Report & Analysis

Q1. The task is to make an ML-based prediction system to predict the job role for a new graduate.

Ans:
The dataset attached consists of 39 columns and 20000 rows which includes the anonymized information for graduates along with their respective scores in various CS subjects. Specifically, the suggested job role is predicted for every graduate based on the courses they have done.

Algorithm Design:

The algorithm design/ steps followed are explained below:

Information of the Dataset:

The dataset was read using the pandas library and stored in the form of a data frame. All the columns in the dataframe is stored in the form of a list.

```
Columns:  ['Acedamic percentage in Operating Systems', 'percentage in Algorithms', 'Percentage in Programming Concepts', 'Percentage in Software Engineering', 'Percentage in Computer Networks', 'Percentage in Electronics Subjects', 'Percentage in Computer Architecture', 'Percentage in Mathematics', 'Percentage in Communication skills', 'Hours working per day', 'Logical quotient rating', 'hackathons', 'coding skills rating', 'public speaking points', 'can work long time before system?', 'self-learning capability?', 'Extra-courses did', 'certifications', 'workshops', 'talenttests taken?', 'olympiads', 'reading and writing skills', 'memory capability score', 'Interested subjects', 'interested career area ', 'Job/Higher Studies?', 'Type of company want to settle in?', 'Taken inputs from seniors or elders', 'interested in games', 'Interested Type of Books', 'Salary Range Expected', 'In a Realtionship?', 'Gentle or Tuff behaviour?', 'Management or Technical', 'Salary/work', 'hard/smart worker', 'worked in teams ever?', 'Introvert', 'Suggested Job Role']
```

Other Information about the Dataset which shows us the type/Count and Non Null information about the dataset:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20000 entries, 0 to 19999
Data columns (total 39 columns):
 #   Column                                  Non-Null Count  Dtype 
---  ------                                  --------------  ----- 
 0   Acedamic percentage in Operating Systems  20000 non-null  int64 
 1   percentage in Algorithms                20000 non-null  int64 
 2   Percentage in Programming Concepts      20000 non-null  int64 
 3   Percentage in Software Engineering      20000 non-null  int64 
 4   Percentage in Computer Networks         20000 non-null  int64 
 5   Percentage in Electronics Subjects      20000 non-null  int64 
 6   Percentage in Computer Architecture     20000 non-null  int64 
 7   Percentage in Mathematics               20000 non-null  int64 
 8   Percentage in Communication skills      20000 non-null  int64 
 9   Hours working per day                   20000 non-null  int64 
 10  Logical quotient rating                 20000 non-null  int64 
 11  hackathons                              20000 non-null  int64 
 12  coding skills rating                    20000 non-null  int64 
 13  public speaking points                  20000 non-null  int64 
 14  can work long time before system?       20000 non-null  object
 15  self-learning capability?               20000 non-null  object
 16  Extra-courses did                       20000 non-null  object
 17  certifications                          20000 non-null  object
 18  workshops                               20000 non-null  object
 19  talenttests taken?                      20000 non-null  object
 20  olympiads                               20000 non-null  object
 21  reading and writing skills              20000 non-null  object
 22  memory capability score                 20000 non-null  object
 23  Interested subjects                     20000 non-null  object
 24  interested career area                  20000 non-null  object
 25  Job/Higher Studies?                     20000 non-null  object
 26  Type of company want to settle in?      20000 non-null  object
 27  Taken inputs from seniors or elders     20000 non-null  object
 28  interested in games                     20000 non-null  object
 29  Interested Type of Books                20000 non-null  object
 30  Salary Range Expected                   20000 non-null  object
 31  In a Realtionship?                      20000 non-null  object
 32  Gentle or Tuff behaviour?               20000 non-null  object
 33  Management or Technical                 20000 non-null  object
 34  Salary/work                             20000 non-null  object
 35  hard/smart worker                       20000 non-null  object
 36  worked in teams ever?                   20000 non-null  object
 37  Introvert                               20000 non-null  object
 38  Suggested Job Role                      20000 non-null  object
dtypes: int64(14), object(25)
```

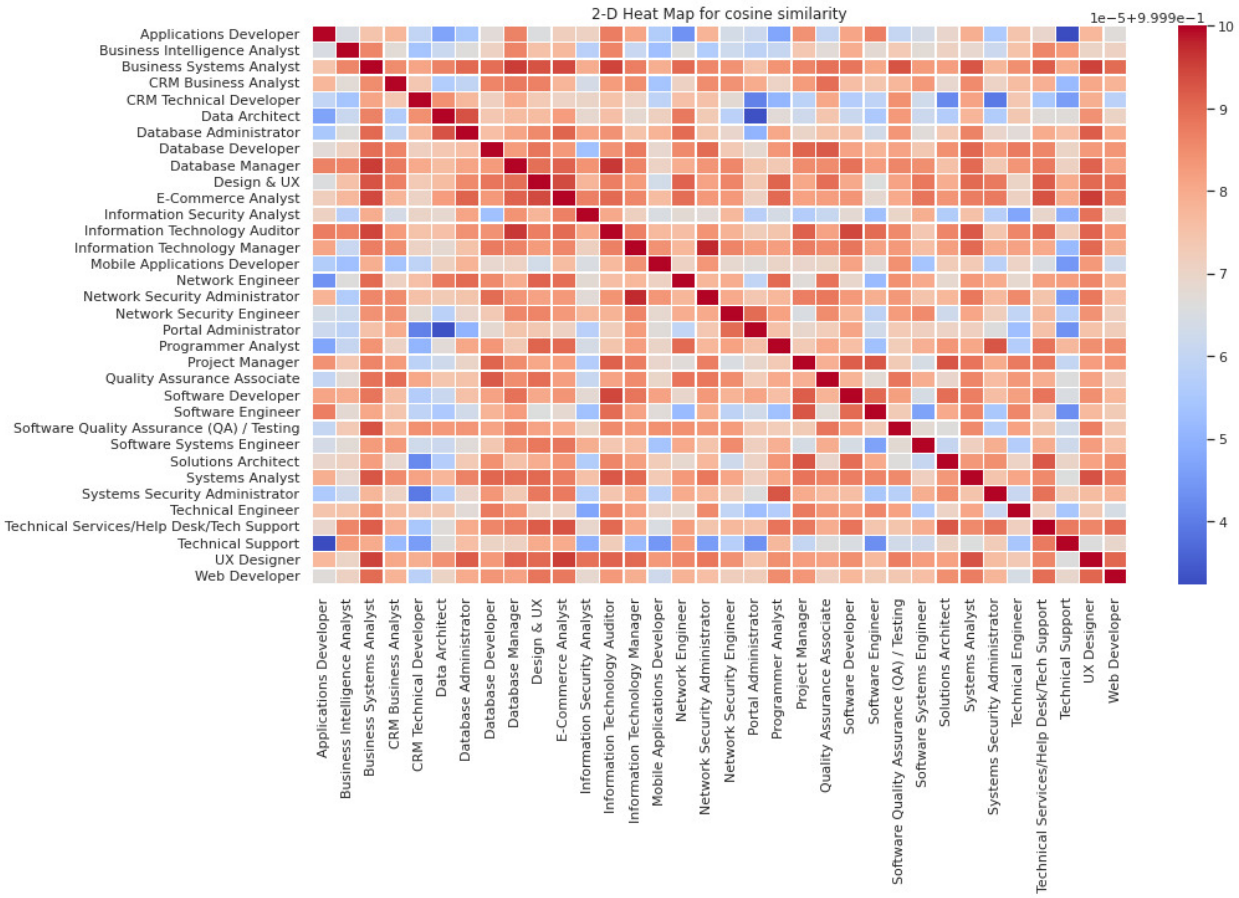
Now, Printing the unique values in each Non Numeric column:

```
Column Name:  can work long time before system?
Number of Unique Values:  2
['no' 'yes']

Column Name:  self-learning capability?
Number of Unique Values:  2
['no' 'yes']

Column Name:  Extra-courses did
Number of Unique Values:  2
['no' 'yes']

Column Name:  certifications
Number of Unique Values:  9
['app development' 'distro making' 'full stack' 'hadoop'
 'information security' 'machine learning' 'python' 'r programming'
 'shell programming']

Column Name:  workshops
Number of Unique Values:  8
['cloud computing' 'data science' 'database security' 'game development'
 'hacking' 'system designing' 'testing' 'web technologies']

Column Name:  talenttests taken?
Number of Unique Values:  2
['no' 'yes']

Column Name:  olympiads
Number of Unique Values:  2
['no' 'yes']

Column Name:  reading and writing skills
Number of Unique Values:  3
['excellent' 'medium' 'poor']

Column Name:  memory capability score
Number of Unique Values:  3
['excellent' 'medium' 'poor']

Column Name:  Interested subjects
Number of Unique Values:  10
['Computer Architecture' 'IOT' 'Management' 'Software Engineering'
 'cloud computing' 'data engineering' 'hacking' 'networks'
 'parallel computing' 'programming']

