In [139…

```python
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
```

In [140…

```python
dataset1=pd.read_csv('C:\\Users\\pragati\\Desktop\\datamining\dataset1.csv')
```

In [141…

```python
dataset1
```

Out[141…

	sl_no	University_iD	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t
0	1	NaN	M	67.00	Others	NaN	Others	Commerce	58.00	Sci&Tech
1	2	12346.0	M	79.33	Central	78.33	Others	Science	77.48	sci&Tech
2	3	NaN	NaN	65.00	NaN	NaN	Central	NaN	NaN	Comm&Mgm
3	4	12348.0	M	56.00	Central	52	Central	Science	52.00	Sci&Tech
4	5	12349.0	M	85.80	Central	73.6	Central	Commerce	73.30	Comm&Mgm
...	...	...	...	...	...	...	...	...	...	..
147	148	12492.0	M	70.00	Central	74	Central	Commerce	65.00	Comm&Mgm
148	149	12493.0	F	77.00	Central	86	Central	Arts	56.00	Others
149	150	12494.0	M	44.00	Central	58	Central	Arts	55.00	Comm&Mgm
150	151	12495.0	M	71.00	Central	58.66	Central	Science	58.00	Sci&Tech
151	152	12496.0	M	65.00	Central	65	Central	Commerce	75.00	Comm&Mgm

152 rows × 17 columns

In [142…

```python
dataset2=pd.read_csv('C:\\Users\\pragati\\Desktop\\datamining\dataset2.csv')
```

In [143…

```python
dataset2
```

Out[143…

	s.n	University_iD	Gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t
0	1	12494.0	K	44.0	Central	58.00	Central	Arts	55.0	Comm&Mgmt
1	2	12495.0	M	71.0	Central	58.66	Central	Science	58.0	Sci&Tech
2	3	12496.0	M	65.0	Central	65.00	Central	Commerce	75.0	Comm&Mgmt
3	4	12497.0	F	75.4	Others	60.50	Central	Science	84.0	Sci&Tech
4	5	12498.0	M	49.0	Others	59.00	Others	Science	50.0	Sci&Tech
...	...	...	...	...	...	...	...	...	...	...
61	62	12555.0	M	80.6	Others	82.00	Others	Commerce	77.6	Comm&Mgmt
62	63	12556.0	M	58.0	Others	60.00	Others	Science	72.0	Sci&Tech
63	64	12557.0	M	67.0	Others	67.00	Others	Commerce	73.0	Comm&Mgmt

	s.n	University_iD	Gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t
64	65	12558.0	F	74.0	Others	66.00	Others	Commerce	58.0	Comm&Mgmt
65	66	12559.0	M	62.0	Central	58.00	Others	Science	53.0	Comm&Mgmt

66 rows × 17 columns

In [144…

```
dataset=[dataset1,dataset2]
```

In [145…

```
finaldataset=pd.concat(dataset)
```

In [146…

```
finaldataset
```

Out[146…

	sl_no	University_iD	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t
0	1.0	NaN	M	67.00	Others	NaN	Others	Commerce	58.00	Sci&Tech
1	2.0	12346.0	M	79.33	Central	78.33	Others	Science	77.48	sci&Tech
2	3.0	NaN	NaN	65.00	NaN	NaN	Central	NaN	NaN	Comm&Mgmt
3	4.0	12348.0	M	56.00	Central	52	Central	Science	52.00	Sci&Tech
4	5.0	12349.0	M	85.80	Central	73.6	Central	Commerce	73.30	Comm&Mgmt
...	...	...	...	...	...	...	...	...	...	...
61	NaN	12555.0	NaN	80.60	Others	82	Others	Commerce	77.60	Comm&Mgmt
62	NaN	12556.0	NaN	58.00	Others	60	Others	Science	72.00	Sci&Tech
63	NaN	12557.0	NaN	67.00	Others	67	Others	Commerce	73.00	Comm&Mgmt
64	NaN	12558.0	NaN	74.00	Others	66	Others	Commerce	58.00	Comm&Mgmt
65	NaN	12559.0	NaN	62.00	Central	58	Others	Science	53.00	Comm&Mgmt

218 rows × 19 columns

In [147…

```
finaldataset['gender'].unique()
```

Out[147…

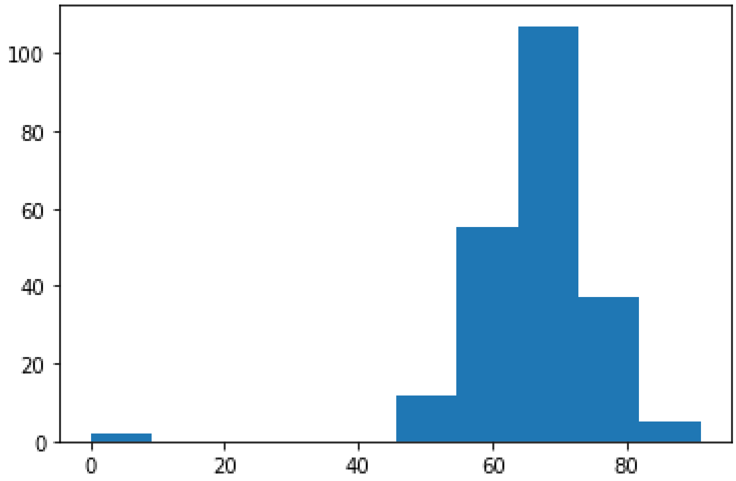
```
array(['M', nan, 'F'], dtype=object)
```

In [148…

```
plt.hist(New_df['degree_p'])
#Skewed to the Left.

#Negatively Skewed Data
```

Out[148…

```
(array([  2.,   0.,   0.,   0.,   0.,  12.,  55., 107.,  37.,   5.]),
 array([ 0. ,  9.1, 18.2, 27.3, 36.4, 45.5, 54.6, 63.7, 72.8, 81.9, 91. ]),
 <BarContainer object of 10 artists>)
```



In [149…

```
finaldataset.head()
```

Out[149…

	sl_no	University_iD	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t
0	1.0	NaN	M	67.00	Others	NaN	Others	Commerce	58.00	Sci&Tech
1	2.0	12346.0	M	79.33	Central	78.33	Others	Science	77.48	sci&Tech
2	3.0	NaN	NaN	65.00	NaN	NaN	Central	NaN	NaN	Comm&Mgmt
3	4.0	12348.0	M	56.00	Central	52	Central	Science	52.00	Sci&Tech
4	5.0	12349.0	M	85.80	Central	73.6	Central	Commerce	73.30	Comm&Mgmt

