amcat-eda

October 4, 2024

1 Exploratory Data Analysis

Introduction

The dataset was released by Aspiring Minds from the Aspiring Mind Employment Outcome 2015 (AMEO). The study is primarily limited only to students with engineering disciplines. The dataset contains the employment outcomes of engineering graduates as dependent variables (Salary, Job Titles, and Job Locations) along with the standardized scores from three different areas – cognitive skills, technical skills and personality skills. The dataset also contains demographic features. The dataset contains around 40 independent variables and 4000 data points. The independent variables are both continuous and categorical in nature. The dataset contains a unique identifier for each candidate

Objective

The aim of this analysis include : * Describing the dataset and its features comprehensively. * Perform Univariate Analysis * Perform Bivariate Analysis * Exploring the relationships between independent variables and the target variable * Identifying any anomalies in the data.

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

[2]:
```
df=pd.read_csv('data.csv')
df.head()
```

[2]:

	Unnamed: 0	ID	Salary	DOJ	DOL	\
0	train	203097	420000.0	6/1/12 0:00	present	
1	train	579905	500000.0	9/1/13 0:00	present	
2	train	810601	325000.0	6/1/14 0:00	present	
3	train	267447	1100000.0	7/1/11 0:00	present	
4	train	343523	200000.0	3/1/14 0:00	3/1/15 0:00	

	Designation	JobCity	Gender	DOB	10percentage	\
0	senior quality engineer	Bangalore	f	2/19/90 0:00	84.3	
1	assistant manager	Indore	m	10/4/89 0:00	85.4	
2	systems engineer	Chennai	f	8/3/92 0:00	85.0	
3	senior software engineer	Gurgaon	m	12/5/89 0:00	85.6	

```
4                      get    Manesar      m  2/27/91 0:00          78.0

   … ComputerScience  MechanicalEngg  ElectricalEngg TelecomEngg  CivilEngg  \
0  …               -1              -1              -1          -1         -1
1  …               -1              -1              -1          -1         -1
2  …               -1              -1              -1          -1         -1
3  …               -1              -1              -1          -1         -1
4  …               -1              -1              -1          -1         -1

   conscientiousness agreeableness extraversion  nueroticism  \
0             0.9737        0.8128       0.5269      1.35490
1            -0.7335        0.3789       1.2396     -0.10760
2             0.2718        1.7109       0.1637     -0.86820
3             0.0464        0.3448      -0.3440     -0.40780
4            -0.8810       -0.2793      -1.0697      0.09163

   openess_to_experience
0                -0.4455
1                 0.8637
2                 0.6721
3                -0.9194
4                -0.1295

[5 rows x 39 columns]
```

[6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3998 entries, 0 to 3997
Data columns (total 39 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Unnamed: 0             3998 non-null   object
 1   ID                     3998 non-null   int64
 2   Salary                 3998 non-null   float64
 3   DOJ                    3998 non-null   object
 4   DOL                    3998 non-null   object
 5   Designation            3998 non-null   object
 6   JobCity                3998 non-null   object
 7   Gender                 3998 non-null   object
 8   DOB                    3998 non-null   object
 9   10percentage           3998 non-null   float64
 10  10board                3998 non-null   object
 11  12graduation           3998 non-null   int64
 12  12percentage           3998 non-null   float64
 13  12board                3998 non-null   object
 14  CollegeID              3998 non-null   int64
```

```
 15  CollegeTier             3998 non-null   int64
 16  Degree                  3998 non-null   object
 17  Specialization          3998 non-null   object
 18  collegeGPA              3998 non-null   float64
 19  CollegeCityID           3998 non-null   int64
 20  CollegeCityTier         3998 non-null   int64
 21  CollegeState            3998 non-null   object
 22  GraduationYear          3998 non-null   int64
 23  English                 3998 non-null   int64
 24  Logical                 3998 non-null   int64
 25  Quant                   3998 non-null   int64
 26  Domain                  3998 non-null   float64
 27  ComputerProgramming     3998 non-null   int64
 28  ElectronicsAndSemicon   3998 non-null   int64
 29  ComputerScience         3998 non-null   int64
 30  MechanicalEngg          3998 non-null   int64
 31  ElectricalEngg          3998 non-null   int64
 32  TelecomEngg             3998 non-null   int64
 33  CivilEngg               3998 non-null   int64
 34  conscientiousness       3998 non-null   float64
 35  agreeableness           3998 non-null   float64
 36  extraversion            3998 non-null   float64
 37  nueroticism             3998 non-null   float64
 38  openess_to_experience   3998 non-null   float64
dtypes: float64(10), int64(17), object(12)
memory usage: 1.2+ MB
```

[7]:
```
print(df.shape)
```

```
(3998, 39)
```

[8]:
```
df.describe()
```

[8]:

	ID	Salary	10percentage	12graduation	12percentage	\
count	3.998000e+03	3.998000e+03	3998.000000	3998.000000	3998.000000	
mean	6.637945e+05	3.076998e+05	77.925443	2008.087544	74.466366	
std	3.632182e+05	2.127375e+05	9.850162	1.653599	10.999933	
min	1.124400e+04	3.500000e+04	43.000000	1995.000000	40.000000	
25%	3.342842e+05	1.800000e+05	71.680000	2007.000000	66.000000	
50%	6.396000e+05	3.000000e+05	79.150000	2008.000000	74.400000	
75%	9.904800e+05	3.700000e+05	85.670000	2009.000000	82.600000	
max	1.298275e+06	4.000000e+06	97.760000	2013.000000	98.700000	

	CollegeID	CollegeTier	collegeGPA	CollegeCityID	CollegeCityTier	\
count	3998.000000	3998.000000	3998.000000	3998.000000	3998.000000	
mean	5156.851426	1.925713	71.486171	5156.851426	0.300400	
std	4802.261482	0.262270	8.167338	4802.261482	0.458489	

```
min        2.000000     1.000000     6.450000       2.000000          0.000000
25%      494.000000     2.000000    66.407500     494.000000          0.000000
50%     3879.000000     2.000000    71.720000    3879.000000          0.000000
75%     8818.000000     2.000000    76.327500    8818.000000          1.000000
max    18409.000000     2.000000    99.930000   18409.000000          1.000000

       …  ComputerScience  MechanicalEngg  ElectricalEngg  TelecomEngg  \
count  …       3998.000000     3998.000000     3998.000000  3998.000000
mean   …         90.742371       22.974737       16.478739    31.851176
std    …        175.273083       98.123311       87.585634   104.852845
min    …         -1.000000       -1.000000       -1.000000    -1.000000
25%    …         -1.000000       -1.000000       -1.000000    -1.000000
50%    …         -1.000000       -1.000000       -1.000000    -1.000000
75%    …         -1.000000       -1.000000       -1.000000    -1.000000
max    …        715.000000      623.000000      676.000000   548.000000

         CivilEngg  conscientiousness  agreeableness  extraversion  \
count  3998.000000        3998.000000    3998.000000   3998.000000
mean      2.683842          -0.037831       0.146496      0.002763
std      36.658505           1.028666       0.941782      0.951471
min      -1.000000          -4.126700      -5.781600     -4.600900
25%      -1.000000          -0.713525      -0.287100     -0.604800
50%      -1.000000           0.046400       0.212400      0.091400
75%      -1.000000           0.702700       0.812800      0.672000
max     516.000000           1.995300       1.904800      2.535400

       nueroticism  openess_to_experience
count  3998.000000            3998.000000
mean     -0.169033              -0.138110
std       1.007580               1.008075
min      -2.643000              -7.375700
25%      -0.868200              -0.669200
50%      -0.234400              -0.094300
75%       0.526200               0.502400
max       3.352500               1.822400

[8 rows x 27 columns]
```

[9]:
```
df.nunique()
```

[9]:
```
Unnamed: 0             1
ID                  3998
Salary               177
DOJ                   81
DOL                   67
Designation          419
JobCity              339
```

```
Gender                      2
DOB                      1872
10percentage              851
10board                   275
12graduation               16
12percentage              801
12board                   340
CollegeID                1350
CollegeTier                 2
Degree                      4
Specialization             46
collegeGPA               1282
CollegeCityID            1350
CollegeCityTier             2
CollegeState               26
GraduationYear             11
English                   111
Logical                   107
Quant                     138
Domain                    243
ComputerProgramming        79
ElectronicsAndSemicon      29
ComputerScience            20
MechanicalEngg             42
ElectricalEngg             31
TelecomEngg                26
CivilEngg                  23
conscientiousness         141
agreeableness             149
extraversion              154
nueroticism               217
openess_to_experience     142
dtype: int64
```

1.0.1 Removing Unwanted Columns

[11]:
```
df = df.drop(columns=['Unnamed: 0', 'ID', 'CollegeID', 'CollegeCityID'])
df.head()
```

[11]:
```
      Salary           DOJ           DOL                Designation    JobCity  \
0   420000.0  6/1/12 0:00       present   senior quality engineer  Bangalore
1   500000.0  9/1/13 0:00       present         assistant manager     Indore
2   325000.0  6/1/14 0:00       present          systems engineer    Chennai
3  1100000.0  7/1/11 0:00       present  senior software engineer    Gurgaon
4   200000.0  3/1/14 0:00  3/1/15 0:00                       get    Manesar

  Gender           DOB  10percentage                          10board  \
```

```
0      f  2/19/90 0:00          84.3  board ofsecondary education,ap
1      m  10/4/89 0:00          85.4                             cbse
2      f   8/3/92 0:00          85.0                             cbse
3      m  12/5/89 0:00          85.6                             cbse
4      m  2/27/91 0:00          78.0                             cbse

   12graduation  …  ComputerScience MechanicalEngg  ElectricalEngg  \
0          2007  …               -1             -1              -1
1          2007  …               -1             -1              -1
2          2010  …               -1             -1              -1
3          2007  …               -1             -1              -1
4          2008  …               -1             -1              -1

  TelecomEngg CivilEngg  conscientiousness  agreeableness extraversion  \
0          -1        -1             0.9737         0.8128       0.5269
1          -1        -1            -0.7335         0.3789       1.2396
2          -1        -1             0.2718         1.7109       0.1637
3          -1        -1             0.0464         0.3448      -0.3440
4          -1        -1            -0.8810        -0.2793      -1.0697

   nueroticism  openess_to_experience
0      1.35490                -0.4455
1     -0.10760                 0.8637
2     -0.86820                 0.6721
3     -0.40780                -0.9194
4      0.09163                -0.1295

[5 rows x 35 columns]
```

1.0.2 Data type conversion

In the DOL column some have responded as **PRESENT**. So, we need to replace the **PRESENT** value in DOL with Date(2024-10-01).

Then we convert the datetype of DOJ and DOL to datetime.

[160]:
```
df['DOL'].replace('present', '2020-08-18')
df['DOL'] = pd.to_datetime(df['DOL'], format='mixed')
df['DOJ'] = pd.to_datetime(df['DOJ'])
df['DOB']= pd.to_datetime(df['DOB'])
df.head()
```

[160]:
```
      Salary        DOJ        DOL                Designation    JobCity  \
0   420000.0 2012-06-01 2020-08-18                      other  Bangalore
1   500000.0 2013-09-01 2020-08-18                      other      other
2   325000.0 2014-06-01 2020-08-18           systems engineer    Chennai
3  1100000.0 2011-07-01 2020-08-18  senior software engineer    Gurgaon
4   200000.0 2014-03-01 2015-03-01                      other      other
```

```
  Gender        DOB  10percentage 10board  12graduation  …  Quant    Domain  \
0      f 1990-02-19          84.3   other          2007  …    525  0.635979
1      m 1989-10-04          85.4    cbse          2007  …    780  0.960603
2      f 1992-08-03          85.0    cbse          2010  …    370  0.450877
3      m 1989-12-05          85.6    cbse          2007  …    625  0.974396
4      m 1991-02-27          78.0    cbse          2008  …    465  0.124502

   ComputerProgramming ElectronicsAndSemicon ComputerScience  \
0                445.0                    0                0
1                  NaN                  466                0
2                395.0                    0                0
3                615.0                    0                0
4                  NaN                  233                0

   conscientiousness  agreeableness extraversion  nueroticism  \
0             0.9737         0.8128       0.5269      1.35490
1            -0.7335         0.3789       1.2396     -0.10760
2             0.2718         1.7109       0.1637     -0.86820
3             0.0464         0.3448      -0.3440     -0.40780
4            -0.8810        -0.2793      -1.0697      0.09163

   openess_to_experience
0                -0.4455
1                 0.8637
2                 0.6721
3                -0.9194
4                -0.1295

[5 rows x 31 columns]
```

1.0.3 Checking if the DOL (Date of leaving) is actually greater than DOJ (Date of joining).

[15]:
```
dates = df[(df['DOL'] < df['DOJ'])].shape[0]
print(f'DOL is earlier than DOJ for {dates} observations.')
print(df.shape)
```

```
DOL is earlier than DOJ for 40 observations.
(3998, 35)
```

[16]:
```
df = df.drop(df[~(df['DOL'] > df['DOJ'])].index)
print(df.shape)
```

```
(3943, 35)
```

1.0.4 Validating 10, 12 percentage and college CGPA

[18]:
```
print((df['10percentage'] <=10).sum())
print((df['12percentage'] <=10).sum())
print((df['collegeGPA'] <=10).sum())
```

```
0
0
12
```

1.0.5 Converting the 12 entries of College GPA to percentage

[20]:
```
df.loc[df['collegeGPA']<=10,'collegeGPA'] = (df.
  ↪loc[df['collegeGPA']<=10,'collegeGPA']/10)*100
df.head()
```

[20]:

	Salary	DOJ	DOL	Designation	JobCity	\
0	420000.0	2012-06-01	2020-08-18	senior quality engineer	Bangalore	
1	500000.0	2013-09-01	2020-08-18	assistant manager	Indore	
2	325000.0	2014-06-01	2020-08-18	systems engineer	Chennai	
3	1100000.0	2011-07-01	2020-08-18	senior software engineer	Gurgaon	
4	200000.0	2014-03-01	2015-03-01	get	Manesar	

	Gender	DOB	10percentage	10board	\
0	f	1990-02-19	84.3	board ofsecondary education,ap	
1	m	1989-10-04	85.4	cbse	
2	f	1992-08-03	85.0	cbse	
3	m	1989-12-05	85.6	cbse	
4	m	1991-02-27	78.0	cbse	

	12graduation	…	ComputerScience	MechanicalEngg	ElectricalEngg	\
0	2007	…	-1	-1	-1	
1	2007	…	-1	-1	-1	
2	2010	…	-1	-1	-1	
3	2007	…	-1	-1	-1	
4	2008	…	-1	-1	-1	

	TelecomEngg	CivilEngg	conscientiousness	agreeableness	extraversion	\
0	-1	-1	0.9737	0.8128	0.5269	
1	-1	-1	-0.7335	0.3789	1.2396	
2	-1	-1	0.2718	1.7109	0.1637	
3	-1	-1	0.0464	0.3448	-0.3440	
4	-1	-1	-0.8810	-0.2793	-1.0697	

	nueroticism	openess_to_experience
0	1.35490	-0.4455
1	-0.10760	0.8637

```
2     -0.86820                0.6721
3     -0.40780               -0.9194
4      0.09163               -0.1295

[5 rows x 35 columns]
```

1.0.6 Dropping the rows where the graduationyear is greater than or equal to date of joining

[22]:
```
len(df[(df['GraduationYear'] > df['DOJ'].dt.year)].index)
```

[22]:
```
79
```

[23]:
```
df = df.drop(df[(df['GraduationYear'] > df['DOJ'].dt.year)].index)
df
```

[23]:
```
         Salary        DOJ        DOL                  Designation  \
0      420000.0 2012-06-01 2020-08-18      senior quality engineer
1      500000.0 2013-09-01 2020-08-18            assistant manager
2      325000.0 2014-06-01 2020-08-18             systems engineer
3     1100000.0 2011-07-01 2020-08-18     senior software engineer
4      200000.0 2014-03-01 2015-03-01                          get
…           …        …          …                          …
3992   800000.0 2014-04-01 2015-04-01                      manager
3993   280000.0 2011-10-01 2012-10-01            software engineer
3995   320000.0 2013-07-01 2020-08-18  associate software engineer
3996   200000.0 2014-07-01 2015-01-01           software developer
3997   400000.0 2013-02-01 2020-08-18      senior systems engineer

               JobCity Gender        DOB  10percentage  \
0            Bangalore      f 1990-02-19         84.30
1               Indore      m 1989-10-04         85.40
2              Chennai      f 1992-08-03         85.00
3              Gurgaon      m 1989-12-05         85.60
4              Manesar      m 1991-02-27         78.00
…                  …    …         …            …
3992            Rajkot      m 1990-06-22         73.00
3993        New Delhi       m 1987-04-15         52.09
3995         Bangalore      m 1991-07-03         81.86
3996  Asifabadbanglore      f 1992-03-20         78.72
3997           Chennai      f 1991-02-26         70.60

                            10board  12graduation  …  ComputerScience  \
0     board ofsecondary education,ap          2007  …               -1
1                               cbse          2007  …               -1
2                               cbse          2010  …               -1
3                               cbse          2007  …               -1
```

```
4                              cbse          2008  …               -1
…                               …             …  …              …
3992                              0          2008  …               -1
3993                           cbse          2006  …               -1
3995                   bse,odisha            2008  …               -1
3996                  state board            2010  …              438
3997                           cbse          2008  …               -1

     MechanicalEngg  ElectricalEngg TelecomEngg CivilEngg  conscientiousness  \
0                -1              -1          -1        -1             0.9737
1                -1              -1          -1        -1            -0.7335
2                -1              -1          -1        -1             0.2718
3                -1              -1          -1        -1             0.0464
4                -1              -1          -1        -1            -0.8810
…              …               …         …       …                …
3992             -1              -1          -1       480             0.3555
3993             -1              -1          -1        -1            -0.1082
3995             -1              -1          -1        -1            -1.5765
3996             -1              -1          -1        -1            -0.1590
3997             -1              -1          -1        -1            -1.1128

      agreeableness extraversion  nueroticism  openess_to_experience
0            0.8128       0.5269      1.35490                -0.4455
1            0.3789       1.2396     -0.10760                 0.8637
2            1.7109       0.1637     -0.86820                 0.6721
3            0.3448      -0.3440     -0.40780                -0.9194
4           -0.2793      -1.0697      0.09163                -0.1295
…             …           …           …                    …
3992        -0.9033       0.9623      0.64983                -0.4229
3993         0.3448       0.2366      0.64980                -0.9194
3995        -1.5273      -1.5051     -1.31840                -0.7615
3996         0.0459      -0.4511     -0.36120                -0.0943
3997        -0.2793      -0.6343      1.32553                -0.6035

[3864 rows x 35 columns]
```

1.0.7 Checking if there are any 0 or -1 values

[25]:
```
(df == 0).sum()
```

[25]:
```
Salary                   0
DOJ                      0
DOL                      0
Designation              0
JobCity                  0
Gender                   0
DOB                      0
```

```
10percentage               0
10board                    0
12graduation               0
12percentage               0
12board                    0
CollegeTier                0
Degree                     0
Specialization             0
collegeGPA                 0
CollegeCityTier         2706
CollegeState               0
GraduationYear             1
English                    0
Logical                    0
Quant                      0
Domain                     0
ComputerProgramming        0
ElectronicsAndSemicon      0
ComputerScience            0
MechanicalEngg             0
ElectricalEngg             0
TelecomEngg                0
CivilEngg                  0
conscientiousness          0
agreeableness              0
extraversion               0
nueroticism                0
openess_to_experience      0
dtype: int64
```

[26]:
```
print((df==0).sum()[(df==0).sum() > 0])
```

```
CollegeCityTier    2706
GraduationYear        1
dtype: int64
```

[27]:
```
df1 = (df==-1).sum()[(df==-1).sum()>0]
df1
```

[27]:
```
Domain                   232
ComputerProgramming      842
ElectronicsAndSemicon   2754
ComputerScience         3001
MechanicalEngg          3640
ElectricalEngg          3717
TelecomEngg             3495
CivilEngg               3825
```

```
dtype: int64
```

[28]:
```
df1/len(df)*100
```

[28]:
```
Domain                    6.004141
ComputerProgramming      21.790890
ElectronicsAndSemicon    71.273292
ComputerScience          77.665631
MechanicalEngg           94.202899
ElectricalEngg           96.195652
TelecomEngg              90.450311
CivilEngg                98.990683
dtype: float64
```

From the above columns we can observe that few subjects are having large number of -1(Null values). So, we will be dropping the columns in which the percentage of -1 values is greater than or equal to 80%. And for rest of the columns we will assign the value as 0.

[30]:
```
df = df.drop(columns =␣
 ↪['MechanicalEngg','ElectricalEngg','TelecomEngg','CivilEngg'])
```

[31]:
```
df['10board'] = df['10board'].replace({'0':np.nan})
df['12board'] = df['12board'].replace({'0':np.nan})
df['GraduationYear'] = df['GraduationYear'].replace({0:np.nan})
df['JobCity'] = df['JobCity'].replace({'-1':np.nan})
df['Domain'] = df['Domain'].replace({-1:np.nan})
df['ElectronicsAndSemicon'] = df['ElectronicsAndSemicon'].replace({-1:0})
df['ComputerScience'] = df['ComputerScience'].replace({-1:0})
df['ComputerProgramming'] = df['ComputerProgramming'].replace({-1:np.nan})
```

[32]:
```
df['10board'].fillna(df['10board'].mode()[0])
df['12board'].fillna(df['12board'].mode()[0])
df['GraduationYear'].fillna(df['GraduationYear'].mode()[0])
df['JobCity'].fillna(df['JobCity'].mode()[0])

df.head()
```

[32]:

	Salary	DOJ	DOL	Designation	JobCity	\
0	420000.0	2012-06-01	2020-08-18	senior quality engineer	Bangalore	
1	500000.0	2013-09-01	2020-08-18	assistant manager	Indore	
2	325000.0	2014-06-01	2020-08-18	systems engineer	Chennai	
3	1100000.0	2011-07-01	2020-08-18	senior software engineer	Gurgaon	
4	200000.0	2014-03-01	2015-03-01	get	Manesar	

	Gender	DOB	10percentage	10board	\
0	f	1990-02-19	84.3	board ofsecondary education,ap	