Column Name:  interested career area
```

```
Number of Unique Values:  6
['Business process analyst' 'cloud computing' 'developer' 'security'
 'system developer' 'testing']

Column Name:  Job/Higher Studies?
Number of Unique Values:  2
['higherstudies' 'job']

Column Name:  Type of company want to settle in?
Number of Unique Values:  10
['BPA' 'Cloud Services' 'Finance' 'Product based' 'SAaS services'
 'Sales and Marketing' 'Service Based' 'Testing and Maintainance Services'
 'Web Services' 'product development']

Column Name:  Taken inputs from seniors or elders
Number of Unique Values:  2
['no' 'yes']

Column Name:  interested in games
Number of Unique Values:  2
['no' 'yes']

Column Name:  Interested Type of Books
Number of Unique Values:  31
['Action and Adventure' 'Anthology' 'Art' 'Autobiographies' 'Biographies'
 'Childrens' 'Comics' 'Cookbooks' 'Diaries' 'Dictionaries' 'Drama'
 'Encyclopedias' 'Fantasy' 'Guide' 'Health' 'History' 'Horror' 'Journals'
 'Math' 'Mystery' 'Poetry' 'Prayer books' 'Religion-Spirituality'
 'Romance' 'Satire' 'Science' 'Science fiction' 'Self help' 'Series'
 'Travel' 'Trilogy']

Column Name:  Salary Range Expected
Number of Unique Values:  2
['Work' 'salary']

Column Name:  In a Realtionship?
Number of Unique Values:  2
['no' 'yes']

Column Name:  Gentle or Tuff behaviour?
Number of Unique Values:  2
['gentle' 'stubborn']

Column Name:  Management or Technical
Number of Unique Values:  2
['Management' 'Technical']
```

```
Column Name:  Salary/work
Number of Unique Values:  2
['salary' 'work']

Column Name:  hard/smart worker
Number of Unique Values:  2
['hard worker' 'smart worker']

Column Name:  worked in teams ever?
Number of Unique Values:  2
['no' 'yes']

Column Name:  Introvert
Number of Unique Values:  2
['no' 'yes']

Column Name:  Suggested Job Role
Number of Unique Values:  34
['Applications Developer' 'Business Intelligence Analyst'
 'Business Systems Analyst' 'CRM Business Analyst'
 'CRM Technical Developer' 'Data Architect' 'Database Administrator'
 'Database Developer' 'Database Manager' 'Design & UX'
 'E-Commerce Analyst' 'Information Security Analyst'
 'Information Technology Auditor' 'Information Technology Manager'
 'Mobile Applications Developer' 'Network Engineer'
 'Network Security Administrator' 'Network Security Engineer'
 'Portal Administrator' 'Programmer Analyst' 'Project Manager'
 'Quality Assurance Associate' 'Software Developer' 'Software Engineer'
 'Software Quality Assurance (QA) / Testing' 'Software Systems Engineer'
 'Solutions Architect' 'Systems Analyst' 'Systems Security Administrator'
 'Technical Engineer' 'Technical Services/Help Desk/Tech Support’
 'Technical Support' 'UX Designer' 'Web Developer']
```

HeatMap:



Analysis:

This shows us the information about the numeric and non-numeric columns in the dataset. Specifically, whether any value is allowed to be null or not is also specified through this information. Printing all the unique values and the count of unique values of all the columns. This information will help us in the label encoding of the dataset. The type of the columns helps us to check if there lie any unwanted values that need to be removed in order to make the dataset perform better. The heatmap show the correlation between the different columns thus this helps us in predicting the important features.

Data Preprocessing:

1. **Handling if any missing values**

   The cells in the whole dataset are checked if there exists any empty values or missing values that need to be removed or filled with any zero value.

   Output:

```
Acedamic percentage in Operating Systems    0
percentage in Algorithms                    0
```

```
Percentage in Programming Concepts          0
Percentage in Software Engineering          0
Percentage in Computer Networks             0
Percentage in Electronics Subjects          0
Percentage in Computer Architecture         0
Percentage in Mathematics                   0
Percentage in Communication skills          0
Hours working per day                       0
Logical quotient rating                     0
hackathons                                  0
coding skills rating                        0
public speaking points                      0
can work long time before system?           0
self-learning capability?                   0
Extra-courses did                           0
certifications                              0
workshops                                   0
talenttests taken?                          0
olympiads                                   0
reading and writing skills                  0
memory capability score                     0
Interested subjects                         0
interested career area                      0
Job/Higher Studies?                         0
Type of company want to settle in?          0
Taken inputs from seniors or elders         0
interested in games                         0
Interested Type of Books                    0
Salary Range Expected                       0
In a Realtionship?                          0
Gentle or Tuff behaviour?                   0
Management or Technical                     0
Salary/work                                 0
hard/smart worker                           0
worked in teams ever?                       0
Introvert                                   0
Suggested Job Role                          0
dtype: int64

Any Empty Cell :  Acedamic percentage in Operating Systems    False
percentage in Algorithms                    False
Percentage in Programming Concepts          False
Percentage in Software Engineering          False
Percentage in Computer Networks             False
Percentage in Electronics Subjects          False
Percentage in Computer Architecture         False
Percentage in Mathematics                   False
```

```
Percentage in Communication skills          False
Hours working per day                       False
Logical quotient rating                     False
hackathons                                  False
coding skills rating                        False
public speaking points                      False
can work long time before system?           False
self-learning capability?                   False
Extra-courses did                           False
certifications                              False
workshops                                   False
talenttests taken?                          False
olympiads                                   False
reading and writing skills                  False
memory capability score                     False
Interested subjects                         False
interested career area                      False
Job/Higher Studies?                         False
Type of company want to settle in?          False
Taken inputs from seniors or elders         False
interested in games                         False
Interested Type of Books                    False
Salary Range Expected                       False
In a Realtionship?                          False
Gentle or Tuff behaviour?                   False
Management or Technical                     False
Salary/work                                 False
hard/smart worker                           False
worked in teams ever?                       False
Introvert                                   False
Suggested Job Role                          False
dtype: bool
```

Analysis

This shows us that there is no missing value present in the data.

2. **Checking if the percentage values lie between 0 to 100:**

   So, the values of the numeric percentage storing columns of the subjects are checked against if they lie within the range of 0 to 100.

   Output:

   ```
   All values are in range
   ```

   Analysis:

   This shows us that there is no percentage that lies outside the boundary.

3. **Label Encoding the Categorical/Non-Numeric Values:**
   So, the non-numeric columns or the categorical columns are labelled encoded.
   Label Encoding is a popular encoding technique for handling categorical variables. In this technique, each label is assigned a unique integer based on alphabetical ordering. We don't need dummy variables that's why One hot encoding is not used over here. So, all the columns from 14 to 38 are label encoded and hence, the numerical values were assigned and stored in the columns.

   Output: Columns after Label Encoding

```
array([[1, 1, 1, ..., 0, 1, 0],
       [1, 0, 1, ..., 0, 0, 1],
       [1, 0, 1, ..., 0, 0, 1],
       ...,
       [1, 1, 1, ..., 0, 0, 1],
       [0, 0, 0, ..., 1, 1, 0],
       [1, 1, 1, ..., 0, 1, 0]])
```

   Analysis:
   Label encoding helps the machine learning algorithms to consider that columns as features since the machines cant understand the text. That's why Label encoding is used to handle the categorical data. One hot encoding creates dummy variables that are of no use regarding the type of the dataset.

4. **Feature Scaling the Numeric Values:**
   So, the numeric data columns need to be handled and should be normalised in order to rescale the whole dataset. Normalization also makes the training process less sensitive to the scale of the features. This results in getting better coefficients after training.

   This process of making features more suitable for training by rescaling is called feature scaling. So, the columns 0 to 14 are normalised between the range 0 to 1.

   Output:

```
array([[0.28508989,  0.26029947,  0.32227553,  ...,  0.          ,
0.01652695,
        0.0330539 ],
            [0.34997994,  0.27818918,  0.32754533,  ...,  0.00448692,
0.00897384,
        0.01346077],
```

```
            [0.29501244,  0.35733901,  0.37811453,  ...,  0.01662042,
0.0041551 ,
        0.01246531],
       ...,
            [0.3430836  ,  0.28934762,  0.33068299,  ...,  0.02480122,
0.00826707,
        0.01240061],
            [0.29479246,  0.37716094,  0.39450167,  ...,  0.01734073,
0.03901665,
        0.02167592],
            [0.31140896,  0.32847247,  0.31567484,  ...,  0.00426588,
0.02986113,
        0.02559526]])
```

Analysis:

As we know that the Machine Learning Algorithms tend to perform better or converge faster when the different features are on a smaller scale. Therefore normalisation plays an important role in the data preprocessing/modification.

Final Input Features:

So, the normalised data and the label encoded data is appended together which forms the final input features.