In [150…

```
finaldataset['ssc_p'].unique()
```

Out[150…

```
array([67.  , 79.33, 65.  , 56.  , 85.8 , 55.  , 46.  , 82.  ,   nan,
       58.  , 69.6 , 47.  , 77.  , 62.  , 63.  , 60.  , 79.  , 69.8 ,
       77.4 , 76.5 , 52.58, 71.  , 76.76, 64.  , 61.  , 87.  , 69.  ,
       51.  , 73.  , 81.  , 78.  , 74.  , 49.  , 76.  , 70.89, 50.  ,
       75.2 , 54.4 , 40.89, 80.  , 60.4 , 68.  , 52.6 , 84.2 , 86.5 ,
       54.  , 83.  , 80.92, 69.7 , 75.  , 84.86, 64.6 , 56.6 , 59.  ,
       66.5 , 84.  , 81.7 , 70.  , 83.84, 59.6 , 66.  , 85.  , 52.  ,
       60.23, 70.5 , 45.  , 61.08, 69.5 , 73.96, 68.2 , 60.8 , 72.  ,
       80.4 , 76.7 , 74.9 , 77.44, 77.67, 89.4 , 44.  , 75.4 , 53.  ,
       51.57, 55.6 , 74.2 , 67.16, 63.3 , 67.9 , 48.  , 59.96, 63.4 ,
       73.24, 77.8 , 56.28, 88.  , 78.5 , 61.8 , 65.2 , 83.96, 54.2 ,
       55.68, 41.  , 83.33, 43.  , 80.6 ])
```

In [151…

```
finaldataset['ssc_b'].unique()
```

Out[151…

```
array(['Others', 'Central', nan, 'central'], dtype=object)
```

In [152…

```
New_df=finaldataset.fillna({'University_iD':0,'hsc_p':0,'ssc_p':0,'salary':0,'degree
                            'Number of years experience':0,'test_p':0,'mba_p':0,'wor
```

In [215…

```
#New_df['Number of years experience'].unique()
#New_df['workex'].unique()
#New_df['gender'].unique()
#New_df.ssc_p.head()
#New_df['ssc_p'].unique()
```

In [155…

```
New_df['gender'].unique()
```

```
Out[155… array(['M', 'Other', 'F'], dtype=object)
```

```
In [158… New_df.columns
```

```
Out[158… Index(['sl_no', 'University_iD', 'gender', 'ssc_p', 'ssc_b', 'hsc_p', 'hsc_b',
                'hsc_s', 'degree_p', 'degree_t', 'workex', 'Number of years experience',
                'test_p', 'specialisation', 'mba_p', 'status', 'salary', 's.n',
                'Gender'],
               dtype='object')
```

```
In [159… New_df['agg_school_pct']=New_df.hsc_p=New_df.ssc_p
         sum=New_df['agg_school_pct']
         sum
```

```
Out[159… 0      67.00
         1      79.33
         2      65.00
         3      56.00
         4      85.80
                ...
         61     80.60
         62     58.00
         63     67.00
         64     74.00
         65     62.00
         Name: agg_school_pct, Length: 218, dtype: float64
```

```
In [160… New_df.degree_p
```

```
Out[160… 0      58.00
         1      77.48
         2       0.00
         3      52.00
         4      73.30
                ...
         61     77.60
         62     72.00
         63     73.00
         64     58.00
         65     53.00
         Name: degree_p, Length: 218, dtype: float64
```

```
In [161… #finaldataset['ssc_p'].fillna(method='ffill')
         #finaldataset['hsc_p'].fillna(method='ffill')
         #finaldataset['test_p'].fillna(method='ffill')
         missing_unknown='Nan'+'na'+'NAN'+'NaN'+'nan'+'na '
         New_df['degree_p'].replace('missing_unknown','sum')
         New_df['University_iD'].replace('missing_unknown','sum')
         New_df['degree_p'].unique()
```

```
Out[161… array([58.  , 77.48,  0.  , 52.  , 73.3 , 67.25, 79.  , 66.  , 61.  ,
                60.  , 78.3 , 65.  , 59.  , 50.  , 69.  , 65.6 , 64.  , 70.  ,
                85.  , 72.23, 64.74, 78.86, 50.2 , 67.5 , 73.  , 66.4 , 81.  ,
                72.  , 57.  , 80.  , 68.  , 68.4 , 56.2 , 53.  , 61.4 , 74.  ,
                72.11, 66.89, 67.4 , 75.  , 67.  , 72.7 , 62.  , 71.  , 78.  ,
                71.72, 70.2 , 77.5 , 71.93, 64.5 , 77.2 , 82.  , 50.8 , 54.  ,
                76.  , 63.  , 83.  , 66.6 , 64.6 , 69.6 , 69.3 , 64.33, 75.5 ,
                77.72, 77.  , 69.5 , 73.43, 70.67, 71.25, 56.  , 55.  , 84.  ,
                59.9 , 60.9 , 57.5 , 77.25, 63.35, 61.26, 64.27, 64.2 , 62.8 ,
                64.21, 59.79, 54.38, 69.2 , 64.8 , 56.3 , 91.  , 56.87, 77.6 ])
```

```
In [216… New_df.head(2)
```

Out[216…

	sl_no	University_iD	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t	...	N e

	sl_no	University_iD	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t	...	N e
0	1.0	0.0	M	67.00	Others	67.00	Others	Commerce	58.00	Sci&Tech	...	
1	2.0	12346.0	M	79.33	Central	79.33	Others	Science	77.48	sci&Tech	...	

2 rows × 21 columns

```
In [163…
#New_df1=New_df.replace({'University_iD'=='0',New_df.University_iD.((ffill+bfill)/2)
#New_df1
#New_df1.University_iD.drop('University_iD'=='na')
#New_df1.University_iD.unique()
```

```
In [164…
#New_df1['University_iD] = (New_df1.University_iD.ffill() + New_df1.University_iD.bf
#New_df1.University_iD
```

```
In [165…
#Transpose swap rows and columns
```

```
In [217…
New_df_t=New_df.T
```

```
In [218…
New_df_t
```

Out[218…

	0	1	2	3	4	5	
sl_no	1	2	3	4	5	6	
University_iD	0	12346	0	12348	12349	12350	
gender	M	M	Other	M	M	M	
ssc_p	67	79.33	65	56	85.8	55	
ssc_b	Others	Central	NaN	Central	Central	Others	
hsc_p	67	79.33	65	56	85.8	55	
hsc_b	Others	Others	Central	Central	Central	Others	
hsc_s	Commerce	Science	NaN	Science	Commerce	Science	
degree_p	58	77.48	0	52	73.3	67.25	
degree_t	Sci&Tech	sci&Tech	Comm&Mgmt	Sci&Tech	Comm&Mgmt	Sci&Tech	Co
workex	0	Yes	0	No	No	Yes	
Number of years experience	0	2	0	0	0	1	
test_p	55	86.5	0	66	96.8	55	
specialisation	NaN	Mkt&Fin	NaN	Mkt&HR	Mkt&Fin	Mkt&Fin	
mba_p	58.8	66.28	0	59.43	55.5	51.58	
status	NaN	Placed	NaN	Not Placed	Placed	Not Placed	
salary	0	200000	0	0	425000	0	
s.n	NaN	NaN	NaN	NaN	NaN	NaN	
Gender	NaN	NaN	NaN	NaN	NaN	NaN	

	0	1	2	3	4	5	
agg_school_pct	67	79.33	65	56	85.8	55	
bins	AvrageGood	AvrageGood	AvrageGood	AvrageGood	AvrageGood	AvrageGood	/

