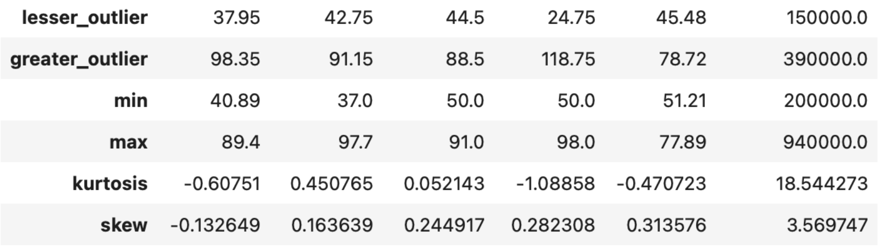
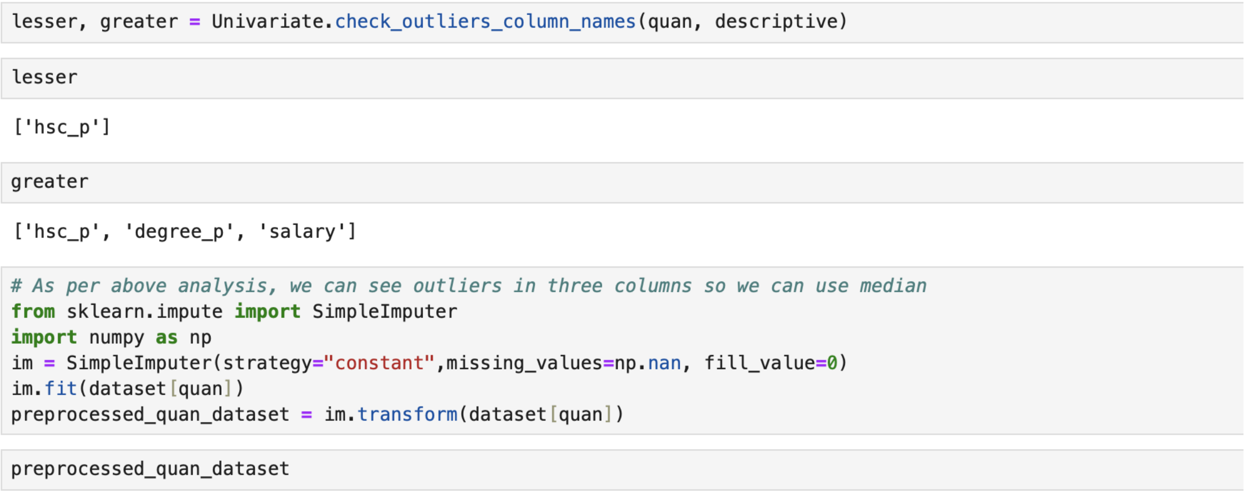
Inferential Analysis with Placement Dataset

1. Replace the NaN value with correct value and Justify why you have chosen the same



lesser_outlier	37.95	42.75	44.5	24.75	45.48	150000.0
greater_outlier	98.35	91.15	88.5	118.75	78.72	390000.0
min	40.89	37.0	50.0	50.0	51.21	200000.0
max	89.4	97.7	91.0	98.0	77.89	940000.0
kurtosis	-0.60751	0.450765	0.052143	-1.08858	-0.470723	18.544273
skew	-0.132649	0.163639	0.244917	0.282308	0.313576	3.569747



```
lesser, greater = Univariate.check_outliers_column_names(quan, descriptive)
```

```
lesser
```

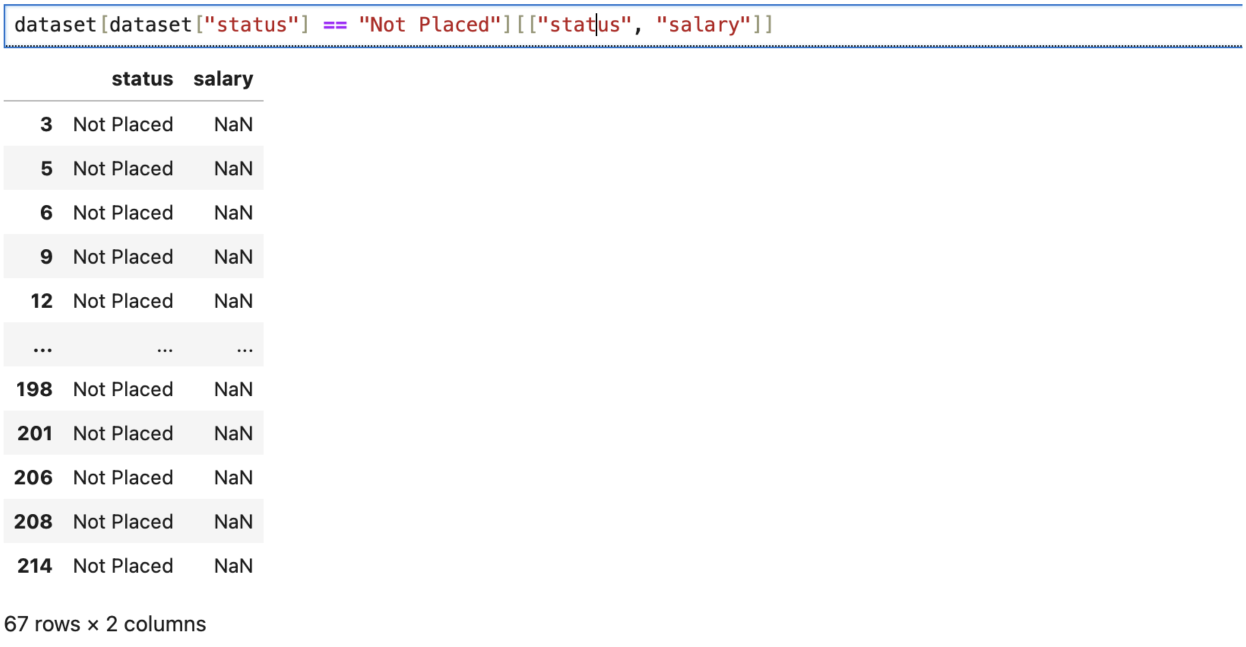
```
['hsc_p']
```

```
greater
```

```
['hsc_p', 'degree_p', 'salary']
```

```
# As per above analysis, we can see outliers in three columns so we can use median
from sklearn.impute import SimpleImputer
import numpy as np
im = SimpleImputer(strategy="constant",missing_values=np.nan, fill_value=0)
im.fit(dataset[quan])
preprocessed_quan_dataset = im.transform(dataset[quan])
```

```
preprocessed_quan_dataset
```



```
dataset[dataset["status"] == "Not Placed"][["status", "salary"]]
```