```
1      m 1989-10-04          85.4                            cbse
2      f 1992-08-03          85.0                            cbse
3      m 1989-12-05          85.6                            cbse
4      m 1991-02-27          78.0                            cbse

   12graduation  …  Quant    Domain  ComputerProgramming  \
0          2007  …    525  0.635979                445.0
1          2007  …    780  0.960603                  NaN
2          2010  …    370  0.450877                395.0
3          2007  …    625  0.974396                615.0
4          2008  …    465  0.124502                  NaN

  ElectronicsAndSemicon ComputerScience  conscientiousness  agreeableness  \
0                     0               0             0.9737         0.8128
1                   466               0            -0.7335         0.3789
2                     0               0             0.2718         1.7109
3                     0               0             0.0464         0.3448
4                   233               0            -0.8810        -0.2793

  extraversion  nueroticism  openess_to_experience
0       0.5269      1.35490                -0.4455
1       1.2396     -0.10760                 0.8637
2       0.1637     -0.86820                 0.6721
3      -0.3440     -0.40780                -0.9194
4      -1.0697      0.09163                -0.1295

[5 rows x 31 columns]
```

[33]:
```
df['Domain'].fillna(df['Domain'].median())
df['ComputerProgramming'].fillna(df['ComputerProgramming'].median())
```

[33]:
```
0       445.0
1       455.0
2       395.0
3       615.0
4       455.0
        …
3992    455.0
3993    345.0
3995    405.0
3996    445.0
3997    435.0
Name: ComputerProgramming, Length: 3864, dtype: float64
```

[34]:
```
df.head()
```

[34]:
```
      Salary        DOJ        DOL                Designation    JobCity  \
0   420000.0 2012-06-01 2020-08-18   senior quality engineer  Bangalore
1   500000.0 2013-09-01 2020-08-18         assistant manager     Indore
2   325000.0 2014-06-01 2020-08-18          systems engineer    Chennai
3  1100000.0 2011-07-01 2020-08-18  senior software engineer    Gurgaon
4   200000.0 2014-03-01 2015-03-01                       get    Manesar

  Gender        DOB  10percentage                          10board  \
0      f 1990-02-19          84.3  board ofsecondary education,ap
1      m 1989-10-04          85.4                              cbse
2      f 1992-08-03          85.0                              cbse
3      m 1989-12-05          85.6                              cbse
4      m 1991-02-27          78.0                              cbse

   12graduation  …  Quant    Domain  ComputerProgramming  \
0          2007  …    525  0.635979                445.0
1          2007  …    780  0.960603                  NaN
2          2010  …    370  0.450877                395.0
3          2007  …    625  0.974396                615.0
4          2008  …    465  0.124502                  NaN

  ElectronicsAndSemicon ComputerScience  conscientiousness  agreeableness  \
0                     0                0             0.9737         0.8128
1                   466                0            -0.7335         0.3789
2                     0                0             0.2718         1.7109
3                     0                0             0.0464         0.3448
4                   233                0            -0.8810        -0.2793

  extraversion  nueroticism  openess_to_experience
0       0.5269      1.35490                -0.4455
1       1.2396     -0.10760                 0.8637
2       0.1637     -0.86820                 0.6721
3      -0.3440     -0.40780                -0.9194
4      -1.0697      0.09163                -0.1295

[5 rows x 31 columns]
```

1.0.8 Outliers in each Numerical column

[36]:
```
numerical_df = df.select_dtypes(include='number')
Q1 = numerical_df.quantile(0.25)
Q3 = numerical_df.quantile(0.75)
IQR = Q3 - Q1
outliers = ((numerical_df < (Q1 - 1.5 * IQR)) | (numerical_df > (Q3 + 1.5 *␣
 ↪IQR))).sum()
print(f"Outliers in each numerical column:\n{outliers}")
```

```
Outliers in each numerical column:
Salary                    103
10percentage               29
12graduation               41
12percentage                1
CollegeTier               288
collegeGPA                 27
CollegeCityTier             0
GraduationYear              1
English                    13
Logical                    17
Quant                      24
Domain                      0
ComputerProgramming        42
ElectronicsAndSemicon       2
ComputerScience           863
conscientiousness          37
agreeableness             116
extraversion               40
nueroticism                14
openess_to_experience      91
dtype: int64
```

[37]:
```
textual_columns =␣
 ↪['Designation','JobCity','10board','12board','Specialization','CollegeState']
for col in textual_columns:
    print(f'Number of unique values in {col} with inconsistency : {df[col].
 ↪nunique()}')
```

```
Number of unique values in Designation with inconsistency : 413
Number of unique values in JobCity with inconsistency : 329
Number of unique values in 10board with inconsistency : 271
Number of unique values in 12board with inconsistency : 335
Number of unique values in Specialization with inconsistency : 42
Number of unique values in CollegeState with inconsistency : 26
```

[ ]:
```

```

Since the number of categories are more in number, we keep the top 10 categories.

[39]:
```
def collapsing_categories(df, data):
    min_count = df[data].value_counts()[:10].min()
    for Designation in df[data].unique():
        counts = df[df[data] == Designation][data].value_counts()
        if not counts.empty and counts.iloc[0] < min_count:
            df.loc[df[data] == Designation, data] = 'other'
```

[40]:
```
for cols in textual_columns:
    collapsing_categories(df, cols)
```

[41]:
```
for cols in textual_columns:
    print('')
    print('Top 10 categories in:', cols)
    print('')
    print(df[cols].value_counts())
    print('')
    print('*'*100)
```

```
Top 10 categories in: Designation

Designation
other                          2205
software engineer               525
software developer              258
system engineer                 201
programmer analyst              137
systems engineer                116
java software engineer          108
software test engineer           98
project engineer                 73
technical support engineer       72
senior software engineer         71
Name: count, dtype: int64

****************************************************************************************************

Top 10 categories in: JobCity

JobCity
other        960
Bangalore    608
Noida        354
Hyderabad    324
Pune         283
Chennai      269
Gurgaon      190
New Delhi    190
Mumbai       108
Kolkata       96
Jaipur        43
Name: count, dtype: int64
```

```
********************************************************************************
********************

Top 10 categories in: 10board

10board
cbse                            1343
state board                     1115
other                            473
icse                             271
ssc                              121
up board                          83
matriculation                     38
rbse                              21
board of secondary education      20
up                                18
mp board                          17
Name: count, dtype: int64

********************************************************************************
********************

Top 10 categories in: 12board

12board
cbse                              1344
state board                       1205
other                              586
icse                               127
up board                            85
isc                                 44
board of intermediate               36
board of intermediate education     31
up                                  19
mp board                            17
rbse                                17
Name: count, dtype: int64

********************************************************************************
********************

Top 10 categories in: Specialization

Specialization
electronics and communication engineering    856
computer science & engineering               714
information technology                       649
computer engineering                         582
```

```
computer application                          232
other                                         222
mechanical engineering                        194
electronics and electrical engineering        185
electronics & telecommunications              119
electrical engineering                         79
electronics & instrumentation eng              32
Name: count, dtype: int64

**********************************************************************************
********************

Top 10 categories in: CollegeState

CollegeState
Uttar Pradesh     888
other             754
Karnataka         359
Tamil Nadu        359
Telangana         307
Maharashtra       252
Andhra Pradesh    219
West Bengal       188
Madhya Pradesh    187
Punjab            177
Haryana           174
Name: count, dtype: int64

**********************************************************************************
********************
```

[42]:
```
df
```

[42]:
```
          Salary        DOJ        DOL                Designation    JobCity  \
0      420000.0 2012-06-01 2020-08-18                      other  Bangalore
1      500000.0 2013-09-01 2020-08-18                      other      other
2      325000.0 2014-06-01 2020-08-18           systems engineer    Chennai
3     1100000.0 2011-07-01 2020-08-18  senior software engineer    Gurgaon
4      200000.0 2014-03-01 2015-03-01                      other      other
…           …          …          …                          …          …
3992   800000.0 2014-04-01 2015-04-01                      other      other
3993   280000.0 2011-10-01 2012-10-01          software engineer      other
3995   320000.0 2013-07-01 2020-08-18                      other  Bangalore
3996   200000.0 2014-07-01 2015-01-01         software developer      other
3997   400000.0 2013-02-01 2020-08-18                      other    Chennai

     Gender        DOB  10percentage      10board  12graduation  …  Quant  \
```

```
0         f 1990-02-19         84.30        other          2007  …    525
1         m 1989-10-04         85.40         cbse          2007  …    780
2         f 1992-08-03         85.00         cbse          2010  …    370
3         m 1989-12-05         85.60         cbse          2007  …    625
4         m 1991-02-27         78.00         cbse          2008  …    465
…     …        …           …          …           …  …    …
3992      m 1990-06-22         73.00          NaN          2008  …    525
3993      m 1987-04-15         52.09         cbse          2006  …    475
3995      m 1991-07-03         81.86        other          2008  …    465
3996      f 1992-03-20         78.72  state board          2010  …    320
3997      f 1991-02-26         70.60         cbse          2008  …    464

        Domain  ComputerProgramming ElectronicsAndSemicon ComputerScience  \
0     0.635979                445.0                     0                0
1     0.960603                  NaN                   466                0
2     0.450877                395.0                     0                0
3     0.974396                615.0                     0                0
4     0.124502                  NaN                   233                0
…        …                    …                       …            …
3992  0.938588                  NaN                     0                0
3993  0.276047                345.0                     0                0
3995  0.488348                405.0                     0                0
3996  0.744758                445.0                     0              438
3997  0.600057                435.0                     0                0

      conscientiousness  agreeableness extraversion  nueroticism  \
0                0.9737         0.8128       0.5269      1.35490
1               -0.7335         0.3789       1.2396     -0.10760
2                0.2718         1.7109       0.1637     -0.86820
3                0.0464         0.3448      -0.3440     -0.40780
4               -0.8810        -0.2793      -1.0697      0.09163
…                  …             …           …           …
3992             0.3555        -0.9033       0.9623      0.64983
3993            -0.1082         0.3448       0.2366      0.64980
3995            -1.5765        -1.5273      -1.5051     -1.31840
3996            -0.1590         0.0459      -0.4511     -0.36120
3997            -1.1128        -0.2793      -0.6343      1.32553

      openess_to_experience
0                   -0.4455
1                    0.8637
2                    0.6721
3                   -0.9194
4                   -0.1295
…                     …
3992                -0.4229
3993                -0.9194
```

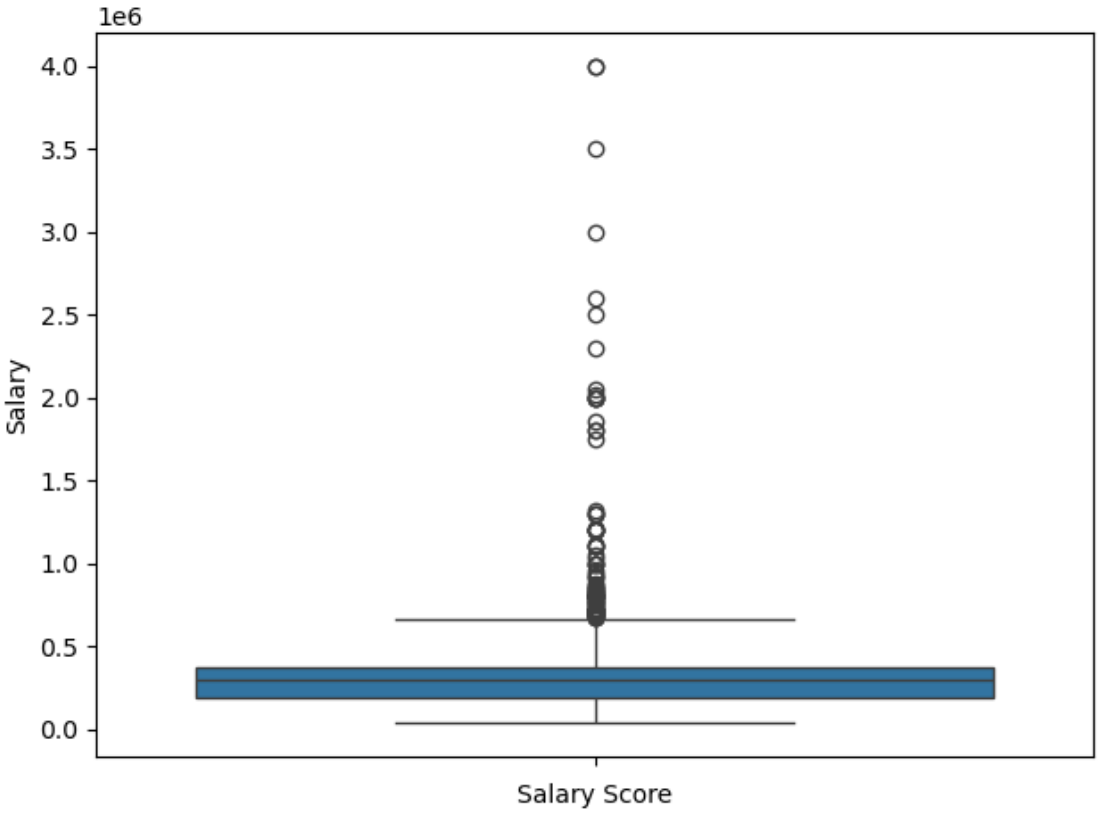
```
3995                -0.7615
3996                -0.0943
3997                -0.6035

[3864 rows x 31 columns]
```

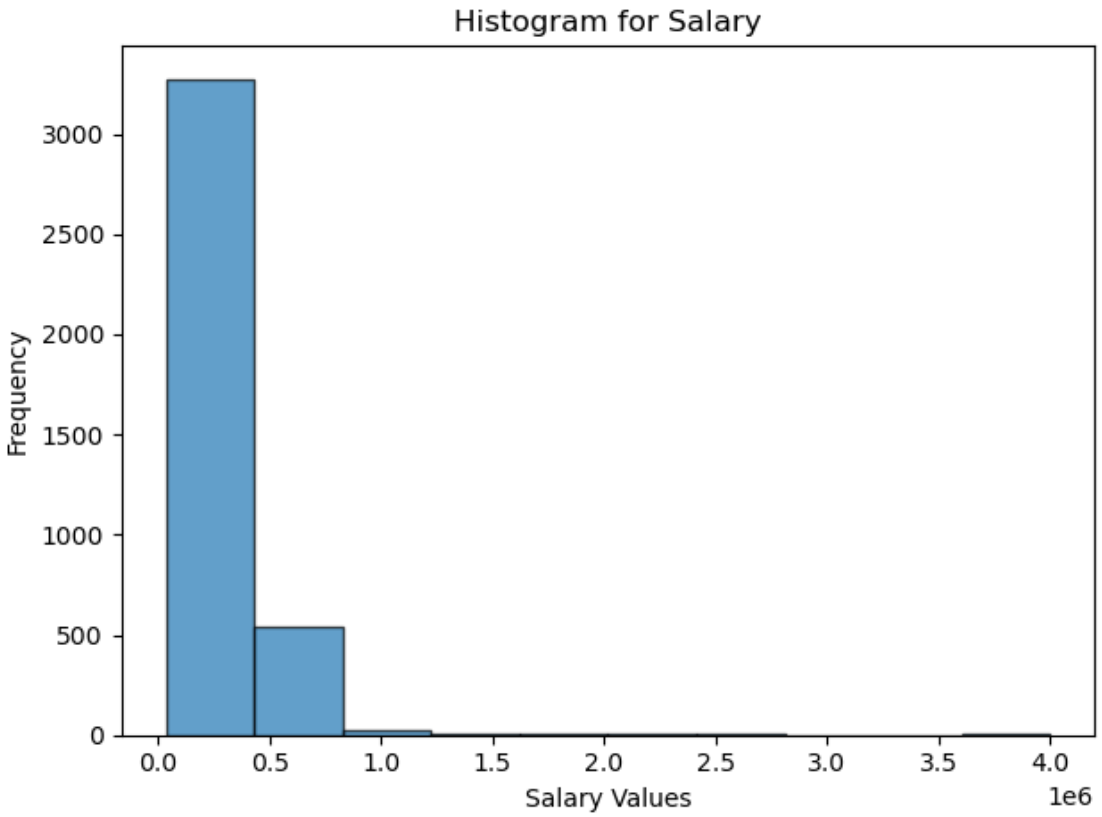
1.1 Univariate Analysis

[44]:
```
colors = plt.cm.viridis(np.linspace(0, 1, 10))
```

[45]:
```
# Summary plot
df['Salary'].describe()[1:].plot( alpha=0.8, marker='D', markersize=8)
plt.title(f'Summary Statistics for {'Salary'}')
plt.xlabel('Statistical Measures')
plt.tight_layout()
plt.show()

# Boxplot
sns.boxplot(df['Salary'])
plt.xlabel(f'{'Salary'} Score')
plt.tight_layout()
plt.show()

# Histogram
plt.hist(df['Salary'].dropna(), bins=10, alpha=0.7, edgecolor='black')
plt.title(f'Histogram for {'Salary'}')
plt.xlabel(f'{'Salary'} Values')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()
```


Summary Statistics for Salary
1e6
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
mean
std
min
25%
50%
75%
max
Statistical Measures


1e6
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Salary
Salary Score



1.2 Conclusions:

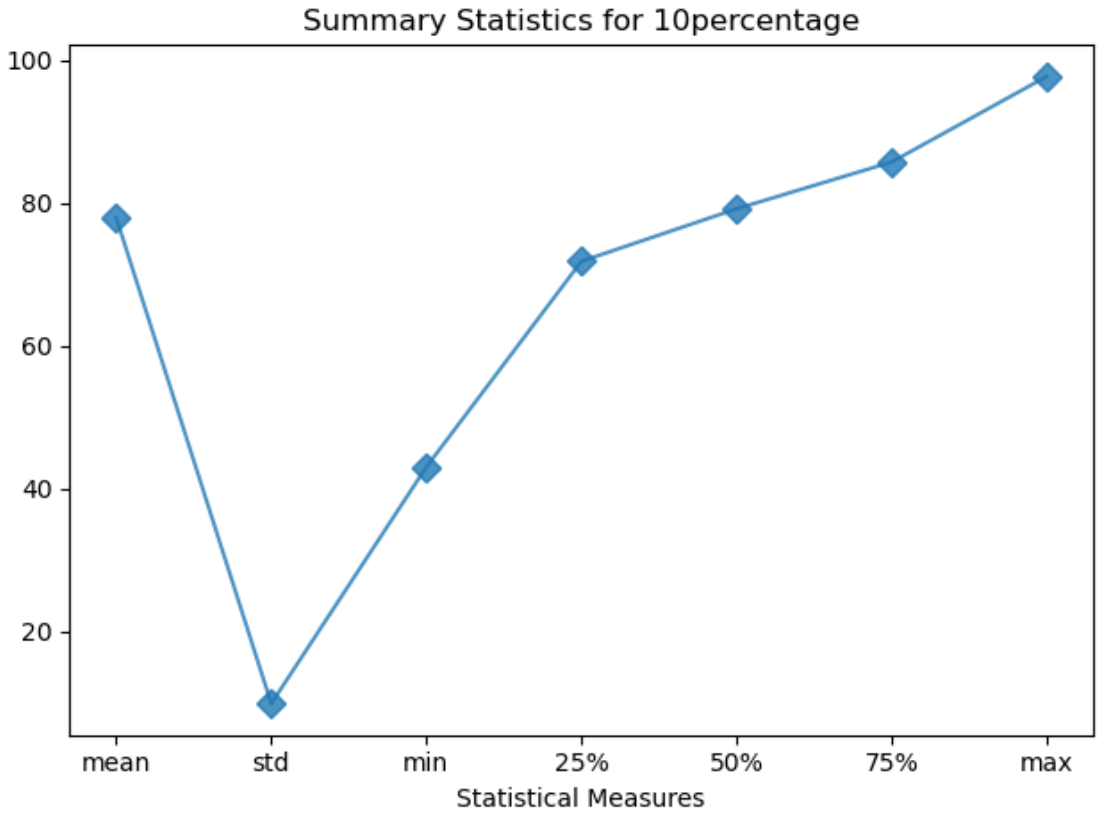
Summary Plot : There is high variation in salary..

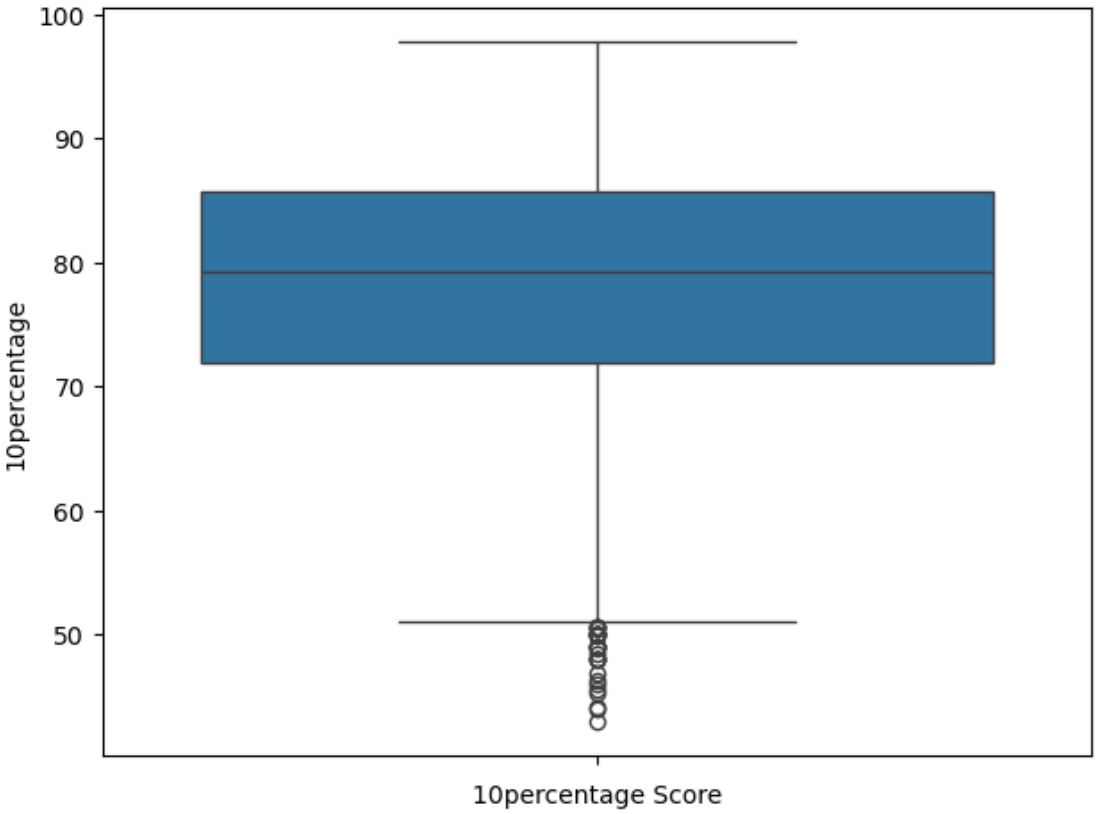
Histogram : The data is positively and highly skewed with skewness 6(approx) which is large as compared to that of normal(0). Mean, median and mode all are approximately equal.

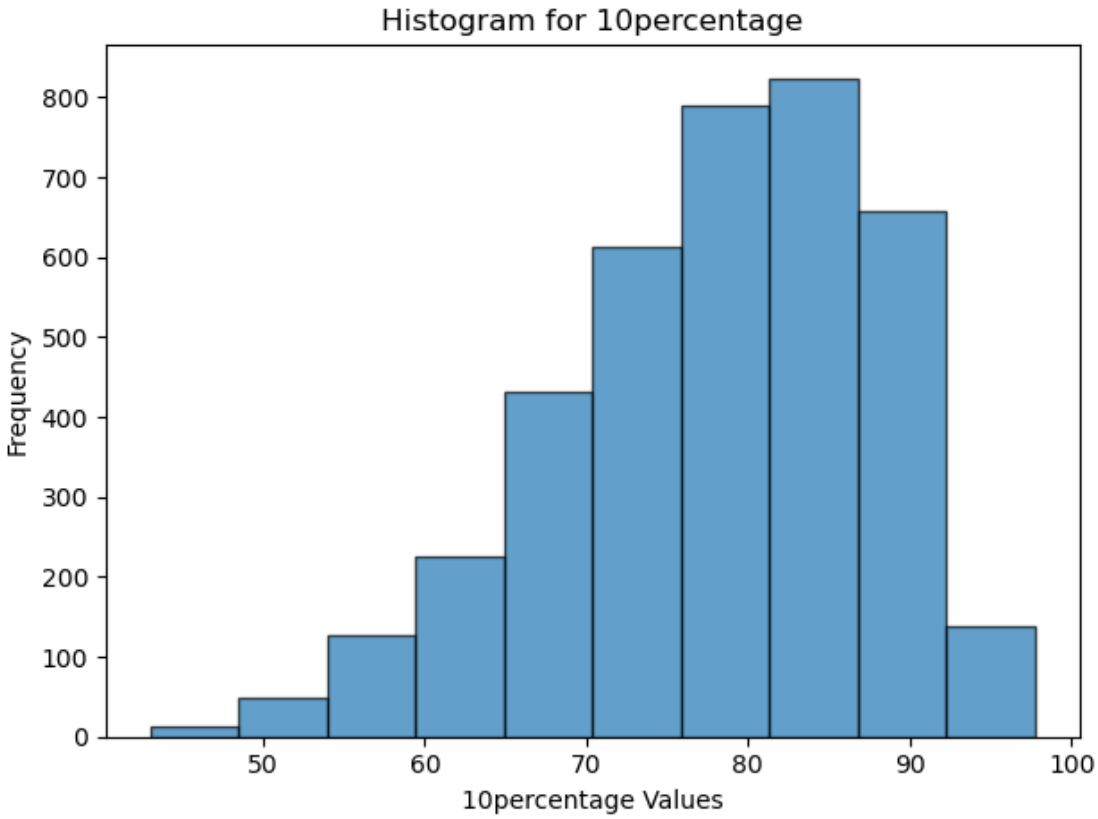
Box Plot : There are large number of data points with high salaries.

[46]:
```
# Summary plot
df['10percentage'].describe()[1:].plot( alpha=0.8, marker='D', markersize=8)
plt.title(f'Summary Statistics for {'10percentage'}')
plt.xlabel('Statistical Measures')
plt.tight_layout()
plt.show()

