Tarea 2: Variables aleatorias

Marco Chacón Soto

Resumen
Este es un trabajo acerca del análisis de probabilidad para variables aleatorias en el cuál se responden diferentes asignaciones con base en los datos de mediciones de una variable física.

Palabras claves
Python — probabilidad — variable aleatoria — función acumulativa — función de densidad

Escuela de Ingeniería Eléctrica, Universidad de Costa Rica
Correspondencia: marco.chaconsoto@ucr.ac.cr

Índice

Introducción

Con el objetivo de aprender más sobre el uso de Git, se creó un usuario en la plataforma Github, luego se clonó un repositorio asignado por el profesor y se hizo fork al repositorio, haciendo un edit al archivo himnovaciones.txt, al cambiar la palabra dsknsa por descansa.

Luego, se analizaron cinco mil datos correspondientes a mediciones de una variable física. Con estos datos, inicialmente se creó un histograma y se analizó qué curva tenía el mejor ajuste, se escogió un ajuste de curva Rayleigh y se calculó la probabilidad en un intervalo [a, b] y se comparó esta con los datos en el documento. Seguidamente se calcularon los primeros cuatro momentos para este modelo y se realizó una transformación Y = sqrt(X), donde X representa los datos originales de las mediciones. Finalmente, se encontró una expresión para función de densidad de Y a partir de la curva de ajuste del histograma de Y.

1. Marco Teórico

Algunas definiciones importantes son:

1. Git: es un sistema de control de versiones gratuito y de código abierto, diseñado para manejar proyectos. Este sistema permite a múltiples los usuarios contribuir a dichos proyectos y trabajar de manera más eficiente y controlada. [1]

2. Variable aleatoria: facilitan una manipulación numérica más robusta de los fenómenos aleatorios mediante herramientas como las funciones de densidad y de distribución acumulativa de probabilidad, las cuales proveen mejores descripciones de los modelos probabilísticos a estudiar. [2]

3. Probabilidad en un intervalo: se puede obtener al calcular la función de distribución acumulativ en ambos extremos del intervalo y restarlos, como se muestra en la siguiente ecuación:

$$P(x_1 < X < x_2) = F_X(x_2) - F_X(x_1) \quad (1)$$

4. Primeros cuatro momentos de las variables aleatorias:
   - Media: es el valor más esperado en un rango de valores medidos.[3]
   - Varianza: es una medida de la disperción alrededor de la media.[3]
   - Inclinación: es un número adimensional que describe la inclinación o el sesgo de la función de distribución acumulativa. Si es cero indica que es simétrica, si es menor a cero indica que tiende a la izquierda y si es mayor a cero indica que tiende a la derecha.[3]
   - Kurtosis: es un número adimensional descriptor del abultamiento de la variable aleatoria. Si es menor a cero indica que el abultamiento es achatado y cuando es mayor a cero indica que es prominente.[3]

2. Resultados

2.1 Histograma con los datos contenidos en el archivo datos.csv

Para generar el histograma se implementa el siguiente código, el cuál lee el archivo y guarda los valores flotantes en un array llamado res. Luego se crea el histograma con los datos en este array como se muestra en el siguiente código

```
# Librerías
import csv
import numpy as np
from scipy import stats
from scipy.stats import norm
from scipy.stats import rayleigh
from scipy.stats import moment
import matplotlib.pyplot as plt
import matplotlib.mlab as mlab

res = []
with open('datos.csv') as csvfile:
    for line in csvfile:
        line = line.strip()
        if line != '':
            try:
                res.append(float(line))
            except ValueError:
                pass

print(res)

# Crear histograma
a, bins, c = plt.hist(res, 200, density
= 1, facecolor='blue', alpha=0.5)
# Se agregan labels, grid, titulo y se
# guarda la figura
plt.xlabel('Magnitud')
plt.ylabel('Frecuencia')
plt.grid(True)
plt.title(r'Histograma de mediciones de
una  variable fisica')
plt.savefig('res4.png')
```

Código 1. Código para la pregunta 4.

En la figura 1 se muestra el histograma

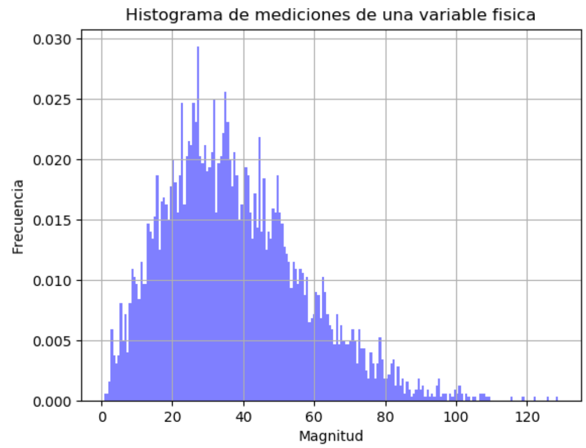

Figura 1. Histograma de mediciones de una variable física

2.2 Mejor curva de ajuste y gráfica de esta con los datos contenidos en datos.csv

De la figura 1 se distingue que el histograma tiene forma de una curva Rayleigh. A partir de esta observación se toma un ajuste de rayleigh y se grafica dicha curva con el siguiente código

```
(mu, sigma) = stats.rayleigh.fit(res)
rayleigh = stats.rayleigh(mu, sigma)
y = stats.rayleigh.pdf(bins, mu, sigma)
plt.plot(bins, y)
cdf = stats.rayleigh.cdf(bins, mu, sigma)
plt.savefig('res5.png')
```

Código 2. Código para la pregunta 5.

