Лабораторная работа №0 по курсу машинного обучения

Выполнил студент группы М80-306Б-19 Полюбин Арсений.

Поставленная задача:

Определить задачу которую вы хотите решить и найти под нее соответствующие данные.

Задача: Предсказать зарабатывают ли люди более 50 тысяч долларов в год или нет. Для этой задачи будем использовать датасет: Adult Income.

Такая задача может пригодиться в банке, где принимается решения давать кредит человеку или нет.

Adult Income

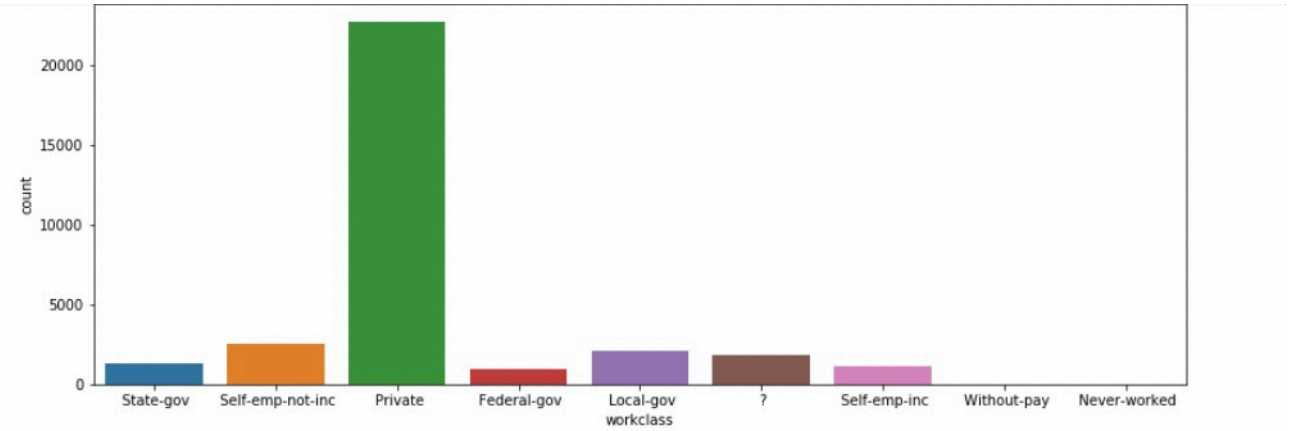
Колонки:

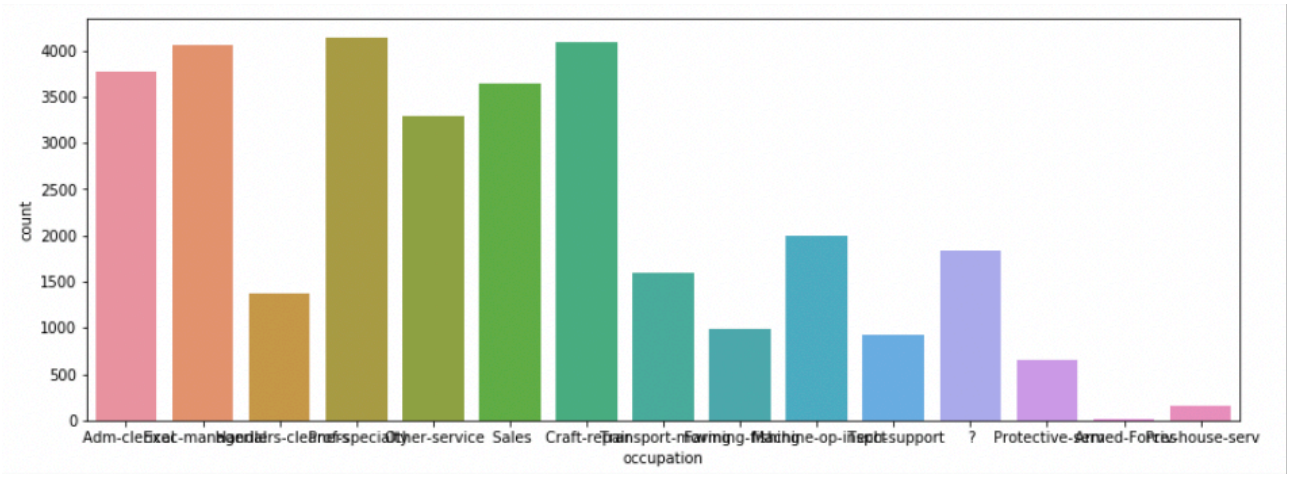
- sex (.object) – пол
- income (.object) – заработок
- race (.object) – раса
- relationship (.object) – отношения
- marital-status (.object) – семейное положение
- workclass (.object) – рабочий класс
- occupation (.object) – сфера работы
- education (.object) – образование
- education-num (.int) – образовательный номер