# Boxplot
sns.boxplot(df['10percentage'])
plt.xlabel(f'{'10percentage'} Score')
plt.tight_layout()
plt.show()
```

```
# Histogram
plt.hist(df['10percentage'].dropna(), bins=10, alpha=0.7,edgecolor='black')
plt.title(f'Histogram for {'10percentage'}')
plt.xlabel(f'{'10percentage'} Values')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()
```




100
90
80
70
60
50
10percentage
10percentage Score



1.3 Conclusion:

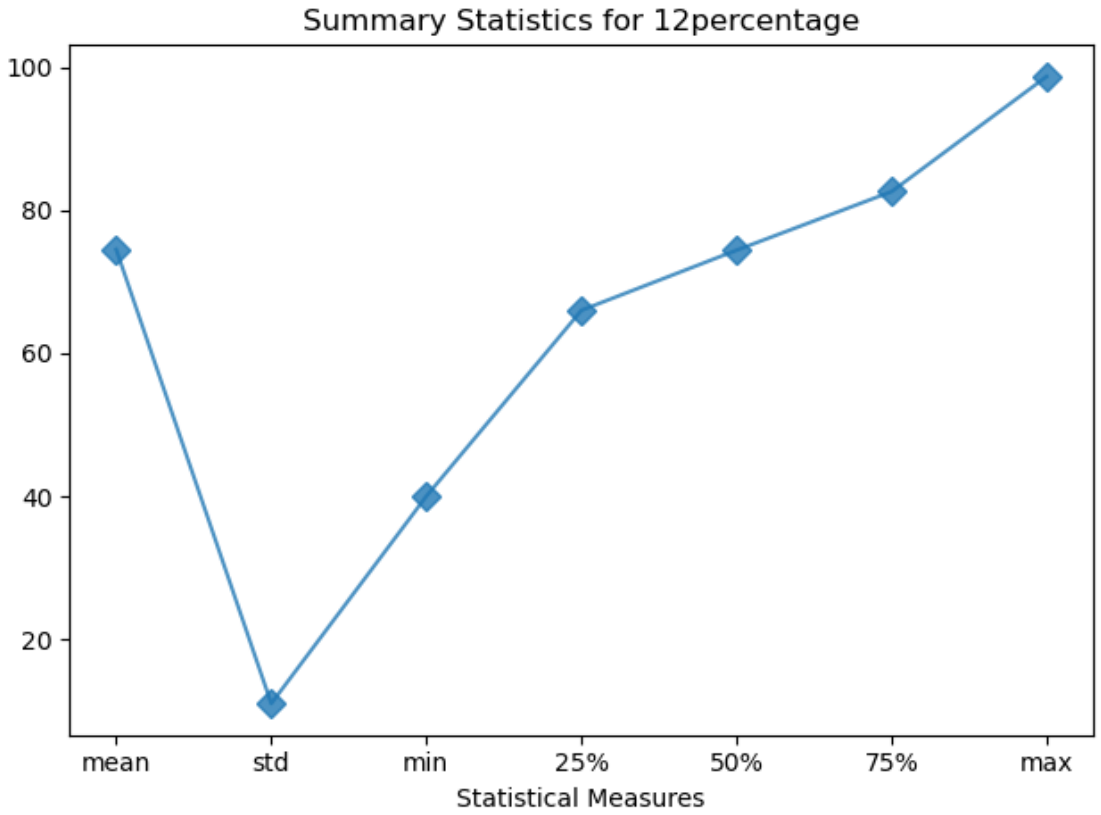
Summary Plot : 50% of students scored less than approximately 80%.

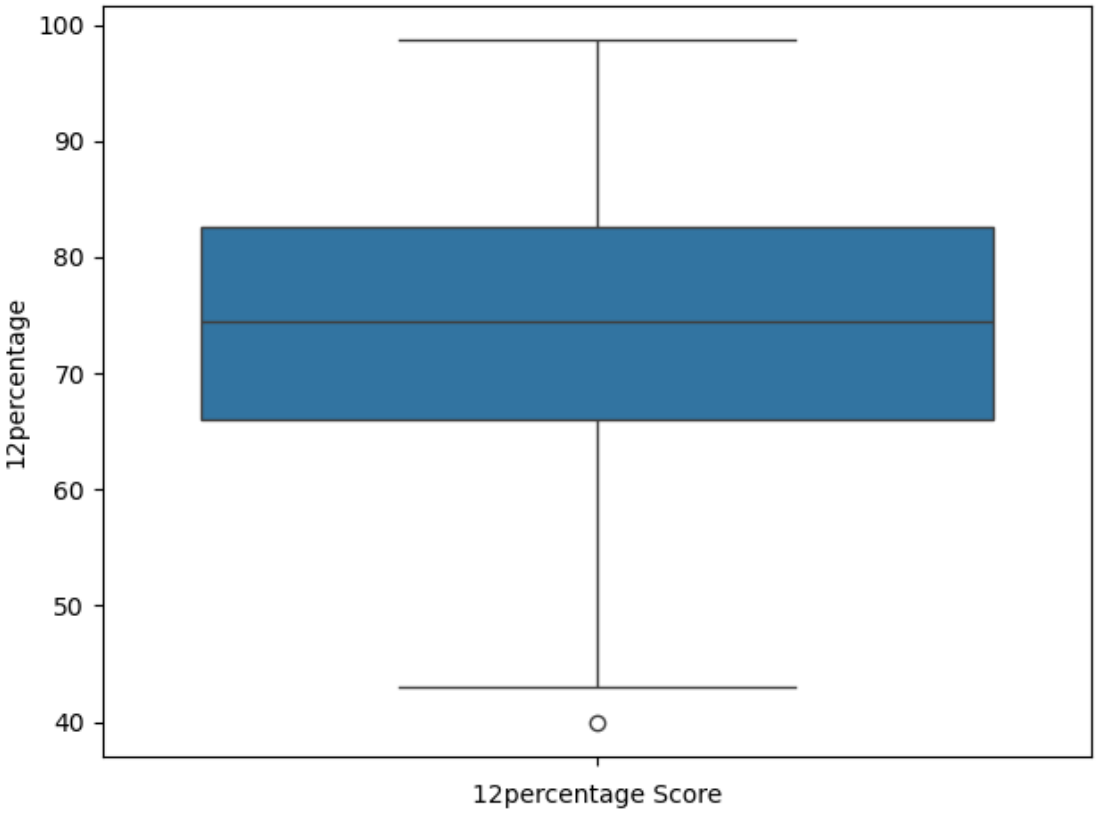
Histogram : There are very less students with low % and the majority of the students scored b/w 75% - 90%. Maximum number of students scored 78% and on average the score was 77%.

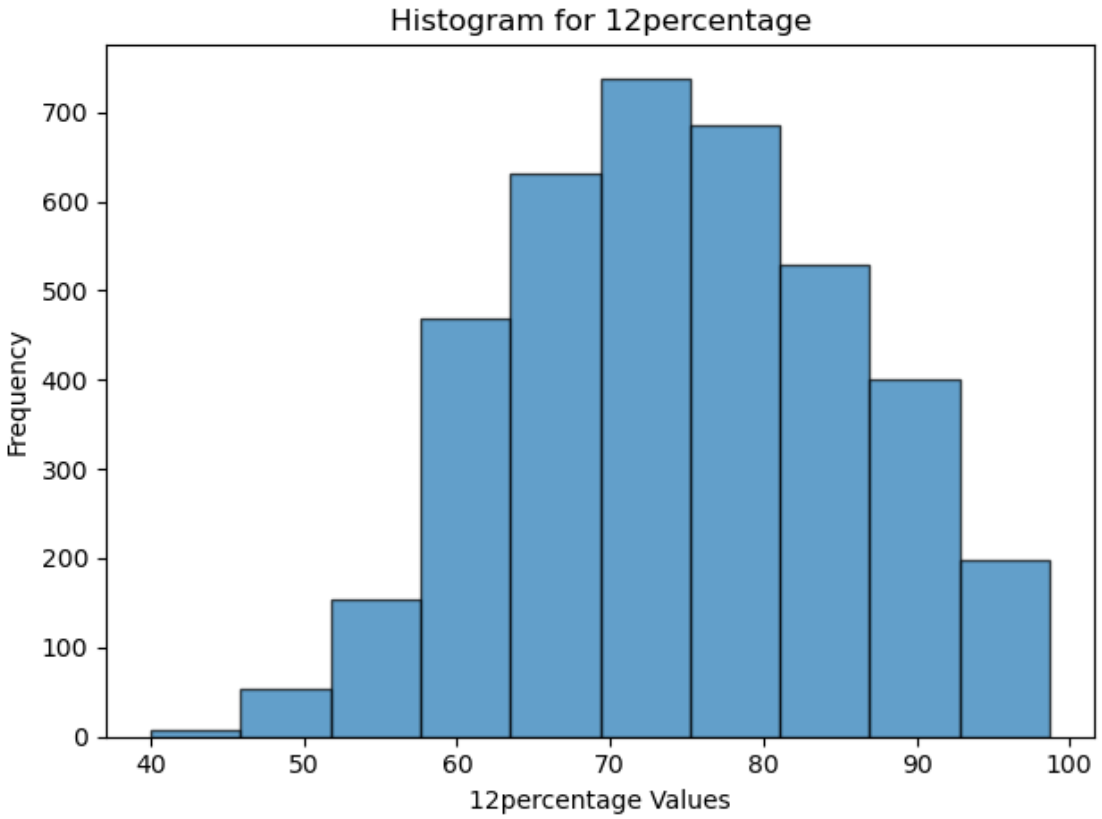
Box Plot : The box plot shows that there are few very outliers.

[47]:
```
# Summary plot
df['12percentage'].describe()[1:].plot( alpha=0.8, marker='D', markersize=8)
plt.title(f'Summary Statistics for {'12percentage'}')
plt.xlabel('Statistical Measures')
plt.tight_layout()
plt.show()

# Boxplot
sns.boxplot(df['12percentage'])
plt.xlabel(f'{'12percentage'} Score')
plt.tight_layout()
plt.show()
```

```
# Histogram
plt.hist(df['12percentage'].dropna(), bins=10, alpha=0.7,edgecolor='black')
plt.title(f'Histogram for {'12percentage'}')
plt.xlabel(f'{'12percentage'} Values')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()
```




100
90
80
70
60
50
40
12percentage
12percentage Score



1.4 Conclusions:

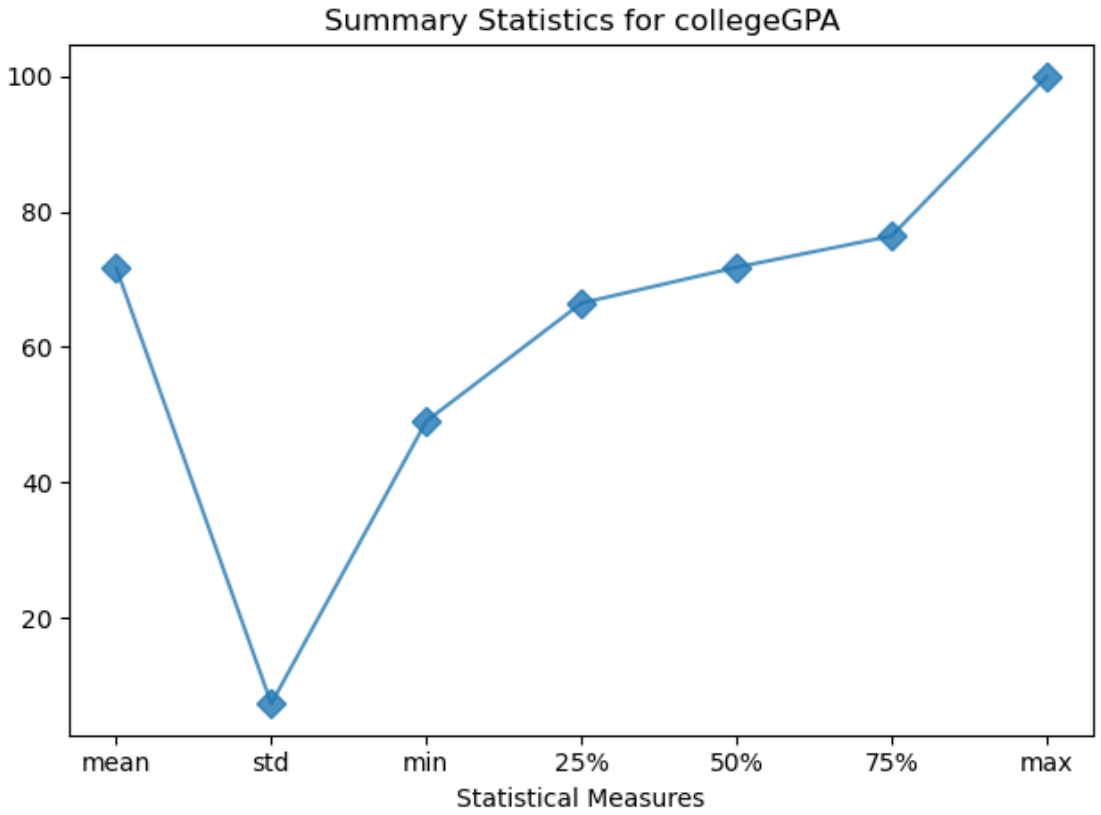
Summary Plot : 50% of students scored less than approximately 78%.

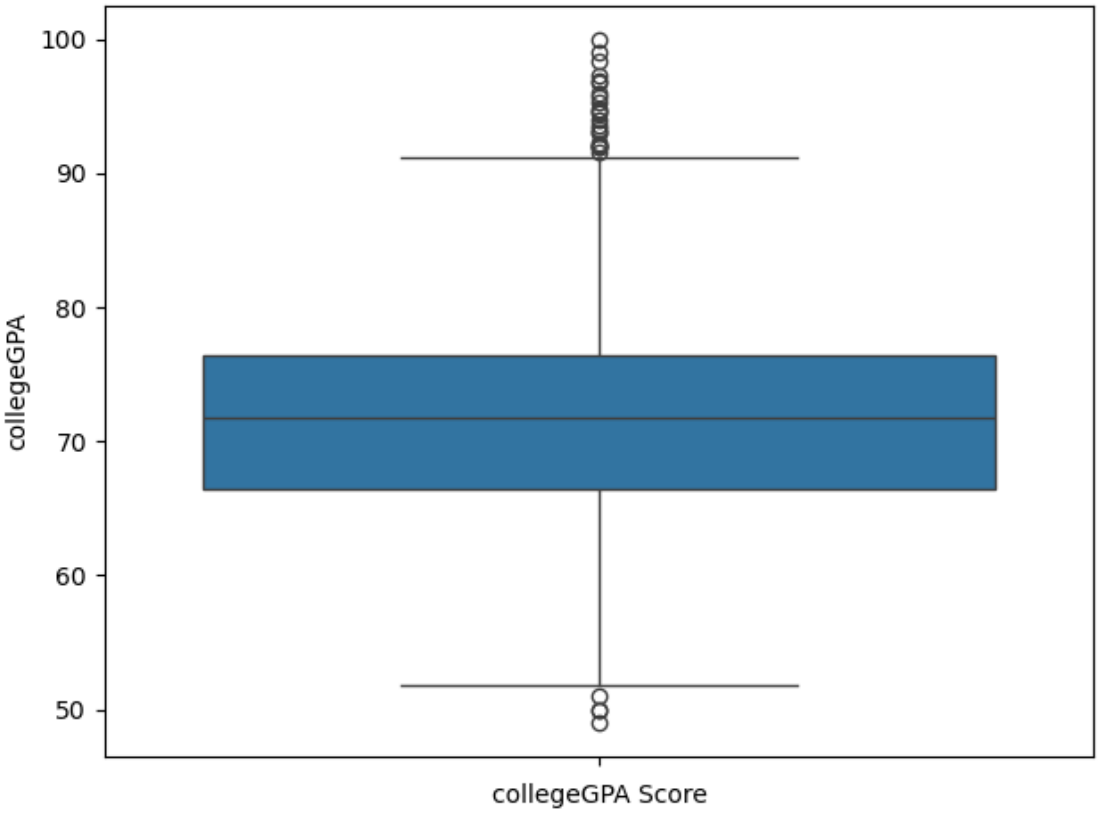
Histogram : There are very less students with low % and the majority of the students scored b/w 69% - 84%. Maximum number of students scored 70% and on average the score was 74%.

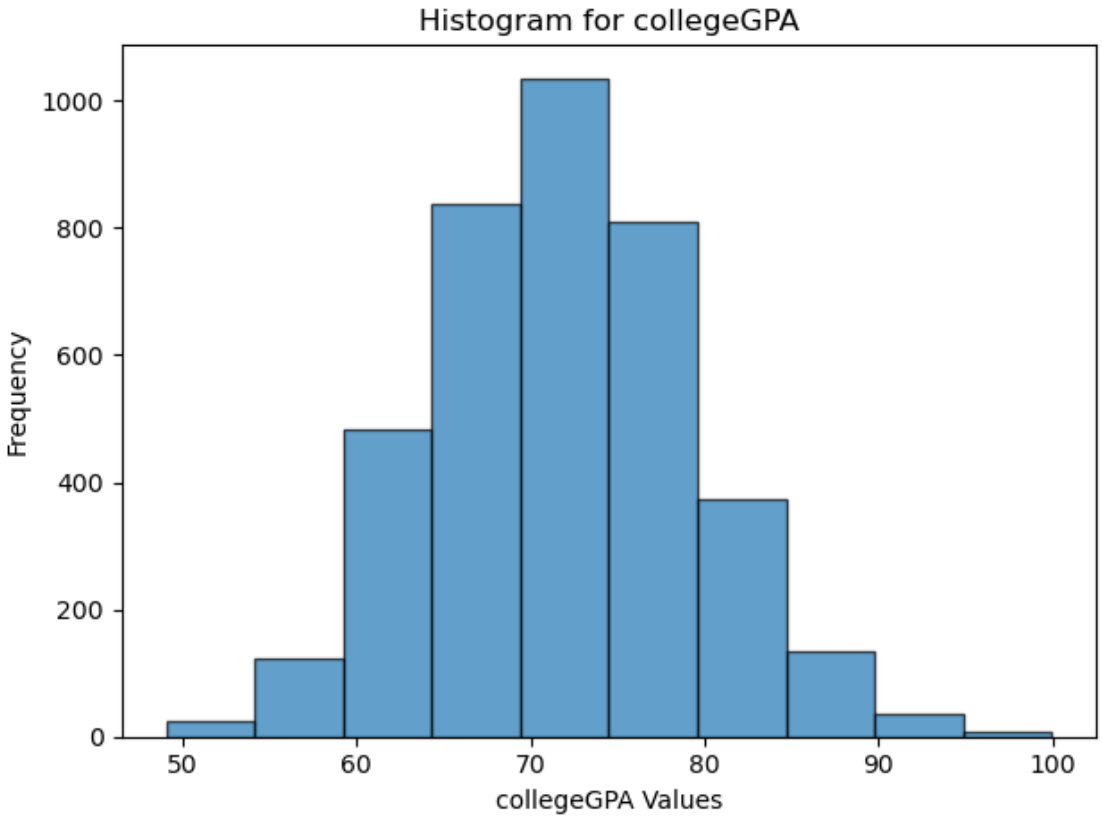
Box Plot : The box plot shows that there is only data point with extreamly low score.

[48]:
```
# Summary plot
df['collegeGPA'].describe()[1:].plot( alpha=0.8, marker='D', markersize=8)
plt.title(f'Summary Statistics for {'collegeGPA'}')
plt.xlabel('Statistical Measures')
plt.tight_layout()
plt.show()

