Privacy Preserving Information Access

Homework 1

Francesco L. De Faveri - ID. 2057069
November 2nd, 2022

Università degli Studi di Padova

Contents

Context

The Heart Disease dataset is important because the data in this table describes the medical conditions of the heart and some blood characteristics of a patient in a hospital.
While they are used by the doctors to see the patient with the higher risk of heart disease, they can also be used by some Machine Learning algorithm during the training phase.

Context

A portion of the dataset is displayed below.

	age	sex	cp	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa	thall	output
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	0
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	0
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	0
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	0

303 rows × 14 columns

Identifiers

Considering the information that can be found in this article "Heartbeat like fingerprints", we can say that in the dataset the **Identifiers** are:

- restecg: Resting electrocardiographic results
- thalach: Maximum heart rate achieved
- oldpeak: ST depression induced by exercise relative to rest
- slp: The slope of the peak exercise ST segment

Quasi-Identifiers

The **Quasi-Identifiers** are:

- age: Age of the patient
- sex: Sex of the patient (0 = female, 1 = male)
- exng: Exercise induced angina (0 = no, 1 = yes)
- caa: Number of major vessels

Sensitive Attributes

We need to say that the majority of the attributes could be seen as sensible information because all of them are about the medical conditions of the patient.
We have found the following **Sensitive** attributes

- cp: Chest Pain (types of angina)
- trtbps: Resting blood pressure (in mm Hg)
- chol: Cholesterol in mg/dl fetched via BMI sensor
- fbs: Fasting blood sugar > 120 mg/dl ($0 =$ false, $1 =$ true)
- thall: Thalassemia rate
- output: Diagnosis of heart disease

Adversaries

Authorized people to access the data:

- Doctors (that works with the patient in the hospital and family doctors)
- Medical staff
- Authorized researchers

Unauthorized people that may be interested in the data:

- Insurance agencies
- Employers without permissions
- Cyber-criminals

As happened in 2021, the cyber-gang LockBit 2.0 posted online a dataset of sensitive information about the ULSS 2-3 in Veneto region.

Snippets of Code: Rounding

Round 'age' and 'chol' values to the closest multiple of a chosen base number

```
[93] def base_round(x, b):
       return (b * round(x/b)).astype('int')

     df = heartdf
     df['round_age'] = base_round(df['age'],4)
     df['round_chol'] = base_round(df['chol'], 6)
     df_concat = pd.concat([heartdf[['age']], df[['round_age']], heartdf[['chol']], df[['round_chol']]], axis=1)
     df_concat
```

	age	round_age	chol	round_chol
0	63	64	233	234
1	37	36	250	252
2	41	40	204	204
3	56	56	236	234
4	57	56	354	354
...	...	...	...	...
298	57	56	241	240
299	45	44	264	264
300	68	68	193	192
301	57	56	131	132
302	57	56	236	234

303 rows × 4 columns

Snippets of Code: pd.cut() vs pd.qcut()

Trade-off between Privacy and Statistical property in grouping people considering the 'chol' attribute.

