# Date_App_OKCupid____Project_ML___

May 18, 2022

Name: Petr Vlasak

Email: petr.vlasakk@gmail.com

GitHub: https://github.com/pvlasak

================================================================================

This project is focused on data analysis and building of machine learning algorithms for the data set provided by OKCupid Company with usage of Python Scikit-learn library.

================================================================================

Data set contains several columns, where information about each individual OKCupid user are registered. Here are the column names shown as a Pandas object:

```
Index(['age', 'body_type', 'diet', 'drinks', 'drugs', 'education', 'essay0',
       'essay1', 'essay2', 'essay3', 'essay4', 'essay5', 'essay6', 'essay7',
       'essay8', 'essay9', 'ethnicity', 'height', 'income', 'job',
       'last_online', 'location', 'offspring', 'orientation', 'pets',
       'religion', 'sex', 'sign', 'smokes', 'speaks', 'status'],
      dtype='object')
```

Following columns in the data set contain NaN value:

```
age                0
body_type       5296
diet           24395
drinks          2985
drugs          14080
education       6628
essay0          5488
essay1          7572
essay2          9638
essay3         11476
essay4         10537
essay5         10850
essay6         13771
essay7         12451
essay8         19225
essay9         12603
```

```
ethnicity       5680
height             3
income             0
job             8198
last_online        0
location           0
offspring      35561
orientation        0
pets           19921
religion       20226
sex                0
sign           11056
smokes          5512
speaks            50
status             0
dtype: int64
```

# 1 Transformation to numerical data

## 1.1 State codes

5 the most frequent states in the data set and counts of OKCupid users living there:

```
State
california       59855
new york            17
illinois             8
massachusetts        5
oregon               4
dtype: int64
```

Majority of OKCupid Users are located in California. State codes are defined as follows:

```
California : 0
Other in US : 1
Other : 2
```

Below you can see python dictionary used for data transformation.

```
{'california': 0, 'new york': 1, 'illinois': 1, 'massachusetts': 1, 'michigan':
1, 'oregon': 1, 'texas': 1, 'arizona': 1, 'florida': 1, 'colorado': 1, 'district
of columbia': 1, 'spain': 2, 'ohio': 1, 'minnesota': 1, 'united kingdom': 2,
'virginia': 1, 'georgia': 1, 'utah': 1, 'hawaii': 1, 'washington': 1,
'missouri': 1, 'pennsylvania': 1, 'germany': 2, 'new jersey': 1, 'montana': 1,
'north carolina': 1, 'rhode island': 1, 'wisconsin': 1, 'tennessee': 1,
'canada': 2, 'switzerland': 2, 'connecticut': 1, 'mississippi': 1, 'idaho': 1,
'west virginia': 1, 'ireland': 1, 'mexico': 1, 'nevada': 1, 'louisiana': 1,
'vietnam': 2, 'netherlands': 2}
```

## 1.2 City codes

5 the most frequent cities and counts of OKCupid users living there:

```
City
san francisco    31064
oakland           7214
berkeley          4212
san mateo         1331
palo alto         1064
dtype: int64
```

Majority of OKCupid Users are located in San Francisco. City codes are defined as follows:

```
san francisco : 0
oakland : 1
berkeley : 2
san mateo : 3
palo alto : 4
other : 5
```

Below you can see python dictionary used for data transformation.

```
{'san francisco': 0, 'oakland': 1, 'berkeley': 2, 'san mateo': 3, 'palo alto':
4, 'alameda': 5, 'san rafael': 5, 'hayward': 5, 'emeryville': 5, 'redwood city':
5, 'daly city': 5, 'san leandro': 5, 'walnut creek': 5, 'vallejo': 5, 'menlo
park': 5, 'richmond': 5, 'south san francisco': 0, 'mountain view': 5, 'novato':
5, 'burlingame': 5, 'pleasant hill': 5, 'castro valley': 5, 'stanford': 5, 'el
cerrito': 5, 'pacifica': 5, 'martinez': 5, 'mill valley': 5, 'san bruno': 5,
'san pablo': 5, 'belmont': 5, 'albany': 5, 'san carlos': 5, 'benicia': 5,
'lafayette': 5, 'sausalito': 5, 'millbrae': 5, 'san anselmo': 5, 'el sobrante':
5, 'san lorenzo': 5, 'fairfax': 5, 'hercules': 5, 'pinole': 5, 'half moon bay':
5, 'fremont': 5, 'green brae': 5, 'orinda': 5, 'moraga': 5, 'larkspur': 5,
'corte madera': 5, 'belvedere tiburon': 5, 'atherton': 5, 'brisbane': 5,
'rodeo': 5, 'crockett': 5, 'el granada': 5, 'foster city': 5, 'kentfield': 5,
'woodacre': 5, 'east palo alto': 5, 'montara': 5, 'ross': 5, 'piedmont': 5,
'westlake': 5, 'woodside': 5, 'los angeles': 5, 'new york': 5, 'lagunitas': 5,
'san geronimo': 5, 'bolinas': 5, 'point richmond': 5, 'moss beach': 5, 'west
oakland': 5, 'colma': 5, 'chicago': 5, 'san diego': 5, 'santa cruz': 5,
'tiburon': 5, 'hillsborough': 5, 'stinson beach': 5, 'portland': 5, 'nicasio':
5, 'brooklyn': 5, 'santa monica': 5, 'bayshore': 5, 'salt lake city': 5,
'redwood shores': 5, 'sacramento': 5, 'petaluma': 5, 'woodbridge': 5, 'los
gatos': 5, 'boston': 5, 'napa': 5, 'san jose': 5, 'long beach': 5, 'kensington':
5, 'santa rosa': 5, 'atlanta': 5, 'irvine': 5, 'tucson': 5, 'washington': 5,
'san quentin': 5, 'minneapolis': 5, 'forest knolls': 5, 'madrid': 5, 'chico': 5,
'rohnert park': 5, 'freedom': 5, 'bellingham': 5, 'jackson': 5, 'hacienda
heights': 5, 'boulder': 5, 'columbus': 5, 'leander': 5, 'santa ana': 5,
'concord': 5, 'cambridge': 5, 'austin': 5, 'south lake tahoe': 5, 'granite bay':
5, 'kula': 5, 'kassel': 5, 'union city': 5, 'philadelphia': 5, 'pacheco': 5,
'oakley': 5, 'bellwood': 5, 'kansas city': 5, 'san luis obispo': 5,
'fayetteville': 5, 'cork': 5, 'marin city': 5, 'brea': 5, 'islip terrace': 5,
'studio city': 5, 'taunton': 5, 'north hollywood': 5, 'south orange': 5, 'costa
mesa': 5, 'rochester': 5, 'providence': 5, 'utica': 5, 'honolulu': 5,
'vancouver': 5, 'edinburgh': 5, 'lake orion': 5, 'seattle': 5, 'stockton': 5,
'crowley': 5, 'riverside': 5, 'ozone park': 5, 'nevada city': 5, 'london': 5,
'milwaukee': 5, 'las vegas': 5, 'grand rapids': 5, 'waterford': 5, 'new
orleans': 5, 'asheville': 5, 'denver': 5, 'murfreesboro': 5, 'hilarita': 5,
'livingston': 5, 'santa clara': 5, 'cincinnati': 5, 'astoria': 5, 'south
wellfleet': 5, 'canyon country': 5, 'olema': 5, 'ashland': 5, 'billings': 5,
'miami': 5, 'nha trang': 5, 'boise': 5, 'phoenix': 5, 'isla vista': 5,
'milpitas': 5, 'vacaville': 5, 'campbell': 5, 'arcadia': 5, 'sunnyvale': 5,
'pasadena': 5, 'san antonio': 5, 'stratford': 5, 'peoria': 5, 'seaside': 5,
'magalia': 5, 'glencove': 5, 'amsterdam': 5, 'modesto': 5, 'oceanview': 5, 'fort
lauderdale': 5, 'port costa': 5, 'guadalajara': 5, 'canyon': 5, 'bonaduz': 5,
'muir beach': 5, 'longwood': 5, 'orange': 5}
```

### 1.3 Job codes

There are several career fields gained from the data set. Job codes are defined as follows:

```
transportation : 0,
hospitality / travel : 1,
student : 2,
artistic / musical / writer : 3,
computer / hardware / software : 4,
banking / financial / real estate : 5,
entertainment / media : 6,
sales / marketing / biz dev : 7,
medicine / health : 8,
science / tech / engineering : 9,
executive / management : 10,
education / academia : 11,
clerical / administrative : 12,
construction / craftsmanship : 13,
political / government : 14,
law / legal services : 15,
military : 16,
unemployed : 17,
retired : 18,
rather not say : 19,
other :20
```

### 1.4 Ethnicity Codes

Ethnicity data contains a lot of unique values. Ethnicity codes have been defined as seen below, the goal was to also reduce the number of unique categories.

```
White : 0
Black or African American : 1
```

```
Asian white : 2
Asian black : 3
Hispanic or latino : 4
Pacific Islander : 5
Other : 6
```

### 1.5 Diet codes

Every diet type has its unique numerical code. Diet codes are defined as follows:

```
 anything : 0,
 mostly anything : 1,
 strictly anything : 2,
 other : 3,
 mostly other : 4,
 strictly other : 5,
 vegetarian : 6,
 mostly vegetarian : 7,
 strictly vegetarian : 8,
 vegan : 9,
 mostly vegan : 10,
 strictly vegan :11,
 halal : 12,
 mostly halal : 13,
 strictly halal : 14,
 kosher : 15,
 mostly kosher : 16,
 strictly kosher : 17
```

### 1.6 Orientation

Sexual orientation codes are defined as follows

```
straight : 0,
gay : 1,
bisexual : 2
```

### 1.7 Drugs codes:

Drug codes are defined as follows:

```
never : 0,
sometimes : 1,
often :2
```

### 1.8 Drinks codes:

Drinks codes are defined as follows:

```
not at all : 0
rarely : 1
```

```
socially : 2
often : 3
very often : 4
desperately : 5
```

### 1.9 Smokes codes:

Smokes codes are defined as follows:

```
sometimes : 2
no : 0
when drinking: 1
yes : 4
trying to quit : 3
```

### 1.10 Education codes:

There are many unique entries of the education description in the data set. Following codes have been defined to code education column data:

```
elementary: 1
secondary: 2
post-secondary: 3
PhD: 4
```

### 1.11 Sex Codes

Sex codes are defined as follows:

```
male : 0
female: 1
```

### 1.12 Income Codes

Four income categories are defined:

```
category 0 : income not specified, valued equal to -1
category 1 : income 0 - 50 000 USD
category 2 : income 50 000 - 100 000 USD
category 3 : income more than 100 000 USD
```

### 1.13 Status code

Status column contains following unique entries.

```
single            55697
seeing someone     2064
available          1865
married             310
unknown              10
Name: status, dtype: int64
```

### 1.14 Body Type

Body type codes are defined as follows:

```
average           : 6,
fit               : 3,
athletic          : 4,
thin              : 1,
curvy             : 8,
a little extra    : 0,
skinny            : 2,
full figured      : 9,
overweight        : 10,
jacked            : 5,
used up           : 7,
rather not say    : 11
```

### 1.15 Relationship to pets

Relationship to pets is coded as follows:

```
like both cats and dogs : 0
like dogs : 1
like cats : 2
dislike both cats and dogs : 3
```

### 1.16 Offspring

Offspring codes are as follows:

```
doesn´t have a kids : 0,
doesn´t have kids and don´t want them : 1,
doens´t have a kids a want them : 2,
have kids : 3
have kids and do not want more : 4,
have kids and want more : 5,
```

### 1.17 Religion codes

Religion codes are as follows:

```
agnosticism : 2,
other : 1,
atheism : 0,
christianity : 3,
catholicism : 4,
judaism : 5,
buddhism : 6,
hinduism : 7,
islam : 8
```

# 2 Data analysis

Different data set variables will be plotted as different types of graph. Motivation is to get an overview about data distribution in the data set and about connections between variables.