# Boxplot
sns.boxplot(df['collegeGPA'])
plt.xlabel(f'{'collegeGPA'} Score')
plt.tight_layout()
plt.show()
```

```
# Histogram
plt.hist(df['collegeGPA'].dropna(), bins=10, alpha=0.7,edgecolor='black')
plt.title(f'Histogram for {'collegeGPA'}')
plt.xlabel(f'{'collegeGPA'} Values')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()
```




100
90
80
70
60
50
collegeGPA
collegeGPA Score



1.5 Conclusions

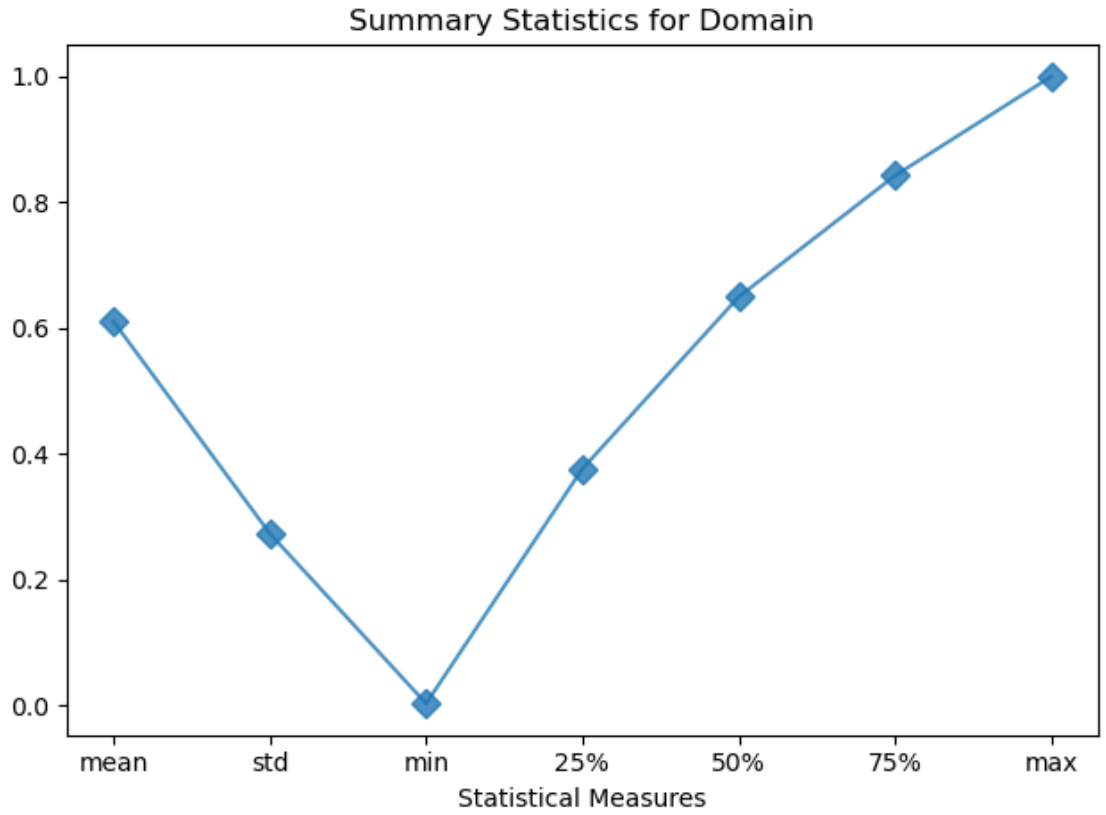
Summary Plot : 75% of students GPA was less than approximately 80%.

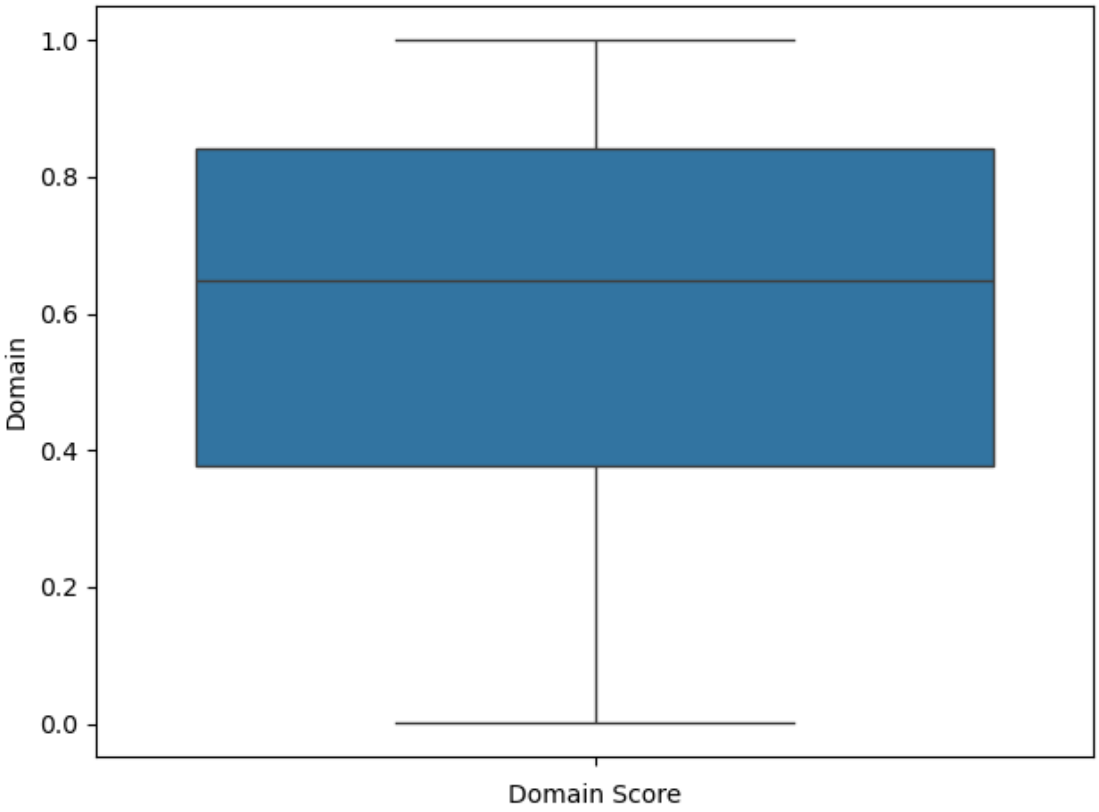
Histogram : Majority of the students GPA were in b/w 63% - 78%. Maximum number of students scored 70% and on average GPA score was 74%.

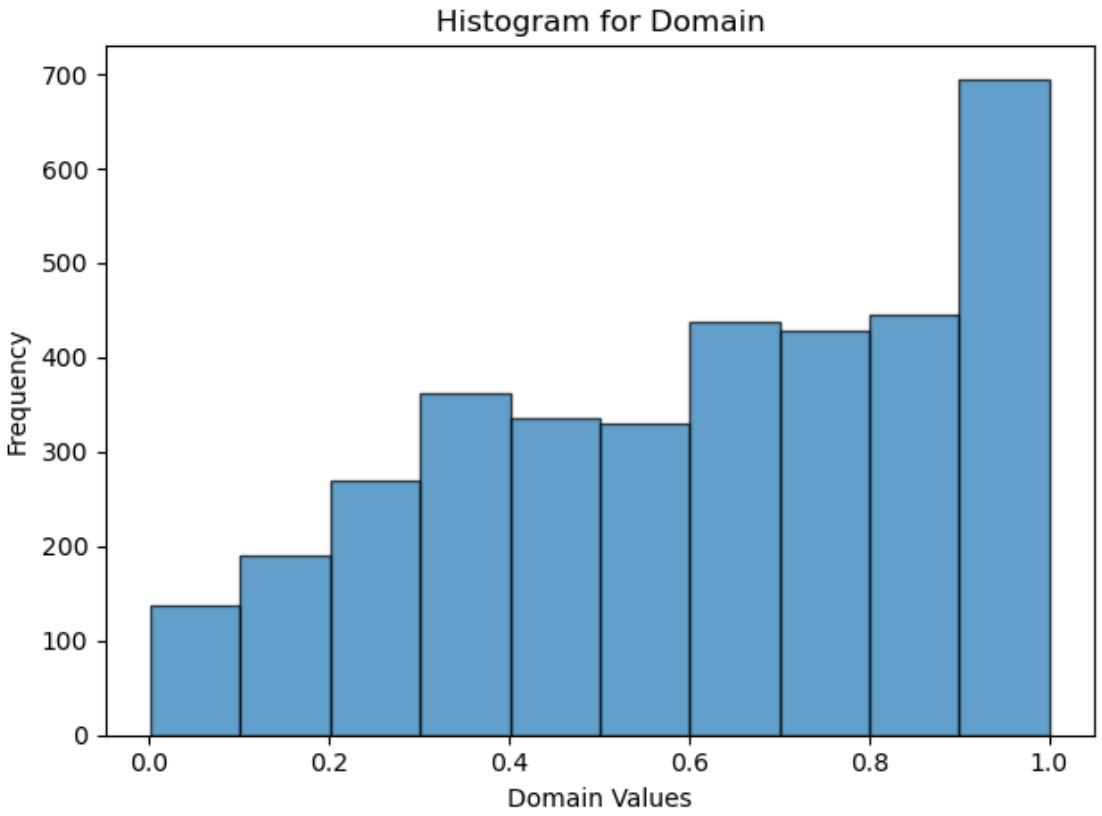
Box Plot : The box plot shows that there exist low extreme values as well as high extreme values.

[49]:
```
# Summary plot
df['Domain'].describe()[1:].plot( alpha=0.8, marker='D', markersize=8)
plt.title(f'Summary Statistics for {'Domain'}')
plt.xlabel('Statistical Measures')
plt.tight_layout()
plt.show()

# Boxplot
sns.boxplot(df['Domain'])
plt.xlabel(f'{'Domain'} Score')
plt.tight_layout()
plt.show()
```

```
# Histogram
plt.hist(df['Domain'].dropna(), bins=10, alpha=0.7,edgecolor='black')
plt.title(f'Histogram for {'Domain'}')
plt.xlabel(f'{'Domain'} Values')
plt.ylabel('Frequency')
plt.tight_layout()
plt.show()
```

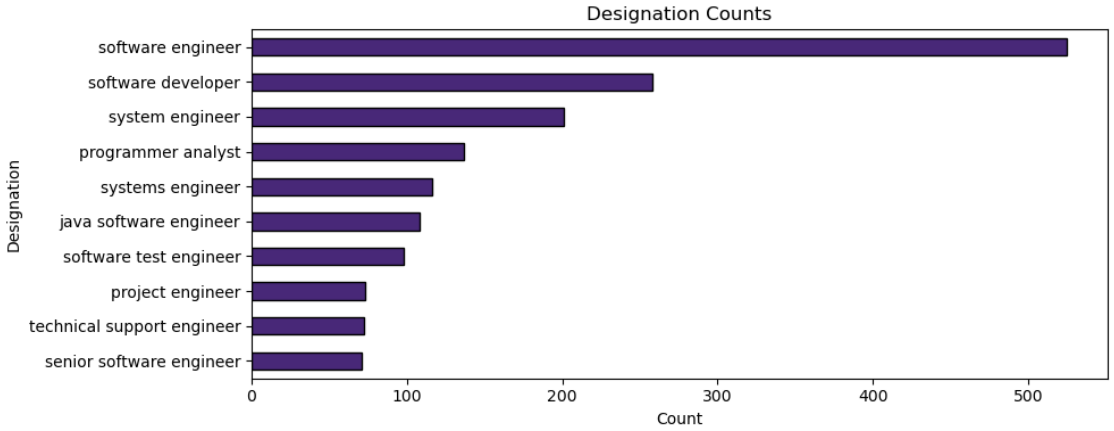



1.0
0.8
0.6
0.4
0.2
0.0
Domain
Domain Score



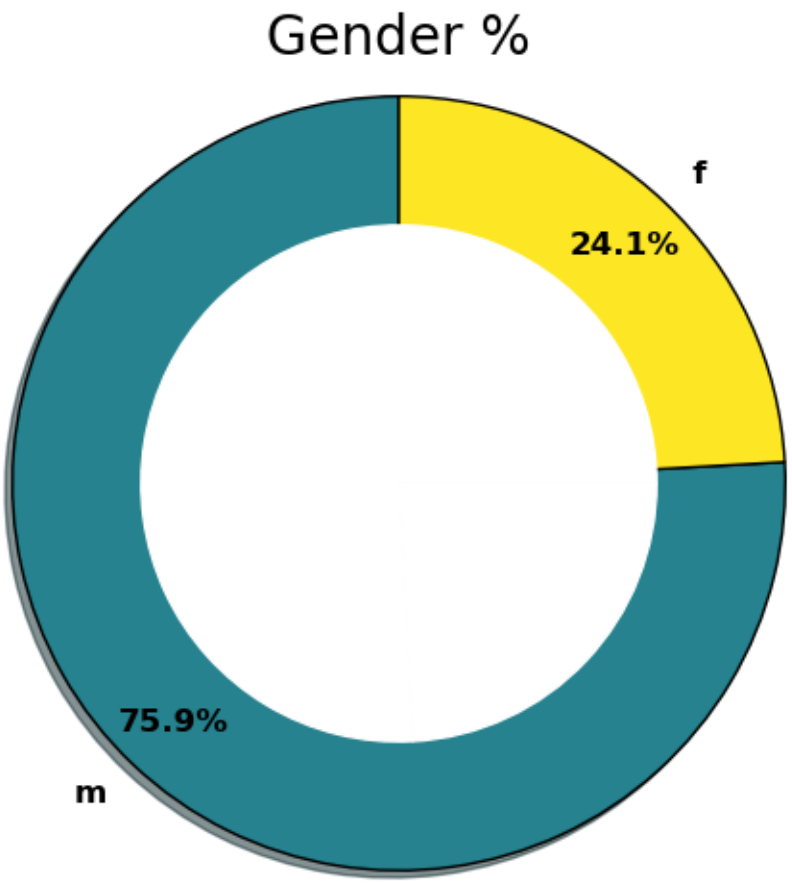
2 Categorical Features

[51]:
```
df['Designation'].value_counts()[1:].sort_values(ascending = True).plot(kind =␣
↪'barh',
                                                                        color =␣
↪colors[1],
                                                                        title =␣
↪'Designation Counts',
                                                                        figsize␣
↪= (10,4),
                                                                        ec = 'k')
plt.ylabel('Designation')
plt.xlabel('Count')
plt.tight_layout()
plt.show()
```



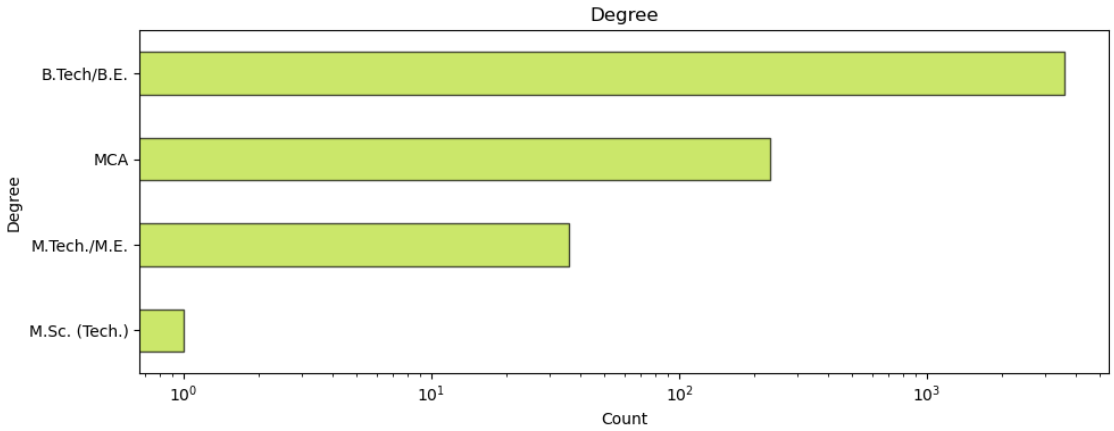
CONCLUSION : Software engineer is the most common desgination of all, followed by system engineer and software developer. NOTE : This graphs the most common designations. There exists OTHER category too.

[52]:
```
plt.pie(df['Gender'].value_counts().tolist(), labels = df['Gender'].
 ↪value_counts().index,
       colors = [colors[4],colors[9]],
       autopct = '%1.1f%%',
       radius = 1.5,
       wedgeprops = {'edgecolor':'k'},
       textprops = {'fontsize':12,'fontweight':'bold'},
       shadow = True,
       #explode = [0.1,0],
       startangle = 90,
       pctdistance = 0.85)
plt.pie(df['Gender'].value_counts().tolist(), colors = ['white'],
        wedgeprops = {'edgecolor':'white'},
       radius = 1)
plt.title('Gender %',pad = 40, size = 20)
plt.tight_layout()
plt.show()
```



CONCLUSION : The dataset is not balanced in terms of gender as the population of Male is really larger as compared to the female one.

[139]:
```
df['Degree'].value_counts().sort_values(ascending = True).plot(kind = 'barh',
                                                                color =␣
 ↪colors[8],
                                                                title =␣
 ↪'Degree',
                                                                figsize␣
 ↪= (10,4),
                                                                ec = 'k',
                                                            alpha = 0.7)
plt.ylabel('Degree')
plt.xlabel('Count')
plt.xscale('log')
plt.tight_layout()
plt.show()
```



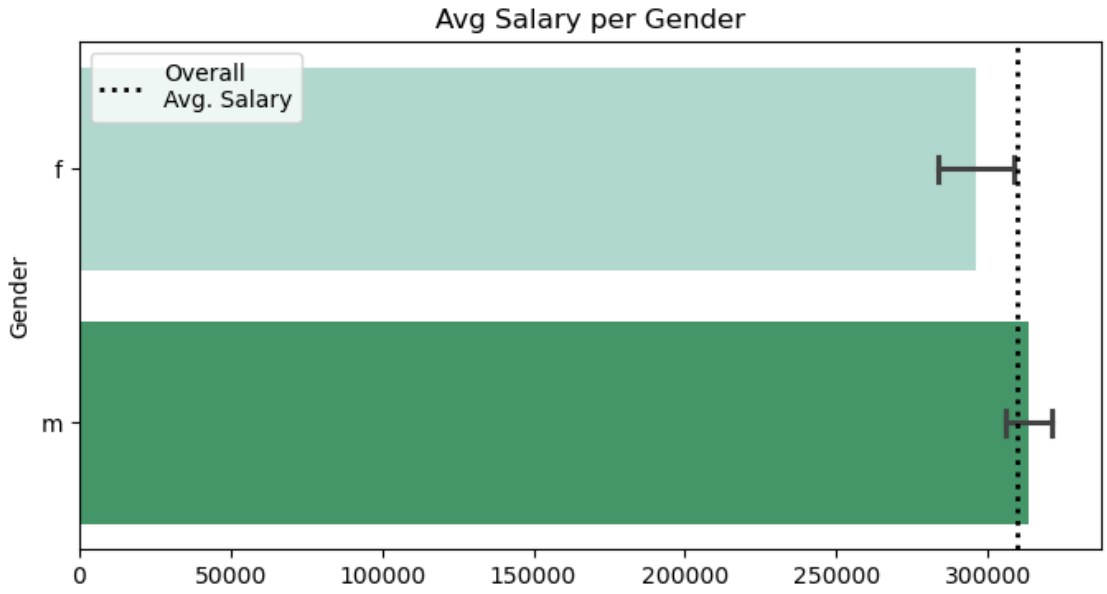
CONCLUSION : Most of the students have done their graduation in B.Tech and there are very less students from M.Sc(Tech)

2.1 Bivariate Analysis

2.1.1 Categorical vs Numerical

[55]:
```
fig, ax = plt.subplots(figsize = (8,4))
sns.barplot(x = 'Salary', y = 'Gender',data = df,palette = 'BuGn',hue =␣
 ↪'Gender',capsize = 0.1,ax = ax)
ax.axvline(df['Salary'].mean(), color = 'k',
           linestyle = ':',
           linewidth = 2, label = 'Overall\nAvg. Salary')
ax.set_title('Avg Salary per Gender')
ax.legend()
ax.set_xlabel('')
```

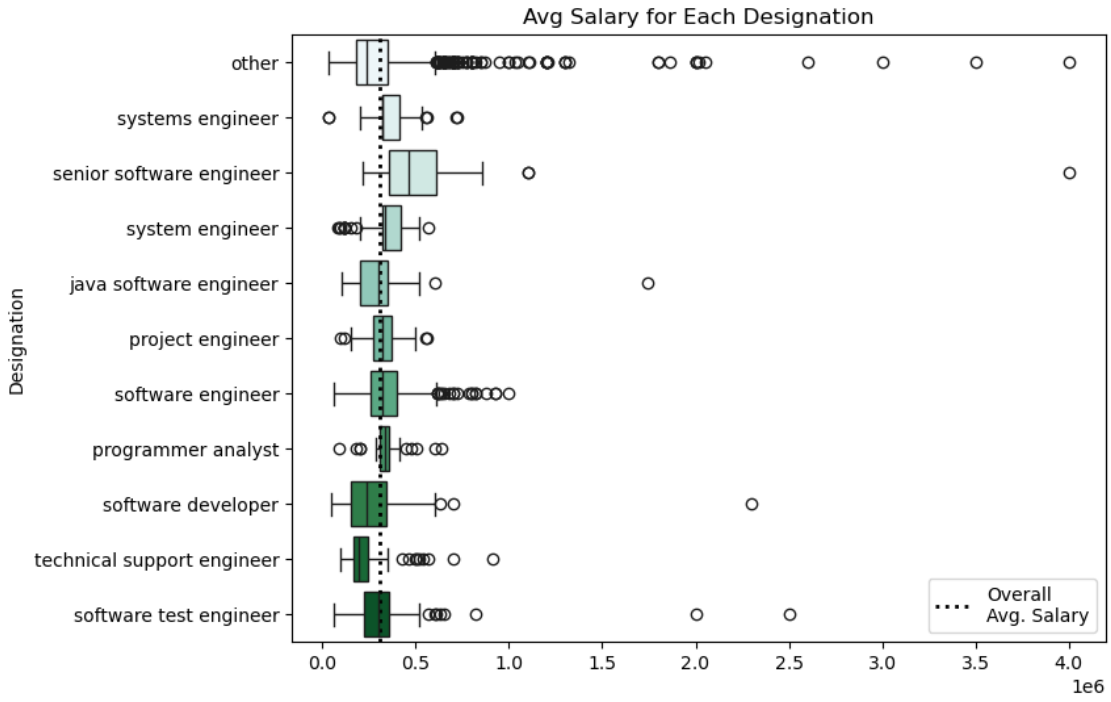
[55]:
```
Text(0.5, 0, '')
```



CONCLUSION: The average salary for both male and female is approximately equal and it implies that there was no gender bias in terms of salary.

[56]:
```
fig, ax = plt.subplots(figsize = (8,6), sharex = True)
sns.boxplot(x = 'Salary', y = 'Designation',data = df,palette = 'BuGn',hue␣
 ↪='Designation',ax = ax)
ax.axvline(df['Salary'].mean(), color = 'k',
           linestyle = ':',
           linewidth = 2, label = 'Overall\nAvg. Salary')
ax.set_title('Avg Salary for Each Designation')
ax.legend()
ax.set_xlabel('')
```

[56]:
```
Text(0.5, 0, '')
```

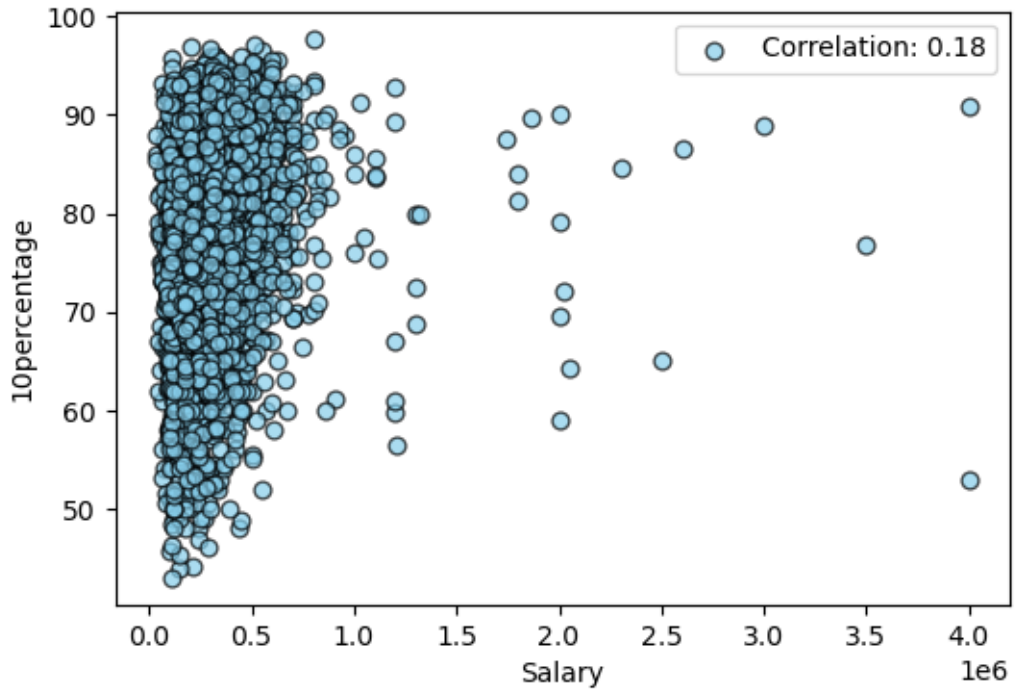


CONCLUSION : Bar plot shows the maximum salary for each designation. Senior Software Engineer has the highest salary but they also has the maximum standard deviation in their salary. There are only two designations namely, software developer and technical support engineer who has salary lower than average salary.

2.1.2 Numerical vs Numerical

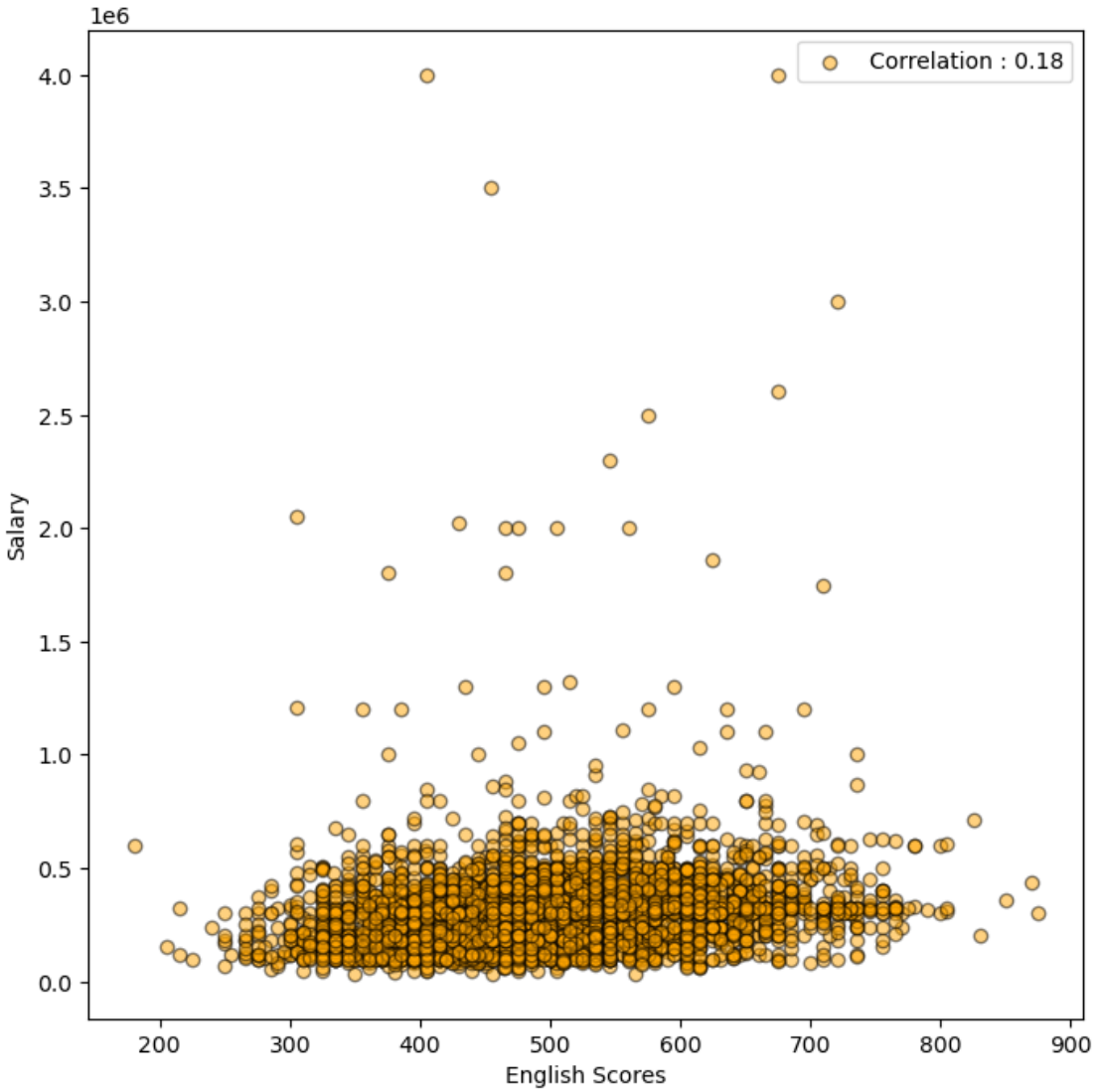
[58]:
```
fig, ax = plt.subplots(figsize = (6,4), sharex = True, sharey = True)

