo Age_Group 20-25 30-35 20-25 35+ 36+	e e 1 1 Rating(1-5) 4 5 6 6 -1 3 3 4 4	Hote Bill LemonTre RedFo LemonTre Bill By RedFo LamonTre	7 Egpes 2 Egpes 2 Egpes 3 A Egpes 4 FoodPri 5 Eg 6 Eg 7 X X X X X X X X X X X X X X X X X X X	Joof Pa Stima Age_Gr :: int / usag Herence B veg 131 Veg 12: Veg 12: Veg 1999larian 90 Non-Veg 1999larian 100 Veg 29:	tedSa oup.1 64(5) e: 92 iii NoOfPax iii NoOfPax 2000 3 3 222 2 244 2 399 2 99 100 -10	sobj4.0+	11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Group.1 20-25 30-35 25-30 20-26 35+ 35+	null null null	int6
9 10 dt cus 0 1 2 3 4 5 6 6 7	stomerID : 3 4 5 6 7 8	Tru Fals boo 20-25 30-35 25-30 20-25 36- 36- 36- 36- 20-25	e e 1 Rating(1-5) 4 5 6 6 -1 3 3 4 7	Hote Bil LemonTre RedFo LemonTre Bil By RedFo LamonTre Bil	7 Egpes 2 Egpes 2 Egpes 3 A Egpes 4 FoodPri 5 B Eg 5 Vi 6 S Vi 7 S Vi 8	Joof Pa Stima Age_Gr :: int / usag Herence B veg 13 Veg 12 Veg 12 Veg 12 Veg 19 Non-Veg 19 Spetarian 10 Veg 29	tedSa oup.1 64(5) e: 92 iii NoOfPax iii NoOfPax 222 2 244 2 399 2 99 -10 56 3	, obj 4.0+ EstimatedS	11 1 1 1 1 1 1 1 1 1	Group.1 20-25 30-35 25-30 20-25 35+ 35+ 20-25	null null null	int6
9 10 dt Cus	stomerID 1 2 3 4 5 6 7 8 9 10	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 36- 36- 36- 36- 20-25 26-30	e e 1 Rating(1-5) 4 5 6 -1 3 4 7 2 5	Hote Bil LemonTre RedFo LemonTre Bil By RedFo LamonTre Bil	7 E S S S S S S S S S S S S S S S S S S	stima ge_Gr : int usag eference B veg 13 Non-Veg 20 Veg 13 Veg 12 veg 12 veg 14 Non-Veg 34 Non-Veg 34 Non-Veg 36 Non-Veg 36 Non-Veg 36 Non-Veg 36 Non-Veg 36	tedSa oup.1 64(5) e: 92 iii NoOfPax 00 2 00 3 3 22 22 2 34 2 99 2 99 2 99 100 -1 106 3	, obj 4.0+ Estimated S	11 11 11 11 11 11 11 11	Group.1 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35	null null null	int6
9 10 dt Cus	stomerID 1 2 3 4 5 6 7 8 9 10	Tru Fals boo 20-25 30-35 25-30 20-25 35+ 35+ 20-25 20-25 30-35	e e 1 Rating(1-5) 4 5 6 -1 3 4 7 2 5	Hotel Hotel Bit LemonTre RedFo LemonTre Bit RedFo	FoodPress Version State	stima ge_Gr : int usag eference B veg 13 Non-Veg 20 Veg 13 Veg 12 veg 12 veg 14 Non-Veg 34 Non-Veg 34 Non-Veg 36 Non-Veg 36 Non-Veg 36 Non-Veg 36 Non-Veg 36	tedSacoup. 1 64(5) e: 92 iii NooiPax 200 3 222 22 39 22 39 24 300 -1 100 300 300 300 300 300 300 300 300 300 3	sobj 4.0+ Estimated S	11 11 11 11 11 11 11 11	Group.1 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35	null null null	int6
9 10 dt Cus	stomerID	Tru Fals boo 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35	e 1 Rating(1-5) 4 5 6 -1 3 4 7 2 5 Rating(1-5)	Hotel Hotel Hotel RedFo LemonTre RodFo LemonTre RodFo LemonTre RodFo Hotel	FoodPre	stima ge_Gr : int usag eference 8 veg 13 Non-Veg 20 Veg 12 yegetarian 9 Non-Veg 19 gegetarian 19 Non-Veg 34 non-Veg 67	tedSacoup. 1 64(5) e: 92 iii NooiPax 200 3 222 22 34 22 39 22 39 23 30 31 10 10 10 10 10 10 10 10 10 10 10 10 10	sobj 4.0+ Estimated S	11 1 1 1 1 1 1 1 1 1	non- non- non- 4) s Group.1 20-25 30-35 25-30 20-26 35+ 35+ 35+ 20-25 20-25 30-35 20-25 20-25 30-35	null null null	int6
9 10 dt Cus 0 1 2 3 3 4 5 6 6 7 7 8 8 0 0 Cus 1 1	stormerID 1 2 3 4 5 6 7 8 9 10	Tru Fals boo 20-25 30-35 25-30 20-25 36+ 36+ 36- 20-25 26-30 30-35 Age_Group 20-25	e e 1 Rating(1-5) 4 5 6 -1 3 4 7 2 5 Rating(1-5) 4	Hotel Hotel Hotel RedFo LemonTre Rod RodFo LemonFo LemonFo	FoodPress Value of FoodPress Val	stima ge_Gr : int usag eference B veg 13 Non-Veg 20 Veg 12 veg 12 veg 19 spetarian 9 Non-Veg 34 Non-Veg 34 Non-Veg 35 spetarian 9 Non-Veg 36 spetarian 9 spetari	tedSacoup. 1 64(5) e: 92 iii NooiPax 200 3 22 2 24 2 299 2 299 2 200 -1 100 3 3 iii NooiPax 4 iii NooiPax 200 3	sobj4.0+ Estimated S	11 1 1 1 1 1 1 1 1 1	non- non- non- 4) s Group.1. 20-25- 20-25- 20-25- 35- 35- 20-25-	null null null	int6
