DS 102 - HW 05

1. Simpson's Paradox

a)

Adults	R	R^c	$R+R^c$	Recovery Rate
Drug (D)	0	1	1	0
No Drug (D^c)	14	24	39	0.359
	14	25	40	

Kids	R	R^c	$R+R^c$	Recovery Rate
Drug (D)	20	19	39	0.513
No Drug (D^c)	2	0	1	2
	22	19	40	

36 Recover
44 no recover
40 Drug
20 R
20 R^c
40 No Drug
16 R
24 R^c

Simpson's paradox can occur when there is a large imbalance in the subgroups. By putting a large majority of kids as drug-users and adults as non-drug users, we can achieve tables s.t. equation 1 holds

b) If D and A are independent, prove D and K, D^c and A, D^c and K are inde

$P(D) + P(D^c) = 1$, $P(A) + P(K) = 1$ by definition

$P(D \cap A) = P(D) \cdot P(A)$; $P(D) = P(D \cap A) + P(D \cap K)$

$P(D \cap K) = P(D) - P(D \cap A) = P(D) - P(D) \cdot P(A) = P(D)(1 - P(A))$
$= P(D)\,P(K)$

So D is independent of A and K and by same argument, D^c is independent of A and K

c) Given D and A are independent, Simpson's Paradox is impossible

$P(R|D,A) < P(R|D^c,A)$ and $P(R|D,K) < P(R|D^c,K)$ implies $P(R|D) < P(R|D^c)$

$P(R|D) > P(R|D^c)$

$\sum_n P(R|D,A_n)\,P(A_n|D) > \sum_n P(R|D^c,A_n)\,P(A_n|D^c)$

$\sum_n P(R|D,A_n)\,P(A_n) > \sum_n P(R|D^c,A_n)\,P(A_n)$

$\sum_n P(R|D,A_n) > \sum_n P(R|D^c,A_n)$

Given $P(R|D,A) < P(R|D^c,A)$, the above statement cannot be true.

The same argument applies for Kids.

Therefore, if D and A, D and K, D^c and A, D^c and K are independent, then Simpson's paradox is not possible

$(x_i - \bar{x})^2 = x_i^2 - 2x_i\bar{x} + \bar{x}^2$

DS 102 - HW 5 11-4-19

2. Experiment Design for Linear Models

a.i) Show $Var(\hat{a}_{DB}) = \frac{4}{n}\sigma^2$, $Var(\hat{a}_{OLS}) = \frac{\sigma^2}{\sum_{i=1}^n (x_i - \bar{x})^2}$; $\bar{x} = \frac{1}{n}\sum_{i=1}^n x_i$

$x \in 1, \dots, n \to x_i = 0$ for $i = 1, \dots, \frac{n}{2}$

$x_i = 1$ for $i = \frac{n}{2}+1, \dots, n$ $\quad \sum_{i=1}^n x_i = \sum_{i=1}^{n/2} x_i + \sum_{n/2+1}^{n} x_i = \frac{n}{2}$

$\bar{x} := \frac{1}{n}\left(\sum_{i=1}^{n/2} x_i + \sum_{n/2+1}^{n} x_i\right) = \frac{1}{n}\left(0 + \frac{n}{2}\right) = \frac{1}{2}$

$\sum_{i=1}^n \left(x_i - \frac{1}{2}\right)^2 = \sum_{i=1}^n x_i^2 - x_i + \frac{1}{4}$

$Var(\hat{a}_{DB}) = \sigma^2 / \left(\sum_{i=1}^n x_i^2 - x_i + 1/4\right) = \frac{\sigma^2}{\frac{n}{2} - \frac{n}{2} + n/4} = \frac{4}{n}\sigma^2$

a.ii) Show $Var(\hat{a}_{quad}) = \frac{3}{2} Var(\hat{a}_{DB}) = \frac{3}{2} \cdot \frac{4}{n}\sigma^2 = \frac{6}{n}\sigma^2$

