



INNOMATICS[®]
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

Analysis of AMCAT Data

About me

I am Dixit Negi, a graduate of Bachelor In Computer Application and I want to learn more about Data Science because I'm curious about how machine learning works in the apps I use every day. I am eager to explore how data can provide insights and improve experiences.

Although I currently do not have any work experience I am dedicated to developing my skills and knowledge in this exciting field. I believe that with the right learning and practice I can contribute meaningfully to data science projects in the future.

Have a look at my portfolio below :

[LinkedIn](#) & [GitHub](#)

Objective

AMCAT (Aspiring Minds Computer Adaptive Test) is a test that helps people get jobs. In this project we analysis the salary of a aspirant who give the exam. In this project we have one csv file which have 3998 rows and 39 columns. AMCAT team were able to gather concrete data with which they hoped to understand what has become of candidates since they took part in the tests and find interesting patterns from the study..

Analysis Workflow

- Introduction Of Data
- Data Types And Fixing the type of data
- Univariate Analysis
- Bivariate Analysis
- Research Questions :
 - 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.” Test this claim with the data given to you.
 - Is there a relationship between gender and specialization? (i.e. Does the preference of Specialisation depend on the Gender?)

Introduction Of Data

Introduction of data

```
[4]: df.shape
[4]: (3998, 39)

[5]: df.size
[5]: 155922

[6]: df.columns
[6]: Index(['Unnamed: 0', 'ID', 'Salary', 'DOJ', 'DOL', 'Designation', 'JobCity',
          'Gender', 'DOB', '10percentage', '10board', '12graduation',
          '12percentage', '12board', 'CollegeID', 'CollegeTier', 'Degree',
          'Specialization', 'collegeSPA', 'CollegeCityID', 'CollegeCityTier',
          'CollegeState', 'GraduationYear', 'English', 'Logical', 'Quant',
          'Domain', 'ComputerProgramming', 'ElectronicsAndSemicon',
          'ComputerScience', 'MechanicalEngg', 'ElectricalEngg', 'TelecomEngg',
          'CivilEngg', 'conscientiousness', 'agreeableness', 'extraversion',
          'nueroticism', 'openess_to_experience'],
          dtype='object')
```

```
[7]: df.isnull().sum()
```

```
[7]: Unnamed: 0      0
     ID           0
     Salary       0
     DOJ          0
     DOL          0
     Designation  0
     JobCity      0
     Gender       0
     DOB          0
     10percentage 0
     10board      0
     12graduation 0
     12percentage 0
     12board      0
     CollegeID    0
     CollegeTier  0
     Degree       0
     Specialization 0
     .....
```

```
[7]: Unnamed: 0      0
     ID           0
     Salary       0
     DOJ          0
     DOL          0
     Designation  0
     JobCity      0
     Gender       0
     DOB          0
     10percentage 0
     10board      0
     12graduation 0
     12percentage 0
     12board      0
     CollegeID    0
     CollegeTier  0
     Degree       0
     Specialization 0
     collegeSPA   0
     CollegeCityID 0
     CollegeCityTier 0
     CollegeState 0
     GraduationYear 0
     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
```

```
[8]: df.duplicated().sum()
```

```
[8]: 0
```