9 10 dt Cus 0 1 1 2 2 3 3 4 5 6 6 7 7 8 8 0 0 Cus 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	stornerID	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35 Age_Group 20-25 30-35	e 1 Rating(1-5) 4 5 6 -1 3 4 7 2 5 Rating(1-5) 4 5	Hotel Hotel Hotel RedFo LemonTre Rod RodFo LemonTre Ibi RedFo Hotel LemonTre Ibi RedFo	FoodPro	stima ge_Gr : int usag eference B veg 13 Non-Veg 20 Veg 12 Veg 12 Veg 12 Veg 12 Veg 14 Non-Veg 34 Non-Veg 34 Non-Veg 35 Veg 13 Non-Veg 20 Veg 13 Non-Veg 20 Veg 13 Veg 14 Non-Veg 20 Veg 15 Veg 16 Veg 16 Veg 17 V	tedSacoup. 1 64(5) e: 92 iii NooiPax 00 2 00 3 02 2 24 2 25 29 2 20 3 4 2 20 3 4 2 20 3 4 2 20 3 20 3 20 3 22 2 20 3 22 2 20 3 20 3	sobj4.0+ Estimated S	11 1 1 1 1 1 1 1 1 1	non- non- non- 4) s Group.1 1 20-25 20-25 25-30 20-25 35-4 35-7 20-25 25-30 30-35 20-30 30 30-30 30 30-30 30 30-30 30 30 30-30 30 30 30 30 30 30 30 30 30 30 30 30 3	null null null	int6
9 10 dt Cust Cust Cust Cust Cust Cust Cust Cus	stornerID	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35 Age_Group 20-25 30-35 25-30	e 1 Rating(1-5) 4 5 6 -1 3 3 4 7 2 5 Rating(1-5) 4 5 6	Hotel Ho	FoodPro	Stima Age_Greence B veg 131 Veg 122 veg 1999starian 100 veg 299 Non-Veg 299 Non-Veg 34 non-Veg 299 Non-Veg 299 Non-Veg 34 Non-Veg 34 veg 13 veg 13 veg 13 veg 13 veg 13	tedSacoup. 1 64(5) e: 92 iii NooiPax 00 2 00 3 02 2 24 2 25 29 2 20 3 4 2 20 3 4 2 20 3 4 2 20 3 20 3 20 3 22 2 20 3 22 2 20 3 20 3	sobj4.0+ Estimated S 11 12 34 Estimated S	11 1 1 1 1 1 1 1 1 1	Group.1.	null null null	int6
9 10 dt Cust Cust Cust Cust Cust Cust Cust Cus	stornerID	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35 Age_Group 20-25 30-35 25-30 20-25	e e 1 Rating(1-5) 4 5 6 -1 3 3 4 7 2 5 Rating(1-5) 4 5 6	Hotel LemonTre RodFo	FoodPress Was a second of the control of the contro	doofPa stima Age_Gr : int ' usag deference B veg 13 Veg 12 Veg 12 Veg 19 Non-Veg 29 Non-Veg 34 Non-Veg 34 Non-Veg 29 Veg 13 Veg 12 Veg 13 Veg 12 Veg 13 Veg 12 Veg 13	tedSa oup.1 64(5) e: 92 iii NoOfPax	sobj4.0+ Estimated S	11 1 1 1 1 1 1 1 1 1	Group.1	null null null	inte
9 10 dt Cust 1 2 3 4 5 6 6 7 7 8 8 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	stornerID	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35 Age_Group 20-25 30-35 25-30 20-25 35+	e e 1 Rating(1-5) 4 5 6 -1 3 3 4 7 2 5 Rating(1-5) 4 5 6 -1 3	Hotel LemonTre RedFo LemonTre RedFo LemonTre RedFo LemonTre RedFo LemonTre RedFo LemonTre RedFo	FoodPress	Stima Age_Greence Barrence Bar	tedSa oup.1 64(5) e: 92 iii NoOfPax 00 2 00 3 322 24 34 2 39 2 00 -1 10 16 3 35 4 iii NoOfPax 00 2 00 3 00 3 00 3 00 3 00 3 00 3 00 3	sobj4.0+ Estimated S 11 12 Estimated S 11 11 Estimated S	11 1 1 1 1 1 1 1 1 1	Group.1 20-25 35-30 35-30 20-25 35-30 30-35 20-25 30-35	null null null	int6
9 10 dt Cus 0 1 1 2 3 3 4 5 6 6 7 7 8	stomerID / 1 / 2 / 3 / 4 / 5 / 6 / 7 / 8 / 9 / 10 / 10 / 2 / 3 / 4 / 5 / 6 / 6 / 7 / 6 / 6 / 7 / 7 / 7 / 7 / 7	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35 Age_Group 20-25 30-35 25-30 20-25 35+ 35+ 35+	e e 1 Rating(1-5) 4 5 6 -1 3 4 7 2 5 Rating(1-5) 4 5 6 -1 3 4 5 4 5 6 4 5 6 4 7 2 5 8 4 7 2 4 5 6 4 5 6 4 6 4 7 8 8 8 8 8 8 8 8 8 8 8 8	Hote Bui LemonTre RedFo LemonTre RedFo LemonTre RedFo LemonTre RedFo LemonTre RedFo LemonTre RedFo	FoodPress	Stima	tedSa oup.1 64(5) e: 92 iii NoOfPax 00 2 00 3 22 244 2 09 2 00 -1 00 66 3 00 3 00 3 00 3 00 3 00 3 00 3 0	sobj4.0+ Estimated S 11 12 Estimated S 11 11 11 11 11 11 11	11 1 1 1 1 1 1 1 1 1	Group.1 20-25 30-35 35-30 30-35 20-25 30-35 20-25 30-35	null null null	int6
9 10 dt Cust 1 2 3 3 4 5 6 6 7 7 8 8 8 10 10 10 10 10 10 10 10 10 10 10 10 10	stomerID	Tru Fals boo Age_Group 20-25 30-35 25-30 20-25 35+ 35+ 20-25 25-30 30-35 20-25 30-35 25-30 20-25 30-35 25-30 20-25 35+ 35+ 35+ 35+	e e 1 Rating(1-5) 4 5 6 -1 3 4 7 2 5 Rating(1-5) 4 5 6 -1 3 4 5 4 5 6 4 5 6 4 7 2 5 8 4 7 2 4 5 6 4 5 6 4 6 4 7 8 8 8 8 8 8 8 8 8 8 8 8	Hote Bui LemonTre RodFo LemonTre Bui RodFo LemonTre RodFo LemonTre RodFo LemonTre RodFo LemonTre RodFo RodFo	FoodPress	Section	tedSa oup.1 64(5) e: 92 iii NoO(Pax ii NoO(Pax iii NoO	5 obj 4.0+ Estimateds 11 12 34 Estimateds 11 11 11 11 13	11 1 1 1 1 1 1 1 1 1	non- non- 4) 5 Group.1 20-25 30-35 25-30 30-35 20-25 25-30 30-35 20-25 20-25 20-25 30-35 3	null null null	int6

