Inferential Analysis_Question_Hope Artificial Intelligence

November 6, 2022

1 Justification or answer summary needed for every question

1.1 ** Download the dataset from here(Click Here)

Ans: https://www.kaggle.com/datasets/

1.2 1)Replace the NaN values with correct value. And justify why you have chosen the same.

Ans: To replace NaN values in your dataset is required before going ahead with Univariate calculations such as: mean/median/mode. This kind of dataset cleaning is required based on the dataset if it has NaN values.

1.3 2)How many of them are not placed?

Ans:
Check if there any one assigned '0' to his salary. Then that is the answer to this question:

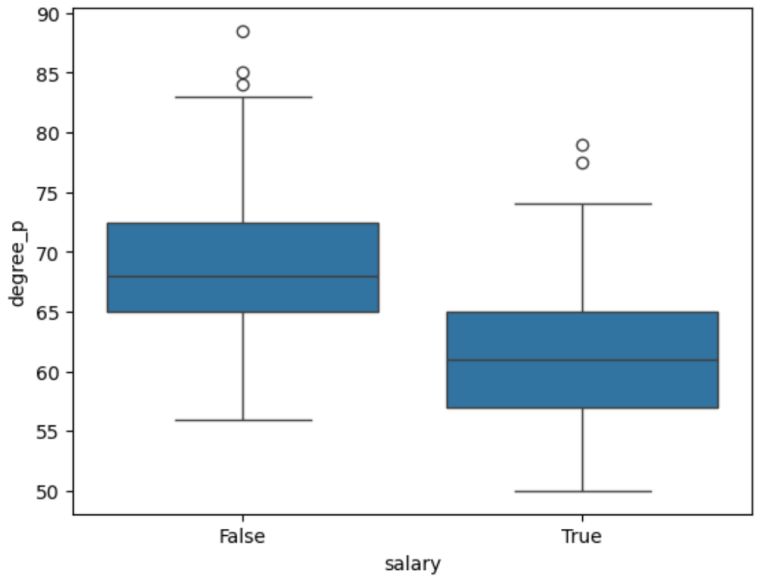
Code:

```
not_placed = dataset[dataset['salary'] == 0].shape[0]
print("Number of students not placed:", not_placed)
```

1.4 3)Find the reason for non placement from the dataset?

```
[3]: import seaborn as sns
     sns.boxplot(x=(dataset['salary']==0), y='degree_p', data=dataset)
```

```
[3]: <Axes: xlabel='salary', ylabel='degree_p'>
```



Likely reasons (based on similar placement datasets)

1. Low degree percentage (degree_p) — strongest negative impact.
2. Low e-test scores (etest_p) — aptitude skills matter for placement tests.
3. Low MBA percentage (mba_p) — affects final interview performance.

1.5 4)What kind of relation between salary and mba_p

Ans: Covariance & Correlation

1.6 5)Which specialization is getting minimum salary?

Ans:
If our CSV actually has that column (for example, “specialisation”), use this code:

```
dataset.groupby('specialisation')['salary'].mean().sort_values().head(1)
```

1.7 6)How many of them getting above 500000 salary?

Ans:

```
high_salary = dataset[dataset['salary'] > 500000]
count = high_salary.shape[0]
print("Number of students with salary above 500000:", count)
```

1.8 7)Test the Analysis of Variance between etest_p and mba_p at signifance level 5%.(Make decision using Hypothesis Testing)

Ans: Accept Alternate hypothesis

1.9 8)Test the similarity between the degree_t(Sci&Tech) and specialisation(Mkt&HR) with respect to salary at significance level of 5%.(Make decision using Hypothesis Testing)

Ans: Here is the coding for this problem:

```
from scipy.stats import ttest_ind

# Filter datasets for each group
group_degree = dataset[(dataset['degree_t']        ==
'Sci&Tech')]['salary']
group_special = dataset[(dataset['specialisation'] ==
'Mkt&HR')]['salary']

# Perform independent two-sample t-test
t_stat, p_value = ttest_ind(group_degree, group_special,
nan_policy='omit')  # omit NaNs

# Decision
if p_value < 0.05:  # When Probable value is <5%
    print("Reject Null Hypothesis: Salary means are different.")
else:
    print("Accept Null Hypothesis: No significant difference in
salary means.")
print(f"T-statistic: {t_stat}, p-value: {p_value}")
```