$x_i = 0$ for $i = 1, \dots, \frac{n}{3}$

$x_i = 0.5$ for $i = \frac{n}{3}+1, \dots, \frac{2}{3}n$

$x_i = 1$ for $i = \frac{2}{3}n+1, \dots, n$

$\bar{x} := \sum^{n/3} x_i + \sum_{n/3+1}^{2/3 n} x_i + \sum_{2/3+1}^{n} x_i = \frac{n}{3} + \frac{n}{3}\cdot\frac{1}{2} + 0 = \frac{n}{3} + \frac{n}{6} = \frac{2n}{6} + \frac{n}{6} = \frac{3n}{6} = \frac{1}{2}n = \frac{1}{2}$

$\sum_{i=1}^n x_i^2 = \sum_1^{n/3} 1^2 + \sum_{n/3+1}^{2n/3} \frac{1}{2}^2 + 0 = \frac{n}{3} + \frac{1}{4}\cdot\frac{n}{3} = \frac{n}{3} + \frac{n}{12} = \frac{4n}{12} + \frac{n}{12} = \frac{5n}{12}$

$\sum_{i=1}^n x_i = \frac{n}{3} + \frac{n}{3}\cdot\frac{1}{2} = \frac{n}{3} + \frac{n}{6} = \frac{n}{2}$

$\frac{5n}{12} - \frac{2n}{4} + \frac{n}{4} = \frac{5n}{12} - \frac{n}{4} = \frac{5n}{12} - \frac{3n}{12}$

$Var(\hat{a}_{quad}) = \sigma^2 / \left(\frac{5n}{12} - \frac{n}{2} + \frac{n}{4}\right) = \sigma^2 / n/6 = \frac{6}{n}\sigma^2$ $\quad \frac{2n}{12} = n/6$

a.iii) Show $Var(\hat{a}_{even}) = \frac{3(n-1)}{n+1} Var(\hat{a}_{DB}) = \frac{3(n-1)}{n+1} \cdot \frac{4\sigma^2}{n} = \frac{12\sigma^2(n-1)}{n(n+1)}$ $\quad \frac{n+1}{2n} \cdot 2 \cdot \frac{n+1}{2}$

evenly spaced $\Rightarrow \sum_{i=1}^n i/n = x_i$

$\bar{x} := \frac{1}{n}\left(\sum_{i=0}^n x_i\right) = \frac{1}{n}\left(\sum_{i=0}^n i/n\right) = \frac{n+1}{2} \cdot \frac{1}{n} = \frac{n+1}{2n}$

$Var(\hat{a}_{even}) = \sigma^2 / \sum_{i=1}^n x_i^2 - 2x_i\bar{x} + \bar{x}^2 = \sigma^2 / \sum_{i=1}^n x_i^2 - \frac{x_i(n+1)}{n} + \left(\frac{n+1}{2n}\right)^2$

$\sum_{i=1}^n x_i^2 = \sum_{i=1}^n \left(\frac{i}{n}\right)^2 \quad \frac{1}{n^2}\sum_{i=1}^n i^2 = \frac{(n+1)(2n+1)}{6n}$

$Var(\hat{a}_{even}) = \sigma^2 / \frac{(n+1)(2n+1)}{6n} - \frac{(n+1)}{2} \cdot \frac{(n+1)}{n} + \frac{n(n+1)^2}{4n^2} = \sigma^2 / \frac{(n+1)(2n+1)}{6n} - \frac{2(n+1)^2}{4n} + \frac{(n+1)^2}{4n} = \sigma^2 / \frac{(n+1)(2n+1)}{6n} - \frac{(n+1)^2}{4n}$

$= \frac{\sigma^2}{\frac{(n-1)(n+1)}{12n}} = \frac{12n\sigma^2}{(n-1)(n+1)}$

11-6-19

PS 102 - HW 5 X_i

3. K arms, P_i for $i=1,\dots,K$ reward distribution

mean $\mu_i = E_{P_i}[X_i]$, $P(a \le X_i \le b) = 1$ for all $a<b$

pulls each K arms, c times

After cK rounds $\rightarrow$ $\hat{\mu}_i = \frac{1}{c}\sum_{s=1}^{c} X_s$

$P_i \sim X_i$ reward distribution

A_t - denote choice of arm to pull

for $t = 1, 2, \dots$

input: # explore pulls, c

$$A_t = \begin{cases} (t \bmod K) + 1 & t \le cK \\ \operatorname{argmax}_{i \in 1,\dots,K} \hat{\mu}_i & t > cK \end{cases}$$