ax.scatter(df['Salary'],df['10percentage'],
                ec = 'k',
                color = 'skyblue',
                alpha = 0.7,
                s = 40,
                label = f"Correlation: {round(df[['Salary','10percentage']].
 ↪corr().iloc[1,0],2)}"
                )
ax.set_xlabel('Salary')
ax.set_ylabel('10percentage')
ax.legend()
```

[58]:
```
<matplotlib.legend.Legend at 0x1c0f0303e00>
```



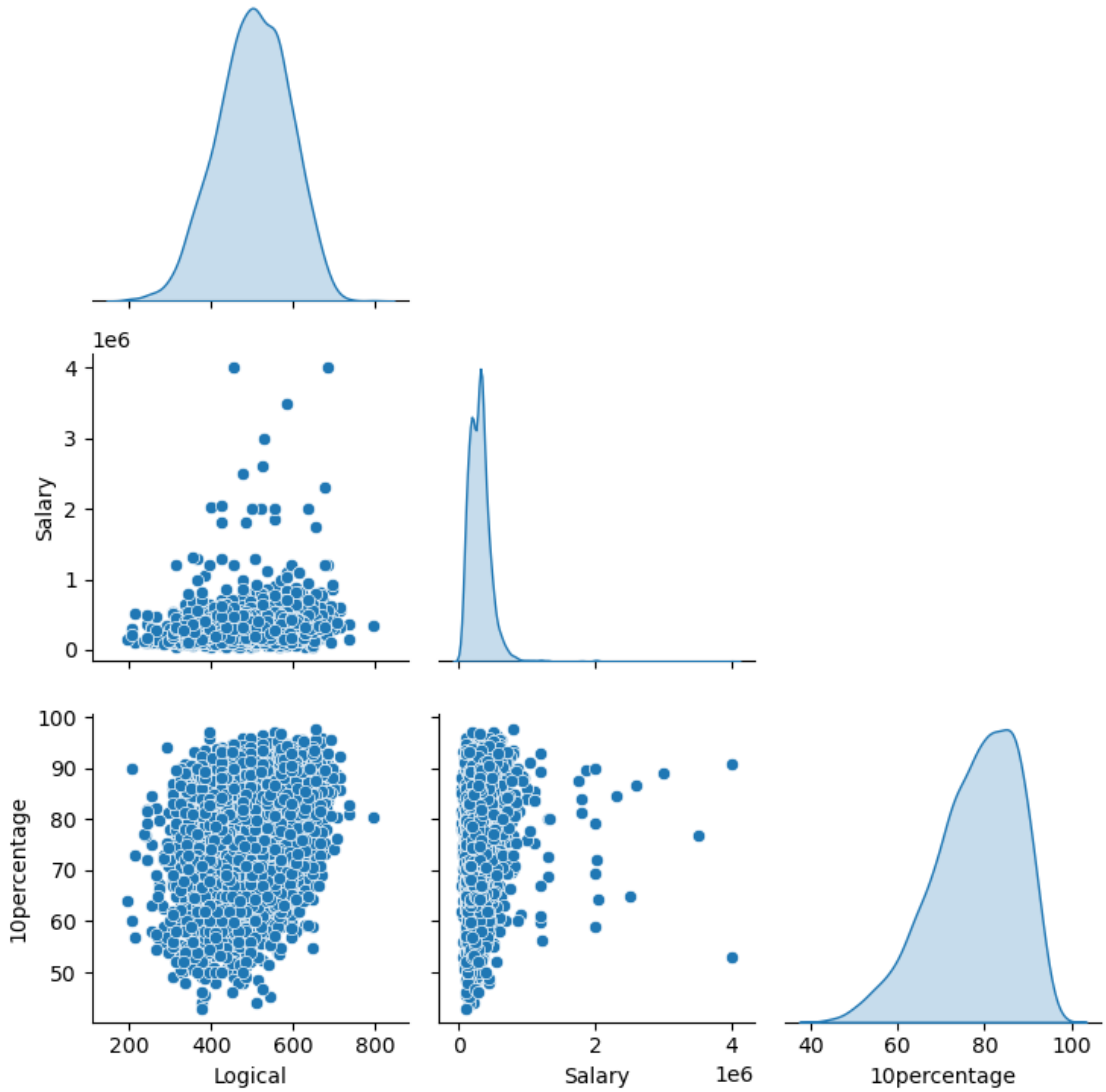
[59]:
```
fig, ax = plt.subplots( figsize = (8,8), sharey = True)
ax.scatter(df['English'],df['Salary'],
                ec = 'k',
                color = 'orange',
                alpha = 0.5,
                label = f"Correlation : {round(df[['English','Salary']].corr().
 ↪iloc[1,0],2)}"
                )
ax.set_ylabel('Salary')
ax.set_xlabel('English Scores')
ax.legend()
```

[59]:
```
<matplotlib.legend.Legend at 0x1c0f25a3650>
```



CONCLUSION : The scatters plots gives adequate evidence that salary is not effected by any of the above scores.

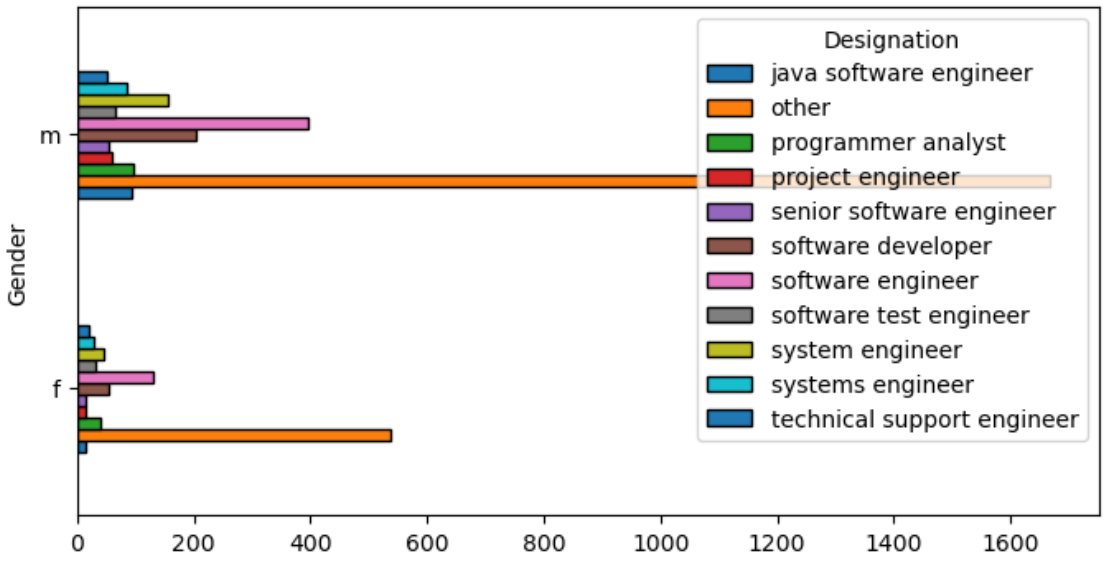
[60]:
```
sns.pairplot(df[['Logical', 'Salary', '10percentage']], diag_kind='kde',␣
↪corner=True)
plt.tight_layout()
plt.show()
```



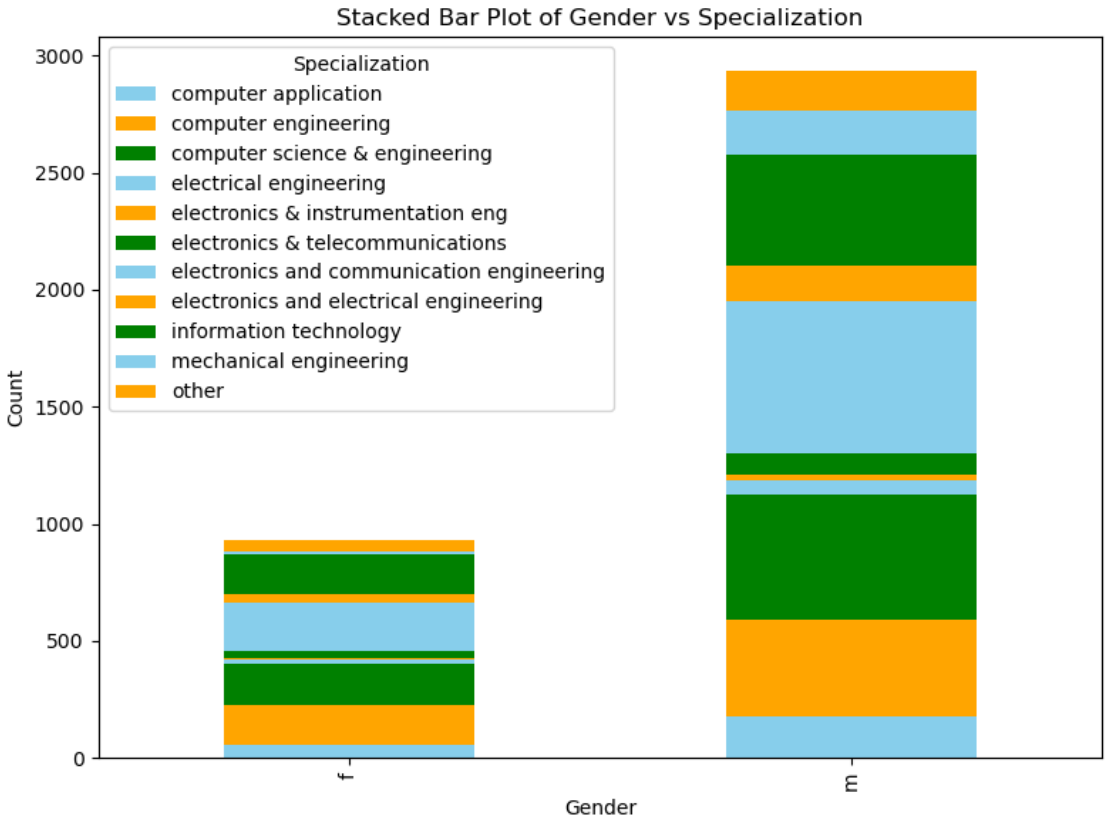
2.1.3 Categorical vs Categorical

[62]:
```
pd.crosstab(df['Designation'],df['Gender']).T.plot(kind = 'barh',ec =␣
↪'k',figsize = (8,4))
```

[62]: <Axes: ylabel='Gender'>



[63]:
```
contingency_table = pd.crosstab(df['Gender'], df['Specialization'])

contingency_table.plot(kind='bar', stacked=True, figsize=(8, 6),␣
↪color=['skyblue', 'orange', 'green'])

plt.title('Stacked Bar Plot of Gender vs Specialization')
plt.xlabel('Gender')
plt.ylabel('Count')
plt.legend(title='Specialization')
plt.tight_layout()
```



2.2 Research Question

Times of India article dated Jan 18, 2019 states that "After doing your Computer Science Engineering if you take up jobs as a Programming Analyst, Software Engineer, Hardware Engineer and Associate Engineer you can earn up to 2.5-3 lakhs as a fresh graduate." Testing of this claim with the data given is done below

[64]:
```
df1 =df[(df['Specialization']== 'computer science & engineering')]
df1
```

[64]:

	Salary	DOJ	DOL	Designation	JobCity	\
6	300000.0	2014-08-01	2015-05-01	java software engineer	other	
18	120000.0	2014-01-01	2014-06-01	other	Gurgaon	
24	335000.0	2014-06-01	2015-06-01	programmer analyst	Hyderabad	
25	435000.0	2012-09-01	2020-08-18	other	Gurgaon	
31	340000.0	2014-08-01	2015-04-01	software engineer	Bangalore	
…	…	…	…	…	…	
3969	330000.0	2015-06-01	2020-08-18	other	other	
3975	300000.0	2014-07-01	2015-04-01	other	Noida	
3981	220000.0	2014-09-01	2020-08-18	software engineer	Gurgaon	
3989	300000.0	2014-09-01	2020-08-18	software engineer	Bangalore	

```
3996  200000.0 2014-07-01 2015-01-01      software developer      other

     Gender        DOB  10percentage      10board  12graduation  …  Quant  \
6         m 1993-02-01         86.08  state board          2010  …    380
18        m 1992-12-07         65.00  state board          2008  …    515
24        m 1993-06-28         88.00  state board          2010  …    630
25        f 1991-03-02         86.80         cbse          2008  …    575
31        m 1992-10-23         77.20  state board          2010  …    450
…     …        …            …           …            …  …    …
3969      m 1993-01-24         76.00  state board          2009  …    630
3975      m 1991-06-03         86.00         cbse          2009  …    535
3981      m 1991-12-17         53.40         cbse          2009  …    645
3989      m 1991-11-23         74.88  state board          2010  …    500
3996      f 1992-03-20         78.72  state board          2010  …    320