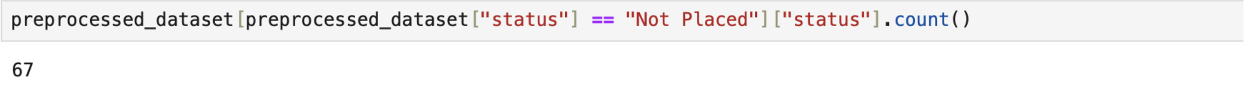
	status	salary
3	Not Placed	NaN
5	Not Placed	NaN
6	Not Placed	NaN
9	Not Placed	NaN
12	Not Placed	NaN
...	...	...
198	Not Placed	NaN
201	Not Placed	NaN
206	Not Placed	NaN
208	Not Placed	NaN
214	Not Placed	NaN

67 rows × 2 columns

I used zero to replace the NaN values

Justification:
As per the above result, we could see outliers in three columns and those are quantitative variables so we can't use mode and if we have outliers then we should not use mean as it considers outliers so best option is to use median as it will ignore the outliers but here, all the NaN values belongs to Salary because those students are not placed so we can't use median so the final option would be replace the NaN values with Zero so that it won't affect the original data

2. How many of them are not placed?

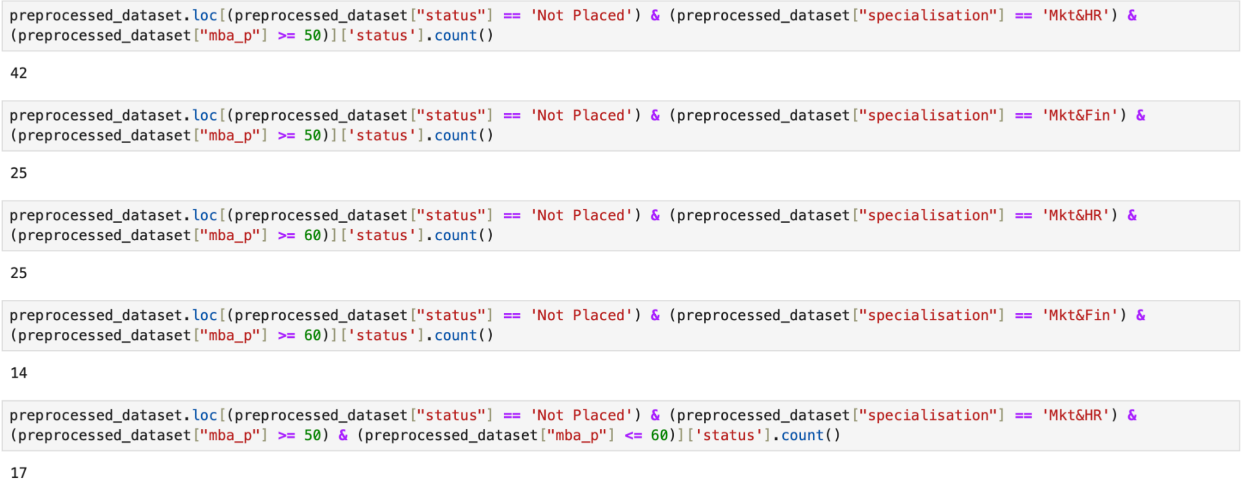


```
preprocessed_dataset[preprocessed_dataset["status"] == "Not Placed"]["status"].count()
```

```
67
```

As per the above result, there are 67 students who are not placed out of 215 students

