



Probabilidad y Funciones_

Sesión Presencial 2

Activación de Conceptos

¿Cómo podemos obtener la frecuencia de una `pd.Series`?

- `pd.Series(array).counts()`
- Contando los elementos uno por uno en un loop.
- `df['array'].value_counts()`

¿Cómo podemos pasar esta frecuencia a un porcentaje?

- No se puede
- `df['array'].value_counts('%')`
- `df['array'].value_counts() / len(df)`

Si sabemos que 8 de cada 10 gatos prefieren Whiskas ¿Cuál es la probabilidad de elegir al azar un gato que no prefiera Whiskas?

- 2%
- 20%
- 2:10
- 1/5

A partir de la siguiente tabla

| Departamento | Hombres | Mujeres | Total |
|--------------|---------|---------|-------|
| A | 825 | 108 | 933 |
| B | 560 | 25 | 585 |
| C | 325 | 593 | 918 |
| D | 417 | 375 | 792 |
| E | 191 | 393 | 584 |
| F | 373 | 341 | 714 |
| Total | 2691 | 1835 | 4526 |

- ¿Cuál es la probabilidad de elegir un hombre en el departamento A?

A partir de la siguiente tabla

| Departamento | Hombres | Mujeres | Total |
|--------------|---------|---------|-------|
| A | 825 | 108 | 933 |
| B | 560 | 25 | 585 |
| C | 325 | 593 | 918 |
| D | 417 | 375 | 792 |
| E | 191 | 393 | 584 |
| F | 373 | 341 | 714 |
| Total | 2691 | 1835 | 4526 |

- ¿Cuál es la probabilidad de elegir una mujer del departamento C en el total de la muestra?

A partir de la siguiente tabla

| Departamento | Hombres | Mujeres | Total |
|--------------|---------|---------|-------|
| A | 825 | 108 | 933 |
| B | 560 | 25 | 585 |
| C | 325 | 593 | 918 |
| D | 417 | 375 | 792 |
| E | 191 | 393 | 584 |
| F | 373 | 341 | 714 |
| Total | 2691 | 1835 | 4526 |

- Dentro del departamento F ¿Cuál es la probabilidad de elegir un hombre al azar?

A partir de la siguiente tabla

| Departamento | Hombres | Mujeres | Total |
|--------------|---------|---------|-------|
| A | 825 | 108 | 933 |
| B | 560 | 25 | 585 |
| C | 325 | 593 | 918 |
| D | 417 | 375 | 792 |
| E | 191 | 393 | 584 |
| F | 373 | 341 | 714 |
| Total | 2691 | 1835 | 4526 |

- Entre las mujeres ¿Cuál es el departamento más probable de ocurrir si eligo una mujer al azar?

Funciones

{desafío}
latam_

¿Qué es una función?

- Permiten abstraernos del código para resolver un problema.
- Facilitan la reutilización de código.

```
In [4]: import math

def hipotenusa(x, y):
    tmp = (x ** 2 + y ** 2)
    tmp = math.sqrt(tmp)
    return tmp
```

```
In [6]: hipotenusa(3, 4)
```

```
Out[6]: 5.0
```

D.R.Y

- **DRY: Don't Repeat Yourself.** Si vamos a utilizar una expresión más de 2 veces, es mejor generar una función a partir de ella. No solo es una práctica inteligente, también evita problemas en el código al compartir definiciones y procedimientos.

Anatomía de una función

```
In [7]: def saludar():  
        print('Hola mundo!')  
  
saludar()
```

Hola mundo!

- Declaración def: Mediante ella señalamos que todo lo escrito será considerado como una función.
- Nombre de la función saludar(): Este es el identificador que utilizaremos para llamarla posteriormente. Cabe destacar que los paréntesis y el doble punto al final de la función son obligatorias.
- La indentación (sangrado) es parte importante, le permite a python saber que parte del código escrito pertenece a la función.
- Expresiones a ejecutar print('Hola Mundo!') estas expresiones tienen que estar correctamente indentadas, el estándar es de **4 espacios**.

Parámetros de una función

- Las funciones permiten aplicar código a nuevos objetos de nuestro entorno de trabajo.
- Para ello podemos declarar valores que podemos ingresar a una función y que genere un retorno.

```
In [ ]: def saludar(nombre):  
        print("Hola", nombre, "!")
```

```
In [8]: saludar("Ignacio")
```

```
Hola Ignacio !
```

Múltiples parámetros

- Una función puede aceptar múltiples parámetros al separarlos con una ,

```
In [ ]: def saludar(nombre, apellido):  
        print("Hola", nombre, apellido, "!")
```

```
In [9]: saludar("Perico", "Los Palotes")
```

```
Hola Perico Los Palotes !
```

Parámetros con valores por defecto

- Se puede generar una función donde se inserten valores en caso que no se especifique un parámetro en la llamada de la función.

```
In [ ]: def saludar(nombre, apellido = "Los Palotes"):
        print("Hola", nombre, apellido, "!")
```

```
In [10]: saludar("perico")
```

```
Hola perico Los Palotes !
```


Return

- Si deseamos seguir utilizando los valores creados por una función, debemos incluir return como la última línea a evaluar.

```
In [ ]: def elevar(x, b=2):  
        return x ** b
```

```
In [11]: elevar(8)
```

```
Out[11]: 64
```

Refactorización y Validación

```
In [54]: def fetch_descriptives(x):
          if isinstance(x, pd.Series) is True:
              if x.dtype != 'object':
                  x = x.dropna()
                  mu = round(np.mean(x), 3)
                  median = np.median(x)
                  mode = stats.mode(x)[0][0]
                  sigma = round(np.std(x), 3)
              else:
                  raise ValueError('El objeto pd.Series no presenta valores numéricos.')
          else:
              raise ValueError('El tipo de dato ingresado no es un objeto pd.Series')
          return mu, median, mode, sigma
```