- native-country (object) – родная страна
- age (.int) – возраст
- capital-loss (.int) – потери капитала
- hours-per-week (.int) – кол-во рабочих часов в неделю
- capital-gain (.int) – прибыль
- fnlwgt (.int) – некоторый числовой параметр

Видим, что много признаков имеют класс .object, что является проблемой, с этим нужно поработать.

Графики распределения значений данных, имеющих пропущенные значения





Как видно, в первом и в третьем случае целесообразно будет осуществлять замену пропущенных данных самым часто встречающимся значением.

3. Для определения стратегии работы с данными типа object посмотрим, сколько уникальных значений в каждой колонке датасета.

	Column_Name	Type	Num_Unique
9	sex	object	2
14	income	object	2
8	race	object	5
7	relationship	object	6
5	marital-status	object	7
1	workclass	object	9
6	occupation	object	15
3	education	object	16
4	education-num	int32	16
13	native-country	object	42
0	age	int32	73
11	capital-loss	int32	92
12	hours-per-week	int32	94
10	capital-gain	int32	119
2	fnlwgt	int32	21648

4.

По полученным результатам имеем следующее: колонки с двумя уникальными значениями представим в виде бинарных признаков, колонки с количеством уникальных элементов до десяти включительно заменим эквивалентными

колонками с бинарными признаками с помощью one-hot encoding, все остальные колонки типа object представим типом category. Колонки capital-loss и capital-gain нормализуем.

Для корректной работы с пропущенными признаками приведем такие данные к типу None с помощью написанной функции.

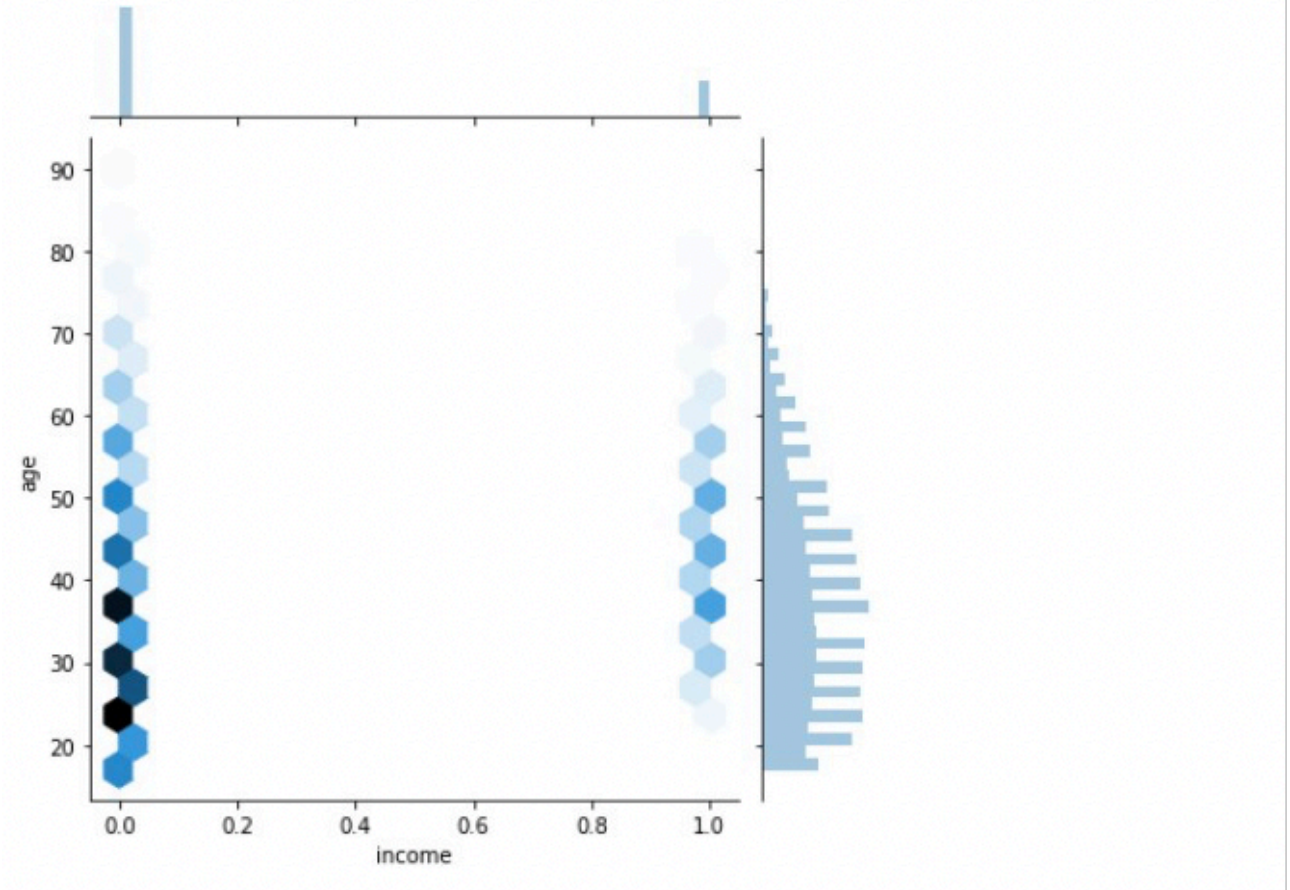
```
def make_NaN(df):
    for col in df.columns:
        df[col][df[col] == ' ?'] = None
```

```
make_NaN(train)
make_NaN(test)
```

