Кольца Илья Вячеславович

Лабораторная работа № 3

Вариант 1

In [45]:

```
import pandas as pd
import warnings
import numpy as np
import scipy.stats as sts
import rpy2.robjects.numpy2ri
from rpy2.robjects.packages import importr
rpy2.robjects.numpy2ri.activate()
stats = importr('stats')
warnings.filterwarnings('ignore')
```

In [23]:

```
df = pd.read_csv('Data.csv', header=0)
df.head()
```

Out[23]:

	Age	AttendedBootcamp	BootcampFinish	BootcampFullJobAfter	BootcampLoanYesNo	BootcampMonthsAgo	BootcampNa
0	28.0	0.0	NaN	NaN	NaN	NaN	N
1	22.0	0.0	NaN	NaN	NaN	NaN	N
2	19.0	0.0	NaN	NaN	NaN	NaN	N
3	26.0	0.0	NaN	NaN	NaN	NaN	N
4	20.0	0.0	NaN	NaN	NaN	NaN	N

5 rows × 113 columns

Новый раздел

In [24]:

```
df = df[['EmploymentField', 'EmploymentStatus', 'Gender', 'JobPref', 'JobWherePref', 'MaritalStatus', 'Income']]
df.head()
```

Out[24]:

	EmploymentField	EmploymentStatus	Gender	JobPref	JobWherePref	MaritalStatus	Income
0	office and administrative support	Employed for wages	male	freelance	NaN	married or domestic	32000.0

	EmploymentField	EmploymentStatus	Gender	JobPref	JobWherePref	MaritalStatus	Income
1	food and beverage	Employed for wages	male	work for a startup	in an office with other developers	NaN	15000.0
2	finance	Employed for wages	male	start your own business	NaN	NaN	48000.0
3	arts, entertainment, sports, or media	Employed for wages	female	work for a startup	from home	NaN	43000.0
4	education	Employed for wages	female	work for a medium-sized company	in an office with other developers	NaN	6000.0

In [25]:

```
df = df.dropna()
df = df[(df.Gender == 'male') | (df.Gender == 'female')]
df.head()
```

Out[25]:

	EmploymentField	EmploymentStatus	Gender	JobPref	JobWherePref	MaritalStatus	Income
59	software development	Employed for wages	male	work for a medium-sized company	in an office with other developers	married or domestic partnership	35000.0
71	education	Employed for wages	male	work for a multinational corporation	from home	married or domestic partnership	56000.0
72	transportation	Employed for wages	male	work for a medium-sized company	from home	married or domestic partnership	35000.0
77	arts, entertainment, sports, or media	Employed for wages	male	work for a medium-sized company	from home	married or domestic partnership	65000.0
90	sales	Employed for wages	male	work for a startup	in an office with other developers	single, never married	30000.0

In [26]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 764 entries, 59 to 15616
Data columns (total 7 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   EmploymentField   764 non-null    object 
 1   EmploymentStatus  764 non-null    object 
 2   Gender            764 non-null    object 
 3   JobPref           764 non-null    object 
 4   JobWherePref      764 non-null    object 
 5   MaritalStatus     764 non-null    object 
 6   Income            764 non-null    float64
dtypes: float64(1), object(6)
memory usage: 47.8+ KB
```

In [27]:

```
df.groupby('Gender').count()
```

Out[27]:

Gender	EmploymentField	EmploymentStatus	JobPref	JobWherePref	MaritalStatus	Income
female	127	127	127	127	127	127
male	637	637	637	637	637	637

In [28]:

```
pairs = [['Gender', 'JobPref'],
         ['Gender', 'JobWherePref'],
         ['JobWherePref', 'MaritalStatus'],
         ['EmploymentField', 'JobWherePref'],
         ['EmploymentStatus', 'JobWherePref']]
```

In [29]:

```
pair = pairs[0]
print(f'Таблица сопряженности для {pair[0]} и {pair[1]}')
ct = pd.crosstab(df[pair[0]], df[pair[1]])
ct
```