	CustomerID	Age_Group	Rating(1-5)	Hotel	FoodPreference	Bill	NoOfPax	Estimated Salary
0	1	20-25	4	Ibis	veg	1300	2	40000
1	2	30-35	5	LemonTree	Non-Veg	2000	3	59000
2	3	25-30	6	RedFox	Veg	1322	2	30000
3	4	20-25	- 1	LemonTree	Veg	1234	2	120000
4	5	35+	3	lbis	Vegetarian	989	2	45000
5	6	35+	3	Ibys	Non-Veg	1909	2	122220
6	7	35+	4	RedFox	Vegetarian	1000	-1	21122
7	8	20-25	7	LemonTree	Veg	2999	-10	345673
8	9	25-30	2	Ibis	Non-Veg	3456	3	-99999
9	10	30-35	5	RedFox	non-Veg	-6755	4	87777
	CustomeriD	Age_Group	Rating(1-5)	Hotel	FoodPreference	Bill	NoOfPax	Estimated Salary
0	1.0	20-25	4.0	Ibis	veg	1300.0	2	40000.0
1	2.0	30-35	5.0	LemonTree	Non-Veg	2000.0	3	59000.0
2	3.0	25-30	NaN	RedFox	Veg	1322.0	2	30000.0
3	4.0	20-25	NaN	LemonTree	Veg	1234.0	2	120000.0
4	5.0	35+	3.0	lbis	Vegetarian	989.0	2	45000.0
5	6.0	35+	3.0	lbys	Non-Veg	1909.0	2	122220.0
6	7.0	35+	4.0	RedFox	Vegetarian	1000.0	-1	21122.0
7	8.0	20-25	NaN	LemonTree	Veg	2999.0	-10	345673.0
8	9.0	25-30	2.0	lbis	Non-Veg	3456.0	3	NaN
9	10.0	30-35	5.0	RedFox	non-Veg	NaN	4	87777.0
				377888390		2557		
	CustomerID	Age_Group	Rating(1-5)	Hotel	FoodPreference	Bill	NoOfPax	Estimated Salary
0	1.0	20-25	4.0	Ibis	veg	1300.0	2.0	40000.0
1	2.0	30-35	5.0	LemonTree	Non-Veg	2000.0	3.0	59000.0
2	3.0	25-30	NaN	RedFox	Veg	1322.0	2.0	30000.0
3	4.0	20-25	NaN	LemonTree	Veg	1234.0	2.0	120000.0
4	5.0	35+	3.0	Ibis	Vegetarian	989.0	2.0	45000.0
5	6.0	35+	3.0	lbys	Non-Veg	1909.0	2.0	122220.0
6	7.0	35+	4.0	RedFox	Vegetarian	1000.0	NaN	21122.0
7	8.0	20-25	NaN	LemonTree	Veg	2999.0	NaN	345673.0
8	9.0	25-30	2.0	Ibis	Non-Veg	3456.0	3.0	NaN
9	10.0	30-35	5.0	RedFox	non-Veg	NaN	4.0	87777.0
	CustomeriD	Age_Group	Rating(1-5)	Hotel	FoodPreference	Bill	NoOfPax	Estimated Salary
0	1.0	20-25	4.0	lbis	Veg	1122000	2.0	P DOLLARS
1	2.0	30-35	5.0		Non-Veg		3.0	
2	3.0	25-30	4.0	RedFox	Veg	1322.0	2.0	
3	4.0	20-25	4.0	LemonTree	Veg	1234.0	2.0	
4	5.0	35+	3.0	Ibis	Veg	989.0	2.0	
5		35+			10000000			
	6.0	30*	3.0	lbis	Non-Veg	1909.0	2.0	
	7.6	26.	4.0	DadEes	1/44			
6	7.0	35+	4.0	RedFox	Veg	1000.0	2.0	
6	8.0	20-25	4.0	LemonTree	Veg	2999.0	2.0	345673.0
6						2999.0 3456.0		345673.0 96755.0