Mean of optimal arm: $\mu^* = \max_{i \in \{1,\dots,K\}} \mu_i$

Sub-optimality gap: $\Delta_i = \mu^* - \mu_i$

Pseudo Regret $\Rightarrow$ $R(n) = n\mu^* - E\left[\sum_{t=1}^{n} X_{A_t}\right]$

a) $T_a(t) \Rightarrow$ # times arm a, pulled by time t

Show regret $\Rightarrow$ $R(n) = \sum_{i=1}^{k} \Delta_i E(T_i(n))$

$R(n) = n\mu^* - \sum_{t=1}^{n} E(X_{A_t}) = \sum_{t=1}^{n} [\mu^* - E[X_{A_t}]]$

$\sum_{t=1}^{n} E[X_{A_t}] = \sum_{i=1}^{K} \mu_i \cdot E(T_i(n))$

$\sum_{t=1}^{n} X_{A_t} \Rightarrow$ Sum of reward distributions for arm A, up to time step n

$\mu_i = E[X_i]$

$\hookrightarrow$ Avg. reward of Arm i

$E[X_{A_t}]$ = Avg. reward of Arm at time step t

$\downarrow$

$\rightarrow E[\mu_i * T_i(n)] = \mu_i E(T_i(n))]$

$R(n) = \sum_{i=1}^{k} (\mu^* - \mu_i) E(T_i(n)) = \sum_{i=1}^{k} \Delta_i E(T_i(n))$

"Expectation of arm i at time step n"

b) Show if $n > cK$, $E[T_i(n)] = c + (n - Kc) P_r(\hat{\mu}_i > \max_{j \ne i} \hat{\mu}_j)$

$T_i(n) = \sum_{s=1}^{n} \mathbb{1}\{A_s = i\} \Rightarrow E[\sum_{s=1}^{n} \mathbb{1}\{A_s = i\}] = \sum_{s=1}^{n} E[\mathbb{1}\{A_s = i\}]$

$= \sum_{s=1}^{cK} E[\mathbb{1}\{A_s = i\}] + \sum_{cK+1}^{n} E[\mathbb{1}\{A_s = i\}]$

$= \underline{\sum_{s=1}^{cK} P_r((s \bmod k) + 1 = s)} + \sum_{s=cK+1}^{n} Pr(\hat{\mu}_i > \max_{j \ne i} \hat{\mu}_j)$

$= c + (n - cK) P_r(\hat{\mu}_i > \max_{j \ne i} \hat{\mu}_j)$

c) Given optimal arm $\overset{i=1}{=} 1$, show for any sub-optimal arm i:

$P_r(\hat{\mu}_i > \max_{j \ne i} \hat{\mu}_j) \le P_r(\hat{\mu}_i > \hat{\mu}_1)$

$\hat{\mu}_1 = \mu^*$

Given optimal arm 1, and any suboptimal arm i:

$\max_{j=1,\dots,K,\, j \ne i} \hat{\mu}_j = \hat{\mu}_1 = \mu^* \Rightarrow P_r(\hat{\mu}_i > \hat{\mu}_1) \le P_r(\hat{\mu}_i > \hat{\mu}_1)$

$P_r(\hat{\mu}_i > \hat{\mu}_1) > P_r(\hat{\mu}_i > \hat{\mu}_j)$ for all $j = 1,\dots,K$

DS 102 - HW5 11-9-19

3. d) Use Hoeffding to show $P(\hat{\mu}_i > \hat{\mu}_1) \le \exp\{-\frac{c\Delta_i^2}{(b-a)^2}\}$ $\Delta_i = \mu^* - \mu_i$

Given : $E[T_i(n)] \le c + (n - Kc)\Pr(\hat{\mu}_i > \hat{\mu}_1)$, All X_i bounded on $[a, b]$

$\hat{\mu}_i = \frac{1}{c}\sum_{s=1}^{c} X_s \Rightarrow$ Constant

$\mu^* = \max_{i \in (1,\dots,K)} \mu_i = $ "$\hat{\mu}_1$"

$\Pr(\hat{\mu}_i > \hat{\mu}_1) = \Pr(\hat{\mu}_i - E(\hat{\mu}_i) > \hat{\mu}_1 - E(\hat{\mu}_i))$