21 rows × 218 columns

In [219…

```
a=New_df_t.drop_duplicates()
a.head(5)
```

Out[219…

	0	1	2	3	4	5	6	7	8	9	...	
sl_no	1	2	3	4	5	6	7	8	9	10	...	I
University_iD	0	12346	0	12348	12349	12350	12351	12352	12345	12354	...	12
gender	M	M	Other	M	M	M	F	M	M	M	...	O
ssc_p	67	79.33	65	56	85.8	55	46	82	0	58	...	
ssc_b	Others	Central	NaN	Central	Central	Others	Others	NaN	Central	Central	...	Ot

5 rows × 218 columns

In [220…

```
New_df.dropna(how="all")
```

Out[220…

	sl_no	University_iD	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t
0	1.0	0.0	M	67.00	Others	67.00	Others	Commerce	58.00	Sci&Tech
1	2.0	12346.0	M	79.33	Central	79.33	Others	Science	77.48	sci&Tech
2	3.0	0.0	Other	65.00	NaN	65.00	Central	NaN	0.00	Comm&Mgmt
3	4.0	12348.0	M	56.00	Central	56.00	Central	Science	52.00	Sci&Tech
4	5.0	12349.0	M	85.80	Central	85.80	Central	Commerce	73.30	Comm&Mgmt
...	...	...	...	...	...	...	...	...	...	...
61	NaN	12555.0	Other	80.60	Others	80.60	Others	Commerce	77.60	Comm&Mgmt
62	NaN	12556.0	Other	58.00	Others	58.00	Others	Science	72.00	Sci&Tech
63	NaN	12557.0	Other	67.00	Others	67.00	Others	Commerce	73.00	Comm&Mgmt
64	NaN	12558.0	Other	74.00	Others	74.00	Others	Commerce	58.00	Comm&Mgmt
65	NaN	12559.0	Other	62.00	Central	62.00	Others	Science	53.00	Comm&Mgmt

218 rows × 21 columns

In [221…

```
# Summarization
```

In [222…

```
New_df.degree_p.unique()
```

```
Out[222… array([58.  , 77.48,  0.  , 52.  , 73.3 , 67.25, 79.  , 66.  , 61.  ,
               60.  , 78.3 , 65.  , 59.  , 50.  , 69.  , 65.6 , 64.  , 70.  ,
               85.  , 72.23, 64.74, 78.86, 50.2 , 67.5 , 73.  , 66.4 , 81.  ,
               72.  , 57.  , 80.  , 68.  , 68.4 , 56.2 , 53.  , 61.4 , 74.  ,
               72.11, 66.89, 67.4 , 75.  , 67.  , 72.7 , 62.  , 71.  , 78.  ,
               71.72, 70.2 , 77.5 , 71.93, 64.5 , 77.2 , 82.  , 50.8 , 54.  ,
               76.  , 63.  , 83.  , 66.6 , 64.6 , 69.6 , 69.3 , 64.33, 75.5 ,
               77.72, 77.  , 69.5 , 73.43, 70.67, 71.25, 56.  , 55.  , 84.  ,
               59.9 , 60.9 , 57.5 , 77.25, 63.35, 61.26, 64.27, 64.2 , 62.8 ,
               64.21, 59.79, 54.38, 69.2 , 64.8 , 56.3 , 91.  , 56.87, 77.6 ])
```

```
In [224…  degree_p_mean_value=New_df['degree_p'].mean()
          degree_p_mean_value
```

```
Out[224…  65.62655963302753
```

```
In [225…  degree_p_median_value=New_df['degree_p'].median()
          degree_p_median_value
```

```
Out[225…  65.8
```

```
In [226…  degree_p_mode_value=New_df['degree_p'].mode()
          degree_p_mode_value
```

```
Out[226…  0    65.0
          dtype: float64
```

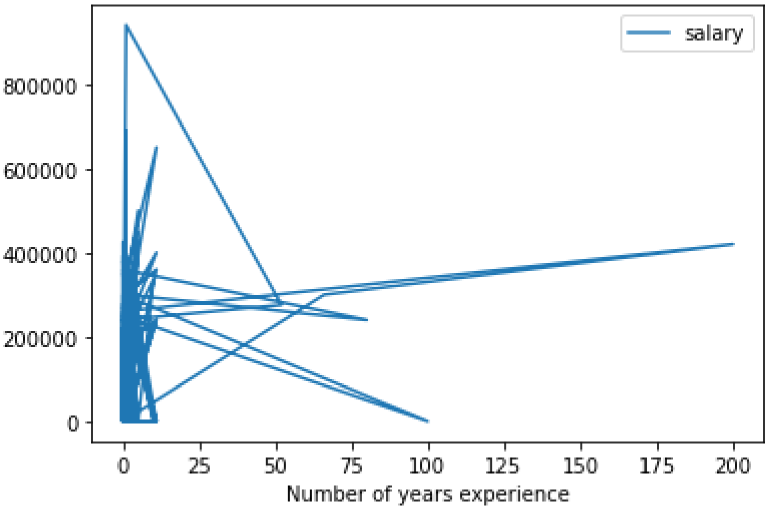
```
In [227…  s = pd.Series(['gender', 'ssc_b', 'hsc_b', 'hsc_s','degree_t', 'workex', 'specialisa
```

```
In [228…  s.str.upper()
```

```
Out[228…  0           GENDER
          1            SSC_B
          2            HSC_B
          3            HSC_S
          4         DEGREE_T
          5           WORKEX
          6    SPECIALISATION
          7           STATUS
          8              S.N
          9           GENDER
          dtype: object
```

```
In [229…  New_df.plot(x='Number of years experience',y='salary',kind='line')
```

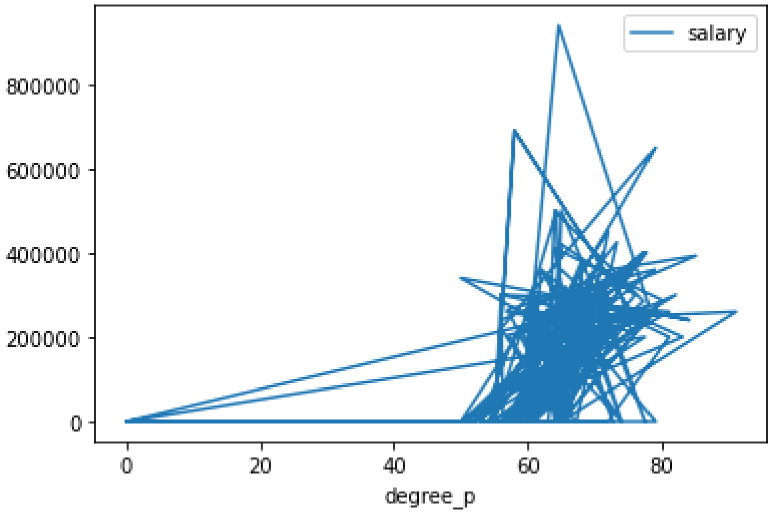
```
Out[229…  <AxesSubplot:xlabel='Number of years experience'>
```



```
New_df.plot(x='degree_p',y='salary',kind='line')
```