Exp-6_Feature Scaling

import numpy as np import pandas as pd df=pd.read_csv('/content/pre-process_datasample.csv')

Country Age Salary Purchased

- **0** France 44.0 72000.0 No
- 1 Spain 27.0 48000.0 Yes
- 2 Germany 30.0 54000.0 No
- 3 Spain 38.0 61000.0 No
- 4 Germany 40.0 NaN Yes
- 5 France 35.0 58000.0 Yes
- 6 Spain NaN 52000.0 No
- **7** France 48.0 79000.0 Yes **8** NaN 50.0 83000.0 No **9** France 37.0 67000.0 Yes

Next steps: df.head()

Generate code with df View recommended plots New interactive sheet

Country Age Salary

Purchased 0 France 44.0

72000.0 No **1**

Spain 27.0 48000.0

Yes 2 Germany

30.0 54000.0 No 3 Spain

38.0 61000.0

No 4 Germany 40 0

NaN Yes

Next steps:

```
features=df.iloc[:,:-1].values
         <ipython-input-5-20665a0bbaa1>:1: FutureWarning: A value is trying to be set on a
         copy of a DataFrame o The behavior will change in pandas 3.0. This inplace method
         will never work because the intermediate ob
         For example, when doing 'df[col].method(value, inplace=True)', try using
         'df.method({col: value},
         inpla df.Country.fillna(df.Country.mode()[0],inplace=True)
    label=df.iloc[:,-1].values
    Start coding or generate with AI.
https://colab.research.google.com/drive/1Qdb3r_JJTzcANnUYmofxmJd30xZGEnKg#scrollTo=KdrqXPjiF0Pn&printMode=true
1/4 10/5/24, 8:09 PM 09.09.2024-sklearn.ipynb - Colab
    from sklearn.impute import SimpleImputer
    age=SimpleImputer(strategy="mean", missing_values=np.nan)
    Salary=SimpleImputer(strategy="mean",missing_values=np.nan)
    age.fit(features[:,[1]])
         ▼ SimpleImputer i
         SimpleImputer()
    Salary.fit(features[:,[2]])
         ▼ SimpleImputer i
         SimpleImputer()
    SimpleImputer()
         ▼ SimpleImputer ill
         SimpleImputer()
    features[:,[1]]=age.transform(features[:,[1]])
    features[:,[2]]=Salary.transform(features[:,[2]])
    features
```

df.Country.fillna(df.Country.mode()[0],inplace=True)

```
['Spain', 27.0, 48000.0],
          ['Germany', 30.0, 54000.0],
          ['Spain', 38.0, 61000.0],
          ['Germany', 40.0, 63777.777777778],
          ['France', 35.0, 58000.0],
          ['Spain', 38.77777777778, 52000.0],
          ['France', 48.0, 79000.0],
          ['France', 50.0, 83000.0],
          ['France', 37.0, 67000.0]], dtype=object)
    from sklearn.preprocessing import OneHotEncoder
    oh = OneHotEncoder(sparse_output=False)
    Country=oh.fit_transform(features[:,[0]])
    Country
         array([[1., 0., 0.],
          [0., 0., 1.],
          [0., 1., 0.],
          [0., 0., 1.],
          [0., 1., 0.],
          [1., 0., 0.],
          [0., 0., 1.],
          [1., 0., 0.],
https://colab.research.google.com/drive/1Qdb3r_JJTzcANnUYmofxmJd30xZGEnKg#scrollTo=KdrqXPjiF0Pn&printMode=true
10/5/24, 8:09 PM 09.09.2024-sklearn.ipynb - Colab
          [1., 0., 0.],
          [1., 0., 0.]])
    final_set=np.concatenate((Country,features[:,[1,2]]),axis=1)
    final_set
         array([[1.0, 0.0, 0.0, 44.0, 72000.0],
          [0.0, 0.0, 1.0, 27.0, 48000.0],
          [0.0, 1.0, 0.0, 30.0, 54000.0],
          [0.0, 0.0, 1.0, 38.0, 61000.0],
          [0.0, 1.0, 0.0, 40.0, 63777.777777778],
          [1.0, 0.0, 0.0, 35.0, 58000.0],
          [0.0, 0.0, 1.0, 38.77777777778, 52000.0],
          [1.0, 0.0, 0.0, 48.0, 79000.0],
          [1.0, 0.0, 0.0, 50.0, 83000.0],
          [1.0, 0.0, 0.0, 37.0, 67000.0]], dtype=object)
    from sklearn.preprocessing import StandardScaler
    sc=StandardScaler() sc.fit(final_set)
    feat_standard_scaler=sc.transform(final_set)
    feat_standard_scaler
```