$= \Pr(\hat{\mu}_i - E[\frac{1}{c}\sum_{s=1}^{c} X_s] > \hat{\mu}_1 - \hat{\mu}_i)$

$= \Pr(\frac{1}{c}\sum_{s=1}^{c} X_s - E[\frac{1}{c}\sum_{s=1}^{c} X_s]) > \Delta_i) = \Pr(\frac{1}{c}\sum_{s=1}^{c}(X_s - E[X_s]) > \Delta_i)$

$= \Pr(\sum_{s=1}^{c}(X_s - E[X_s]) > c\Delta_i)$

$\le \exp\left\{-\frac{c^2\Delta_i^2}{\sum_{s=1}^{c}(b_s - a_s)^2}\right\} = \exp\left\{-\frac{c^2\Delta_i^2}{c(b-a)^2}\right\} = \exp\left\{-\frac{c\Delta_i^2}{(b-a)^2}\right\}$

e) $E[T_i(n)] \le c + (n - Kc)\exp\{-\frac{c\Delta_i^2}{(b-a)^2}\}$

Min. Sub-optimality : $\Delta = \min_{i>1} \Delta_i$, Then for each sub-optimal arm $i = 2, \dots, K$:

$E[T_i(n)] \le c + n \cdot \exp\{-\frac{c\Delta^2}{(b-a)^2}\}$ w/ $(n - Kc)$ upper bounded by n

Find value of c s.t. $\exp\left(-\frac{c\Delta^2}{(b-a)^2}\right) \le \frac{1}{n}$

$-\frac{c\Delta^2}{(b-a)^2} \le \ln\frac{1}{n} = -\ln(n)$

$-c\Delta^2 \le -\ln(n)\cdot(b-a)^2$, $c\Delta^2 \ge \ln(n)\cdot(b-a)^2$

$c \ge \frac{\ln(n)\cdot(b-a)^2}{\Delta^2}$

$\exp\left(-\frac{\frac{2\ln(n)(b-a)^2}{\Delta^2}\Delta^2}{(b-a)^2}\right) \le \frac{1}{n}$

$\exp(-2\ln(n)) \le \frac{1}{n}$

$\exp(\ln(\frac{1}{n^2})) \le \frac{1}{n}$

$\frac{1}{n^2} \le \frac{1}{n}$ ✓

Pseudo-Regret : $R(n) = \sum_{i=1}^{K} \Delta_i E(T_i(n))$

$\le \sum_{i=1}^{K} \Delta_i\left(c + n\exp\left(-\frac{c\Delta^2}{(b-a)^2}\right)\right)$

In [1]:
```
import matplotlib.pyplot as plt
import numpy as np
import scipy.stats
import seaborn as sns
import math
```

Part B

In [2]:
```
n = 100
even_spacing = [i/n for i in range(n)]
dumbbell = [0] * int(n/2) + [1] * int(n/2)
quad = [0] * int(n/3) + [1] * int(n/3) + [0.5] * int(n/3) + [0.5]

list_of_lists1 = [even_spacing, dumbbell, quad]
```

In [26]:
```
a_const = 5
b_const = 8
# eps = np.random.normal(loc=0, scale=0.5, size=1)
```

In [27]:
```
y1_even = []
y2_dumb = []
y3_quad = []
list_of_lists2 = [y1_even, y2_dumb, y3_quad]
for i in range(3):
    for x in list_of_lists1[i]:
        y = a_const * x + np.random.normal(loc=0, scale=0.5, size=1)
        list_of_lists2[i].append(y[0])
```