5.

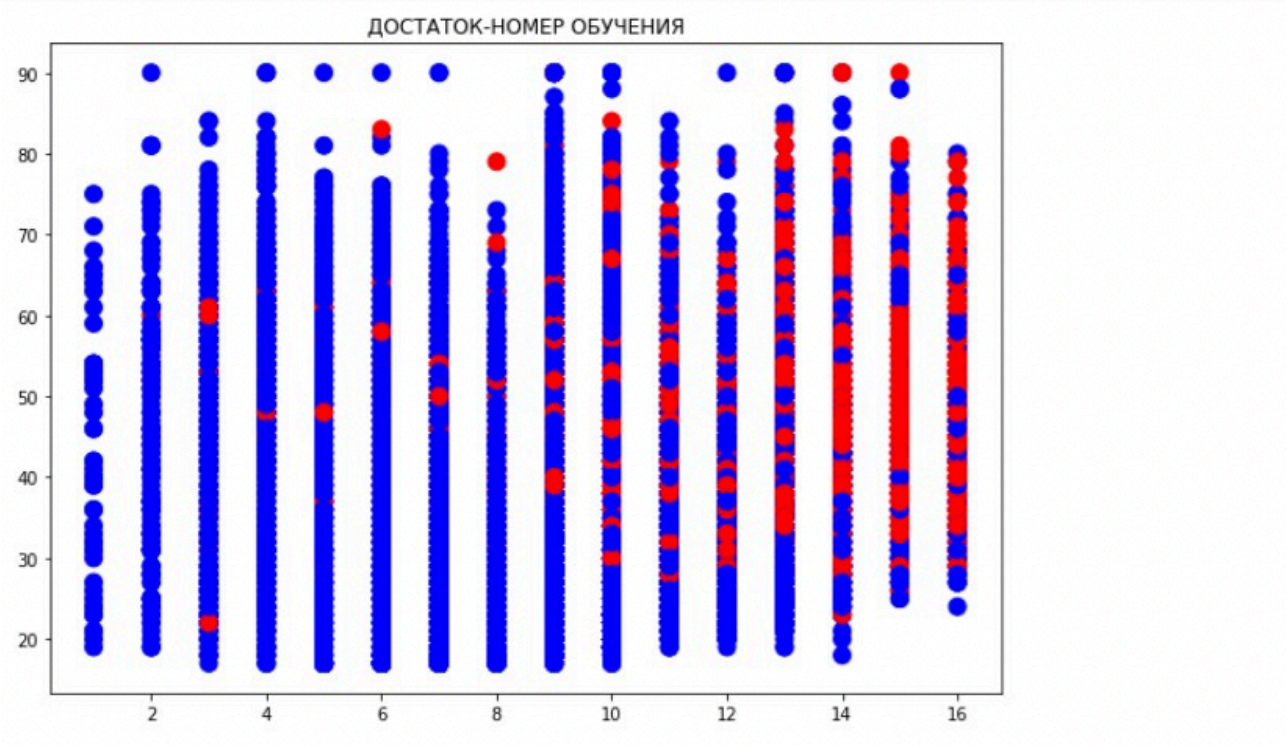
```
# ДОСТАТОК-ВОЗРАСТ
fig = plt.figure(figsize=(40, 40))
sns.jointplot(x='income', y='age', data=train, kind ='hex',gridsize=20)
```

```
<seaborn.axisgrid.JointGrid at 0x7f8dd4ae9438>

<Figure size 2880x2880 with 0 Axes>
```