3. Find the reason for non placement from the dataset



```
preprocessed_dataset.loc[(preprocessed_dataset["status"] == 'Not Placed') & (preprocessed_dataset["specialisation"] == 'Mkt&HR') &
(preprocessed_dataset["mba_p"] >= 50)]['status'].count()
```

```
42
```

```
preprocessed_dataset.loc[(preprocessed_dataset["status"] == 'Not Placed') & (preprocessed_dataset["specialisation"] == 'Mkt&Fin') &
(preprocessed_dataset["mba_p"] >= 50)]['status'].count()
```

```
25
```

```
preprocessed_dataset.loc[(preprocessed_dataset["status"] == 'Not Placed') & (preprocessed_dataset["specialisation"] == 'Mkt&HR') &
(preprocessed_dataset["mba_p"] >= 60)]['status'].count()
```

```
25
```

```
preprocessed_dataset.loc[(preprocessed_dataset["status"] == 'Not Placed') & (preprocessed_dataset["specialisation"] == 'Mkt&Fin') &
(preprocessed_dataset["mba_p"] >= 60)]['status'].count()
```

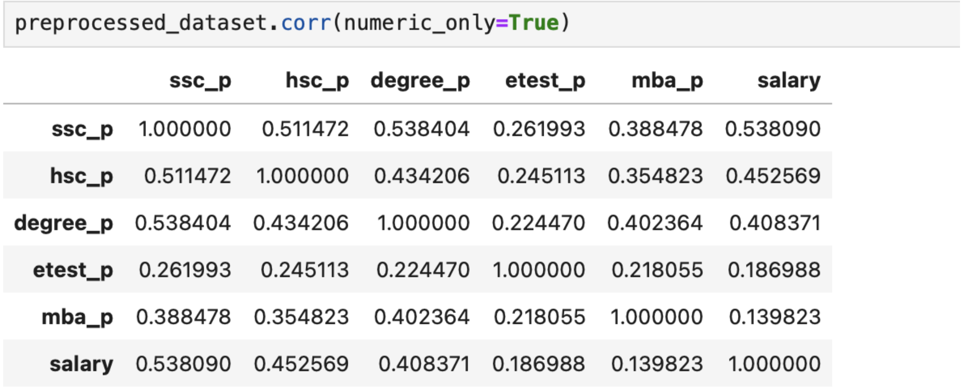
```
14
```

```
preprocessed_dataset.loc[(preprocessed_dataset["status"] == 'Not Placed') & (preprocessed_dataset["specialisation"] == 'Mkt&HR') &
(preprocessed_dataset["mba_p"] >= 50) & (preprocessed_dataset["mba_p"] <= 60)]['status'].count()
```

```
17
```

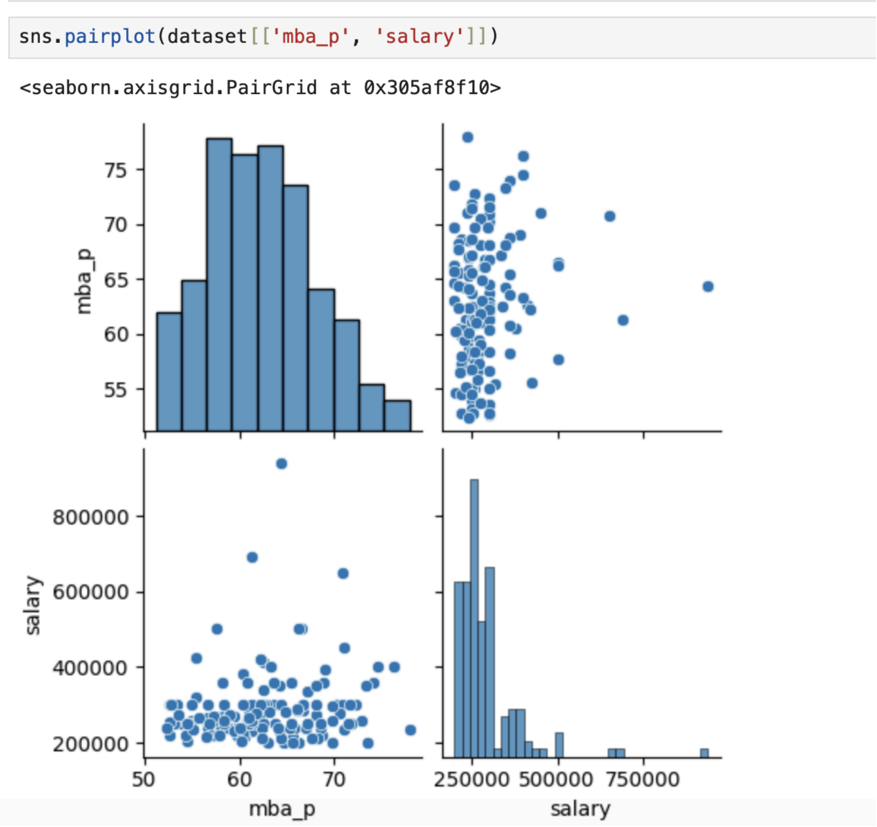
As per the above result, most of the students who are part of Marketing and HR are not placed because 17 students are getting marks between 50 and 60 and the correlation between mba_p and salary is positive but it is too low.

