Parcial integrador turno mañana 2do 2021

Se hizo en dos bloques con tres preguntas cada uno, una hora cada bloque.

In [122…

```python
import numpy as np
import scipy.stats
import math
```

Ejercicio 1

Pregunta **1**

Respuesta guardada

Puntúa como 10,00

Marcar pregunta

Considere el vector aleatorio (X, Y) con densidad de probabilidad conjunta $f(x,y) = 2xe^{-2x}e^{-xy}I_{[0,+\infty)}(x)I_{[0,+\infty)}(y)$

a. (4 puntos) Calcular la probabilidad de que $X \geq 0.56$ 0,3263

b. (4 puntos) Calcular $P(X \geq 0.56, Y \geq 1.79)$ 0,0632

c. (2 puntos) Calcular $P(Y \geq 1.79 | X \geq 0.56)$ 0,1937

In [123…

```python
e = np.e
def f(x,y):
    if (0 <= x < np.inf) and (0 <= y < np.inf):
        return 2*x*e**(-2*x)*e**(-x*y)
    else:
        return 0
```

In [124…

```python
# Para hacer integrales dobles
scipy.integrate.dblquad(f, -np.inf, np.inf, -np.inf, np.inf)
```

Out[124…

```
(0.9999999999999338, 1.3748788567325646e-08)
```

In [125…

```python
# Pregunta A
a = scipy.integrate.dblquad(f, -np.inf, np.inf, 0.56, np.inf)[0]
a
```

Out[125…

```
0.3262797946229661
```

In [126…

```python
# Pregunta B
b = scipy.integrate.dblquad(f, 1.79, np.inf, 0.56, np.inf)[0]
b
```

Out[126…

```
0.06318939209365672
```

In [127…

```python
# Pregunta C
c = b/a
c
```

Out[127…

```
0.19366627396181696
```

Ejercicio 2

Pregunta **2**

Sin responder aún

Puntúa como 10,00

Marcar pregunta

Una fábrica produce arandelas (piezas en forma de anillo) de metal cuyo diámetro sigue una distribución normal de media 33 mm y desvío estándar 5 mm. Las arandelas se empacan en cajas de 60 unidades. Una arandela se considera inutilizable si su diámetro supera los 39.5 mm. Si en una caja de arandelas el 10% o más son inutilizables esta caja se descarta para la venta.

Suponer que los diámetros son independientes.

a. (2 puntos) Calcular la probabilidad de que una arandela sea inutilizable. 0.0968

b. (4 puntos) Calcular la probabilidad de que una caja sea descartada para la venta. 0,5298

c. (4 puntos) Se realiza una inspección de las cajas. Calcular la probabilidad de se tengan que revisar como mucho 5 cajas para encontrar que 1 de ellas debe ser descartadas. 0,9770

Defino X = "Diámetro de la arandela"

$X \sim N(33, 25)$

Arandela inutilizable $\iff X > 39.5$

Arandelas por caja = 60

Defino Y = "Cantidad de arandelas defectuosas en una caja"

$Y \sim Bin(60, p)$ con $p = P(X > 39.5)$

Caja descartada para la venta $\iff Y \geq 6$

In [128…

```python
media = 33
sd = math.sqrt(25)
va = scipy.stats.norm(loc=33, scale=sd)
```

Pregunta A

Calcular $P(X > 39.5) = 1 - P(X < 39.5)$

In [129…

```python
p = 1-va.cdf(39.5)
p
```

Out[129…

```
0.0968004845856103
```

Pregunta B

Calcular $P(Y \geq 6) = 1 - P(Y < 6)$

In [130…
```
va = scipy.stats.binom(n=60,p=p)
a = 1-va.cdf(5)
a
```
Out[130…
```
0.5298433638337553
```

Pregunta C

Se realiza una inpección de las cajas. Calcular la probabilidad de que se tengan que revisar como mucho 5 cajas para encontrar que 1 de ellas debe ser descartada.

Defino Z = "Cantidad de cajas revisadas hasta encontrar una que debe ser descartada"

$Z \sim Geo(\alpha)$ con $\alpha = P(Y \geq 6)$

Luego busco $P(Z \leq 5)$

In [131…
```
va = scipy.stats.geom(a)
va.cdf(5)
```
Out[131…
```
0.9770272570924351
```

Ejercicio 3

Pregunta **3**
Sin responder aún
Puntúa como 10,00
Marcar pregunta

Una reserva natural de agua sabe que la cantidad de sodio en mg/L presente en el agua sigue una distribución normal con desvío estándar 9.4.