## 2.1 Analysis of Foreign Languages

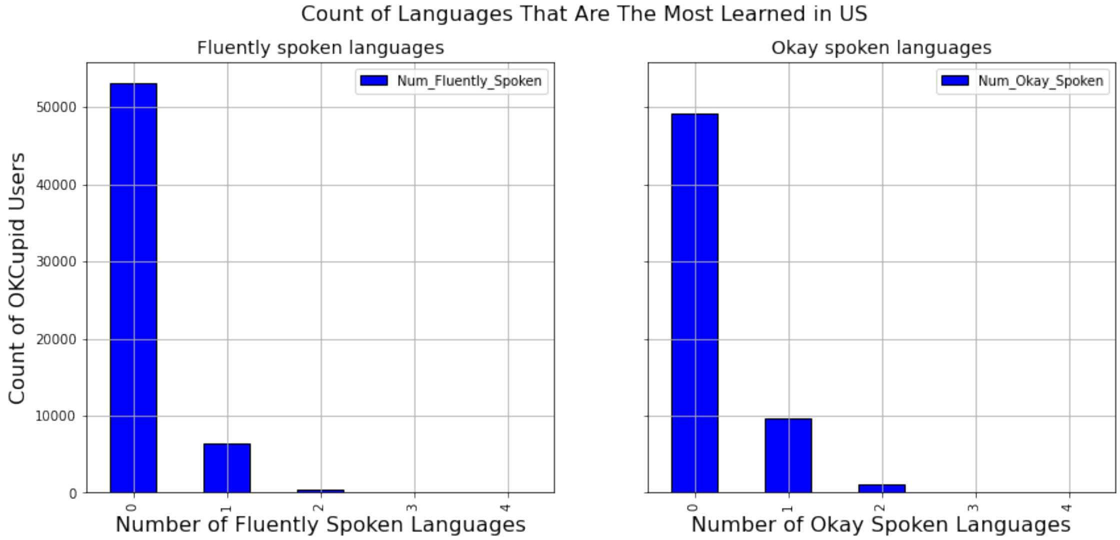
The most learned languages in US are listed below:

(https://www.infoplease.com/us/society-culture/most-studied-foreign-languages-us)

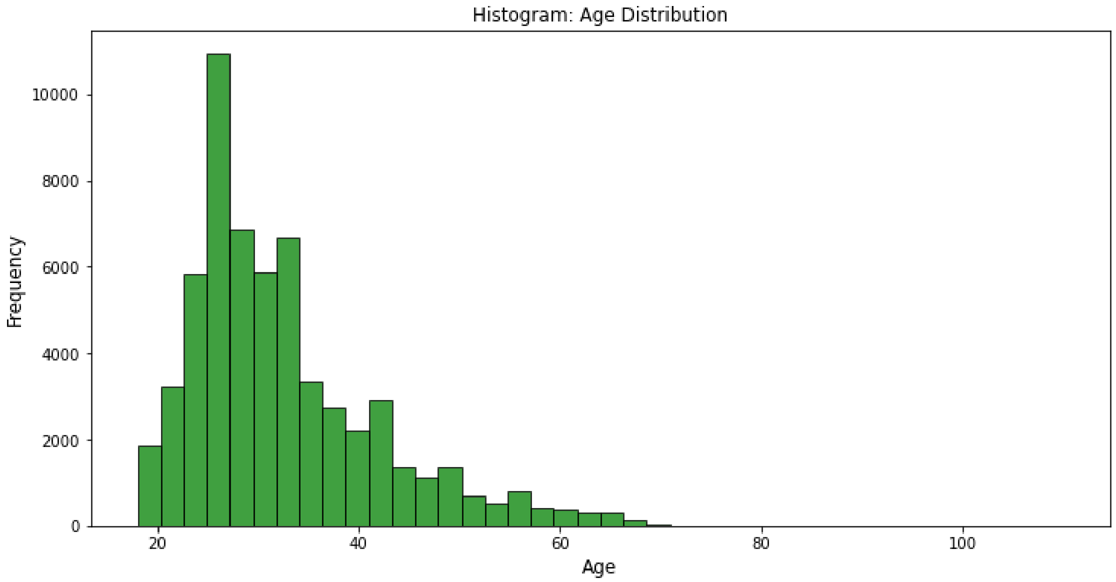
`spanish | french | german | italian | japanese | chinese | arabic | russian | portuguese | latin | korean`

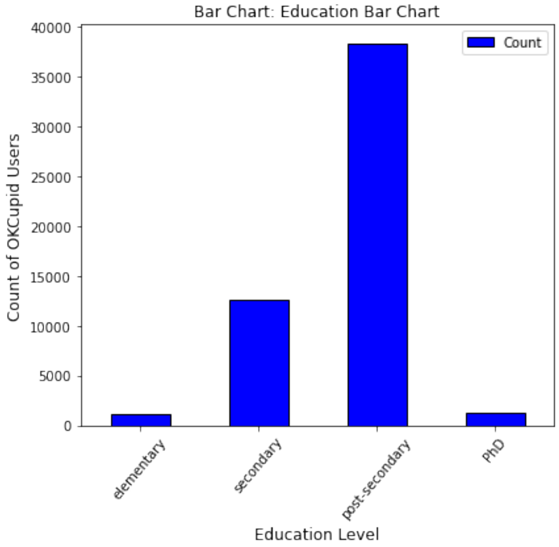
The graphs below are showing the number of OKCupid Users devided into categories based on the number of fluently and okay speaking languages. Only the most in US learned foreign languages are considered.

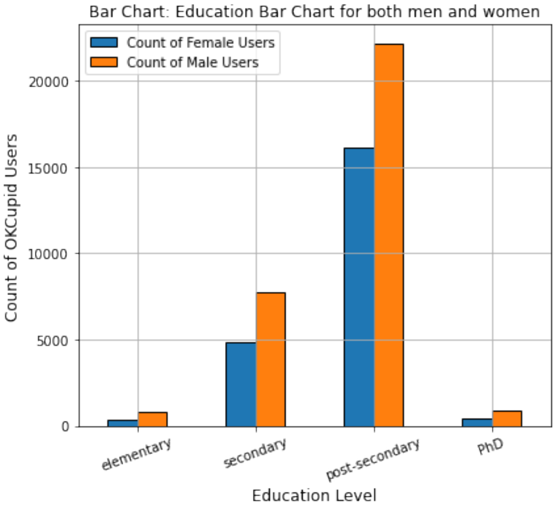
It clearly shows that majority of users speaks only english without knowledge of any foreign language.



## 2.2 Age Distribution and Education Level Counts





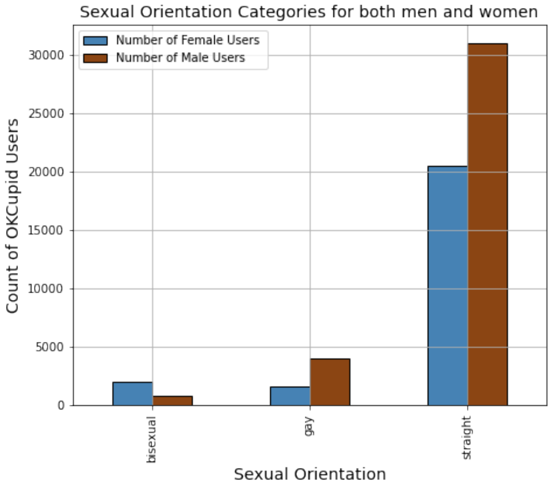


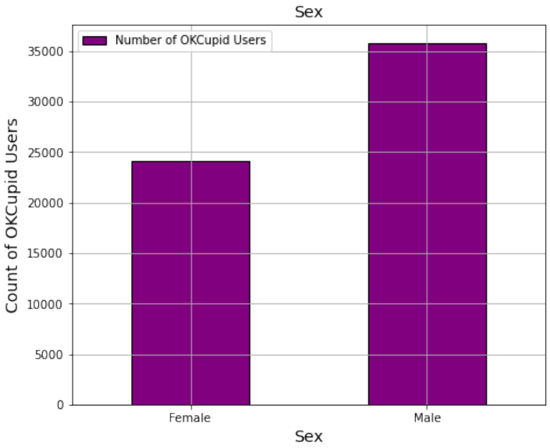
a. Majority of OKCupid Users are around 30 years old -> OKCupid dating app is very popular for young people of age between 25 - 35 years old.

b. Most of them also have finished their post-secondary studies as seen on the right bar chart.

c. Number of PhD graduates is very low in comparison with post-secondary and secondary education category.

## 2.3 Sexual orientation and Gender





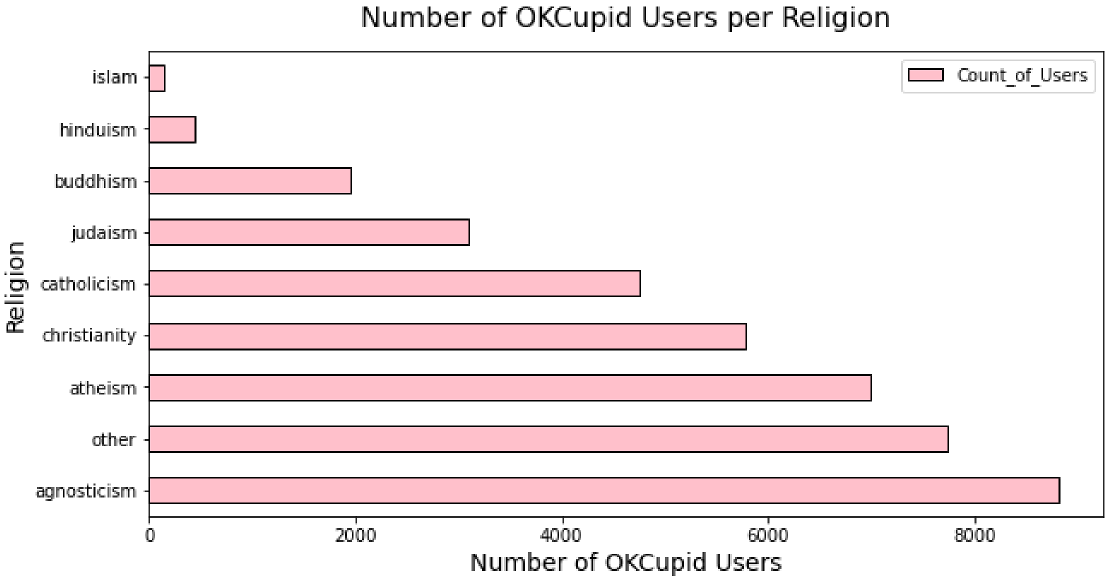
```
Data set OKCupid contains 35829 male and 24117 female profiles.
Percentage values are: 59.8 % male and 40.2 % female profiles.
```

a. Bisexual orientation is more common for female sex.

b. Gay orientation is more common for male sex.

c. Number of male profiles is higher by 11712.

