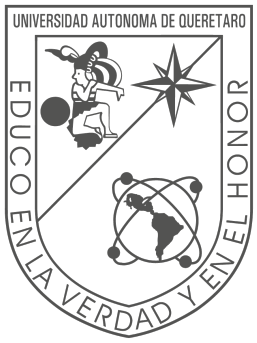
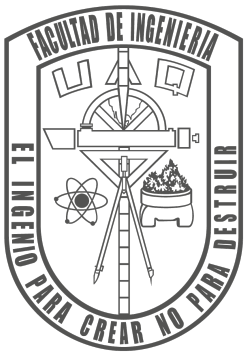
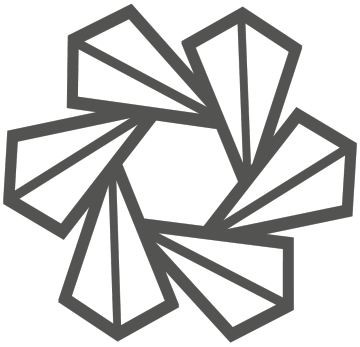
Universidad Autónoma de Querétaro

Facultad de Ingeniería
División de Investigación y Posgrado







Práctica 5

Pruebas Estadísticas de Hipótesis

Maestría en Ciencias en Inteligencia Artificial
Optativa de especialidad IV - Machine Learning

Aldo Cervantes Marquez
Expediente: 262775
Profesor: Dr. Marco Antonio Aceves Fernández

Santiago de Querétaro, Querétaro, México
Semestre 2023-1
9 de Junio de 2023

Índice

1. Objetivo

Esta práctica tiene como objetivo el de realizar comparaciones entre conjuntos de datos para conocer algunas características de similitud y comprobar hipótesis que consista en aceptar o rechazar una afirmación sobre una población (conjunto de datos). Todo esto con el fin de poder obtener más información y contraste entre los datos, lo que permitirá tomar decisiones en el procesamiento y elección de modelos de machine learning.

2. Introducción

Esta práctica consiste en realizar pruebas de hipótesis en diferentes bases de datos con diferentes características, realizando preprocesamiento de datos si es requerido, y obteniendo afirmaciones o negaciones de las muestras de los datos. Se ocuparán diferentes métodos de pruebas de hipótesis, como los basados en pruebas de normalidad, pruebas de correlación, pruebas de estacionareidad, pruebas de hipótesis estadísticas paramétricas y pruebas de hipótesis estadísticas no paramétricas.

3. Marco Teórico

Una prueba estadística de hipótesis es una regla que especifica si se puede aceptar o rechazar una afirmación acerca de una población dependiente de la evidencia proporcionada [1]. Todas estas pruebas serán realizadas con funciones de Python y serán aplicadas a datos.

3.1. Pruebas de Hipótesis de normalidad

Estas pruebas consisten en conocer si una muestra pertenece a un conjunto de tipo de distribución normal o no lo es, desde un punto de vista más formal se define:

$$H_0 : \text{La distribución es normal}$$
$$H_1 : \text{La distribución no es normal}$$

Esto implica que:

$$\begin{aligned} H_0 &: X \backsim \mathcal{N}(\mu, \sigma^2) \\ H_1 &: X \nsim \mathcal{N}(\mu, \sigma^2) \end{aligned} \tag{1}$$

3.1.1. Prueba de Shapiro-Wilk

La prueba de Shapiro-Wilk es una forma de saber si una muestra aleatoria proviene de una distribución normal [2]. La prueba te da un valor W; los valores pequeños indican que su muestra no tiene una distribución normal (puede rechazar la hipótesis nula de que su población tiene una distribución normal si sus valores **están por debajo de cierto umbral** $U < W$) [3, 4, 5]. La fórmula para el valor de W es:

$$W = \frac{(\sum_{i=1}^{m} a_i x_{(i)})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{2}$$

Donde:

- n es la longitud de los datos
- $m = \frac{n}{2}$ si la longitud es par y $m = \frac{n-1}{2}$ si es impar
- x_i son los valores de muestra aleatorios ordenados
- a_i son las constantes generadas a partir de las covarianzas varianzas y medias de la muestra (tamaño n) de una muestra normalmente distribuida. Obteniendolo a partir de constantes y utilizando la tabla de pesos para n valores.
- $x_{(i)} = x_{n-i+1} - x_i$, lo que significa la diferencia entre cada extremo de los datos ordenados

En python se manda a llamar la función como:

```
from scipy.stats import shapiro
```

3.1.2. Prueba D'Agostino´s - Pearson

Esta prueba toma los valores de la curtosis y de la asimetría de la distribución. Como se observa en la Figura 1. Donde la asimetría da un grado de desviación del centroide natural de los datos, mientras que la curtosis explica la manera en la que los datos se concentran de una manera achatada o puntiaguda.

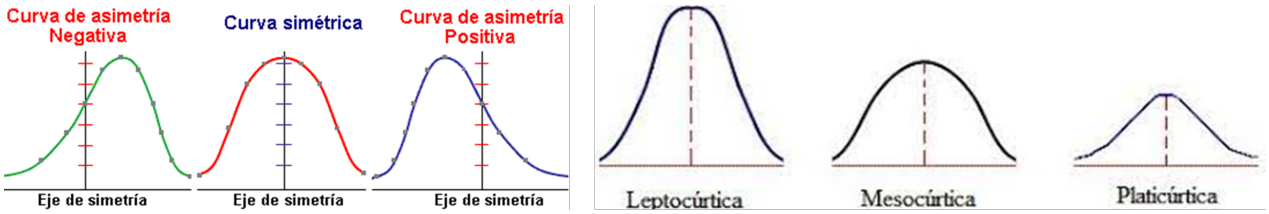


Figura 1: Asimetría y curtosis de una ditribución normal.

Se define a la asimetría como [6], teniendo variantes:

$$\begin{array}{cc} \text{Fórmula básica} & \text{Fórmula completa} \\ C_{as1} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})^3}{n\sigma^3} & C_{as2} = \frac{n\sum_{i=1}^{n}(x_i-\bar{x})^3}{(n-1)(n-2)\sigma^3} \end{array} \tag{3}$$

Se define a la curtosis de igual manera con su función básica y completa como:

$$\begin{array}{cc} \text{Fórmula básica} & \text{Fórmula completa} \\ C_{kurt1} = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^4}{\sigma^4} & C_{Kurt2} = \frac{n(n+1)\sum_{i=1}^{n}(x_i-\bar{x})^4}{(n-1)(n-2)(n-3)\sigma^3} - \frac{3(n-1)^2}{(n-2)(n-3)} \end{array} \tag{4}$$

Entonces el coeficiente para la prueba D'Angostino's se obtiene mediante la fórmula:

$$Z_k^2 + Z_s^2 = \mathcal{K}^2 \tag{5}$$

Donde los coeficientes Z_k y Z_s también tienen su manera básica y completa (tomando en cuenta el tamaño de la población):

Forma Básica

$$\begin{aligned} Z_s &= \tfrac{C_{as1}}{s_e}, & s_e &= \sqrt{\tfrac{6n(n-1)}{(n-2)(n+1)(n+3)}} \\ Z_k &= \tfrac{C_{kurt1}}{s_{e2}}, & s_{e2} &= 2(n-1)\sqrt{\tfrac{6n}{(n-2)(n-3)(n+3)(n+5)}} \end{aligned} \tag{6}$$

Forma Completa (contemplando una población $n > 20$ y distribuidos en el rango [0,1], $\in \mathcal{N}(0,1)$):

$$\begin{aligned} Z_s &= b \cdot ln(u + \sqrt{u^2+1}) & Z_k &= \tfrac{1-r-v^{\frac{1}{3}}}{\sqrt{r}} \\ c &= \tfrac{3(n^2+27n-70)(n+1)(n+3)}{(n-2)(n+5)(n+7)(n+9)} & d &= \sqrt{\tfrac{(n+1)^2(n+3)(n+5)}{24n(n-2)(n-3)}} \\ w^2 &= -1 + \sqrt{2(c-1)} & e &= \tfrac{6(n^2-5n+2)}{(n+7)(n+9)}\sqrt{\tfrac{6(n+3)(n+5)}{n(n-2)(n-3)}} \\ a &= \sqrt{\tfrac{w^2+1}{2}}, b = \tfrac{1}{ln(\sqrt{w})} & f &= 6 + \tfrac{8}{e_1}(\tfrac{2}{e_1} + \sqrt{1+\tfrac{4}{e_1^2}}) \\ u &= a \cdot C_{as2}\sqrt{\tfrac{(n+1)(n+3)}{6(n-2)}} & g &= d(C_{kurt2} - \tfrac{3(n-1)}{n+1})\sqrt{\tfrac{2}{f-4}} \\ & & v &= \tfrac{1-\frac{2}{f}}{1+g}, r = \tfrac{2}{9f} \end{aligned} \tag{7}$$

De igual manera, se define un umbral y se compara si es perteneciente a un conjunto de datos normal. También se define como una función de scipy.

```
from scipy.stats import normaltest
```

3.1.3. Prueba de Anderson-Darling

La prueba de Anderson-Darling se usa para probar si una muestra proviene de una distribución en especifico [5]. Es una modificación de la prueba de Kolmogorov-Smirnov (K-S) dándole más importancia a las colas de la distribución. Tiene la ventaja de ser mas sensitivo para cierto tipo de pruebas, sin embargo, tiene la desventaja de que los valores críticos tienen que ser calculado en todo momento, por lo que al tener nuevos datos, se deben recalcular los parámetros [7]. Se expresa mediante la siguiente ecuación:

$$A^2 = -N - S \tag{8}$$

donde N es la longitud de los datos y S es para una muestra de datos:

$$S = \sum_{i=1}^{N} \frac{2i-1}{N}[ln(F(Y_i)) + ln(1 - F(Y_{N+1-i}))] \tag{9}$$

Donde F es la función de distribución acumulativa de la distribución dada y Y son los datos ordenados.

También existe una opción basada en distancias de the Cramér–von Mises.

$$A = n \int_{-\infty}^{\infty} \frac{(F_n(x) - F(x))^2}{F(x)(1 - F(x))} dF(x) \tag{10}$$

Al momento de aplicarlo a la función de scipy, se tienen que tomar en cuenta algunos aspectos; como es el valor critico para cada distribución (dados por la función de distribución a la que se quiere aproximar véase Ecuación (11)), lo que implica que si supera esos valores se puede rechazar la hipótesis nula. Además de tener el parámetro del coeficiente de Anderson.

Normal/Exponencial ('norm','expon')	15 %	10 %	5 %2.5 %	1 %		
Logística ('logistic')	25 %	10 %	5 %2.5 %	1 %	0.5 %	(11)
Gumbel ('gumbel')	25 %	10 %	5 %2.5 %	1 %		

Se define en scipy con la función:

```
from scipy.stats import anderson
```

3.2. Pruebas de correlación

Estas pruebas consisten en obtener la referencia de la naturaleza de relación entre distintas variables, mediante una magnitud de relación entre dichas variables, Todo esto en variables cuantitativas. Planteando la siguiente hipótesis.

$$H_0 : \text{Las dos muestras son independientes}$$
$$H_1 : \text{Hay dependencia entre las muestras}$$

Formalmente:

$$\begin{aligned} H_0 &: X \not\perp Y \\ H_1 &: X \perp Y \end{aligned} \tag{12}$$

La cual explica sobre la dependencia e independencia de las variables.

3.2.1. Prueba de Pearson

Se utiliza en variables cuantitativas. Calcula un índice de grado de covariación entre distintas variables relacionadas linealmente [8]. Esto significa que puede haber variables fuertemente correlacionadas pero no de forma lineal, en cuyo caso no es posible aplicar la prueba de correlación de Pearson [9]. Esta se calcula mediante la siguiente Ecuación:

$$\rho = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{(n-1)\sigma_x \sigma_y} \tag{13}$$

Donde:

- n es el numero de observaciones
- $\bar{x}, \bar{y}$ es la media estandar de la muestrá

- $\sigma_x, \sigma_y = \sqrt{\frac{\sum((x_i,y_i)-(\bar{x},\bar{y}))^2}{n-1}}$ es la desviación estándar de la muestra

Teniendo intervalos de confianza que muestra de manera bilateral la certeza del resultado con respecto al parámetro de observaciones dado.

Para el caso de Pearson, se tiene un intervalo de confianza bilateral de $(1-\alpha)100\,\%$ para ρ es $(\rho_L, \rho_\cup)$, donde ρ_L es el límite inferior y el límite superior $\rho_\cup$, partiendo de la transformación de Fisher:

$$\begin{aligned} z &= 0.5ln(\tfrac{1+\rho}{1-\rho}) \\ S_{ez} &= \tfrac{1}{\sqrt{n-3}} \end{aligned} \tag{14}$$

Donde z la transformación de Fisher y S_{ez} es el error estandar de la transformación de Fisher.

A continuación se muestran las formulas para obtener el intervalo de confianza del método.

$$\begin{aligned} \rho_L &= \tanh(z - (z \times S_e z)) \\ \rho_\cup &= \tanh(z + (z \times S_e z)) \end{aligned} \tag{15}$$

En python con la función de scipy:

```
from scipy.stats import pearsonr
```

La cual retorna el coeficiente y el valor p, **P-value**: que indica la probabilidad de que un valor estadistico calculado sea posible dad una hipotesis nula cierta. Generalmente, se utiliza un umbral de significancia predefinido (como 0.05 o 0.01) para tomar decisiones. Si el valor p es menor que el umbral de significancia, se considera que los resultados son estadísticamente significativos y se rechaza la hipótesis nula. Si el valor p es mayor que el umbral de significancia, no se tienen suficientes pruebas para rechazar la hipótesis nula y se considera que no hay evidencia suficiente para respaldar la hipótesis alternativa.

3.2.2. Prueba de correlación de Spearman

Es una medida no paramétrica de la correlación del rango (dependencia estadistica del ranking entre dos variables). Esto quiere decir que mide la correlación con respecto a la posición de los datos ya sea en un orden ascendente o descendente (generalmente descendente) [10, 11].

Midiendo la fuerza y dirección de la asociación de dos variables.

Se debe conocer que las funciones pueden ser de 3 tipos (véase Figura 3).

1. *Monótonamente en aumento*: es aquella que nunca disminuye la variable dependiente mientras la independiente incrementa.

2. *Monótonamente en decremento*: Es cuando la variable independiente aumenta pero la variable dependiente nunca aumenta

3. *No monótona*: Cuando la variable intependiente aumenta y la variable dependiente a veces aumenta y a veces disminuye

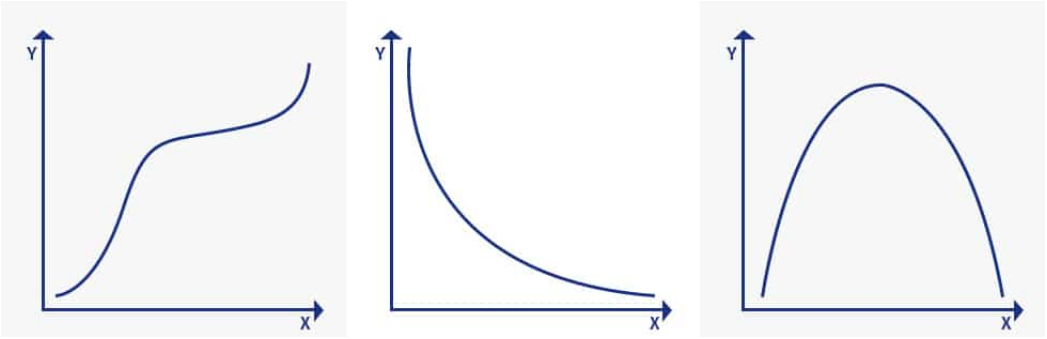


Figura 2: Funciones monotónicas y no monótona.

Se calcula mediante la ecuación, que tiene un rango de $[-1, 1]$:

$$r_R = 1 - \frac{6 \sum_i d_i^2}{n(n^2 - 1)} \tag{16}$$

donde n es la cantidad de observaciones, d_i es la diferencia en el rango del elemento. Por lo que un valor cercano a 1 significa que hay una buena asociación positiva de rango, 0 significa que no hay asociación entre rangos y -1 significa que hay una asociación negativa entre los rangos.

En python se determina con la función siguiente, al igual que la función de Pearson, regresa el valor de significancia y el valor p.

```
from scipy.stats import spearmanr
```

3.2.3. Prueba de correlación de rangos de Kendall

Es una prueba de datos no paramétrica (no analiza datos con alguna distribución en particular pero se basan en hipótesis y no están organizados de manera normal), con un valor entre 0 y 1 donde 0 es que no hay relación y 1 que es una relación perfecta. Algo que hay que tomar en cuenta es que se trabajan con datos **clasificados** por lo que se pueden considerar como valores categóricos o continuos en los que si importa el orden. Pueden haber valores negativos, pero esto se debe a los signos, por lo que se debe tomar el valor absoluto. Se define con la variable τ, sin embargo, existen varias pruebas (variaciones) del método [12, 13]. El más común es τ_b, el cual se obtiene de la siguiente manera.

$$\tau_b = \frac{C - D}{C + D} \tag{17}$$

donde C es el numero de pares concordantes y D es el número de pares discordantes, el primero se calcula con la concordancia encontrando los valores son mayores y ese numero será un valor de concordancia, en caso de encontrar un valor no concordante (diferencia de columnas no es 0) se asigna el mismo valor antes de el. Para la discordancia, se debe encontrar los valor concordantes y encontrar los valores más pequeños a este.