Además afirma que la cantidad media de sodio en el agua es 77 mg/L. Sin embargo, Una compañía que vende agua embotellada sospecha que la cantidad media de sodio es mucho mayor al declarado por esta reserva.

La compañía decide realizar un test de hipótesis para analizar si da lugar dejar de proveerse de esta reserva. Para ello, toman una muestra aleatoria de tamaño 43 obteniéndose una media muestral de 79.58 mg/L.

La compañía quiere que la probabilidad de dejar de suplirse de esta reserva cuando en realidad la reserva informa correctamente la cantidad media de sodio, sea 0.03.

a. (1 punto) Sobre la hipótesis nula de la forma $H_0 : \mu = \mu_0$. Responda 1 si $\mu_0 = 77$ o responda 2 si $\mu_0 = 79.58$. 1

b. (1 punto) Sobre la hipótesis alternativa H_1. Responda 1 si $H_1 : \mu > \mu_0$, responda 2 si $H_1 : \mu < \mu_0$ o responda 3 si $H_1 : \mu \neq \mu_0$. 1

c. (1 puntos) En el test propuesto, ¿cuál es la probabilidad de error de tipo I? 0.03

d. (2 puntos) Con base en la muestra observada y el test propuesto, indique qué decisión toma la compañía. Responda 1 si rechaza H_0 o responda 2 si NO rechaza H_0. 2

e. (2 puntos) Calcule el p-valor. 0.0359

f. (3 puntos) Calcule la probabilidad de que la compañía siga obteniendo el agua de esta reservaa cuando en realidad la cantidad media de sodio verdadero es de 80 mg/L. 0,4161

Defino X = "Cantidad de sodio (mg/L) presente en el agua" con $X \sim N(\mu, 9.4^2)$

Datos para el test de hipótesis:

- $\mu_0 = 77$
- $n = 43$
- $\overline{X_n} = 79.58$
- $\sigma = 9.4$
- $\alpha = 0.03$

Planteo del test de hipótesis:

$H_0 \Rightarrow \mu_0 = 77$ $H_a \Rightarrow \mu > \mu_0$

Definición del pivote:

Datos normales, sigma conocido, media desconocida $\Rightarrow T = \frac{\overline{X_n} - \mu_0}{\sigma / \sqrt{n}} \sim N(0, 1)$

Pregunta A

$\mu_0 = 77 \Rightarrow 1$

Pregunta B

$\mu > \mu_0 \Rightarrow 1$

Pregunta C

Proba error de tipo I = $\alpha \Rightarrow 0.03$

Pregunta D

In [132…
```
va = scipy.stats.norm()
t = (79.58 - 77)/(9.4/math.sqrt(43))
z = va.ppf(0.97)
t > z
```
Out[132…
```
False
```

In [133…
```
def rechazarH0NormalesParaMuConVar(tipo, alfa, mediaMuestral, mu_0, sigma, n):
    """
    Tipo 0: mu < mu_0
    Tipo 1: mu > mu_0
    Tipo 2: mu != mu_0
    """
    va = scipy.stats.norm()
    k = (mediaMuestral - mu_0)/(sigma/math.sqrt(n))
    print("mu = {}".format(k))
    if tipo == 0:
        z = -va.ppf(1-alfa)
```

```python
        print("z = {}".format(z))
        if k < z:
            return True
        else: return False
    elif tipo == 1:
        z = va.ppf(1-alfa)
        print("z = {}".format(z))
        if k > z:
            return True
        else: return False
    else:
        z = va.ppf(1-alfa/2)
        print("z = {}".format(z))
        if (k < -z) or (k > z):
            return True
        else: return False
```

In [134…

```python
# Uso rechazarH0NormalesParaMuConVar
tipo = 1
alfa = 0.03
mediaMuestral = 79.58
mu_0 = 77
sigma = 9.4
n = 43
rechazarH0NormalesParaMuConVar(tipo, alfa, mediaMuestral, mu_0, sigma, n)
```

```
mu = 1.7998075949679946
z = 1.8807936081512509
```

Out[134…

```
False
```

Luego no se rechaza la $H_0 \Rightarrow 2$

Pregunta E

Calcular el p-value.

