Detail Report

DATA SCIENCE – IN CLASS CHALLENGE 1

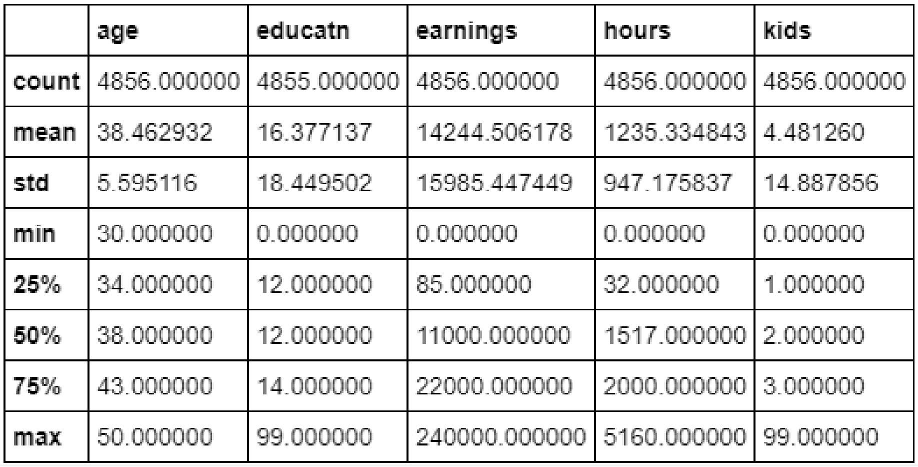
GROUP 10 – TEAM_LSEG

1. Introduction

The Panel Study of Income Dynamics (PSID) dataset contains information about 4856 people. It contains their age, education, earnings, hours, number of kids and their marital status. We are trying to analyze whether the number of hours a person work has an impact on his/her earnings.

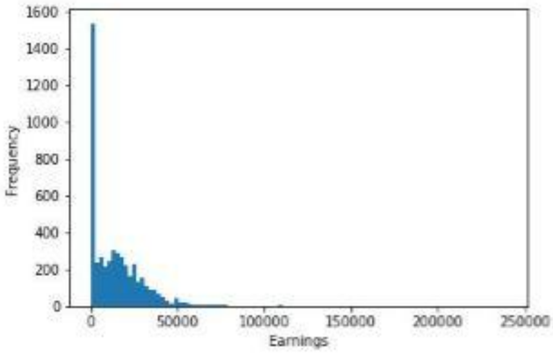
2. Experiments

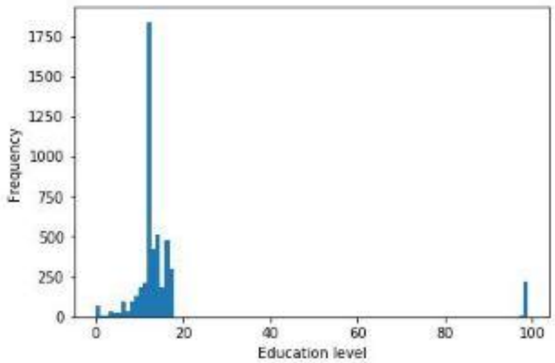
We observe all the statistics of the dataset first.

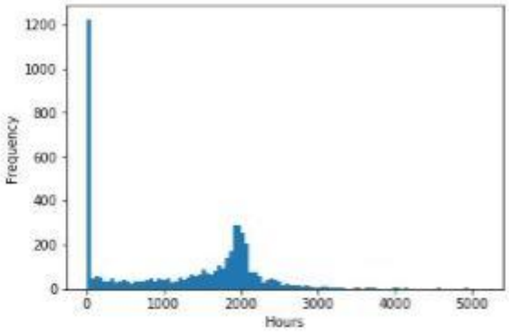


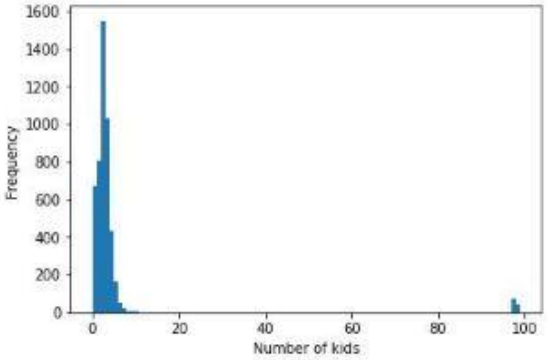
	age	educatn	earnings	hours	kids
count	4856.000000	4855.000000	4856.000000	4856.000000	4856.000000
mean	38.462932	16.377137	14244.506178	1235.334843	4.481260
std	5.595116	18.449502	15985.447449	947.175837	14.887856
min	30.000000	0.000000	0.000000	0.000000	0.000000
25%	34.000000	12.000000	85.000000	32.000000	1.000000
50%	38.000000	12.000000	11000.000000	1517.000000	2.000000
75%	43.000000	14.000000	22000.000000	2000.000000	3.000000
max	50.000000	99.000000	240000.000000	5160.000000	99.000000

Then we removed the outliers with the help of following histograms.