Además de que cuenta con una puntuación z para una distribución normal (solo con $n \geq 10$).

$$z = 3\tau \frac{\sqrt{n(n-1)}}{\sqrt{2(2n+5)}} \tag{18}$$

donde τ es el valor de Kendall, n es el numero de pares. *Se recomienda cambiar el valor z a valor p mediante tablas.*

En caso de que no haya ningún empate de concordancia se utilizará la siguiente formula [14]:

$$\tau = \frac{S}{\frac{1}{2}N(N-1)} \tag{19}$$

Donde S es la puntuación efectiva de los rangos (Y), lo que se debe de hacer es ordenar la variable independiente de menor a mayor y pasar número por número de Y obteniendo cuantos números son mayores y cuantos menores (los mayores restan 1 y los mayores suman 1) y finalmente sumar todas las diferencias de números.

En python se realiza mediante la función siguiente:

```
from scipy.stats import kendalltau
```

En donde al introducir los datos, se obtendrá el valor de significancia (τ) y el valor p.

3.2.4. Prueba Chi-Cuadrado

Es una prueba no paramétrica que es utilizada para examinar diferencias entre variables categóricas de la misma población. El principio se basa en comparar los valores observados y los esperados si la hipótesis nula fuera cierta. Entonces se puede determinar si hay una diferencia en los valores se debe al azar o si se debe a una relación entre las variables que se están estudiando [15].

Dentro de estas pruebas se encuentra la prueba chi-cuadrada de contingencia o independencia. Esta sirve para comprobar la independencia de frecuencias entre dos variables aleatorias, X, Y. Partiendo de la condicion de independencia que se muestra a continuación [16]:

$$X \text{ e } Y \text{son independientes} \Leftrightarrow \forall\, x, y\ f(x,y) = f(x) \cdot f(y) \tag{20}$$

Esto implica que se necesita estimar las funciones de probabilidad de ambas variables separadas y de la función de probabilidad conjunta [17].

Entonces se debe definir una tabla de dimensión $n \times m$, donde definimos las sumas de renglones, columnas y total como:

x \ y	y_1	y_2	$\cdots$	y_j	$\cdots$	y_m	$F_i = \sum_i^n O_{ij}$
x_1	O_{11}	O_{12}	$\cdots$	O_{1j}	$\cdots$	O_{1m}	F_1
x_2	O_{21}	O_{22}	$\cdots$	O_{2j}	$\cdots$	O_{2m}	F_2
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
x_i	O_{i1}	O_{i2}	$\cdots$	O_{ij}	$\cdots$	O_{im}	F_i
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
x_n	O_{n1}	O_{n2}	$\cdots$	O_{nj}	$\cdots$	O_{nm}	F_n
$C_i = \sum_j^m O_{ij}$	C_1	C_2	$\cdots$	C_j	$\cdots$	C_m	T

Figura 3: Chi-cuadrada tabla de contingencia.

Entonces se obtiene que:

$$T = \sum_{ij} O_{ij} = \sum_i F_i = \sum_j C_j \tag{21}$$

Entonces se plantea la hipótesis de independencia implica que:

$$\forall\ i, j\ \frac{O_{ij}}{T} = \frac{F_i \cdot C_j}{T^2} \tag{22}$$

Convirtiendop en frecuencias absolutas multiplicado por T:

- Si X e Y son independientes, O_{ij} debe ser igual a $\frac{F_i C_j}{T}$ y por tanto:
- bajor la hipótesis de independencia, $\frac{F_i C_j}{T}$ es el valor esperado de O_{ij} (E_{ij})

Basandose en la prueba anterior, si las variables son independientes,es decir, E_{ij} son realmente los valores esperados de las frecuencias O_{ij}, se puede calcular un parámetro chi-cuadrado.

$$\sum_{ij} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \approx \chi^2 \tag{23}$$

En python se encuentra la función de chi-cuadrada.

```
from scipy.stats import chi2_contingency
```

3.3. Pruebas de estacionareidad

Estas pruebas consisten en el análisis de series de tiempo para determinar si una serie de datos establece un comportamiento estacionario o no. Lo que implica que sus propiedades estadísticas (expectativa, varianza, auto correlación) no varían con el tiempo. Es importante conocer el concepto de raíz unitaria, la cual es una tendencia estocástica en una serie de tiempo. Esto significa que si una serie de tiempo tiene raíz unitaria, muestra un patrón sistemático impredecible.

El ruido blanco es un ejemplo de una serie de tiempo estacionaria. Permitiendo si una una serie de tiempo tiene tendencia, estacionalidad u otras características que pueden afectar su comportamiento y análisis [18]. La hipótesis nula que se establece es:

$$H_0 : \text{Hay una raíz unitaria presente (la serie no es estacionaria)}$$
$$H_1 : \text{No hay una raíz unitaria presente (la serie es estacionaria)}$$

La mayoria de estos modelos se basan en la autorregresión, la cual básicamente es una regresión de Pearson entre los valores actuales y los retrasados (k valores antes) de la serie de tiempo [19].

$$ACF(k) = \frac{\sum_{i=1+k}^{n}(y_i - \bar{y})(y_{t-k} - \bar{y})}{\sigma^2} \tag{24}$$

3.3.1. Prueba de raíz unitaria aumentada Dickey-Fuller

Existen bastantes métodos para la prueba ADF, sim embargo, se expondrá la más común que es la de modelos de regresión y aproximación del valor p con MacKinnon [20].

Primeramente se debe utilizar la estimación de [21, 22]

$$\begin{gathered}
\text{Modelo con un solo coeficiente constante} \\
\nabla y_t = \beta_0 + YY_{t-1} + \delta_1 \nabla Y_{t-1} + \delta_2 \nabla Y_{t-2} + \cdots + \delta_p \nabla Y_{t-p} + \epsilon_t \\
\text{Modelo con un coeficiente constante y uno lineal} \\
\nabla y_t = \beta_0 + \beta_1 t + YY_{t-1} + \delta_1 \nabla Y_{t-1} + \delta_2 \nabla Y_{t-2} + \cdots + \delta_p \nabla Y_{t-p} + \epsilon_t \\
\text{Un modelo con un coeficiente constante, un coeficiente lineal y un coeficiente cuadrático} \\
\nabla y_t = \beta_0 + \beta_1^t + \beta_2^{t^2} + YY_{t-1} + \delta_2 \nabla Y_{t-2} + \cdots + \delta_p \nabla Y_{t-p} + \epsilon_t \\
\text{Un modelo sin coeficientes de regresión} \\
\nabla Y_t = YY_{t-1} + \delta_1 \nabla Y_{t-1} + \delta_2 \nabla Y_{t-2} + \cdots + \delta_p \nabla Y_{t-p} + \epsilon_t
\end{gathered} \tag{25}$$

Donde:

- $Y_1, Y_2, \cdots, Y_T$ son los valores de las series temporales observadas en el tiempo $1, 2, 3, \cdots, T$.
- ∇Y_t la diferencia de dos observaciones consecutivas en el tiempo t,$Y_t - Y_{t-1}$, donde $t = 2, \cdots, T$.
- β_0 el término constante de la regresión.
- β_1 el coeficiente de una tendencia de tiempo lineal en un modelo de regresión.
- β_2 el coeficiente de una tendencia de tiempo cuadrática en un modelo de regresión.
- p el orden de retraso del proceso autorregresivo.
- ϵ_t el término de error independiente en serie en el momento t para $t = 2, ..., T$

Para probar la hipótesis se plantea:

$$H_0 : Y = 0, \quad H_1 : Y < 0$$

Teniendo como estadistica de prueba.

$$\frac{\hat{Y}}{EE(\hat{Y})} \tag{26}$$

Donde $\hat{Y}$ es la estimación del coeficiente mínimo cuadrado de la Y coeficiente y $EE(\hat{Y})$ es el error estándar de la estimación de mínimos cuadrados de la Y coeficiente del modelo de regresión.

Por lo que para obtener los valores aproximados de *p-value* de MacKinnon para los niveles de significancia 0.01, 0.05 y 0.1 se aplica la fórmula [23]:

$$\beta_\infty + \frac{\beta_1}{n} + \frac{\beta_2}{n^2} + \frac{\beta_3}{n^3} \tag{27}$$

donde n es el número de observaciones que el analisis utiliza para ajustarse al modelo de regresión. Los valores $\beta_{\infty,1,2,3}$ provienen de [23].

En python se encuentra la función de ADF.

```
from statsmodels.tsa.stattools import adfuller
```

3.3.2. Prueba Kwiatkowski-Phillips-Schmidt-Shin

Esta prueba provee de una prueba de estacionareidad mediante la raíz unitaria de una manera directa. Para esto, hay que considerar una serie de tiempo de tres componentes representada de la forma $Y_1, Y_2, \cdots, Y_n$ como la suma de una tendencia determinista, un camino aleatorio y un residual estacionario utilizando la formula [24]:

$$Y_t = \beta t + (r_t + \alpha) + e_t \tag{28}$$

Donde:

- $r_t = r_{t-1} + u_t$ es un camino aleatorio, con un valor inicial $r_0 = \alpha$ como un atajo.
- t es el tiempo.
- u_t son distribuciones independientes idénticas $(0, \sigma_\mu^2)$.

El modelo simplificado también es de utilidad y no contempla u_t para determinar el nivel de estacionareidad. Por lo que de una manera sencilla se puede se formula la hipótesis:

$$H_0 : Y_t \text{ tiende a un nivel estacionario } || \ \sigma_\mu^2 = 0$$
$$H_1 : Y_t \text{ es un proceso de raíz unitaria}$$

Además de que con la prueba de *p-value* se puede realizar la hipótesis nula.

En python se encuentra la función de KPSS.

```
from statsmodels.tsa.stattools import kpss
```

3.4. Pruebas de hipótesis estadísticas paramétricas

as pruebas paramétricas son una herramienta estadística que se utiliza para el análisis de los factores de la población. Este método requiere que se especifique la forma de distribución de la población materna estudiada. Las pruebas paramétricas están basadas en la ley de distribución de la variable que se estudia [25].

Una prueba paramétrica debe cumplir con los siguientes elementos:

1. **Normalidad**: El análisis y observaciones que se obtienen de las muestras deben considerarse normales. Para esto se deben realizar pruebas de bondad de ajuste donde se describe que tan adaptadas se encuentran las observaciones y cómo discrepan de los valores esperados.

2. **Homocedasticidad**: Los grupos deben presentar variables uniformes, es decir, que sean homogéneas.

3. **Errores**: Los errores que se presenten deben de ser independientes. Esto solo sucede cuando los sujetos son asignados de forma aleatoria y se distribuyen de forma normal dentro del grupo.

La hipótesis que se plantea es:

$$H_0 : \text{las medias de las muestras son iguales}$$
$$H_1 : \text{las medias de las muestras son diferentes}$$

Cabe destacar que la mayoría de la información fue obtenida en [26], por lo que formalmente se puede establecer como:

$$\begin{aligned} H_0 &: \mu_1 = \mu_2 \\ H_1 &: \mu_1 \neq \mu_2 \end{aligned} \tag{29}$$

3.4.1. Prueba de t de Student

Esta prueba estadística determina si hay una diferencia significativa entre las medias de dos grupos de datos [27, 28]. Se debe hacer las siguientes suposiciones: las observaciones deben ser independientes, las poblaciones con distribución normal, Las mediciones se deben elaborar en una escala de intervalo que tengan la misma magnitud (normalizada) y las varianzas de los grupos deben ser homogéneas.

Para realizar esta prueba se debe calcular la desviación estándar ponderada.

$$\sigma_p = \frac{\sqrt{\sum x_1^2 + \sum x_2^2}}{N_1 + N_2 - 2} \tag{30}$$

Donde en el numerador se realiza la suma de cuadrados de las muestras y $N_{1,2}$ es el tamaño de la muestra 1 y 2 respectivamente.

Una vez que se obtuvo la desviación estándar ponderada, se calcula el valor t de Student. el cual puede ser utilizado para 3 situaciones

$$\nabla_1 x_{1,2} = \bar{x} - \mu \quad \nabla_2 x_{1,2} = \bar{x}_1 - \bar{x}_2 \quad \nabla_3 x_{1,2} = \bar{x}_{1j} - \bar{x}_{2j} \tag{31}$$

Donde $\nabla_1 x_{1,2}$ representa una sola muestra que se compara con la población del mismo grupo, $\nabla_2 x_{1,2}$ representa la comparación de dos muestras independientes del grupo y $\nabla_3 x_{1,2}$ representa dos muestras relacionadas, es decir, siendo la misma muestra pero medida en dos momentos diferentes.

$$t = \frac{\nabla_b}{\frac{\sigma_p}{\sqrt{N}}} \tag{32}$$

Indicando el valor t la cantidad de unidades estándares que están separando las medias de los dos grupos.

En python se encuentra la función de t de Student.

```
from scipy.stats import ttest_ind
```

3.4.2. Prueba t de student emparejada

Esta prueba al igual que la anterior, compara 2 grupos, en este caso son muestras independientes con varianzas no homogéneas, también es conocida como prueba t de Student-Welch, agregando los grados de libertad [26].

Para calcular los grados de libertad se utiliza la formula:

$$gdl = \frac{(\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2})^2}{\frac{(\frac{\sigma_1^2}{n_1})^2}{n_1 - 1} + \frac{(\frac{\sigma_2^2}{n_2})^2}{n_2 - 1}} \times 2 \tag{33}$$

Entonces la formula de la prueba t de Student-Welch es:

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} \tag{34}$$

habiendo una variante en muestras independientes para la varianza, obtenida:

$$\sigma^2 = \frac{\sum_{i=1}^{n_1}(x_1 - \bar{x}_1)^2 + \sum_{j=1}^{n_2}(x_j - \bar{x}_2)^2}{n_1 + n_2 - 2} = \sigma_{1,2}^2 \tag{35}$$

Entonces el proceso general de t de Student-Welch es:

1. Determinar la media, la varianza y el tamaño de la muestra de cada población de estudio.
2. Aplicar ecuación t.
3. Calcular los grados de libertad gdl de acuerdo con la ecuación dada.

4. Comparar el valor t calculado respecto a los grados de libertad con valores t criticos.
5. Decidir si se acepta o rechaza la hipótesis.

En python se encuentra la función de t de Student emparejada.

```
from scipy.stats import ttest_rel
```

3.4.3. Prueba de análisis de varianza (ANOVA)

EL analisis de varianza es un método que se utiliza para comparar varianza entre las medias de diferentes grupos. Una variedad de contextos lo utilizan para determinar si existe alguna diferencia entre las medias de los diferentes grupos. Esto implica que el método se basa de la variación total entre los datos y la descomposición de esta en diversos factores.

Respondiendo a la hipótesis de la existencia de diferencias significativas entre las medias de las poblaciones. Evaluando la importancia de uno o más factores al comparar las medias de una variable de respuesta en los diferentes niveles de los factores [29, 30].

El analisis ANOVA utiliza el mismo marco conceptual que la regresión lineal. En este caso se trabaja con el ANOVA unidireccional, lo que siginifica que el experimento tiene una sola variable independiente (factor) con dos o más niveles [31]. Partiendo del número de factores el modelo ANOVA se puede generalizar como:

$$y_i = \beta_0 + \sum_{i=1\cdots p} \beta_{k_{(i,j)}j} + \epsilon_i \tag{36}$$

donde y_i es el valor observado de la variable dependiente para la observación $i, k(i,j)$ es el índice de la categoría (o nivel) del factor j para la observación i y $_i$ es el error del modelo.

Sin embargo se suele utilzar la tabla ANOVA para estos casos, la cual se puede reducir al siguiente conjunto de ecuaciones sencillas, partiendo de los factores y el error de la Ecuación (36) [32, 33].

Fuente de variación	suma de cuadrados	grados de libertad	Cuadrado medio	valor F
Factor	$SS_F = \sum_{i=1}^{k} n_i(\bar{y}_i - \bar{y})^2$	$GL_F = k-1$	$MSE_F = \frac{SS_F}{k-1}$	$F = \frac{MSE_F}{MSE_E}$
Error	$SS_E = \sum_{i=1}^{k}\sum_{j=1}^{n_i}(y_{ij} - \bar{y}_i)^2$	$GL_E = N-k$	$MSE_E = \frac{SS_E}{n-K}$	
Total	$SS_T = \sum_{i=1}^{k}\sum_{j=1}^{n}{}_i(y_{ij} - \bar{y})^2$			

(37)

Donde k es la cantidad de atributos, N es la cantidad de datos por atributo, n_i es el número de datos en el atributo i.

Finalmente se define la probabilidad $p-value$ de que el valor F de Snedecor y se obtiene $P[F > F_r]$, con el que se resolver la hipótesis nula [34].

En ciencia de datos, permite conocer las caracteristicas más fiables y utiles para poder formar modelos de Machine Learning. Minimizando el número de variables de entrada para reducir la complejidad del modelo. Ayudando a determinar si una variable independiente está influyendo en una variable objetivo

En python se encuentra la función de ANOVA.

```
from scipy.stats import f_oneway
```

3.5. Pruebas de hipótesis estadísticas no paramétricas

Estas pruebas a diferencia de las paramétricas, no requieren supuestos específicos sobre la distribución de los datos, permitiendo analizar datos en escala nominal (tienen jerarquía pero la diferencia o distancia entre los valores no es necesariamente cuantificable) y categórica (representan categorías o grupos sin un orden especifico) [35]. Por lo que la hipótesis a plantear es.

$$H_0 : \text{proceden de poblaciones continuas idénticas}$$
$$H_1 : \text{no proceden de poblaciones continuas idénticas}$$

3.5.1. Prueba U de Mann-Whitney

Utiliza el rango de cada caso para probar si los grupos se han extraído de la misma población. La prueba de Mann-Whitney contrasta si dos poblaciones muestreadas son equivalentes en su posición. Se debe suponer que las variables son independientes y permite realizar un análisis cuantitativo mediante la diferencia entre 2 medianas, por lo que se basa en rangos en lugar buscar parámetros de muestra [35].