array([['France', 44.0, 72000.0],

```
array([[ 1.00000000e+00, -5.00000000e-01, -6.54653671e-01,
     7.58874362e-01, 7.49473254e-01],
      [-1.00000000e+00, -5.0000000e-01, 1.52752523e+00,
      -1.71150388e+00, -1.43817841e+00],
      [-1.00000000e+00, 2.00000000e+00, -6.54653671e-01,
      -1.27555478e+00, -8.91265492e-01],
      [-1.00000000e+00, -5.0000000e-01, 1.52752523e+00,
      -1.13023841e-01, -2.53200424e-01],
      [-1.00000000e+00, 2.00000000e+00, -6.54653671e-01,
     1.77608893e-01, 6.63219199e-16],
      [ 1.00000000e+00, -5.00000000e-01, -6.54653671e-01,
      -5.48972942e-01, -5.26656882e-01],
     [-1.00000000e+00, -5.00000000e-01, 1.52752523e+00,
     0.00000000e+00, -1.07356980e+00],
      [ 1.00000000e+00, -5.00000000e-01, -6.54653671e-01,
     1.34013983e+00, 1.38753832e+00],
      [ 1.00000000e+00, -5.00000000e-01, -6.54653671e-01,
     1.63077256e+00, 1.75214693e+00],
      [ 1.00000000e+00, -5.00000000e-01, -6.54653671e-01, -2.58340208e-
    01, 2.93712492e-01]])
from sklearn.preprocessing import MinMaxScaler
mms=MinMaxScaler(feature_range=(0,1)) mms.fit(final_set)
feat minmax scaler=mms.transform(final set)
feat_minmax_scaler
    array([[1., 0., 0., 0.73913043, 0.68571429],
      [0., 0., 1., 0., 0.],
      [0., 1., 0., 0.13043478, 0.17142857],
     [0., 0., 1., 0.47826087, 0.37142857],
    [0. , 1. , 0. , 0.56521739, 0.45079365],
    [1. , 0. , 0. , 0.34782609, 0.28571429],
                                              [0.,0.,1.,
    0.51207729, 0.11428571],
      [1., 0., 0., 0.91304348, 0.88571429],
      [1., 0., 0., 1., 1.],
      [1., 0., 0., 0.43478261, 0.54285714]])
```

Exp-7_Data Preprocessing

```
pandas as pd
df=pd.read_csv("/content/pre-process_datasample.csv") df

Country Age Salary Purchased
0 France 44.0 72000.0 No
1 Spain 27.0 48000.0 Yes
2 Germany 30.0 54000.0 No
3 Spain 38.0 61000.0 No
4 Germany 40.0 NaN Yes
5 France 35.0 58000.0 Yes 6 Spain NaN 52000.0 No
7 France 48.0 79000.0 Yes 8
```

import numpy as np import

NaN 50.0 83000.0 No

```
9 France 37.0 67000.0 Yes Double-click
(or enter) to edit
df.info()
     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 10 entries, 0 to 9 Data
     columns (total 4 columns):
     # Column Non-Null Count Dtype
     --- -----
    0 Country 9 non-null object
     1 Age 9 non-null float64
     2 Salary 9 non-null float64 3 Purchased 10 non-null object dtypes: float64(2), object(2) memory usage:
      448.0+ bytes
df.Country.mode()
        Country
     0 France
df.Country.mode()[0]
type(df.Country.mode())
      pandas.core.series.Series def __init__(data=None, index=None, dtype: Dtype | None=None,
      name=None, copy: bool | None=None,
      fastpath: bool=False) -> None -index is not None, the resulting Series is
      reindexed with the index values._ dtype : str, numpy.dtype, or
      ExtensionDtype, optional
       Data type for the output Series. If not specified, this will be
      inferred from `data`.
       See the :ref:`user guide <basics.dtypes>` for more usages. name
      : Hashable, default None
          The name to give to the Series
df.Country.fillna(df.Country.mode()[0],inplace=True)
df.Age.fillna(df.Age.median(),inplace=True)
df.Salary.fillna(round(df.Salary.mean()),inplace=True)
df
      https://colab.research.google.com/drive/1EfIGC8IXnHLCKH8kXH1QwiDhUp6tMHjW#printMode=true 1/3
```

10/5/24, 6:12 PM 10th Day DataPreprocessing.ipynb - Colab

```
Country Age Salary Purchased
      0 France 44.0 72000.0 No
      1 Spain 27.0 48000.0 Yes
      2 Germany 30.0 54000.0 No
      3 Spain 38.0 61000.0 No
      4 Germany 40.0 63778.0 Yes
      5 France 35.0 58000.0 Yes 6 Spain 38.0 52000.0 No
      7 France 48.0 79000.0 Yes
      8 France 50.0 83000.0 No
      9 France 37 0 67000 0 Yes
pd.get_dummies(df.Country)
         France Germany Spain
```

0 True False False

```
1 False False True
           2 False True False
           3 False False True
           4 False True False
           5 True False False
           6 False False True
           7 True False False
           8 True False False
           9 True False False
    updated_dataset=pd.concat([pd.get_dummies(df.Country),df.iloc[:,[1,2,3]]],axis=1)
    updated_dataset
             France Germany Spain Age Salary Purchased
           0 True False False 44.0 72000.0 No
           1 False False True 27.0 48000.0 Yes
           2 False True False 30.0 54000.0 No 3 False False True
            38.0 61000.0 No
           4 False True False 40.0 63778.0 Yes
           5 True False False 35.0 58000.0 Yes
           6 False False True 38.0 52000.0 No
           7 True False False 48.0 79000.0 Yes
           8 True False False 50.0 83000.0 No.
           9 True False False 37 0 67000 0 Yes
    df.info()
          <class 'pandas.core.frame.DataFrame'>
          RangeIndex: 10 entries, 0 to 9 Data
          columns (total 4 columns):
          # Column Non-Null Count Dtype
          0 Country 10 non-null object
          1 Age 10 non-null float64
          2 Salary 10 non-null float64 3 Purchased 10 non-null object dtypes: float64(2), object(2) memory usage:
            448.0+ bytes
    updated_dataset.Purchased.replace(['No','Yes'],[0,1],inplace=True)
           https://colab.research.google.com/drive/1EfIGC8IXnHLCKH8kXH1QwiDhUp6tMHjW#printMode=true 2/3 10/5/24,
6:12 PM 10th Day DataPreprocessing.ipynb - Colab
    updated dataset
```

```
France Germany Spain Age Salary Purchased
0 True False False 44.0 72000.0 0
```

