

# k-Anonymity Library Demo with k=3

```
In [1]: import warnings
warnings.filterwarnings('ignore')
```

```
In [2]: import kAnonymityLib as daio_dpt
import pandas as pd
dai_anonymization = daio_dpt.kAnonymity()
print(dai_anonymization)
```

k-Anonymity Class Library with k=3

```
In [3]: names = ['age',
'workclass',
'fnlwt',
'education',
'education-num',
'marital-status',
'occupation',
'relationship',
'race',
'sex',
'capital-gain',
'capital-loss',
'hours-per-week',
'native-country',
'income']
dai_anonymization.set_headers(names)
```

```
In [4]: dai_anonymization.read_datafile("adult-all.txt")
```

```
In [5]: df = dai_anonymization.dataframe
```

```
In [6]: age_range = lambda age: ("<= 20" if age <= 20
else "21 - 30" if age <= 30
else "31 - 40" if age <= 40
else "41 - 50" if age <= 50
else "51 - 60" if age <= 60
else "61 - 70" if age <= 70 else "> 70")))))))
```

```
In [7]: df["age"] = df.apply(lambda x: age_range(x.age), axis=1)
df["workclass"] = df.apply(lambda x: x.workclass.replace(" ", ""), axis=1)
df["workclass"] = df.apply(lambda x: "Others" if x.workclass=="?" else x.workclass, axis=1)
df["education"] = df.apply(lambda x: x.education.replace(" ", ""), axis=1)
df["marital-status"] = df.apply(lambda x: x["marital-status"].replace(" ", ""), axis=1)
df["occupation"] = df.apply(lambda x: x.occupation.replace(" ", ""), axis=1)
df["occupation"] = df.apply(lambda x: "Others" if x.occupation=="?" else x.occupation, axis=1)
df["relationship"] = df.apply(lambda x: x.relationship.replace(" ", ""), axis=1)
df["race"] = df.apply(lambda x: x.race.replace(" ", ""), axis=1)
df["sex"] = df.apply(lambda x: x.sex.replace(" ", ""), axis=1)
df["native-country"] = df.apply(lambda x: x["native-country"].replace(" ", ""), axis=1)
df["income"] = df.apply(lambda x: x.income.replace(" ", ""), axis=1)
```

```
In [8]: categorical = ['age',
    'workclass',
    'education',
    'marital-status',
    'occupation',
    'relationship',
    'race',
    'sex',
    'native-country',
    'income']
```

```
In [9]: feature_columns = ['race', 'sex', 'age']
```

```
In [10]: dai_anonymization.set_categorical(categorical)
```

```
In [11]: dai_anonymization.set_feature_columns(feature_columns)
```

```
In [12]: df.head()
```

Out[12]:

	age	workclass	fnlwgt	education	education-num	marital-status	occupation	relationship	race	sex
<b>0</b>	31 - 40	State-gov	77516	Bachelors	13	Never-married	State-gov	Not-in-family	White	Male
<b>1</b>	41 - 50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Self-emp-not-inc	Husband	White	Male
<b>2</b>	31 - 40	Private	215646	HS-grad	9	Divorced	Private	Not-in-family	White	Male
<b>3</b>	51 - 60	Private	234721	11th	7	Married-civ-spouse	Private	Husband	Black	Male
<b>4</b>	21 - 30	Private	338409	Bachelors	13	Married-civ-spouse	Private	Wife	Black	Female

```
In [13]: dd = pd.Series({c: df[c].unique() for c in df})
print(dd)
```

```

age          ['31 - 40', '41 - 50', '51 - 60', '21 - 30', '...
workclass    ['State-gov', 'Self-emp-not-inc', 'Private', '...
fnlwtgt      [77516, 83311, 215646, 234721, 338409, 284582,...
education    ['Bachelors', 'HS-grad', '11th', 'Masters', '9...
education-num [13, 9, 7, 14, 5, 10, 12, 11, 4, 16, 15, 3, 6,...
marital-status ['Never-married', 'Married-civ-spouse', 'Divor...
occupation   ['State-gov', 'Self-emp-not-inc', 'Private', '...
relationship  ['Not-in-family', 'Husband', 'Wife', 'Own-chil...
race          ['White', 'Black', 'Asian-Pac-Islander', 'Amer...
sex           ['Male', 'Female']
Categories (2, object): ['F...
capital-gain  [2174, 0, 14084, 5178, 5013, 2407, 14344, 1502...
capital-loss  [0, 2042, 1408, 1902, 1573, 1887, 1719, 1762, ...
hours-per-week [40, 13, 16, 45, 50, 80, 30, 35, 60, 20, 52, 4...
native-country ['United-States', 'Cuba', 'Jamaica', 'India', ...
income        ['<=50k', '>50k']
Categories (2, object): ['<=...
dtype: object

```

```

In [14]: # dai_anonymization.partition_dataset()
# dai_anonymization.build_anonymized_dataset()
dai_anonymization.generate_anonymized_dataset()

```

```

In [15]: print(f"population size = {df.age.size}")
results_df = dai_anonymization.results_df
print(f"anonymized dataset size = {results_df.age.size}")
deleted_df = dai_anonymization.removed_df
print(f"deleted dataset size = {deleted_df.age.size}")

```

```

population size = 48842
anonymized dataset size = 48837
deleted dataset size = 5