Data Types And Fixing The Types Of Data

Data Types And Fixing the type of data

```
[9]: 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   DOS                 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  neuroticism         3998 non-null  float64 
38  openness_to_experience 3998 non-null  float64 
dtypes: float64(10), int64(17), object(12)
memory usage: 1.2+ MB
```

```
[10]: df['DOJ'] = pd.to_datetime(df['DOJ'])

C:\Users\ADMINI~1\AppData\Local\Temp\ipykernel_25392\1267864286.py:11: UserWarning: Could not infer format, so each element will be parsed individual
ly, falling back to 'dateutil'. To ensure parsing is consistent and as-expeted, please specify a format.
  df['DOJ'] = pd.to_datetime(df['DOJ'])

[11]: df['DOL'] = pd.to_datetime(df['DOL'])

C:\Users\ADMINI~1\AppData\Local\Temp\ipykernel_25392\1267864286.py:11: UserWarning: Could not infer format, so each element will be parsed individual
ly, falling back to 'dateutil'. To ensure parsing is consistent and as-expeted, please specify a format.
  df['DOL'] = pd.to_datetime(df['DOL'])

[12]: df.dtypes

[13]: Unnamed: 0      object
      ID             int64
      Salary        float64
      DOJ            datetime64[ns]
      DOL            object
      Designation    object
      JobCity        object
      Gender         object
      DOS            datetime64[ns]
      10percentage   float64
      10board        object
      12graduation   int64
      12percentage   float64
      12board        object
      collegeID      int64
      collegeTier    int64
      Degree         object
      Specialization object
      collegeGPA     float64
      collegeCityID int64
      collegeCityTier int64
      collegeState   object
      GraduationYear int64
      English        int64
      Logical        int64
      Quant          int64
      Domain         float64
      ComputerProgramming int64
      ElectronicsAndSemicon int64
      ComputerScience int64
      MechanicalEngg int64
      ElectricalEngg int64
      TelecomEngg    int64
      CivilEngg      int64
      conscientiousness float64
      agreeableness  float64
      extraversion   float64
      neuroticism    float64
      openness_to_experience float64
```

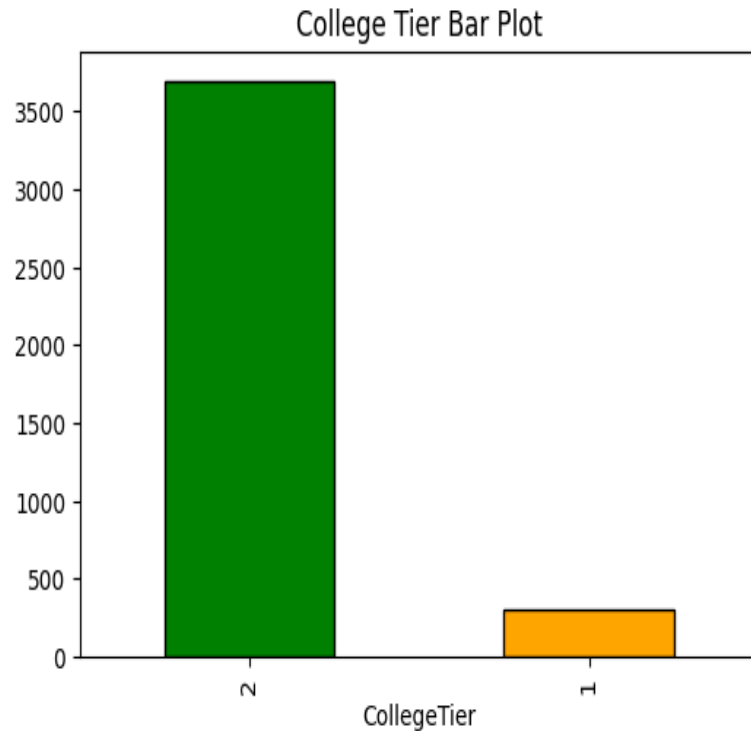
```
[13]: df['Unnamed: 0'].unique()
```

```
[13]: array(['train'], dtype=object)
```