```
Shape of X - Input:  (20000, 38)

array([[0.28508989, 0.26029947, 0.32227553, ..., 0.        , 1.        ,
        0.        ],
       [0.34997994, 0.27818918, 0.32754533, ..., 0.        , 0.        ,
        1.        ],
       [0.29501244, 0.35733901, 0.37811453, ..., 0.        , 0.        ,
        1.        ],
       ...,
       [0.3430836 , 0.28934762, 0.33068299, ..., 0.        , 0.        ,
        1.        ],
       [0.29479246, 0.37716094, 0.39450167, ..., 1.        , 1.        ,
        0.        ],
       [0.31140896, 0.32847247, 0.31567484, ..., 0.        , 1.        ,
        0.        ]])
```

Combining the Output Labels:

1. **Manually Combining the labels:**

   All the unique labels present inside the 'Suggested Job Role' columns are combined using the manual method into 5 classes. The initial count of all the jobs are printed and later combined such that the count of all the new classes remains almost the same. The classes were clustered using the real-life definition of the jobs. Later, the obtained classes are label encoded into the number of classes present.

   Output:

   Initial Count of all the labels

```
Counter({'Applications Developer': 551,
         'Business Intelligence Analyst': 540,
         'Business Systems Analyst': 582,
         'CRM Business Analyst': 584,
         'CRM Technical Developer': 567,
         'Data Architect': 564,
         'Database Administrator': 593,
         'Database Developer': 581,
         'Database Manager': 570,
         'Design & UX': 588,
         'E-Commerce Analyst': 546,
         'Information Security Analyst': 543,
         'Information Technology Auditor': 558,
         'Information Technology Manager': 591,
         'Mobile Applications Developer': 538,
         'Network Engineer': 621,
         'Network Security Administrator': 1112,
         'Network Security Engineer': 630,
         'Portal Administrator': 593,
         'Programmer Analyst': 529,
         'Project Manager': 602,
         'Quality Assurance Associate': 565,
         'Software Developer': 587,
         'Software Engineer': 590,
         'Software Quality Assurance (QA) / Testing': 571,
         'Software Systems Engineer': 575,
         'Solutions Architect': 578,
         'Systems Analyst': 550,
         'Systems Security Administrator': 562,
         'Technical Engineer': 557,
         'Technical Services/Help Desk/Tech Support': 558,
         'Technical Support': 565,
         'UX Designer': 589,
         'Web Developer': 570})
```

#Creating the output columns to 5 to 7 classes

#Converting 'Business Intelligence Analyst', 'Business Systems Analyst','CRM Business Analyst', 'E-Commerce Analyst' 'Information Security Analyst', 'Systems Analyst', 'Programmer Analyst'

#Converting 'Applications Developer','Database Developer','Mobile Applications Developer','Web Developer', 'Software Developer' to 'Developer;

#Converting 'Quality Assurance Associate','Software Quality Assurance (QA) / Testing' to 'Quality Assurance'

#Converting 'Network Engineer', 'Network Security Administrator','Network Security Engineer', 'Portal Administrator','Systems Security Administrator' to 'Network Security'

#Converting 'UX Designer','Design & UX' 'Developer' to 'UX Designer'

#Converting 'CRM Technical Developer', 'Technical Engineer','Technical Services/Help Desk/Tech Support', 'Technical Support'

#Converting 'Data Architect' 'Database Administrator' 'Database Manager' to 'Database Engineer

#Converting 'Information Technology Auditor', 'Information Technology Manager', 'Software Engineer', 'Software Systems Engineer','Solutions Architect' to Software Engineer'

#Converting 'Developer', 'UX Designer' to 'Developer/UX Designer'

#Converting 'Software Engineer' , 'Quality Assurance' to 'Software Engineer/Quality Assurance'

#Converting 'Technical Support' , 'Database Engineer' to 'Technical Support/Database Engineer'

#Converting 'Project Manager' , 'Network Security' to 'Network Security/Project Manager'

Final Count of all the labels after combining the classes

```
Counter({'Analyst': 3874,
         'Developer/UX Designer': 4004,
         'Network Security/Project Manager': 4120,
         'Software Engineer/Quality Assurance': 4028,
         'Technical Support/Database Engineer': 3974})
```

```
Total Number of Classes: 5

Label Encoded Output:
array([1, 2, 2, ..., 0, 3, 1])
```

Analysis:

So, this combines the 34 unique jobs into 5 jobs through the manual method. Through this, the machine learning model will only have to predict the 5 classes in the classification task.

2. **Combining Features using Cosine Similarity between labels**

All the unique labels present inside the 'Suggested Job Role' columns are combined using the cosine similarity method into 6 classes. The initial count of all the jobs are printed and later combined such that the count of all the new classes remains almost the same. The classes were clustered using the cosine similarity between the jobs. The data of each class is separated and the average of each class formed through the columns is stored. Then for each class, a list of the average of each column is present. Then the cosine similarity is found out between 2 classes one by one. Later the top 10 classes are combined such that there is no repetition and new 6 classes are formed. Later, the obtained classes are labeled encoded into the number of classes present.

Output:

```
Initial Count of all the labels

Counter({'Applications Developer': 551,
         'Business Intelligence Analyst': 540,
         'Business Systems Analyst': 582,
         'CRM Business Analyst': 584,
         'CRM Technical Developer': 567,
         'Data Architect': 564,
         'Database Administrator': 593,
         'Database Developer': 581,
         'Database Manager': 570,
         'Design & UX': 588,
         'E-Commerce Analyst': 546,
         'Information Security Analyst': 543,
         'Information Technology Auditor': 558,
         'Information Technology Manager': 591,
         'Mobile Applications Developer': 538,
         'Network Engineer': 621,
         'Network Security Administrator': 1112,
```

```
'Network Security Engineer': 630,
'Portal Administrator': 593,
'Programmer Analyst': 529,
'Project Manager': 602,
'Quality Assurance Associate': 565,
'Software Developer': 587,
'Software Engineer': 590,
'Software Quality Assurance (QA) / Testing': 571,
'Software Systems Engineer': 575,
'Solutions Architect': 578,
'Systems Analyst': 550,
'Systems Security Administrator': 562,
'Technical Engineer': 557,
'Technical Services/Help Desk/Tech Support': 558,
'Technical Support': 565,
'UX Designer': 589,
'Web Developer': 570})
```

Mean Values of each of the Class label

```
Label:  Applications Developer
[76.52450090744102,       77.0617059891107,       76.93466424682396,
76.19600725952813,       76.4010889292196,       77.65880217785843,
76.38656987295826,       77.20145190562613,       76.9382940108893,
8.107078039927405,       4.778584392014519,       2.882032667876588,
4.947368421052632,      4.907441016333938,      0.49727767695099817,
0.49364791288566245,     0.49183303085299457,     4.192377495462795,
3.442831215970962,      0.5444646098003629,      0.5117967332123412,
0.9528130671506352,      1.0235934664246824,      4.5535390199637025,
2.4500907441016335,      0.47186932849364793,      4.511796733212341,
0.47005444646098005,      0.4863883847549909,     15.343012704174228,
0.49909255898366606,     0.5045372050816697,     0.49909255898366606,
0.5335753176043557,      0.4827586206896552,      0.5027223230490018,
0.49727767695099817, 0.4882032667876588]