Para calcular la prueba de U de Mann-Whitney se debe construir a partir de la suma de rangos de una de las muestras [36]. Siguiendo los siguientes pasos:

- *Ordenar*: Se ordenan todos los datos en orden creciente.
- Se suman los rangos de cada uno de los intervalos y se calcula la suma de los rangos de los datos de cada uno de los grupos (R_1, R_2)
- Se calculan los valores U_1, U_2 y se selecciona al más grande (U_{cal}).

$$\begin{aligned} U_1 &= n_1 n_2 + \tfrac{n_1(n_1+1)}{2} - R_1 \\ U_2 &= n_1 n_2 + \tfrac{n_2(n_2+1)}{2} - R_2 \end{aligned} \tag{38}$$

- Se comprueba la significancia estadística (valor-p) y se comprueba con el valor estadístico U_{crit}.

En python se encuentra la función de U de Mann-Whitney.

```
from scipy.stats import mannwhitneyu
```

3.5.2. Prueba de rangos con signo de Wilcoxon

La prueba de Wilcoxon comprueba si los valores medios de dos grupos dependientes difieren significativamente entre sí. La prueba de Wilcoxon es una prueba no paramétrica y, por tanto, está sujeta a muchos menos supuestos que su homóloga paramétrica, la prueba t para muestras dependientes. Por tanto, en cuanto dejan de cumplirse las condiciones límite de la prueba t para muestras dependientes, se utiliza la prueba de Wilcoxon [37]. Cabe destacar que esta prueba funciona tanto para datos categoricos como continuos.

Esta prueba toma en cuenta que para muestras pequeñas se sigue alguna distribución especifica, en caso de muestras grandes ($n \geq 20$), se aproxima a una distribución normal [38, 39].

Para el caso de muestras pequeñas se utiliza la suma de rangos de las diferencias positivas entre datos S_+.

$$S_+ = \sum_{i=1} R_i, \quad R_i = \text{máx}(\nabla^+) - \text{mín}(\nabla^-) \tag{39}$$

En caso de muestras grandes se puede obtener directamente el valor p mediante la fórmula.

$$Z = \frac{S_+ - \frac{n(n+1)}{4}}{\sqrt{\frac{n(n+1)(2n+1)}{24}}} \tag{40}$$

En python se encuentra la función de prueba de rangos con signo de Wilcoxon.

```
from scipy.stats import wilcoxon
```

3.5.3. Prueba H de Kruskal-Wallis

La prueba de Kruskal-Wallis es una extensión de la prueba U de Mann-Whitney. La prueba es el análogo no paramétrico de análisis de varianza de un factor y detecta diferencias en la ubicación de distribución. La prueba supone que no hay ningún orden a priori de las poblaciones k de las cuales se extraen las muestras [35, 26]. **Esta prueba se utiliza cuando hay más de dos muestras independientes**.

$$H = \frac{\frac{12}{n(n+1)} \sum \frac{\sum R_c^2}{n_i} - 3(n+1)}{L} \tag{41}$$

Donde L se calcula:

$$L = 1 - \frac{\sum (L_i^3 - L_i)}{n^3 - n} \tag{42}$$

Donde:

- H es el valor estadístico de la prueba de Krustal-Wallis.
- n es el tamaño de la muestra.
- R_c^2 es la sumatoria de los rangos elevados al cuadrado.

- n_i es el tamaño de la muestra en cada grupo.
- L es el ajuste dado por el ajuste de ligas o empates de rangos.
- L_i valor de numero de empates en un rango.

En python se encuentra la función de prueba H de Krustal-Wallis.

```
from scipy.stats import kruskal
```

3.5.4. Prueba de Friedman

La prueba de Friedman Test es una ampliación de Prueba de Wilcoxon de los rangos con signo y el análogo no paramétrico de medidas repetidas de un factor. Friedman contrasta la hipótesis nula de que las k variables relacionadas procedan de la misma población. Para cada caso, a las k variables se les asignan los rangos 1 a k. El estadístico de contraste se basa en estos rangos [35]. Siendo un complemento del procedimiento de análisis de varianza de una entrada de Kruskal-Wallis [26]. Se supone que las observaciones no tienen una distribución normal, pero tienden a ubicarse en una escala de intervalo. Se calcula mediante la siguiente fórmula.

$$X_r^2 = \frac{12}{HK(K+1)} \sum R_c^2 - 3H(k+1) \tag{43}$$

Donde:

- X_r^2 es el calculo estadistico de Friedman
- H numero de instancias.
- k número de atributos.
- $\sum R_c^2$ es la suma de rangos por columnas al cuadrado.

En python se encuentra la función de prueba Friedman.

```
from scipy.stats import friedmanchisquare
```

4. Materiales y Métodos

4.1. Materiales

4.1.1. Base de datos

Las bases de datos fueron obtenidas de distintos medios, por lo que se trabajarán con datos acordes a cada prueba, Se buscarán las bases de datos pertinentes y en la sección de resultados se especificarán de donde se obtuvieron los datos. Por lo que los datos se pueden expresar como un

conjunto de otros datos obtenidos de diferentes bases de datos (*subsets*). Entonces para cada prueba se utilizará un subconjunto de la base de datos.

$$\begin{gathered} Data \in \{St_1 \cup ST_2 \cup ST_3 \cup \cdots \cup St_n\} \\ p, z = prueba_{H_0}(Data[a, b]) \end{gathered} \tag{44}$$

4.1.2. Librerías y entorno de desarrollo

El análisis de los datos se llevará a cabo en el lenguaje de programación Python dentro del entorno de desarrollo de Jupyter Notebook. Ocupando las librerías matplotlib, seaborn, numpy, Pandas y scipy.

4.2. Metodología

La metodología consiste en la recopilación de las bases de datos, aplicar métodos de preprocesamiento de datos, realizar cada prueba para el conjunto de datos designado, obtener los datos para rechazar o aceptar las hipótesis nulas (un valor p y el coeficiente de la prueba) y mostrar los resultados para las hipótesis nulas.

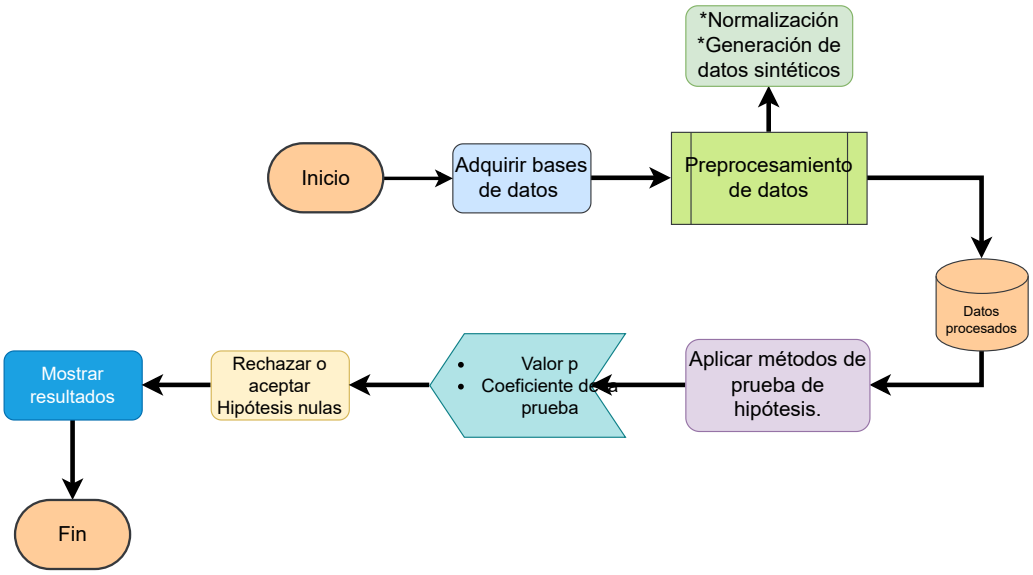


Figura 4: Metodología de la práctica.

5. Pseudocódigo

Algoritmo 1 Pseudocódigo Pruebas de Hipótesis.

```
Inicio
 Class prueba_normalidad():
        def Shapiro  Wilk()
        def DAgostino()
        def Anderson-Darling()

  Class prueba  corr():
        def Pearson()
        def Spearman()
        def Kendall()
        def Chi  cuadrado()

  Class prueba_ estacionariedad():
        def DFA()
        def KPSS()

  Class prueba_par():
        def t_student()
        def t  student  a()
        def ANOVA()

  Class prueba_no_par():
        def U_mann()
        def Wilcoxon()
        def H_krustall()
        def Friedman()

Data ← Data ∈ {A_0, A_1, A_2, ⋯, A_n}

Para prueba en pruebas:
    Para metodo en prueba.longitud():
          c_p, p_valor ← Prueba.metodo(Data[A_metodo])
          Si  p_valor < p_std:
              Regresar Hipotesis← 0
          Sino:
              Regresar Hipotesis← 1
Fin
```

6. Resultados

6.1. Consideraciones

Para las pruebas de normalidad se utilizará un atributo de una base de datos que involucra a personas con ciertas características en común, el riesgo de tener diabetes. Se obtuvo de kaggle en donde únicamente se trabajará con el atributo continuo de peso del paciente. Esta consta de 390 instancias (véase Figura 6). Cabe destacar que no se tuvo que realizar ningún preprocesamiento del dato.

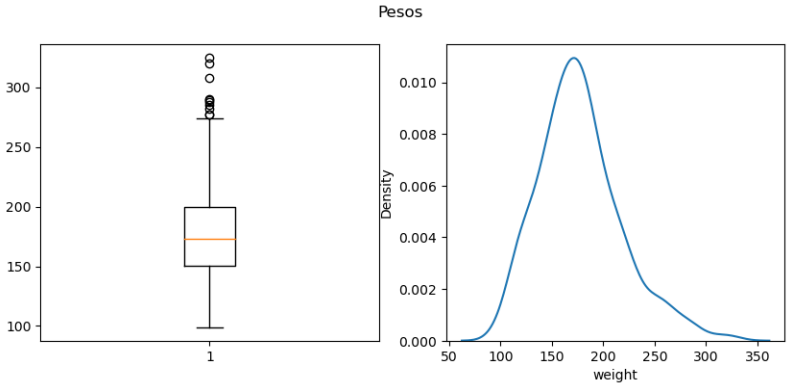


Figura 5: Distribución de pesos.

Es posible apreciar que la distribución tiende a ser Gaussiana (normal) y por lo tanto al tomar muestras aleatorias es probable que el resultado que arroje sea la hipótesis nula rechazada. Se tomaron muestras aleatorias.

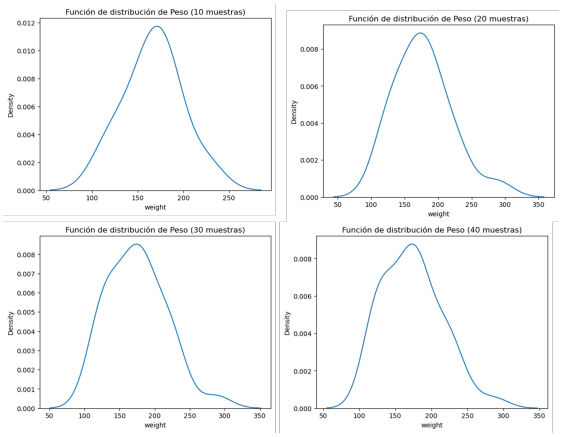


Figura 6: Distribución de las muestras tomadas.

Para las pruebas para correlacionar **datos continuos** se ocupará una base de datos obtenida de kaggle tomando en cuenta los atributos de precio y de pies cuadrados de la propiedad, conteniendo 102 instancias. Observando sus propiedades a continuación:

$$\mu_{precio} = \$490029 \quad \sigma_{precio} = \$95553$$
$$\mu_{sqft} = 7753.4ft^2 \quad \sigma_{sqft} = 1591.2ft^2$$

Teniendo una distribución de cada atributo y su diagrama de dispersión en la Figura 7:

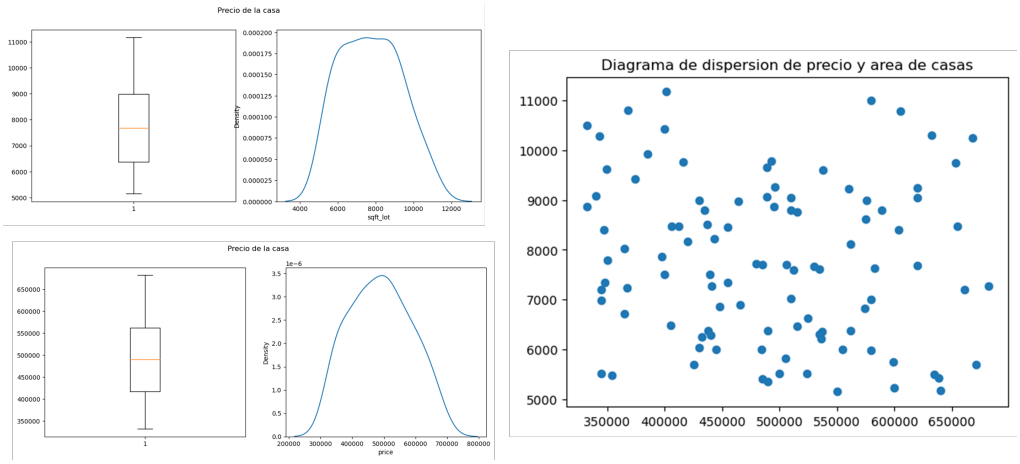


Figura 7: Distribución y dispersión de datos del precio y área de la casa.

Para pruebas de correlación con datos categoricos, ordinales, etc. se utilizará la base de datos de preferencias de comida entre dos poblaciones (Norteamericanos y japoneses) en donde cada uno prueba los mismos platillos y los califican, la base de datos cuenta con 2 atributos (nacionalidad) y calificación por platillo. A continuación se observan su histograma.

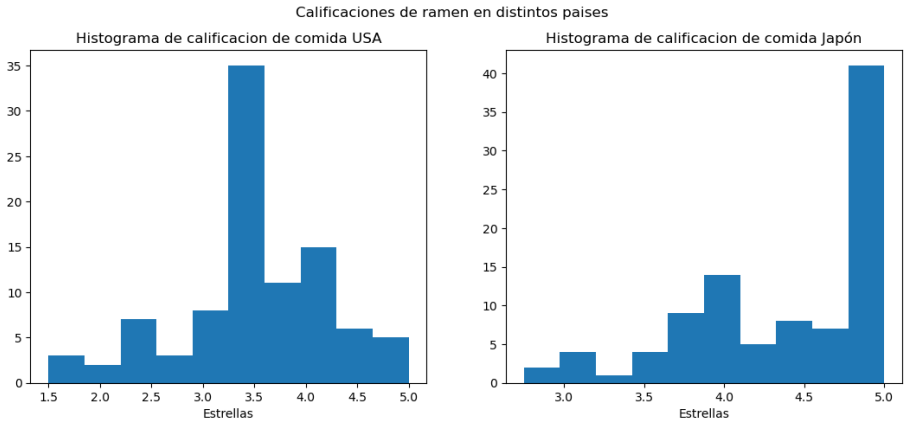


Figura 8: Histograma de calificaciones de ramen en distintos países.

Para series de tiempo se utilizará una base de datos meteorológica de la ciudad de manaus en brasil. Consta de las temperaturas registradas por mes y por año, en este caso se enfocará en el mes de Marzo. Se observa la serie de tiempo a continuación:

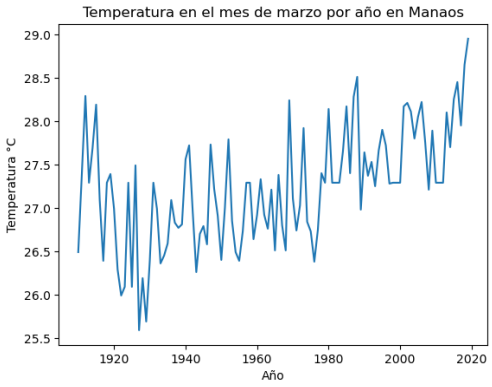


Figura 9: Temperatura a lo largo de los años en el mes de Marzo en Manaos, Brasil.

Para las pruebas de hipótesis paramétricas y algunas no paramétricas, se utilizará una base de datos que consiste en que a una población de hombres, otra de mujeres y un ultimo grupo de otros, se introdujeron en un mundo virtual y según sus experiencias y emociones, soportaron estar un cierto tiempo en dicho ambiente. Se registró el tiempo en minutos y se utilzará para comparar una relacion entre el tiempo que soporta un hombre, una mujer y otros, teniendo 3 atributos y 325 instancias. A continuación se muestran las distribuciones y diagramas (Figura 10).