4. What is the relation between salary and mba_p?



```
preprocessed_dataset.corr(numeric_only=True)
```

	ssc_p	hsc_p	degree_p	etest_p	mba_p	salary
ssc_p	1.000000	0.511472	0.538404	0.261993	0.388478	0.538090
hsc_p	0.511472	1.000000	0.434206	0.245113	0.354823	0.452569
degree_p	0.538404	0.434206	1.000000	0.224470	0.402364	0.408371
etest_p	0.261993	0.245113	0.224470	1.000000	0.218055	0.186988
mba_p	0.388478	0.354823	0.402364	0.218055	1.000000	0.139823
salary	0.538090	0.452569	0.408371	0.186988	0.139823	1.000000

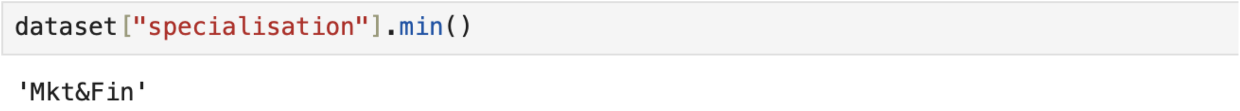


```
sns.pairplot(dataset[['mba_p', 'salary']])
```

```
<seaborn.axisgrid.PairGrid at 0x305af8f10>
```

As per the above result, Correlation between mba_p and salary is 0.13 which is less than 0.54 so it is low degree or weak positive correlation. both increase with 13%

5. **Which mba_p specialization is getting minimum salary?**

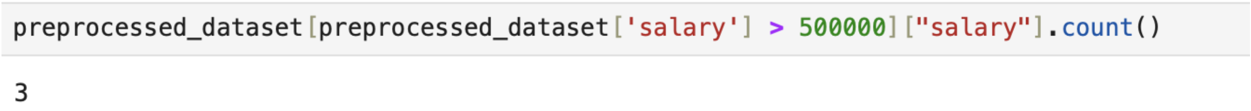


```
dataset["specialisation"].min()
```

```
'Mkt&Fin'
```

As per the above result, **marketing and Finance** is having minimum salary

6. **How many of them getting above 500000 salary**



```
preprocessed_dataset[preprocessed_dataset['salary'] > 500000]["salary"].count()
```

```
3
```

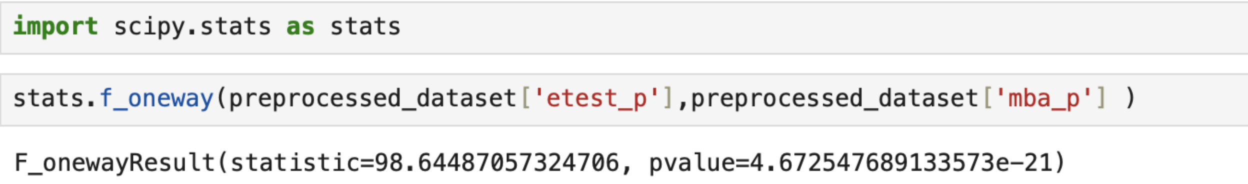
Only **three** of them are getting more than 500000 lakhs of salary

7. **Test the analysis of Variance between etest_p and mba_p at significant level 5% (Make decision using hypothesis testing)**

Null hypothesis (Ho) - There is no significant changes in etest_p and mba_p
Alternate Hypothesis (H1) - There is significant changes in etest_p and mba_p

Accept/Reject - if p < 0.0.5 reject null hypothesis or else accept null hypothesis and reject alternative hypothesis

This falls under one way classification of ANOVA



```
import scipy.stats as stats
```

```
stats.f_oneway(preprocessed_dataset['etest_p'],preprocessed_dataset['mba_p'] )
```

```
F_onewayResult(statistic=98.64487057324706, pvalue=4.672547689133573e-21)
```

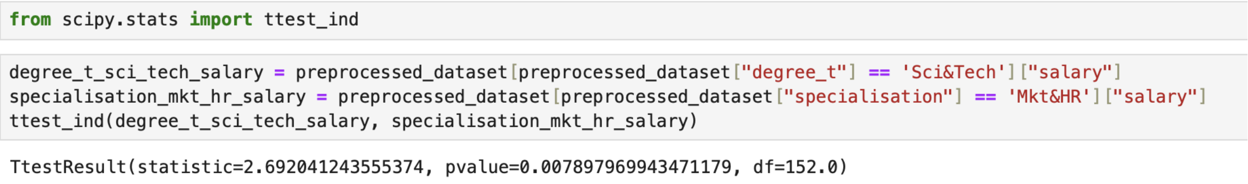
Here the p is 4.6 > 0.05 so we can accept null hypothesis (Ho)

8. **Test the similarity between degree_t(sci&tech) and specialisation (Mkt&HR) with respect to salary at significant level 5% (make decision using hypothesis testing)**

Null hypothesis (Ho) - There is no significant changes degree_t(sci&tech) and specialisation (Mkt&HR) with respect to salary
Alternate Hypothesis (H1) - There is significant changes degree_t(sci&tech) and specialisation (Mkt&HR) with respect to salary

Accept/Reject - if p < 0.0.5 reject null hypothesis or else accept null hypothesis and reject alternative hypothesis