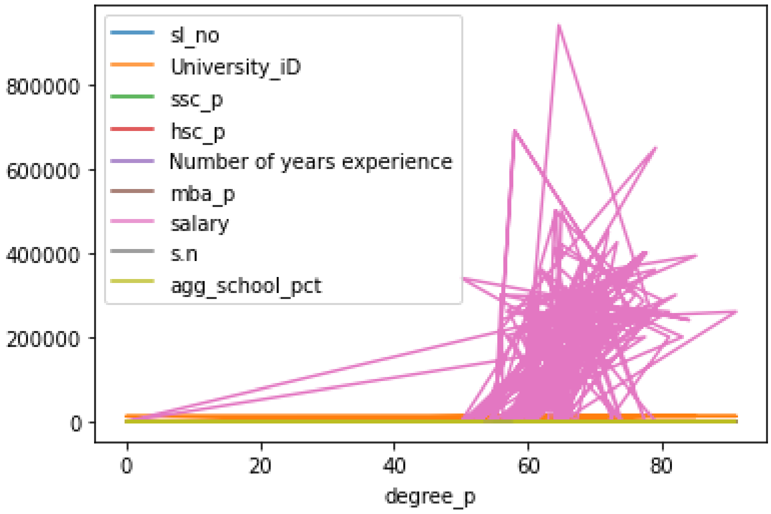
In [230…

Out[230…
```
<AxesSubplot:xlabel='degree_p'>
```



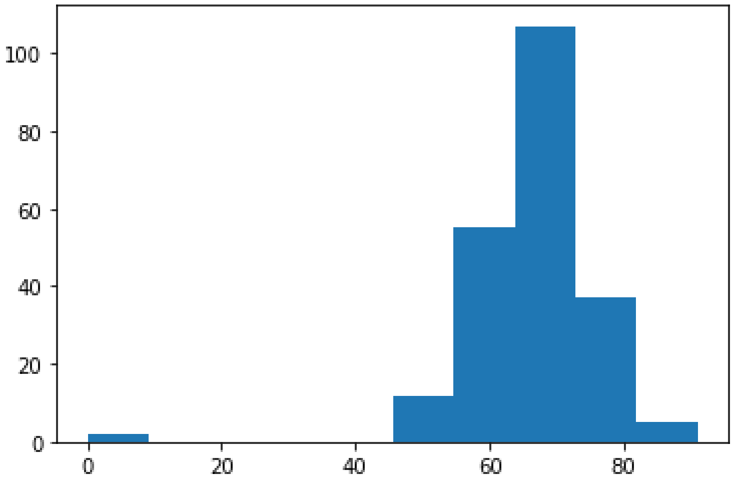
In [232…
```
New_df.plot('degree_p')
```

Out[232…
```
<AxesSubplot:xlabel='degree_p'>
```



In [233…
```
plt.hist(New_df['degree_p'])
```

Out[233…
```
(array([  2.,   0.,   0.,   0.,   0.,  12.,  55., 107.,  37.,   5.]),
 array([ 0. ,  9.1, 18.2, 27.3, 36.4, 45.5, 54.6, 63.7, 72.8, 81.9, 91. ]),
 <BarContainer object of 10 artists>)
```



In [234…
```
selected_columns = New_df[["degree_p", "salary"]]
```

```
New_df1 = selected_columns.copy()
print(New_df1)
```

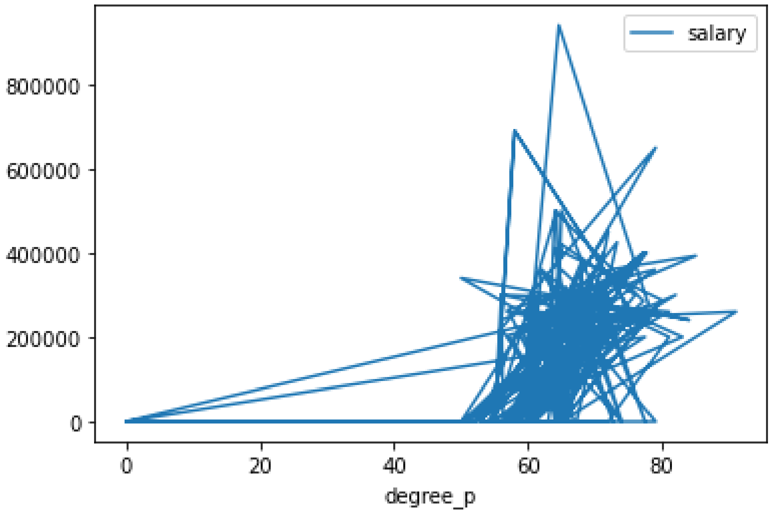
```
    degree_p    salary
0      58.00       0.0
1      77.48  200000.0
2       0.00       0.0
3      52.00       0.0
4      73.30  425000.0
..       ...       ...
61     77.60  400000.0
62     72.00  275000.0
63     73.00  295000.0
64     58.00  204000.0
65     53.00       0.0

[218 rows x 2 columns]
```

In [235…

```
New_df1.plot(x='degree_p',y='salary',kind='line')
```

Out[235…

```
<AxesSubplot:xlabel='degree_p'>
```



In [236…

```
salary_min_value = New_df['salary'].min()
salary_max_value = New_df['salary'].max()
print(salary_min_value)
print(salary_max_value)
```

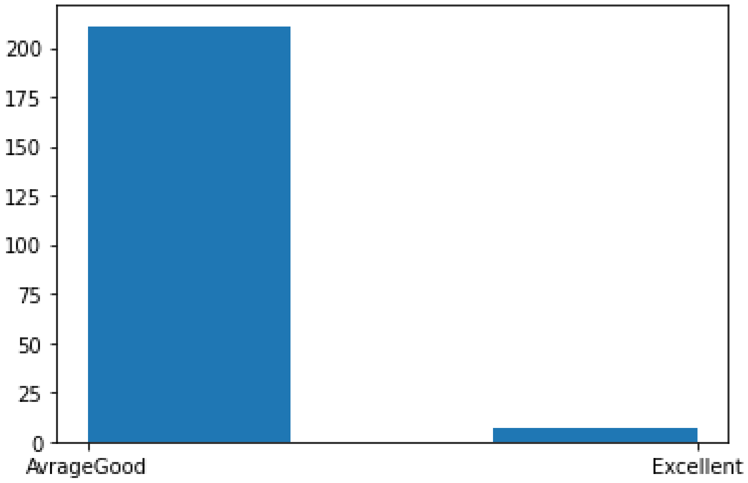
```
0.0
940000.0
```

In [237…

```
import numpy as np
bins = np.linspace(salary_min_value,salary_max_value,3)
bins
labels = ['Avrage' 'Good', 'Excellent']
New_df['bins'] = pd.cut(New_df['salary'], bins=bins, labels=labels, include_lowest=T
plt.hist(New_df['bins'], bins=3)
#New_df.salary.count()
```

Out[237…

```
(array([211.,   0.,   7.]),
 array([0.        , 0.33333333, 0.66666667, 1.        ]),
 <BarContainer object of 3 artists>)
```