In [135…

```python
1-va.cdf(t)
```

Out[135…

```
0.035945512151310255
```

Pregunta F

Calcule la probabilidad de que la compañía siga obteniendo agua de esta reserva cuando en realidad la cantidad media de sodio verdadero es de 80mg/L

$\mu = 80 \Rightarrow H_0$ es falsa

Luego si la empresa sigue comprando, implica un error de tipo II = no rechazar la H_0 cuando esta es falsa

Luego busco la proba de error de tipo II

In [136…

```python
def probaErrorTipoII(tipo, alfa, mu_0, mu, sigma, n):
    """
    Tipo 0: mu < mu_0
    Tipo 1: mu > mu_0
    Tipo 2: mu != mu_0
    """
    va = scipy.stats.norm()
    const = (mu_0 - mu)/(sigma/math.sqrt(n))
    if tipo == 0:
        z = -va.ppf(1-alfa)
        return 1 - va.cdf(z+const)
    elif tipo == 1:
        z = va.ppf(1-alfa)
        return va.cdf(z+const)
    else:
        z = va.ppf(1-alfa/2)
        return va.cdf(z+const) - va.cdf(-z+const)
```

In [137…

```python
# Uso error proba tipo II
tipo = 1
alfa = 0.03
mu_0 = 77
mu = 80
sigma = 9.4
n = 43
probaErrorTipoII(tipo, alfa, mu_0, mu, sigma, n)
```

Out[137…

```
0.41605120756169117
```

Ejercicio 4

Pregunta **1**

Sin responder aún

Puntúa como 10,00

Marcar pregunta

El gasto semanal en dólares por hogar en una cierta ciudad sigue una distribución desconocida. Se quiere dar un estimador en forma de intervalo para la media y la varianza de esta distribución.

Para esto se hace una encuesta 63 jefes de hogar obteniéndose una media muestral de 104 y varianza muestral de 87.36

Atención Usar la convención de dar el intervalo que deja igual cantidad de área a cada lado.

a. (2.5 puntos) Construir un intervalo de confianza de nivel asintótico 92% para la media de esta distribución. ¿Cuál es el extremo izquierdo de este intervalo estimado? 101,9384

b. (2.5 puntos) ¿Cuál es el extremo derecho de este intervalo estimado? 106,0616

c. (2.5 puntos) Si suponemos que los datos siguen una distribución $\mathcal{N}(\mu, \sigma^2)$ pero desconocemos sus parámetros. Construir un intervalo de confianza de nivel EXACTO 92% para σ^2. Dar el extremo izquierdo del intervalo estimado. 65,4143

d. (2.5 puntos) Dar el extremo derecho del intervalo estimado. 123,2450

Datos

- Distribución desconocida
- $n = 63$
- $\overline{X_n} = 104$
- $\text{varMuestral} = 87.36$
- $sd = \sqrt{\text{varMuestral}}$
- $\alpha = 0.08$

Planteo del intervalo de confianza:

Dado que los datos provienen de una distribución desconocida, uso el TCL.

$IC = [\overline{X_n} - z_{\alpha/2}\frac{sd}{\sqrt{n}}, \overline{X_n} + z_{\alpha/2}\frac{sd}{\sqrt{n}}]$

Pregunta A y B

In [138…

```
def getIntervaloParaMuConVar(alfa, var, mediaMuestral, n):
    va = scipy.stats.norm(0,1)
    z = va.ppf(1-(alfa/2))
    sd = math.sqrt(var)
    return (mediaMuestral - (z*sd/math.sqrt(n)),(mediaMuestral + (z*sd/math.sqrt(n))))
```

In [139…

```
alfa = 0.08
var = 87.36
mediaMuestral = 104
n = 63
getIntervaloParaMuConVar(alfa, var, mediaMuestral, n)
```

Out[139…

```
(101.9384479022253, 106.0615520977747)
```

Pregunta C y D

In [140…

```
def getIntervaloParaVarSinMu(alfa, desvioMuestral, n):
    va = scipy.stats.chi2(df=n-1)
    xMin = va.ppf(1-a/2)
    xMax = va.ppf(a/2)
    ter = (n-1)*(desvioMuestral**2)
    return (ter/xMin, ter/xMax)
```

In [141…

```
alfa = 0.08
desvioMuestral = math.sqrt(87.36)
n = 63
getIntervaloParaVarSinMu(alfa, desvioMuestral, n)
```

Out[141…

```
(79.01054377115423, 99.12172781391477)
```

Ejercicio 5

Pregunta **2**
Respuesta incompleta
Puntúa como 10,00
Marcar pregunta

El peso en kg de un tipo de pez es una variable aleatoria Y de una cierta distribución. La ley de pesca prohibe pescar especímenes cuy peso esté por debajo (estrictamente) del valor crítico 3.84kg, en este caso el pescador devuelve el pez al río.