La curva de mejor ajuste para el histograma se muestra en la figura 2

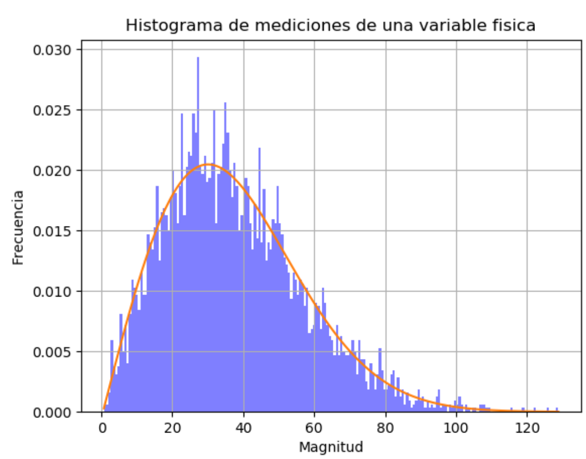

Figura 2. Curva de mejor ajuste para el histograma

2.3 Probabilidad en el intervalo [18, 68] en el modelo encontrado y contraste con la frecuencia relativa de los elementos de datos.csv que están en realidad en ese mismo intervalo.

La probabilidad en este intervalo se calcula a partir del CDF, para esto se realiza la operación $cdf(68) - cdf(18)$. Esto se implementa en python con la librería stats.rayleigh como se muestra en el siguiente código

```
print("La probabilidad en el intervalo
[18, 68] es:
", rayleigh.cdf(68) - rayleigh.cdf(18))
contrast = 0
for i in range(0, len(res)):
    if res[i] <= 68 and res[i] >= 18:
        contrast += 1

contraste = float(contrast)/len(res)

print("La probabilidad relativa de este
intervalo es: ", contraste)
```

Código 3. Código para la pregunta 6.

A partir de esto se obtienen los siguientes resultados:

La probabilidad en el intervalo [18, 68] es: 0.76473448903379859
La probabilidad relativa de este intervalo es: 0.7674

2.4 Primeros momentos, el "significado" de cada uno y la correspondencia con la gráfica observada y los valores teóricos.

Para calcular la media y la varianza se utilizó la función rayleight.stats(), sin embargo los resultados de inclinación y kurtosis obtenidos con este método eran aproximaciones y no los reales obtenidos con los datos del problema. Por esto se utilizaron las funciones stats.skew() y stats.kurtosis para obtener la inclinación y la kurtosis a partir de los datos reales del problema. Además, a partir de la inclinación se obtuvo la desviación estándar con el objetivo de obtener la medida de la dispersión de la función alrededor de la media. En el siguiente código se muestra dicha implementación

```
mean, var = rayleigh.stats(moments='mv')
skew = stats.skew(res)
kurt = stats.kurtosis(res)
desv = np.sqrt(skew)
print("Los primeros cuatro momentos y
la desviacion:")
print("     Mean: ", float(mean))
print("     Var: ", float(var))
print("     Desv: ", float(dev))
print("     Skew: ", float(skew))
print("     Kurt: ", float(kurt))
```

Código 4. Código para la pregunta 7.

Obteniendo los siguientes resultados:

Media: 37.572999385863326
Varianza: 376.6130040184076
Inclinación: 0.7121383888062807
Desviación: 0.8438829236370888
Kurtosis: 0.4740060737512528

El valor de la media obtenido concuerda con lo esperado al observar la figura 2, ya que este número cercano a 40 se puede considerar como un punto donde la suma de las distancias desde este a todos los puntos en este conjunto es la más pequeña.

La varianza nos indica la dispersión alrededor de la media, que se puede observar es muy pequeña. Además se analiza la desviación estándar ya que esta y la varianza están íntimamente relacionados, nuevamente en la figura 2 se observa que al rededor de la media es muy dispersa, lo cuál concuerda con los resultados obtenidos.

Por otra parte, una inclinación positiva, como la que se obtuvo en los resultados, indica que alrededor de la media, la curva se va a sesgar a la derecha tal y como se observa en la figura.

Finalmente, la kurtosis es una medida del abuntamiento, y al tener un valor positivo, indica que posee un abultamiento prominente, esto se puede observar en el histograma con la curva de mejor ajuste, particularme alrededor de la media.