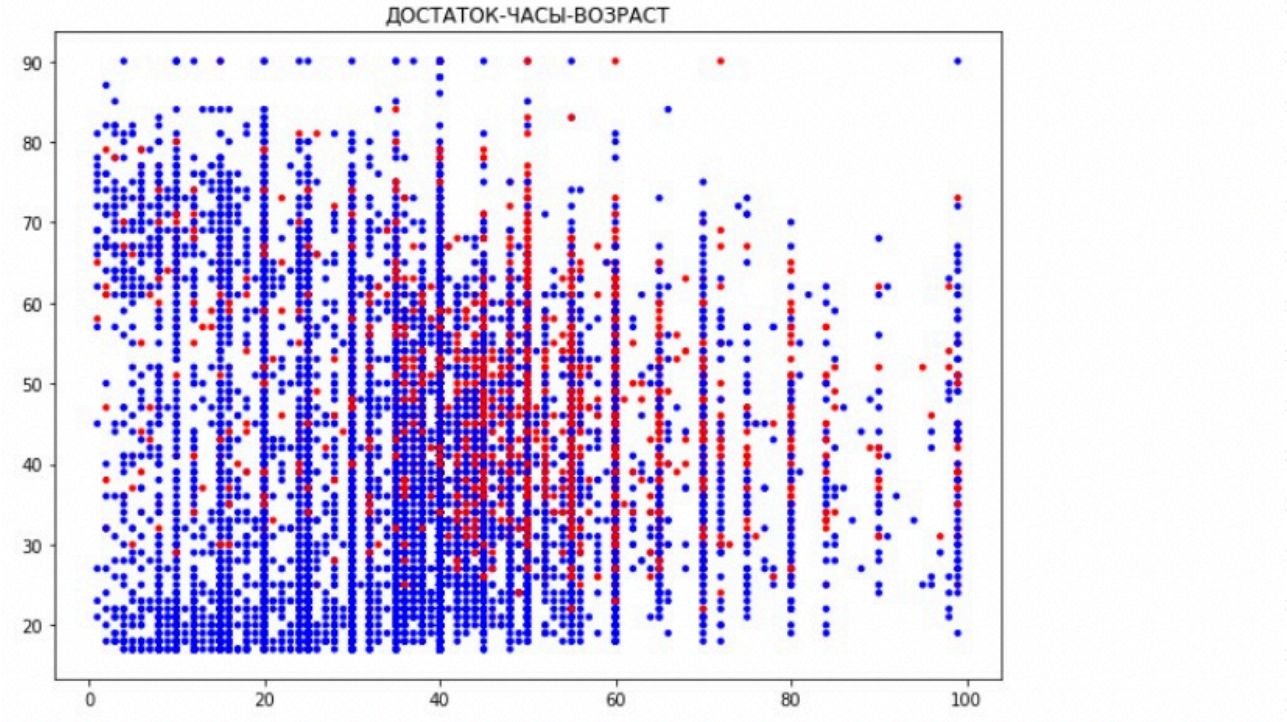
Как видно, большинство людей до 40 лет имеют заработок менее 50 тысяч долларов в год.

```
# ДОСТАТОК-НОМЕР ОБУЧЕНИЯ-ВОЗРАСТ
colors = ListedColormap(["blue", "red"])
# 0 - blue
# 1 - red
plt.figure(figsize=(10,7))
plt.title("ДОСТАТОК-НОМЕР ОБУЧЕНИЯ")
plt.scatter(train['education-num'], train['age'], c=train['income'], cmap=colors, s=100)
plt.show()
```



Большинство людей, имеющих большой заработок, относятся к 12 — 16 номеру обучения.