This falls under unpaired-t test



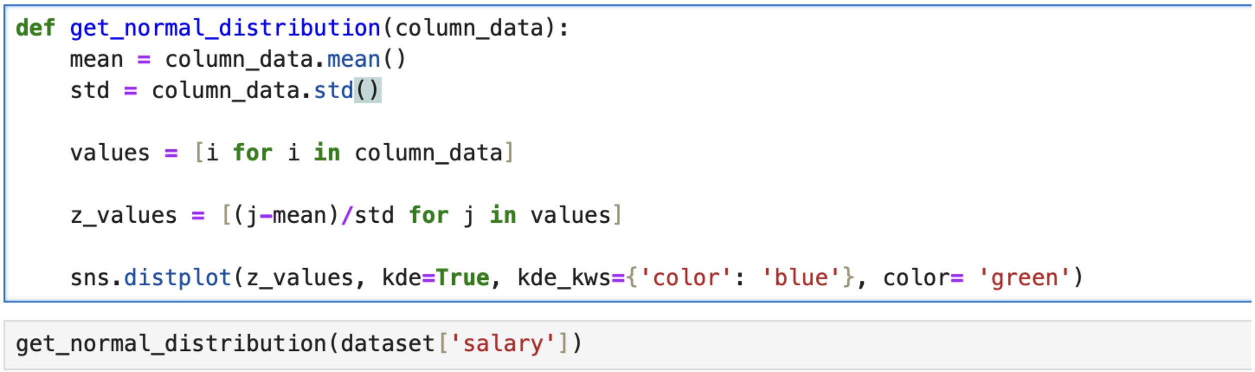
```
from scipy.stats import ttest_ind
```

```
degree_t_sci_tech_salary = preprocessed_dataset[preprocessed_dataset["degree_t"] == 'Sci&Tech']["salary"]
specialisation_mkt_hr_salary = preprocessed_dataset[preprocessed_dataset["specialisation"] == 'Mkt&HR']["salary"]
ttest_ind(degree_t_sci_tech_salary, specialisation_mkt_hr_salary)
```

```
TtestResult(statistic=2.692041243555374, pvalue=0.007897969943471179, df=152.0)
```

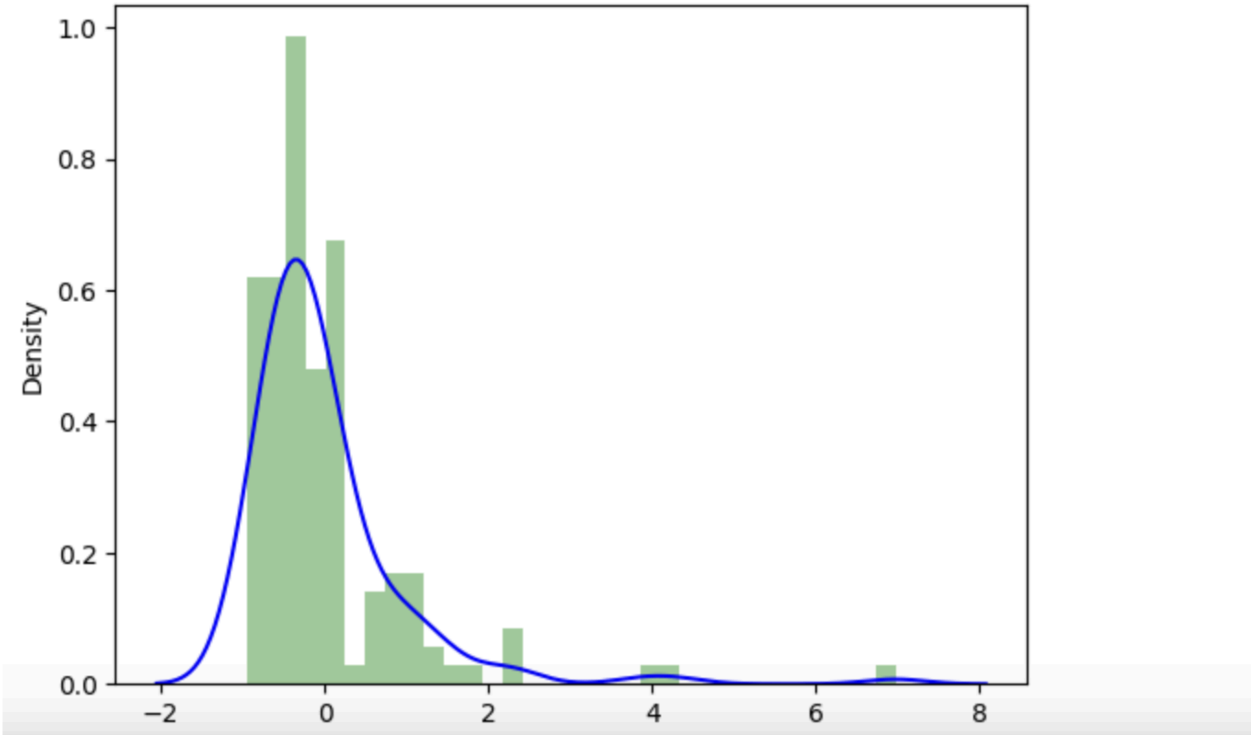
Here the p is 0.007< 0.05 so we can reject null hypothesis (Ho) and accept (H1) alternative hypothesis

9. **Convert normal distribution to standard normal distribution for salary column**