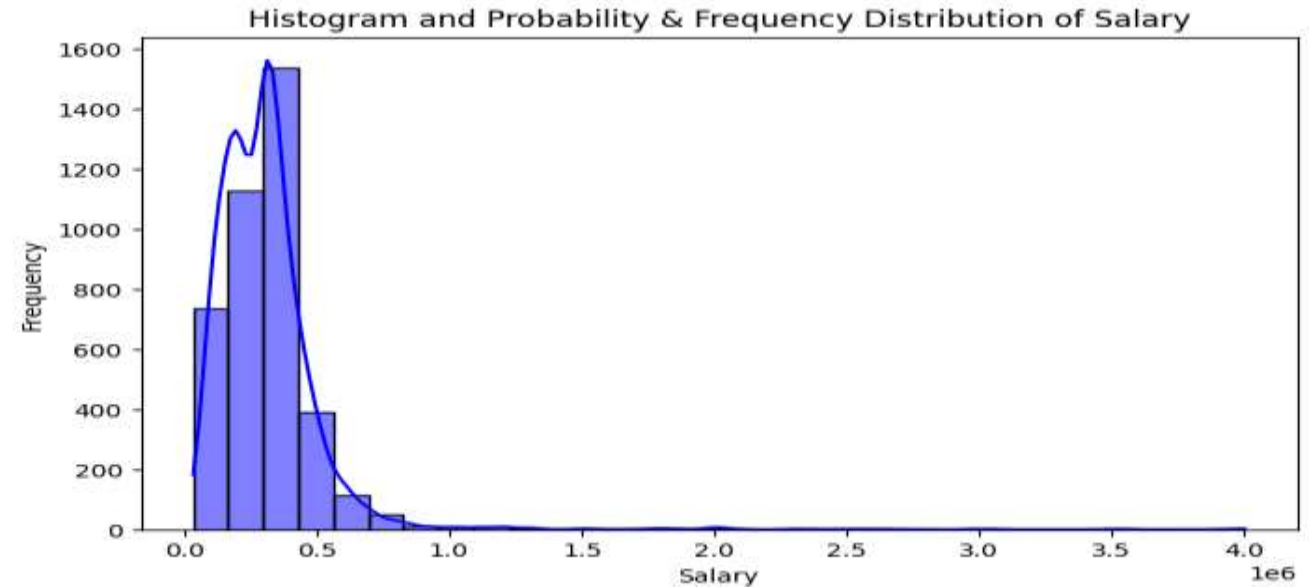
```
[14]: df=df.drop('Unnamed: 0',axis=1)
```

```
[15]: df['Salary'] = df['Salary'].astype(int)
```

Univariate Analysis

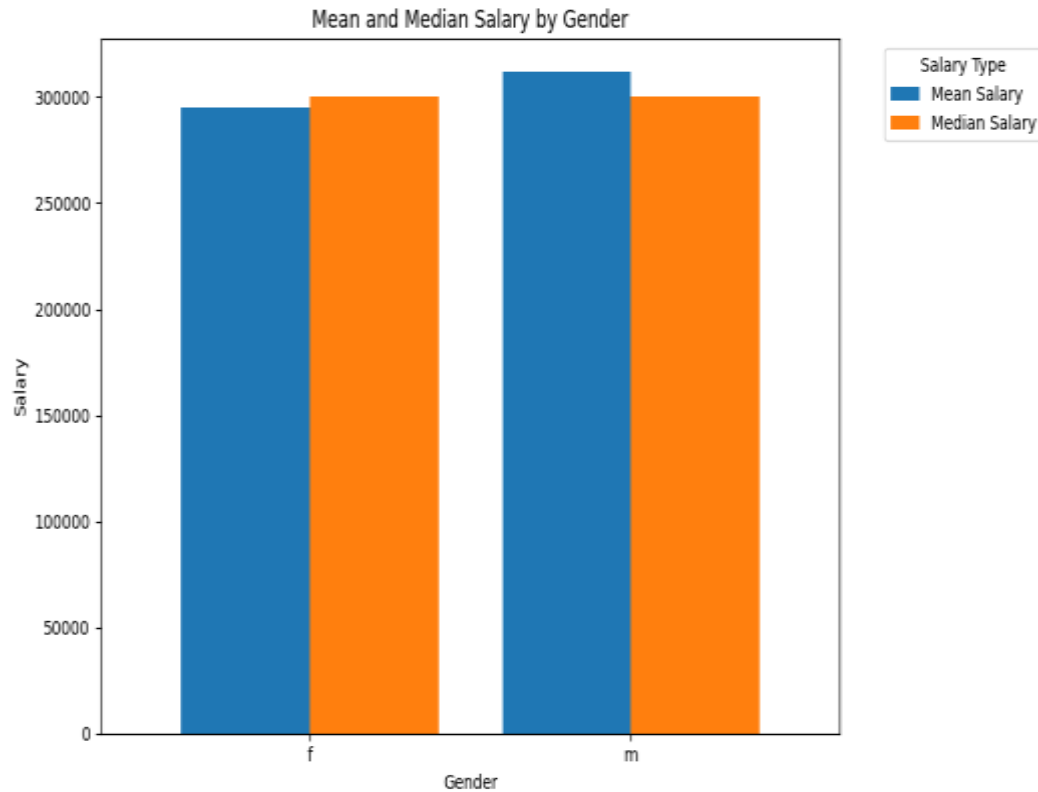


In College Tier more candidates come from Tier 2 colleges as compared to Tier 1 colleges.