## 2.4 Religion

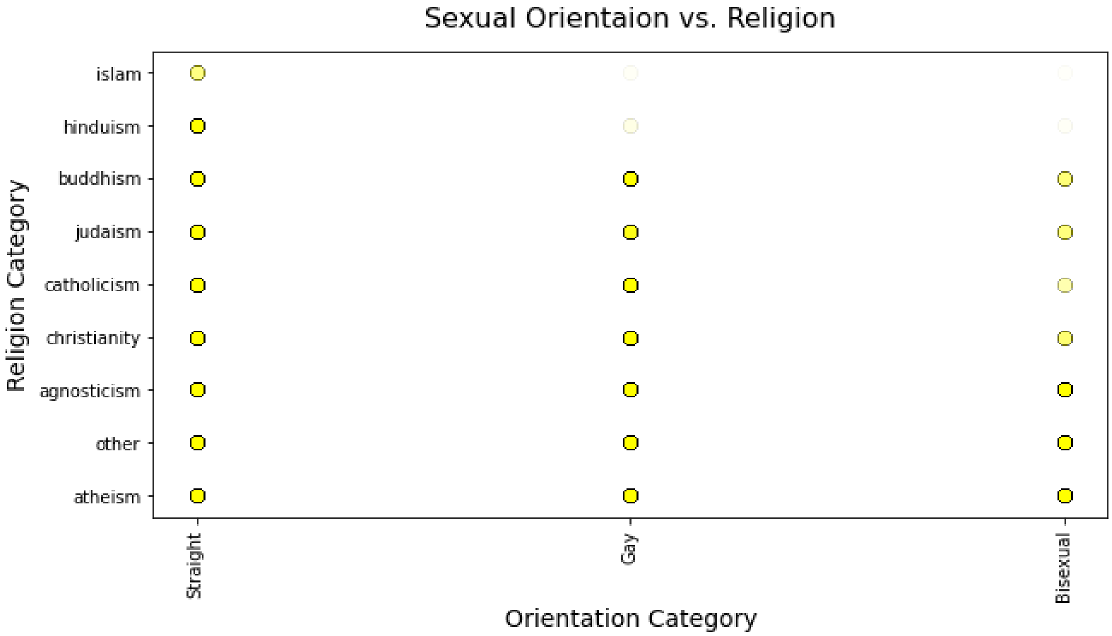


a. Islam and hinduism are minor religions in the data set.

b. The most common religions in the dataset are agnosticism, atheism and other spiritual approaches.

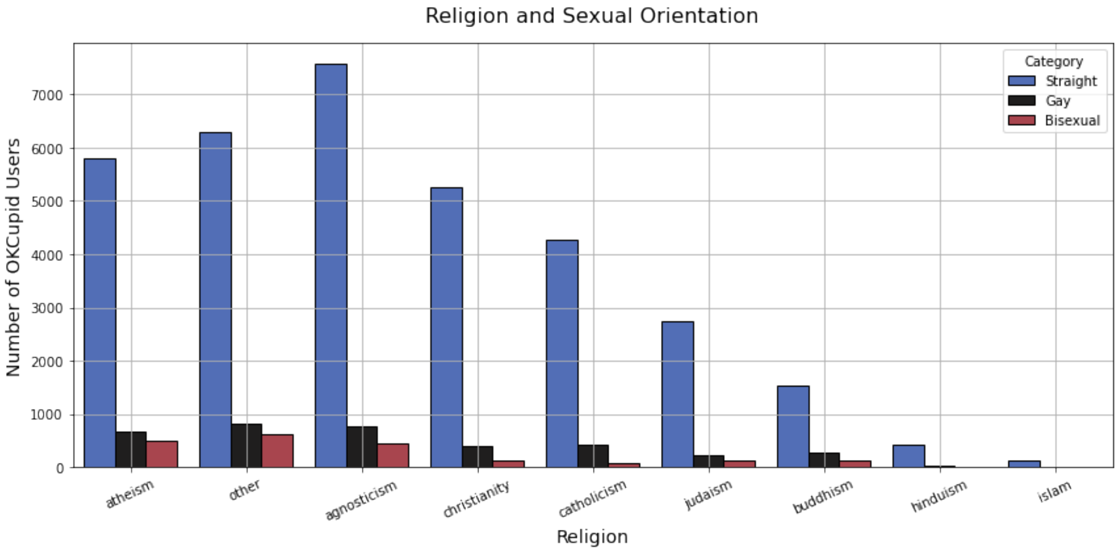
### 2.5 Sexual Orientation and Religion

Relationship between sexual orientation and religion feature are visualized by the scatter plot.



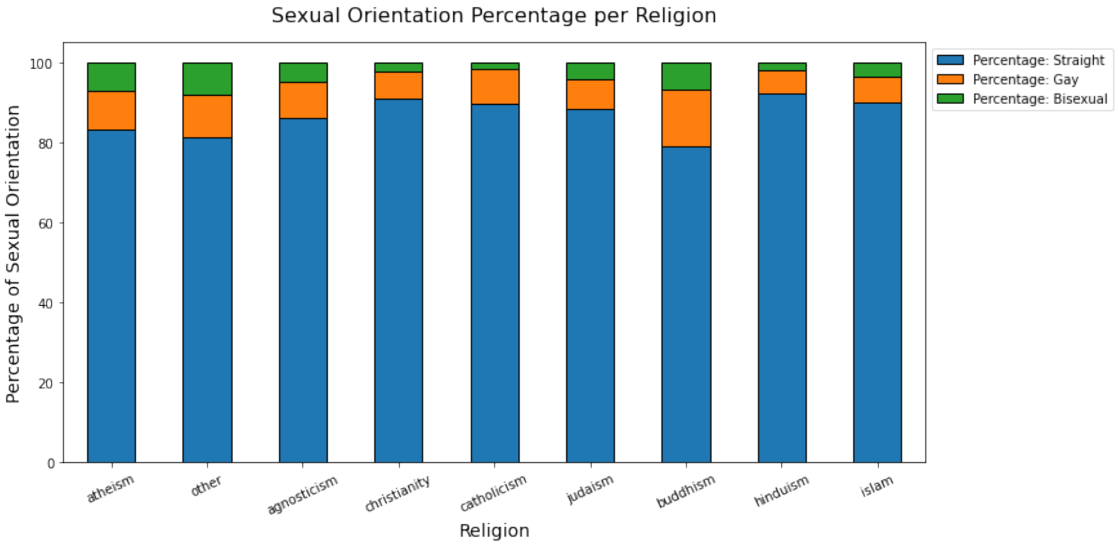
a. Scatter plot shows lower number of OKCupid users with bisexual orientation in Catolicism, Judaism and Budhism religion group. The total number of islamic and hinduistic users is generally low and therefore also the scatter plot shows very bright spots for Gay and Bisexual Category for those two religions.

b. Bisexual people are mostly linked to atheism, other or agnosticism religion.



c. The major sexual orientation for every religion is “straight” one. Christianity and Catholicism generally shows low number of bisexual population.

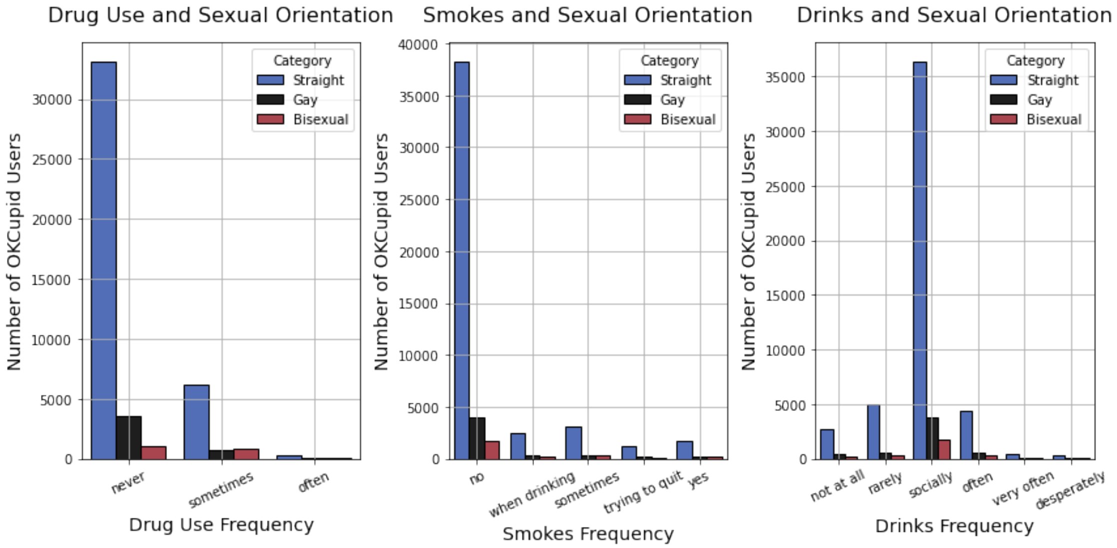
d. Atheism, Other and Agnosticism are the most frequently selected religions in the data set.



d. Highest percentage of the gay population can be identified for the buddhism religion.

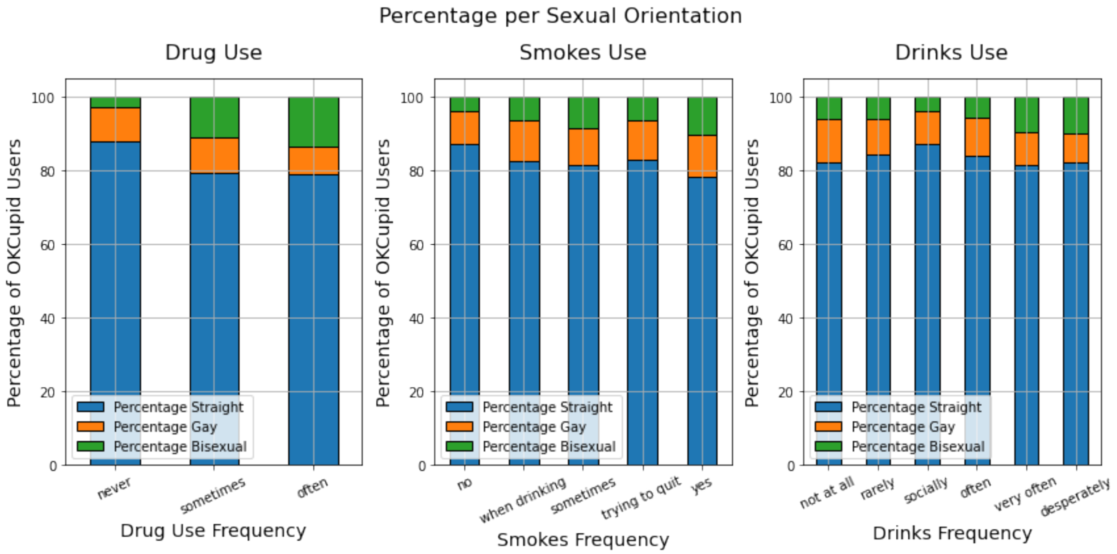
e. Islam, Catholicism, Christianity and Hinduism have the highest percentage of heterosexuals.

## 2.6 Smokes, Drugs, Drinks Use and Sexual Orientation



a. Most of OKCupid users never use drugs and do not smoke at all.

b. Majority of OKCUpid users drink alcoholic beverages socially.