In [238...

```
selected_columns = New_df[["ssc_p","degree_p","test_p","mba_p", "salary"]]
New_df2 = selected_columns.copy()
print(New_df2)
```

```
    ssc_p  degree_p test_p  mba_p    salary
0   67.00     58.00     55  58.80       0.0
1   79.33     77.48   86.5  66.28  200000.0
2   65.00      0.00      0   0.00       0.0
3   56.00     52.00     66  59.43       0.0
4   85.80     73.30   96.8  55.50  425000.0
..    ...       ...    ...    ...       ...
61  80.60     77.60     91  74.49  400000.0
62  58.00     72.00     74  53.62  275000.0
63  67.00     73.00     59  69.72  295000.0
64  74.00     58.00     70  60.23  204000.0
65  62.00     53.00     89  60.22       0.0

[218 rows x 5 columns]
```

In [239...

```
#selected_columns = df[["ssc_p","degree_p","test_p","mba_p", "salary"]]
#new_df = selected_columns.copy()
#print(new_df)

# copy the data
df_max_scaled = New_df2.copy()
print(df_max_scaled)
# apply normalization techniques
for selected_columns in df_max_scaled:
    df_max_scaled[selected_columns] = df_max_scaled[selected_columns] / (df_max_scal
# view normalized data
```

```
    ssc_p  degree_p test_p  mba_p    salary
0   67.00     58.00     55  58.80       0.0
1   79.33     77.48   86.5  66.28  200000.0
2   65.00      0.00      0   0.00       0.0
3   56.00     52.00     66  59.43       0.0
4   85.80     73.30   96.8  55.50  425000.0
..    ...       ...    ...    ...       ...
61  80.60     77.60     91  74.49  400000.0
62  58.00     72.00     74  53.62  275000.0
63  67.00     73.00     59  69.72  295000.0
64  74.00     58.00     70  60.23  204000.0
65  62.00     53.00     89  60.22       0.0

[218 rows x 5 columns]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-239-14b584ed71b4> in <module>
      8 # apply normalization techniques
      9 for selected_columns in df_max_scaled:
---> 10     df_max_scaled[selected_columns] = df_max_scaled[selected_columns] / (df_
max_scaled[selected_columns].abs().max())
     11 # view normalized data

C:\Anaconda3\lib\site-packages\pandas\core\generic.py in abs(self)
   9721         3    7    40   -50
   9722         """
-> 9723         return np.abs(self)
   9724 
   9725     def describe(

C:\Anaconda3\lib\site-packages\pandas\core\series.py in __array_ufunc__(self, ufunc,
method, *inputs, **kwargs)
    724 
    725         inputs = tuple(extract_array(x, extract_numpy=True) for x in inputs)
--> 726         result = getattr(ufunc, method)(*inputs, **kwargs)
    727 
    728         name = names[0] if len(set(names)) == 1 else None

TypeError: bad operand type for abs(): 'str'
```

In [240…

```
df_max_scaled
```

Out[240…

	ssc_p	degree_p	test_p	mba_p	salary
0	0.749441	0.637363	55	58.80	0.0
1	0.887360	0.851429	86.5	66.28	200000.0
2	0.727069	0.000000	0	0.00	0.0
3	0.626398	0.571429	66	59.43	0.0
4	0.959732	0.805495	96.8	55.50	425000.0
...	...	...	...	...	...
61	0.901566	0.852747	91	74.49	400000.0
62	0.648770	0.791209	74	53.62	275000.0
63	0.749441	0.802198	59	69.72	295000.0
64	0.827740	0.637363	70	60.23	204000.0
65	0.693512	0.582418	89	60.22	0.0

218 rows × 5 columns

In [241…

```
New_df_a=New_df.describe()
New_df_a
```

Out[241…

	sl_no	University_iD	ssc_p	hsc_p	degree_p	Number of years experience	mba_p	
count	152.000000	218.000000	218.000000	218.000000	218.000000	218.000000	218.000000	21
mean	76.500000	12280.238532	66.868028	66.868028	65.626560	4.041284	61.694633	19692
std	44.022721	1455.299563	11.778459	11.778459	9.785816	16.913800	8.305929	15951
min	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
25%	38.750000	12395.250000	60.272500	60.272500	61.000000	0.000000	57.922500	
50%	76.500000	12451.500000	67.000000	67.000000	65.800000	1.000000	61.885000	24000
75%	114.250000	12503.750000	75.350000	75.350000	72.000000	2.000000	66.187500	28375

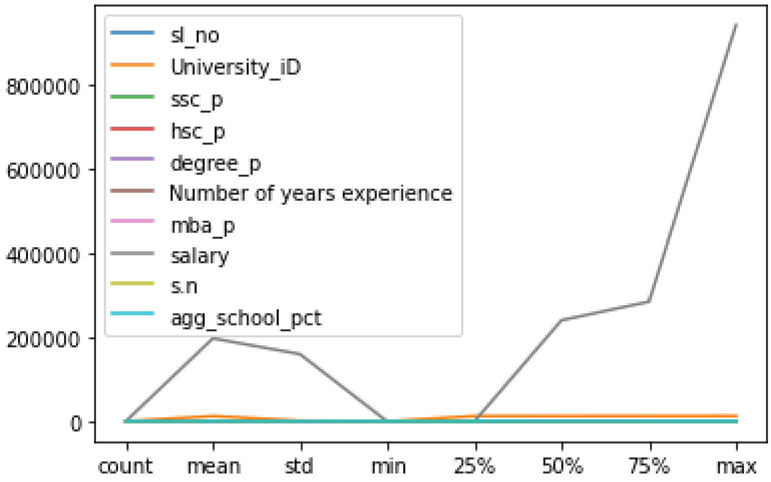
	sl_no	University_iD	ssc_p	hsc_p	degree_p	Number of years experience	mba_p	
max	152.000000	12559.000000	89.400000	89.400000	91.000000	200.000000	77.890000	94000

In [242…

```
New_df_a.plot()
```

Out[242…

```
<AxesSubplot:>
```



In [ ]:

```
New_plot1=pd.plotting.scatter_matrix(New_df)
```

In [245…

```
New_plot1.plot(figsize=(5,3));
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-245-fb8b69b0d3ce> in <module>
----> 1 New_plot1.plot(figsize=(5,3));

AttributeError: 'numpy.ndarray' object has no attribute 'plot'
```

