Resolución parcial integrador

Segundo cuatrimestre turno noche

Indice

In [30]:
```python
import scipy.stats
import math
import pandas as pd
import numpy as np
from statsmodels.distributions.empirical_distribution import ECDF
```

In [31]:
```python
def resFinalParser(n):
    print(str(round(n,4)).replace(".",","))
```

Ejercicio 1

Pregunta 1
Parcialmente correcta
Puntúa 4,00 sobre 5,00
Marcar pregunta

(5 puntos) Sea X una variable aleatoria con $E(X) = -5$, $Var(X) = 4$ y $Var(X^2) = 4$. Seleccionar todas las identidades que son siempre verdaderas.

Seleccione una o más de una:

- [ ] $E((X^2-2)^2) = 733$
- [x] $E(8X - X^4) = -885$
- [ ] $Var(6X^2+8) = 32$
- [ ] $E(X^2 - 3X + 2) = 42$.
- [x] $Var(3 - 8X^2) = 256$

La respuesta correcta es: $E((X^2-2)^2) = 733$, $Var(3-8X^2) = 256$, $E(8X - X^4) = -885$

Pregunta 1
Sin responder aún
Puntúa como 5,00
Marcar pregunta

(5 puntos) Sea X una variable aleatoria con $E(X) = -5$, $Var(X) = 4$ y $Var(X^2) = 4$. Seleccionar todas las identidades que son siempre verdaderas.

Seleccione una o más de una:

- $E((X^2-2)^2) = 733$
- $E(8X - X^4) = -885$
- $Var(6X^2+8) = 32$
- $E(X^2 - 3X + 2) = 42$.
- $Var(3 - 8X^2) = 256$

$E(x) = -5$ $\quad Var(x) = 4$ $\quad Var(x^2) = 4$

$Var(x) = E(x^2) - E(x)^2 \Rightarrow 4 = E(x^2) - 25 \Rightarrow E(x^2) = 29$

$Var(x^2) = E((x^2)^2) - E(x^2)^2 \Rightarrow 4 = E((x^2)^2) - 841 \Rightarrow E(x^4) = 845$

① $E((x^2-2)^2) = (E(x^2-2))^2 = (E(x^2) - E(2))^2 = (29-2)^2 = 27^2 = 729$

② $E(8x - x^4) = E(8x) - E(x^4) = 8E(x) - E(x^4) = 8 \cdot -5 - 845 = -885$

④ $E(X^2 - 3x + 2) = E(x^2) - 3E(x) + E(2) = 29 - 3.(-5) + 2 = 46 \neq 42$

③ $Var(6x^2+8) = \overset{32}{Var(6x^2)} = 36\ Var(x^2) = 36 \cdot 4 = 144 \neq 32$

⑤ $Var(3 - 8x^2) = Var(-8x^2 + 3) = Var(-8x^2) = 64\ Var(x^2) = 64 \cdot 4$
$= 256$

Ejercicio 2

Pregunta 2
Correcta
Puntúa 11,00 sobre 11,00
Marcar pregunta

(11 puntos) La distancia, en kilometros (km) que recorre un remise en cada uno de sus viajes es una variable aleatoria con la siguiente función de densidad

$$f_X(x) = \begin{cases} -x/200 + 3/25 & \text{si } 4 < x \leq 24 \\ 0 & \text{en otro caso} \end{cases}.$$

a. (2 puntos) Hallar la probabilidad de que un viaje sea de menos de 7 km. 0,2775

b. (7 puntos) Para simplificar el cobro, la empresa decide cobrar los viajes de la siguiente manera: cobra 50 pesos por cada viaje de menos de 7 km y 100 pesos por cada viaje de más de 7 km. Hallar la esperanza de la variable aleatoria que mide el dinero que cobra el remise por cada viaje. 86,1250

c. (2 puntos) Hallar la varianza de la variable definida en el ítem anterior. 501,2344

a. La probabilidad es 0.2775.
b. La esperanza es 86.125.
c. La varianza es 501.2344.