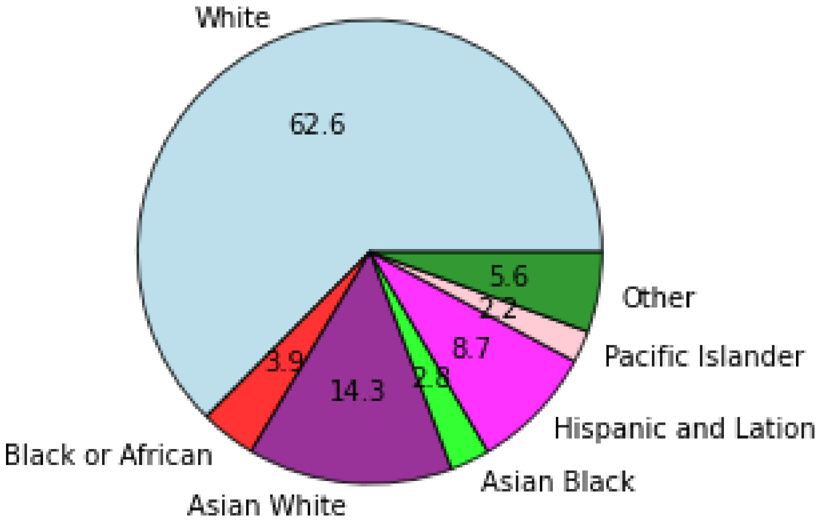
c. Bisexual population from the data set is more likely to use drugs. Amount of bisexual people that never use drugs is much lower.

d. Percentage of gay population from dataset is almost equal for every category of smoking.

e. Bisexual people have the largest percentage value in “very often” and “desperately” category of drinks.

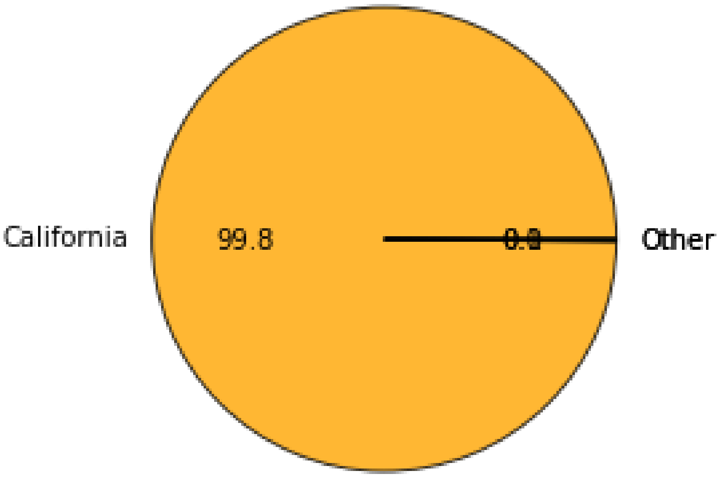
f. Gay percentage for all drugs, smokes and drinks category is almost similar.

### 2.7 Ethnicity percentage in the dataset



a. The majority of persons in data app have a “white” ethnicity. This enthicity group makes up of 63% of all profiles. Secondly the White Asian (14%) and Hispanic (8%) populatioin are also very frequent.

### 2.8 States and their percentage in the dataset

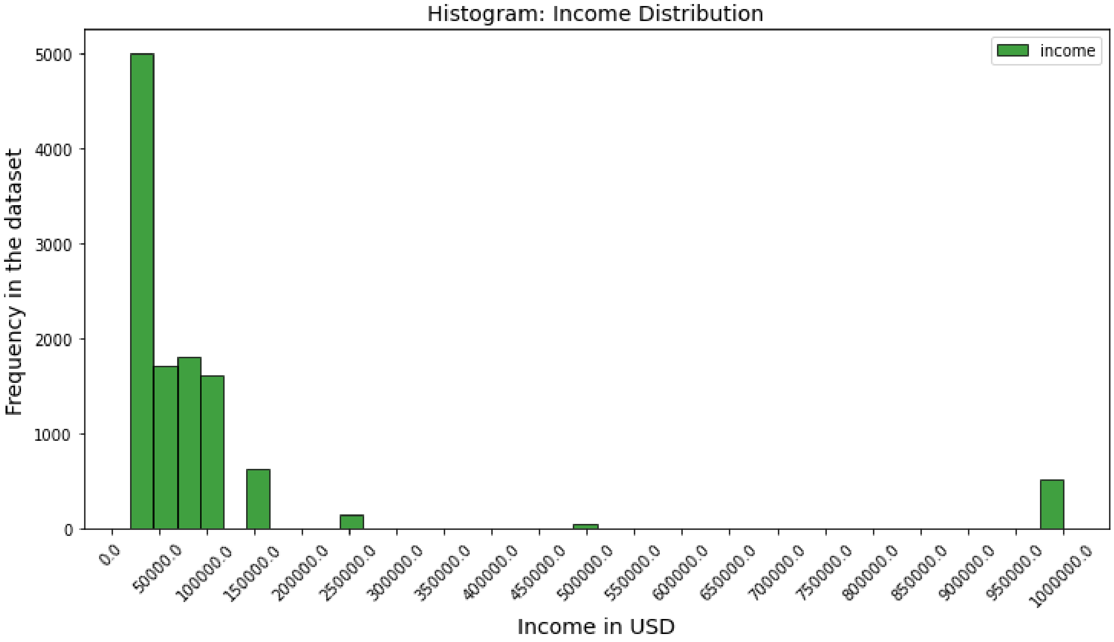


a. 99,8% of OKCupid Users are located in California.

### 2.9 Income distribution in state California

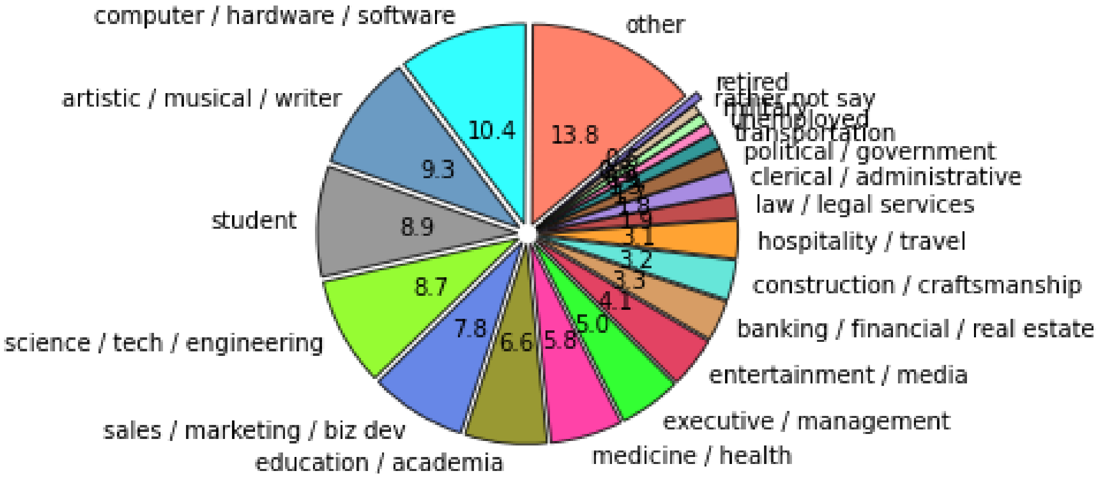
Majority is OKCupid users come from State California and therefore the income distribution is plotted only based on data points of all californian people in the data set.

The value -1 has the highest occurence in the "income" column. Dataframe row with income < 0 have been dropped before histogram plot.

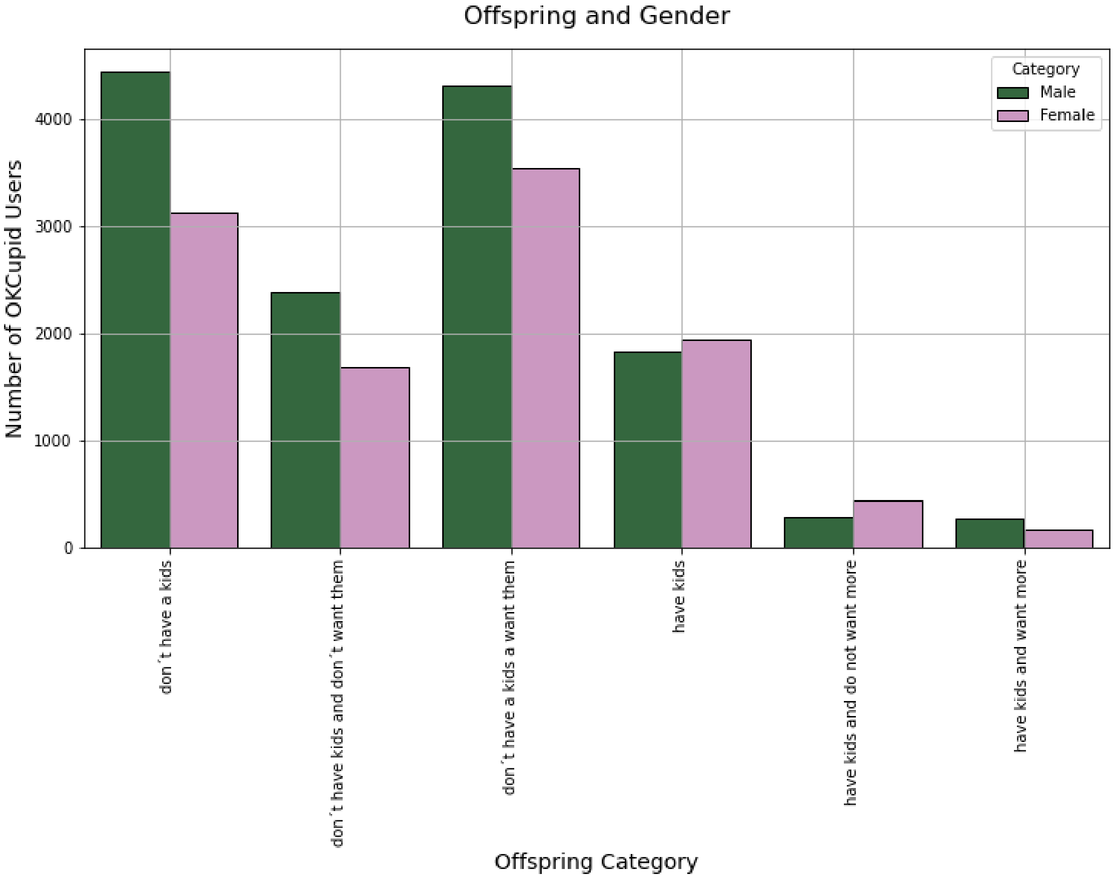


a. The most common income values are below 50 000 USD, the income category nr. 1 is the most typical for majority of users that have specified their income.