- 1 False False True 27.0 48000.0 1
- 2 False True False 30.0 54000.0 0
- 3 False False True 38.0 61000.0 0
- 4 False True False 40.0 63778.0 1
- 5 True False False 35.0 58000.0 1 6 False False True 38.0 52000.0 0
- 7 True False False 48.0 79000.0 1
- 8 True False False 50.0 83000.0 0
- 9 True False False 37 0 67000 0 1

Exp-8 EDA -

import seaborn as sns import pandas as pd import numpy as

```
np import matplotlib.pyplot as
plt %matplotlib inline

tips=sns.load_dataset('tips')

tips.head()
```

total_bill tip sex smoker day time size

0 16.99 1.01 Female No Sun Dinner 2

1 10.34 1.66 Male No Sun Dinner 3

2 21.01 3.50 Male No Sun Dinner 3

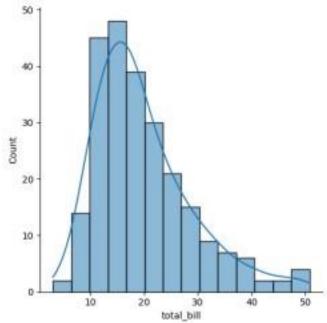
3 23.68 3.31 Male No Sun Dinner 2

4 24.59 3.61 Female No Sun Dinner 4

Code Text

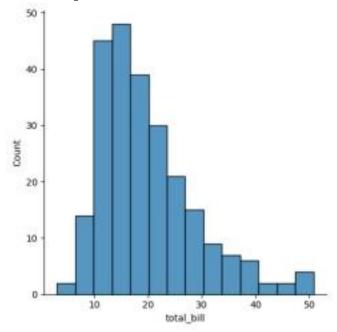
sns.displot(tips.total_bill,kde=True)

<seaborn.axisgrid.FacetGrid at 0x79bb4c7ea680>



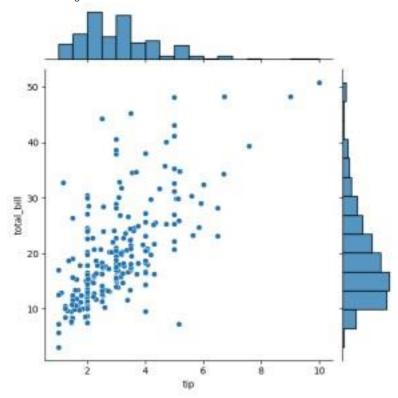
sns.displot(tips.total_bill,kde=False)

<seaborn.axisgrid.FacetGrid at 0x79bb0b0af580>

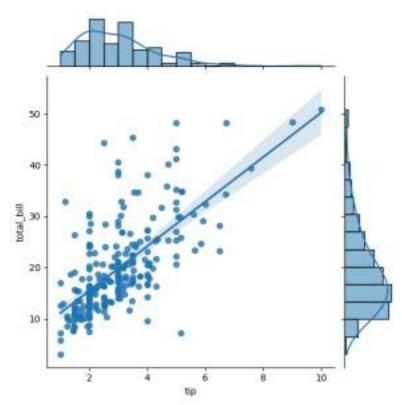


sns.jointplot(x=tips.tip,y=tips.total_bill)

<seaborn.axisgrid.JointGrid at 0x79bb08fc96c0>

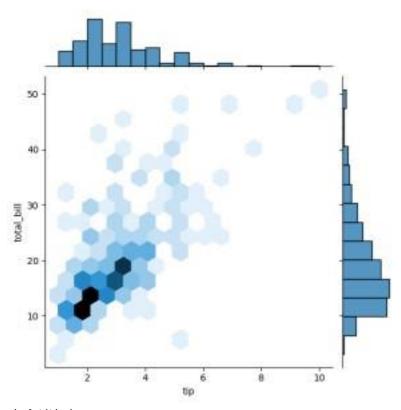


sns.jointplot(x=tips.tip,y=tips.total_bill,kind="reg"
) <seaborn.axisgrid.JointGrid at 0x79bb08fc9cf0>

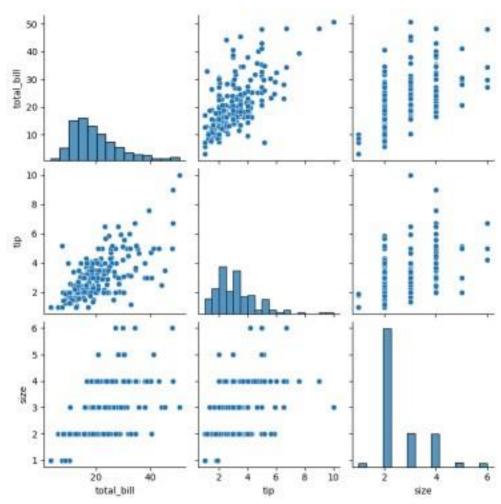


sns.jointplot(x=tips.tip,y=tips.total_bill,kind="hex")