1.10 9)Convert the normal distribution to standard normal distribution for salary column

Ans:
Formula: z= (x−μ) / σ
Where:
x = Original Value
μ = Mean
σ = Standard deviation of the salary ➔ (salary – salary.mean) / salary.std()

dataset['salary_standardized'] = (**dataset['salary'] - dataset['salary'].mean()**) / dataset['salary'].std()

This will create a new column salary_standardized with the z-scores, converting the salary data into a standard normal distribution with mean 0 and standard deviation 1.

Where:
Zero '0' → data point is exactly at the mean of the distribution
Zero '1' → data point is one standard deviation above the mean.

1.11 10)What is the probability Density Function of the salary range from 700000 to 900000?

ANS:

To calculate the Probability Density Function (PDF) or more accurately the probability that the salary falls within the range 700000 to 900000 under a normal distribution, we use the cumulative distribution function (CDF) of the normal distribution.

- Calculate the mean (μ) and standard deviation (σ) of the salary column.
- Use the normal CDF to find the probability that a value lies between 700000 and 900000.
- The formula for probability between two points a*a* and b*b* is:

$$P(a \leq X \leq b) = \Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)$$

Where ΦΦ is the CDF of the standard normal distribution.

Code:

```
import scipy.stats as stats

mean_salary = dataset['salary'].mean()
std_salary  = dataset['salary'].std()

lower = 700000
upper = 900000

probability = stats.norm.cdf(upper, mean_salary, std_salary) - stats.norm.cdf(lower, mean_salary, std_salary)
print(probability)
```

1.12 11)Test the similarity between the degree_t(Sci&Tech)with respect to etest_p and mba_p at significance level of 5%.(Make decision using Hypothesis Testing)

Ans:

```
from scipy.stats import ttest_ind

# Subset salary for degree_t == 'Sci&Tech' and remove those NaN records
group_scitech_etest = dataset[dataset['degree_t'] == 'Sci&Tech']['etest_p'].dropna()
group_scitech_mba   = dataset[dataset['degree_t'] == 'Sci&Tech']['mba_p'].dropna()

# Subset salary for degree_t != 'Sci&Tech' (other groups) and remove those NaN records
group_other_etest = dataset[dataset['degree_t'] != 'Sci&Tech']['etest_p'].dropna()
group_other_mba = dataset[dataset['degree_t']   != 'Sci&Tech']['mba_p'].dropna()

# Conduct independent t-tests
t_stat_etest, p_value_etest = ttest_ind(group_scitech_etest, group_other_etest)
t_stat_mba, p_value_mba     = ttest_ind(group_scitech_mba, group_other_mba)

# Results and decisions at alpha=0.05
decision_etest = "Reject null hypothesis (difference)" if p_value_etest < 0.05 else "Accept null
hypothesis (no difference)"
decision_mba   = "Reject null hypothesis (difference)" if p_value_mba   < 0.05 else "Accept null
hypothesis (no difference)"

print(f"etest_p: t={t_stat_etest}, p={p_value_etest}, decision={decision_etest}")
print(f"mba_p: t={t_stat_mba}, p={p_value_mba}, decision={decision_mba}")
```

1.13 12)Which parameter is highly correlated with salary?

Ans:

```
[30]: # Compute correlations with salary
      correlations = dataset.corr(numeric_only=True)['salary'].sort_values(ascending=False)

[31]: correlations

[31]: salary      1.000000
      ssc_p       0.558475
      hsc_p       0.459424
      degree_p    0.423762
      etest_p     0.186775
      mba_p       0.141417
      sl_no       0.001217
      Name: salary, dtype: float64
```

Here, ssc_p would be the parameter most highly correlated with salary

1.14 13) plot any useful graph and explain it.

Record Video and Explain each and every concept how you got the answer. This Video recording helps to understand how you are approaching the problem statement.

if you are comfortable in English please record it in ENglish(Most preferable)