AQA HW #2 Workbook

Tyler Maule

Homework #2 Part (A) $S_R^2 = \frac{(Y_1+Y_2-(2n/3))^2}{2n/9}$

$S_U^2 = \sum_{i=1}^{3} \frac{(Y_i - n\hat{\pi}_{0,i})^2}{n\hat{\pi}_{0,i}}$

Homework #2 Part (B) Note that due to my poor choice of variable names, y_i refer to probability values associated with the i_{th} outcome, and not the counts of those outcomes.

```
import scipy.stats as sps
import numpy as np
```

```
def get_probs(n, y1, y2, y3):

  lR = n * ((6 * (((y1 + y2)/(2)) - (1/3))**2) + (3 * ((y3)-(1/3))**2))
  lU = 3 * n * (((y1)-(1/3))**2 + ((y2)-(1/3))**2 + ((y3)-(1/3))**2)

  aPR = sps.ncx2.sf(x=3.8415, df=1, nc=lR)
  aPU = sps.ncx2.sf(x=5.9915, df=2, nc=lU)

  Ry1 = np.empty(1)
  Ry2 = np.empty(1)

  runs = 10000

  for i in range(runs):
    simY = sps.multinomial.rvs(n, np.array([y1, y2, y3]), size=1)

    Ry1 = np.append(Ry1,simY[0][0])
    Ry2 = np.append(Ry2,simY[0][1])

  SR = ((Ry1 + Ry2 - (2 * n/3))**2)/((2/9)*n)
  SU = (((Ry1 - (y1 * n))**2)/(n * ((y1)/(1)))) + (((Ry2 - (y2 * n)))**2)/(n * ((y2)/(1)))

  PR = np.mean(np.where(SR >= 3.8415,1,0))
  PU = np.mean(np.where(SU >= 5.9915,1,0))

  print(n, y1, y2, y3)
  print("PR: ", PR)
  print("lR: ", lR, " aPR: ", aPR)
  print("PU: ", PU)
  print("lU: ", lU, " aPU: ", aPU)

  return
```

```
get_probs(250, (0.22), (0.40), (0.38))
```

```
## 250 0.22 0.4 0.38
## PR:  0.3785621437856214
## lR:  2.4499999999999997  aPR:  0.34673378605875427
## PU:  0.020897910208979104
## lU:  14.6  aPU:  0.9380512934174401
```

Sample size n	True probability p_T	P_R	aP_R	P_U	aP_U
75	(1/3), (1/3), (1/3)	0.037596240[illegible]	[illegible]	[illegible]	[illegible]169854882
75	(1/4), (1/4), (2/4)	0.82491750[illegible]	[illegible]	[illegible]	[illegible]38759543
75	(1/6), (3/6), (2/6)	0.03569643[illegible]	[illegible]	[illegible]	[illegible]91510478
75	(0.2), (0.3), (0.5)	0.81901809[illegible]	[illegible]	[illegible]	[illegible]71848225
250	(1/3), (1/3), (1/3)	0.05049495[illegible]	[illegible]	[illegible]	[illegible]300916674
250	0.22, 0.4667, 0.3333	0.09719028[illegible]	[illegible]	[illegible]	[illegible]016418592
250	0.25, 0.3, 0.45	0.97110288[illegible]	[illegible]	[illegible]	[illegible]596511949
250	0.22, 0.40, 0.38	0.36796320[illegible]	[illegible]	[illegible]	[illegible]1293417440

I observe generally that power values are lowest when the null hypothesis is true. Unrestricted power values increase as the probabilities become more dissimilar. In the case of restricted power values, however, the values are still lower if the true parameters don't align with the restricted alternative hypothesis. But in the case that true parameters do align with the alternative hypothesis, restricted power values will be higher than unrestricted power values. Oddly restricted exact probabilities are roughly similar to restricted approximate probabilities, but in the unrestricted cases there is far less agreement.

```
n = 128

y1 = 0.25
y2 = .3
y3 = 0.45

lR = n * ((6 * (((y1 + y2)/2) - (1/3))**2) + (3 * ((y3)-(1/3))**2))
lU = 3 * n * (((y1)-(1/3))**2 + ((y2)-(1/3))**2 + ((y3)-(1/3))**2)

print(sps.ncx2.sf(x=3.8415, df=1, nc=lR, loc=0, scale=1))
```

Homework #2 Part (C)

```
## 0.7995539264244444
```

```
print(sps.ncx2.sf(x=5.9915, df=2, nc=lU, loc=0, scale=1))
```

```
## 0.7355154291587768
```

True probability p_T	n_R	n_U
(1/3), (1/3), (1/3)	784900000	963500000
(1/4), (1/4), (2/4)	63	78
(1/6), (3/6), (2/6)	> 1e300	58
(0.2), (0.3), (0.5)	63	69
(0.3), (0.3), (0.4)	393	482
0.22, 0.4667, 0.3333	13040	105
0.25, 0.3, 0.45	129	149

True probability p_T	n_R	n_U
0.22, 0.40, 0.38	801	164

Unsurprisingly, the highest sample size required for unrestricted alternative hypotheses is for the case when the null hypothesis is true. With restricted alternative hypothesis, however, we see an even higher sample size required when probability values are (1/6, 3/6, 2/6). Sample sizes needed are lowest when true probabilities are (0.2, 0.3, 0.5).