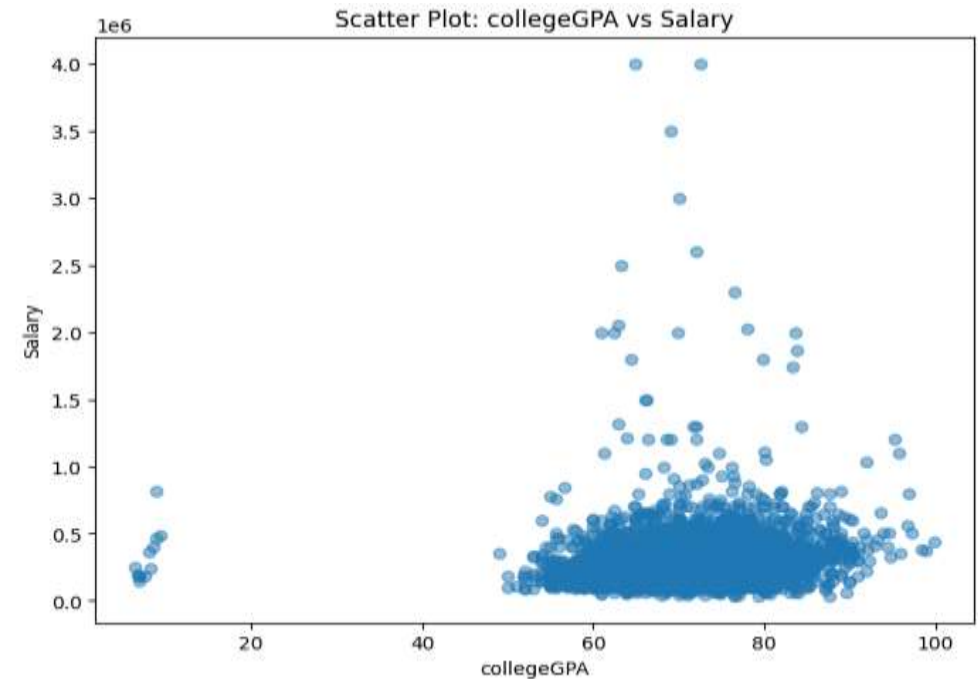


In salary Right-skewed distribution with most prices concentrated towards the lower end.