Label:  Business Intelligence Analyst
[76.74259259259259,       78.05555555555556,       76.6462962962963,
77.28703703703704,       76.5962962962963,       76.7611111111111,
76.69074074074074,       76.63333333333334,       76.52777777777777,
8.12037037037037,    4.814814814814815,    3.0,    5.020370370370371,
5.064814814814815,      0.5166666666666667,      0.5111111111111111,
0.4944444444444446,      4.022222222222222,      3.564814814814815,
0.4962962962962963,      0.5166666666666667,      0.987037037037037,
1.0148148148148148,      4.462962962962963,      2.4444444444444446,
0.48703703703703705,      4.518518518518518,      0.4981481481481482,
0.4537037037037037,      15.533333333333333,      0.5481481481481482,
0.4777777777777778,      0.4925925925925926,      0.4981481481481482,
0.48148148148148145, 0.45, 0.4666666666666667, 0.49074074074074076]
```

Label: Business Systems Analyst
[76.9639175257732, 77.15463917525773, 77.09278350515464,
77.18556701030928, 76.78694158075601, 76.91065292096219,
77.17697594501718, 76.72852233676976, 76.79381443298969,
7.828178694158075, 4.981099656357388, 3.008591065292096,
5.054982817869416, 5.001718213058419, 0.5120274914089347,
0.5223367697594502, 0.5154639175257731, 4.219931271477663,
3.4673539518900345, 0.5274914089347079, 0.4896907216494845,
0.9896907216494846, 0.9914089347079038, 4.606529209621993,
2.551546391752577, 0.5120274914089347, 4.367697594501718,
0.4982817869415807 6, 0.5017182130584192, 15.427835051546392,
0.4810996563573883 3, 0.5051546391752577, 0.4862542955326460 4,
0.4673539518900343 7, 0.5, 0.5, 0.5, 0.5395189003436426]

Label: CRM Business Analyst
[76.81849315068493, 76.95719178082192, 77.0736301369863,
76.11301369863014, 77.07191780821918, 76.66267123287672, 76.875,
77.16609589041096, 76.4332191780822, 7.964041095890411,
5.198630136986301, 3.0547945205479454, 5.006849315068493,
5.191780821917808, 0.488013698630137, 0.5462328767123288,
0.541095890410959, 3.892123287671233, 3.4674657534246576,
0.4914383561643836, 0.4931506849315068, 0.9246575342465754,
0.976027397260274, 4.417808219178082, 2.440068493150685,
0.4914383561643836, 4.532534246575342, 0.5034246575342466,
0.5273972602739726, 14.57876712328767, 0.4537671232876712,
0.5188356164383562, 0.5205479452054794, 0.5051369863013698,
0.4811643835616438, 0.4743150684931507, 0.4537671232876712,
0.4948630136986301]

…….
Label: Technical Services/Help Desk/Tech Support
[76.9605734767025, 76.99283154121864, 76.15412186379929,
77.15591397849462, 76.64336917562724, 76.71863799283155,
76.76702508960574, 76.92831541218638, 76.47849462365592,
7.991039426523297, 5.087813620071684, 3.046594982078853,
5.094982078853047, 5.066308243727598, 0.5519713261648745,
0.5412186379928315, 0.503584229390681, 3.9372759856630823,
3.57168458781362, 0.553763440860215, 0.5376344086021505,
1.010752688172043, 0.9802867383512545, 4.465949820788531,
2.4802867383512543, 0.478494623655914, 4.496415770609319,
0.4749103942652329 6, 0.510752688172043, 15.412186379928315,
0.514336917562724, 0.496415770609319, 0.5125448028673835,
0.507168458781362, 0.532258064516129, 0.4767025089605735,
0.5089605734767025, 0.507168458781362]

Label: Technical Support

[77.3929203539823, 77.55221238938053, 76.65663716814159,
78.29380530973451, 76.94336283185841, 76.58938053097346,
77.02477876106195, 76.92743362831858, 76.4353982300885,
8.138053097345132, 5.070796460176991, 3.2601769911504426,
5.256637168141593, 4.899115044247788, 0.4849557522123894,
0.4902654867256637, 0.48672566371681414, 4.063716814159292,
3.6814159292035398, 0.5150442477876106, 0.4920353982300885,
1.0106194690265486, 0.95929203539823, 4.430088495575221,
2.51858407079646, 0.49911504424778763, 4.6300884955752215,
0.4831858407079646, 0.4725663716814159, 15.739823008849557,
0.5026548672566372, 0.5168141592920354, 0.5150442477876106,
0.4761061946902655, 0.5185840707964602, 0.5079646017699115,
0.4725663716814159, 0.5539823008849557]

Label: UX Designer
[77.21222410865875, 76.86757215619694, 77.2937181663837,
77.25127334465195, 76.6723259762309, 77.23089983022071,
77.51103565365025, 77.12563667232598, 77.27843803056027,
7.923599320882852, 5.030560271646859, 2.9507640067911716,
5.016977928692699, 4.921901528013582, 0.5161290322580645,
0.5229202037351444, 0.4617996604414261, 4.168081494057725,
3.5534804753820035, 0.4855687606112054, 0.5076400679117148,
0.9864176570458404, 0.9932088285229203, 4.422750424448218,
2.507640067911715, 0.4770797962648557, 4.721561969439728,
0.5144312393887945, 0.47877758913412566, 14.923599320882852,
0.5144312393887945, 0.49745331069609505, 0.5178268251273345,
0.5195246179966044, 0.5212224108658744, 0.5297113752122241,
0.4838709677419355, 0.5280135823429541]

Label: Web Developer
[77.26491228070175, 77.10701754385966, 76.69824561403509,
77.16491228070176, 76.77894736842106, 77.24035087719298,
77.94385964912281, 77.31052631578947, 76.72456140350877,
7.777192982456141, 4.873684210526315, 2.9719298245614034,
5.06140350877193, 4.970175438596491, 0.47017543859649125,
0.5263157894736842, 0.5052631578947369, 3.93859649122807,
3.5982456140350876, 0.48947368421052634, 0.4982456140350877,
1.012280701754386, 1.0140350877192983, 4.491228070175438,
2.5052631578947366, 0.4842105263157895, 4.282456140350877,
0.5245614035087719, 0.49473684210526314, 15.398245614035087,
0.5263157894736842, 0.49298245614035086, 0.5, 0.48771929824561405,
0.5140350877192983, 0.4964912280701754, 0.5175438596491229,
0.531578947368421]

Cosine Similar top 10 Classes
Class: Applications Developer
Applications Developer

Database Manager
Software Engineer
Information Technology Auditor
Project Manager
Information Technology Manager
Software Developer
Network Security Administrator
Systems Analyst
UX Designer

Class: Business Intelligence Analyst
Business Intelligence Analyst
Technical Services/Help Desk/Tech Support
Business Systems Analyst
Information Technology Auditor
Database Manager
Technical Support
E-Commerce Analyst
Design & UX
Software Developer
Technical Engineer

Class: Business Systems Analyst
Business Systems Analyst
Database Manager
E-Commerce Analyst
Design & UX
UX Designer
Technical Services/Help Desk/Tech Support
Information Technology Auditor
Network Engineer
Database Developer
Web Developer

Class: CRM Business Analyst
CRM Business Analyst
Database Manager
Systems Analyst
Quality Assurance Associate
Information Technology Manager
Network Security Administrator
Database Developer
UX Designer
Portal Administrator

Software Systems Engineer

Class: CRM Technical Developer
CRM Technical Developer
Data Architect
Software Quality Assurance (QA) / Testing
Systems Analyst
Database Manager
UX Designer
Quality Assurance Associate
Business Systems Analyst
Database Administrator
Network Security Administrator

Class: Data Architect
Data Architect
Database Administrator
CRM Technical Developer
Network Engineer
UX Designer
Software Quality Assurance (QA) / Testing
Business Systems Analyst
Systems Analyst
E-Commerce Analyst
Database Manager

…….

Class: UX Designer
UX Designer
Systems Analyst
E-Commerce Analyst
Business Systems Analyst
Database Administrator
Information Technology Manager
Database Manager
Network Security Administrator
Information Security Analyst
Web Developer

Class: Web Developer
Web Developer
Technical Services/Help Desk/Tech Support

Business Systems Analyst
E-Commerce Analyst
Design & UX
UX Designer
Solutions Architect
Programmer Analyst
Systems Analyst
Database Developer

#Creating the output columns to 5 to 7 classes

#Converting 'Applications Developer','Database Manager','Software Engineer','Information Technology Auditor','Project Manager','Information Technology Manager'

#Converting 'Business Intelligence Analyst','Technical Services/Help Desk/Tech Support','Business Systems Analyst','Technical Support','E-Commerce Analyst','Design & UX' to 'Analyst;
#Converting 'CRM Business Analyst','Systems Analyst','Quality Assurance Associate','Network Security Administrator','Database Developer','UX Designer', to 'Administrator'

#Converting 'CRM Technical Developer','Data Architect','Software Quality Assurance (QA) / Testing','Portal Administrator','Database Administrator' to 'QA'

#Converting 'Information Security Analyst','Software Systems Engineer','Network Security Engineer','Web Developer','Network Engineer'to 'Engineer'

#Converting 'Systems Security Administrator', 'Mobile Applications Developer','Programmer Analyst','Software Developer','Solutions Architect', 'Technical Engineer' to 'Developer'

Final Count of all the labels after combining the classes

```
Counter({'Administrator': 3981,
         'Analyst': 3379,
         'Developer': 3351,
         'Engineer': 2939,
         'QA': 2888,
         'Software Field': 3462})

Total Number of Classes: 6
```

Label Encoded Output:

```
array([0, 4, 4, ..., 1, 4, 5])
```

Analysis:

So, this combines the 34 unique jobs into 6 jobs through the cosine similarity method. Through this, the machine learning model will only have to predict the 6 classes in the classification task. Cosine similarity is a metric used to measure how similar the documents are irrespective of their size. So, this ssimiarity helps us to cluster the same jobs into one class.

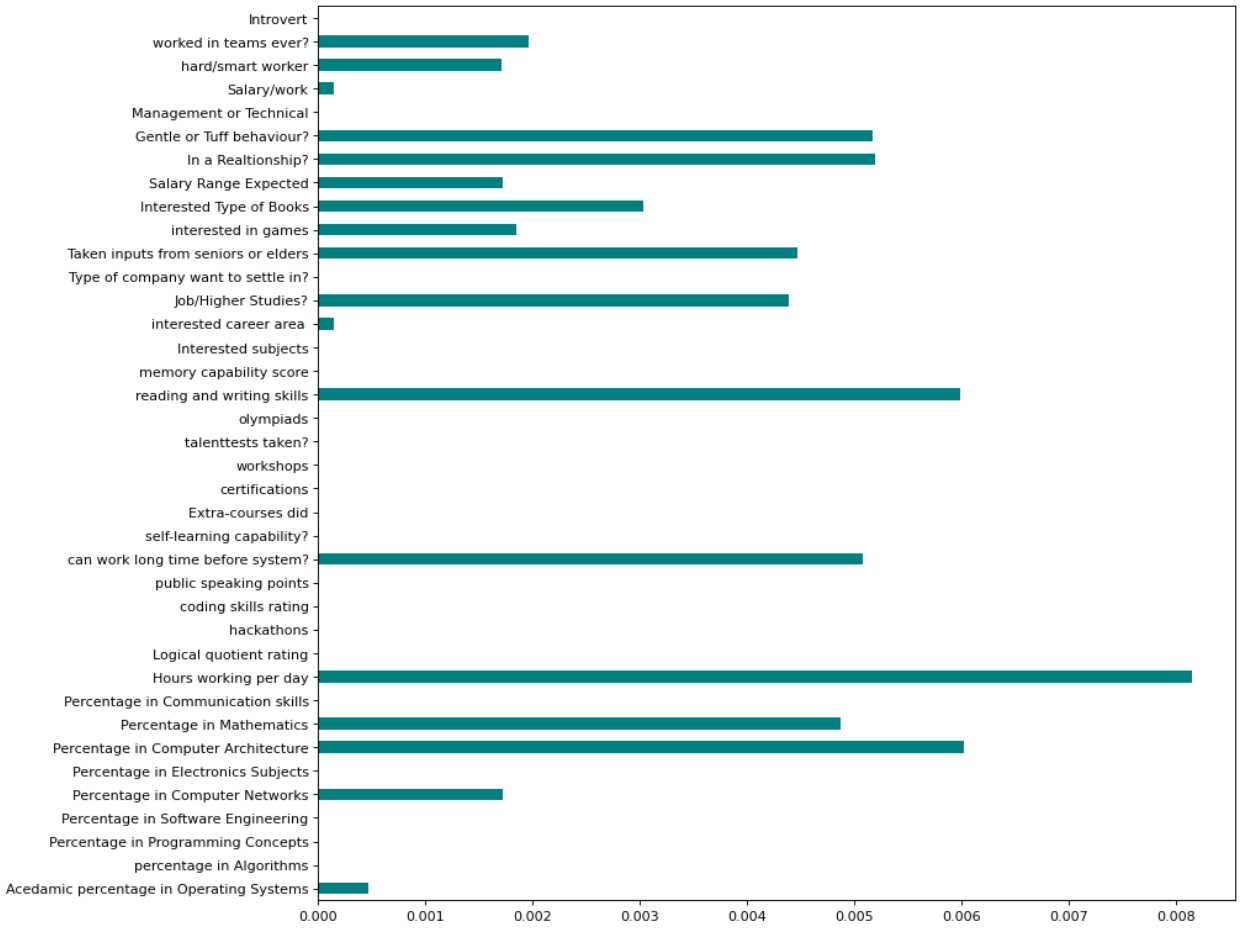
Feature Selection:

So, the mutual_info_classif from the feature selection of the sklearn was used in order to predict the important features which are adding fruit to the machine learning model. Information gain calculates the reduction in entropy from the transformation of a dataset. It can be used for feature selection by evaluating the Information gain of each variable in the context of the target variable. The top 10 features are extracted and used tosend to the ANN model and see if any much improvement in accuracy occurs.

Output:

```
Acedamic percentage in Operating Systems    0.000466
percentage in Algorithms                    0.000000
Percentage in Programming Concepts          0.000000
Percentage in Software Engineering          0.000000
Percentage in Computer Networks             0.001717
Percentage in Electronics Subjects          0.000000
Percentage in Computer Architecture         0.006016
Percentage in Mathematics                   0.004877
Percentage in Communication skills          0.000000
Hours working per day                       0.008149
Logical quotient rating                     0.000000
hackathons                                  0.000000
coding skills rating                        0.000000
public speaking points                      0.000000
can work long time before system?           0.005079
self-learning capability?                   0.000000
Extra-courses did                           0.000000
certifications                              0.000000
workshops                                   0.000000
talenttests taken?                          0.000000
olympiads                                   0.000000
```

reading and writing skills	0.005990
memory capability score	0.000000
Interested subjects	0.000000
interested career area	0.000143
Job/Higher Studies?	0.004392
Type of company want to settle in?	0.000000
Taken inputs from seniors or elders	0.004473
interested in games	0.001847
Interested Type of Books	0.003029
Salary Range Expected	0.001724
In a Realtionship?	0.005192
Gentle or Tuff behaviour?	0.005167
Management or Technical	0.000000
Salary/work	0.000146
hard/smart worker	0.001707
worked in teams ever?	0.001964
Introvert	0.000000



Analysis:

So, the top 10 features are extracted which are adding more information to the Machine Learning models and hence they are used further to predict the job profile. The goal of feature selection in machine learning is to find the best set of features that allows one to build useful models of studied phenomena.

Model Evaluation/Experiments:

1. **ANN Model on both types of Output Clubbing Techniques and Varying Test Sizes:**
   So, The MLP Classifier of the sklearn is used which classifies the data into the following output classes using the number of iterations and hidden layer sizes as one of the parameters.
   The Accuracy, Confusion Matrix, Classwise Accuracy and Classification Report on the test dataset is printed after the model is trained on varying sizes of the data.

   a. **Output Columns Clustering Technique 1 - Manually with Varying Test Size:**

Test Size = 0.5

```
Accuracy score calculated on Test Data :  0.2707

Confusion matrix :  [[650 372 252 307 314]
 [399 576 359 292 288]
 [403 540 400 366 359]
 [383 420 382 446 426]
 [384 418 263 366 635]]

Classwise accuracy :  [0.34300792 0.30094044 0.1934236
0.21682061 0.30735721]

Classification report :
              precision    recall  f1-score   support

           0       0.29      0.34      0.32      1895
           1       0.25      0.30      0.27      1914
           2       0.24      0.19      0.21      2068
           3       0.25      0.22      0.23      2057
           4       0.31      0.31      0.31      2066

    accuracy                           0.27     10000
   macro avg       0.27      0.27      0.27     10000
weighted avg       0.27      0.27      0.27     10000
```

Test Size = 0.4

Accuracy score calculated on Test Data : **0.292875**

```
Confusion matrix :  [[506 330 233 183 240]
 [285 564 248 184 266]
 [277 403 427 256 263]
 [307 358 331 320 382]
 [295 359 234 223 526]]

Classwise accuracy :  [0.33914209 0.3645766  0.26260763
0.18845701 0.32131949]

Classification report :
              precision    recall  f1-score   support

           0       0.30      0.34      0.32      1492
           1       0.28      0.36      0.32      1547
           2       0.29      0.26      0.28      1626
           3       0.27      0.19      0.22      1698
           4       0.31      0.32      0.32      1637

    accuracy                           0.29      8000
   macro avg       0.29      0.30      0.29      8000
weighted avg       0.29      0.29      0.29      8000
```

Test Size = 0.3

Accuracy score calculated on Test Data : **0.31216666666666665**

```
Confusion matrix :  [[394 130 230 196 162]
 [256 194 332 208 187]
 [228 138 425 222 188]
 [224 129 272 446 210]
 [219 145 244 207 414]]

Classwise accuracy :  [0.35431655 0.16482583 0.35387177
0.3481655  0.33685924]

Classification report :
              precision    recall  f1-score   support

           0       0.30      0.35      0.32      1112
           1       0.26      0.16      0.20      1177
           2       0.28      0.35      0.31      1201
           3       0.35      0.35      0.35      1281
           4       0.36      0.34      0.35      1229

    accuracy                           0.31      6000
```

```
   macro avg       0.31      0.31      0.31      6000
weighted avg       0.31      0.31      0.31      6000
```

Test Size = 0.2

Accuracy score calculated on Test Data : **0.322**

```
Confusion matrix :  [[242 133 154  79 137]
 [116 258 159  85 162]
 [120 145 304  85 153]
 [127 160 207 166 191]
 [117 140 164  78 318]]

Classwise  accuracy  :    [0.32483221  0.33076923  0.37670384
0.19506463 0.38922889]