<seaborn.axisgrid.JointGrid at 0x79bb088f4730>



sns.pairplot(tips)
 <seaborn.axisgrid.PairGrid at 0x79bb06fc3d30>



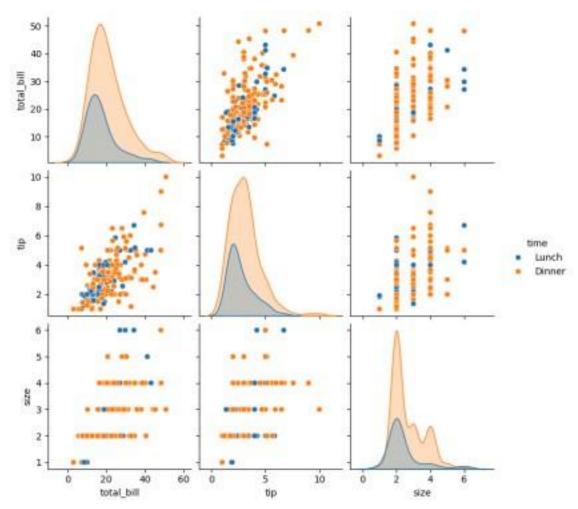
time

Dinner 176

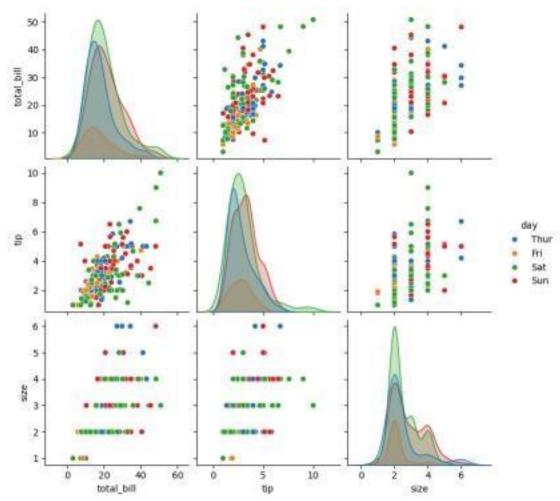
Lunch 68 dtype:

int64

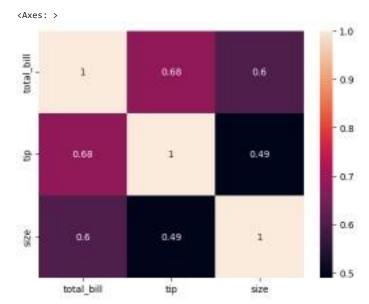
sns.pairplot(tips,hue='time')
<seaborn.axisgrid.PairGrid at 0x79bb088f4670>



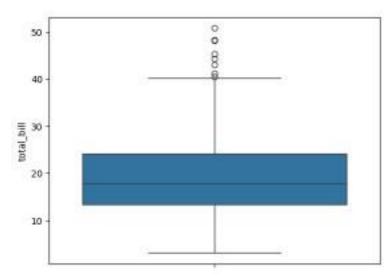
sns.pairplot(tips,hue='day')
<seaborn.axisgrid.PairGrid at 0x79bb08f1f6a0>



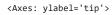
sns.heatmap(tips.corr(numeric_only=True),annot=True)

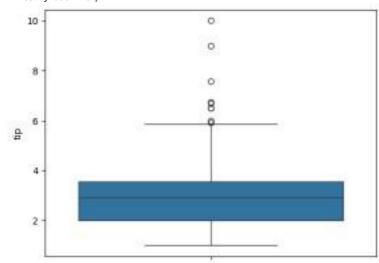


sns.boxplot(tips.total_bill)
 <Axes: ylabel='total_bill'>



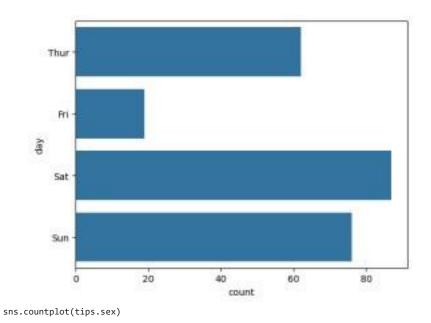
sns.boxplot(tips.tip)





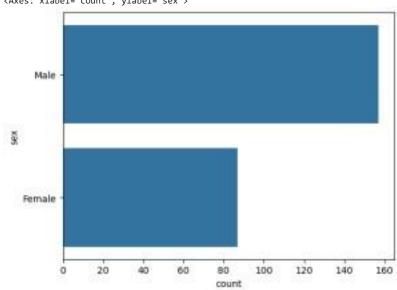
sns.countplot(tips.day)

<Axes: xlabel='count', ylabel='day'>



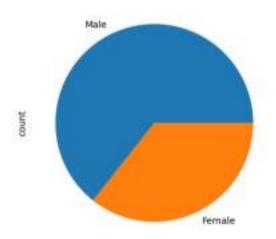
https://colab.research.google.com/drive/1ixdO2LyjKtMYUgtZcoc8jSInDGmeKn4_#scrollTo=J9uBGy0XX3rZ&printMode=true 7/9 10/1/24, 9:52 AM 9.9.2024-Visualization.ipynb - Colab



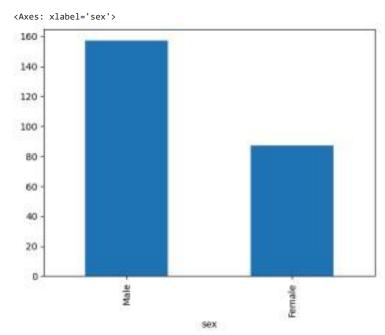


tips.sex.value_counts().plot(kind='pie')