### 2.10 Job Categories in state California



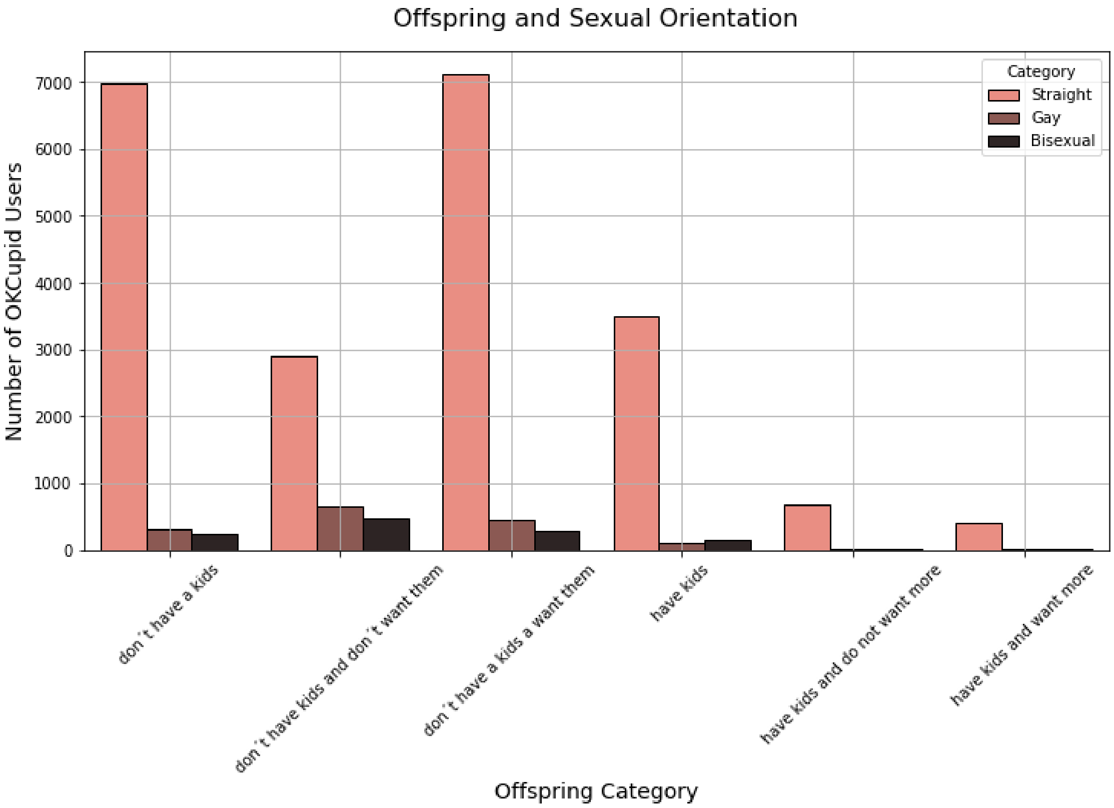
### 2.11 Offspring and Gender



a. Majority of OKCupid users do not have any child, regardless of sex.

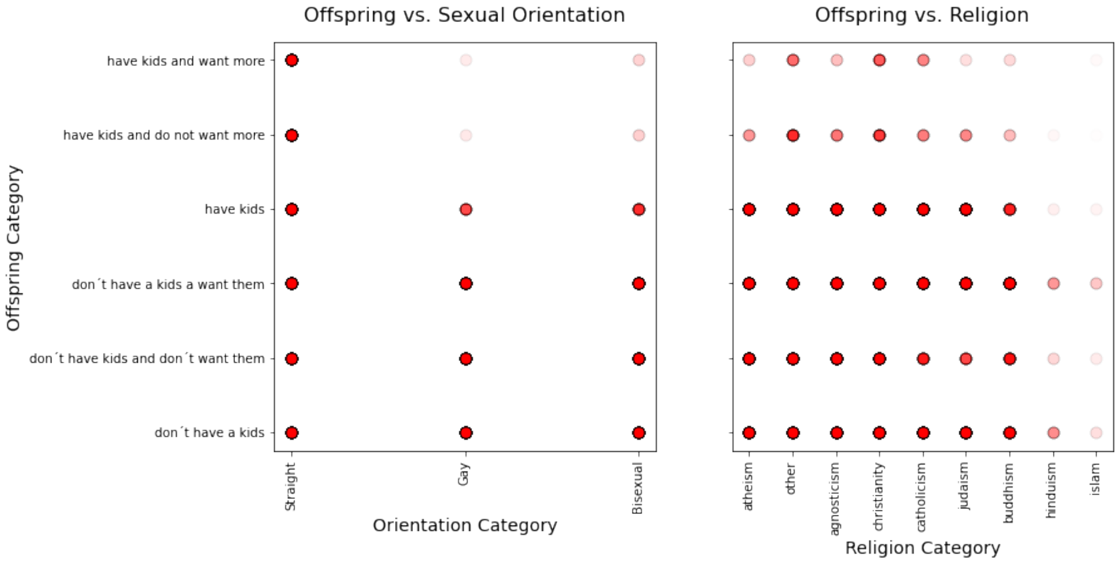
b. Amount of male and female population that have selected option “have kids” is almost equal. Category “have kids and want more/do not want more” contains very low number of responses.

## 2.12 Offspring, Sexual orientation and Religion



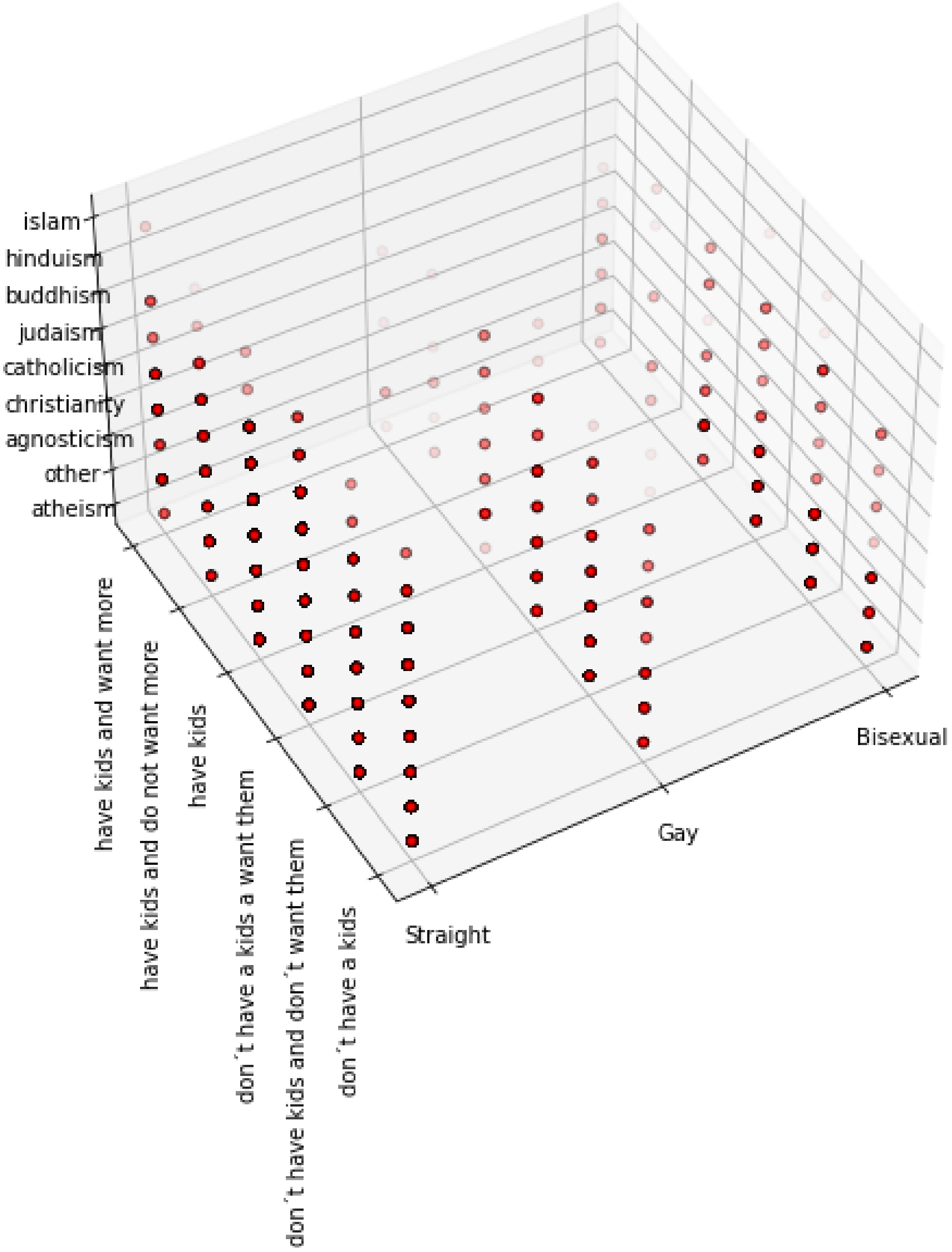
a. Most of gay or bisexual population from the data set do not have kids, but minor portion of them still have kids.

b. Most of the OKCupid users declared that they do not have kids and almost 7000 of users do not want children at all.



c. 2D Scatter plot does not clearly confirm presumption that the gay and bisexual people don´t have childern. There are still some users that have kids and have gay or bisexual orientation.

d. Offspring category “have kids and want more” or “have kids and do not want more” is less common for religion types other than christianity, catholicism and other.



e. 3D scatter plot indicates very low amount of bisexual people with kids that profess catholicism religion.

f. Bisexual people are mostly linked to atheism, other or agnosticism religion.

g. Gay people in the most cases don´t have kids, but some of them are still motivated to have them.

h. Gay people with kids mostly profess “other” religion.

# 3 Machine Learning

## 3.1 K-Nearest Neighbors algoritm

### 3.1.1 Normalization

Data has to be normalized in order to make every datapoint have the same scale so each feature is equally important.

1. min-max normaliztion is the most common way to normalize data -> data are transformed into a decimal between 0 and 1. This method does not handle outliers, but all features will have the exact same scale.
2. Z-Score normalization is a strategy of normalizing data that avoids outlier issue, but does not produced normalized data with the exact same scale.

### 3.1.2 Strategies to analyze predictive power of classification algorithm

A. Accuracy - measures how many classifications the algorithm got correct out of every classification it made.

Accuracy = True Positives + True Negatives / True Positives + True Negatives + False Positives + False Negatives

B. Recall - measures the percentage of relevant items the algorithm was able to successfully find.

Recall = True Positive / True Positive + False Negative

C. Precision - measures the percentage of items the algorithm found that were actually relevant.

Precision = True Positive / True Positive + False Positive

D. F1 Score - combination of accuracy, recall and precision.

F1 Score = 2 * Precision * Recall / (Precision + Recall)

## 3.2 Can we predict zodiac sign ?

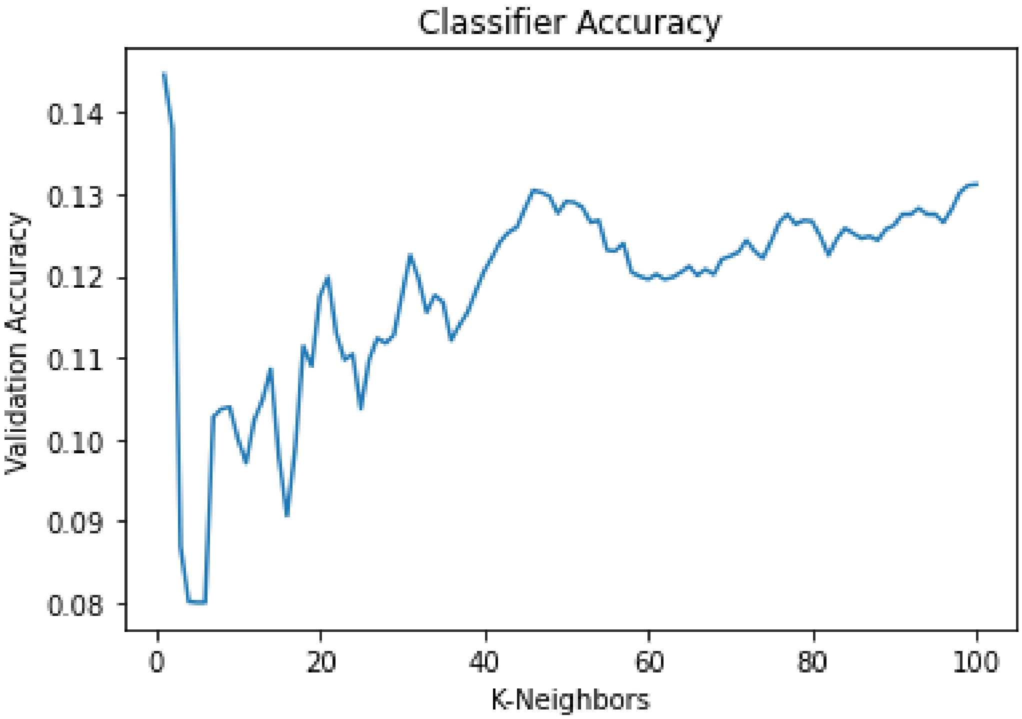
Feature data are scaled from 0 to 1. Feature data are:

```
drugs code,
drinks code,
smokes code,
length of all essays,
average word length in the essay
```

Data are randomly splitted on the 80% train set and 20% test set.

