M1-ISL, Initiation à la Recherche. TD.
Étudier des variations de la réponse randomisée de Warner.

Jean-François COUCHOT
couchot[arobase]femto-st[point]fr

8 janvier 2025

Dans ce TD, on considère la méthode de réponse randomisée [1] qui permet de répondre de manière "anonyme" à une question sur des données sensibles déjà vue en CM. L'objectif est de répondre à la Question de Recherche suivante : « peut on remplacer les 2 tirages aléatoires pile/face de la méthode de Warner par deux tirages selon des probabilités définies avec le même niveau de protection et une utilité plus grande ? »

On commencera par une étude pratique (Section 1) et on continuera par une étude théorique plus approfondie (Section 2).

1 Algorithmes de réponse randomisée

Exercice 1.1 (Code implantant le nettoyage de la réponse). *Donner le code (3 lignes tout au plus) de la fonction* `reponse_randomisee(rep_o)` *qui prend en paramètre une réponse originale booléenne et qui retourne une réponse du même type selon l'algorithme de réponse randomisée original (avec des pièces de monnaie) donné dans le cours.*

Exercice 1.2 (Code implantant l'estimateur). *Donner le code (3 lignes tout au plus) de la fonction* `estimation_pourcent_oui_initiaux(reps)` *qui prend en paramètre la série de réponses nettoyées et retourne une estimation du pourcentage de* OUI *qui étaient présents initialement.*

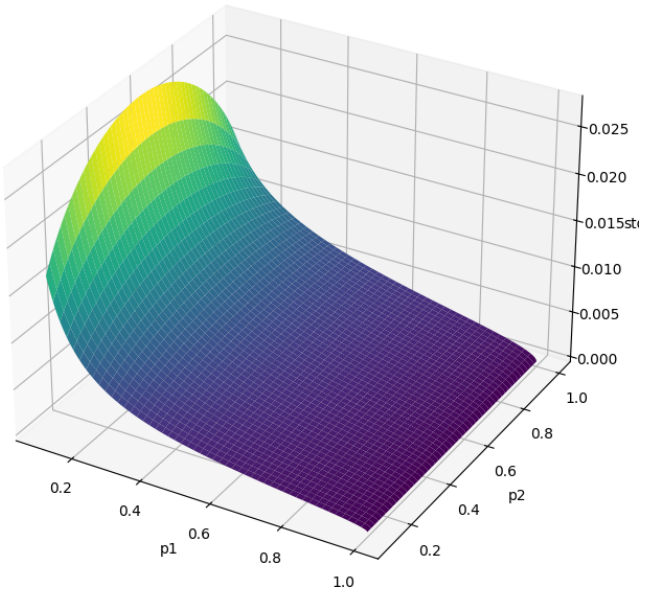
Exercice 1.3 (Evaluation numérique statistique du mécanisme).

1. *Commencer par donner le code de la fonction* `eval_rep_randomisee(nb_exp,data)` *qui prend en paramètres le nombre* `nb_exp` *et la séquence de valeurs binaires originales* `data`. `nb_exp` *fois, le jeu de données* `data` *est nettoyé selon l'algorithme de réponse randomisée puis le pourcentage de* OUI *qui étaient présents initialement est estimé. Cette estimation est accumulée dans une liste qui est retournée.*
2. *Donner le code qui calcule la moyenne, la variance et l'écart-type de cette liste.*
3. *On considère le jeu de données adult. La donnée sensible est le fait qu'une personne soit vendeuse (Occupation = Sales). Évaluer la démarche de réponse randomisée sur cette question en répétant 100 fois l'expérimentation.*

Exercice 1.4 (Extension de l'algorithme).

1. *Construire théoriquement le nouvel estimateur $\hat{f}'$ du pourcentage initial de réponses* OUI *d'une réponse randomisée qui repose cette fois sur les probabilités p_1 de dire la vérité et p_2 de sortir un* OUI *si le tirage ne contraint pas de dire la vérité.*
2. *Compléter le code précédent pour qu'il prenne en compte cette extension.*
3. *On considère que $p_1 \in [0.1, 0.2, \ldots, 1.0]$ et $p_2 \in [0.1, 0.2, \ldots, 1.0]$. Pour chaque paire (p_1, p_2), estimer pratiquement l'écart-type de $\hat{f}'$ avec 2 évaluations avec les mêmes données qu'à la question 3 de l'exercice précédent.*

1. Warner, S. L. (1965). Randomized response : A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association, 60(309), 63-69.



4. *Illustrer ceci à l'aide d'un histogramme en 3D, comme donné à la figure 1.4. On pourra s'inspirer de la référence* [2]
5. *Quels semblent être les bons candidats* (p_1, p_2) *pour minimiser l'écart-type ?*

2 Aspects théoriques

Exercice 2.1 (Premières preuves).

1. *Estimer théoriquement la variance de l'estimateur dans l'algorithme original de Warner.*

   (a) *On rappelle que la variance est définie comme la moyenne des carrés des écarts à la moyenne. Montrer que pour une VAR* X, $\mathrm{Var}[X] = E[X^2] - E[X]^2$.