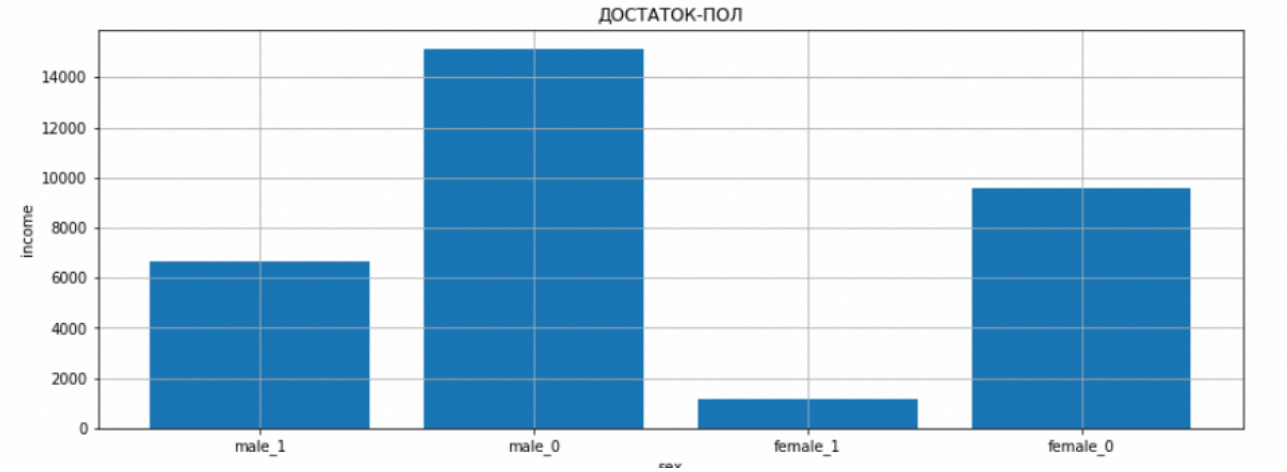
```
# ДОСТАТОК-ЧАСЫ-ВОЗРАСТ
colors = ListedColormap(["blue", "red"])
# 0 - blue
# 1 - red
plt.figure(figsize=(10,7))
plt.title("ДОСТАТОК-ЧАСЫ-ВОЗРАСТ")
plt.scatter(train['hours-per-week'], train['age'], c=train['income'], cmap=colors, s=10)
plt.show()
#те кто мало или много работают, скорее всего, мало не получают
```



Данный график показывает, что люди, имеющие большой заработок, преимущественно работают от 40 до 60 часов в неделю.

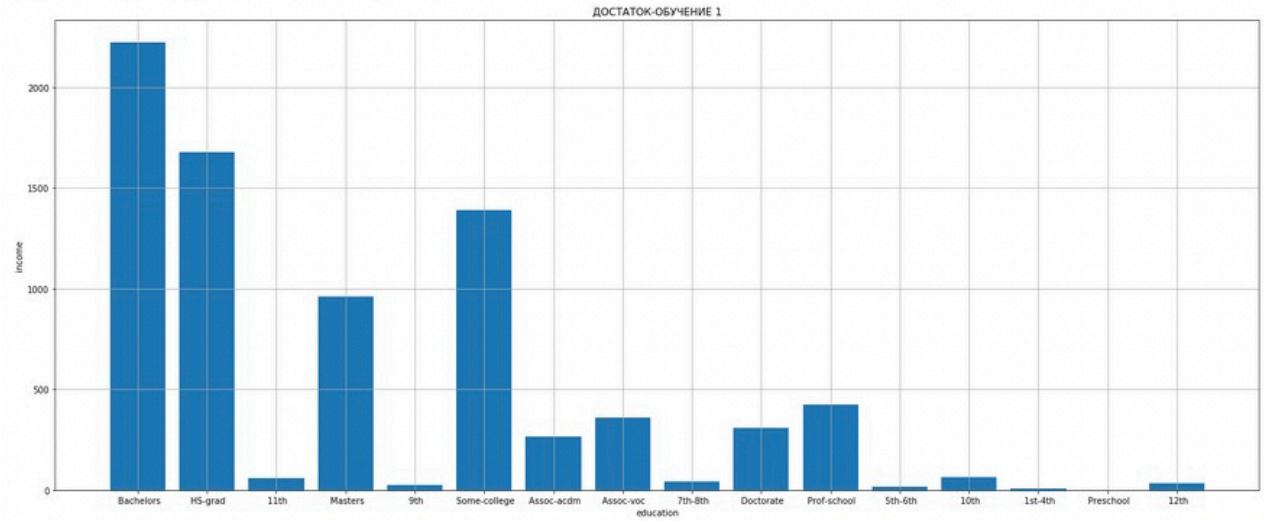
```
# ДОСТАТОК-ПОЛ

sex = ['male_1','male_0', 'female_1', 'female_0']

male_1 = train[train.sex == ' Male'][['income']].income.sum()
male_0 = train[(train.sex == ' Male') & (train.income == 0)][['sex']].count()
female_1 = train[train.sex == ' Female'][['income']].income.sum()
female_0 = train[(train.sex == ' Female') & (train.income == 0)][['sex']].count()

cls = [male_1, male_0, female_1, female_0]

fig = plt.figure(figsize=(14,5))
ax1 = fig.add_subplot()
ax1.set_xlabel('sex')
ax1.set_ylabel('income')
ax1.set_title('ДОСТАТОК-ПОЛ')
ax1.grid()
ax1.bar(sex, cls)
```