K-Neighbors Classifier has been validated by looping over “k” in range from 0 to 100.

Validation accuracy for each number of neighbors (k) is plotted below.



Classification accuracy is very poor. Prediction of zodiac sign won´t be further studied.

### 3.3 K-Nearest Neighbor Classifier and Regressor

K-Nearest Neighbor algorithm can be used for regression and classification. When k is small, overfitting occurs and the accuracy is relatively low. On the other hand, when k gets too large, underfitting occurs and accuracy starts to

Classifier returns label as a single guess, regressor is returning a number.

By using weighted average, data points that are extremely similar to the input point will have more of a say in the final results of regression.

### 3.4 Can we predict income with age, education, job, sex and city?

There are many data point with income value equal to -1. Those OKCupid users were probably not motivated to share their income. Those entries have been dropped before the training process started.

3 income categories have been established:

```
category 1 : income 0 - 50 000 USD
category 2 : income  50 000 - 100 000 USD
category 3 : income higher than 100 000 USD
```

Number of data points in every income category is as follows:

```
 1    4253
 2    2843
 3     850
 Name: income_code, dtype: int64
```

Features to predict income category are:

```
age
education code
job code
sex code
City code
```

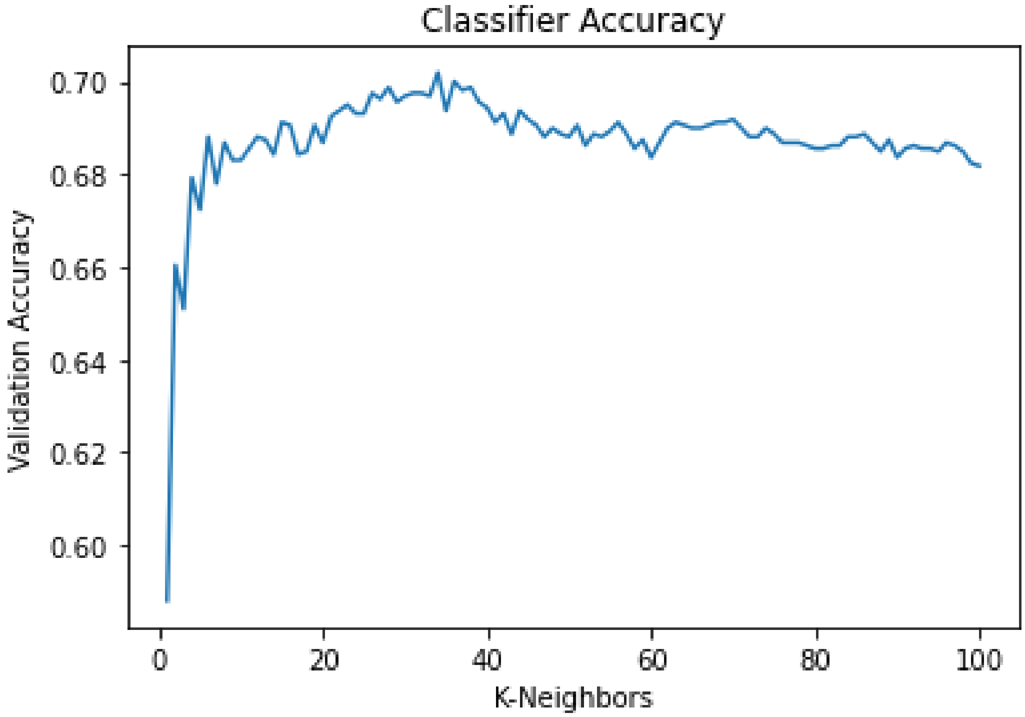
Feature data are splitted on train data and test data. Train features and test features are later scaled to get values 0-1.

Classifier is being validated:

```
N-Neighbors = 1, Score: 0.5880503144654088
N-Neighbors = 2, Score: 0.660377358490566
N-Neighbors = 3, Score: 0.6509433962264151
N-Neighbors = 4, Score: 0.6792452830188679
N-Neighbors = 5, Score: 0.6723270440251572
N-Neighbors = 6, Score: 0.6880503144654088
N-Neighbors = 7, Score: 0.6779874213836478
N-Neighbors = 8, Score: 0.6867924528301886
N-Neighbors = 9, Score: 0.6830188679245283
N-Neighbors = 10, Score: 0.6830188679245283
N-Neighbors = 11, Score: 0.6855345911949685
N-Neighbors = 12, Score: 0.6880503144654088
N-Neighbors = 13, Score: 0.6874213836477987
N-Neighbors = 14, Score: 0.6842767295597484
N-Neighbors = 15, Score: 0.6911949685534591
N-Neighbors = 16, Score: 0.690566037735849
N-Neighbors = 17, Score: 0.6842767295597484
N-Neighbors = 18, Score: 0.6849056603773584
N-Neighbors = 19, Score: 0.690566037735849
N-Neighbors = 20, Score: 0.6867924528301886
N-Neighbors = 21, Score: 0.6924528301886792
N-Neighbors = 22, Score: 0.6937106918238993
N-Neighbors = 23, Score: 0.6949685534591195
N-Neighbors = 24, Score: 0.6930817610062893
N-Neighbors = 25, Score: 0.6930817610062893
N-Neighbors = 26, Score: 0.6974842767295597
N-Neighbors = 27, Score: 0.6962264150943396
N-Neighbors = 28, Score: 0.6987421383647798
N-Neighbors = 29, Score: 0.6955974842767295
N-Neighbors = 30, Score: 0.6968553459119496
N-Neighbors = 31, Score: 0.6974842767295597
N-Neighbors = 32, Score: 0.6974842767295597
```

```
N-Neighbors = 33, Score: 0.6968553459119496
N-Neighbors = 34, Score: 0.7018867924528301
N-Neighbors = 35, Score: 0.6937106918238993
N-Neighbors = 36, Score: 0.7
N-Neighbors = 37, Score: 0.6981132075471698
N-Neighbors = 38, Score: 0.6987421383647798
N-Neighbors = 39, Score: 0.6955974842767295
N-Neighbors = 40, Score: 0.6943396226415094
N-Neighbors = 41, Score: 0.6911949685534591
N-Neighbors = 42, Score: 0.6930817610062893
N-Neighbors = 43, Score: 0.6886792452830188
N-Neighbors = 44, Score: 0.6937106918238993
N-Neighbors = 45, Score: 0.6918238993710691
N-Neighbors = 46, Score: 0.690566037735849
N-Neighbors = 47, Score: 0.6880503144654088
N-Neighbors = 48, Score: 0.689937106918239
N-Neighbors = 49, Score: 0.6886792452830188
N-Neighbors = 50, Score: 0.6880503144654088
N-Neighbors = 51, Score: 0.690566037735849
N-Neighbors = 52, Score: 0.6861635220125786
N-Neighbors = 53, Score: 0.6886792452830188
N-Neighbors = 54, Score: 0.6880503144654088
N-Neighbors = 55, Score: 0.6893081761006289
N-Neighbors = 56, Score: 0.6911949685534591
N-Neighbors = 57, Score: 0.6886792452830188
N-Neighbors = 58, Score: 0.6855345911949685
N-Neighbors = 59, Score: 0.6874213836477987
N-Neighbors = 60, Score: 0.6836477987421383
N-Neighbors = 61, Score: 0.6867924528301886
N-Neighbors = 62, Score: 0.689937106918239
N-Neighbors = 63, Score: 0.6911949685534591
N-Neighbors = 64, Score: 0.690566037735849
N-Neighbors = 65, Score: 0.689937106918239
N-Neighbors = 66, Score: 0.689937106918239
N-Neighbors = 67, Score: 0.690566037735849
N-Neighbors = 68, Score: 0.6911949685534591
N-Neighbors = 69, Score: 0.6911949685534591
N-Neighbors = 70, Score: 0.6918238993710691
N-Neighbors = 71, Score: 0.689937106918239
N-Neighbors = 72, Score: 0.6880503144654088
N-Neighbors = 73, Score: 0.6880503144654088
N-Neighbors = 74, Score: 0.689937106918239
N-Neighbors = 75, Score: 0.6886792452830188
N-Neighbors = 76, Score: 0.6867924528301886
N-Neighbors = 77, Score: 0.6867924528301886
N-Neighbors = 78, Score: 0.6867924528301886
N-Neighbors = 79, Score: 0.6861635220125786
N-Neighbors = 80, Score: 0.6855345911949685
```

```
N-Neighbors = 81, Score: 0.6855345911949685
N-Neighbors = 82, Score: 0.6861635220125786
N-Neighbors = 83, Score: 0.6861635220125786
N-Neighbors = 84, Score: 0.6880503144654088
N-Neighbors = 85, Score: 0.6880503144654088
N-Neighbors = 86, Score: 0.6886792452830188
N-Neighbors = 87, Score: 0.6867924528301886
N-Neighbors = 88, Score: 0.6849056603773584
N-Neighbors = 89, Score: 0.6874213836477987
N-Neighbors = 90, Score: 0.6836477987421383
N-Neighbors = 91, Score: 0.6855345911949685
N-Neighbors = 92, Score: 0.6861635220125786
N-Neighbors = 93, Score: 0.6855345911949685
N-Neighbors = 94, Score: 0.6855345911949685
N-Neighbors = 95, Score: 0.6849056603773584
N-Neighbors = 96, Score: 0.6867924528301886
N-Neighbors = 97, Score: 0.6861635220125786
N-Neighbors = 98, Score: 0.6849056603773584
N-Neighbors = 99, Score: 0.6823899371069182
N-Neighbors = 100, Score: 0.6817610062893081
```



K-Nearest Neighbor is able to classify the income category with 70 % accuracy, for K = 34.

It can be used as K-Neighbor Regressor in the next step to predict the income value by using of weigted average.