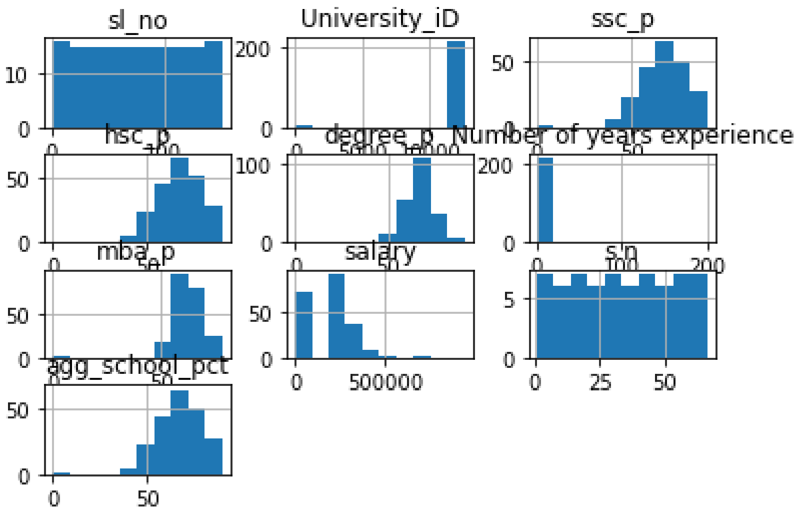
In [246…

```
New_df.plot(x="degree_p", y=["degree_p_mean_value", "degree_p_median_value", "degree
plt.show()
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-246-c6e2e80ea0b4> in <module>
----> 1 New_df.plot(x="degree_p", y=["degree_p_mean_value", "degree_p_median_value",
"degree_p_mode_value"])
      2 plt.show()

C:\Anaconda3\lib\site-packages\pandas\plotting\_core.py in __call__(self, *args, **k
wargs)
    933 
    934                 # don't overwrite
--> 935                 data = data[y].copy()
    936 
    937                 if isinstance(data, ABCSeries):

C:\Anaconda3\lib\site-packages\pandas\core\frame.py in __getitem__(self, key)
   2906             if is_iterator(key):
   2907                 key = list(key)
-> 2908             indexer = self.loc._get_listlike_indexer(key, axis=1, raise_miss
ing=True)[1]
   2909 
   2910         # take() does not accept boolean indexers

C:\Anaconda3\lib\site-packages\pandas\core\indexing.py in _get_listlike_indexer(sel
f, key, axis, raise_missing)
   1252                 keyarr, indexer, new_indexer = ax._reindex_non_unique(keyarr)
   1253 
-> 1254             self._validate_read_indexer(keyarr, indexer, axis, raise_missing=rai
se_missing)
   1255             return keyarr, indexer
   1256 

C:\Anaconda3\lib\site-packages\pandas\core\indexing.py in _validate_read_indexer(sel
f, key, indexer, axis, raise_missing)
   1296                 if missing == len(indexer):
   1297                     axis_name = self.obj._get_axis_name(axis)
-> 1298                     raise KeyError(f"None of [{key}] are in the [{axis_name}]")
   1299 
   1300                 # We (temporarily) allow for some missing keys with .loc, except
in

KeyError: "None of [Index(['degree_p_mean_value', 'degree_p_median_value', 'degree_p
_mode_value'], dtype='object')] are in the [columns]"
```

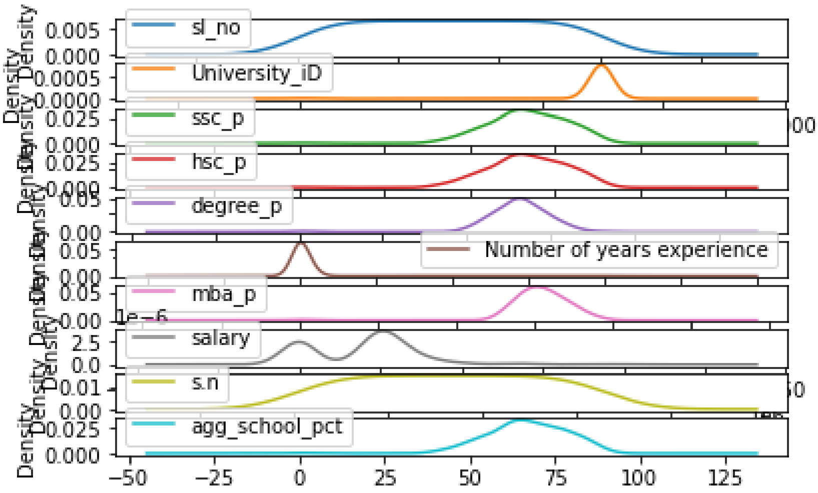
In [247…

```
New_df.hist()
plt.show()
```



In [248…

```
New_df.plot(kind='density',subplots=True,sharex=False)
plt.show()
```



In [249…

```
selected_plot_columns = New_df[["degree_p"]]
New_df3 = selected_plot_columns.copy()
print(New_df3)
```

```
    degree_p
0      58.00
```

```
1      77.48
2       0.00
3      52.00
4      73.30
..       ...
61     77.60
62     72.00
63     73.00
64     58.00
65     53.00

[218 rows x 1 columns]
```

In [250…

```
selected_plot_columns = New_df[["salary"]]
New_df4 = selected_plot_columns.copy()
print(New_df4)
```

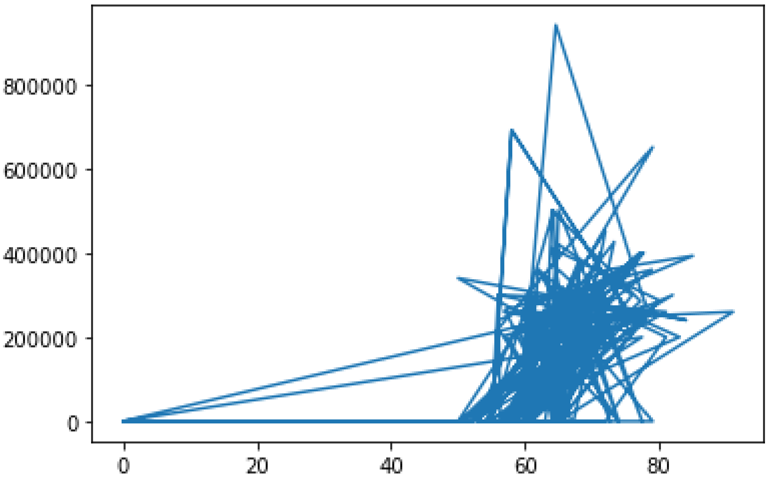
```
      salary
0        0.0
1   200000.0
2        0.0
3        0.0
4   425000.0
..       ...
61  400000.0
62  275000.0
63  295000.0
64  204000.0
65       0.0

[218 rows x 1 columns]
```

In [251…

```
plt.plot(New_df3, New_df4)
```

Out[251…

```
[<matplotlib.lines.Line2D at 0x28498352d60>]
```

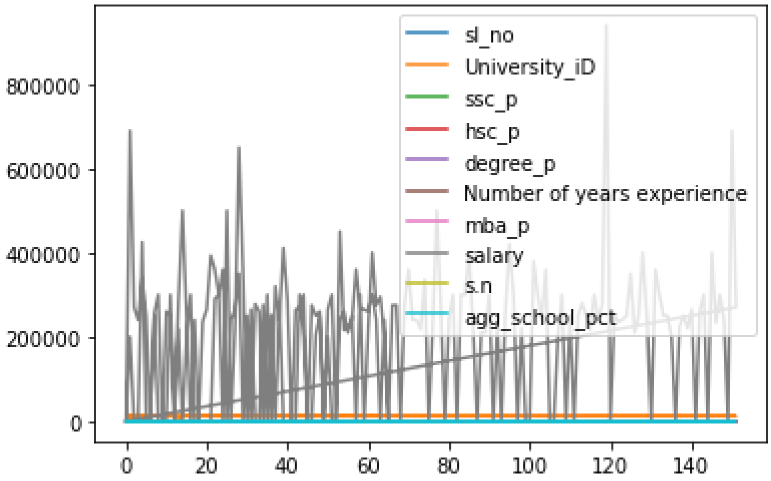


In [252…

```
New_df.plot()
```

Out[252…

```
<AxesSubplot:>
```



In [253…

```
selected_plot_columns = New_df[["salary","degree_p"]]
New_df5 = selected_plot_columns.copy()
print(New_df5)
```