<Axes: ylabel='count'>



tips.sex.value_counts().plot(kind='bar')



sns.countplot(tips[tips.time=='Dinner']['day'])

https://colab.research.google.com/drive/1ixdO2Ly

Exp-9_Regression

In []: In [19]:
In [3]: In [4]:

```
In [5]: import numpy as np import
pandas as pd
df=pd.read_csv('Salary_data.csv')
df.info()
<class 'pandas.core.frame.DataFrame'> RangeIndex: 30 entries, 0 to 29
Data columns (total 2 columns):
# Column Non-Null Count Dtype --- ----- 0 YearsExperience 30
non-null float64 1 Salary 30 non-null int64 dtypes: float64(1), int64(1) memory
usage: 612.0 bytes
df.dropna(inplace=True)
df.info()
<class 'pandas.core.frame.DataFrame'> RangeIndex: 30 entries, 0 to 29
Data columns (total 2 columns):
# Column Non-Null Count Dtype --- ----- 0 YearsExperience 30
non-null float64 1 Salary 30 non-null int64 dtypes: float64(1), int64(1) memory
usage: 612.0 bytes df.describe()
          Out[5]: YearsExperience Salary
        count 30.000000 30.000000
        mean 5.313333 76003.000000
         std 2.837888 27414.429785 min
                  1.100000 37731.000000 25%
                  3.200000 56720.750000
                  50% 4.700000 65237.000000 75%
                  7.700000 100544.750000 max
                  10.500000 122391.000000
In [6]: In [7]: In [20]:
features=df.iloc[:,[0]].values label=df.iloc[:,[1]].values
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(features,label,test_size=0.2,random_st
from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(x_train,y_train)
      Out[20]: ▼ LinearRegression LinearRegression()
                           localhost:8888/notebooks/Regresion.ipynb# 1/2
```

9/16/24, 3:49 AM Regresion - Jupyter Notebook

```
In [21]:
model.score(x_train,y_train)
      Out[21]: 0.9603182547438908
In [23]:
model.score(x_test,y_test)
      Out[23]: 0.9184170849214232
In [24]: model.coef_
      Out[24]: array([[9281.30847068]])
In [25]:
model.intercept_
      Out[25]: array([27166.73682891])
In [26]:
In [27]: In [28]:
In []: In [29]:
In [ ]:
import pickle
pickle.dump(model,open('SalaryPred.model','wb'))
model=pickle.load(open('SalaryPred.model','rb'))
yr_of_exp=float(input("Enter Years of Experience: "))
yr_of_exp_NP=np.array([[yr_of_exp]]) Salary=model.predict(yr_of_exp_NP)
Enter Years of Experience: 44
print("Estimated Salary for {} years of experience is {}: " .format(yr_of_exp,Salary)
Estimated Salary for 44.0 years of experience is
```

Exp-10_KNN

```
import math
import numpy as np
from statsmodels.stats.weightstats import ztest
from scipy.stats import norm
sample\_marks = [650,730,510,670,480,800,690,530,590,620,710,670,640,780,650,490,800,600,510,700]
# Method 1 : Using Z-score
sample_mean = np.mean(sample_marks)
sample_size = np.count_nonzero(sample_marks)
population_mean = 600
population_std = 100
alpha = 0.05
z_score = (sample_mean-population_mean)/(population_std/math.sqrt(sample_size))
critical_value = 1.645 # from z table
if(z_score<critical_value):</pre>
  print('Null hypothesis is accepted!')
else:
  print('Null hypothesis is rejected. \nAlternate hypothesis is accepted!')
# Method 2: Using built in function of ztest
ztest_score, pval = ztest(sample_marks,value=population_mean,alternative='larger')
print('Z-test Score:',ztest_score,'\nP-value:',pval)
if(pval>alpha):
  print('Null hypothesis is accepted!')
```

```
else:
  print('Null hypothesis is rejected. \nAlternate hypothesis is accepted!')
# Method 3: Creating a function
def ztest(x,mu,sigma,n):
  deno = sigma/math.sqrt(n)
  z = (x-mu)/deno
  p = 2*(1-norm.cdf(abs(z)))
  return z,p
s_mean = np.mean(sample_marks)
p_mean = 600
p_std = 100
s_size = np.count_nonzero(sample_marks)
ztest(s_mean,p_mean,p_std,s_size)
ztest(641,600,100,20)
```

```
Null hypothesis is rejected.
Alternate hypothesis is accepted!
Z-test Score: 1.831744911595958
P-value: 0.03349471703839336
Null hypothesis is rejected.
Alternate hypothesis is accepted!

(1.8335757415498277, 0.06671699590108493)
```

Exp-11_Logistics Regression

Import necessary libraries

import numpy as np

from scipy import stats

Given student scores

```
student_scores = np.array([72, 89, 65, 73, 79, 84, 63, 76, 85, 75])

# Hypothesized population mean
mu = 70

# Perform one-sample t-test
t_stat, p_value = stats.ttest_1samp(student_scores, mu)
print("T statistic:", t_stat)
print("P-value:", p_value)

# Setting significance level
alpha = 0.05

# Interpret the results
if p_value < alpha:
    print("Reject the null hypothesis; there is a significant difference between the sample mean and the</pre>
```

print("Fail to reject the null hypothesis; there is no significant difference between the sample mean and the hypothesized population mean.")

T statistic: 2.2894683580127317

hypothesized population mean.")

else:

P-value: 0.047816221110566944

Reject the null hypothesis; there is a significant difference between the sample mean and the hypothesized population mean.

EXP-12_K-Mean clustering

import numpy as np

from scipy.stats import f_oneway

Sample data: Exam scores for three teaching methods

np.random.seed(42)

```
method_A_scores = np.random.normal(loc=80, scale=10, size=30)
method_B_scores = np.random.normal(loc=85, scale=10, size=30)
method_C_scores = np.random.normal(loc=90, scale=10, size=30)

# Perform one-way ANOVA
f_statistic, p_value = f_oneway(method_A_scores, method_B_scores, method_C_scores)
print("F-Statistic:", f_statistic)
print("P-Value:", p_value)
```

F-Statistic: 12.20952551797281 P-Value: 2.1200748140507065e-05