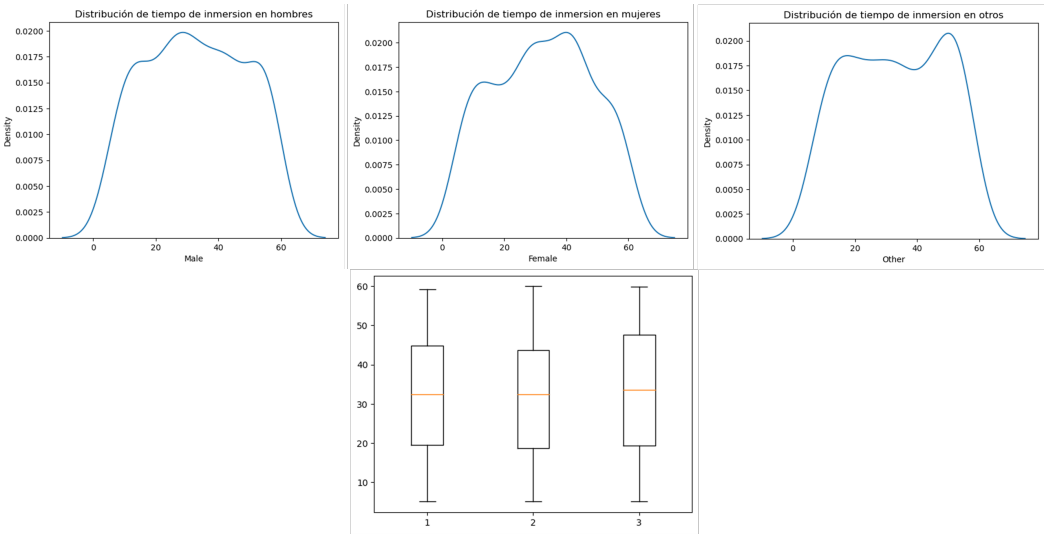


Figura 10: Distribución del tiempo de inmersión en hombres, mujeres y otros en un ambiente virtual.

Finalmente se utilizará una base de datos para las pruebas no paramétricas, donde se analizará la efectividad de ciertos tratamientos ante la misma enfermedad. Esta base de datos consta de 17 instancias y 4 atributos, donde se resaltan los 3 tipos de tratamientos demarcados como 'DX4,DX5,DX6'. Teniendo a continuación su diagrama de frecuencias (véase Figura 11).

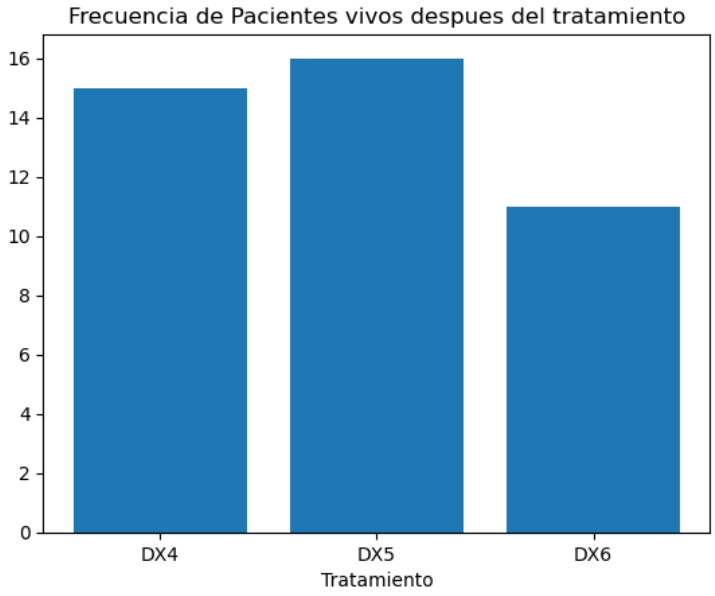


Figura 11: Frecuencias de pacientes vivos después del tratamiento ante la misma enfermedad.

NOTA: En todas las pruebas se realiza un intervalo de confianza del 95 % (confidence_level=.95)

6.2. Prueba de Shapiro-Wilk

Suposiciones: Muestra univariada, continuos, independientes y uniformemente distribuidos.

Con base en la base de datos trabajada y las muestras obtenidas, se obtuvieron los siguientes resultados:

Tabla 1: Resultados de la prueba Shapiro-Wilk con diferentes tamaños de muestra.

Muestras	Shapiro-Wilk	Valor p	Resultado de la hipótesis nula
10	0.970	0.891	Probablemente Normal
20	0.951	0.385	Probablemente Normal
30	0.954	0.214	Probablemente Normal
40	0.952	0.09	Probablemente Normal
390	0.966	0	Probablemente no Normal

Como se observa en la Tabla 1 y en la Figura 12 que hay una tendencia a la baja del valor p, por lo que es posible obsevar que hay una mayor probabilidad de que no pueda ser normal la distribución, sin embargo hay que tomar en cuenta que para muestras más grandes se tiene mayor exactitud en el resultado de Shapiro-Wilk pero el valor p tiende a ser menos exacto [3].

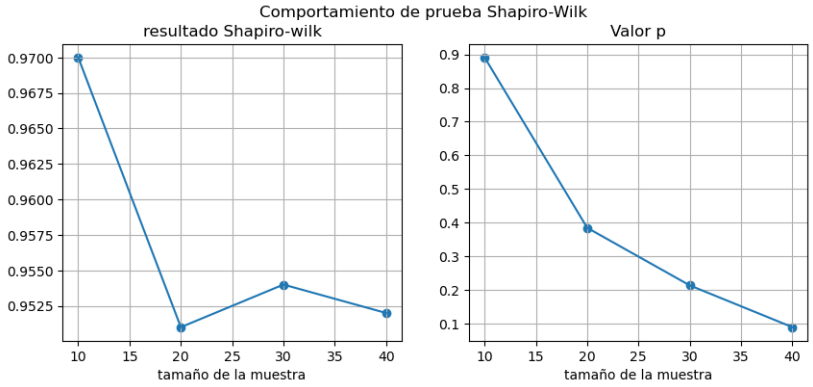


Figura 12: Comportamiento de la prueba con diferentes muestras.

6.3. Prueba D'Agostino´s

Suposiciones: Muestra univariada, continuos, independientes y uniformemente distribuidos.

Se obtuvieron los resultados siguientes:

Tabla 2: Resultados de la prueba D'Agostino´s con diferentes tamaños de muestra.

Muestras	D'Agostino's	Valor p	Resultado de la hipótesis nula
10	0.289	0.866	Probablemente Normal
20	3.723	0.155	Probablemente Normal
30	2.509	0.285	Probablemente Normal
40	2.631	0.268	Probablemente Normal
390	35.974	0	Probablemente no Normal

Se observa de igual manera que tiende a decrecer el valor p, tomando en cuenta que la muestra va creciendo, esto se puede deber a la aleatoriedad de las muestras y sus relaciones.

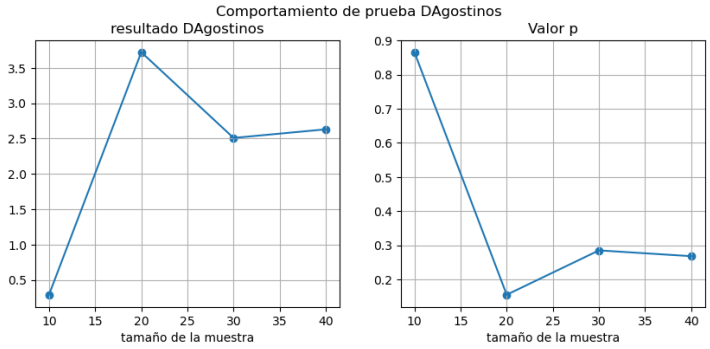


Figura 13: Comportamiento de la prueba D'Agostino´s con diferentes muestras.

6.4. Prueba de Anderson Darling

Suposiciones: Muestra univariada, continuos, independientes y uniformemente distribuidos.
Se obtuvieron los siguientes resultados:

Tabla 3: Resultados de la prueba Anderson-Darling con diferentes tamaños de muestra y diferentes distribuciones.

Muestras	Anderson -Darling	Valores críticos %					Resultado de la hipótesis nula
Dist. Normal		15	10	5	2.5	1	
10	0.226	0.501	0.57	0.684	0.798	0.95	Probablemente Normal
20	0.304	0.506	0.577	0.692	0.807	0.96	Probablemente Normal
30	0.355	0.521	0.593	0.712	0.83	0.988	Probablemente Normal
40	0.514	0.531	0.605	0.726	0.847	1	Probablemente Normal
390	2.94	0.57	0.649	0.779	0.909	1.08	Probablemente no Normal
Dist. Exponencial		15	10	5	2.5	1	
10	1.42	0.87	1.017	1.26	1.51	1.84	P. no Exp
20	2.04	0.895	1.047	1.302	1.559	1.9	P. no Exp
30	2.8	0.904	1.05	1.3	1.57	1.91	P. no Exp
40	3.65	.908	1.06	1.321	1.58	1.92	P. no Exp
390	$st \to \infty$	0.92	1.07	1.33	1.6	1.95	P. no Exp
Dist. Logistica		25	10	5	2.5	1	0.5	
10	0.21	0.416	0.549	0.644	0.75	0.884	0.985	P. Logistica
20	0.23	0.421	0.556	0.652	0.76	0.895	0.998	P. Logistica
30	0.35	0.42	0.558	0.655	0.763	0.899	1	P. Logistica
40	0.5	0.423	0.56	0.656	0.764	0.9	1	P. no Logistica
390	1.55	0.426	0.563	0.66	0.769	0.9	1	P. no Logistica
Dist. Gumbel		25	10	5	2.5	1	
10	0.36	0.446	0.599	0.712	0.825	0.976	P. Gumbel
20	0.9	0.454	0.61	0.725	0.839	0.994	P. no Gumbel
30	0.98	0.457	0.615	0.73	0.846	1	P. no Gumbel
40	1.27	0.459	0.617	0.734	0.85	1	P. no Gumbel
390	15.69	0.469	0.631	0.749	0.868	1	P. no Gumbel

En donde se observa que con muestras pequeñas no hay mucha discrepancia entre los valores críticos de significancia, sin embargo, con un valor crítico que no coincida, se rechaza la hipótesis nula. Por parte del comportamiento de los valores estadísticos (Figura 14), se observa que todos incrementan en diferente medida (tasa de cambio), por lo que en muestras grandes también se ve afectado el valor estadístico de la prueba.

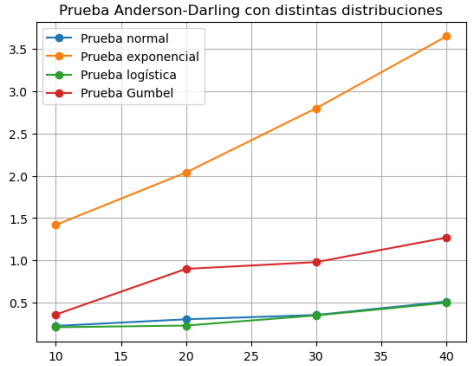


Figura 14: Comportamiento de la prueba Anderson-Darling con diferentes muestras y distribuciones.

6.5. Prueba de Pearson

Suposiciones: las observaciones en cada muestra son independientes e idénticamente distribuidas, Las observaciones en cada muestra siguen una distribución normal, Las observaciones en cada muestra tienen la misma varianza.

Se realizó la prueba de Pearson en los datos de costo de propiedad contra área, obteniendo el resultado siguiente:

$$r_p = -0.124 \qquad p_{valor} = 0.214$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Las muestras son independientes}$$

Observando que las variables tienen independencia pues, su distribución es muy cercana a la forma normal, cumpliendo con las suposiciones. Teniendo una correlación muy débil y negativa (tendencia de que mientras una variable incrementa la otra decrementa).

6.6. Prueba de Correlación de Rango de Spearman

Suposiciones: Las observaciones en cada muestra son independientes e idénticamente distribuidas (iid), las observaciones pueden (o no) ser clasificadas por rango en cada muestra.

Se realizó la prueba de Spearman en la base de datos de costo de propiedad contra área y se obtuvo el resultado siguiente:

$$r_s = -0.139 \qquad p_{valor} = 0.164$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Las muestras son independientes}$$

6.7. Prueba de correlación de rangos de Kendall

Suposiciones: Las observaciones en cada muestra son independientes e idénticamente distribuidas (iid), las observaciones pueden (o no) ser clasificadas por rango en cada muestra.

Se realizó la prueba de Kendall, obteniendo el resultado siguiente:

$$r_k = 0.202 \qquad p_{valor} = 0.011$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Las muestras son dependientes}$$

Como se observa, se tuvo una dependencia en las pruebas, siendo un valor bajo de p, significando que puede haber un umbral de incertidumbre. Lo que implicaría que los datos son caóticos.

6.8. Prueba Chi-Cuadrado

Suposiciones: Las observaciones utilizadas en el cálculo de la tabla de contingencia son independientes, hay 25 o más ejemplos en cada celda de la tabla de contingencia.

Se utilizó la base de datos de las calificaciones del ramen (Figura 8), obteniendo el siguiente resultado.

$$\chi^2 = 193.094 \qquad p_{valor} = 0.066$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Las muestras son independientes}$$

Por lo que en este caso se observa que hay independencia entre las variables, sin embargo, en la prueba anterior se obtuvo lo contrario. Esto se realizó con un margen muy bajo y cercano al intervalo de confianza, por lo que pudiera haber un umbral de incertidumbre en los datos, concluyendo que son caóticos y que a pesar de que muestran cierta relación, dependiendo de la prueba, puede dar resultados diferentes a la misma hipótesis.

6.9. Prueba de raíz unitaria aumentada Dickey-Fuller

Suposiciones: Las observaciones estan ordenadas temporalmente (series de tiempo).

Se realizó la prueba de Dickey-Fuller aumentada con los datos de temperatura en la ciudad de Manaos en el mes de marzo (Figura 9) y se obtuvo:

$$ADF_c = 1.315 \qquad p_{valor} = 0.622$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Hay una raíz unitaria (no estacionaria)}$$

Entonces esto significa que hay mucha probabilidad de que no sea estacionaria, y por lo tanto tenga alguna tendencia (ya sea ascendente o descendente).

6.10. Prueba de Kwiatkowski-Phillips-Schmidt-Shin (KPSS)

Suposiciones: Las observaciones estan ordenadas temporalmente (series de tiempo).

Se realizó la prueba de KPSS aumentada con los datos de temperatura en la ciudad de Manaos en el mes de marzo (Figura 9) y se obtuvo:

$$KPSS = 1.253 \qquad p_{valor} = 0.01$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Hay una raíz unitaria (no estacionaria)}$$

Entonces esto significa que hay mucha probabilidad de que no sea estacionaria, y por lo tanto tenga alguna tendencia (ya sea ascendente o descendente).

6.11. Prueba t de Student

Suposiciones: Las observaciones son independientes e idénticamente distribuidas, tienen una distribución normal y las observaciones en cada muestra tienen la misma varianza.

Se realizó la prueba t de Student para la base de datos que mide el tiempo de inmersión de una población de hombres y mujeres (véase Figura 10).

$$T_{student} = 0.313 \qquad p_{valor} = 0.75$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Las medias de las muestras son iguales}$$

Se observa visualmente que las distribuciones de los atributos son similares, al tratarse de una misma prueba para distintas poblaciones, es muy probable que tengan la misma distribución. Matematicamente hablando, se encontraban muy cercanas ($\sigma_M = 15.77 \approx \sigma_F = 15.7$ y $\mu_M = 32.62 \approx \mu_F = 32.23$)

6.12. Prueba t de Student emparejada

Suposiciones: Las observaciones son independientes e idénticamente distribuidas, tienen una distribución normal y las observaciones en cada muestra tienen la misma varianza.

Se realizó la prueba t de Student aumentada para la base de datos que mide el tiempo de inmersión de una población de hombres y mujeres (véase Figura 10).

$$T_{studentemparejada} = 0.320 \qquad p_{valor} = 0.749$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Las medias de las muestras son iguales}$$

Se obtuvo un resultado demasiado similar a la prueba de t de student y fue posible observar la tendencia tan marcada en los datos de tener medias iguales.

6.13. ANOVA

Suposiciones: Las observaciones son independientes e idénticamente distribuidas, tienen una distribución normal y las observaciones en cada muestra tienen la misma varianza.

Se utilizó la base de datos de tiempo de inmersión en toda la población de las personas que realizaron la inmersión (véase Figura 10). Obteniendo la siguiente resolución de la hipótesis.

$$ANOVA = 0.243 \qquad p_{valor} = 0.784$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Las medias de las muestras son iguales}$$

Observándose que los 3 atributos tienen valores muy similares pues como ya se comentó las distribuciones tienden a ser muy similares.

6.14. Prueba de U de Mann-Whitney

Suposiciones: Las observaciones son independientes e idénticamente distribuidas, observaciones en cada muestra pueden ser clasificadas.

Se realizó la prueba U de Mann-Whitney para la base de datos que mide el tiempo de inmersión de una población de hombres y mujeres (véase Figura 10). Obteniendo el siguiente resultado.

$$U_{cal} = 0.243 \qquad p_{valor} = 0.784$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Provienen de poblaciones idénticas}$$

Por lo que se puede observar, las características de los atributos son bastante similares, por lo que se puede decir que provienen del mismo tipo de distribución debido a la naturaleza del problema.

6.15. Prueba de rangos con signo de Wilcoxon

Suposiciones: Las observaciones en cada muestra son independientes e idénticamente distribuidas, las observaciones en cada muestra pueden ser ordenadas por rangos, Las observaciones entre las muestras están emparejadas.

Para esta prueba se utilizó la base de datos de la supervivencia de pacientes ante una enfermedad con un tratamiento, se comparará cada tratamiento con la supervivencia 11. Se obtuvo el siguiente resultado.