Considerar la variable aleatoria X que vale 1 si un pez se devuelve al río y 0 sino.

Se extrae una muestra aleatoria de los pesos 23 peces obteniéndose los siguientes datos:

$Y = (5.11, 4.31, 6.27, 5.36, 5.4, 2.62, 4.09, 5.54, 3.23, 4.49, 4.01, 5.24, 4.51, 5.19, 3.85, 6.15, 4.39, 4.96, 4.4, 3.43, 4.59, 5.58, 4.89)$

a. (1 punto) Usando la distribución empírica estimar $P(Y \leq 4.85)$ 0.5217

b. (1 punto) ¿Qué distribución sigue la variable X? (1: Normal, 2: Bernoulli, 3: Binomial, 3: Poisson, 5: Otra) 2

c. (3.5 puntos) Si p es el parámetro de la distribución de X, calcular el estimador de máxima verosimilitud de p, $\hat{p}$ y en base a las observaciones dar una estimación de p. 0,1304

d. (1.5 puntos) Responder con V o F según corresponda. $\hat{p}$ es asintóticamente insesgado pero no consistente F

e. (1.5 puntos) Responder con V o F según corresponda. $\hat{p}$ es consistente V

f. (1.5 puntos) Responder con V o F según corresponda. $\hat{p}$ es insesgado V

In [142…

```
datos = np.array([5.11,4.31,6.27,5.36,5.40,2.62,4.09,5.54,3.23,4.49,4.01,5.24,\
         4.51,5.19,3.85,6.15,4.39,4.96,4.40,3.43,4.59,5.58,4.89])
```

In [143…

```
# Pregunta A
(datos <= 4.85).mean()
```

Out[143…

```
0.5217391304347826
```

Pregunta B

X = "Devolver un pez al río o no" $X \sim Ber(p)$

El EMV de una $Ber(p) = \overline{X_n}$

Pregunta C

In [144…

```
# Busco la distribución empirica de X
dx = (datos < 3.84)
```

In [145…

```
dx.mean()
```

Out[145…

```
0.13043478260869565
```

Ejercicio 6

Pregunta **3**

Respuesta incompleta

Puntúa como 10,00

Marcar pregunta

El ingreso anual en millones de dólares de una persona en un cierto sector de la población es una variable aleatoria $Y = \frac{3}{X}$, donde X es una variable aleatoria continua con función de densidad dada por $f_X(x) = \frac{3}{x^4}\mathbf{1}_{[1,\infty)}(x)$.

a. (4 puntos) Calcular la esperanza de Y. 2,25

b. (4 puntos) Calcular la varianza de Y. 0,3375

c. (2 puntos) Se extrae una muestra aleatoria de 129 personas de dicho sector. ¿Cuál es la probabilidad APROXIMADA de que en promedio superen los 2.32 millones de dólares. 0,0856

In [146…

```
def fe(x):
    if 1 <= x < np.inf:
        return (1/x)*(3/(x**4))
    else: return 0
```

In [147…

```
# Pregunt A
a = -np.inf
b = np.inf
esp = 3*scipy.integrate.quad(fe, a, b)[0]
esp
```

Out[147… 2.2499999999999996

In [148…

```
def fvar(x):
    if 1 <= x < np.inf:
        return (1/(x**2)) * (3/(x**4))
    else: return 0
```

In [149…

```
# Pregunta B
a = 1
b = np.inf
var = 9*scipy.integrate.quad(fvar, a, b)[0] - (esp**2)
var
```

Out[149… 0.33750000000000124

In [150…

```
# Pregunta C
# 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 [151…

```
suma = 2.32 * 129
TCL_suma(esp, var, 129, suma)
```

Out[151… 0.085572288961557