Task13_1

December 13, 2022

[35]:
```python
import numpy as np
np.random.seed(42)
```

[36]:
```python
#a the Poisson distribution tends to Gauss if μ = lambda and sigma^2 = lambda
```

[37]:
```python
#b
#the ecdf is calculated by summing over all lower values of x
def cdf(sample, x, sort = False):
    if sort:
        sample.sort()
    cdf = sum(sample <= x)
    cdf = cdf / len(sample)
    return cdf
```

[38]:
```python
# we define the function that actually performs the Kolmogorov-Smirnov test
def ks_2samp(sample1, sample2, alpha):
    observations = np.concatenate((sample1, sample2))
    D_ks = []
    k = np.sqrt(-np.log(alpha/2)/2)
    for x in observations:
        cdf_sample1 = cdf(sample = sample1, x  = x)
        cdf_sample2 = cdf(sample = sample2, x  = x)
        D_ks.append(abs(cdf_sample1 - cdf_sample2))
    ks_stat = max(D_ks)
    m, n = int(len(sample1)), int(len(sample2))
    en = np.sqrt(m*n/(m+n))
    d = en*ks_stat
    if d <= k:
        result = "proved"
    else:
        result = "rejected"
    return ks_stat, d, k, result
```

[39]:
```python
#s1 = np.random.normal(loc=0, scale=1.0, size=100)
#s2 = np.random.normal(loc=0, scale=1.0, size=100)
#print(ks_2samp(s1, s2, 0.3))
```

[40]:
```python
#c
alpha = 0.005
l=1
np.random.seed(42)

while 1:
    gauss = np.random.normal(loc=l, scale=np.sqrt((l)), size=10000)
    gauss_bin = np.histogram(gauss, 100, range = (l-5*np.sqrt(l), l+5*np.
 ↪sqrt(l)))
    gauss_bin_1 = gauss_bin[0]
    poisson = np.random.poisson(l, 10000)
    poisson_bin = np.histogram(poisson, 100, range = (l-5*np.sqrt(l), l+5*np.
 ↪sqrt(l)))
    poisson_bin_1 = poisson_bin[0]
    ks = ks_2samp(gauss_bin_1, poisson_bin_1, alpha)
    if ks[3] == 'proved':
        break
    l = l+1

print('lambda = ', l)
# the first lambda where the hypothesis is accepted is lambda = 33
```

```
lambda =  33
```

[41]:
```python
#d
np.random.seed(42)

alpha = [0.025, 0.0001]
l=1
while 1:
    gauss = np.random.normal(loc=l, scale=np.sqrt((l)), size=10000)
    gauss_bin = np.histogram(gauss, 100, range = (l-5*np.sqrt(l), l+5*np.
 ↪sqrt(l)))
    gauss_bin_1 = gauss_bin[0]
    poisson = np.random.poisson(l, 10000)
    poisson_bin = np.histogram(poisson, 100, range = (l-5*np.sqrt(l), l+5*np.
 ↪sqrt(l)))
    poisson_bin_1 = poisson_bin[0]
    ks = ks_2samp(gauss_bin_1, poisson_bin_1, 0.025)
    if ks[3] == 'proved':
        break
    l = l+1
print('lambda = ', l)


l=1
while 1:
```

```
    gauss = np.random.normal(loc=l, scale=np.sqrt((l)), size=10000)
    gauss_bin = np.histogram(gauss, 100, range = (l-5*np.sqrt(l), l+5*np.
 ↪sqrt(l)))
    gauss_bin_1 = gauss_bin[0]
    poisson = np.random.poisson(l, 10000)
    poisson_bin = np.histogram(poisson, 100, range = (l-5*np.sqrt(l), l+5*np.
 ↪sqrt(l)))
    poisson_bin_1 = poisson_bin[0]
    ks = ks_2samp(gauss_bin_1, poisson_bin_1, 0.0001)
    if ks[3] == 'proved':
        break
    l = l+1
print('lambda = ', l)


#for a = 0.025 we get 44 and
#for a = 0.0001 we get 28 for the first value for lambda where the hypothesis␣
 ↪is accepted.
```

```
lambda =  44
lambda =  28
```