Tratamiento DX4

$$S^+ = 0 \qquad p_{valor} = 0.0.157$$
$$p_{valor} > (1 - 0.95) \rightarrow \text{Provienen de poblaciones idénticas}$$

Tratamiento DX5

$$S^+ = 0 \qquad p_{valor} = 0.317$$
$$p_{valor} > (1 - 0.95) \rightarrow \text{Provienen de poblaciones idénticas}$$

Tratamiento DX6

$$S^+ = 0 \qquad p_{valor} = 0.014$$
$$p_{valor} > (1 - 0.95) \rightarrow \text{Provienen de poblaciones diferentes}$$

6.16. H de kruskal-Wallis

Suposiciones: Las observaciones en cada muestra son independientes e idénticamente distribuidas, las observaciones en cada muestra pueden ser clasificadas.

Para esta prueba se utilizó la base de datos de la supervivencia de pacientes ante una enfermedad con un tratamiento, se comparará cada tratamiento con la supervivencia 11. Se obtuvo el siguiente resultado.

Tratamiento DX4

$$H = 2.06 \qquad p_{valor} = 0.0.151$$
$$p_{valor} > (1 - 0.95) \rightarrow \text{Provienen de poblaciones idénticas}$$

Tratamiento DX5

$$H = 1 \qquad p_{valor} = 0.317$$
$$p_{valor} > (1 - 0.95) \rightarrow \text{Provienen de poblaciones idénticas}$$

Se observa que únicamente el tratamiento DX6 es diferente a la población, por lo que se podría decir que es el menos efectivo de los 3 además de que no es cercana a las demás en cuanto a su efectividad.

Tratamiento DX6

$$H = 7.07 \qquad p_{valor} = 0.008$$
$$p_{valor} > (1 - 0.95) \rightarrow \text{Provienen de poblaciones diferentes}$$

Se observa que únicamente el tratamiento DX6 es diferente a la población, por lo que se podría decir que es el menos efectivo de los 3 además de que no es cercana a las demás en cuanto a su efectividad.

6.17. Prueba de Friedman

Suposiciones: Las observaciones en cada muestra son independientes e idénticamente distribuidas, las observaciones en cada muestra pueden ser clasificadas, las observaciones entre cada muestra están emparejadas.

En este caso se analizarán los 3 tratamientos entre si, con el fin de observar si hay parentesco en sus resultados y permiten tener una homogeneidad entre ellas para futuras muestras. Obteniendo el siguiente resultado.

$$X_r^2 = 5.25 \qquad p_{valor} = 0.072$$
$$p_{valor} > (1 - 0.95) \quad \rightarrow \text{Provienen de la misma población}$$

Se observa que resultaron salir de la misma población debido a la naturaleza del problema, por lo que se puede decir que tuvieron mucha homogeneidad y posiblemente un comportamiento similar para futuras muestras que puedan seguir

7. Conclusiones

La presente práctica mostró una investigación de los métodos de pruebas de hipótesis incluyendo un marco teórico solido que incluyo, sus formulas, casos de uso, aplicaciones, ejemplos y suposiciones de los datos para realizar las pruebas. Posteriormente se obtuvieron bases de datos con las características que requerían las pruebas, pudiendo observar como es que funcionaban y permitían sacar conclusiones de los resultados obtenidos tanto de una manera gráfica como tabular.

Todo esto tiene mayor sentido cuando se aplica en Machine Learning, pues permite obtener una referencia de los datos y conocerlos mejor, de esta manera podemos relacionar algunas tendencias en los datos y elegir mejores métodos de preprocesamiento, selección de modelo, elección de hiperparametros y también analizar los resultados de los mismos, desde el punto de vista de conjuntos y comprobación de resultados con validaciones (aprendizaje supervisado por ejemplo).

Finalmente se puede concluir que estos métodos son complementarios y permiten realizar un entendimiento mejor de los datos y los resultados mediante pruebas entre datos con el fin de observar tendencias, patrones o inclusive predecir su dependencia. Todos estos métodos pueden ser utilizados para comprobar de manera estadística y probabilística los resultados obtenidos, teniendo mayor certeza de las acciones que se realizan, las suposiciones y fiabilidad de los datos.

Referencias

[1] Minitab, "¿qué es una prueba de hipótesis?." https://support.minitab.com/es-mx/minitab/20/help-and-how-to/statistics/basic-statistics/supporting-topics/basics/what-is-a-hypothesis-test/. (Accessed on 06/03/2023).

[2] " prueba de shapiro-wilk: definición, cómo ejecutarla en spss en 2023 → statologos®." https://statologos.com/prueba-de-shapiro-wilk/#google_vignette. (Accessed on 06/03/2023).

[3] C. Zaiontz, "Shapiro-wilk test — real statistics using excel." https://real-statistics.com/tests-normality-and-symmetry/statistical-tests-normality-symmetry/shapiro-wilk-test/. (Accessed on 06/03/2023).

[4] J. A. Villasenor Alva and E. González Estrada, "A generalization of shapiro–wilk's test for multivariate normality," *Communications in Statistics - Theory and Methods*, vol. 38, no. 11, pp. 1870–1883, 2009.

[5] A. Darling and S. Wilk, "7.2.1.3. tests." https://www.itl.nist.gov/div898/handbook/prc/section2/prc213.htm. (Accessed on 06/03/2023).

[6] C. Zaiontz, "Skewness & kurtosis analysis — real statistics using excel." https://real-statistics.com/tests-normality-and-symmetry/analysis-skewness-kurtosis/. (Accessed on 06/03/2023).

[7] "1.3.5.14. anderson-darling test." https://www.itl.nist.gov/div898/handbook/eda/section3/eda35e.htm. (Accessed on 06/03/2023).

[8] "Microsoft word - correlacion.doc." https://personal.us.es/vararey/adatos2/correlacion.pdf. (Accessed on 06/04/2023).

[9] "Métodos y fórmulas para correlación - minitab." https://support.minitab.com/es-mx/minitab/20/help-and-how-to/statistics/basic-statistics/how-to/correlation/methods-and-formulas/methods-and-formulas/. (Accessed on 06/04/2023).

[10] "¿cómo calcular la correlación de rango de spearman en excel en 2023 ? statologos®." https://statologos.com/correlacion-de-rango-de-spearman-excel/. (Accessed on 06/04/2023).

[11] "¿qué es el coeficiente de correlación de spearman?." https://www.questionpro.com/blog/es/coeficiente-de-correlacion-de-spearman/. (Accessed on 06/04/2023).

[12] "Tau de kendall (coeficiente de correlación de rango de kendall) en 2023 → statologos®." https://statologos.com/kendalls-tau-2/. (Accessed on 06/04/2023).

[13] "Tau de kendall: definición + ejemplo en 2023 → statologos®." https://statologos.com/kendalls-tau/. (Accessed on 06/04/2023).

[14] J. Rodríguez, D. Rodríguez, S. Ramírez, V. Sosa, K. Serrano, and M. Velásquez, "Coeficiente de correlación de rango tau de kendall." Universidad Central de Venezuela, Mayo 2018.

[15] "Prueba de chi-cuadrado: ¿qué es y cómo se realiza?." https://www.questionpro.com/blog/es/prueba-de-chi-cuadrado-de-pearson/. (Accessed on 06/05/2023).

[16] "Prueba chi-cuadrado para tablas de contingencia — ¿cómo analizar?." https://gplresearch.com/prueba-chi-cuadrado/. (Accessed on 06/05/2023).

[17] "EstadÍstica bÁsica 2." https://www.ucm.es/data/cont/media/www/pag-54183/APUNTES%20ESTAD%C3%8DSTICA%203.pdf. (Accessed on 06/05/2023).

[18] "Raíz unitaria (dickey-fuller) y estacionalidad en excel — xlstat help center." https://help.xlstat.com/es/6697-raiz-unitaria-dickey-fuller-y-estacionalidad-en-excel. (Accessed on 06/05/2023).

[19] "Función de autocorrelación simple — 2023 — economipedia." https://economipedia.com/definiciones/funcion-de-autocorrelacion-simple.html. (Accessed on 06/05/2023).

[20] J. Hamilton, *Time Series Analysis*. Princeton, 1994.

[21] "Adf: prueba de dickey fuller aumentada en 2023 → statologos®." https://statologos.com/prueba-de-dickey-fuller-aumentada-adf/. (Accessed on 06/05/2023).

[22] "Métodos y fórmulas para prueba de dickey-fuller aumentada - minitab." https://support.minitab.com/es-mx/minitab/21/help-and-how-to/statistical-modeling/time-series/how-to/augmented-dickey-fuller-test/methods-and-formulas/methods-and-formulas/. (Accessed on 06/05/2023).

[23] M. A. Carnero, J. Olmo, and L. Pascual, "Modelling the dynamics of fuel and eu allowance prices during phase 3 of the eu ets," *Energies*, vol. 11, no. 11, p. 3148, 2018.

[24] "Kwiatkowski-phillips-schmidt-shin (kpss) test." https://rtmath.net/assets/docs/finmath/html/695835bf-570e-411f-9d76-05ee2570d0d7.htm. (Accessed on 06/05/2023).

[25] "¿qué son las pruebas paramétricas?." https://www.questionpro.com/blog/es/pruebas-parametricas/. (Accessed on 06/05/2023).

[26] "Pruebas paramétricas y no parametricas." https://enviromigration.files.wordpress.com/2016/04/pruebas-paramc3a9tricas-y-no-parametricas.pdf. (Accessed on 06/05/2023).

[27] "Prueba t: Qué es, ventajas y pasos para realizarla." https://www.questionpro.com/blog/es/prueba-t-de-student/. (Accessed on 06/06/2023).

[28] "Prueba t de student proyecto papime unam pe." https://slideplayer.es/slide/16982894/. (Accessed on 06/06/2023).

[29] "anova." https://www.uv.es/montes/biomecanica2004/anova. (Accessed on 06/06/2023).

[30] R. Lowry, *Concepts and Applications of Inferential Statistics*. VassarStats, 2014.

[31] "Anova (analisis de la varianza) — software estadístico excel." https://www.xlstat.com/es/soluciones/funciones/anova-ancova. (Accessed on 06/06/2023).

[32] "Tabla anova en excel: cómo crearla e interpretarla." https://www.ninjaexcel.com/formulas-y-funciones-de-excel/tabla-anova/. (Accessed on 06/06/2023).

[33] "Análisis de la varianza (anova)." https://www.probabilidadyestadistica.net/analisis-de-la-varianza-anova/. (Accessed on 06/06/2023).

[34] "¿cómo hacer la tabla anova con spss?.." https://estamatica.net/tabla-anova-con-spss/. (Accessed on 06/06/2023).

[35] IBM, "Pruebas no paramétricas." https://www.ibm.com/docs/es/spss-statistics/beta?topic=features-nonparametric-tests, 12 2021. (Accessed on 06/06/2023).

[36] "Prueba u de mann-whitney: Qué es y cómo funciona." https://www.questionpro.com/blog/es/prueba-u-de-mann-whitney/#:~:text=Un%20ejemplo%20del%20uso%20de,difieren%20en%20funci%C3%B3n%20del%20g%C3%A9nero. (Accessed on 06/06/2023).

[37] DATAtab, "Prueba de wilcoxon." https://datatab.es/tutorial/wilcoxon-test. (Accessed on 06/06/2023).

[38] "3.2 wilcoxon." https://www.uv.es/webgid/Inferencial/32_wilcoxon.html. (Accessed on 06/06/2023).

[39] "Prueba de los rangos con signo de wilcoxon." https://www.cienciadedatos.net/documentos/18_prueba_de_los_rangos_con_signo_de_wilcoxon. (Accessed on 06/06/2023).

Pruebas_Hipotesis

June 9, 2023

[1]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
```

```
C:\Users\aldoa\anaconda3\envs\env2\lib\site-packages\scipy\__init__.py:146:
UserWarning: A NumPy version >=1.16.5 and <1.23.0 is required for this version
of SciPy (detected version 1.23.5
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

[2]:
```
def norm_min_max(x,a,b): # Para todo el dataFrame
    l=list(x.columns)
    v_max=0
    v_min=0
    res=pd.DataFrame()
    for val in l:
        v_max=x[val].max()
        v_min=x[val].min()
        r_dt=v_max-v_min
        r_norm=b-a
        d=x[val]-v_min
        dpct=d/r_dt
        dnorm=r_norm*dpct
        data=a+dnorm
        aa=pd.DataFrame(data,columns=[val])
        res[val]=data
    return res
```

1 Preprocesamiento de datos

[3]:
```
#a0=pd.read_csv('flavors_of_cacao.csv')
a1=pd.read_csv('Pesos.csv')
a1.shape
```

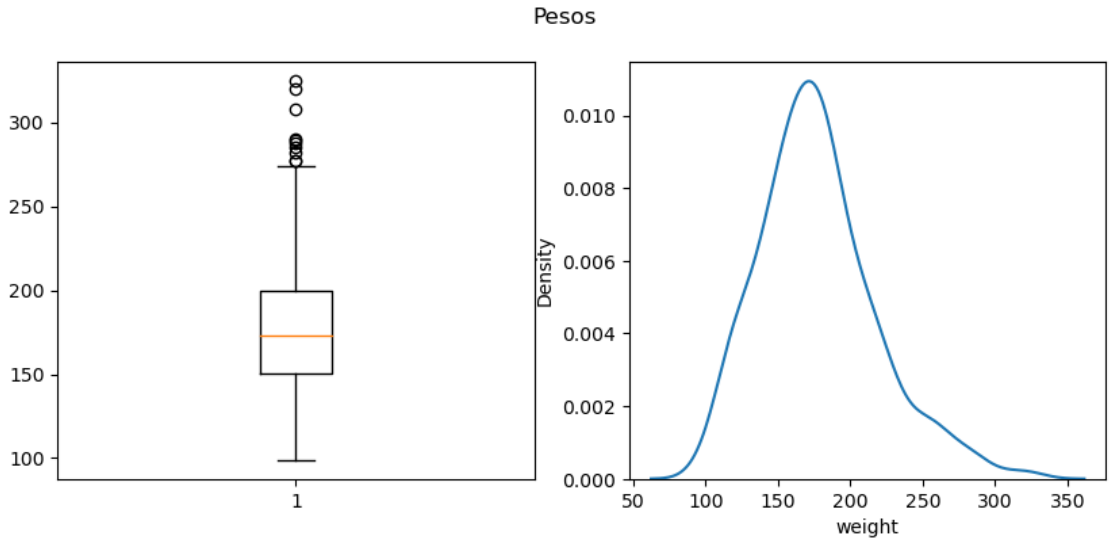
[3]: (390, 2)

[ ]:

1.1 Datos para pruebas de normalidad

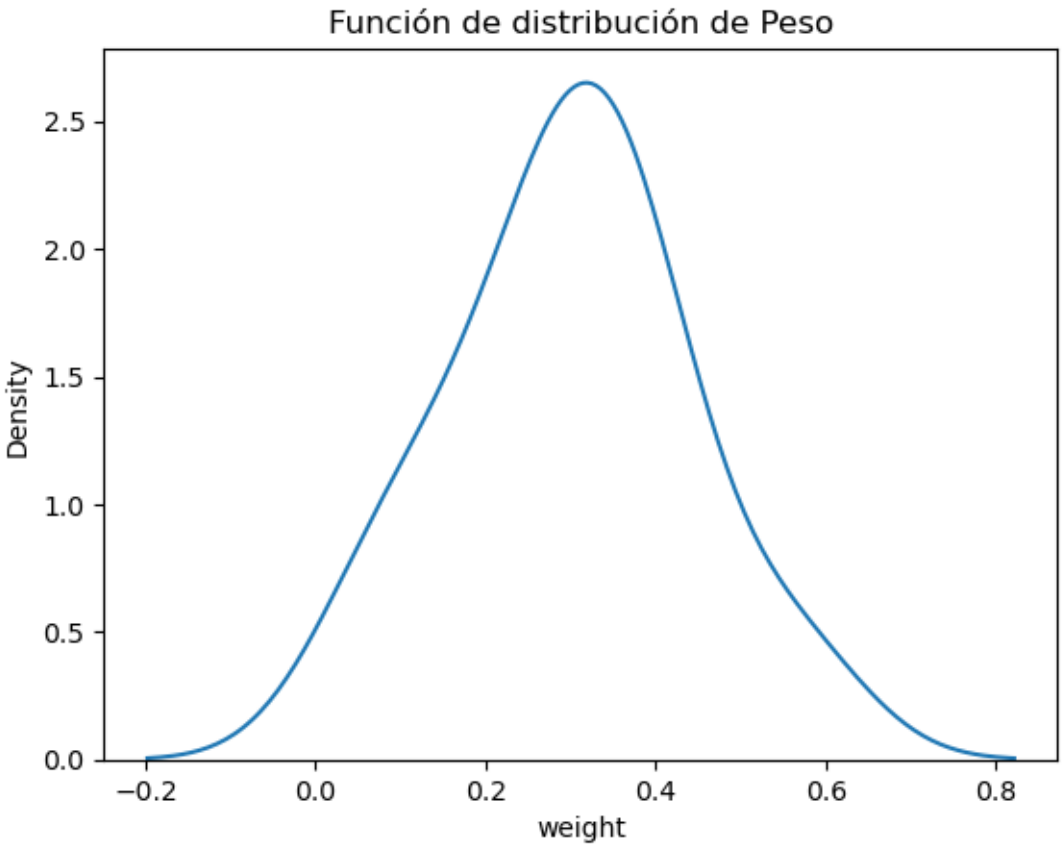
[4]:
```
fig, axs = plt.subplots(1,2,figsize=(10,4))
fig.suptitle('Pesos')

axs[0].boxplot(a1['weight'])
#axs[0].set_title('gg')
sns.kdeplot(a1['weight'],ax=axs[1])
#axs[1].set_title('gg')
```