Bivariate Analysis

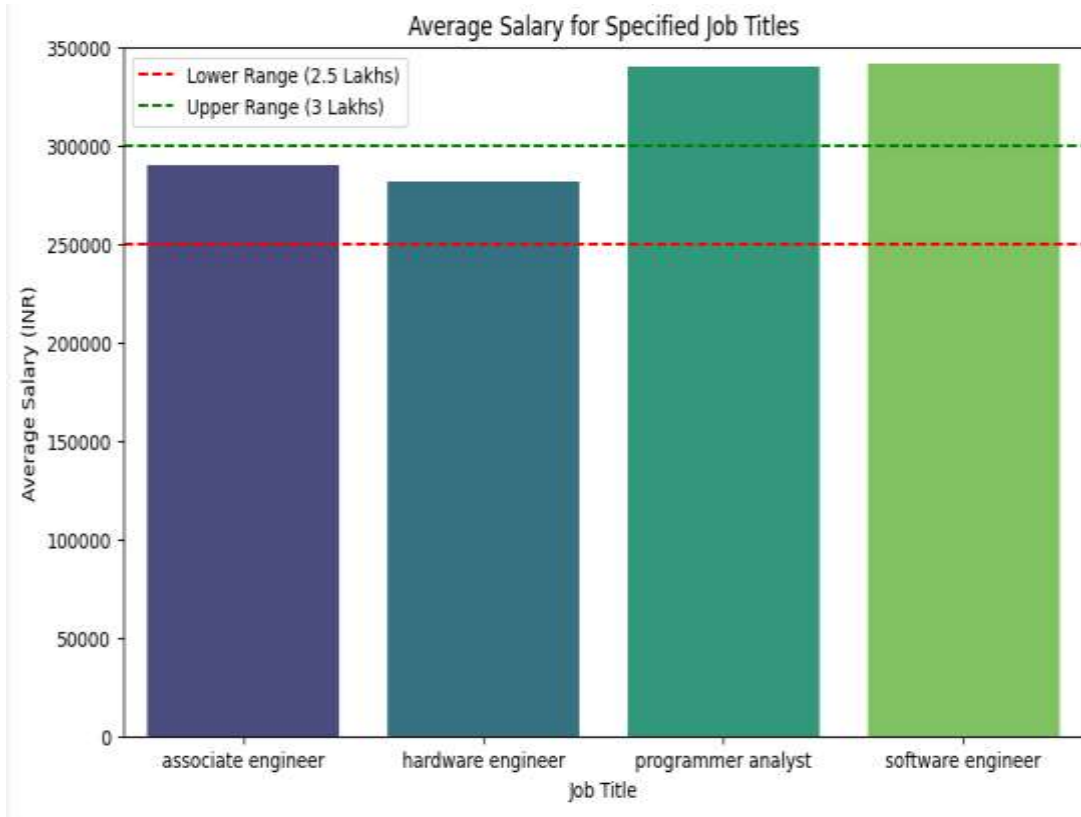


As we see the mean of the salary of male will be higher than female but the median are same

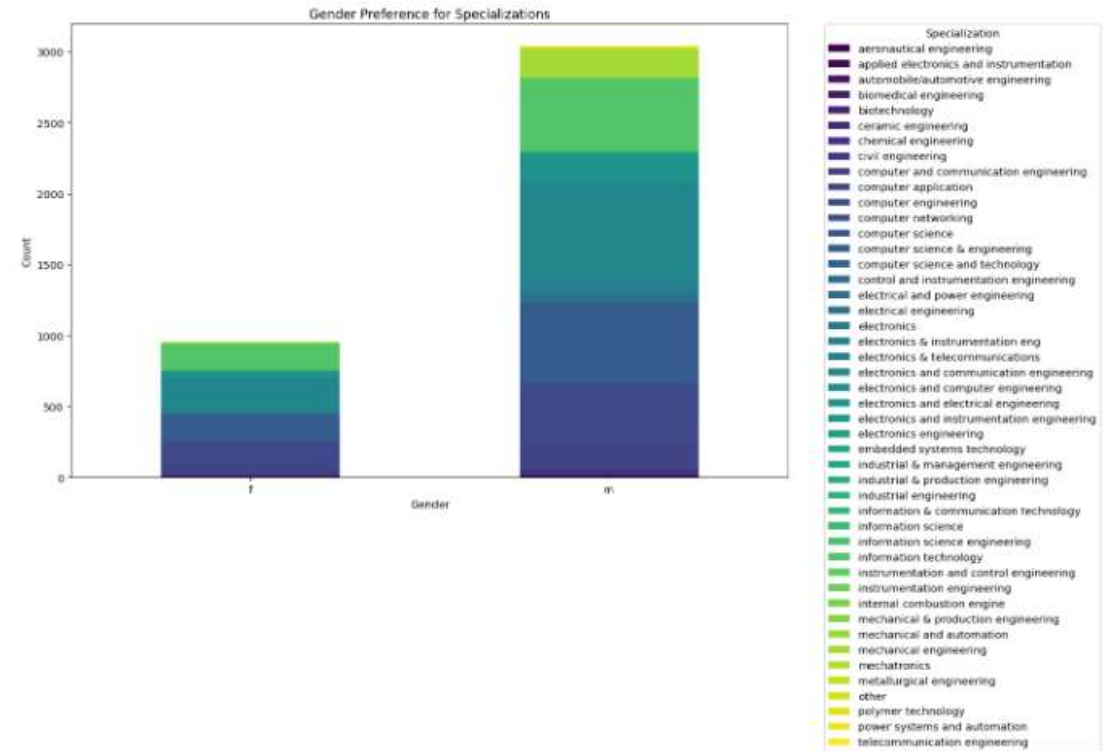


Found a positive correlation between college CGPA and salary.

Research Questions



Average salary for specified job title is between 2.5 lpa to 3 lpa



It show Gender Preference for Specializations

Conclusion

Following the insights generated from my analysis, I can make the following conclusions:

- More candidates come from Tier 2 colleges as compared to Tier 1 colleges.
- Salary Analysis:- The average salary for those who done our degree in M.Tech./M.E. .
- Most of the people who have the Location of the job city is Bangalore .

For the full analysis check my [Github](#) repository.

THANK
YOU