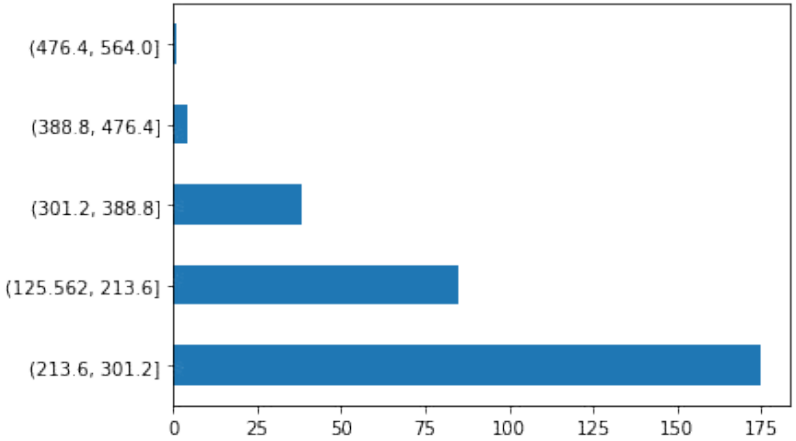

Figure: Same size for bins

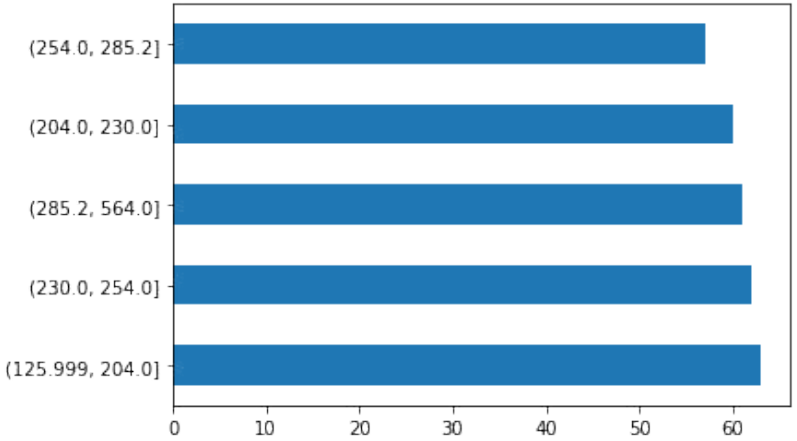

Figure: Different size for bins

Snippets of Code: Record Linkage

Steps:

- Creating a copy of the heartdf, rounding of 'Cholesterol', 'Max heart beat' and 'Resting blood preassure'.
- Creating IDs for both datasets.
- Naively, consider all possible combinations for matching for a finite set of attributes.
- Print remarkable results.

Record Linkage tool kit docs

```
potential_matches = features[features.sum(axis=1) > 1].reset_index()
potential_matches['Score'] = potential_matches.loc[:, 'Kind of arithmia':'Talassemia'].sum(axis=1)
```

```
potential_matches
```

	id	ID Patient	Kind of arithmia	Generation	Max beat	Kind of ST segment	Talassemia	Score
0	426672	477505	0	0	0	1	1	2
1	426672	609573	1	0	0	0	1	2
2	426672	77493	1	1	0	1	1	4
3	58223	520517	1	0	0	0	1	2
4	58223	32620	1	0	0	0	1	2
...	...	...	...	...	...	...	...	...
22781	342137	922331	0	0	0	1	1	2
22782	342137	797980	1	0	0	1	0	2
22783	342137	826848	1	0	0	1	1	3
22784	342137	583662	0	0	0	1	1	2
22785	342137	8525	0	0	0	1	1	2

22786 rows × 8 columns

Snippets of Code: Record Linkage

Example of probable match:

```
heartdf.loc[426672,:]

age          63.0
sex           1.0
cp            3.0
trtbps      145.0
chol        233.0
fbs           1.0
restecg       0.0
thalachh    150.0
exng          0.0
oldpeak       2.3
slp           0.0
caa           0.0
thall         1.0
output        1.0
Name: 426672, dtype: float64
```

Figure: Patient heartdf - id. 426672

```
df_modified.loc[77493,:]

years                        63.0
gender                        1.0
Chest Pain                    3.0
Resting blood preassure     143.0
Cholesterol                 234.0
Fasting blood sugar           1.0
Resting Electrocardiogram     0.0
Maximum heart beat          152.0
Exercise                      0.0
ST depression                 2.3
Slope of ST                   0.0
Major Vessels                 0.0
Talassemia rate               1.0
Diagnosis                     1.0
Name: 77493, dtype: float64
```

Figure: Patient df_modified - ID. 77493

Conclusions

Possible further implementations:

- Optimization of the record linkage: Which attributes to obfuscate? How?
- Optimization of the record linkage: Naive vs Probabilistic/Distance-based.
- Classification of patients with ML algorithms and information loss.