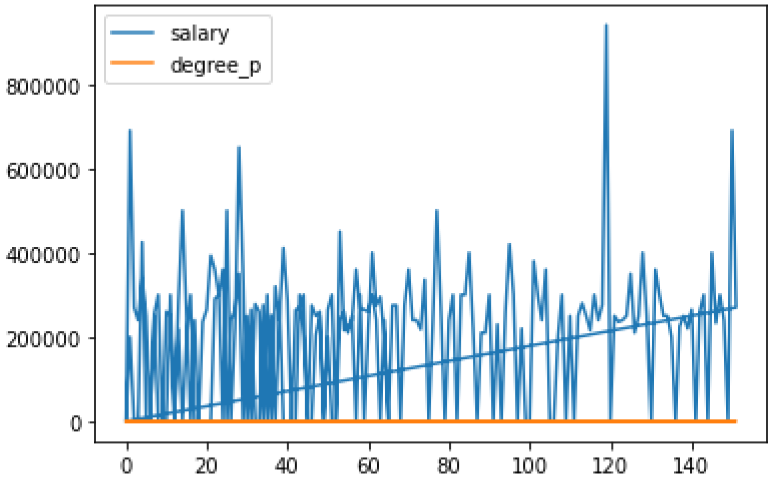
```
      salary  degree_p
0        0.0     58.00
1   200000.0     77.48
2        0.0      0.00
3        0.0     52.00
4   425000.0     73.30
..       ...       ...
61  400000.0     77.60
62  275000.0     72.00
63  295000.0     73.00
64  204000.0     58.00
65       0.0     53.00

[218 rows x 2 columns]
```

In [254…

```
New_df5.plot()
```

Out[254…

```
<AxesSubplot:>
```



In [255…

```
selected_plot_columns = New_df[["salary","Number of years experience"]]
New_df6 = selected_plot_columns.copy()
print(New_df6)
```

```
      salary  Number of years experience
0        0.0                         0.0
1   200000.0                         2.0
2        0.0                         0.0
3        0.0                         0.0
4   425000.0                         0.0
..       ...                         ...
```

```
61  400000.0                        11.0
62  275000.0                         1.0
63  295000.0                         1.0
64  204000.0                         1.0
65       0.0                         0.0

[218 rows x 2 columns]
```

In [256…

```
New_df.max()
```

Out[256…

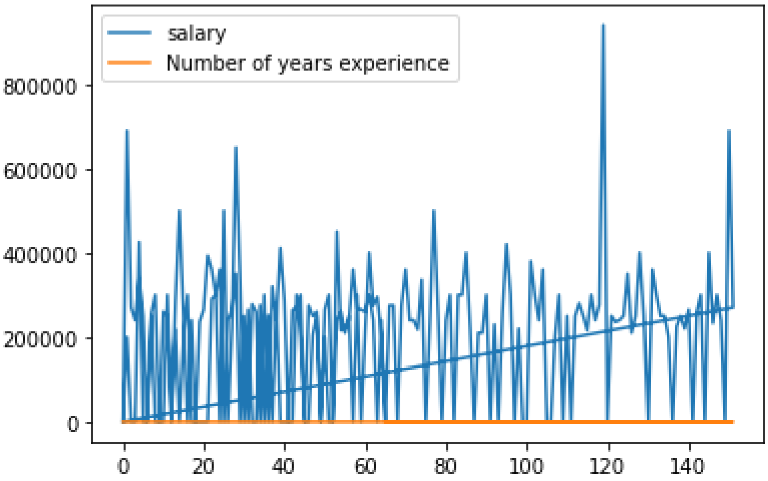
```
sl_no                                152
University_iD                      12559
gender                             Other
ssc_p                               89.4
hsc_p                               89.4
degree_p                              91
degree_t                        sci&Tech
Number of years experience           200
mba_p                              77.89
salary                            940000
s.n                                   66
agg_school_pct                      89.4
bins                           Excellent
dtype: object
```

In [257…

```
New_df6.plot()
```

Out[257…

```
<AxesSubplot:>
```

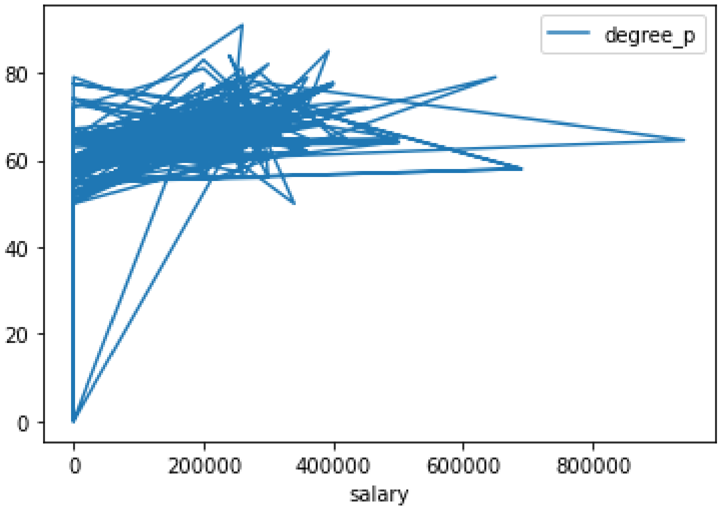


In [258…

```
New_df.plot(x='salary', y='degree_p')
```

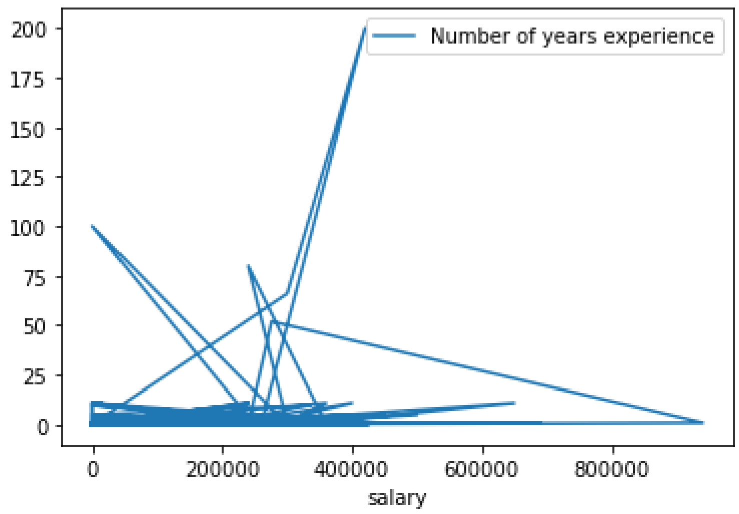
Out[258…

```
<AxesSubplot:xlabel='salary'>
```