In [32]:
```
def f(x):
    if (4 < x <= 24):
        return -x/200 + 3/25
    else:
        return 0
```

In [33]:
```
scipy.integrate.quad(f,-np.inf, np.inf)
```

Out[33]:
```
(1.0000000013337809, 1.3883774130363236e-09)
```

In [34]:
```
a = scipy.integrate.quad(f, 4, 7)[0]
resFinalParser(a)
```

```
0,2775
```

Pregunta B

Defino Y = "Dinero cobrado por viaje"

Y se distribuye:

- 50 si x < 7
- 100 si x > 7

Luego E(Y) = 50 *P(X < 7) + 100* P(X > 7)

In [35]:
```
# Proba x < 7 es la calculada en A
b = 1 - a
res = 50*a + 100*b
resFinalParser(res)
```

```
86,125
```

Pregunta C

$Var(Y) = E(Y^2) - E(Y)^2$

In [36]:
```
t1 = (50**2)*a + (100**2)*b
t2 = res**2
res = t1 - t2
resFinalParser(res)
```

```
501,2344
```

Ejercicio 3

Pregunta **3**
Correcta
Puntúa 7,00 sobre 7,00
Marcar pregunta

(7 puntos) Una urna tiene 14 bolitas rojas y 7 blancas. Se repite en forma independiente el experimento de extraer 3 bolitas juntas de la urna, mirar los colores y volverlas a meter. ¿Cuál es la probabilidad de que en el intento 4 sea la primera vez que se obtienen 3 rojas?

Respuesta: 0,1049

La probabilidad es 0.1049.

La respuesta correcta es: 0,1049

Tiene pinta de Hypergeométrica, pues tengo una población, un tipo que me interesa y un n extraído.

Luego tengo una geométrica de parametro la proba anterior

Busco la proba de la geometrica = 4

Defino X = "Cantidad de bolitas rojas extraídas de una población de 21 bolitas, de las cuales 14 rojas y 7 blancas"

$X \sim Hyper(21, 14, 3)$

Luego defino Y = "Cantidad de repeticiones necesarias hasta el primer éxito"

$Y \sim Geo(p)$ con $p = P(X = 3)$

Por lo tanto busco $P(Y = 4)$

In [37]:
```
# Hipergeométrica
M = 21 # Nro total del objetos
n = 14 # Nro total de objetos del tipo A
N = 3 # Cantidad de elementos que se extraen sin reposición
va = scipy.stats.hypergeom(M, n, N)
k = va.pmf(3)
```

In [38]:
```
# Geométrica
p = k
va = scipy.stats.geom(p)
res = va.pmf(4)
resFinalParser(res)
```

```
0,1049
```

Ejercicio 4

Pregunta **4**
Correcta
Puntúa 7,00 sobre 7,00
Marcar pregunta

(7 puntos) La cantidad de visitas a cada nuevo video de un youtuber famoso en los primeros 34 minutos desde su publicación se modela con una variable aleatoria Poisson de parámetro 5. Se eligen en forma independiente 10 videos del youtuber. Calcular la probabilidad de que en 2 de ellos haya exactamente 4 visitas en los primeros 34 minutos.

Respuesta: 0,2960

La probabilidad es 0.296.

La respuesta correcta es: 0,296

lambda = 5 visitas cada 34 minutos

Defino X = "Cantidad de visitas en 34 minutos"

X $\sim Poi(5)$

Proba 4 visitas en los pirmeros 34 minutos $\Rightarrow$ P(X=4)

Defino Y = "Cantidad de exitos en 10 repeticiones"

Y $\sim Bin(10, p)$ con $p = P(X = 4)$

In [39]:
```
# Poisson
mu = 5
va = scipy.stats.poisson(mu)
k = va.pmf(4)
```

In [40]:
```
# Binomial
n = 10
p = k
va = scipy.stats.binom(n,p)
res = va.pmf(2)
resFinalParser(res)
```

```
0,296
```

Ejercicio 5

Pregunta 1
Correcta
Puntúa 10,00 sobre 10,00
Marcar pregunta

(10 puntos) La duración en segundos de un proceso de computadora está dado por una variable aleatoria $T = \frac{8}{X}$, donde X es una variable aleatoria continua con función de densidad dada por $f_X(x) = 3x^2$ para $x \in [0, 1]$.

