HW1

February 4, 2020

1 Homework 1

Maxime TCHIBOZO (mt3300)

```
In [18]: import numpy as np
         import pandas as pd
```

1.1 Exercise 1

Provide code to simulate randomized controlled experimental data where

(a) The user-level causal effect δ_i is constant for each user in an online experiment.

(b) The user-level causal effect δ_i is gaussian random, with average $E[\delta_i] = 1$.

(c) The user-level causal effect δ_i depends on a gaussian random variable U an $E[\delta_i] = 1$.

```
In [19]: N=10000
```

1.2 User-level causal effect is constant for each user

```
In [20]: deltas = np.ones(N) # constant delta_i for each user

         y0 = np.random.normal(size = N)
         y1 = y0+deltas

         df = pd.DataFrame({'$Y^{(D=1)}$':y1,
                            '$Y^{(D=0)}$':y0,
                            '$\delta$':deltas})
         df['$D$'] = np.random.binomial(1, p=0.5,size=N)
         df['$Y$'] = df['$D$']*df['$Y^{(D=1)}$']+(np.ones(N)-df['$D$'])*df['$Y^{(D=0)}$']
         display(df.head())
```

```
   $Y^{(D=1)}$  $Y^{(D=0)}$  $\delta$  $D$       $Y$
0     0.424543    -0.575457       1.0    0 -0.575457
1     0.204752    -0.795248       1.0    1  0.204752
2     0.878451    -0.121549       1.0    0 -0.121549
3     2.013948     1.013948       1.0    0  1.013948
4    -0.803852    -1.803852       1.0    1 -0.803852
```

1.3 User-level causal effect is gaussian with mean 1

```
In [21]: deltas = np.random.normal(loc=1,size=N) #gaussian delta_i with mean 1

         y0 = np.random.normal(size = N)
         y1 = y0+deltas

         df = pd.DataFrame({'$Y^{(D=1)}$':y1,
                            '$Y^{(D=0)}$':y0,
                            '$\delta$':deltas})
         df['$D$'] = np.random.binomial(1, p=0.5,size=N)
         df['$Y$'] = df['$D$']*df['$Y^{(D=1)}$']+(np.ones(N)-df['$D$'])*df['$Y^{(D=0)}$']
         display(df.head())
```

```
   $Y^{(D=1)}$  $Y^{(D=0)}$  $\delta$  $D$       $Y$
0     3.318538     1.754475  1.564062    0  1.754475
1    -2.832527    -2.025529 -0.806998    1 -2.832527
2     0.529197    -0.843590  1.372787    0 -0.843590
3    -0.505287    -0.682514  0.177227    0 -0.682514
4    -0.743828    -1.234168  0.490340    1 -0.743828
```

1.4 User-level causal effect depends on a gaussian random variable U and has mean 1

Let $U \sim N(1,1)$
Let $\delta \sim N(U,1)$
$\mathbb{E}[\delta] = \mathbb{E}[\mathbb{E}[\delta|U]] = \mathbb{E}[U] = 1$

```
In [22]: N = 10000
         U = np.random.normal(loc=1,size=N)

         deltas = []
         for i in range(N):
             deltas.append(np.random.normal(loc=U[i]))
         deltas = np.array(deltas) #delta_i is normally distributed around the mean of a norma
         #random variable U


         y0 = np.random.normal(size = N)
         y1 = y0+deltas

         df = pd.DataFrame({'$Y^{(D=1)}$':y1,
                            '$Y^{(D=0)}$':y0,
                            '$\delta$':deltas})
         df['$D$'] = np.random.binomial(1, p=0.5,size=N)
         df['$Y$'] = df['$D$']*df['$Y^{(D=1)}$']+(np.ones(N)-df['$D$'])*df['$Y^{(D=0)}$']
         display(df.head())
```

```
   $Y^{(D=1)}$  $Y^{(D=0)}$  $\delta$  $D$       $Y$
```

```
0    -0.147595     0.454518 -0.602113    1 -0.147595
1     0.657162     0.222109  0.435053    1  0.657162
2     2.470180     1.392297  1.077883    0  1.392297
3     0.830302    -0.049271  0.879573    0 -0.049271
4     0.881244     0.094640  0.786604    0  0.094640
```

```
In [23]: np.mean(df["$\delta$"])
```

```
Out[23]: 0.9725416921452817
```

The naive estimator of the causal effect δ is approximately equal to 1

2 Exercise 2

Suppose a product manager suggests running an experiment to measure the effect of changing the size of the result font for an online search engine on click-throughs for the top result. Suppose the initial click-through rate is $p = 0.05$.

(a) What is the initial standard deviation in the click-through rate?

If we choose the click-through rate as our KPI, we can consider that the event "User i clicks on the link" follows a Bernouilli distribution with probability p = 0.05.
This means that the standard deviation is initially $\sigma_p = \sqrt{p(1-p)}$

(b) What sample size would be required to measure a change of 0.01, or 20%

For 80% power at 95% confidence that there is an experimental effect, the number of users necessary to detect a change of x (i.e x*100 %) is:
$n = \frac{16\cdot p(1-p)}{(x\cdot p)^2}$
Indeed, the inital rate is p, so the δ must be $x \cdot p$.
For x = 0.2 (20% change):
n = $\frac{16\cdot 0.05\cdot 0.95}{(0.2\cdot 0.05)^2}$
n = 7,600
As we have seen, when δ is large, the sample size necessary to detect a difference in click through rate is small.

2.1 Exercise 3

Your colleague is designing an experiment to test a new recommendation system. The system runs at the bottom of product pages on an e-commerce site. They suggest comparing average revenue for people who have clicked through the recommendation system with that of people who have not, and comparing the resulting averages to estimate the average effect of their recommendation system.

(a) Can you suggest a better metric?

As Lewis and Rao have shown in their paper "On the Near Impossibility of Measuring the Returns to Advertising" (2013), extra revenue generated is not a good metric for evaluating the effectiveness of a system. The standard deviation of revenue per customer is very high in comparison to the average revenue as only a small portion of users make purchases, and these purchases tend to be of a very large value. The effect of the recommendation system is dwarfed by the variation in sales, so we can not measure it effectively under all the noise.

A more pertinent metric would be the average click through rates for the users of the website who have been shown the recommendation system, and for those who have been shown a "control recommendation system", which looks similar, but does not recommend items in the same way.

(b) What other problems do you see with this design?

The design of the original experiment is likely subject to "differential treatment effect bias". Suppose that for some reason the recommendation system works well for some users, and poorly for others (perhaps the system ignores a particular niche, meaning users who are interested in this niche do not get good recommendations). Over time, users learn whether or not the recommendation system works for them or not, and decide to either keep using it or to ignore it. We will measure that the recommendation is effective over the whole population, as those who benefit from the recommendation system will spend more. This is simply not the case. In reality, the recommendation system works well for some people and poorly for others, even before they use it. This is why we need randomization.

A more implicit problem with this design could be that the SUTVA assumption is not verified. Since recommendation systems work based on the behaviour of other users, it could be the case that the recommendation system be more or less functional depending on the size of the group who have clicked through the recommendation system. As more people use the recommendation system, a new user who will see the recommendation system will be recommended more appropriate items, so in turn become more likely to use the system. In this case, the potential outcume of user i would depend on the behavior of a user j.