M2 ISL. Sécurité avancée
Protection de la vie privée.

18 décembre 2023

Tous les documents sont autorisés. Épreuve individuelle qui ne peut être réalisée que par vous ! Toute communication est interdite. Toutes les réponses doivent être justifiées. **Sans justification, une réponse est considérée comme fausse.**

Toutes les réponses, même théoriques, doivent être consignées dans un notebook jupyter que vous enverrez à mon adresse mail (`coucho@femto-st.fr`) à la fin de la séance.

1 Etude des données

Dans toute cette partie, on considère le jeu de données Adults déjà vu en TP. On souhaite estimer si une personne se présentant à nous va gagner plus de 50k$ par an à partir des données présentes dans le jeu de données.

La figure 1 donne un extrait de 10 lignes en ne se concentrant que sur les attributs âge, éducation, genre et le salaire (si celui est supérieur ou inférieur à 50k$).

Âge	Éducation	Genre	Salaire
67	Master	Female	<= 50k$
27	HS-grad	Male	<=50k$
48	HS-grad	Female	<=50k$
45	Bachelors	Female	<=50k$
30	HS-grad	Male	<=50k$
39	Master	Male	>50k$
30	Master	Male	>50k$
33	Bachelors	Female	>50k$
43	Bachelors	Female	>50k$
35	Bachelors	Male	>50k$

FIGURE 1 – Extrait de 10 lignes du jeu de données Adults.

2 Apprentissage supervisé sans PVP sur un extrait des données

1. Pourquoi cette section se nomme-t-elle “Apprentissage supervisé. . .” ?
2. Peut-on mettre en place une démarche de régression linéaire sur ces données pour cet apprentissage ? Quels sont les types d’attributs notamment ?
3. Se présente Gladys, une dame de 35 ans, un Master en poche. Mettre en place complètement une démarche d’apprentissage bayésien naïf pour estimer si cette personne a une probabilité plus élevée de gagner plus de 50k$ par an que le contraire.
4. Discuter de la pertinence de l’estimation précédente en regard des probabilités obtenues en dernière étape de la question précédente.

3 Apprentissage pratique sans PVP sur un extrait des données

A partir de cette section, ne vont être retenus que les attributs suivants :
`['workclass', 'education', 'marital-status', 'occupation', 'relationship', 'race', 'sex', 'native-country','salary']`.

5. Quel est le type de chacun des attributs retenus ?
6. Un notebook Jupiter est fourni et synthétisé à la figure 2. Il contient un code mettant en place de l'apprentissage naïf bayésien catégoriel sur le jeu de données. Expliquez chacune des parties (identifiées par *#partie 1*, *#partie 2*...) de ce code.
7. Exécutez ce code. Que constatez-vous en terme de précision d'apprentissage ? Discuter.

4 Assainissement en amont des données personnelles

Dans cette section, chaque personne a la possibilité de nettoyer elle-même ses données.

8. Sur quelle hypothèse de confiance sont basés les algorithmes de traitement des données vérifiant la confidentialité différentielle originale (ou dite centralisée) ?
9. Lorsqu'une personne nettoie elle-même ses données, cette hypothèse est-elle encore nécessaire ? Discuter.

4.1 Le mécanisme $\mathcal{M}_{SUE}$ d'Encodage Unaire Symétrique

Vous allez mettre en place le mécanisme dit d'encodage unaire symétrique [1]. On va expliquer l'application de ce mécanisme à l'attribut `education` de la figure 1, mais qu'il faudrait l'appliquer à tous les attributs du jeu de données.

a. On récupère d'abord le domaine $[v_1, \ldots, v_k]$ de l'attribut stocké sous la forme d'une séquence ordonnée. Pour l'attribut `education` de la figure 1, il s'agit de `[Bachelors,Master,HS-grad]`.

b. La $i^{\text{ème}}$ valeur v_i est codée comme un vecteur de k bits, tous nul sauf le $i^{\text{ème}}$ qui vaut 1. Par exemple, la valeur `Master` de la première ligne de cette figure serait codée en `[0,1,0]`, un vecteur de trois bits. La valeur `HS-grad` de la seconde ligne de cette figure serait quant-à elle codée en `[0,0,1]`.
On parle classiquement de "one-hot encoding" en anglais.

c. Chaque bit b de ce vecteur est alors bruité selon les probabilités suivantes en fonction de ϵ :

$$\Pr[\mathcal{M}_{SUE}(b)=1] = \begin{cases} p = \dfrac{e^{\epsilon/2}}{e^{\epsilon/2}+1} & \text{si } b=1 \\ q = \dfrac{1}{e^{\epsilon/2}+1} & \text{si } b=0 \end{cases} \quad \Pr[\mathcal{M}_{SUE}(b)=0] = \begin{cases} p = \dfrac{e^{\epsilon/2}}{e^{\epsilon/2}+1} & \text{si } b=0 \\ q = \dfrac{1}{e^{\epsilon/2}+1} & \text{si } b=1 \end{cases} \tag{1}$$