```
Таблица сопряженности для Gender и JobPref
```

Out[29]:

JobPref	work for a medium-sized company	work for a multinational corporation	work for a startup
Gender			
female	82	19	26
male	362	103	172

In [30]:

```
print(f'Таблица ожидаемых значений для {pair[0]} и {pair[1]}')
et = pd.DataFrame(sts.contingency.expected_freq(ct), index=ct.index, columns=ct.columns)
et
```

```
Таблица ожидаемых значений для Gender и JobPref
```

Out[30]:

JobPref	work for a medium-sized company	work for a multinational corporation	work for a startup
Gender			
female	73.806283	20.280105	32.913613
male	370.193717	101.719895	165.086387

In [31]:

```
print(f"Проверка гипотезы об отсутствии связи между {pair[0]} и {pair[1]}:")
z, p, _, _ = sts.chi2_contingency(ct, correction=False)
print("Статистика критерия =", z)
print("Достигаемый уровень значимости =", p)
if p < 0.05:
    print("Гипотеза об отсутствии связи отвергается.")
else:
    print("Гипотеза об отсутствии связи принимается.")
```

```
Проверка гипотезы об отсутствии связи между Gender и JobPref:
Статистика критерия = 2.9296663743177596
Достигаемый уровень значимости = 0.2311165414688363
Гипотеза об отсутствии связи принимается.
```

In [32]:

```
pair = pairs[1]
print(f'Таблица сопряженности для {pair[0]} и {pair[1]}')
ct = pd.DataFrame(pd.crosstab(df[pair[0]], df[pair[1]]))
ct
```

```
Таблица сопряженности для Gender и JobWherePref
```

Out[32]:

	in an office with other

JobWherePref	from home	in an office with other developers	no preference
Gender			
female	38	57	32
male	149	317	171

In [33]:

```
print(f'Таблица ожидаемых значений для {pair[0]} и {pair[1]}')
et = pd.DataFrame(sts.contingency.expected_freq(ct), index=ct.index, columns=ct.columns)
et
```

```
Таблица ожидаемых значений для Gender и JobWherePref
```

Out[33]:

JobWherePref	from home	in an office with other developers	no preference
Gender			
female	31.085079	62.170157	33.744764
male	155.914921	311.829843	169.255236

In [34]:

```
print(f"Проверка гипотезы об отсутствии связи между {pair[0]} и {pair[1]}:")
z, p, _, _ = sts.chi2_contingency(ct, correction=False)
print("Статистика критерия =", z)
print("Достигаемый уровень значимости =", p)
if p < 0.05:
    print("Гипотеза об отсутствии связи отвергается.")
else:
    print("Гипотеза об отсутствии связи принимается.")
```

```
Проверка гипотезы об отсутствии связи между Gender и JobWherePref:
Статистика критерия = 2.468792878230615
Достигаемый уровень значимости = 0.29101035183846335
Гипотеза об отсутствии связи принимается.
```

In [35]:

```
pair = pairs[2]
print(f'Таблица сопряженности для {pair[0]} и {pair[1]}')
ct = pd.DataFrame(pd.crosstab(df[pair[0]], df[pair[1]]))
ct
```

```
Таблица сопряженности для JobWherePref и MaritalStatus
```

Out[35]:

MaritalStatus	divorced	married or domestic partnership	separated	single, never married
JobWherePref				
from home	12	153	2	20
in an office with other developers	14	291	2	67
no preference	11	149	4	39

In [36]:

```
print(f'Таблица ожидаемых значений для {pair[0]} и {pair[1]}')
et = pd.DataFrame(sts.contingency.expected_freq(ct), index=ct.index, columns=ct.columns)
et
```

```
Таблица ожидаемых значений для JobWherePref и MaritalStatus
```

Out[36]:

MaritalStatus	divorced	married or domestic partnership	separated	single, never married
JobWherePref				
from home	9.056283	145.145288	1.958115	30.840314
in an office with other developers	18.112565	290.290576	3.916230	61.680628
no preference	9.831152	157.564136	2.125654	33.479058

In [39]:

```
ct.values
```

Out[39]:

```
array([[ 12, 153,   2,  20],
       [ 14, 291,   2,  67],
       [ 11, 149,   4,  39]])
```

In [44]:

```
print(f"Проверка гипотезы об отсутствии связи между {pair[0]} и {pair[1]}:")
test = stats.fisher_test(ct.values)
p = test[0][0]
#print("Статистика критерия =", z)
print("Достигаемый уровень значимости =", p)
if p < 0.05:
    print("Гипотеза об отсутствии связи отвергается.")
else:
    print("Гипотеза об отсутствии связи принимается.")
```

```
Проверка гипотезы об отсутствии связи между JobWherePref и MaritalStatus:
Достигаемый уровень значимости = 0.06912479693278728
Гипотеза об отсутствии связи принимается.
```

In [46]:

```
pair = pairs[3]
print(f'Таблица сопряженности для {pair[0]} и {pair[1]}')
ct = pd.DataFrame(pd.crosstab(df[pair[0]], df[pair[1]]))
ct
```

```
Таблица сопряженности для EmploymentField и JobWherePref
```

Out[46]:

JobWherePref	from home	in an office with other developers	no preference
EmploymentField			
architecture or physical engineering	8	15	4
arts, entertainment, sports, or media	17	25	7
construction and extraction	1	11	4
education	29	46	26
farming, fishing, and forestry	1	1	1
finance	8	27	8
food and beverage	9	13	10
health care	10	18	14
law enforcement and fire and rescue	1	4	2
legal	2	3	2
office and administrative support	14	33	23
sales	11	22	16
software development	0	4	0

JobWherePref	from home	in an office with other developers	no preference
software development and IT	69	143	82
transportation	7	9	4
EmploymentField			

In [47]:

```
print(f'Таблица ожидаемых значений для {pair[0]} и {pair[1]}')
et = pd.DataFrame(sts.contingency.expected_freq(ct), index=ct.index, columns=ct.columns)
et
```

```
Таблица ожидаемых значений для EmploymentField и JobWherePref
```

Out[47]:

JobWherePref	from home	in an office with other developers	no preference
EmploymentField			
architecture or physical engineering	6.608639	13.217277	7.174084
arts, entertainment, sports, or media	11.993455	23.986911	13.019634
construction and extraction	3.916230	7.832461	4.251309
education	24.721204	49.442408	26.836387
farming, fishing, and forestry	0.734293	1.468586	0.797120
finance	10.524869	21.049738	11.425393
food and beverage	7.832461	15.664921	8.502618
health care	10.280105	20.560209	11.159686
law enforcement and fire and rescue	1.713351	3.426702	1.859948
legal	1.713351	3.426702	1.859948
office and administrative support	17.133508	34.267016	18.599476
sales	11.993455	23.986911	13.019634
software development	0.979058	1.958115	1.062827
software development and IT	71.960733	143.921466	78.117801
transportation	4.895288	9.790576	5.314136

In [51]:

```
print(f"Проверка гипотезы об отсутствии связи между {pair[0]} и {pair[1]}:")
test = stats.fisher_test(ct.values, simulate_p_value=True)
p = test[0][0]
#print("Статистика критерия =", z)
print("Достигаемый уровень значимости =", p)
if p < 0.05:
    print("Гипотеза об отсутствии связи отвергается.")
else:
    print("Гипотеза об отсутствии связи принимается.")
```

```
Проверка гипотезы об отсутствии связи между EmploymentField и JobWherePref:
Достигаемый уровень значимости = 0.5817091454272864
Гипотеза об отсутствии связи принимается.
```

In [57]:

```
pair = pairs[4]
print(f'Таблица сопряженности для {pair[0]} и {pair[1]}')
ct = pd.DataFrame(pd.crosstab(df[pair[0]], df[pair[1]]))
ct
```