Classification report :
              precision    recall  f1-score   support

           0       0.34      0.32      0.33       745
           1       0.31      0.33      0.32       780
           2       0.31      0.38      0.34       807
           3       0.34      0.20      0.25       851
           4       0.33      0.39      0.36       817

    accuracy                           0.32      4000
   macro avg       0.32      0.32      0.32      4000
weighted avg       0.32      0.32      0.32      4000
```

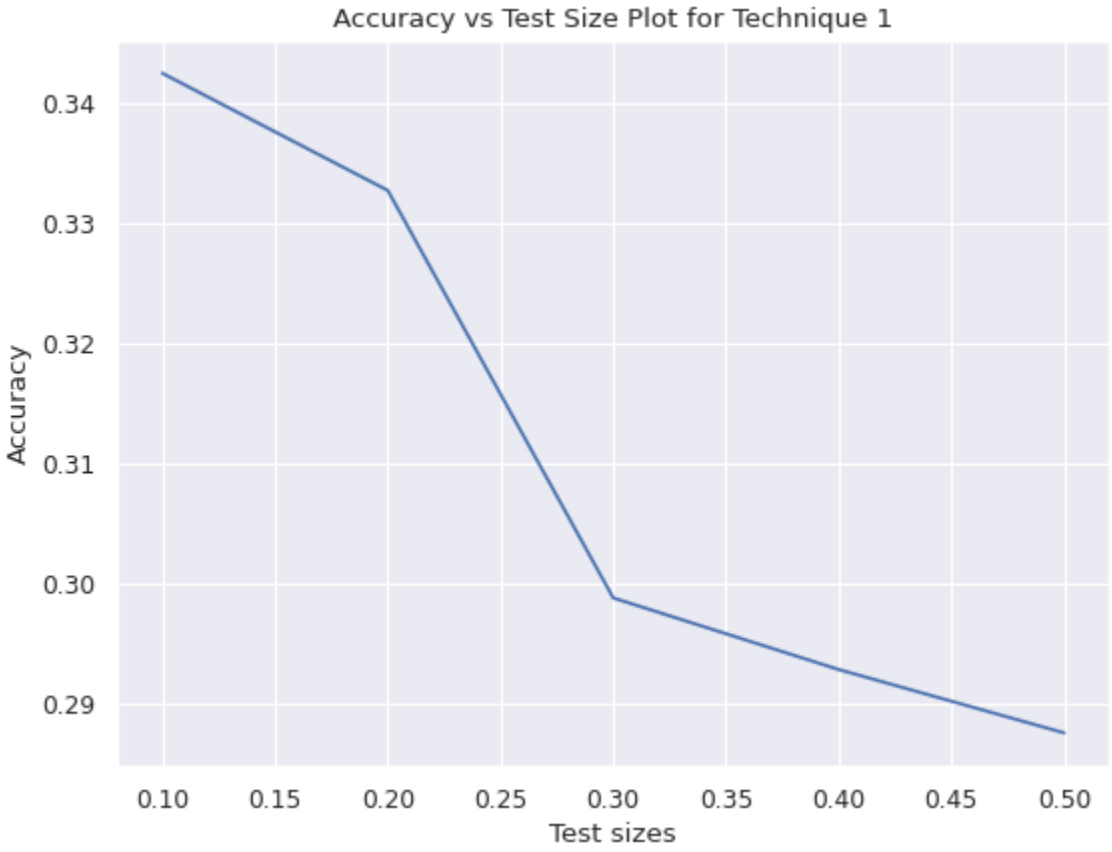
Test Size = 0.1

Accuracy score calculated on Test Data : **0.333**

```
Confusion matrix :  [[113  63  61  89  56]
 [ 59 131  74  75  61]
 [ 61  72 129  80  53]
 [ 60  70  74 137  64]
 [ 45  69  64  84 156]]

Classwise  accuracy  :   [0.29581152  0.3275        0.32658228
0.3382716  0.37320574]

Classification report :
              precision    recall  f1-score   support

           0       0.33      0.30      0.31       382
           1       0.32      0.33      0.33       400
           2       0.32      0.33      0.32       395
           3       0.29      0.34      0.31       405
```

```
                4       0.40      0.37      0.39       418

         accuracy                           0.33      2000
        macro avg       0.33      0.33      0.33      2000
     weighted avg       0.34      0.33      0.33      2000
```

Accuracy vs Test Size Plot:



b. Output Columns Clustering Technique 2 - Cosine Simialrity with Varying Test Size:

Test Size = 0.5

```
Accuracy score calculated on Test Data :  0.2643

Confusion matrix :  [[663 345 264 160 159 350]
 [295 594 287 139 124 266]
 [317 351 419 180 121 297]
 [236 280 300 201 153 272]
 [254 275 241 122 188 374]
 [297 322 253 148 175 578]]
```

```
Classwise accuracy :  [0.34157651 0.3483871  0.24866469
0.13938974 0.12929849 0.32600113]

Classification report :
              precision    recall  f1-score   support

           0       0.32      0.34      0.33      1941
           1       0.27      0.35      0.31      1705
           2       0.24      0.25      0.24      1685
           3       0.21      0.14      0.17      1442
           4       0.20      0.13      0.16      1454
           5       0.27      0.33      0.30      1773

    accuracy                           0.26     10000
   macro avg       0.25      0.26      0.25     10000
weighted avg       0.26      0.26      0.26     10000
```

Test Size = 0.4

```
Accuracy score calculated on Test Data :  0.27775

Confusion matrix :  [[488 220 253 196  85 282]
 [244 380 256 202  57 223]
 [226 181 441 202  75 242]
 [192 173 222 309  74 197]
 [196 152 205 187 113 289]
 [219 209 237 182 100 491]]

Classwise accuracy :  [0.32020997 0.27900147 0.32260424
0.26478149 0.09894921 0.34144645]

Classification report :
              precision    recall  f1-score   support

           0       0.31      0.32      0.32      1524
           1       0.29      0.28      0.28      1362
           2       0.27      0.32      0.30      1367
           3       0.24      0.26      0.25      1167
           4       0.22      0.10      0.14      1142
           5       0.28      0.34      0.31      1438

    accuracy                           0.28      8000
   macro avg       0.27      0.27      0.27      8000
weighted avg       0.27      0.28      0.27      8000
```

Test Size = 0.3

```
Accuracy score calculated on Test Data :  0.2853333333333333
```

```
Confusion matrix :  [[458 166 135 127  73 183]
 [241 304 127 131  78 163]
 [215 195 237 107  86 172]
 [202 127 132 204  69 148]
 [213 117  94 120 130 176]
 [212 175 119 109  76 379]]

Classwise accuracy :  [0.40105079 0.29118774 0.23418972
0.23129252 0.15294118 0.35420561]

Classification report :
              precision    recall  f1-score   support

           0       0.30      0.40      0.34      1142
           1       0.28      0.29      0.29      1044
           2       0.28      0.23      0.26      1012
           3       0.26      0.23      0.24       882
           4       0.25      0.15      0.19       850
           5       0.31      0.35      0.33      1070

    accuracy                           0.29      6000
   macro avg       0.28      0.28      0.27      6000
weighted avg       0.28      0.29      0.28      6000
```

Test Size = 0.2

```
Accuracy score calculated on Test Data :  0.3095

Confusion matrix :  [[272 104 121  90  64 103]
 [105 228 108  85  83  89]
 [113  85 215  83  71 109]
 [107  72 117 142  77  81]
 [109  69  87  75 142  90]
 [105  98 120  64  78 239]]

Classwise  accuracy  :    [0.36074271  0.32664756  0.31804734
0.23825503 0.24825175 0.33948864]