2.5 Transformación Y = sqrt(X)

En el array resy se encuentran los datos de las mediciones, se pasan todos los datos dentro de este array por la transformación Y = sqrt(X) de la siguiente manera:

```
resultS = []
for numero in resy:
        resultS.append(np.sqrt(numero))
a, bins, c = plt.hist(resultS, 100, facecol
plt.xlabel('Magnitud')
plt.ylabel('Frecuencia')
plt.grid(True)
plt.title(r'Histograma de mediciones de la
plt.savefig('resfinal.png')
```

Código 5. Código para la pregunta 8.

Al ejecutar este código se obtiene la figura 3

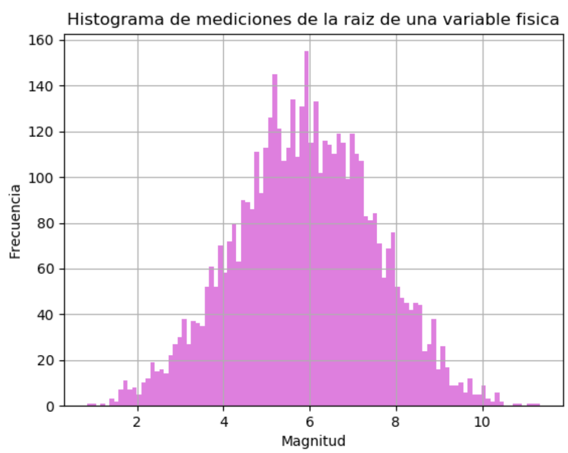

Figura 3. Histograma de mediciones de la raiz de una variable fisica

En este caso, en el histograma se observa menor cantidad de datos en valores pequeños y grandes de magnitud y una mayor concentración de estos cerca de la media. Por esto se puede considerar que la mejor curva de ajuste sería la de una distribución Gausseana. En la figura 4 se observa esta curva al realizar un ajuste normal

```
(mu, sigma) = stats.norm.fit(resultS)
normalfit = stats.norm.pdf(bins, mu, sigma)
print("mu: ", mu)
print("sigma: ", sigma)
plt.plot(bins, normalfit)
plt.savefig('res72.png')
```

Código 6. Código para la pregunta 8.

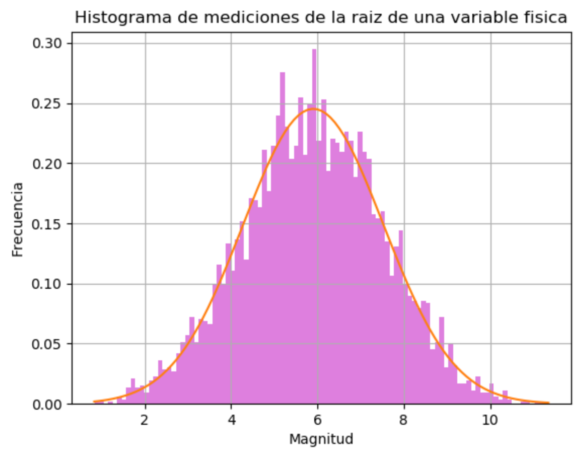

Figura 4. Curva de mejor ajuste para el histograma de la transformada

2.6 Expresión para la función de densidad de Y

Dado al ajuste normal realizado al histograma de la figura ??, es posible modelar la función de densidad normalizada como:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} exp\left[\frac{-(x-\mu_x)^2}{2\sigma_x^2}\right] \tag{2}$$

Donde σ_x y μ_x son constantes reales, conocidas como desviación estándar y media, respectivamente. Estos valores se obtienen en el código 6, obteniendo mu: 5.9063850448899702 y sigma: 1.6267631301145551.

3. Conclusiones

- Se logró graficar un histograma a partir de los datos obtenidos de mediciones de una variable física.
- Se implentó la curva de mejor ajuste para este histograma, a partir de un ajuste de Rayleigh.
- Se calcularon los primeros cuatro momentos para los datos obtenidos y se verificaron los resultados con el histograma.
- Se graficó el histograma de la transformada, se obtuvo la curva de mejor ajuste y se obtuvo la función de densidad de los datos obtenidos de las mediciones.

Bibliografía

[1] S. F. Conservancy, *Git.* [Página Web]. Disponible en: https://git-scm.com/. Accesado el 3 de junio del 2020.

[2] F. A. Calderón, *Variables aleatorias: Definición, propiedades y funciones.* Disponible en: https://mv1.mediacionvirtual.ucr.ac.cr/mod/page/view.php?id=609734. Accesado el 3 de junio del 2020.

[3] F. A. Calderón, *Valor esperado y momentos de una variable aleatoria.* Disponible en: https://mv1.mediacionvirtual.ucr.ac.cr/mod/page/view.php?id=610882. Accesado el 3 de junio del 2020.