a. (3 puntos) Calcular la esperanza T. 12,0000 ✓

b. (2 puntos) Calcular la varianza de T. 48,0000 ✓

c. (5 puntos) Si se repite el proceso 59 veces en forma independiente, aproximar la probabilidad de que el tiempo total de procesamiento supere los 13 minutos. (Recordar que la variable T está dada en segundos) 0,0880 ✓

a. $E(T) = \int_0^1 \frac{8}{x} \cdot 3x^2 dx = \int_0^1 8 \cdot 3x dx = 24 \frac{x^2}{2}\Big|_0^1 = 12$
b. $E(T^2) = \int_0^1 \left(\frac{8}{x}\right)^2 \cdot 3x^2 dx = \int_0^1 64 \cdot 3 dx = 192.$ Luego $Var(T) = E(T^2) - E(T)^2 = 192 - 144 = 48.$
c. Tenemos $T_1, \ldots, T_{59}$ v. a. i. i. d. $S_{59} = T_1 + \cdots + T_{59}$ representa el tiempo (en segundos) que llevan los 59 procesos de computadora. 13 minutos equivalen a $13 \cdot 60 = 780$ segundos. $P(S_{59} > 780) = P(\frac{S_{59} - 59 \cdot 12}{\sqrt{59 \cdot 48}} > \frac{780 - 59 \cdot 12}{\sqrt{59 \cdot 48}}) \cong 1 - \Phi(1.35) = 0.0885.$ La probabilidad es aproximadamente $0.0885.$

Pregunta A

In [65]:
```
def f(x):
    if 0 <= x <= 1:
        return (1/x) * (3*(x**2))
    else: return 0
```

In [66]:
```
res = 8*scipy.integrate.quad(f,0, 1)[0]
resFinalParser(res)
```

```
12,0
```

Pregunta B

In [43]:
```
def f(x):
    if 0 <= x <= 1:
        return (1/(x**2)) * (3*(x**2))
    else: return 0
```

In [44]:
```
res = 64*scipy.integrate.quad(f,0, 1)[0] - 144
resFinalParser(res)
```

```
48,0
```

Pregunta C

In [47]:
```
# Proba de Sn > x
def TCL_suma(esp, var, n, x):
    va = scipy.stats.norm()
    k = (x - esp * n)/(math.sqrt(var*n))
    return 1 - va.cdf(k)
```

In [64]:
```
mu = 12
var = 48
res = TCL_suma(mu, var, 59, 13*60)
resFinalParser(res)
```

```
0,088
```

Ejercicio 6

Pregunta 2
Parcialmente correcta
Puntúa 3,00 sobre 6,00
Marcar pregunta

(6 puntos) La siguiente tabla muestra los resultados de 200 tiradas de un dado no equilibrado:

Numero	Frecuencia
1	9
2	42
3	29
4	21
5	94
6	5

a. (3 puntos) Calcular la función de distribución acumulada empírica en 4. 0,1050 ✗

b. (3 puntos) Calcular la media podada al 26%. 4,1979 ✓

a. El valor de la empírica es 0.505.
b. La media podada es 4.1979.

In [49]:
```
# Pregunta A - MAL
res = 21/200
resFinalParser(res)
```

```
0,105
```

In [50]:
```
# Pregunta B
datos = [1]*9 + [2]*42 + [3]*29 + [4]*21 + [5]*94 + [6]*5
datos = np.array(datos)
mediaPodada = scipy.stats.trim_mean(datos,proportiontocut = 0.26)
resFinalParser(mediaPodada)
```

```
4,1979
```

In [63]:
```
# Corrección pregunta A - Era acumulada a 4, no puntual
res = (200-94-5)/200
resFinalParser(res)
```

```
0,505
```

Ejercicio 7

Pregunta 3
Correcta
Puntúa 7,00 sobre 7,00
Marcar pregunta

(7 puntos) El tiempo de demora en minutos en atender a cada paciente que entra en la guardia de un conocido sanatorio es una variable aleatoria Normal con desvío estándar $\sigma = 83$ minutos. El tiempo medio de demora que declara el sanatorio desde que ingresa el paciente hasta que es dado de alta o derivado es de 210 minutos. Los pacientes han realizado múltiples quejas manifestando que el tiempo de demora es superior al informado oficialmente y por ello han presentado múltiples demandas al departamento de control correspondiente. Los encargados en estudiar el caso desean realizar un test de hipótesis para analizar si dan lugar a los reclamos e inician un sumario contra el sanatorio por ello. Toman en forma aleatoria los tiempos de demora de 36 pacientes obteniendo un promedio muestral de 236.28 minutos de demora. Se quiere que la probabilidad de dar lugar a la demanda cuando en realidad el sanatorio informa correctamente sea 0.03. Calcular la probabilidad de no demandar al sanatorio cuando en realidad el tiempo medio de demora es de 213 minutos.