        Domain  ComputerProgramming ElectronicsAndSemicon ComputerScience  \
6     0.356536                405.0                    0              346
18    0.563268                425.0                    0                0
24    0.356536                475.0                    0              346
25    0.744758                565.0                    0              438
31    0.622643                485.0                    0              407
…        …                    …                      …            …
3969       NaN                  NaN                    0                0
3975  0.968237                605.0                    0                0
3981  0.953900                575.0                    0              530
3989  0.356536                465.0                    0              346
3996  0.744758                445.0                    0              438

      conscientiousness  agreeableness extraversion  nueroticism  \
6                1.7081        -0.1054      -1.0379      -2.0092
18              -0.1590         0.3789       1.3933      -0.2344
24               0.4155         0.8027       0.1357      -0.9950
25               0.0464         1.2028      -0.9245       0.5323
31              -0.0154         1.2114       1.0859      -1.5021
…                 …             …          …           …
3969             0.5591         0.7119       0.0100      -0.2344
3975             0.5591         0.5454       0.1637       0.3995
3981             0.1282        -0.2871      -0.1437      -1.1218
3989             0.1282         0.0459       1.2396       1.0333
3996            -0.1590         0.0459      -0.4511      -0.3612

      openess_to_experience
6                   -1.0872
18                   1.4386
24                  -0.6692
25                  -0.2875
31                   0.2889
```

```
…                      …
3969                0.8637
3975                0.4805
3981                1.4386
3989                0.6721
3996               -0.0943

[714 rows x 31 columns]
```

[65]:
```
df2=df1[(df1["Designation"]=="programmer␣
 ↪analyst")|(df1["Designation"]=="software␣
 ↪engineer")|(df1["Designation"]=="hardware engineer")
       |(df1["Designation"]=="associate engineer")]
df2
```

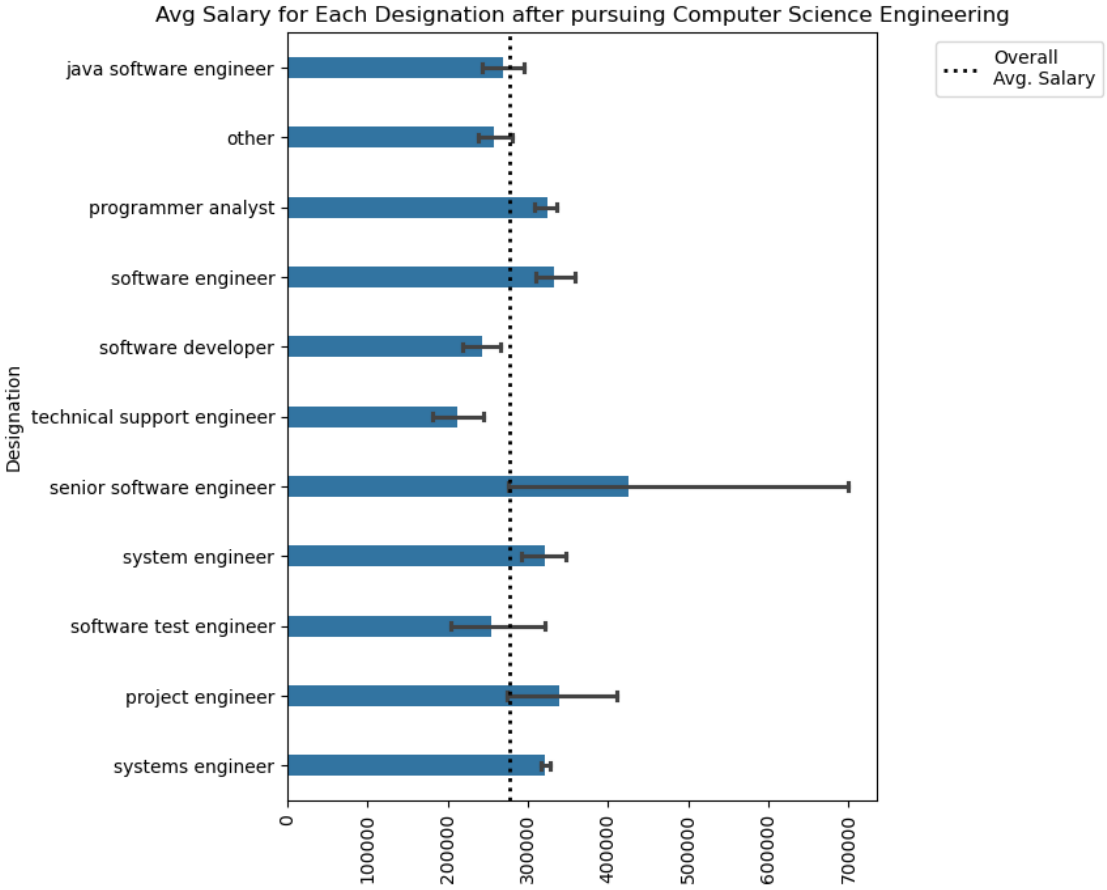
[65]:
```
        Salary        DOJ        DOL         Designation    JobCity Gender  \
24    335000.0 2014-06-01 2015-06-01  programmer analyst  Hyderabad      m
31    340000.0 2014-08-01 2015-04-01   software engineer  Bangalore      m
48    390000.0 2013-09-01 2020-08-18   software engineer  Bangalore      m
52    400000.0 2015-04-01 2020-08-18   software engineer      other      m
55    250000.0 2014-08-01 2020-08-18   software engineer      other      m
…          …          …          …                  …          …    …
3917  105000.0 2014-10-01 2015-04-01   software engineer      other      f
3939  100000.0 2013-07-01 2014-12-01   software engineer  Hyderabad      m
3959  390000.0 2014-01-01 2015-04-01   software engineer    Gurgaon      m
3981  220000.0 2014-09-01 2020-08-18   software engineer    Gurgaon      m
3989  300000.0 2014-09-01 2020-08-18   software engineer  Bangalore      m

            DOB  10percentage      10board  12graduation  …  Quant  \
24   1993-06-28         88.00  state board          2010  …    630
31   1992-10-23         77.20  state board          2010  …    450
48   1991-02-28         86.60         cbse          2009  …    565
52   1992-03-09         85.20         icse          2010  …    485
55   1992-02-13         90.80  state board          2010  …    595
…         …            …           …           …  …    …
3917 1991-12-14         93.00  state board          2009  …    370
3939 1992-07-05         65.00  state board          2009  …    470
3959 1991-09-30         89.60         cbse          2009  …    575
3981 1991-12-17         53.40         cbse          2009  …    645
3989 1991-11-23         74.88  state board          2010  …    500

        Domain  ComputerProgramming ElectronicsAndSemicon ComputerScience  \
24    0.356536                475.0                    0              346
31    0.622643                485.0                    0              407
48    0.356536                475.0                    0              346
52    0.600057                435.0                    0                0
55    0.486747                485.0                    0              376
```

```
…        …                    …                      …              …
3917  0.670743                455.0                      0              0
3939  0.377551                375.0                      0              0
3959  0.842248                545.0                      0            469
3981  0.953900                575.0                      0            530
3989  0.356536                465.0                      0            346

      conscientiousness  agreeableness extraversion  nueroticism  \
24               0.4155         0.8027       0.1357     -0.99500
31              -0.0154         1.2114       1.0859     -1.50210
48              -2.5039         0.0328       0.3817      0.26793
52               1.1336         0.3789       1.0859      0.65300
55              -0.3027         0.7119      -0.2974      1.16010
…                 …              …            …           …
3917            -0.3027         0.5454      -0.6048     -1.62890
3939            -0.3027        -1.9521      -0.6048      1.16010
3959            -0.1590         0.7119       0.9322     -0.74150
3981             0.1282        -0.2871      -0.1437     -1.12180
3989             0.1282         0.0459       1.2396      1.03330

      openess_to_experience
24                  -0.6692
31                   0.2889
48                   0.5024
52                   0.2889
55                  -0.4776
…                    …
3917                 0.2889
3939                -1.8189
3959                 0.2889
3981                 1.4386
3989                 0.6721

[160 rows x 31 columns]
```

[66]:
```
fig, ax = plt.subplots(figsize=(10, 7))
sns.barplot(x='Salary', y='Designation',
            data=df1,
            capsize=0.1,
            width=0.3,
            ax=ax)
ax.axvline(df1['Salary'].mean(), color='k',
           linestyle=':',
           linewidth=2, label='Overall\nAvg. Salary')
ax.set_title('Avg Salary for Each Designation after pursuing Computer Science␣
 ↪Engineering')
ax.legend(loc='upper right', bbox_to_anchor=(1.4, 1))
```

```
ax.set_xlabel('')
ax.set_xticklabels(ax.get_xticklabels(), rotation=90)

plt.tight_layout()
plt.show()
```

```
C:\Users\Dell\AppData\Local\Temp\ipykernel_8016\1844288622.py:13: UserWarning:
set_ticklabels() should only be used with a fixed number of ticks, i.e. after
set_ticks() or using a FixedLocator.
  ax.set_xticklabels(ax.get_xticklabels(), rotation=90)
```



[147]:
```
df2["Salary"]
```

[147]:
```
24      335000.0
31      340000.0
48      390000.0
52      400000.0
55      250000.0
          ...
```

```
3917    105000.0
3939    100000.0
3959    390000.0
3981    220000.0
3989    300000.0
Name: Salary, Length: 160, dtype: float64
```

[149]:
```
ab=df2["Salary"]
bc=[]
for i in ab:
    bc.append(i)
print(bc)
```

```
[335000.0, 340000.0, 390000.0, 400000.0, 250000.0, 330000.0, 325000.0, 375000.0,
325000.0, 360000.0, 170000.0, 305000.0, 560000.0, 300000.0, 785000.0, 330000.0,
210000.0, 320000.0, 275000.0, 300000.0, 475000.0, 240000.0, 335000.0, 300000.0,
345000.0, 300000.0, 450000.0, 370000.0, 180000.0, 360000.0, 320000.0, 375000.0,
420000.0, 215000.0, 350000.0, 340000.0, 310000.0, 350000.0, 85000.0, 330000.0,
420000.0, 335000.0, 515000.0, 350000.0, 275000.0, 300000.0, 315000.0, 370000.0,
325000.0, 450000.0, 240000.0, 120000.0, 300000.0, 275000.0, 335000.0, 400000.0,
275000.0, 450000.0, 350000.0, 305000.0, 120000.0, 305000.0, 300000.0, 315000.0,
450000.0, 310000.0, 120000.0, 330000.0, 300000.0, 225000.0, 335000.0, 200000.0,
300000.0, 330000.0, 240000.0, 310000.0, 340000.0, 400000.0, 300000.0, 350000.0,
315000.0, 310000.0, 320000.0, 600000.0, 315000.0, 590000.0, 305000.0, 200000.0,
310000.0, 300000.0, 350000.0, 240000.0, 380000.0, 350000.0, 400000.0, 350000.0,
180000.0, 550000.0, 350000.0, 400000.0, 1000000.0, 335000.0, 400000.0, 350000.0,
300000.0, 500000.0, 305000.0, 110000.0, 220000.0, 360000.0, 340000.0, 200000.0,
210000.0, 350000.0, 325000.0, 400000.0, 240000.0, 430000.0, 230000.0, 150000.0,
360000.0, 180000.0, 300000.0, 305000.0, 250000.0, 195000.0, 320000.0, 280000.0,
600000.0, 360000.0, 325000.0, 300000.0, 480000.0, 240000.0, 290000.0, 550000.0,
315000.0, 360000.0, 315000.0, 925000.0, 400000.0, 300000.0, 300000.0, 240000.0,
325000.0, 95000.0, 500000.0, 300000.0, 350000.0, 145000.0, 240000.0, 335000.0,
315000.0, 300000.0, 600000.0, 105000.0, 100000.0, 390000.0, 220000.0, 300000.0]
```

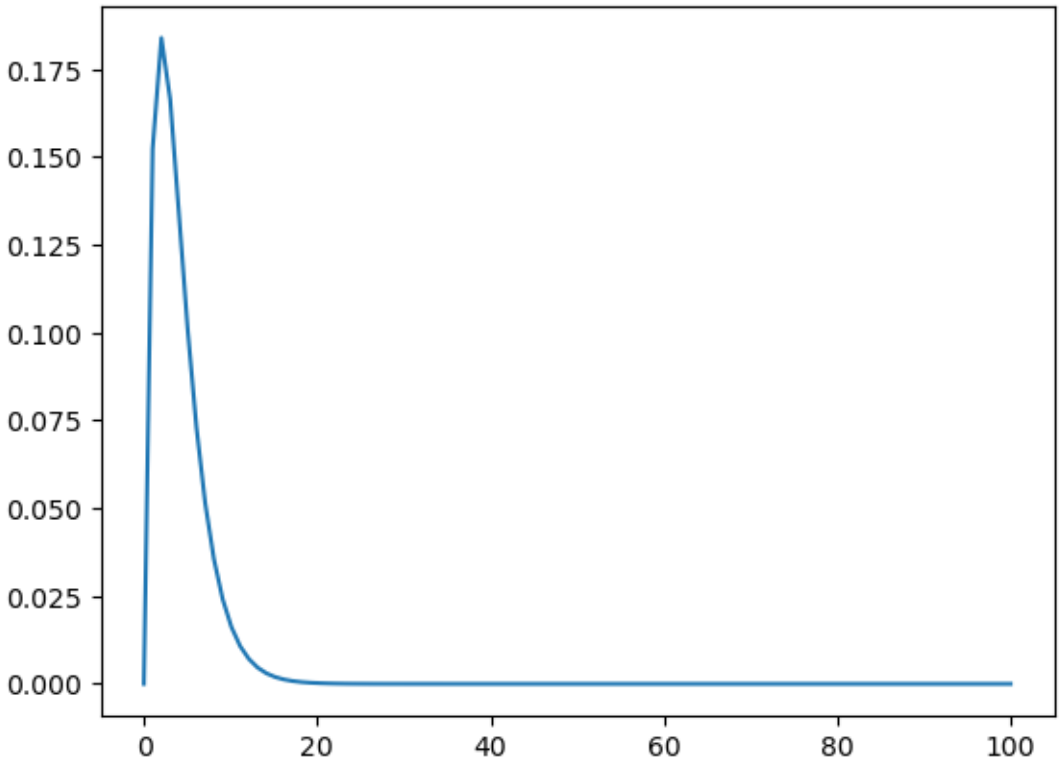
[151]:
```
import random
n=40  #taking few samples for observation out of 662
cd=random.sample(bc,n)
print(cd)
```

```
[250000.0, 400000.0, 320000.0, 335000.0, 280000.0, 325000.0, 200000.0, 350000.0,
195000.0, 450000.0, 420000.0, 325000.0, 600000.0, 110000.0, 450000.0, 275000.0,
590000.0, 240000.0, 315000.0, 95000.0, 300000.0, 225000.0, 400000.0, 145000.0,
335000.0, 275000.0, 240000.0, 85000.0, 325000.0, 210000.0, 335000.0, 430000.0,
300000.0, 340000.0, 310000.0, 315000.0, 105000.0, 335000.0, 350000.0, 275000.0]
```

2.2.1 Is there a relationship between gender and specialization? (i.e. Does the preference of Specialisation depend on the Gender?)

[68]:
```
from scipy.stats import chi2, chi2_contingency
x = np.linspace(0,100, 100)
y = chi2.pdf(x, df=4)
plt.plot(x, y)
```

[68]:
```
[<matplotlib.lines.Line2D at 0x1c0f056a450>]
```



[69]:
```
obs = pd.crosstab(df.Specialization,df.Gender)
obs
```

[69]:

Gender	f	m
Specialization		
computer application	55	177
computer engineering	169	413
computer science & engineering	178	536
electrical engineering	17	62
electronics & instrumentation eng	10	22
electronics & telecommunications	27	92
electronics and communication engineering	209	647

```
electronics and electrical engineering      32  153
information technology                     172  477
mechanical engineering                      10  184
other                                       53  169
```

[70]:
```
chi2_contingency(obs)
```

[70]:
```
Chi2ContingencyResult(statistic=54.17388309280396, pvalue=4.503495976866196e-08,
dof=10, expected_freq=array([[ 55.95859213, 176.04140787],
       [140.37888199, 441.62111801],
       [172.2173913 , 541.7826087 ],
       [ 19.05486542,  59.94513458],
       [  7.7184265 ,  24.2815735 ],
       [ 28.70289855,  90.29710145],
       [206.4679089 , 649.5320911 ],
       [ 44.62215321, 140.37784679],
       [156.53933747, 492.46066253],
       [ 46.79296066, 147.20703934],
       [ 53.54658385, 168.45341615]]))
```

[153]:
```
chi2_test_stat = chi2_contingency(obs)[0]
pval = chi2_contingency(obs)[1]
data = chi2_contingency(obs)[2]
```

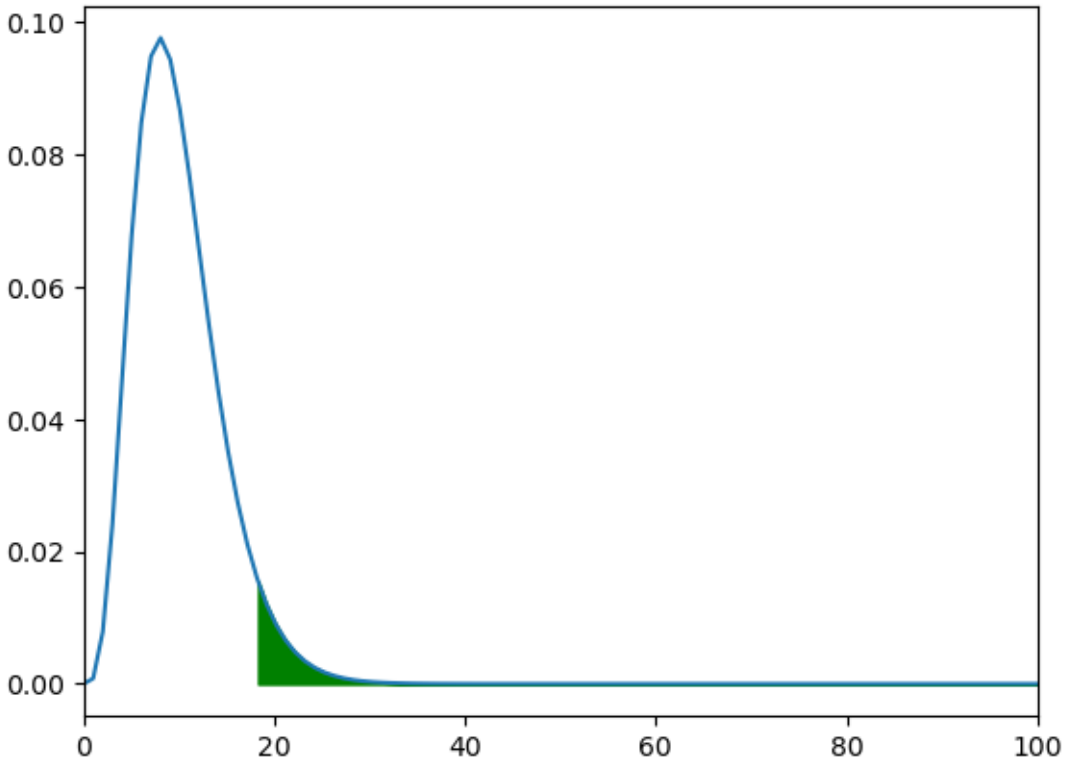
[155]:
```
confidence_level = 0.95

alpha = 1 - confidence_level

chi2_critical = chi2.ppf(1 - alpha, data)

chi2_critical
```

[155]:
```
18.307038053275146
```

[157]:
```
x_min = 0
x_max = 100

# Ploting the graph and setting the x limits
x = np.linspace(x_min, x_max, 100)
y = chi2.pdf(x, data)
plt.xlim(x_min, x_max)
plt.plot(x, y)


