4.1.2

Determine las funciones de densidad marginales 𝑓𝑋(𝑥)fX(x) y 𝑓𝑌(𝑦)fY(y) a partir de los datos y utilizando un modelo de mejor ajuste.

4.1.3

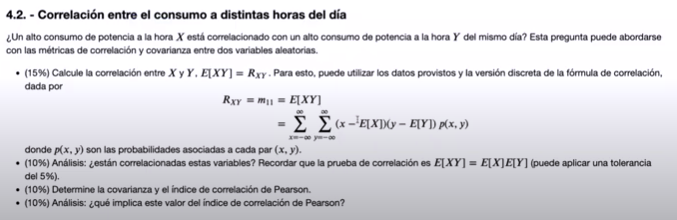
Con las densidades marginales del punto anterior se pueden encontrar los valores esperados

Con la densidad marginal se tiene el modelo y de ahí se puede extraer el promedio

Los valores esperados de los datos (o del modelo) E[X] y E[Y]

Parte4.2

4.2.1



Rxy=xyp o en format xy\*p(x,y)

E[X] y E[Y] = E[X\*Y]

El profe dice que tiene que ser alrededor de cero y no de la media

xyp = [xbins, ybins, hist[i][j]]

def correlacion (xbins, ybins, hist):

for i in range(bins) for j in range(bins):

correlacion = x\*y\*p

correlacion = correlacion + 1

print(correlacion)

4.2.2

4.2.3

4.2.4

#Calcule la correlación entre 𝑋 y 𝑌 , 𝐸[𝑋𝑌]=𝑅𝑋𝑌.

import numpy as np

def extraer\_datos(archivo\_json, hora):

'''Importa la base de datos completa y devuelve los

datos de potencia a la hora indicada en un

array de valores.

'''

# Cargar el "DataFrame"

df = pd.read\_json(archivo\_json)

# Convertir en un array de NumPy

datos = np.array(df)

# Crear vector con los valores demanda en una hora

demanda = []

# Extraer la demanda en la hora seleccionada

for i in range(len(datos)):

instante = datetime.fromisoformat(datos[i][0]['fechaHora'])

if instante.hour == hora:

demanda.append(datos[i][0]['MW'])

return demanda

import numpy as np

def extraer\_datos(archivo\_json, hora):

'''Importa la base de datos completa y devuelve los

datos de potencia a la hora indicada en un

array de valores.

'''

# Cargar el "DataFrame"

df = pd.read\_json(archivo\_json)

# Convertir en un array de NumPy

datos = np.array(df)

# Crear vector con los valores demanda en una hora

demanda = []

# Extraer la demanda en la hora seleccionada

for i in range(len(datos)):

instante = datetime.fromisoformat(datos[i][0]['fechaHora'])

if instante.hour == hora:

demanda.append(datos[i][0]['MW'])

return demanda

import random

digitos = 43333

def horas\_asignadas(digitos):

'''Elige una hora A en periodo punta

y una hora B de los otros periodos,

con los dígitos del carné como "seed"

'''

random.seed(digitos)

punta = [11, 12, 18, 19, 20]

valle = [7, 8, 9, 10, 13, 14, 15, 16, 17]

nocturno = [21, 22, 23, 0, 1, 2, 3, 4, 5, 6]

otro = valle + nocturno

HX = punta[random.randrange(0, len(punta))]

HY = otro[random.randrange(0, len(otro))]

horas = 'Mi hora X: {}, Mi hora Y: {}'.format(HX, HY)

return horas

mis\_horas = horas\_asignadas(digitos)

hora1 = 18

hora2 = 8

# Se eligen las dos horas que desean estudiarse

hora\_1 = extraer\_datos('demanda\_2019.json', hora1)

hora\_2 = extraer\_datos('demanda\_2019.json', hora2)

#4.2.3 Determine la covarianza y el índice de correlación de Pearson.

'''Con base en la formula, la covarianza viene dada por:

Cov = (18-1457.978549019608\*10^6)\*(8-1330.9543921568627\*10^6)\*(0.99998) = 1.9404\*10^18'''