Respuesta: 0,9519 ✓

$\mu =$ ``tiempo medio de demora en minutos en atender a un paciente de la guardia"

Muestra: $X_1, \ldots, X_{36}$ v. a. i. i. d. $N(\mu, 83^2)$.

$H_0 : \mu = 210$ vs $H_A : \mu > 210$.

Es un test exacto.

Estadístico de prueba: $Z = \frac{\overline{X}_{36} - 210}{83/\sqrt{36}}$.

Si H_0 es verdadera, $Z \sim N(0, 1)$.

$RR = \{Z \geq z_\alpha\} = \{Z \geq 1.88\}$ ya que $1 - \Phi(1.88) = 0.03$.

También se puede escribir $RR = [1.88, +\infty)$.

El valor observado del estadístico de prueba es:

$Z_{obs} = 1.8998$, luego como Z_{obs} pertenece a RR entonces rechazamos la hipótesis nula a nivel 0.03.

Se pide $\beta(213) = P(\text{no rechazar } H_0 | \mu = 213)$. Entonces:

$\beta(213) = P(Z < 1.88 | \mu = 213) = P\left(\frac{\overline{X}_{36} - 210}{83/\sqrt{36}} < 1.88 | \mu = 213\right)$

$= P\left(\frac{\overline{X} - 213}{83/\sqrt{36}} < 1.88 + \frac{210 - 213}{83/\sqrt{36}}\right) = \Phi\left(1.88 + \frac{210 - 213}{83/\sqrt{36}}\right) = 0.9515.$

La respuesta correcta es: 0,9515

Defino X = "Tiempo de demora en minutos en atender a un paciente"

$X \sim N(\mu, 83^2)$

Datos para el test de hipótesis:

- $\mu_0 = 210$
- $n = 36$
- $\overline{X_n} = 236.28$
- $\alpha = 0.03$
- $\sigma = 83$

Defino el test de hipótesis como:

$H_0 \Rightarrow \mu_0 = 210$ $H_a \Rightarrow \mu > \mu_0$

Defino el pivote T como:

Datos datos normales, sigma conocido, mu desconocido $\Rightarrow T = Normal(0, 1)$

Me piden:

Probabilidad de no demandar al sanatorio, cuando en realidad el tiempo medio de demora es 213 minutos

Error de tipo II = No rechazar H0 cuando H0 es falsa, en este caso, no demandar al sanatorio cuando H0 es falsa

Luego busco el error de tipo 2 cuando $\mu = 213$

In [74]:
```
# Datos
alfa = 0.03
```

```
mu_0 = 210
mu = 213
sigma = 83
n = 36

va = scipy.stats.norm()
const = (mu_0 - mu)/(sigma/math.sqrt(n))
z = va.ppf(1-alfa)
k = va.cdf(z+const)
resFinalParser(k)
```

```
0,9519
```

Ejercicio 8

Pregunta 4
Correcta
Puntúa 7,00 sobre 7,00
Marcar pregunta

(7 puntos) Un futbolista afirma que la velocidad promedio de sus tiros libres es 94 km/h. Su entrenador sospecha que es mayor para lo cual realiza un test de hipótesis suponiendo que la velocidad de un tiro libre tiene distribución normal. Toma una muestra aleatoria de 29 datos y obtiene una desviación estándar muestral igual a 2.59. Hallar el valor de la media muestral sabiendo que 0.005 es el menor nivel de significación al cual se rechaza H_0 en base a esta muestra dada.

Respuesta: 95,3290

$\bar{X}_{obs} = 95.329$

La respuesta correcta es: 95,329

Datos

- $\mu_0 = 94$
- $n = 29$
- $sd = 2.59$
- $\alpha = 0.005$

Test de tipo 1 - $\mu > \mu_0$

Definición del pivote:

$T = \frac{\overline{X}_n - \mu_0}{sd/\sqrt{n}} \sim T_{n-1}$

Busco el z limite para el test $\Rightarrow$

In [58]:
```
n = 29
alfa = 0.005
va = scipy.stats.t(df=n-1)
z = va.ppf(1-alfa)
```

In [61]:
```
s = 2.59
n = 29
mu_0 = 94
XnRaya = ((z*s)/math.sqrt(n))+mu_0
resFinalParser(XnRaya)
```

```
95,329
```