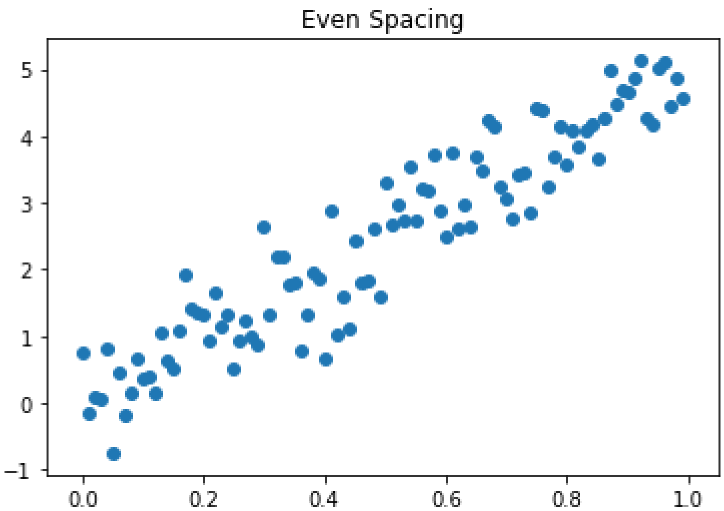
In [28]:
```
a_ols = []

for i in range(3):
    xs = list_of_lists1[i]
    ys = list_of_lists2[i]
    x_bar = np.mean(xs)
    y_bar = np.mean(ys)
    alpha = 0
    for j in range(n):
        alpha += (xs[j] - x_bar) * (ys[j] - y_bar) / (xs[j] - x_bar)**2
#     a = np.sum([x - x_bar for x in xs])*np.sum([y - y_bar for y in y
s])/np.sum([(x - x_bar)**2 for x in xs])
    a_ols.append(alpha)
```

```
/usr/local/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:
10: RuntimeWarning: invalid value encountered in double_scalars
  # Remove the CWD from sys.path while we load stuff.
```

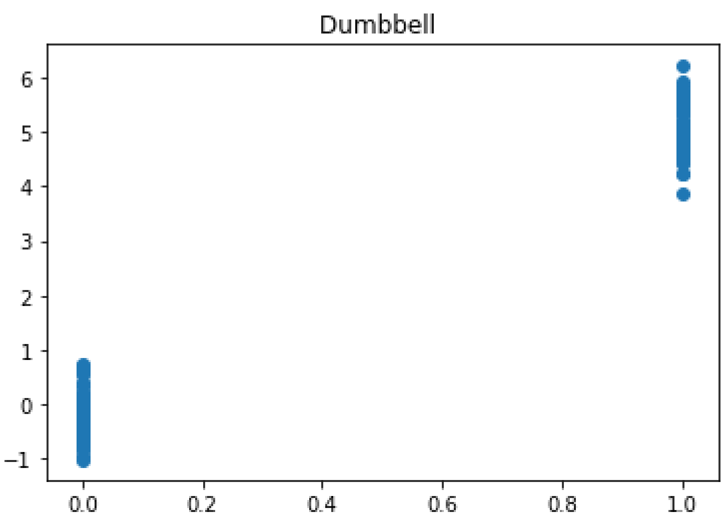
In [29]:
```
a_ols
```

Out[29]:
```
[955.5048739574764, 524.3193985660329, nan]
```

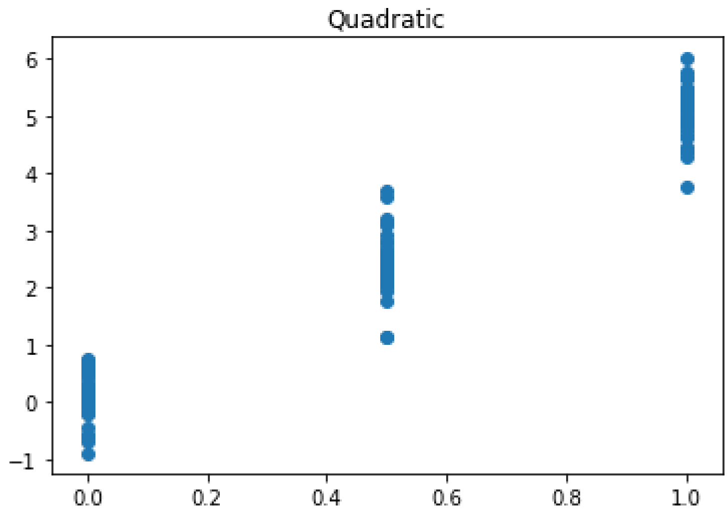
In [34]:
```
plt.scatter(even_spacing, y1_even)
# plt.plot(even_spacing, y1_even)
plt.title('Even Spacing')
plt.show()
```



In [31]:
```
plt.scatter(dumbbell, y2_dumb)
# plt.plot(even_spacing, y1_even)
plt.title('Dumbbell ')
plt.show()
```