```

```

In [16]: results_df.groupby(feature_columns).size()

```

```

Out[16]:
race      sex      age
Amer-Indian-Eskimo  Female  21 - 30      51
                  31 - 40      51
                  41 - 50      34
                  51 - 60      24
                  61 - 70       4
                  ...
White      Male     41 - 50     6547
                  51 - 60     3959
                  61 - 70     1720
                  <= 20      1645
                  > 70       517

Length: 66, dtype: int64

```

```

In [17]: deleted_df.groupby(feature_columns).size()

```

```

Out[17]:
race      sex      age
Amer-Indian-Eskimo  Female  > 70      1
Other              Female  61 - 70     2
                  > 70      1
                  Male    > 70      1

dtype: int64

```

```

In [18]: deleted_df.head()

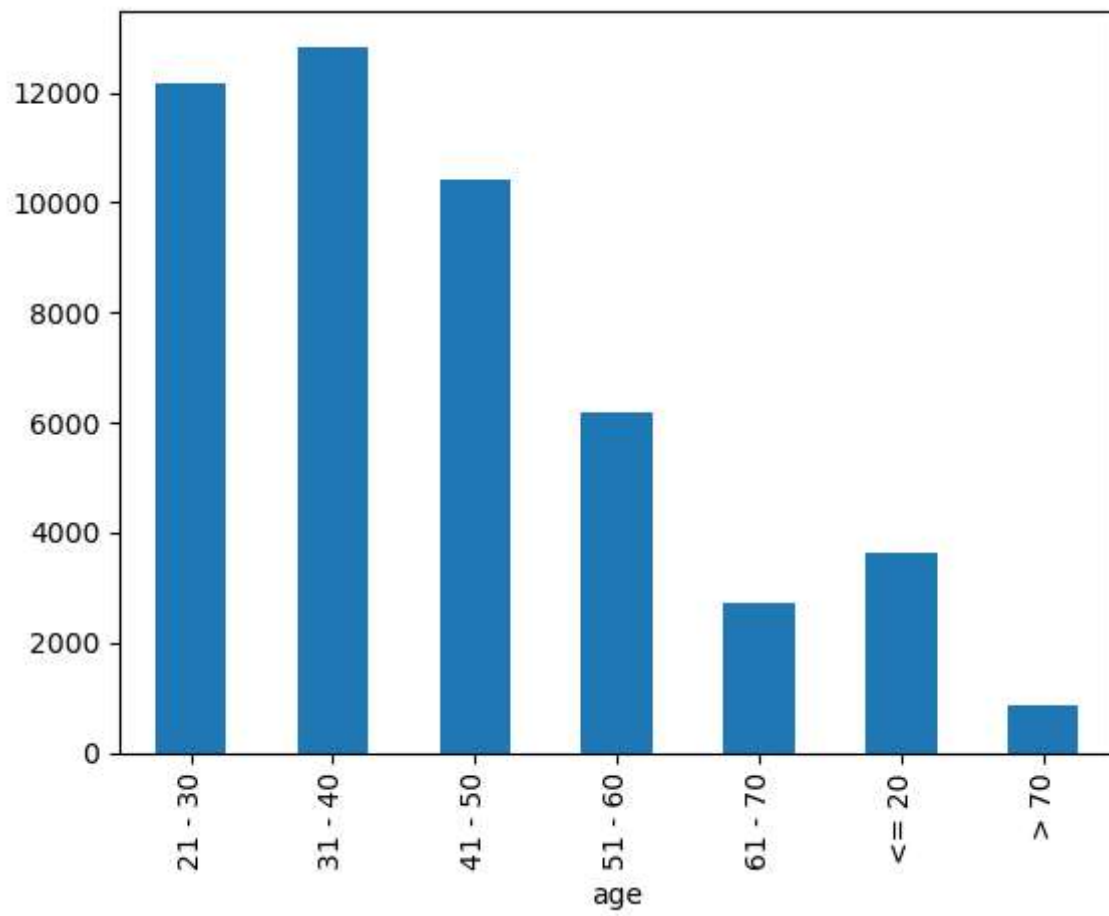
```

Out[18]:

	age	workclass	education	marital-status	occupation	relationship	race	sex	native-country	income
<b>0</b>	> 70	Local-gov	HS-grad	Widowed	Local-gov	Unmarried	Amer-Indian-Eskimo	Female	United-States	<=50k
<b>1</b>	61 - 70	Others	HS-grad	Widowed	Others	Not-in-family	Other	Female	Puerto-Rico	<=50k
<b>2</b>	61 - 70	Private	7th-8th	Separated	Private	Not-in-family	Other	Female	Mexico	<=50k
<b>3</b>	> 70	Others	Bachelors	Widowed	Others	Not-in-family	Other	Female	United-States	<=50k
<b>4</b>	> 70	Private	7th-8th	Married-civ-spouse	Private	Husband	Other	Male	United-States	<=50k

In [19]: `results_df.groupby("age").size().plot.bar()`  
`print(results_df.groupby("age").size())`

```
age
21 - 30    12170
31 - 40    12838
41 - 50    10403
51 - 60     6202
61 - 70     2736
<= 20      3623
> 70         865
dtype: int64
```



```
In [20]: results_df.to_csv("result.csv")
```

```
In [ ]:
```