   (b) *On considère un·e utilisateur·rice dont la réponse originale à la question est* OUI. *Soit* X *la VAR qui vaut 1 si sa réponse randomisée est* OUI *et 0 sinon. Évaluer* $E[X]$, $E[X^2]$. *En déduire que* $\mathrm{Var}[X] = \delta_1 (1 - \delta_1)$ *avec* $\delta_1 = \frac{3}{4}$.

   (c) *De manière similaire, on considère un·e utilisateur·rice dont la réponse originale à la question est* NON. *Soit* Y *la VAR qui vaut 1 si sa réponse randomisée est* OUI *et 0 sinon. Montrer que* $\mathrm{Var}[Y] = \delta_0(1 - \delta_0)$ *avec* $\delta_0 = 1/4$.

2. https://jakevdp.github.io/PythonDataScienceHandbook/04.12-three-dimensional-plotting.html

(d) Soit R la VAR qui compte le nombre de réponses randomisées égales à OUI *parmi les N observations de F utilisateur·rices dont la réponse originale à la question est* OUI *et des $N - F$ utilisateur·rices dont la réponse originale à la question est* NON.

Montrer que $\mathrm{Var}[R] = F.\mathrm{Var}[X] + (N - F).\mathrm{Var}[Y]$.

(e) Montrer alors que $\mathrm{Var}[\hat{f}] = \frac{4}{N^2} \times \mathrm{Var}[R] = \frac{3}{4N}$. *Discuter de cette variance.*

2. *On va reprendre la question précédente, mais en considérant cette fois les probabilités p_1 et p_2 comme à l'exercice précédent.*

(a) En déduire que $V[X'] = \delta_1(1 - \delta_1)$ avec $\delta_1 = p_1 + (1 - p_1)p_2$.

(b) De manière similaire, montrer que $\mathrm{Var}[Y'] = \delta_0(1 - \delta_0)$ *avec* $\delta_0 = (1 - p_1)p_2$.

(c) Soit R' la VAR qui compte le nombre de réponse randomisées égales à OUI. *Montrer que* $\mathrm{Var}[R'] = F.\mathrm{Var}[X'] + (N - F).\mathrm{Var}[Y']$.

(d) Montrer alors que $\mathrm{Var}[\hat{f}'] = \frac{\mathrm{Var}[R']}{N^2.p_1^2}$.

(e) On admettra pour la suite que

$$\mathrm{Var}[\hat{f}'] = \frac{F.p_1.(1 - 2p_2).(1 - p_1) + N.p_2.(1 - (1 - p_1)p_2).(1 - p_1)}{N^2 * p_1^2} \quad (1)$$

(f) *Pourquoi peut-on poser* $F = \alpha N$ *avec* $\alpha \in [0,1]$ *? Simplifier* $\mathrm{Var}[\hat{f}']$ *en conséquence.*

(g) *Montrer que cette variation est plus utile si et seulement si l'inéquation suivante est établi*

$$3p_1^2 - 4\alpha.p_1.(1-2p_2).(1-p_1) - 4p_2.(1-(1-p_1)p_2).(1-p_1) > 0 \tag{2}$$

Exercice 2.2 (Intégrer la contrainte de PVP). *1. Dans ce contexte d'extension, quels sont les rapports de probabilités maximaux suivants* $\frac{\Pr[\mathcal{M}(x_1)=y]}{\Pr[\mathcal{M}(x_2)=y]}$ *pour* $x_1, x_2, r \in \{\text{OUI}, \text{NON}\}$

		y	
		OUI	NON
x	OUI		
	NON		

2. *On souhaite garantir que ce mécanisme ne dévoilera pas plus d'information que celui de réponse randomisée initial. Montrer que ceci nécessite que les deux contraintes suivantes soient simultanément établies :*

$$p_1 + (1-p_1)p_2 \leq 3(1-p_1)p_2 \tag{3}$$
$$p_1 + (1-p_1)(1-p_2) \leq 3(1-p_1)(1-p_2) \tag{4}$$

3. *On affirme que le code suivant renvoie une liste vide. Que peut-on en conclure ?*

```
import numpy as np
p1_values, p2_values = np.arange(0, 1.005, 0.005), np.arange(0, 1.005, 0.005)
valid_solutions = []

def varDif(p1,p2,alpha):
  return 3*(p1**2)-4*((alpha*p1*(1-2*p2)*(1-p1)+p2*(1-(1-p1)*p2)*(1-p1)))

for p1 in p1_values:
    for p2 in p2_values:
        if p1+(1-p1)*p2<=3*((1-p1)*p2) and p1+(1-p1)*(1-p2)<=3*((1-p1)*(1-p2)):
          for alpha in np.arange(0, 1.005, 0.005):
              vd = varDif(p1,p2,alpha)
              if vd > 0 :
                  valid_solutions.append((p1,p2,alpha,vd))
print(valid_solutions)
```