[4]:
```
<AxesSubplot:xlabel='weight', ylabel='Density'>
```



[5]:
```
#plt.hist(a0['REF'])
#plt.boxplot(a1['height'])
#plt.hist(a1['height'],bins=6)
pp=norm_min_max(a1[['weight']],0,1)
np.random.seed(45)
d_sw=pp['weight'].sample(10)
#d_sw=a1['height'].sample(10)
sns.kdeplot(d_sw)
plt.title('Función de distribución de Peso')
```

[5]:
```
Text(0.5, 1.0, 'Función de distribución de Peso')
```



2 Pruebas de Normalidad

Plantean hipótesis nula que una muestra proviene de una normal, mediante un porcentaje de probabilidad

2.1 Shapiro-Wilk Test

2.1.1 Prueba si una muestra de datos tiene una distribución Gaussiana

La prueba de Shapiro-Wilk es una forma de saber si una muestra aleatoria proviene de una distribución normal . La prueba te da un valor W; los valores pequeños indican que su muestra no tiene una distribución normal (puede rechazar la hipótesis nula de que su población tiene una distribución normal si sus valores están por debajo de cierto umbral). La fórmula para el valor de W es:

$$W = \frac{(\sum_{i=1}^{m} a_i x_{(i)})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

donde: * n es la longitud de los datos * $m = \frac{n}{2}$ si la longitud es par y $m = \frac{n-1}{2}$ si es impar * x_i son los valores de muestra aleatorios ordenados * a_i son las constantes generadas a partir de las

covarianzas varianzas y medias de la muestra (tamaño n) de una muestra normalmente distribuida. Obteniendolo a partir de constantes y utilizando la tabla de pesos para n valores

- $x_{(i)} = x_{n-i+1} - x_i$, lo que significa la diferencia entre cada extremo de los datos ordenados

Por lo que se puede formalizar que

$$H_0 : \text{La distribución es normal}$$
$$H_1 : \text{La distribucion no es normal}$$

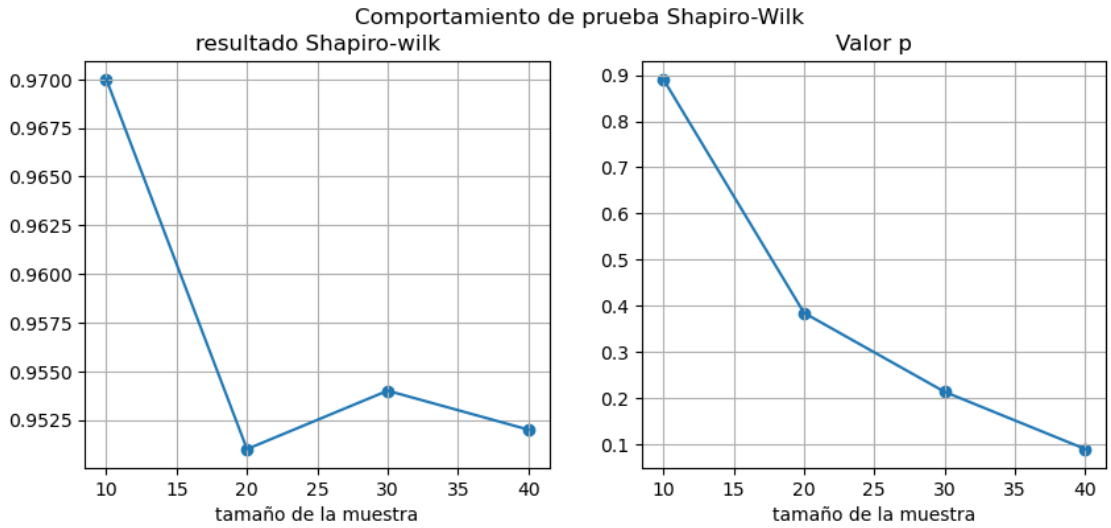
Esto implica que

$$H_0 : X \backsim \mathcal{N}(\mu, \sigma^2)$$
$$H_1 : X \nsim \mathcal{N}(\mu, \sigma^2)$$

[6]:
```
from scipy.stats import shapiro
data_s_w=d_sw#np.random.normal(0,10,4)#d_sw# np.random.uniform(0,1,52)#
stat, p = shapiro(data_s_w)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente Gaussiana')
else:
 print('Probablemente no Gaussiana')
```

```
stat=0.970, p=0.891
Probablemente Gaussiana
```

[7]:
```
fig, axs = plt.subplots(1,2,figsize=(10,4))
fig.suptitle('Comportamiento de prueba Shapiro-Wilk')
# Resultados
a=[0.970,0.951,0.954,0.952]
b=[0.891,0.385,0.214,0.09]
c=[10,20,30,40]
axs[0].plot(c,a)
axs[0].scatter(c,a)
axs[0].set_title('resultado Shapiro-wilk')
axs[0].set_xlabel('tamaño de la muestra')
axs[0].grid()
axs[1].plot(c,b)
axs[1].scatter(c,b)
axs[1].set_title('Valor p')
axs[1].set_xlabel('tamaño de la muestra')
axs[1].grid()
```



2.2 Prueba de D'Agostino´s

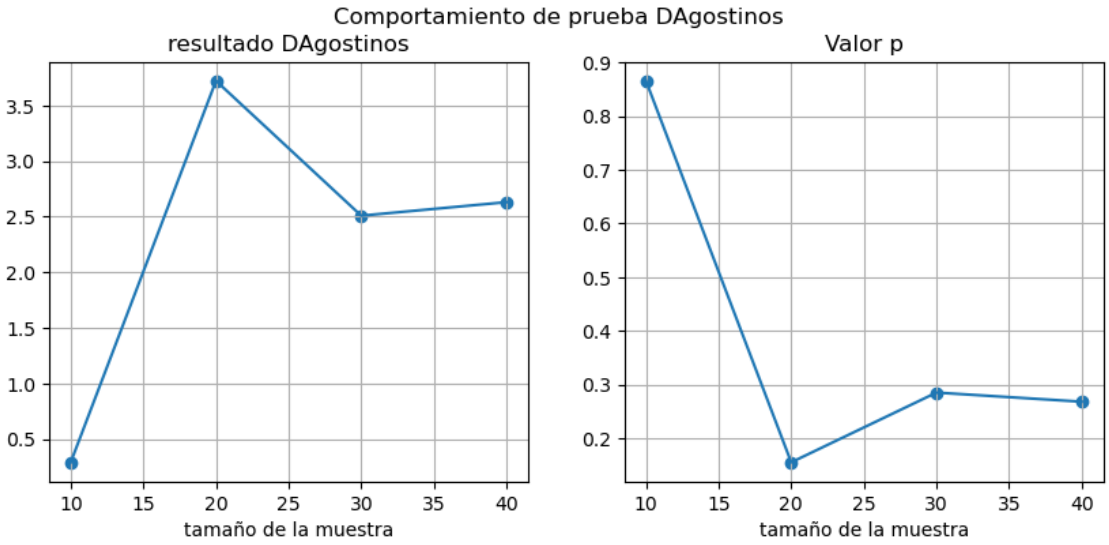
2.2.1 Prueba si una muestra de datos tiene una distribución Gaussiana

[8]:
```
# Example of the D'Agostino's K^2 Normality Test
np.random.seed(45)
d_sw=a1['weight'].sample(40)

#pp=norm_min_max(a1[['weight']],0,1)
#np.random.seed(45)
#d_sw=pp['weight'].sample(30)

from scipy.stats import normaltest
data = d_sw
stat, p = normaltest(data)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente Gaussiana')
else:
 print('Probablemente no Gaussiana')
```

```
stat=2.631, p=0.268
Probablemente Gaussiana
```

[9]:
```
fig, axs = plt.subplots(1,2,figsize=(10,4))
fig.suptitle('Comportamiento de prueba DAgostinos')
# Resultados
a=[0.289,
3.723,
2.509,
```

```
2.631]
b=[0.866,
0.155,
0.285,
0.268
]
c=[10,20,30,40]
axs[0].plot(c,a)
axs[0].scatter(c,a)
axs[0].set_title('resultado DAgostinos')
axs[0].set_xlabel('tamaño de la muestra')
axs[0].grid()
axs[1].plot(c,b)
axs[1].scatter(c,b)
axs[1].set_title('Valor p')
axs[1].set_xlabel('tamaño de la muestra')
axs[1].grid()
```



2.3 Prueba de Anderson-Darling

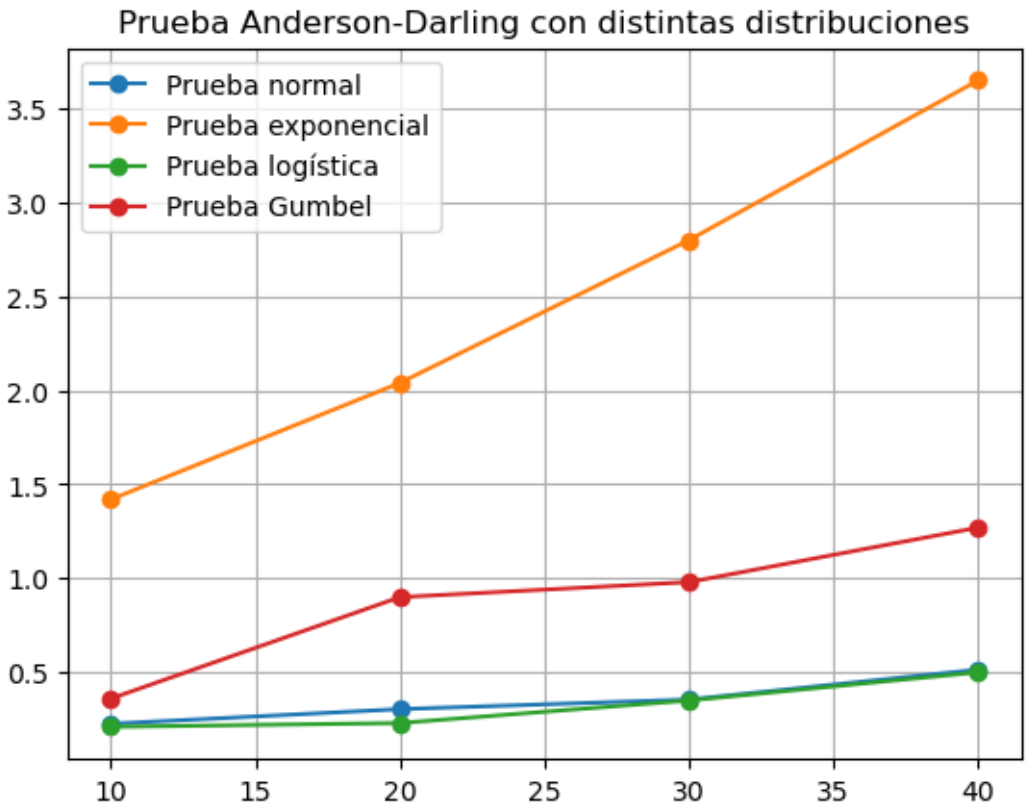
2.3.1 Prueba si una muestra de datos tiene una distribución Gaussiana

[10]:
```
from scipy.stats import anderson
#np.random.seed(45)
#d_sw=a1['weight'].sample(40)

pp=norm_min_max(a1[['weight']],0,1)
np.random.seed(45)
d_sw=pp['weight']#.sample(40)
```

```
result = anderson(d_sw,dist='gumbel') # 'norm', 'expon', 'logistic', 'gumbel'
print('Resultado estadistico= ',result.statistic)
for a in range(len(result.critical_values)):
    sv,cv=result.significance_level[a], result.critical_values[a]
    if result.statistic <cv:
        print(cv,'>',result.statistic,'No se puede rechazar la hipótesis nula de␣
 ↪',result.significance_level[a],'%')
    else:
        print(cv,'Se rechaza la hipótesis nula *****')
print(result.critical_values)
```

```
Resultado estadistico=  15.693702019429224
0.469 Se rechaza la hipótesis nula *****
0.631 Se rechaza la hipótesis nula *****
0.749 Se rechaza la hipótesis nula *****
0.868 Se rechaza la hipótesis nula *****
1.028 Se rechaza la hipótesis nula *****
[0.469 0.631 0.749 0.868 1.028]
```

[11]:
```
m=[10,20,30,40]
p_n=[0.226,0.304,0.355,0.514]
p_e=[1.42,2.04,2.8,3.65]
p_log=[0.21,0.23,0.35,0.5]
p_gum=[0.36,0.9,0.98,1.27]
plt.plot(m,p_n,linestyle='-', marker='o')
plt.plot(m,p_e,linestyle='-', marker='o')
plt.plot(m,p_log,linestyle='-', marker='o')
plt.plot(m,p_gum,linestyle='-', marker='o')
plt.legend(['Prueba normal', 'Prueba exponencial', 'Prueba logística','Prueba␣
 ↪Gumbel'])
plt.title('Prueba Anderson-Darling con distintas distribuciones')
plt.grid()
```



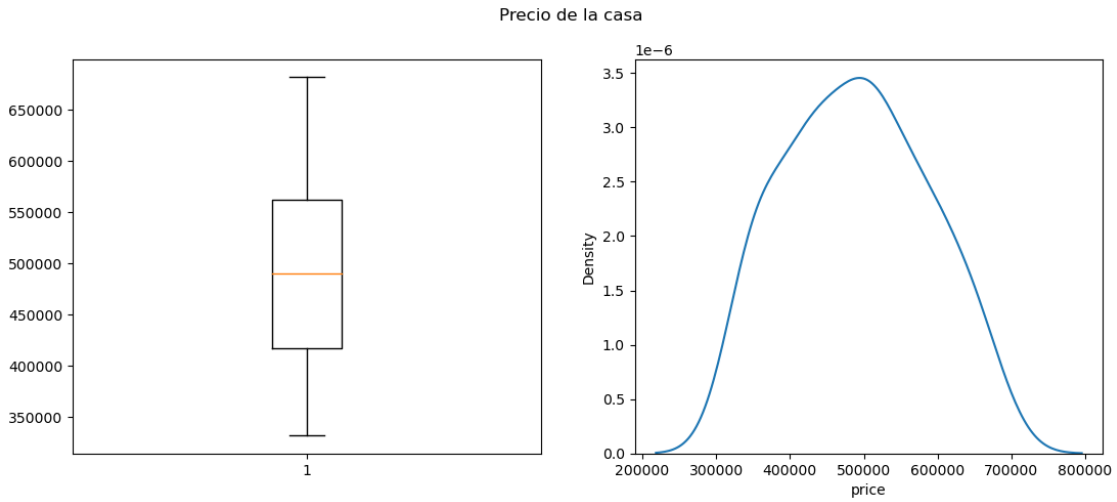
3 Datos para pruebas de correlación continuas

[12]:
```
## Base de datos
house=pd.read_csv('house_reg.csv')
print(house.shape)
print(house.columns)
print(house.describe())
```

```
(102, 3)
Index(['Unnamed: 0', 'price', 'sqft_lot'], dtype='object')
       Unnamed: 0          price      sqft_lot
count  102.000000     102.000000    102.000000
mean   207.568627  490029.394608   7753.401961
std    117.669659   95553.547161   1591.286908
min      4.000000  332000.000000   5150.000000
25%    108.750000  417000.000000   6380.000000
50%    218.000000  490000.000000   7674.000000
75%    300.750000  561875.000000   8992.500000
max    399.000000  681716.000000  11172.000000
```

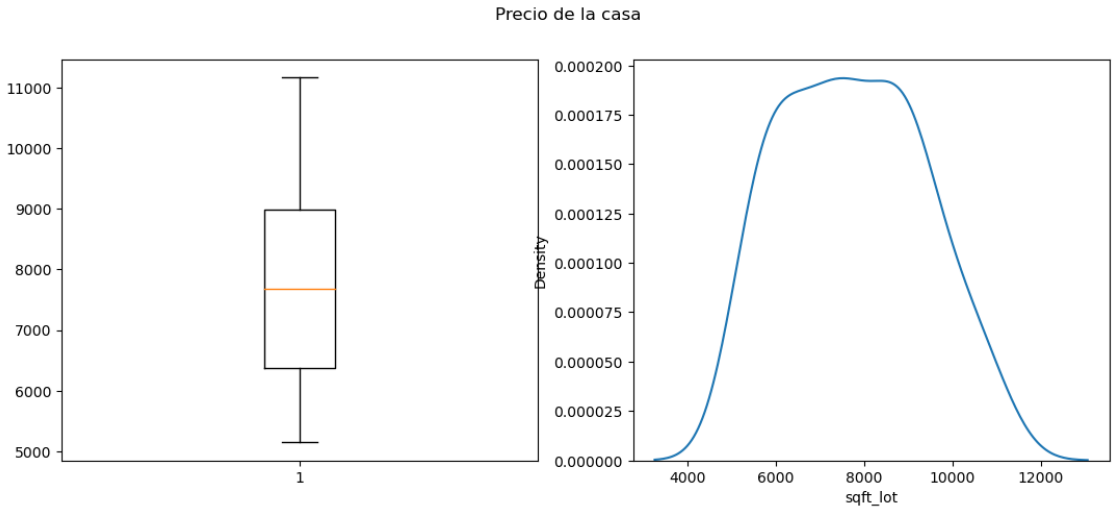
[13]:
```
fig, axs = plt.subplots(1,2,figsize=(13,5))
fig.suptitle('Precio de la casa')

axs[0].boxplot(house['price'])
#axs[0].set_title('gg')
sns.kdeplot(house['price'],ax=axs[1])
#axs[1].set_title('gg')
```