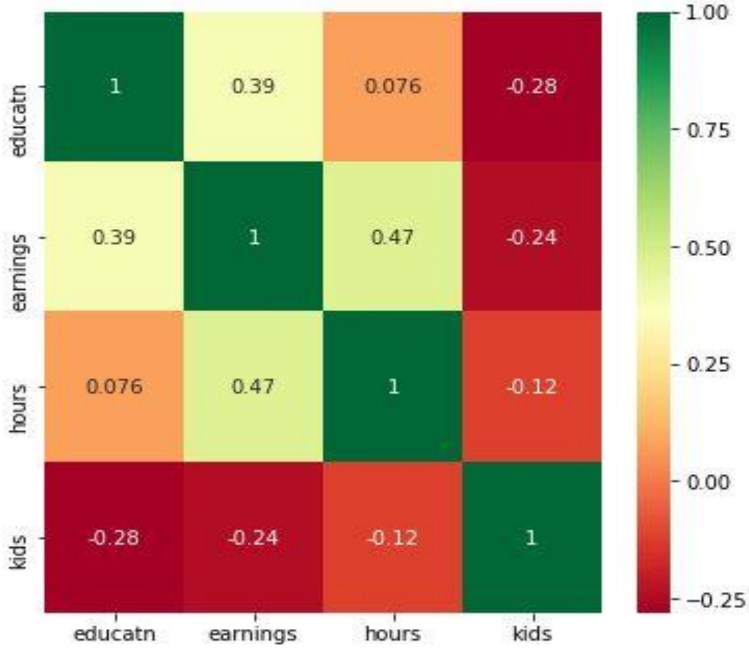








We used **Pearson's correlation coefficient** to identify related variables. We observed the high correlation (=0.466571) between "earnings" and "hours" variables.



Therefore following hypothesis is concluded by the team,

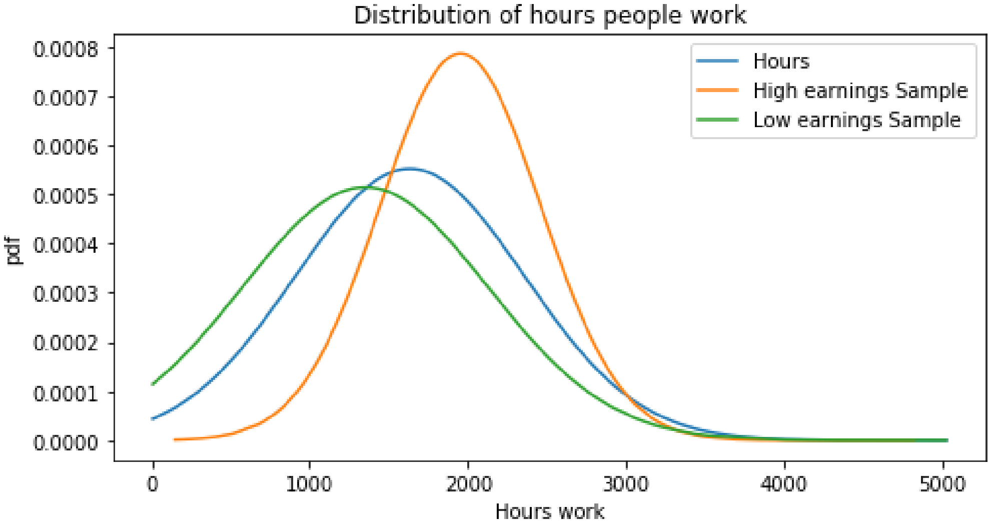
Ho = People with different salaries work the same number of hours
Ha = People who get high salary work more hours

3. Output

Our claim can be proved with the following statistics.

Statistics=20.562, p=0.000
People who get high salary work more hours (reject Ho)

The probability distribution of population and samples below clearly states our claim.



4. Code

1. Python libraries used (**Scipy.stats, pandas, matplotlib, numpy, math, csv**)
2. Missing value handling

```
my_data['educatn'] = my_data['educatn'].fillna(my_data['educatn'] .mean())
```

3. Invalid data removal code Example as below,

```
my_data = my_data.drop(my_data[my_data.educatn > 20].index)
```

4. Verifying whether the hours are normally distributed.

```
value, p = stat.normaltest(my_data['hours'].sample(50))
if p >= 0.05:
      print('It is likely that hours is normally distributed.')
```

5. Correlation calculations and heat map generation code.

```
my_data.corr(method='pearson')
```

6. Sampling code.

```
Sample = my_data[my_data.earnings> median].hours.sample(size).sort_values()
```

7. Hypothesis testing code.

```
statistics, p = stat.ttest_ind(Hours_of_high_salary,Hours_of_low_salary)
alpha = 0.05  # 5% area under the normal graph
if p > alpha:
 print('People with different salaries work equal hours (fail to reject H0)')
else:
 print('People who get high salary work more hours (reject H0)')
```