```
Таблица сопряженности для EmploymentStatus и JobWherePref
```

Out[57]:

	in an office with other

JobWherePref	from home	in an office with other developers	no preference
EmploymentStatus			
Employed for wages	164	330	187
Self-employed business owner	5	10	4
Self-employed freelancer	18	34	12

In [58]:

```
print(f'Таблица ожидаемых значений для {pair[0]} и {pair[1]}')
et = pd.DataFrame(sts.contingency.expected_freq(ct), index=ct.index, columns=ct.columns)
et
```

```
Таблица ожидаемых значений для EmploymentStatus и JobWherePref
```

Out[58]:

JobWherePref	from home	in an office with other developers	no preference
EmploymentStatus			
Employed for wages	166.684555	333.369110	180.946335
Self-employed business owner	4.650524	9.301047	5.048429
Self-employed freelancer	15.664921	31.329843	17.005236

In [59]:

```
print(f"Проверка гипотезы об отсутствии связи между {pair[0]} и {pair[1]}:")
test = stats.fisher_test(ct.values)
p = test[0][0]
#print("Статистика критерия =", z)
print("Достигаемый уровень значимости =", p)
if p < 0.05:
    print("Гипотеза об отсутствии связи отвергается.")
else:
    print("Гипотеза об отсутствии связи принимается.")
```

```
Проверка гипотезы об отсутствии связи между EmploymentStatus и JobWherePref:
Достигаемый уровень значимости = 0.6162678097062643
Гипотеза об отсутствии связи принимается.
```

In [70]:

```
df['income_cat'] = pd.cut(df.Income,
                          bins=[df.Income.quantile(0), df.Income.quantile(0.33), df.Inco
me.quantile(0.67), df.Income.quantile(1)],
                          labels=['low', 'mid', 'high'])
df.head()
```

Out[70]:

	EmploymentField	EmploymentStatus	Gender	JobPref	JobWherePref	MaritalStatus	Income	income_cat
59	software development	Employed for wages	male	work for a medium-sized company	in an office with other developers	married or domestic partnership	35000.0	mid
71	education	Employed for wages	male	work for a multinational corporation	from home	married or domestic partnership	56000.0	high
72	transportation	Employed for wages	male	work for a medium-sized company	from home	married or domestic partnership	35000.0	mid
77	arts, entertainment, sports, or media	Employed for wages	male	work for a medium-sized company	from home	married or domestic partnership	65000.0	high
					in an office with			

90	EmploymentField	sales	EmploymentStatus	Employed for wages	Gender	male	JobPref	work for a startup	JobWherePref	other developers	MaritalStatus	single, never married	income	20000.0	income_cat	low

In [71]:

```
pair = ['income_cat', 'Gender']
print(f'Таблица сопряженности для {pair[0]} и {pair[1]}')
ct = pd.DataFrame(pd.crosstab(df[pair[0]], df[pair[1]]))
ct
```

Таблица сопряженности для income_cat и Gender

Out[71]:

Gender	female	male
income_cat		
low	37	204
mid	49	215
high	40	207

In [67]:

```
print(f'Таблица ожидаемых значений для {pair[0]} и {pair[1]}')
et = pd.DataFrame(sts.contingency.expected_freq(ct), index=ct.index, columns=ct.columns)
et
```

Таблица ожидаемых значений для income_cat и Gender

Out[67]:

Gender	female	male
income_cat		
low	40.380319	200.619681
mid	44.234043	219.765957
high	41.385638	205.614362

In [69]:

```
print(f"Проверка гипотезы об отсутствии связи между {pair[0]} и {pair[1]}:")
z, p, _, _ = sts.chi2_contingency(ct, correction=False)
print("Статистика критерия =", z)
print("Достигаемый уровень значимости =", p)
if p < 0.05:
    print("Гипотеза об отсутствии связи отвергается.")
else:
    print("Гипотеза об отсутствии связи принимается.")
```

```
Проверка гипотезы об отсутствии связи между income_cat и Gender:
Статистика критерия = 1.012521167081471
Достигаемый уровень значимости = 0.6027452855370555
Гипотеза об отсутствии связи принимается.
```