[13]: <AxesSubplot:xlabel='price', ylabel='Density'>



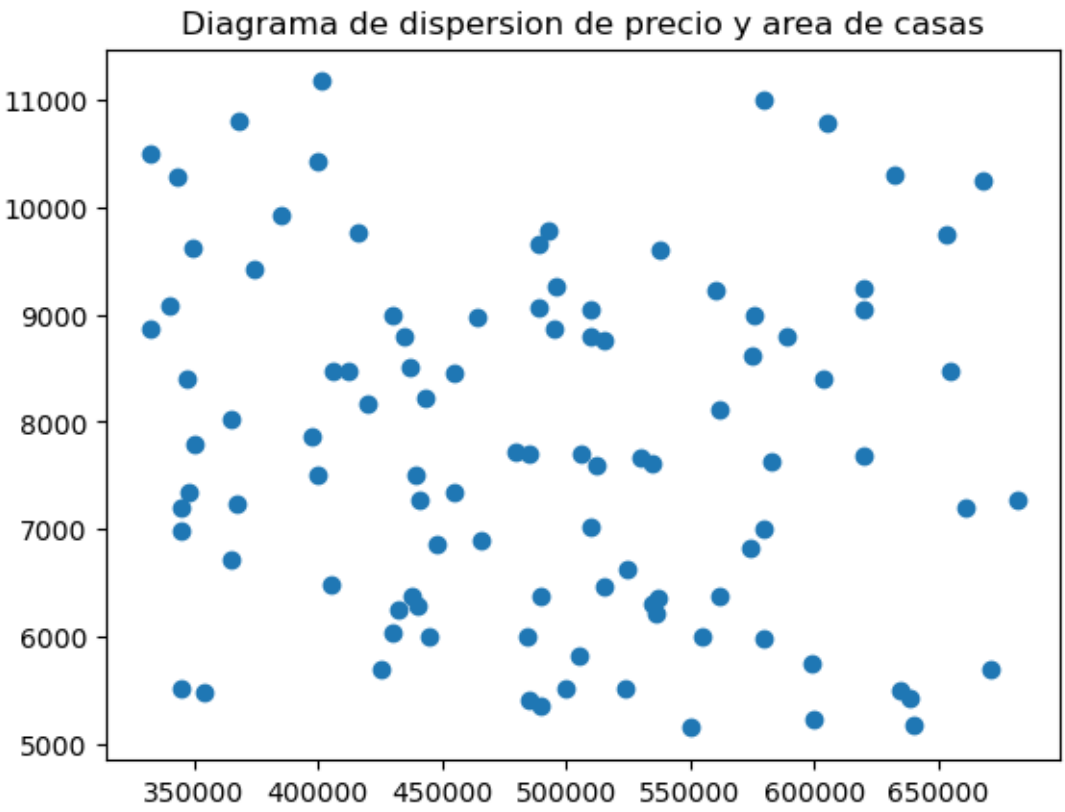
[14]:
```
fig, axs = plt.subplots(1,2,figsize=(13,5))
fig.suptitle('Precio de la casa')

axs[0].boxplot(house['sqft_lot'])
#axs[0].set_title('gg')
sns.kdeplot(house['sqft_lot'],ax=axs[1])
#axs[1].set_title('gg')
```

[14]: <AxesSubplot:xlabel='sqft_lot', ylabel='Density'>



[15]:
```
plt.scatter(house['price'],house['sqft_lot'])
plt.title('Diagrama de dispersion de precio y area de casas')
```

[15]: Text(0.5, 1.0, 'Diagrama de dispersion de precio y area de casas')



3.1 Prueba de Correlación de Pearson

3.1.1 Prueba si dos muestras tienen una relación lineal.

[16]:
```
from scipy.stats import pearsonr
data1 = house['price']
data2 = house['sqft_lot']
stat, p = pearsonr(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente independiente')
else:
 print('Probablemente dependiente')
```

```
stat=-0.124, p=0.214
Probablemente independiente
```

3.1.2 Correlación de rango de Spearman

[17]:
```
from scipy.stats import spearmanr
data1 = house['price']
data2 = house['sqft_lot']
stat, p = spearmanr(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
        print('Probablemente independiente')
else:
        print('Probablemente dependiente')
```

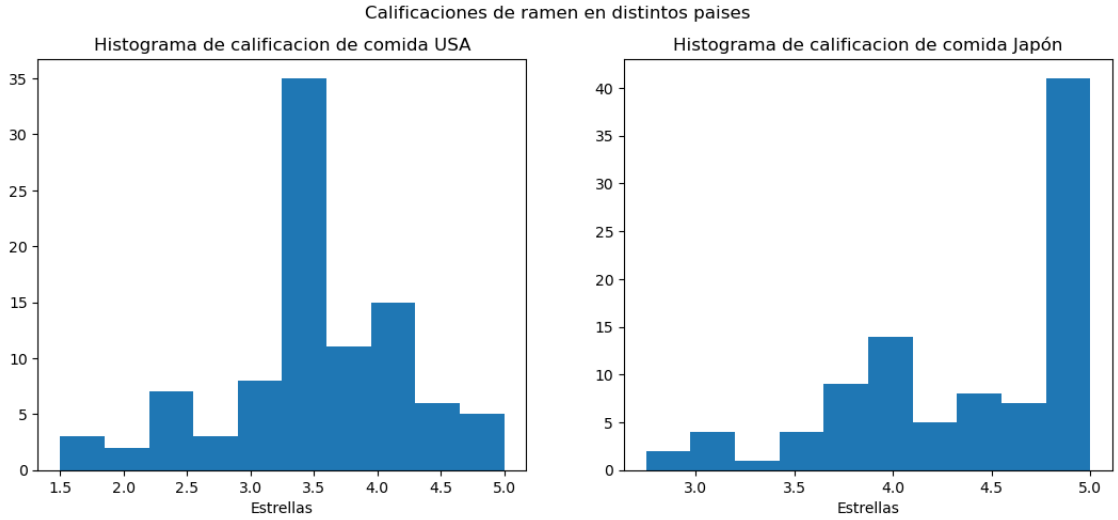
```
stat=-0.139, p=0.164
Probablemente independiente
```

3.1.3 Correlación de rangos de Kendall

[18]:
```
f_rt=pd.read_csv('Food_rating.csv')
```

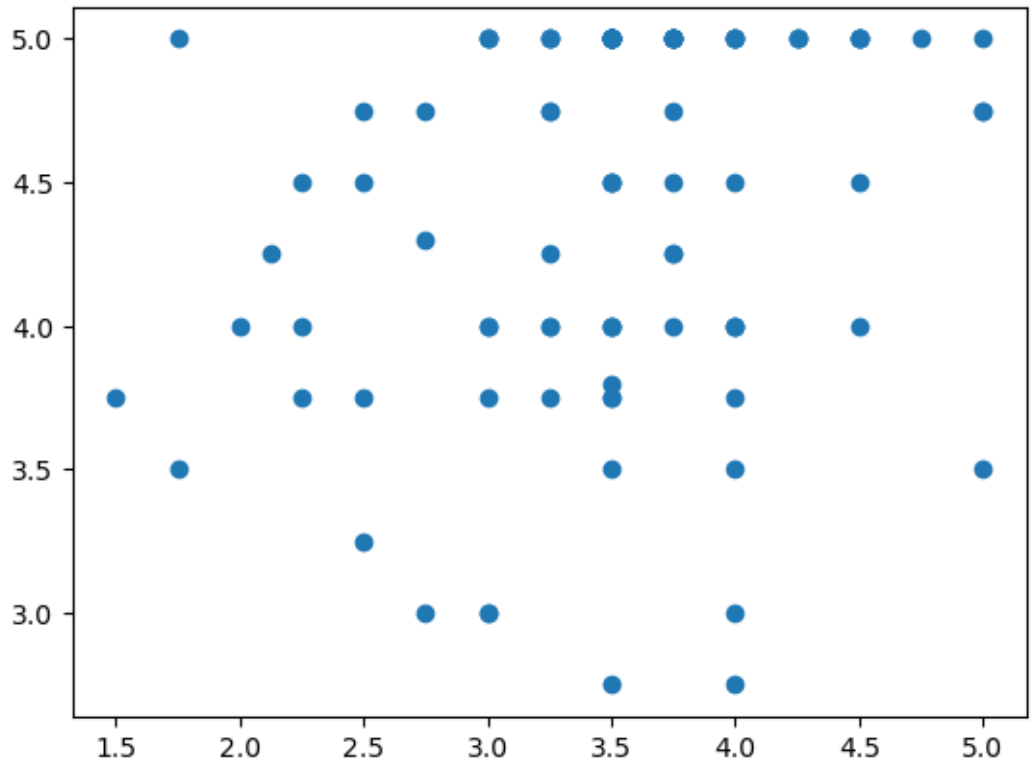
[19]:
```
fig, axs = plt.subplots(1,2,figsize=(13,5))
fig.suptitle('Calificaciones de ramen en distintos paises')

axs[0].hist(f_rt['usa'],bins=10)
axs[0].set_title('Histograma de calificacion de comida USA')
axs[0].set_xlabel('Estrellas')
axs[1].hist(f_rt['japan'],bins=10)
axs[1].set_title('Histograma de calificacion de comida Japón')
axs[1].set_xlabel('Estrellas')
```

[19]:
```
Text(0.5, 0, 'Estrellas')
```



[20]:
```
plt.scatter(f_rt['usa'],f_rt['japan'])
```

[20]: <matplotlib.collections.PathCollection at 0x26126a0ba90>



[21]:
```
from scipy.stats import kendalltau
data1 = f_rt['usa']
data2 = f_rt['japan']
stat, p = kendalltau(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
        print('Probablemente independiente')
else:
        print('Probablemente dependiente')
```

```
stat=0.202, p=0.011
Probablemente dependiente
```

3.1.4 Prueba Chi-Cuadrado

[22]:
```
from scipy.stats import chi2_contingency
tabla_contingencia=pd.crosstab(f_rt['japan'],f_rt['usa'])
stat, p, dof, expected = chi2_contingency(tabla_contingencia)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
    print('Probablemente independiente')
else:
    print('Probablemente dependiente')
```

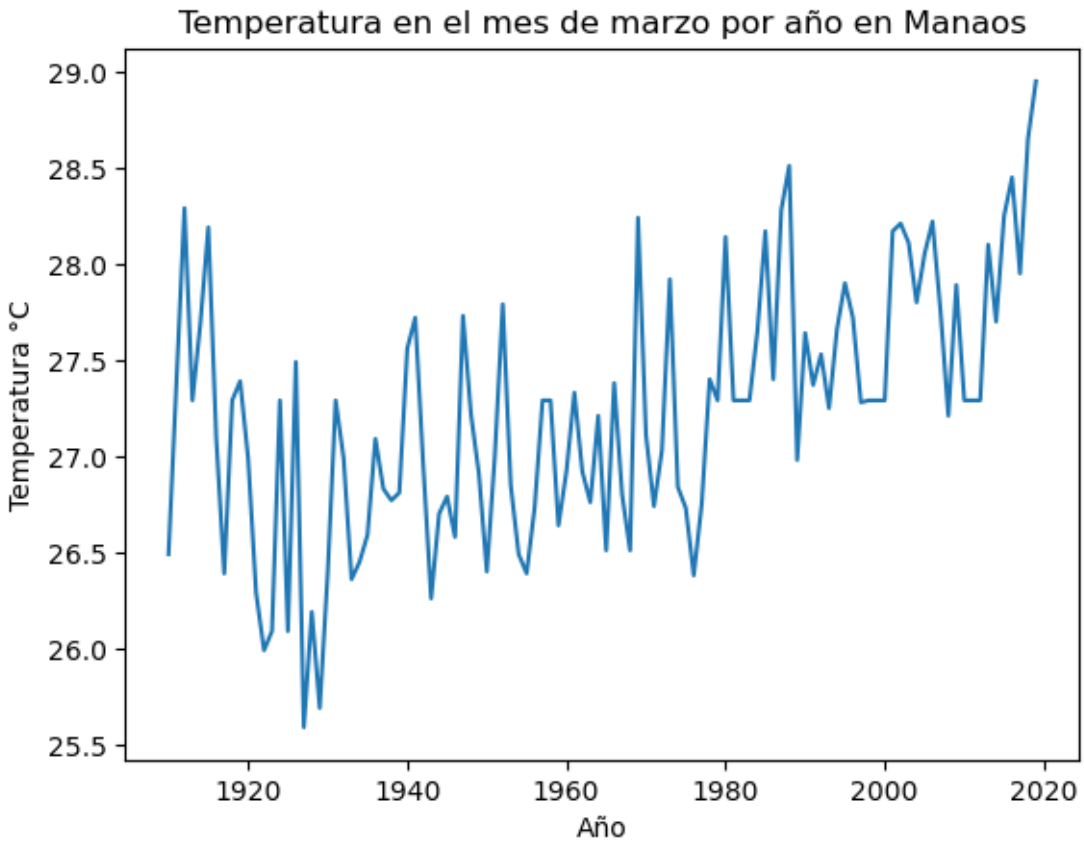
```
stat=193.094, p=0.066
Probablemente independiente
```

3.2 Base de datos para series de tiempo

Se utilizará una base de datos de la temperatura de manaos en Brasil. Por año y mes

[23]:
```
d_br=pd.read_csv('manaos_temp.csv')

plt.plot(d_br['YEAR'].to_numpy(),d_br['MAR'].to_numpy())
plt.title('Temperatura en el mes de marzo por año en Manaos')
plt.ylabel('Temperatura °C')
plt.xlabel('Año')
```

[23]:
```
Text(0.5, 0, 'Año')
```



3.3 Pruebas de Estacionalidad

3.3.1 Prueba de raíz unitaria aumentada Dickey-Fuller

[24]:
```
from statsmodels.tsa.stattools import adfuller

data = d_br['MAR']


tt=adfuller(data)
tt
```

[24]:
```
(-1.315104592720674,
 0.6222339255791948,
 4,
 105,
 {'1%': -3.4942202045135513,
  '5%': -2.889485291005291,
  '10%': -2.5816762131519275},
 154.93783642046546)
```

3.3.2 Prueba de Kwiatkowski-Phillips-Schmidt-Shin

[25]:
```
from statsmodels.tsa.stattools import kpss
data = d_br['MAR']
stat, p, lags, crit = kpss(data)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente estacionaria')
else:
 print('Probablemente no estacionaria')
```

```
stat=1.253, p=0.010
Probablemente no estacionaria
```

```
C:\Users\aldoa\anaconda3\envs\env2\lib\site-
packages\statsmodels\tsa\stattools.py:2018: InterpolationWarning: The test
statistic is outside of the range of p-values available in the
look-up table. The actual p-value is smaller than the p-value returned.

  warnings.warn(
```

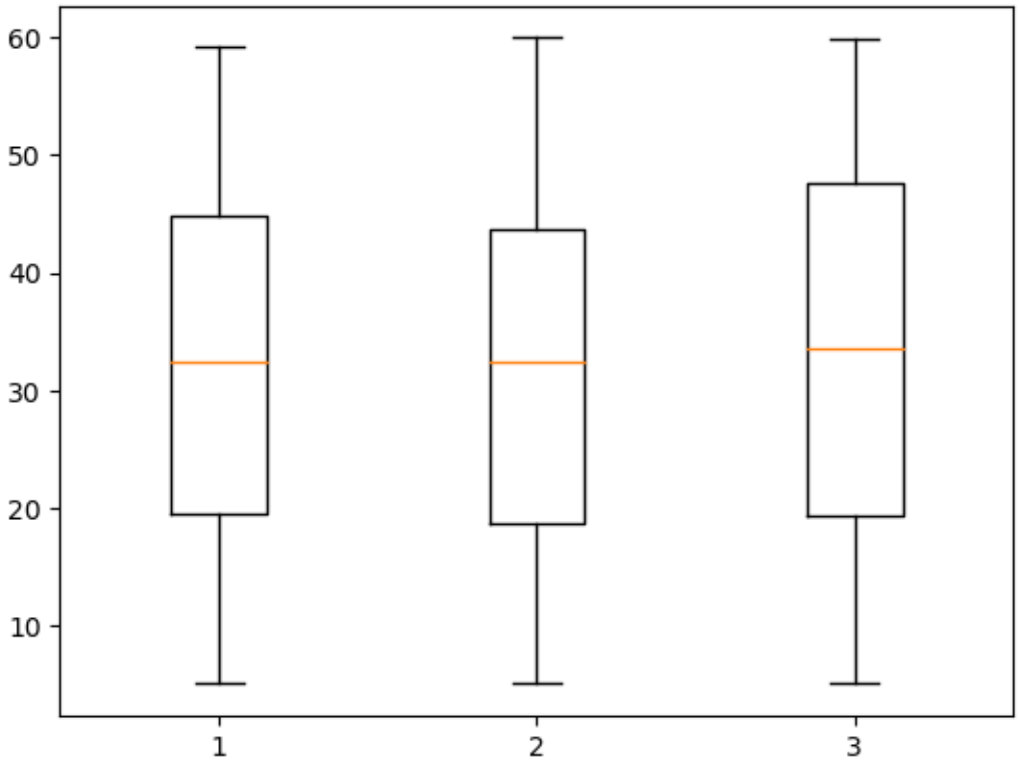
3.4 Base de datos para pruebas parámetricas