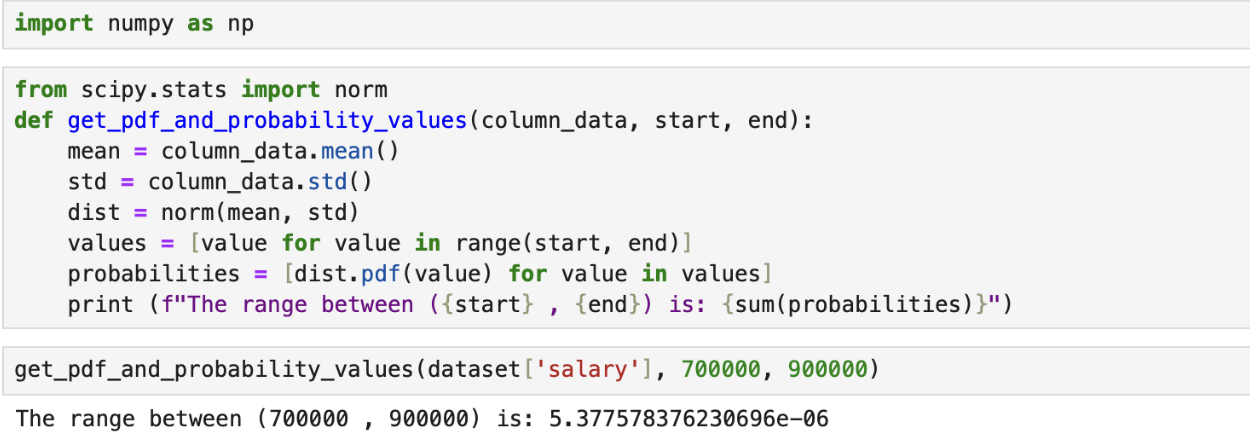
```
def get_normal_distribution(column_data):
    mean = column_data.mean()
    std = column_data.std()

    values = [i for i in column_data]

    z_values = [(j-mean)/std for j in values]

    sns.distplot(z_values, kde=True, kde_kws={'color': 'blue'}, color= 'green')
```

```
get_normal_distribution(dataset['salary'])
```



Using above function, we converts the normal distribution to standard normal distribution of salary values

10. **What is the probability Density Function for salary range from 700000 to 900000**



```
import numpy as np
```

```
from scipy.stats import norm
def get_pdf_and_probability_values(column_data, start, end):
    mean = column_data.mean()
    std = column_data.std()
    dist = norm(mean, std)
    values = [value for value in range(start, end)]
    probabilities = [dist.pdf(value) for value in values]
    print (f"The range between ({start} , {end}) is: {sum(probabilities)}")
```

```
get_pdf_and_probability_values(dataset['salary'], 700000, 900000)
```

```
The range between (700000 , 900000) is: 5.377578376230696e-06
```

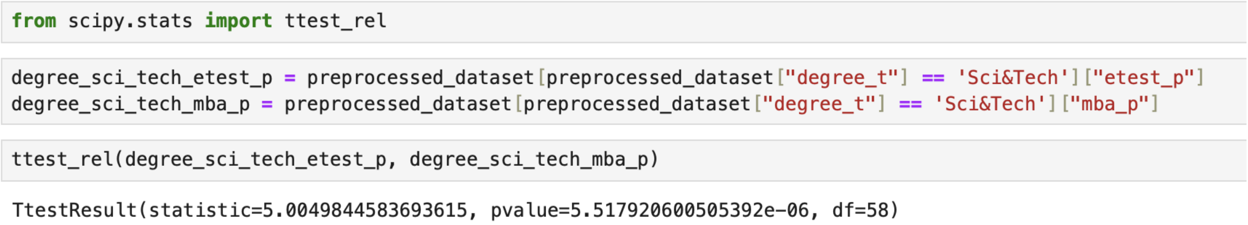
As per the above result, there is 0.005% of probability to get salary between 700000 to 900000

11. **Test the similarity between degree_t(sci&tech) and with respect to etest_p and mba_p at significant level 5% (make decision using hypothesis testing)**

Null hypothesis (Ho) - There is no significant changes degree_t(sci&tech) and with respect to etest_p and mba_p
Alternate Hypothesis (H1) - There is significant changes degree_t(sci&tech) and specialisation (Mkt&HR) with respect to salary

Accept/Reject - if p < 0.0.5 reject null hypothesis or else accept null hypothesis and reject alternative hypothesis