Classification report :
              precision    recall  f1-score   support

           0       0.34      0.36      0.35       754
           1       0.35      0.33      0.34       698
           2       0.28      0.32      0.30       676
           3       0.26      0.24      0.25       596
           4       0.28      0.25      0.26       572
           5       0.34      0.34      0.34       704
```

accuracy			0.31	4000
macro avg	0.31	0.31	0.31	4000
weighted avg	0.31	0.31	0.31	4000

Test Size = 0.1

Accuracy score calculated on Test Data : 0.322

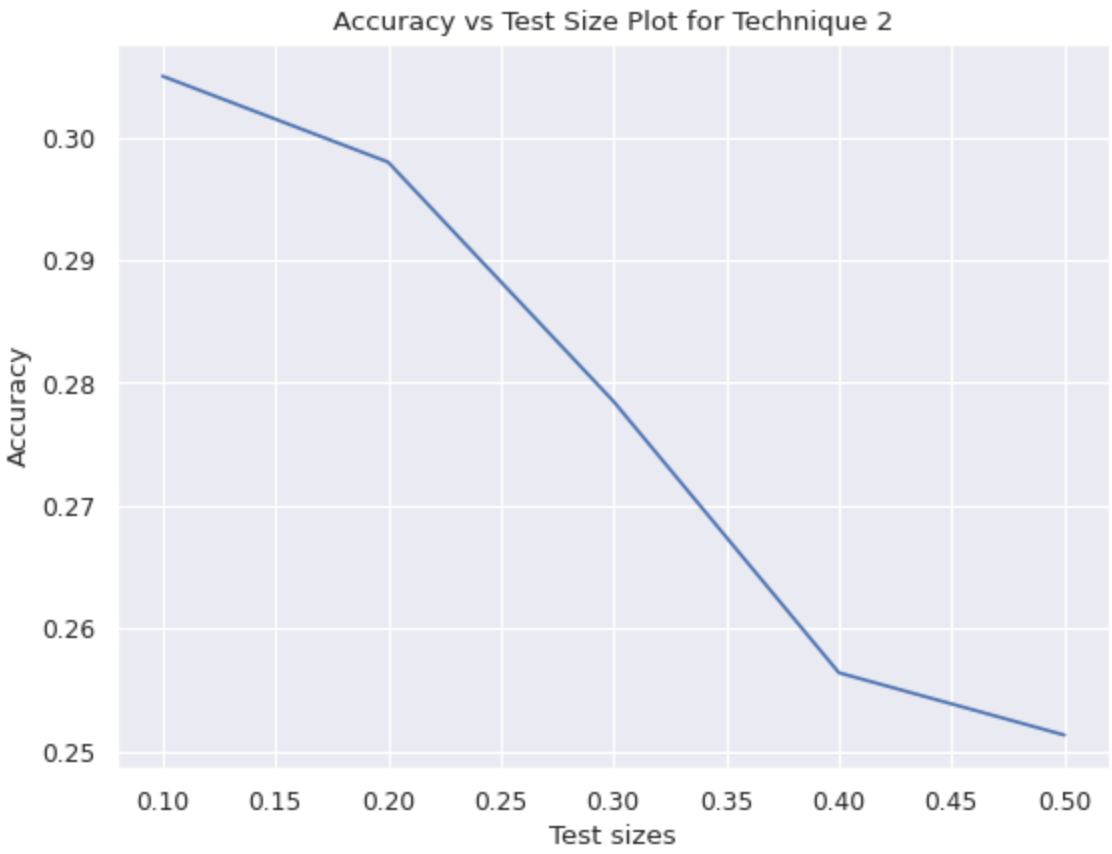
Confusion matrix : [[131 48 53 54 22 61]
[51 113 51 69 17 49]
[66 50 118 49 19 51]
[46 33 38 114 10 40]
[51 32 48 79 43 50]
[57 53 44 49 16 125]]

Classwise accuracy : [0.35501355 0.32285714 0.33427762 0.40569395 0.14191419 0.36337209]

Classification report :

	precision	recall	f1-score	support
0	0.33	0.36	0.34	369
1	0.34	0.32	0.33	350
2	0.34	0.33	0.33	353
3	0.28	0.41	0.33	281
4	0.34	0.14	0.20	303
5	0.33	0.36	0.35	344
accuracy			0.32	2000
macro avg	0.33	0.32	0.31	2000
weighted avg	0.33	0.32	0.32	2000

Accuracy vs Test Size Plot:



Analysis:

1. As we can see that the as the test sizes decreases the accuracy increases for both techniques. Since, keeping the test size as the least 0.1 will overfit the data and test size as 0.5 is underfitting the data.
2. So, test size play a very important role in training the data properly hence, the appropriate value as 0.2 is chosen as the test size.
3. The technique do not play much role in varying test sizes as both the techniques give almost same results only.

2. ANN Model on both types of Output Clubbing Techniques and Varying Hidden Layers and Neurons:

So, The MLP Classifier of the sklearn is used which classifies the data into the following output classes using the number of iterations and hidden layer sizes as one of the parameters. Here, the neurons and layers of the hidden layer are modified in order to obtain the final results.

The Accuracy, Confusion Matrix, Classwise Accuracy and Classification Report on the test dataset is printed after the model is trained on varying sizes of the data.

a. Output Columns Clustering Technique 1 - Manually with varying neurons and layers of the hidden layer:

```
hidden_layers=[(50),(50,50),(25,25),(50,25),(50,50,50),(50,50,25),(50,25,25),(50,50,25,25),(50,50,50,25),(50,50,50,50)]
```

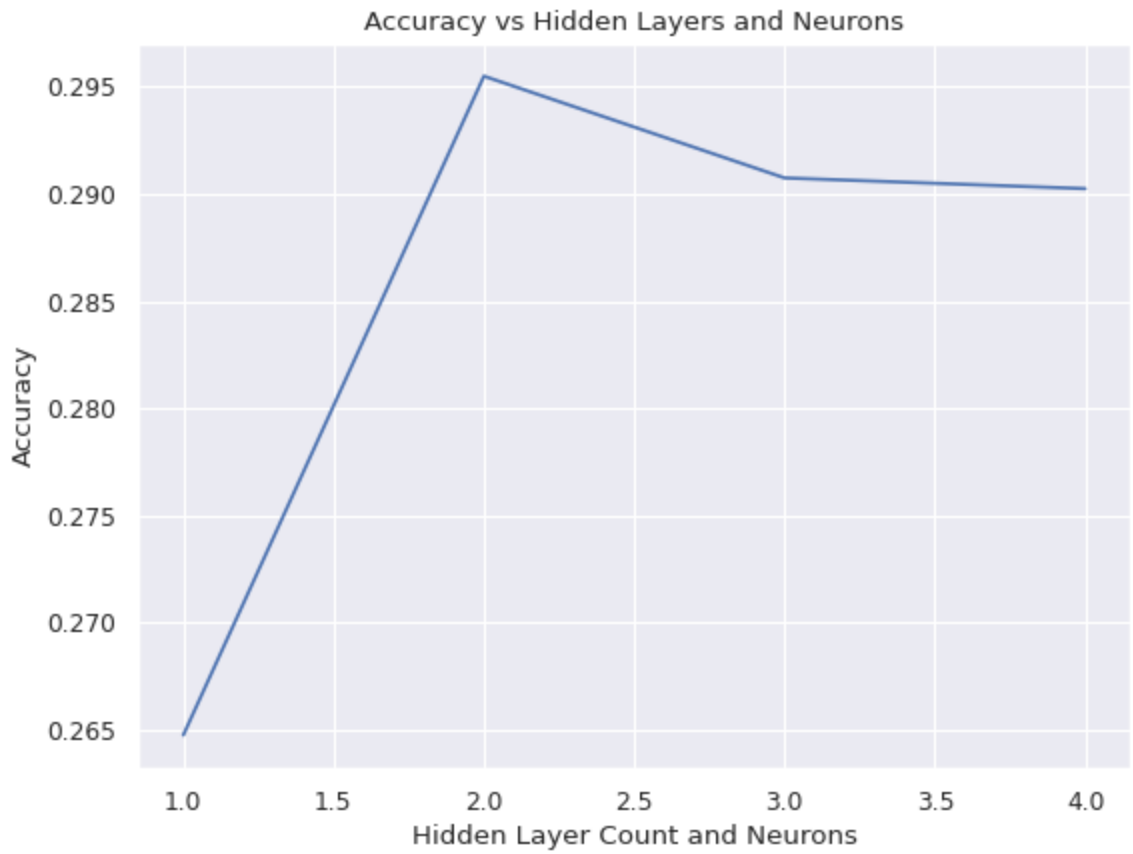
```
Hidden Layer = (50)
Accuracy score calculated on Test Data :  0.26475

Confusion matrix :  [[299  74 107  94  40 140]
 [150 133 155  88  35 137]
 [148 108 146 106  35 133]
 [123  60  92 157  46 118]
 [139  56  70  83  60 164]
 [141  71  87  98  43 264]]

Classwise  accuracy  :    [0.39655172  0.19054441  0.21597633
0.26342282 0.1048951  0.375     ]

Classification report :
              precision    recall  f1-score   support

           0       0.30      0.40      0.34       754
           1       0.26      0.19      0.22       698
           2       0.22      0.22      0.22       676
           3       0.25      0.26      0.26       596
           4       0.23      0.10      0.14       572
           5       0.28      0.38      0.32       704

    accuracy                           0.26      4000
   macro avg       0.26      0.26      0.25      4000
weighted avg       0.26      0.26      0.25      4000

Hidden Layer = (50,25)
Accuracy score calculated on Test Data :  0.33375

Confusion matrix :  [[261 120 149  95 120]
 [138 250 134 120 138]
 [129 123 288 127 140]
 [118 132 152 279 170]
 [114 144 158 144 257]]

Classwise  accuracy  :    [0.35033557  0.32051282  0.35687732
0.32784959 0.31456548]