In [32]:
```
plt.scatter(quad, y3_quad)
plt.title('Quadratic')
plt.show()
```



In [35]:
```
y1_even = []
y2_dumb = []
y3_quad = []
list_of_lists2 = [y1_even, y2_dumb, y3_quad]
for i in range(3):
    for x in list_of_lists1[i]:
        y = b_const * (x - 0.5)**2 + np.random.normal(loc=0, scale=0.5,
size=1)
        list_of_lists2[i].append(y[0])
```

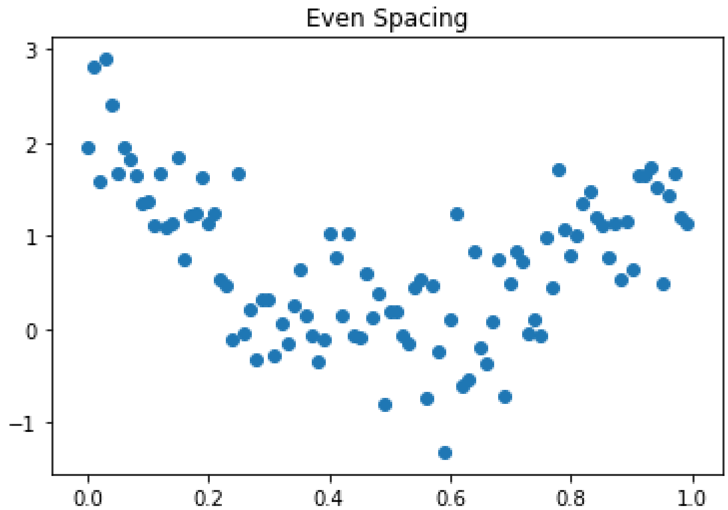
In [36]:
```
a_ols = []

for i in range(3):
    xs = list_of_lists1[i]
    ys = list_of_lists2[i]
    x_bar = np.mean(xs)
    y_bar = np.mean(ys)
    alpha = 0
    for j in range(n):
        alpha += (xs[j] - x_bar) * (ys[j] - y_bar) / (xs[j] - x_bar)**2
    a_ols.append(alpha)
```

```
/usr/local/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:
10: RuntimeWarning: invalid value encountered in double_scalars
  # Remove the CWD from sys.path while we load stuff.
```

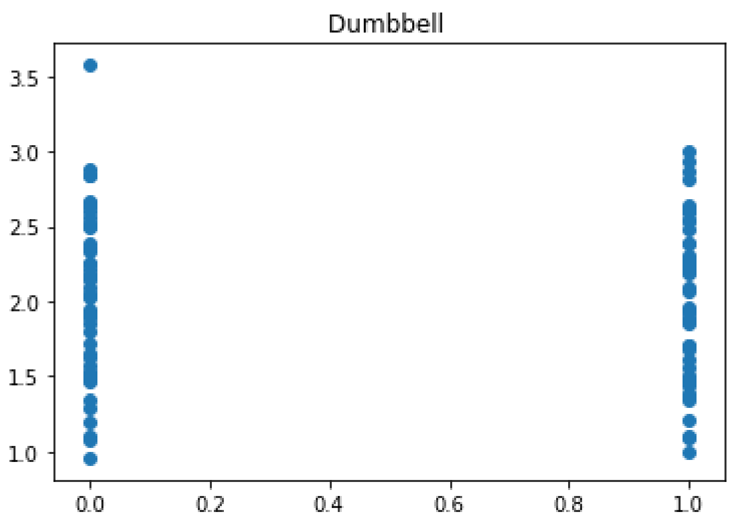
In [37]:
```
a_ols
```

Out[37]:
```
[105.8799360342669, -2.7380566902176735, nan]
```

In [38]:
```
plt.scatter(even_spacing, y1_even)
# plt.plot(even_spacing, y1_even)
plt.title('Even Spacing')
plt.show()
```



In [39]:
```
plt.scatter(dumbbell, y2_dumb)
# plt.plot(even_spacing, y1_even)
plt.title('Dumbbell ')
plt.show()
```



In [40]:
```
plt.scatter(quad, y3_quad)
plt.title('Quadratic')
plt.show()
```