# Setting Chi2 Critical value
chi2_critical_right = chi2_critical
```

```
# Shading the right rejection region
x1 = np.linspace(chi2_critical_right, x_max, 100)
y1 = chi2.pdf(x1, data)
plt.fill_between(x1, y1, color='green')
```

[157]: <matplotlib.collections.PolyCollection at 0x1c0ef623d10>



Observation * As the result of the second research question we see that there is a relationship between Gender and specialization. * We test this claim through Chi-Square test and find the result that both the categorical variables are dependent on each other.

3 Conclusion

3.1 Data Understanding:

- The dataset encompasses the employment outcomes of engineering graduates, focusing on target variable Salary.
- Additionally, it includes standardized scores in three distinct areas: cognitive skills, technical skills, and personality skills.

3.2 Data Manipulation:

- Upon initial observation, the dataset consists of 4000 rows and 40 columns.

- The dataset exhibits numerous duplicate values, necessitating data manipulation.
- Initially, we remove redundant rows and columns.
- Subsequently, we assess for the presence of any missing values (NaN).
- Following data cleaning, we proceed with visualization.

3.3 Data Visualization:

Univariate Analysis:

- Univariate analysis encompasses various plots, including Cumulativee Distribution Functions (CDF), Histograms, Box Plots, and Summary Plots.
- These visualizations illustrate probability and frequency distributions.

Bivariate Analysis: * Bivariate analysis comprises Scatterplots, Barplots, Crosstabs, Pivot tables, pie charts. * This analysis helps in comparing percentages across different variables. * Additionally, it aids in identifying outliers, as observed through Boxplots. * For instance, Countplots assist in identifying outliers within categorical variables, such as Job City, by highlighting the cities with higher employee counts.

[ ]:

INNOMATICS®
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

ANALYSIS ON AMCAT DATA (EDA)

About me

- **Background ?**

Hi, I'm Yamini. A data enthusiast, currently learning various things to crack an opportunity to go further.

• Apart from this, I possess the problem solving ability and I am good at learning new things that makes me an ideal candidate to follow my dreams.....

- **Linked in profile url:**

https://www.linkedin.com/in/yamini-j9010

- **Git hub url:**

https://github.com/YaminiRajaRao

INNOMATICS RESEARCH LABS

Business Problem :-

• The key business problem for the AMCAT dataset is to enhance the recruitment process by accurately predicting a candidate's job performance and suitability based on their test scores, educational background, and demographic details. Recruiters and employers face the challenge of efficiently matching candidates to roles where they can excel, while minimizing hiring costs and improving retention rates. By leveraging the data, companies can better identify high-potential candidates and streamline their recruitment efforts

Use case domain :-

• In the recruitment domain, the AMCAT dataset can be used to build predictive models that score candidates based on their likelihood to succeed in specific job roles or industries. For instance, a model could predict which candidates are best suited for IT roles, engineering, or managerial positions, based on their scores in relevant sections like Computer Programming, Logical Reasoning, or English Communication. This allows employers to focus their attention on top-tier talent for each specific domain, ultimately improving the efficiency and effectiveness of hiring decisions

Objective:

The aim of this analysis include :

- Describing the dataset and its features comprehensively.
- Perform Univariate Analysis
- Perform Bivariate Analysis
- Exploring the relationships between independent variables and the target variable
- Identifying any anomalies in the data.

INNOMATİCS
RESEARCH LABS

Summary of the data

- There are 38 columns in total that are used to find the individual impacts on salary.
- Out of 38 columns, there are 29 numerical columns and 9 categorical columns.
- With 3998 Datapoints that make our analysis to the optimal insights with all the necessary information.

INNOMATİCS
RESEARCH LABS

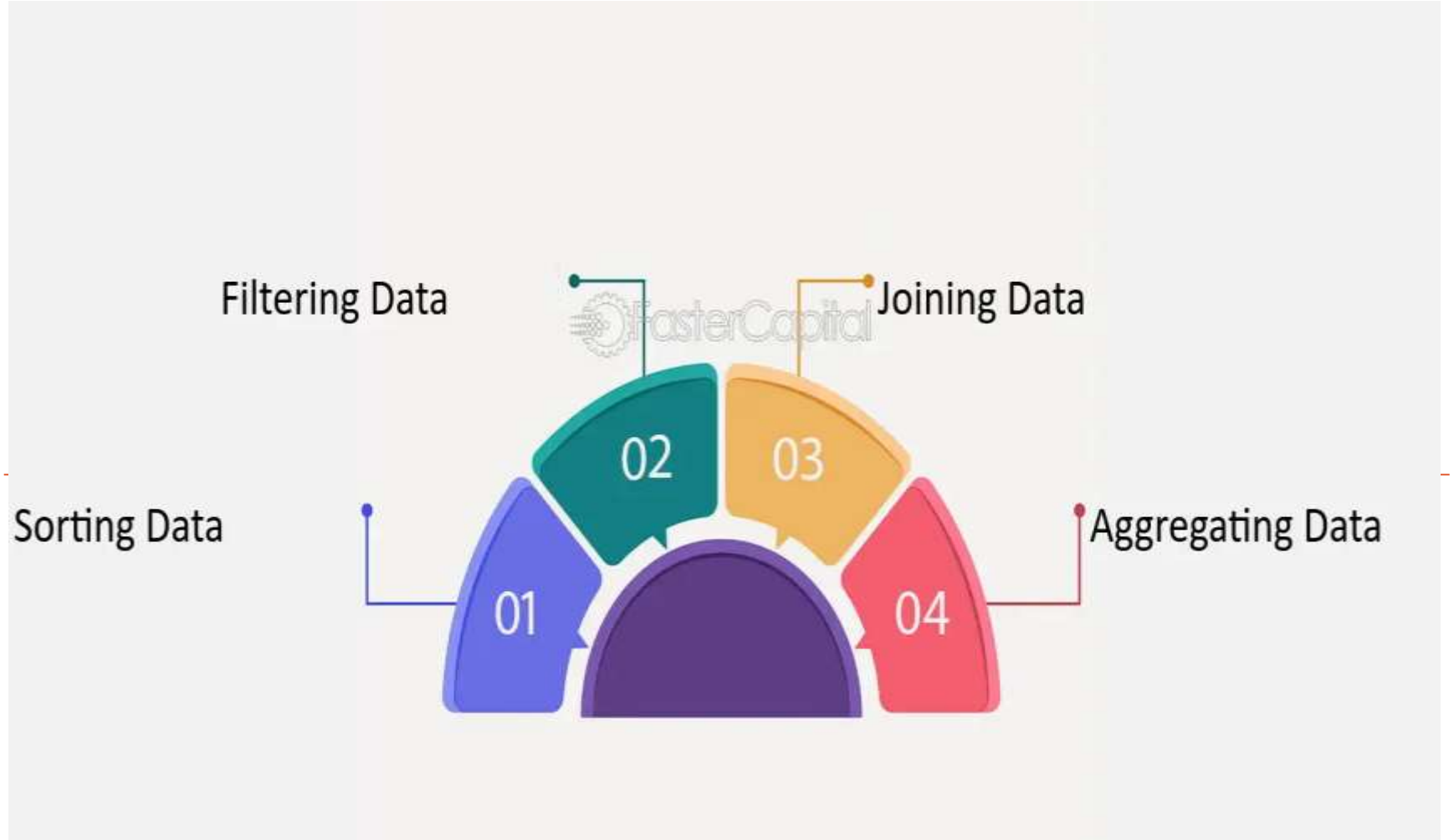
EXPLORATORY DATA ANALYSIS

INNOMATİCS
RESEARCH LABS

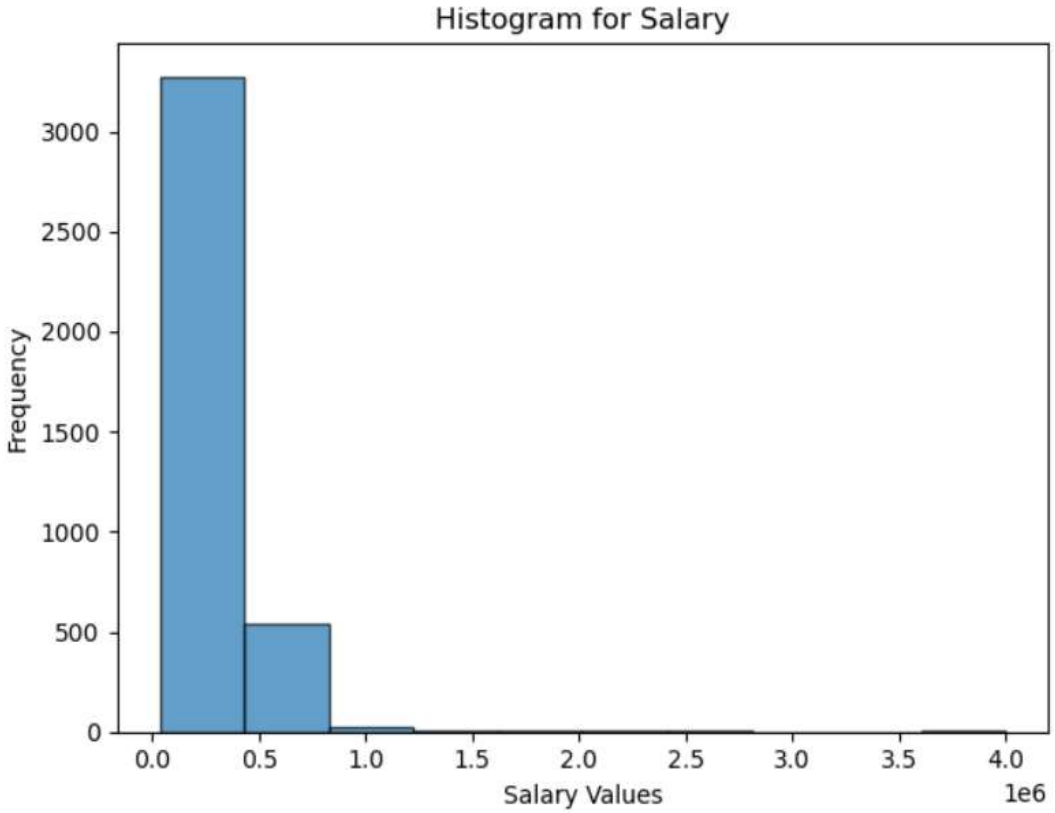
1. Data Cleaning
2. Data Manipulation
3. Univariate Analysis
4. Bivariate Analysis

INNOMATICS
RESEARCH LABS

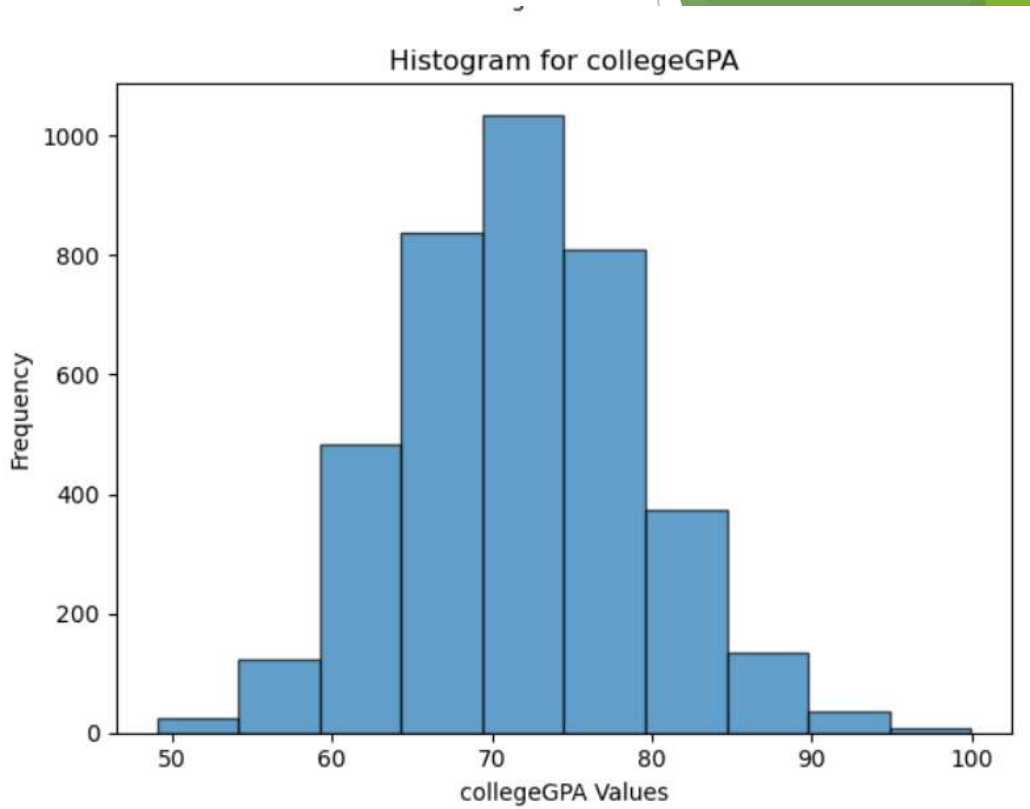
INNOMATİCS RESEARCH LABS