It falls under paried-t test



```
from scipy.stats import ttest_rel
```

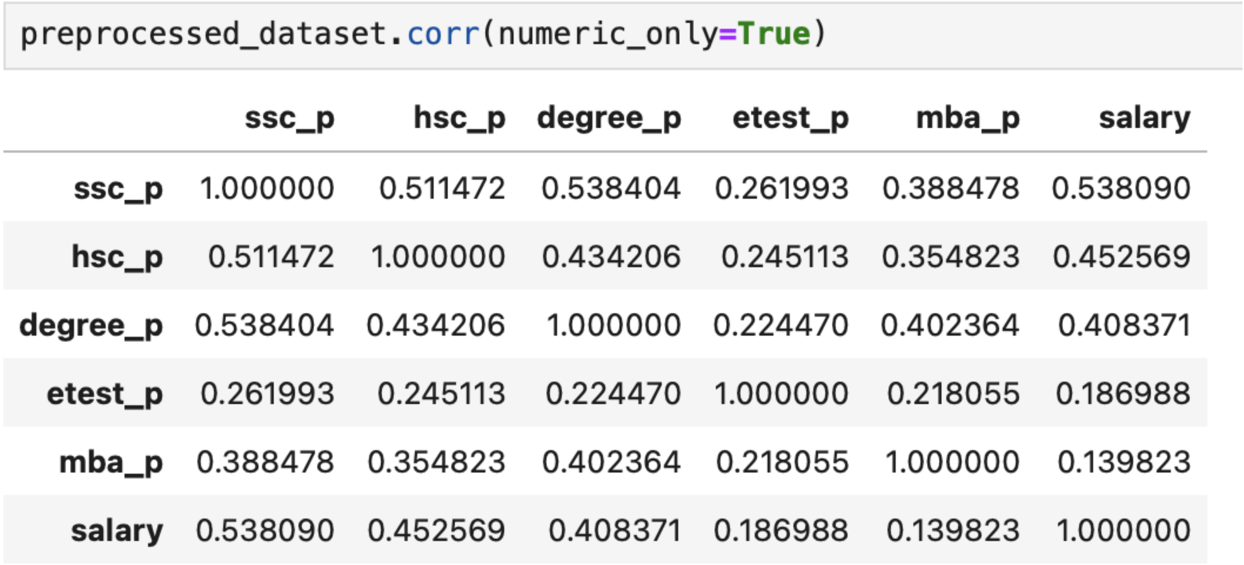
```
degree_sci_tech_etest_p = preprocessed_dataset[preprocessed_dataset["degree_t"] == 'Sci&Tech']["etest_p"]
degree_sci_tech_mba_p = preprocessed_dataset[preprocessed_dataset["degree_t"] == 'Sci&Tech']["mba_p"]
```

```
ttest_rel(degree_sci_tech_etest_p, degree_sci_tech_mba_p)
```

```
TtestResult(statistic=5.0049844583693615, pvalue=5.517920600505392e-06, df=58)
```

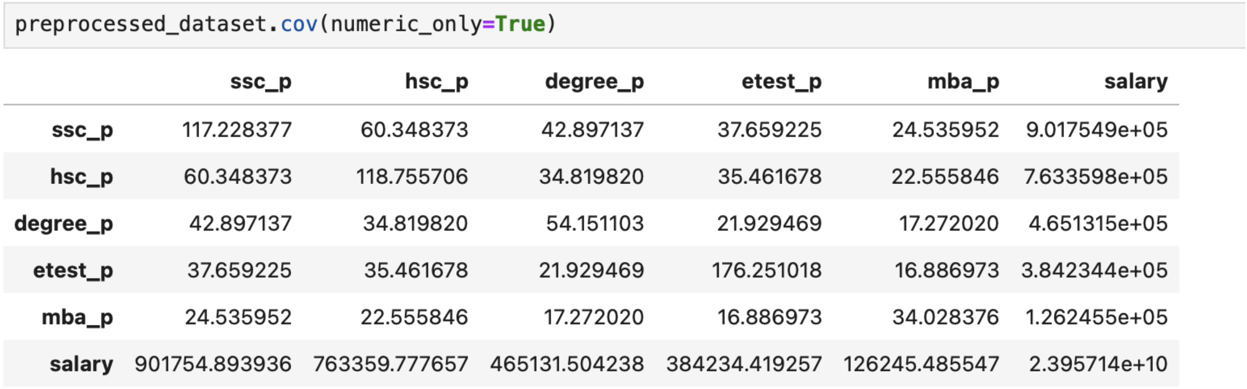
Here the p value 5.5 > 0.005 so we can accept Ho (null hypothesis)

12. Which parameter is highly correlated with salary



```
preprocessed_dataset.corr(numeric_only=True)
```

	ssc_p	hsc_p	degree_p	etest_p	mba_p	salary
ssc_p	1.000000	0.511472	0.538404	0.261993	0.388478	0.538090
hsc_p	0.511472	1.000000	0.434206	0.245113	0.354823	0.452569
degree_p	0.538404	0.434206	1.000000	0.224470	0.402364	0.408371
etest_p	0.261993	0.245113	0.224470	1.000000	0.218055	0.186988
mba_p	0.388478	0.354823	0.402364	0.218055	1.000000	0.139823
salary	0.538090	0.452569	0.408371	0.186988	0.139823	1.000000