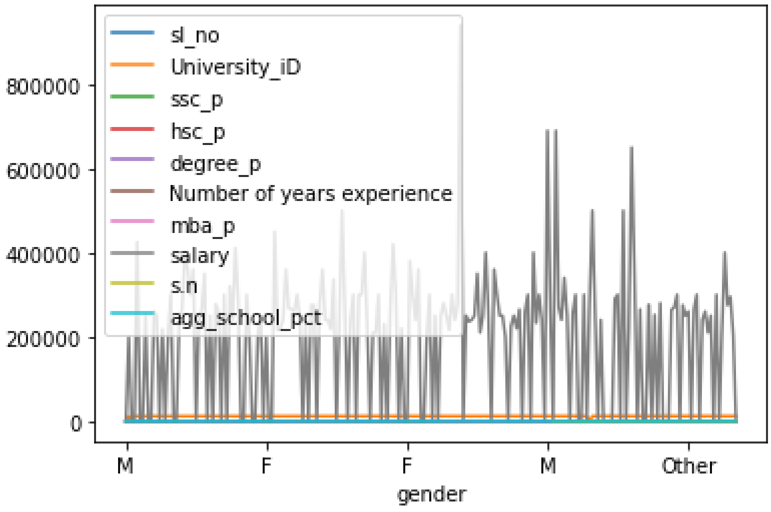
```
In [259… New_df.plot(x='salary', y='Number of years experience')
```

```
Out[259… <AxesSubplot:xlabel='salary'>
```



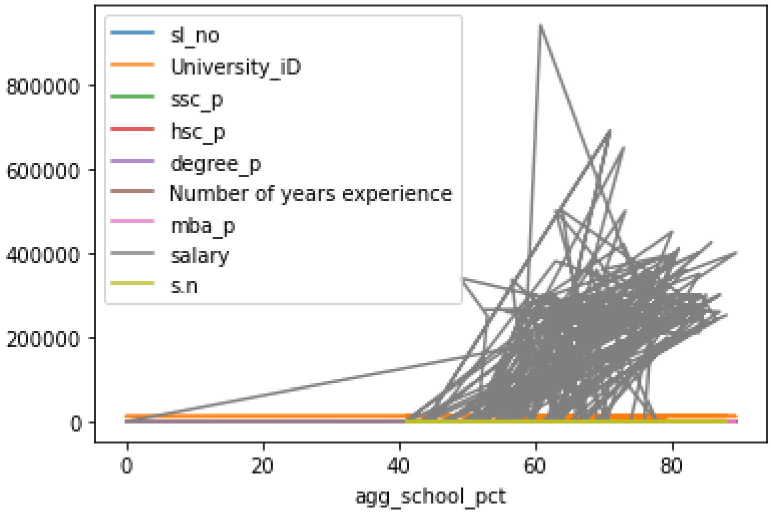
```
In [260… New_df.plot('gender')
```

```
Out[260… <AxesSubplot:xlabel='gender'>
```



```
In [261… New_df.plot('agg_school_pct')
```

```
Out[261… <AxesSubplot:xlabel='agg_school_pct'>
```



```
In [262… New_df.shape
```

```
#total rows and columns
```

Out[262…

```
(218, 21)
```

In [263…

```
New_df.columns
```

Out[263…

```
Index(['sl_no', 'University_iD', 'gender', 'ssc_p', 'ssc_b', 'hsc_p', 'hsc_b',
       'hsc_s', 'degree_p', 'degree_t', 'workex', 'Number of years experience',
       'test_p', 'specialisation', 'mba_p', 'status', 'salary', 's.n',
       'Gender', 'agg_school_pct', 'bins'],
      dtype='object')
```

In [264…

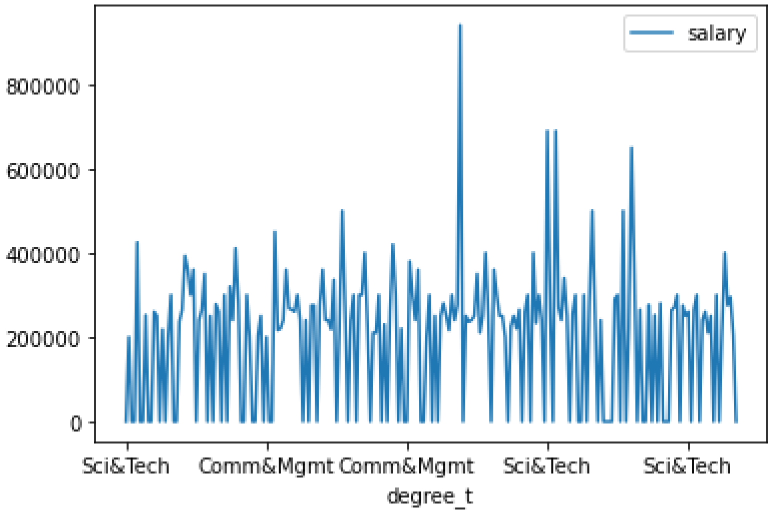
```
#hist_plot_1=New_df.hist('agg_school_pct',bin=10,density=True)
#plt.show()
```

In [265…

```
New_df.plot(x='degree_t', y='salary')
```

Out[265…

```
<AxesSubplot:xlabel='degree_t'>
```

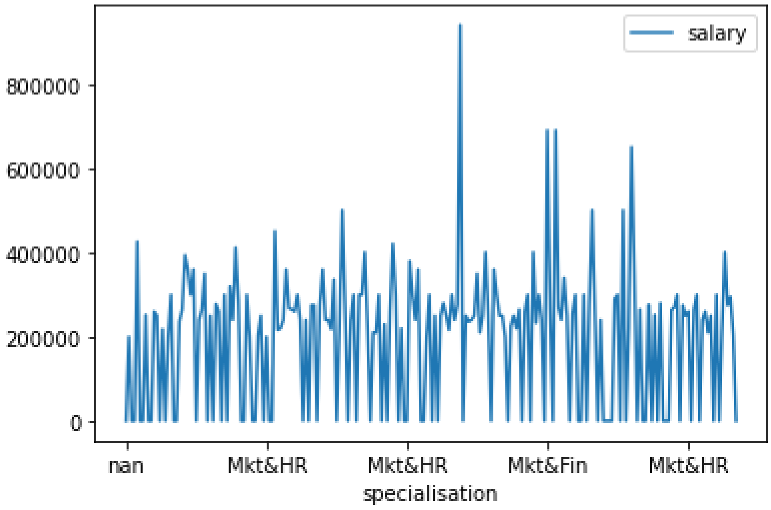


In [267…

```
New_df.plot(x='specialisation', y='salary')
```

Out[267…

```
<AxesSubplot:xlabel='specialisation'>
```



In [ ]:

```

```