Univariate Analysis

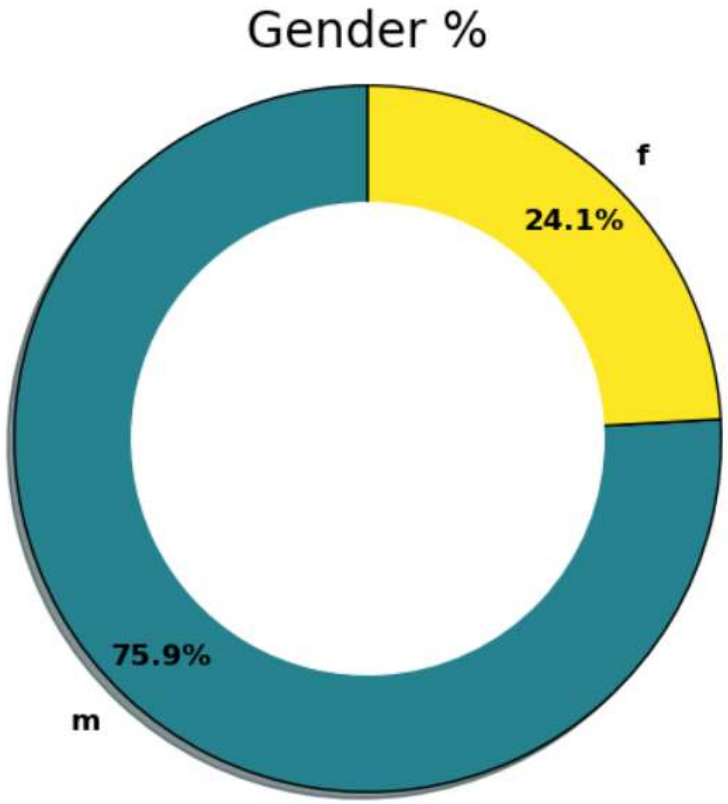


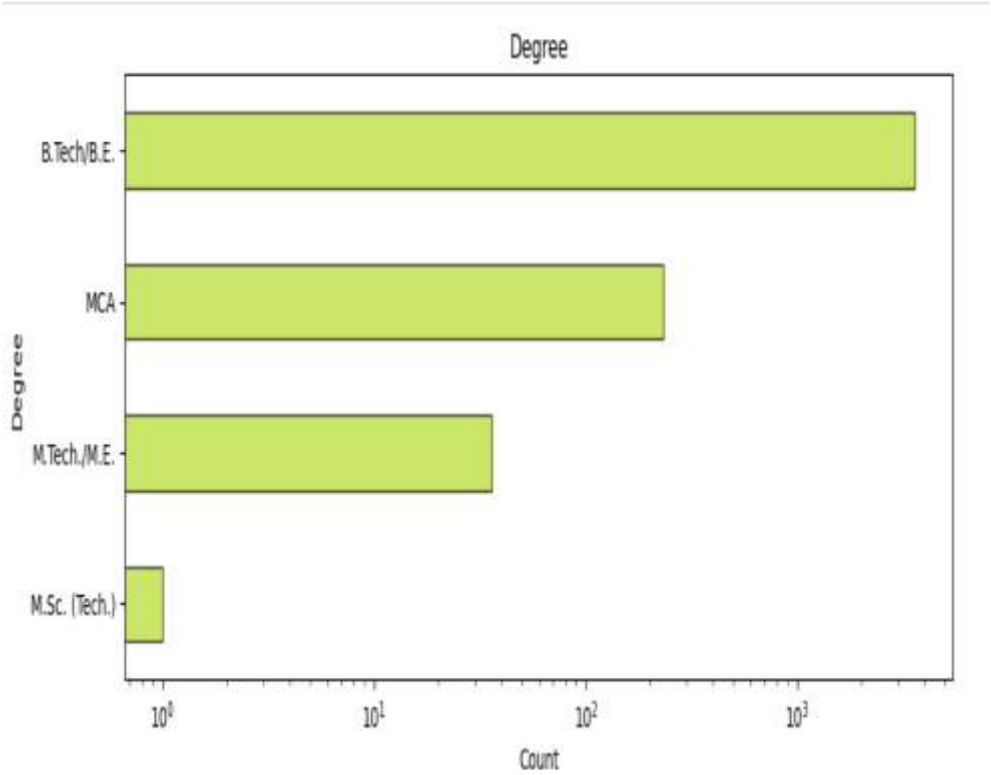
Histogram : The data is positively and highly skewed with skewness 6(approx) which is large as compared to that of normal(0). Mean, median and mode all are approximately equal.



Histogram : Majority of the students GPA were in b/w 63% - 78%. Maximum number of students scored 70% and on average GPA score was 74%.

INNOMATICS RESEARCH LABS

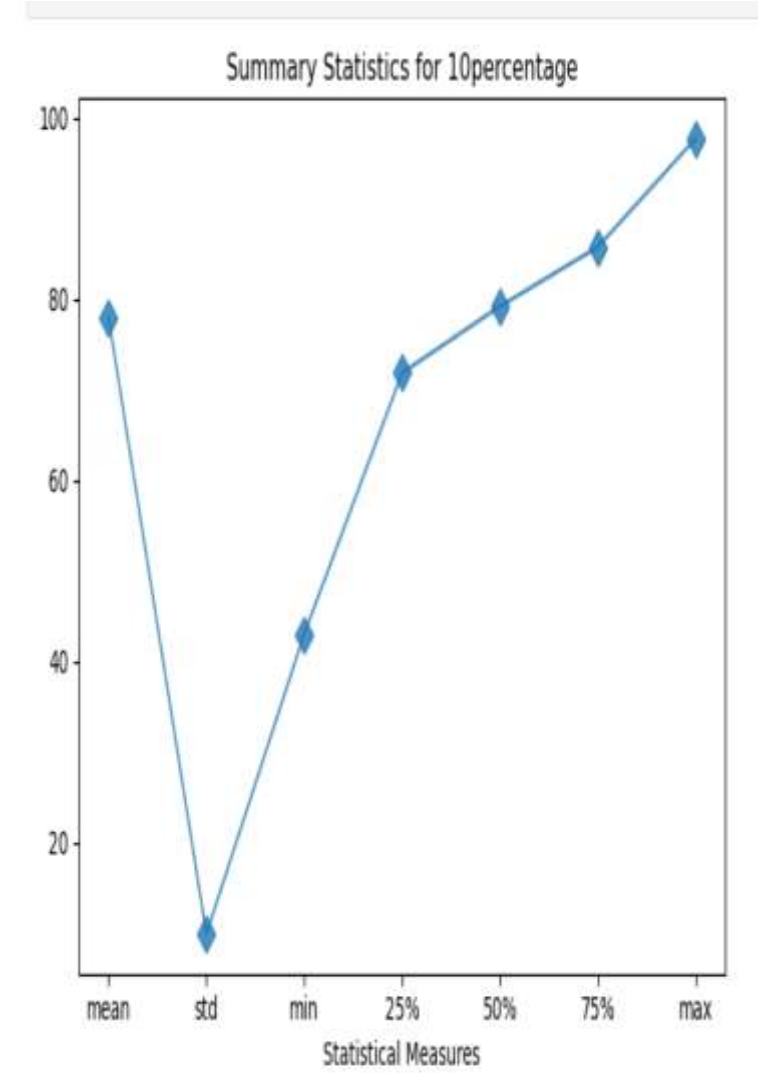




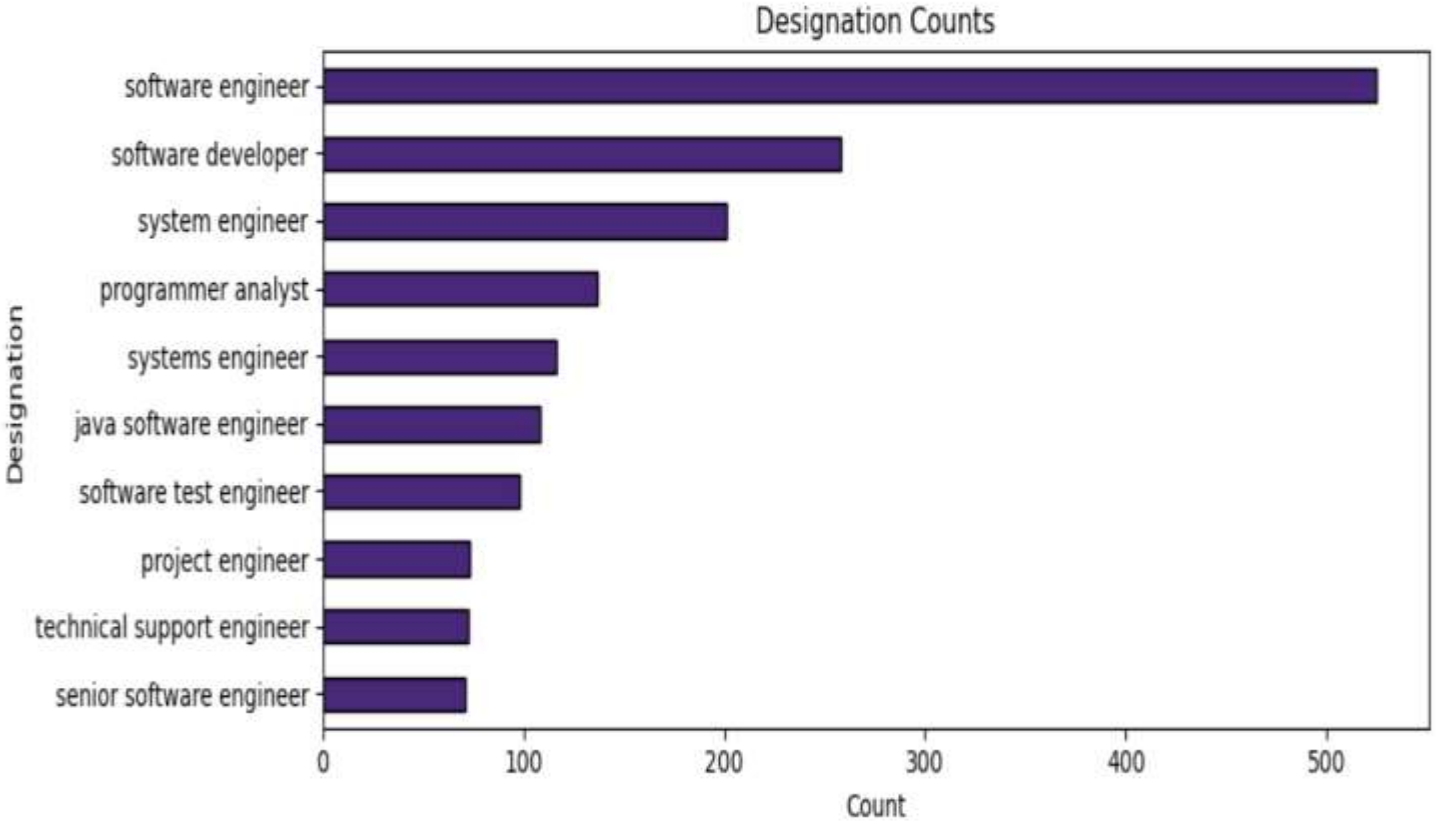
The dataset is not balanced in terms of gender as the population of Male is really larger as compared to the female one.

50% of students scored less than approximately 80%.

INNOMATİCS RESEARCH LABS



50% of students scored less than approximately 80%.

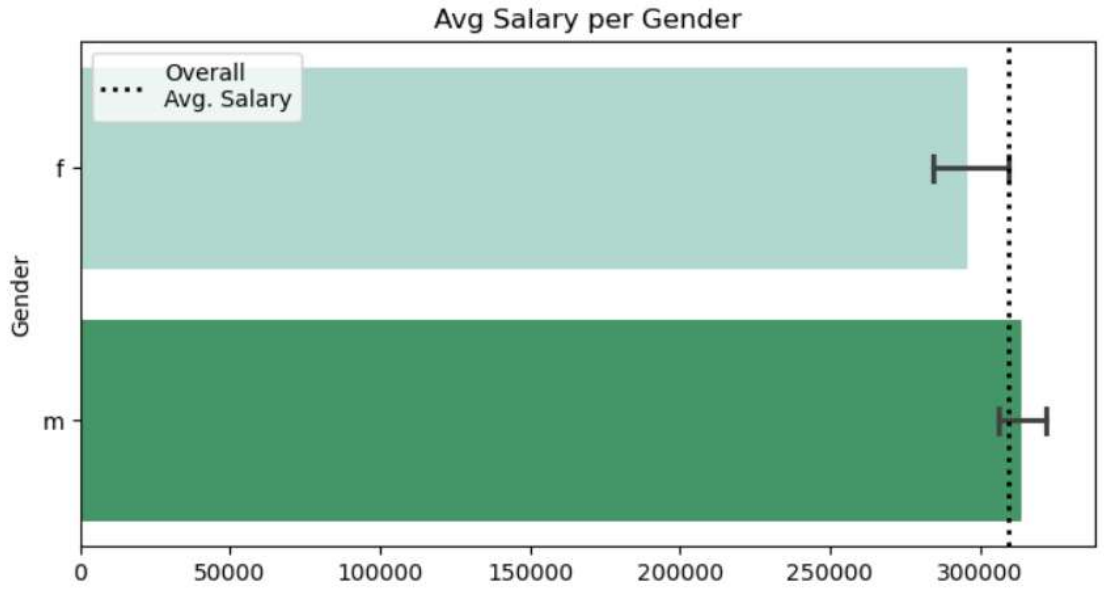


Software engineer is the most common desgination of all, followed by system engineer and software developer. NOTE : This graphs the most common designations. There exists OTHER category too.

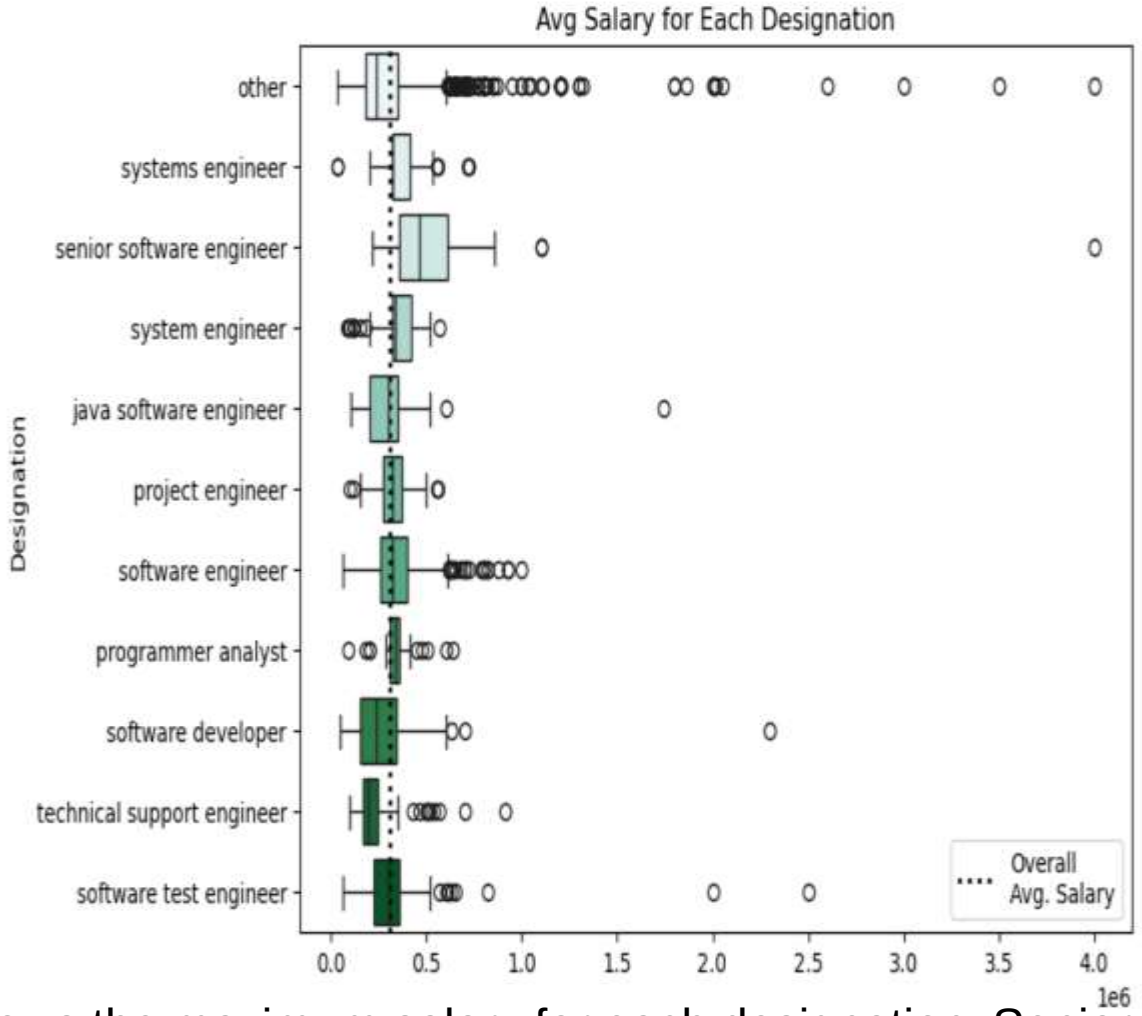
INNOMATİCS RESEARCH LABS

Bivariant Analysis

Category vs numerical



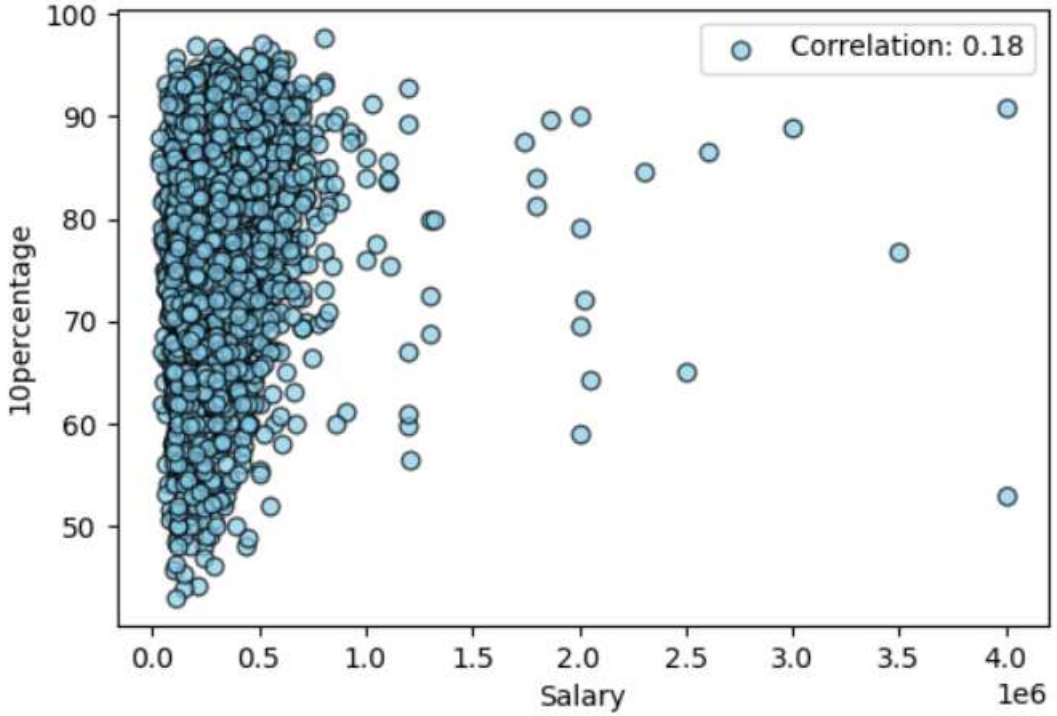
The average salary for both male and female is approximately equal and it implies that there was no gender bias in terms of salary.

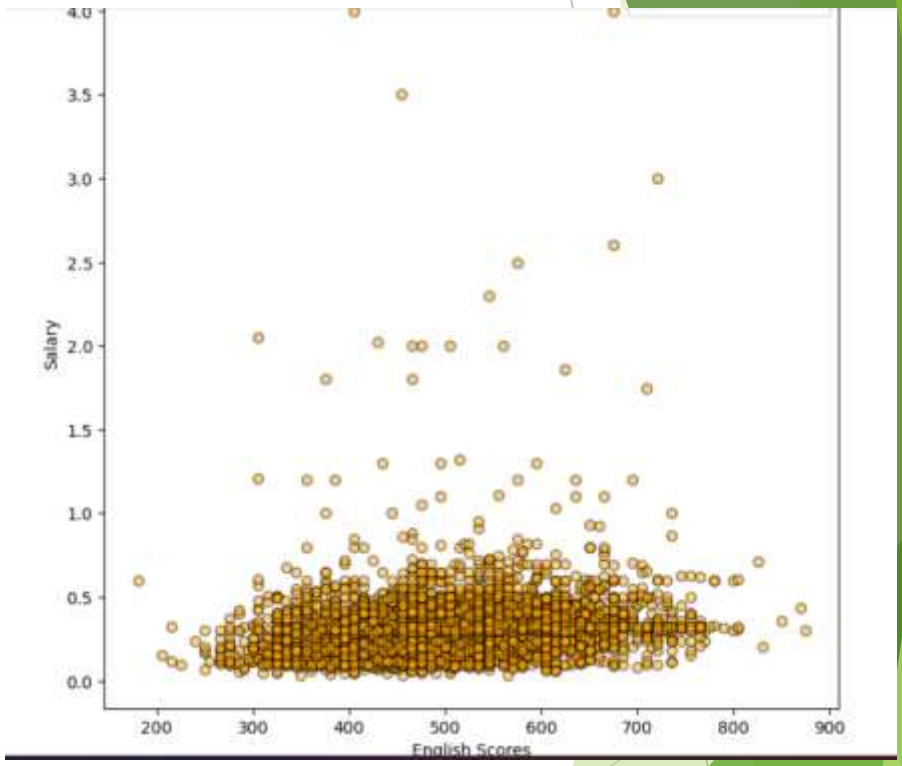


Bar plot shows the maximum salary for each designation. Senior Software Engineer has the highest salary but they also has the maximum standard deviation in their salary. There are only two designations namely, software developer and technical support engineer who has salary lower than average salary.

INNOMATICS RESEARCH LABS

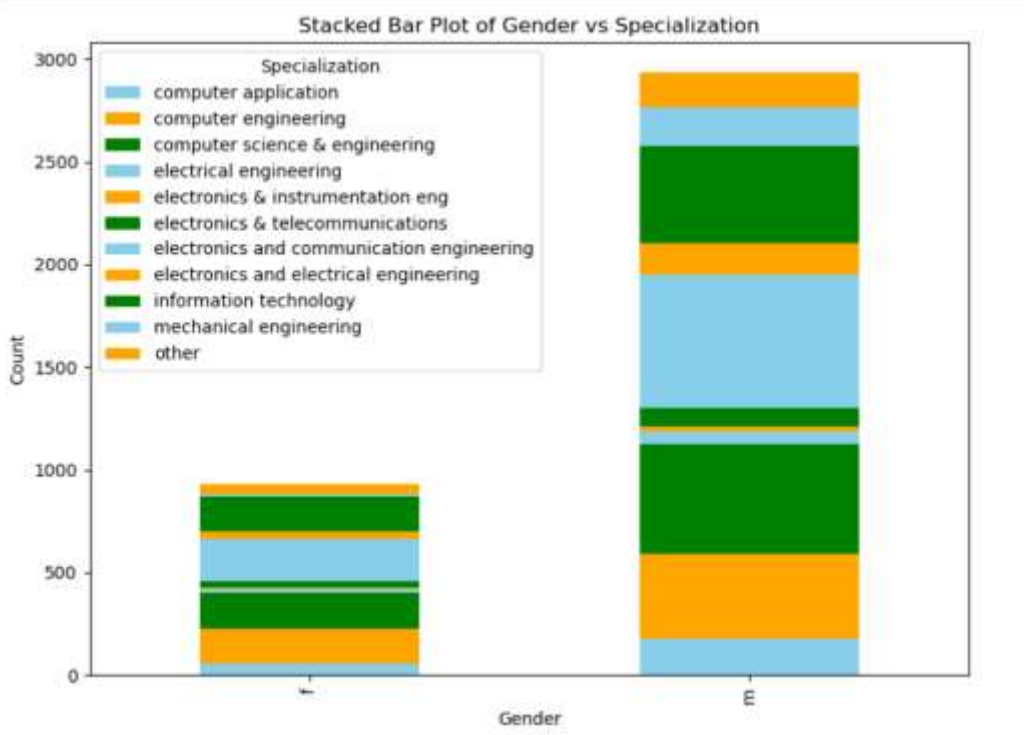
Numerical vs Numerical

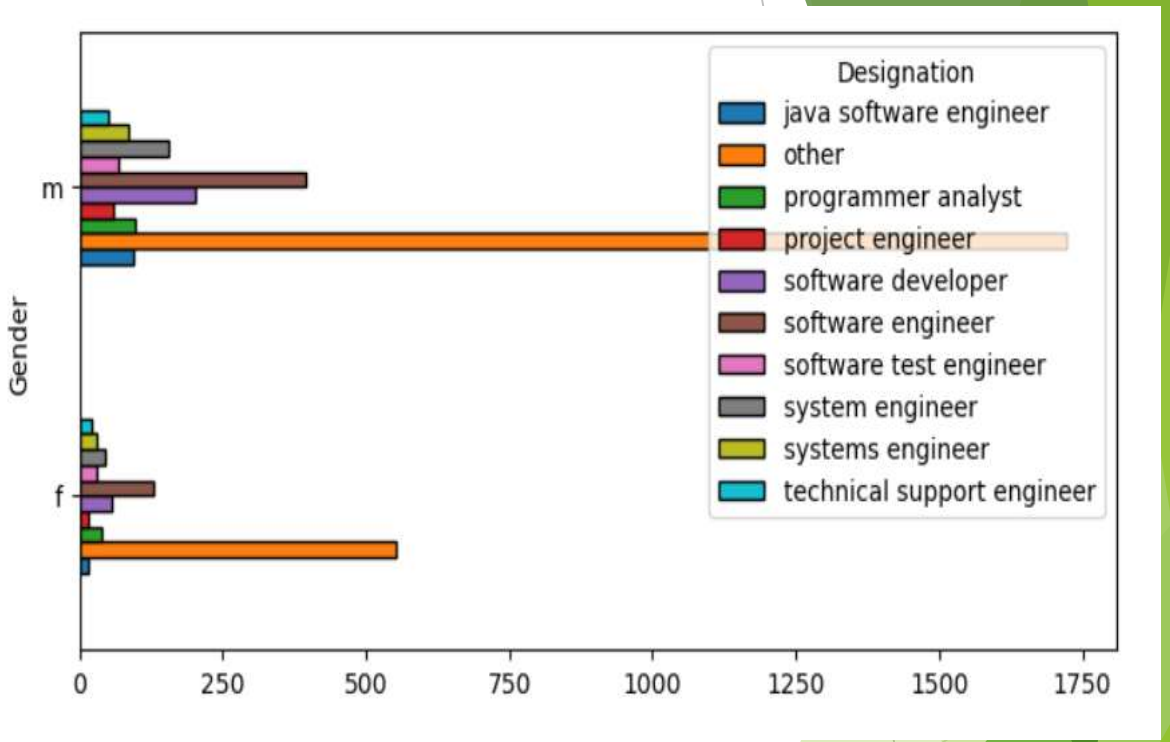




INNOMATİCS RESEARCH LABS

category vs category





INNOMATICS RESEARCH LABS

KEY BUSINESS QUESTION

How do various factors such as cognitive skills, technical skills, personality traits, and demographic information influence the salary outcomes of engineering graduates?“

- This question aims to understand which features or combinations of features have the most significant impact on predicting salary, helping to identify patterns that could improve job placement and salary forecasting for graduates

CONCLUSION

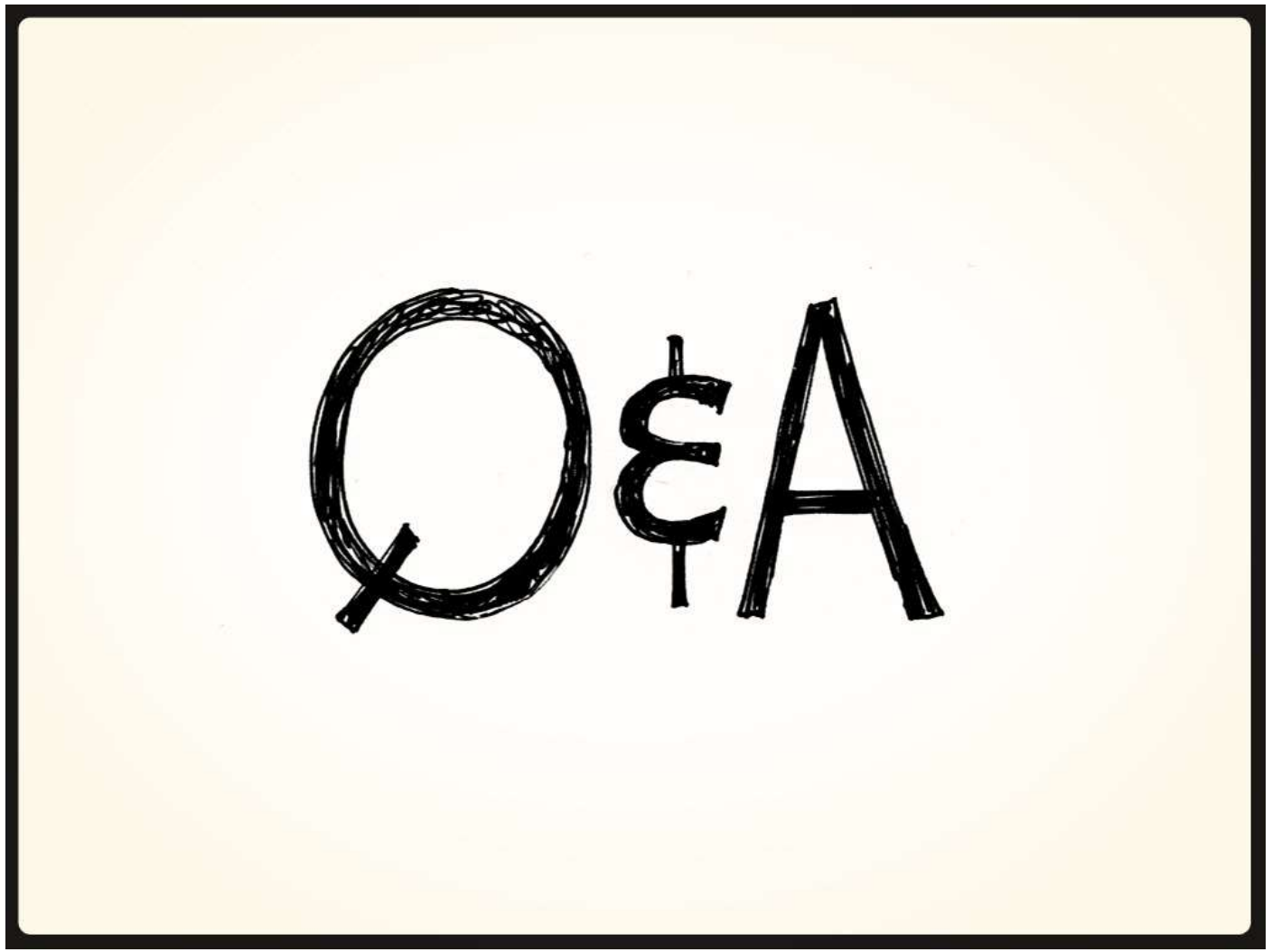
• WHILE WORKING ON THE AMCAT DATASET PROJECT, ONE OF THE MAIN CHALLENGES WAS HANDLING THE COMPLEXITY AND VARIETY OF DATA FEATURES, INCLUDING COGNITIVE, TECHNICAL, AND PERSONALITY SKILLS, ALONG WITH DEMOGRAPHIC INFORMATION. ENSURING DATA QUALITY THROUGH CLEANING WAS CRITICAL, AS MISSING AND INCONSISTENT VALUES IN VARIABLES LIKE SALARY, JOB LOCATION, AND EDUCATION POSED HURDLES DURING ANALYSIS.

• ANOTHER CHALLENGE WAS INTERPRETING THE RELATIONSHIPS BETWEEN MULTIPLE INDEPENDENT VARIABLES AND THE TARGET VARIABLE (SALARY). PERFORMING MEANINGFUL FEATURE SELECTION AND UNDERSTANDING THE CORRELATION BETWEEN SKILLS AND SALARY REQUIRED THOROUGH ANALYSIS THROUGH VISUALIZATION AND STATISTICAL TECHNIQUES. BALANCING CATEGORICAL AND NUMERICAL VARIABLES ALSO ADDED COMPLEXITY, ESPECIALLY IN CREATING EFFECTIVE VISUALIZATIONS TO UNCOVER TRENDS AND INSIGHTS.

• DESPITE THESE CHALLENGES, THE PROJECT PROVIDED VALUABLE EXPERIENCE IN HANDLING REAL-WORLD DATA AND APPLYING VARIOUS TECHNIQUES TO EXTRACT MEANINGFUL PATTERNS, ULTIMATELY HELPING PREDICT SALARY OUTCOMES BASED ON MULTIPLE FACTORS. THAN

EXPERIENCE

- While working on the AMCAT dataset project, one of the main challenges was handling the complexity and variety of data features, including cognitive, technical, and personality skills, along with demographic information. Ensuring data quality through cleaning was critical, as missing and inconsistent values in variables like salary, job location, and education posed hurdles during analysis.
- Another challenge was interpreting the relationships between multiple independent variables and the target variable (salary). Performing meaningful feature selection and understanding the correlation between skills and salary required thorough analysis through visualization and statistical techniques. Balancing categorical and numerical variables also added complexity, especially in creating effective visualizations to uncover trends and insights.
- Despite these challenges, the project provided valuable experience in handling real-world data and applying various techniques to extract meaningful patterns, ultimately helping predict salary outcomes based on multiple factors



INNOMATICS
RESEARCH LABS

Thank You