4.2 Quelques questions théoriques à propos de $\mathcal{M}_{SUE}$

10. Montrer que le mécanisme $\mathcal{M}_{SUE}$ vérifie la confidentialité différentielle locale. On pourra évaluer le rapport $\dfrac{\Pr[\mathcal{M}(x_1)=y]}{\Pr[\mathcal{M}(x_2)=y]}$ avec x_1 et x_2 deux vecteurs de k bits partout nuls sauf en 1 valeur et y une sortie.
11. On considère N personnes qui ont assaini la valeur de cet attribut. Le nombre entier f_i (respectivement r_i) est le nombre de fois où la valeur v_i apparaît initialement (respectivement après application de $\mathcal{M}_{SUE}$ à chaque réponse individuelle). Montrer qu'un estimateur non biaisé de f_i est

$$\hat{f}_i = \frac{r_i - Nq}{p-q} \tag{2}$$

1. T. Wang, J. Blocki, N. Li, and S. Jha, "Locally differentially private protocols for frequency estimation," in USENIX Security Symposium, 2017, pp. 729–745.

```python
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import CategoricalNB
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score
...
df = pd.read_csv(path+'adult.csv')
df.dropna(inplace=True)

discrete_cols = ['workclass', 'education','marital-status', 'occupation',
                 'relationship', 'race', 'sex','native-country','salary']
df = df[discrete_cols]
df = df.astype('category')

# partie 1
label_encoder = LabelEncoder()
for col in discrete_cols:
    df[col] = label_encoder.fit_transform(df[col])

# partie 2
start=0.1
nbexp = 20
end = 100
nbrep = 10
sequence = np.logspace(np.log10(start), np.log10(end), num=nbexp)

#partie 3
def experimentCategoricalNB(X,y):
    X_train, X_test,y_train,y_test = train_test_split(X,y,test_size=0.2)
    non_private_clf = CategoricalNB()
    non_private_clf.fit(X_train.values, y_train.values.ravel())
    y_pred = np.array(non_private_clf.predict(X_test.values))
    return accuracy_score(y_test,y_pred)

#partie 4
acc=[]
X = df.drop("salary", axis=1)
y = df[["salary"]]
for eps in sequence :
    listOfExpStair = np.array([experimentCategoricalNB(X,y) for _ in range(nbrep)])
    means = np.mean(listOfExpStair)
    stds = np.std(listOfExpStair)
    acc += [(eps,means,stds)]
fd = open(path+"resCategoricalNB","w")
fd.write(str(acc))
```

FIGURE 2 – Code mettant en place de l'apprentissage naïf bayésien catégoriel.

12. Montrer que la variance de cet estimateur est :

$$\mathrm{Var}[\hat{f}_i] = \frac{Nq(1-q)}{(p-q)^2}. \tag{3}$$

13. En ce qui concerne la calcul de fréquence de réponses, on souhaite comparer l'utilité de ce mécanisme avec celle du mécanisme $\mathcal{M}_{GRR}$ vu en cours/TD. Quelle étude faudrait-il entreprendre ? De combien de variables cette étude dépend-elle ?

4.3 Mise en place $\mathcal{M}_{SUE}$ avant un apprentissage Naïf bayésien catégoriel

Pour exprimer les valeurs sous la forme de one-hot encoding, on peut exploiter le code ci-dessous :

```
df_encoded = pd.get_dummies(df, columns=discrete_cols)
df_encoded.drop(['salary_0'], axis=1, inplace=True)
df_encoded.rename(columns={'salary_1': "Outcome"}, inplace=True)
```

où `discrete_cols` est la séquence des attributs discrets retenus pour cette étude.

14. Donner le code de la fonction `sanitizeWithSUE(df,eps)` qui prend en paramètre un dataframe `df` (déjà encodé en one-hot) et un budget de fuite `eps` par utilisateur et qui bruite chacune des valeurs binaires en appliquant le mécanisme $\mathcal{M}_{SUE}$.
15. Une expérience consiste à bruiter les données selon $\mathcal{M}_{SUE}$ de paramètre ϵ, puis à mettre en place un apprentissage naïf bayésien catégoriel et enfin à récupérer la précision de ce dernier. Pour chaque valeur de ϵ, on répète 10 fois la même expérience et on calcule la valeur moyenne des précisions et leur écart type. On mémorise ceci sous la forme d'un triplet `(eps,mean,std)`.

    Construire le code python permettant de construire les 20 triplets `[(eps_1,mean_1,std_1),...,(eps_20,mean_20,std_20)]` où les epsilons suivent une échelle logarithmique entre 0.1 et 100 (comme cela est réalisé dans la partie 2 du code présenté à la figure 2) et où chaque triplet `(eps_i,mean_i,std_i)` est le résultat d'une expérience.
16. Que constatez vous en termes de précision par rapport à l'apprentissage sans PVP ?
17. Concernant un apprentissage naïf bayésien catégoriel, comparer expérimentalement les mécanismes $\mathcal{M}_{SUE}$ et $\mathcal{M}_{GRR}$ sur ce jeu de données en faisant varier epsilon.