### 3.4.1 Prediction

Guess 1:

```
Age = 46
Education = PhD
job = Computer / software
Sex = Man
City = San Francisco
```

Guess 2:

```
Age = 27
Education = Elementary
Job = student
Sex = Women
City = Berkeley
```

Guess 3:

```
Age = 35
Education = Post secondary
Job = law services - 15
Sex = Man
City = Other in California
```

```
Predicted income values are following: Guess 1: 209308.7 USD, Guess 2: 115395.8
USD, Guess 3: 123074.3 USD
```

### 3.4.2 User defined input to predict

```
Enter Age (18-69):23
Enter Education ID:
                        elementary: 1
                        secondary: 2
                        post-secondary: 3
                        PhD: 4
                        3
Enter Job ID:
                        transportation : 0
                        hospitality / travel : 1
                        student : 2
                        artistic / musical / writer : 3
                        computer / hardware / software : 4
                        banking / financial / real estate : 5
                        entertainment / media : 6
                        sales / marketing / biz dev : 7
                        medicine / health : 8
                        science / tech / engineering : 9
                        executive / management : 10
                        education / academia : 11
                        clerical / administrative : 12
                        construction / craftsmanship : 13
                        political / government : 14
                        law / legal services : 15
                        military  : 16
                        unemployed : 17
                        retired : 18
                        rather not say : 19
                        other :20…
                        8
Enter Sex:
                         Male : 0
                        Female : 1
                        1
Enter City ID:
                        san francisco : 0
                        oakland : 1
                        berkeley : 2
                        san mateo : 3
                        palo alto : 4
                        other : 5
                        0
Predicted income 40000.0 :
```

### 3.5 Random Decision Forest

Random forests are used to avoid overfitting. By aggregating the classification of multiple trees, having overfitted trees in a random forest is less impactful.

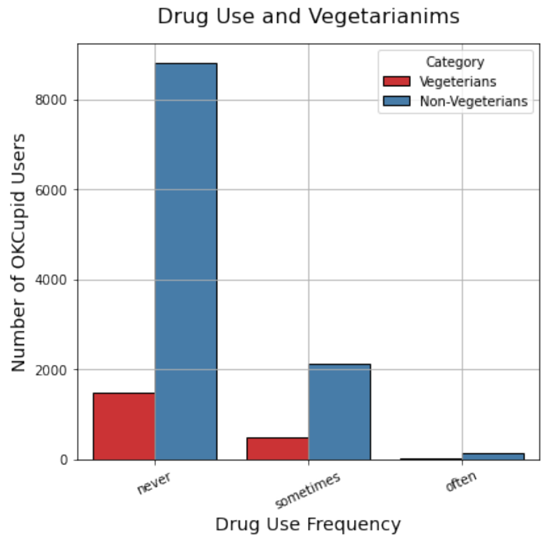
Every decision tree in a random forest is created by using a different subset of data points from the training set. Those data points are chosen at random with replacement, which means a single data point can be chosen more than once. This process is known as bagging.

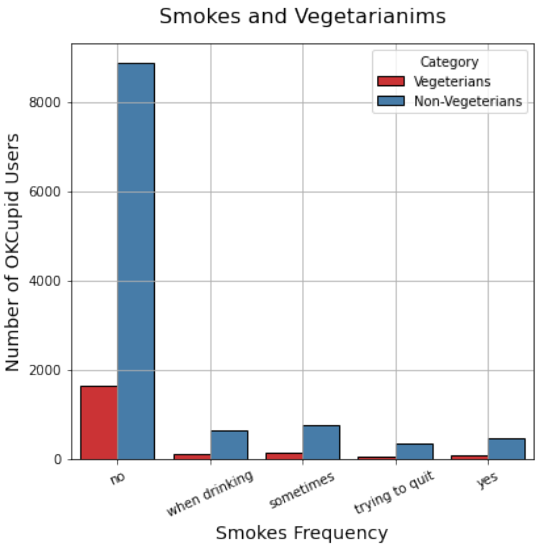
### 3.6 Can we predict vegetarianism with body type, smokes, drugs and relationship to pets?

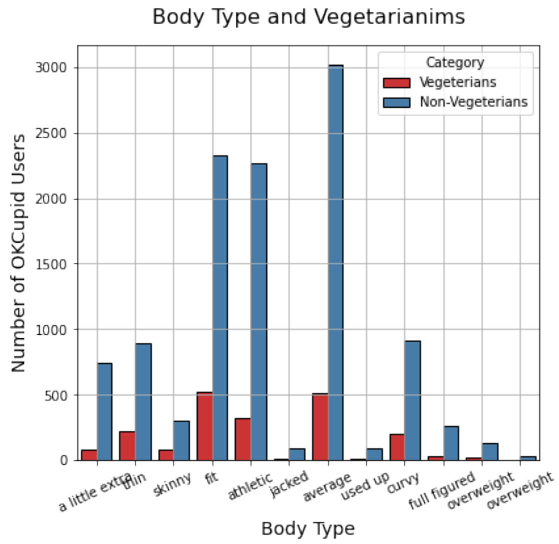
For prediction of vegetarianism a random decision forest classifier has been chosen.

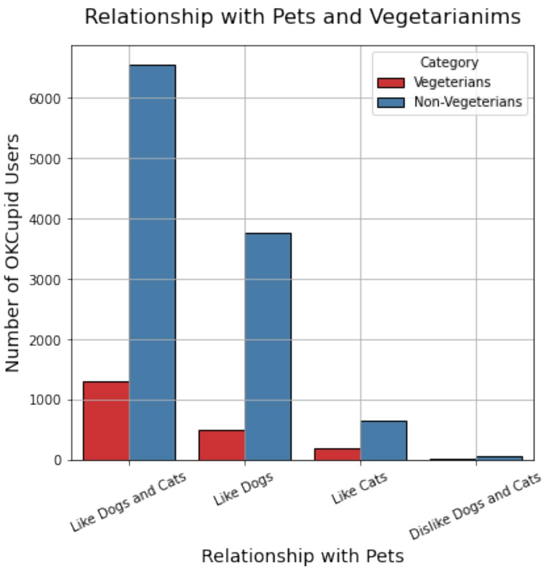
All data points that have vegetarianism or vegan diet have been labeled by value 1 in “veg_status” column. All other rows have value 0 assigned in “veg_status” column.

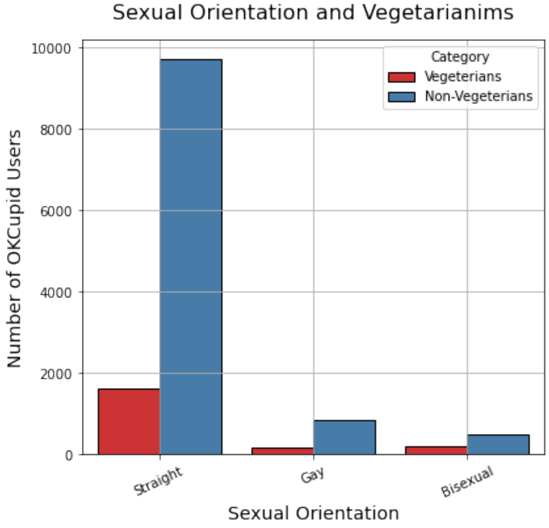
Following graphs are showing number of vegetarians and non-vegetarians in relationship to different parameters like e.g. drugs, smokes, attitude to pets, sexual orientation, body type and religion.

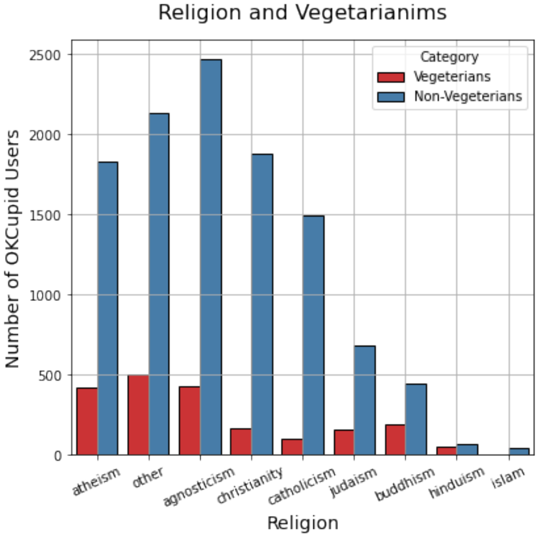

Drug Use and Vegetarianims


Smokes and Vegetarianims


Body Type and Vegetarianims


Relationship with Pets and Vegetarianims


Sexual Orientation and Vegetarianims


Religion and Vegetarianims

a. Vegarian / Vegan people in the data set usually do not smoke and never use drugs. They have also positive attitude to pets, they usually like both dogs and cats.

b. They have mostly straight sexual orientation, average or fit body type.

c. They also usually profess 3 the most popular religion types in the data set.

### 3.6.1 Imbalance Data Handling

Data set is obviously imbalanced. Imbalanced data is a common problem in data science. Having an imbalanced dataset decreases the sensitivity of the model towards minority classes. Classifier usually predicts mostly the same value which is resulting in very high accuracy of classification. The traditional approach of classification and model accuracy calculation is therefore not useful in the case of the imbalanced dataset.

Most machine learning algorithms work best when the number of samples in each class are about equal. This is because most algorithms are designed to maximize accuracy and reduce error.

While in every machine learning problem, it's a good rule of thumb to try a variety of algorithms, it can be especially beneficial with imbalanced datasets. Decision trees frequently perform well on imbalanced data. They work by learning a hierarchy of if/else questions and this can force both classes to be addressed.

The other metrics such as precision is the measure of how accurate the classifier's prediction of a specific class and recall is the measure of the classifier's ability to identify a class. For an imbalanced class dataset F1 score is a more appropriate metric.

There are many techniques how to deal with imbalanced data:

a. Resample

This technique is used to upsample or downsample the minority or majority class. When we are using an imbalanced dataset, we can oversample the minority class using replacement. This technique is called oversampling. Similarly, we can randomly delete rows from the majority class to match them with the minority class which is called undersampling.

b. Equal random sampling

### 3.6.2 Data Resampling

Data set has been resampled and minority part has been upscaled to match majority class.

Number of values after upsampling are as follows:

```
0    11049
1    11049
Name: veg_status, dtype: int64
```

Features used to train a classifier are:

```
smokes_code,
drugs code
```

```
body type code
orientation code
religion code
sex code
city code
ethnicity code
```

Random train test split has been performed. 20% of data is used for testing phase of ML algorithm.

Random Forest Classifier with 100 decision trees has been defined.

```
 RandomForestClassifier(random_state=42)
```

Decission Tree Forest classifier has been trained and following score parameters have been achieved:

```
Recall score: 0.838
Accuracy score: 0.786
Precision score: 0.7655
F1 score: 0.8001
```

Number of true negatives, true positives, false negatives and false positives are as follows:

```
True Negative: 1581
True Positive: 1893
False Negative: 366
False Positive: 580
```

Score Metrics:

```
Test score: 0.785972850678733
```

```
Train score: 0.8252064713202851
```

The reported train and test score is very high, but can we really on this? Is really the algorithm performance so good?

The aswer is NO .

The problem that occured is called data leakage. Upscaling applied on the data set before splitting to train set and test set leads to repeating the labels and the algorithm has a perfect Recall Score of 84 %. It is too optimistic results and is supposed to be totally wrong. Due to data leakage from train set to test set we have exactly same data in the train set and test set.

It is always needed to apply resampling only on the train set data to avoid duplicities in the train and test set.

### 3.6.3 Resampling applied only on the training set

Features used to train a random decission forrest classifier are:

```
smokes code
drugs code
body type code
religion code
sex code
```

```
pets code
```

Number of items for both classes in the train set is as follows:

```
 0    8824
 1    1625
 Name: veg_status, dtype: int64
```

After resampling the number of items for both classes is equal. Sklearn resample module randomly replicate the samples from the minority class.

```
 0    8824
 1    8824
 Name: veg_status, dtype: int64

 RandomForestClassifier(random_state=42)
```

Score metrics:

```
Recall score: 0.5129
Accuracy score: 0.6506
Precision score: 0.2156
F1 score: 0.3036
```

Recall score get worse to 51 % and F1 score is only 30%. Random Forest Classifier score is very poor! Upsampling of minority class in the train set does not lead to sufficient score value.