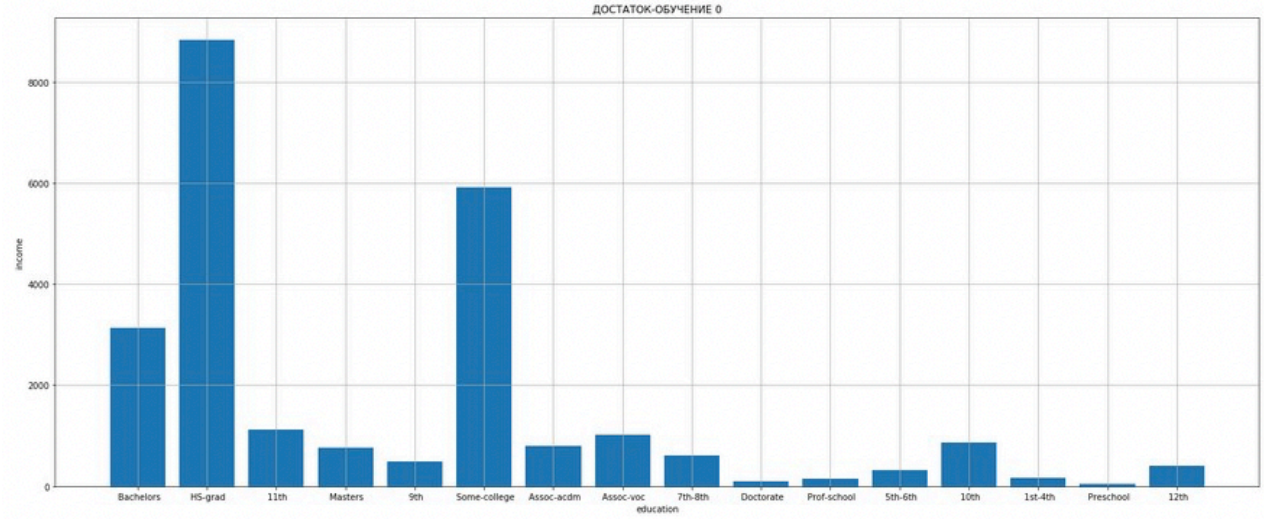
```
<BarContainer object of 4 artists>
```



Здесь явно прослеживается, что треть мужчин имею большой заработок.

```
<BarContainer object of 16 artists>
```





По данному графику видно, что многие виды образования сильно влияют на заработок человека.

6. Далее произведем отбор наиболее значимых численных признаков с помощью корреляции.

```
train.corr()
# age, education-num, capital-gain, capital-loss, hours-per-week
```

	age	fnlwgt	education-num	capital-gain	capital-loss	hours-per-week	income
age	1.000000	-0.076646	0.036527	0.077674	0.057775	0.068756	0.234037
fnlwgt	-0.076646	1.000000	-0.043195	0.000432	-0.010252	-0.018768	-0.009463
education-num	0.036527	-0.043195	1.000000	0.122630	0.079923	0.148123	0.335154
capital-gain	0.077674	0.000432	0.122630	1.000000	-0.031615	0.078409	0.223329
capital-loss	0.057775	-0.010252	0.079923	-0.031615	1.000000	0.054256	0.150526
hours-per-week	0.068756	-0.018768	0.148123	0.078409	0.054256	1.000000	0.229689
income	0.234037	-0.009463	0.335154	0.223329	0.150526	0.229689	1.000000

Видно, что меньше всего income коррелирует с fnlwgt, поэтому исключим его из рассмотрения.

Таким образом мы выбрали наиболее значимые признаки для нашей модели, осталось ее обучить!