Classification report :
              precision    recall  f1-score   support
```

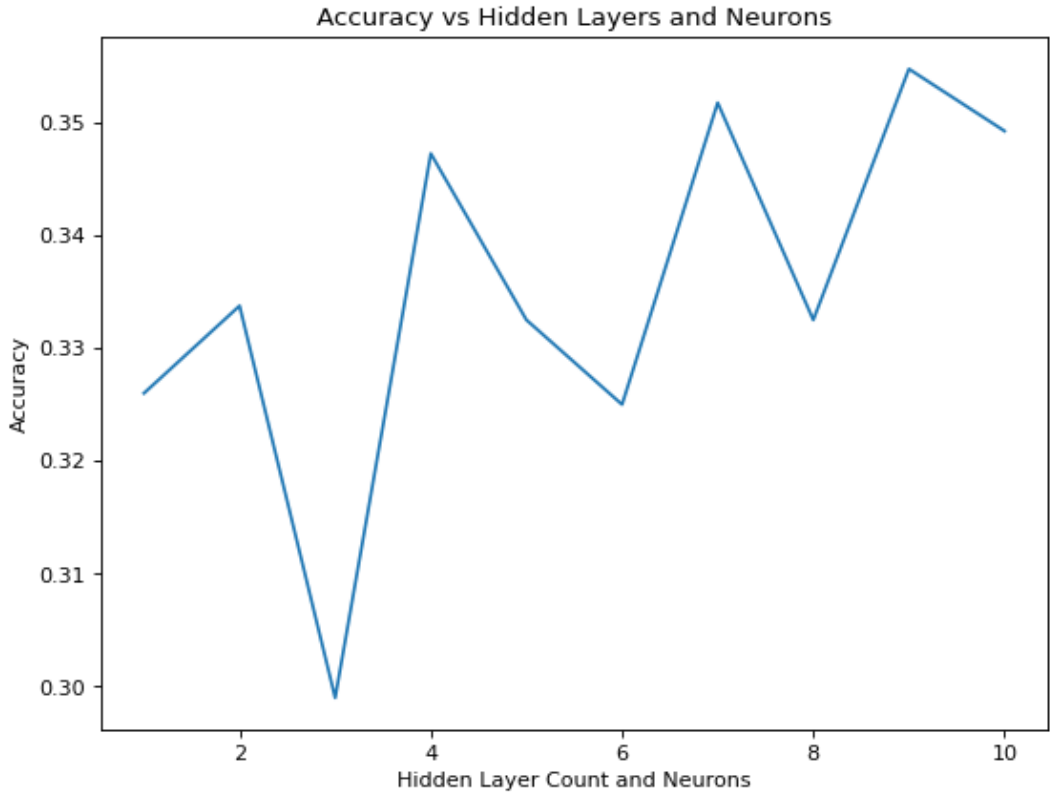
```
              0       0.34      0.35      0.35       745
              1       0.33      0.32      0.32       780
              2       0.33      0.36      0.34       807
              3       0.36      0.33      0.35       851
              4       0.31      0.31      0.31       817

    accuracy                           0.33      4000
   macro avg       0.33      0.33      0.33      4000
weighted avg       0.33      0.33      0.33      4000

Hidden Layer = (50,50)
Accuracy score calculated on Test Data :  0.2955

Confusion matrix :  [[273  85 100  95  63 138]
 [132 204  91  77  80 114]
 [135  82 190  91  61 117]
 [104  73  74 165  76 104]
 [ 96  63  81  79  96 157]
 [118  79  90  84  79 254]]

Classwise  accuracy  :    [0.36206897  0.29226361  0.28106509
0.27684564 0.16783217 0.36079545]

Classification report :
              precision    recall  f1-score   support

           0       0.32      0.36      0.34       754
           1       0.35      0.29      0.32       698
           2       0.30      0.28      0.29       676
           3       0.28      0.28      0.28       596
           4       0.21      0.17      0.19       572
           5       0.29      0.36      0.32       704

    accuracy                           0.30      4000
   macro avg       0.29      0.29      0.29      4000
weighted avg       0.29      0.30      0.29      4000

Hidden Layer = (25,25)
Accuracy score calculated on Test Data :  0.299

Confusion matrix :  [[254 124 166  50 151]
 [140 151 243  78 168]
 [132 102 309  82 182]
 [117  71 199 134 330]
 [114  91 193  71 348]]
```

```
Classwise  accuracy  :    [0.3409396    0.19358974  0.38289963
0.15746181 0.42594859]

Classification report :
              precision    recall  f1-score   support

           0       0.34      0.34      0.34       745
           1       0.28      0.19      0.23       780
           2       0.28      0.38      0.32       807
           3       0.32      0.16      0.21       851
           4       0.30      0.43      0.35       817

    accuracy                           0.30      4000
   macro avg       0.30      0.30      0.29      4000
Weighted avg       0.30      0.30      0.29      4000


Hidden Layer = (50,50,25)
Accuracy score calculated on Test Data :  0.29075

Confusion matrix :  [[275  89 112  85 106  87]
 [131 200  98  82  98  89]
 [138 103 187  76  92  80]
 [122  78  85 153  74  84]
 [113  70  74  73 170  72]
 [143  93  90  83 117 178]]

Classwise  accuracy  :    [0.36472149  0.28653295  0.27662722
0.25671141 0.2972028  0.25284091]

Classification report :
              precision    recall  f1-score   support

           0       0.30      0.36      0.33       754
           1       0.32      0.29      0.30       698
           2       0.29      0.28      0.28       676
           3       0.28      0.26      0.27       596
           4       0.26      0.30      0.28       572
           5       0.30      0.25      0.28       704

    accuracy                           0.29      4000
   macro avg       0.29      0.29      0.29      4000
weighted avg       0.29      0.29      0.29      4000


Hidden Layer = (50,50,25,25)
Accuracy score calculated on Test Data :  0.29025
```

```
Confusion matrix :  [[272  86 113  99  93  91]
 [146 180 106  86  90  90]
 [140  72 195  92  82  95]
 [110  68  92 170  82  74]
 [116  61  81  84 154  76]
 [136  94  94 104  86 190]]

Classwise  accuracy  :    [0.36074271  0.25787966  0.28846154
0.2852349  0.26923077 0.26988636]

Classification report :
              precision    recall  f1-score   support

           0       0.30      0.36      0.32       754
           1       0.32      0.26      0.29       698
           2       0.29      0.29      0.29       676
           3       0.27      0.29      0.28       596
           4       0.26      0.27      0.27       572
           5       0.31      0.27      0.29       704

    accuracy                           0.29      4000
   macro avg       0.29      0.29      0.29      4000
weighted avg       0.29      0.29      0.29      4000
```

Accuracy vs Hidden Layers:


Accuracy vs Hidden Layers and Neurons
0.295
0.290
0.285
0.280
0.275
0.270
0.265
Accuracy
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Hidden Layer Count and Neurons


Accuracy vs Hidden Layers and Neurons
0.35
0.34
0.33
0.32
0.31
0.30
Accuracy
2
4
6
8
10
Hidden Layer Count and Neurons

Analysis:

1. As we can see that the more hidden layers over fits the data and hence the accuracy of the data increases for a point and then falls down. The increase in the number of neurons also reduces the data modelling.
2. Less number of hidden layers and neurons underfits the data and hence this is not the correct number of layers.
3. The labelling technique does not play many roles in varying test sizes as both the techniques give almost the same results only.
4. Now, the best number of hidden layers is 3 which is to be (50,50,25).

3. ANN Model on Feature Selection Data on both thee techniques:

So, the top 10 features extracted from the data were tried with the best-hidden layers neurons and the test size and the output as accuracy was predicted using that.

a. Output on Technique 1:

```
Accuracy score calculated on Test Data :  0.292875

Confusion matrix :  [[506 330 233 183 240]
 [285 564 248 184 266]
 [277 403 427 256 263]
 [307 358 331 320 382]
 [295 359 234 223 526]]

Classwise  accuracy  :    [0.33914209  0.3645766    0.26260763
0.18845701 0.32131949]

Classification report :
              precision    recall  f1-score   support

           0       0.30      0.34      0.32      1492
           1       0.28      0.36      0.32      1547
           2       0.29      0.26      0.28      1626
           3       0.27      0.19      0.22      1698
           4       0.31      0.32      0.32      1637

    accuracy                           0.29      8000
   macro avg       0.29      0.30      0.29      8000
weighted avg       0.29      0.29      0.29      8000
```

b. Output on Technique 2:

```
Accuracy score calculated on Test Data :  0.31216666666666665

Confusion matrix :  [[394 130 230 196 162]
 [256 194 332 208 187]
```

```
 [228 138 425 222 188]
 [224 129 272 446 210]
 [219 145 244 207 414]]

Classwise  accuracy  :     [0.35431655  0.16482583  0.35387177
0.3481655  0.33685924]

Classification report :
              precision    recall  f1-score   support

           0       0.30      0.35      0.32      1112
           1       0.26      0.16      0.20      1177
           2       0.28      0.35      0.31      1201
           3       0.35      0.35      0.35      1281
           4       0.36      0.34      0.35      1229

    accuracy                           0.31      6000
   macro avg       0.31      0.31      0.31      6000
weighted avg       0.31      0.31      0.31      6000
```

Analysis:

1. Feature Selection donot increase the accuracy as expected which shows that the features extracted are playing an important role but the data is very abstract.
2. Feature Selection bring out almost the same accuracies.