### 3.6.4 Equal random sampling

```
Minimum number of items from for Veterianism class is: 2004
```

```
2004 of samples have been randomly selected from the majority class.
```

Number of samples for both classes is equal.

```
 0    2004
 1    2004
 Name: veg_status, dtype: int64
```

Features used to train a random decission forrest classifier are:

```
smokes code
drugs code
body type code
religion code
sex code
pets code
```

Score metrics:

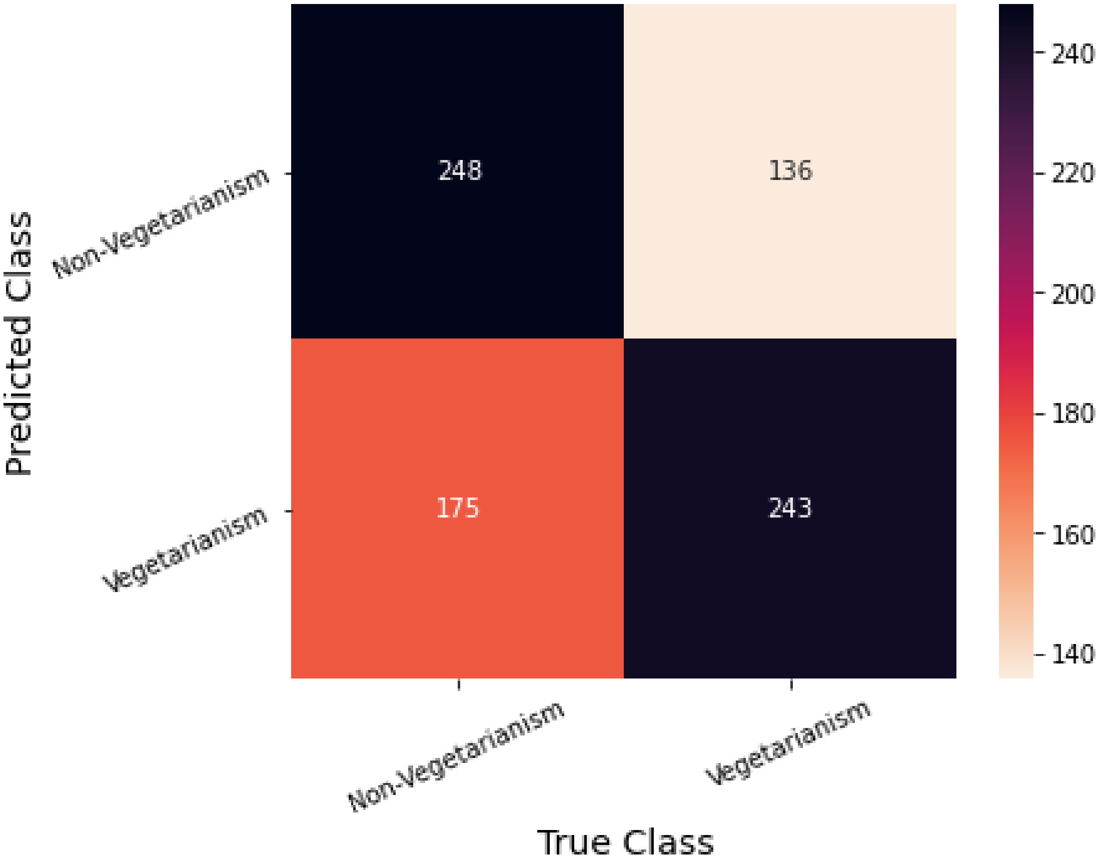
```
Recall score: 0.5813
Accuracy score: 0.6122
```

```
Precision score: 0.6412
F1 score: 0.6098
```

F1 score achieves value of 61 %. Classifier score got better in comparison to upsampling approach of minority class.

```
True Negative: 248
True Positive: 243
False Negative: 175
False Positive: 136
```

Confusion matrix can be visualised like a heat map:



Number of true positive and true negative is higher than number of false negative/positives. Classifier score is still very low and not acceptable.

### 3.7 Support Vector Machines

SVM is powerful supervised machine learning model used for classification. An SVM makes classifications by defining a decision boundary and then seeing what side of the boundary an unclassified point falls on. Decision boundaries exist even when your data has more than two features. If there are three features, the decision boundary is now a plane rather than a line.

As the number of dimensions grows past 3, it becomes very difficult to visualize these points in space. Nonetheless, SVMs can still find a decision boundary. However, rather than being a separating line, or a separating plane, the decision boundary is called a separating hyperplane.

(Codecademy - Chapter Support Vector Machines)

## 3.8 Can we predict sexual orientation with offspring, religion, drugs, smokes, sex and number of fluently speaking languages?

To predict the sexual orientation of the OKCupid users the ML algorithm called Support Vector Machine has been implemented.

Data are scaled from 0 to 1 before splitting data set to train set and test set.

As seen on the bar charts above the sexual orientation classes unequally occur in the data set. Therefore the minimum number of items for each class is used as parameter for the random sample selection from dataset.

```
Minimum number of items from all 3 sexual orientation classes is: 710
```

```
For each sexual orientation class 710 random samples have been selected.
```

Number of samples for each class is equal.

```
 0    710
 2    710
 1    710
 Name: orientation_code, dtype: int64
```

Scaling transformation avoids the problem of having only some features influence the algorithm´s optimization process and helps make the computations exact, smooth and fast.

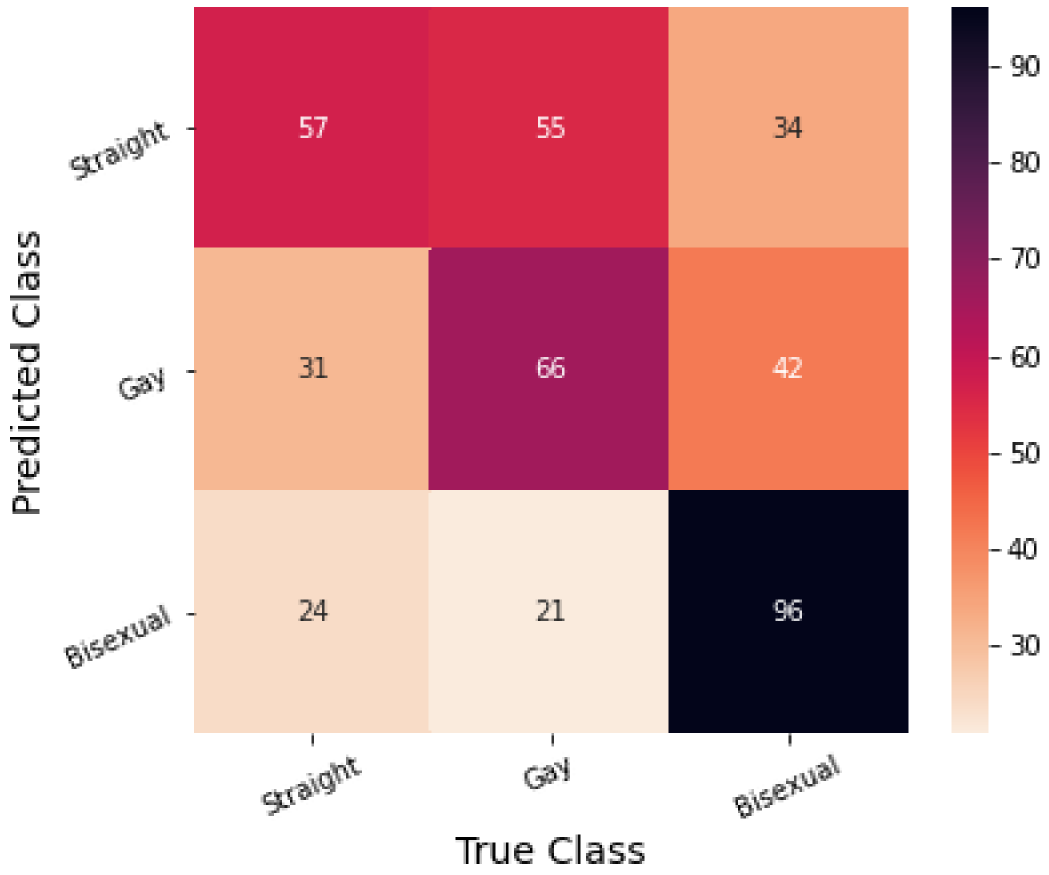
Cross validation is started to check the algorithm score. The “cv” parameter specifies the cross-validation strategy. In this case the integer number of folds was set to 10. List of cross-validation accuracies for each fold is returned.

```
Cross validation accuracies for K-folds: [0.60233918 0.52046784 0.57309942
0.53216374 0.55294118 0.55882353
 0.48823529 0.61764706 0.57647059 0.64117647]
```

```
Mean Cross-validation accuracy: 0.566
```

```
Test score: 0.514
```

Confusion matrix visualized like a heat map:



Classifier is able to predict bisexual orientation with higher accuracy than in case of “straight” or “gay” sexual orientaion. The reason could be that the bisexual orientaion is determined to be more typical for female sex and offspring code or religion code also show typical values for this bisexual orientation class.

Classifier returned in 55 cases a “straight” sexual orientation while the true class is “gay” orientation.

Classifier is also not very successful in distinguishing a bisexual person from a gay person.

### 3.8.1 Classifier Validation

Grid Search CV is an exhaustive search over specified parameter values. It implements “fit” and “score” method. Cross-validation splitting strategy can be applied by specifying cv parameter. It determines the number of folds in K-Fold Cross Validation.

SVM use different types of kernel method to seperate the data points:

A. linear - decision boundary is linear

B. polynomial - used for data that are non-linerly separable.

C. radial basis funcion - Gamma and c parameters can be tuned to get model more or less sensitive to training data. A higher gamma, say 100, will put more importance on the training data and could result in overfitting. Conversely, A lower gamma like 0.01 makes the points in the training data less relevant and can result in underfitting.

Classifier is tuned for following gamma and c paremters:

```
C-parameter range: [1.e-03 1.e-02 1.e-01 1.e+00 1.e+01 1.e+02 1.e+03]
Gamma-parameter range: [1.e-03 1.e-02 1.e-01 1.e+00 1.e+01 1.e+02]
--------------------------------------------------------------------------------
--------------------
Best parameter:{'C': 1.0, 'gamma': 1.0, 'kernel': 'rbf'}
Best score 0.562
```

# 4 Conclusion

1. Linear machine learning models (linear and logistic regression) have not been chosen to predict or classify the data. Almost no linear relationship between features and values have been recognized for this particular data set.

2. Non-linear ML algorithms like K-Neighbors Classifier/Regressor, Random Decision Forest Classifier and Support Vector Machines have been implemented to classify income category and predict income value, classify the vegetarianism and classify the sexual orientation respectively.

3. It was hard to deal with imbalanced data set to predict vegetarianism or sexual orientation. Different approaches have been tested and model score have been analyzed. Best accuracy results were achieved for equally balanced data set, that has been gained by random selection of samples from majority class.

4. Quality of data used for training of machine learning model is essential.

5. Cross-validation approach using K-folds has been implemented to validate the accuracy of SVM classifier and algorithm parameters have been tuned by GridSearchCV function

6. Income category has been classified with 70% accuracy by usage of K-Neighbors Classifier. K-Neighbors Regression has been later used to predict the income value for specific input.

7. Vegetarianism classification achieved F1 Score of 61%. Classifier score is not enough high.

8. Sexual classification achieved best score of 56% after SVM parameter tuning. Classifier score is not enough high.

9. ML models have to be improved in order to provide better predictions or classification score. Method how to improve the models have to be studied further.