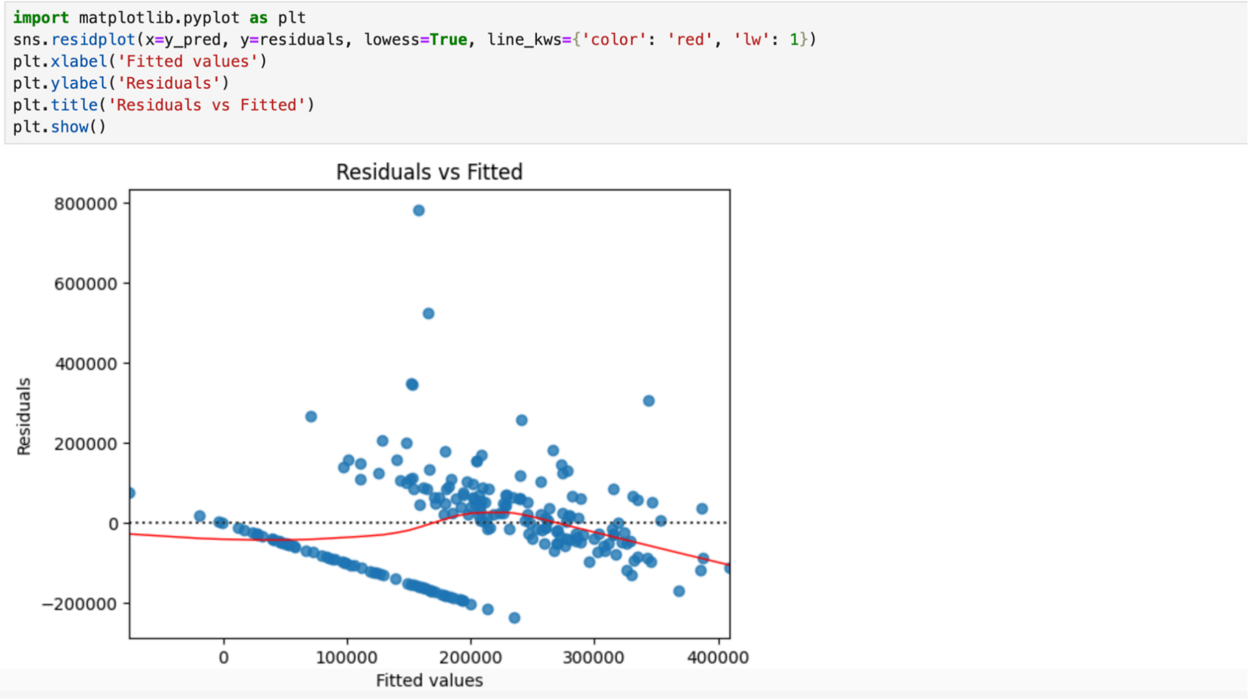
```
preprocessed_dataset.cov(numeric_only=True)
```

	ssc_p	hsc_p	degree_p	etest_p	mba_p	salary
ssc_p	117.228377	60.348373	42.897137	37.659225	24.535952	9.017549e+05
hsc_p	60.348373	118.755706	34.819820	35.461678	22.555846	7.633598e+05
degree_p	42.897137	34.819820	54.151103	21.929469	17.272020	4.651315e+05
etest_p	37.659225	35.461678	21.929469	176.251018	16.886973	3.842344e+05
mba_p	24.535952	22.555846	17.272020	16.886973	34.028376	1.262455e+05
salary	901754.893936	763359.777657	465131.504238	384234.419257	126245.485547	2.395714e+10

As per the above result, ssc_p and salary has 0.5 which is positive and greater than other correlation values and covariance of between ssc_p and salary is 9 which is positive (both increase/decrease together with difference 9%) and greater than other covariance values so **ssc_p** is highly correlated with salary

13. Plot any useful graph and explain it

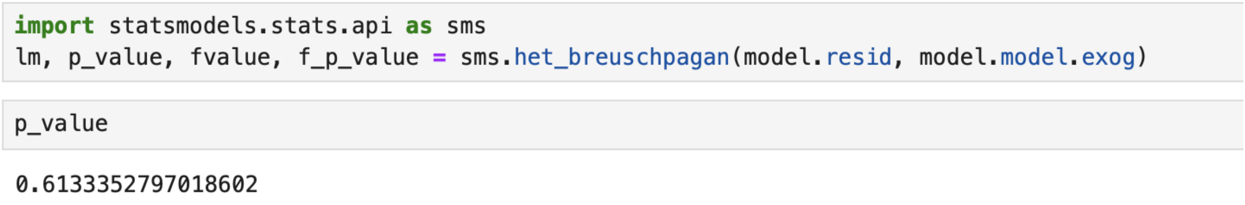
Let check if the dataset has homoscedasticity or heteroscedasticity



```
import matplotlib.pyplot as plt
sns.residplot(x=y_pred, y=residuals, lowess=True, line_kws={'color': 'red', 'lw': 1})
plt.xlabel('Fitted values')
plt.ylabel('Residuals')
plt.title('Residuals vs Fitted')
plt.show()
```

Here , there is no pattern in the interval of residuals so we can say the dataset has homoscedasticity And also I checked the p-value using Breusch-Pagan test to check if we have heteroscedasticity in the dataset or not

The null hypothesis (H0): Signifies that Homoscedasticity is present.
The alternative hypothesis: (Ha): Signifies that the Homoscedasticity is not present (i.e. heteroscedasticity exists)
If $p < 0.05$, reject null hypothesis



```
import statsmodels.stats.api as sms
lm, p_value, fvalue, f_p_value = sms.het_breuschpagan(model.resid, model.model.exog)
```

```
p_value
```

```
0.6133352797018602
```

Since the p-value (0.61) is greater than 0.05 so we couldn't reject the null hypothesis. Hence, We do not have enough proof to say that heteroscedasticity is present in the regression model