[26]:
```
dq=pd.read_csv('data_inmersion2.csv')
```

[27]:
```
plt.boxplot(dq)
```

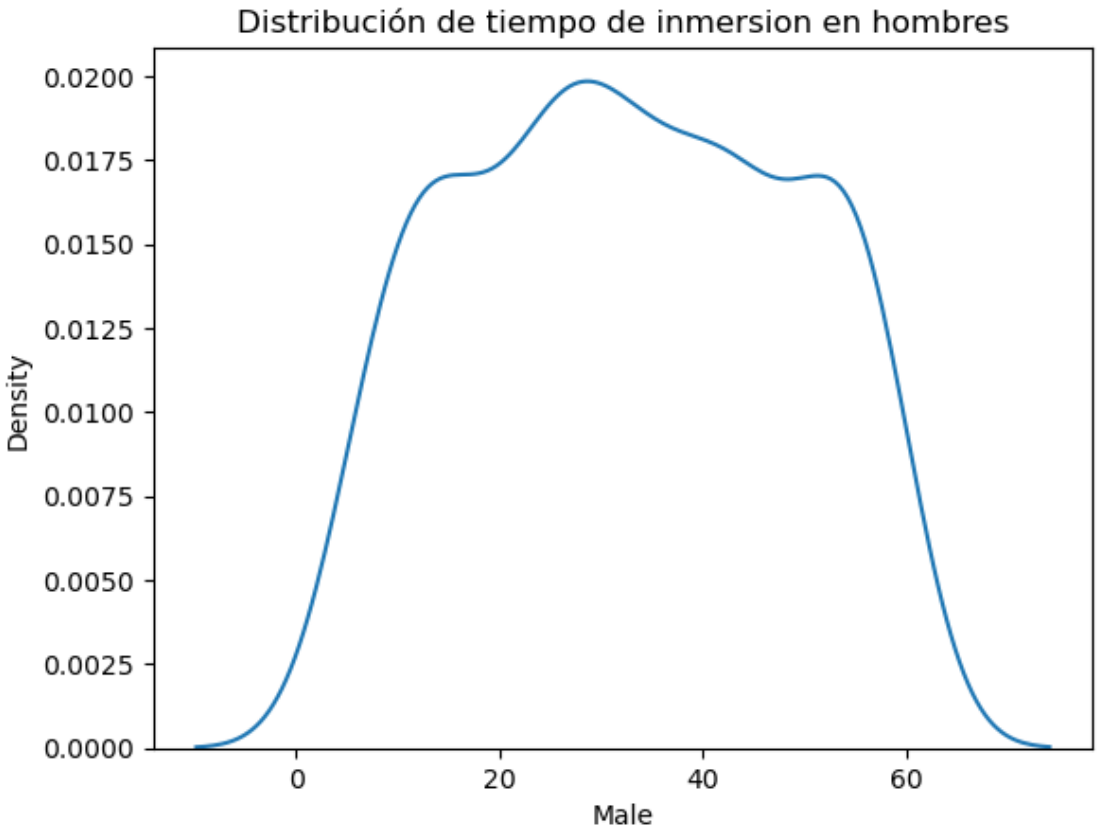
[27]:
```
{'whiskers': [<matplotlib.lines.Line2D at 0x26127efa9a0>,
  <matplotlib.lines.Line2D at 0x26127eec430>,
  <matplotlib.lines.Line2D at 0x26127f09c70>,
  <matplotlib.lines.Line2D at 0x26127f09f40>,
  <matplotlib.lines.Line2D at 0x261291130a0>,
  <matplotlib.lines.Line2D at 0x26129113370>],
 'caps': [<matplotlib.lines.Line2D at 0x26127efae20>,
  <matplotlib.lines.Line2D at 0x26127f09130>,
  <matplotlib.lines.Line2D at 0x26129105250>,
  <matplotlib.lines.Line2D at 0x26129105520>,
  <matplotlib.lines.Line2D at 0x26129113640>,
  <matplotlib.lines.Line2D at 0x26129113910>],
 'boxes': [<matplotlib.lines.Line2D at 0x26127efa6d0>,
  <matplotlib.lines.Line2D at 0x26127f099a0>,
  <matplotlib.lines.Line2D at 0x26129105d90>],
 'medians': [<matplotlib.lines.Line2D at 0x26127f09400>,
  <matplotlib.lines.Line2D at 0x261291057f0>,
  <matplotlib.lines.Line2D at 0x26129113be0>],
 'fliers': [<matplotlib.lines.Line2D at 0x26127f096d0>,
  <matplotlib.lines.Line2D at 0x26129105ac0>,
  <matplotlib.lines.Line2D at 0x26129113eb0>],
```

```
'means': []}
```



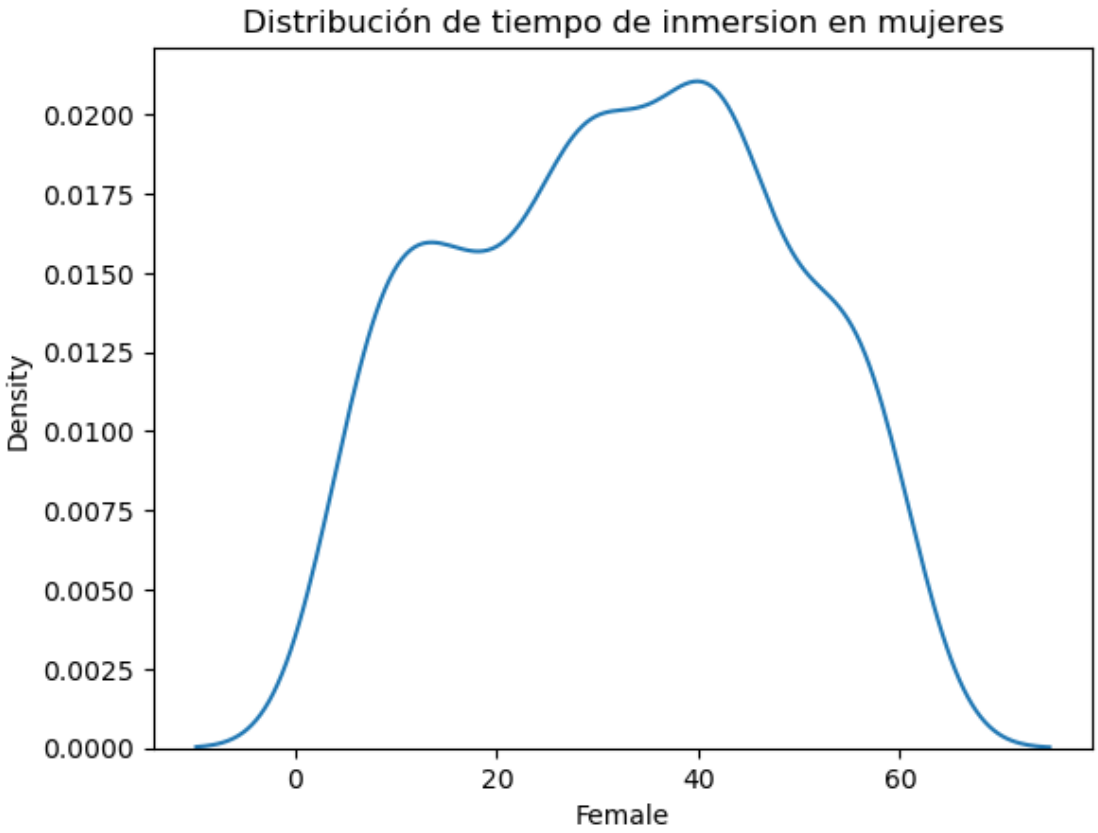
[28]:
```
sns.kdeplot(dq['Male'])
plt.title('Distribución de tiempo de inmersion en hombres')
print(dq.columns)
```

```
Index(['Male', 'Female', 'Other'], dtype='object')
```



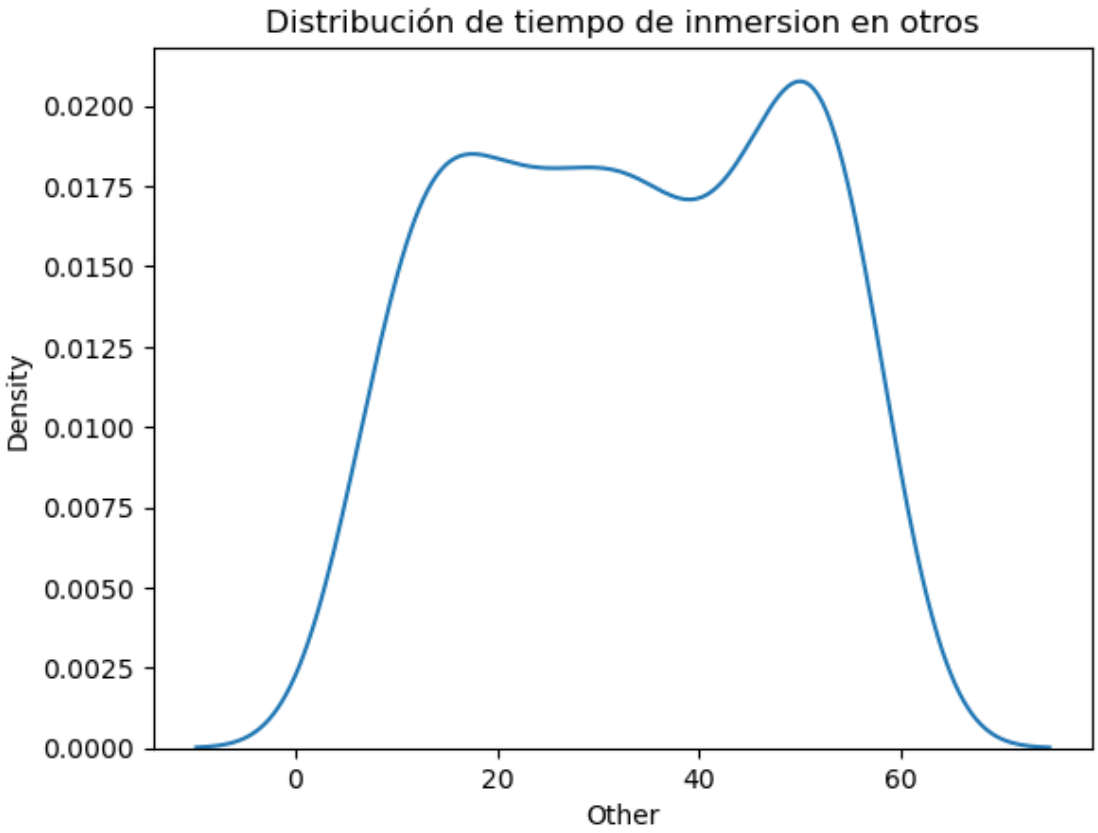
[29]:
```
sns.kdeplot(dq['Female'])
plt.title('Distribución de tiempo de inmersion en mujeres')
```

[29]:
```
Text(0.5, 1.0, 'Distribución de tiempo de inmersion en mujeres')
```



[30]:
```
sns.kdeplot(dq['Other'])
plt.title('Distribución de tiempo de inmersion en otros')
```

[30]:
```
Text(0.5, 1.0, 'Distribución de tiempo de inmersion en otros')
```



3.5 Pruebas de hipótesis estadísticas paramétricas

3.5.1 Prueba t de Student

[31]:
```
from scipy.stats import ttest_ind
data1 = dq['Male']
data2 = dq['Female']
stat, p = ttest_ind(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Las medias de las muestras son iguales')
else:
 print('Las medias de las muestras no son iguales')
```

```
stat=0.313, p=0.754
Las medias de las muestras son iguales
```

[32]:
```
dq.describe()
```

[32]:

	Male	Female	Other
count	325.000000	325.000000	325.000000

```
mean    32.626522   32.239297   33.098660
std     15.777011   15.738350   15.660991
min      5.095207    5.039439    5.008672
25%     19.481425   18.722275   19.300458
50%     32.331670   32.415662   33.510716
75%     44.772571   43.693336   47.655706
max     59.166574   59.983723   59.857616
```

3.5.2 Prueba t de Student emparejada

[33]:
```
from scipy.stats import ttest_rel
data1 = dq['Male']
data2 = dq['Female']
stat, p = ttest_rel(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente la misma distribución')
else:
 print('Probablemente diferente distribución')
```

```
stat=0.320, p=0.749
Probablemente la misma distribución
```

3.5.3 Prueba de Análisis de Varianza (ANOVA)

[34]:
```
from scipy.stats import f_oneway
data1 = dq['Male']
data2 = dq['Female']
data3 = dq['Other']
stat, p = f_oneway(data1, data2, data3)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente la misma distribución')
else:
 print('Probablemente diferente distribución')
```

```
stat=0.243, p=0.784
Probablemente la misma distribución
```

3.5.4 Prueba de U de Mann-Whitney

[35]:
```
from scipy.stats import mannwhitneyu
data1 = dq['Male']
data2 = dq['Female']
stat, p = mannwhitneyu(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente es la misma distribución')
```

```
else:
 print('Probablemente es diferente distribución')
```

```
stat=53359.000, p=0.820
Probablemente es la misma distribución
```

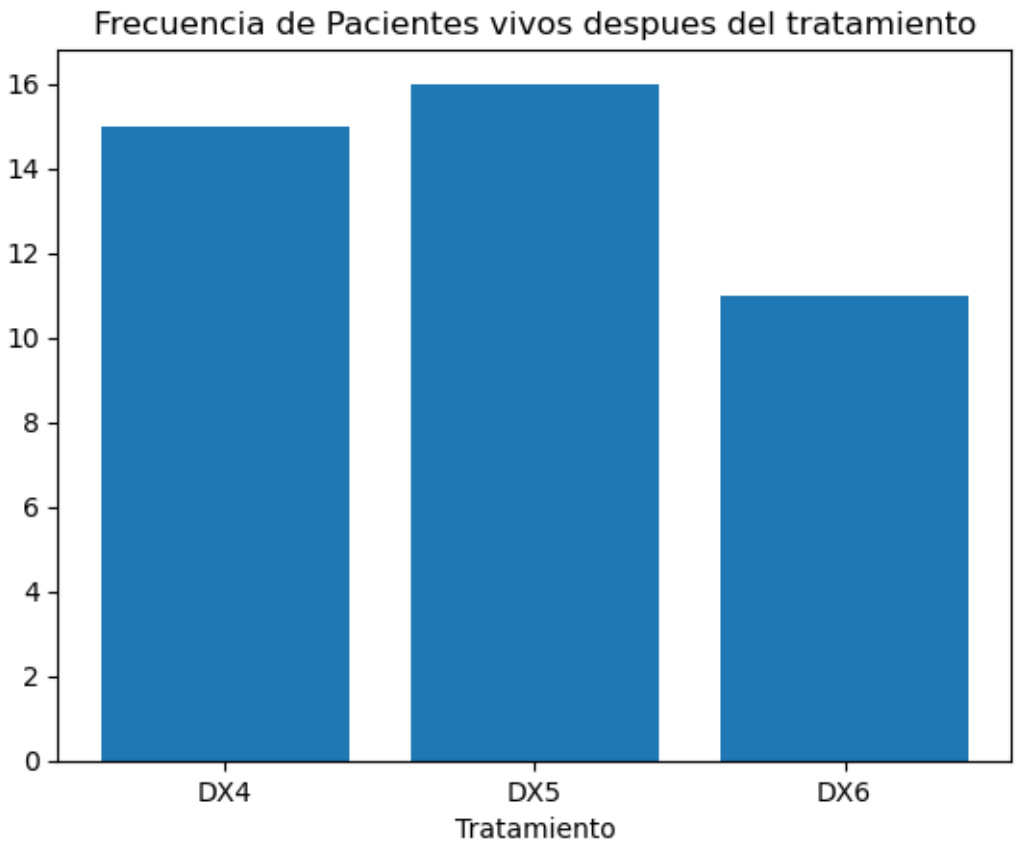
3.6 Base de Datos categoricos para wilcoxon

[36]:
```
rr=pd.read_csv('Tratamiento.csv')
rr.columns
```

[36]:
```
Index(['DX4', 'DX5', 'DX6', 'vivos'], dtype='object')
```

[37]:
```
counts=rr.drop('vivos',axis=1).sum()
plt.bar(counts.index,counts.values)
plt.title('Frecuencia de Pacientes vivos despues del tratamiento')
plt.xlabel('Tratamiento')
```

[37]:
```
Text(0.5, 0, 'Tratamiento')
```



3.6.1 Prueba de rangos con signo de Wilcoxon

[38]:
```
from scipy.stats import wilcoxon
data1 = rr['vivos']
data2 = rr['DX6']
stat, p = wilcoxon(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente es la misma distribución')
else:
 print('Probablemente es diferente distribución')
```

```
stat=0.000, p=0.014
Probablemente es diferente distribución
```

```
C:\Users\aldoa\anaconda3\envs\env2\lib\site-
packages\scipy\stats\morestats.py:3141: UserWarning: Exact p-value calculation
does not work if there are ties. Switching to normal approximation.
  warnings.warn("Exact p-value calculation does not work if there are "
C:\Users\aldoa\anaconda3\envs\env2\lib\site-
packages\scipy\stats\morestats.py:3155: UserWarning: Sample size too small for
normal approximation.
  warnings.warn("Sample size too small for normal approximation.")
```

3.6.2 Prueba H de Kruskal-Wallis

[39]:
```
from scipy.stats import kruskal
data1 = rr['vivos']
data2 = rr['DX6']
stat, p = kruskal(data1, data2)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente es la misma distribución')
else:
 print('Probablemente es diferente distribución')
```

```
stat=7.071, p=0.008
Probablemente es diferente distribución
```

3.6.3 Prueba de Friedman

[40]:
```
from scipy.stats import friedmanchisquare
data1 = rr['DX4']
data2 = rr['DX5']
data3 = rr['DX6']
stat, p = friedmanchisquare(data1, data2, data3)
print('stat=%.3f, p=%.3f' % (stat, p))
if p > 0.05:
 print('Probablemente es la misma distribución')
```

```
else:
 print('Probablemente es diferente distribución')
```

```
stat=5.250, p=0.072
Probablemente es la misma distribución
```