Campus Querétaro

Herramientas computacionales: el arte de la analítica (Gpo 101)

Semestre: Febrero-Junio de 2024

Actividad 5 - Estadística básica

Docente

Prof. Pedro Oscar Pérez Murueta

Equipo

Edgar Roann Santillán Bernal | A00572737

```
[ ] file_path = "/gdrive/MyDrive/SemanaTec/datasets/insurance.csv"
```

```
[ ] # Carga el conjunto de datos al ambiente de Google Colab y muestra los primeros
    # 6 renglones.
    df = pd.read_csv(file_path)


    print(df.head(6))
```

```
   age     sex     bmi  children smoker     region      charges
0   19  female  27.900         0    yes  southwest  16884.92400
1   18    male  33.770         1     no  southeast   1725.55230
2   28    male  33.000         3     no  southeast   4449.46200
3   33    male  22.705         0     no  northwest  21984.47061
4   32    male  28.880         0     no  northwest   3866.85520
5   31  female  25.740         0     no  southeast   3756.62160
```

El conjunto de datos contiene información demográfica sobre los asegurados en una compañía de seguros:

- **age**: Edad del asegurado principal
- **sex**: Género del asegurado. female o male
- **bmi**: Índice de masa corporal
- **children**: Número de hijos que estan cubiertos con la poliza.
- **smoke**: ¿El beneficiario fuma? (*yes/no*)
- **region**: ¿Dónde vive el beneficiario? Estos datos son de Estados Unidos. Regiones disponibles: northeast, southeast, southwest, northwest
- **charges**: Costo del seguro.

```
# Crea una tabla resumen con los estadísticas generales de las variables
# numéricas.

# Crea una tabla resumen con las estadísticas generales de las variables numéricas.
summary_table = df.describe()

# Muestra la tabla resumen.
print(summary_table)
```

```
                age          bmi     children       charges
count  1338.000000  1338.000000  1338.000000   1338.000000
mean     39.207025    30.663397     1.094918  13270.422265
std      14.049960     6.098187     1.205493  12110.011237
min      18.000000    15.960000     0.000000   1121.873900
25%      27.000000    26.296250     0.000000   4740.287150
50%      39.000000    30.400000     1.000000   9382.033000
75%      51.000000    34.693750     2.000000  16639.912515
max      64.000000    53.130000     5.000000  63770.428010
```

```
[ ] # ¿Cómo se correlacionan las varaibles numéricas entre sí?
    # Calcula la matriz de correlación entre las variables numéricas.
    correlation_matrix = df.corr()

    # Muestra la matriz de correlación.
    print(correlation_matrix)
```

```
                age       bmi  children   charges
age        1.000000  0.109272  0.042469  0.299008
bmi        0.109272  1.000000  0.012759  0.198341
children   0.042469  0.012759  1.000000  0.067998
charges    0.299008  0.198341  0.067998  1.000000
<ipython-input-9-19b4fe39a5af>:3: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only val
  correlation_matrix = df.corr()
```

```
[ ] # Determina si existe o no una correlación entre el índice de masa corporal
    # (bmi) y el costo del seguro.
    #Respuesta:
    #el coeficiente de correlación entre bmi y charges es aproximadamente 0.198341. Dado que este valor está más cerca de 0 que de 1, indica una correlación positiva débil entre el

    #Por lo tanto, sí existe cierta correlación entre el índice de masa corporal y el costo del seguro, pero no es muy significativa.
```

```
# ¿Cuántas personas aseguradas son hombre y cuántas son mujeres?

gender_counts = df['sex'].value_counts()


print(gender_counts)
```

```
male      676
female    662
Name: sex, dtype: int64
```

```
[ ] # ¿Cuántos hombres y mujeres asegurados viven en cada región?
    # Agrupa el DataFrame por la columna 'region' y 'sex', luego cuenta el número de cada categoría.
    gender_region_counts = df.groupby(['region', 'sex']).size()

    # Muestra el conteo de hombres y mujeres en cada región.
    print(gender_region_counts)
```

```
region     sex
northeast  female    161
           male      163
northwest  female    164
           male      161
southeast  female    175
```

```python
# En promedio, ¿quién paga más de cuota de seguro? ¿Los fumadores o los no
# fumadores? Muéstralo con los datos.


average_charges_smoker = df[df['smoker'] == 'yes']['charges'].mean()
average_charges_non_smoker = df[df['smoker'] == 'no']['charges'].mean()


print("Promedio de cuota de seguro para fumadores:", average_charges_smoker)
print("Promedio de cuota de seguro para no fumadores:", average_charges_non_smoker)
```

```
Promedio de cuota de seguro para fumadores: 32050.23183153284
Promedio de cuota de seguro para no fumadores: 8434.268297856204
```

```python
[ ] # ¿Cuáles son las cuotas mínimas y máximas que las personan pagan dependiendo
    # del género y del número de hijos?

    # Encuentra las cuotas mínimas y máximas pagadas por género y número de hijos.
    min_max_charges = df.groupby(['sex', 'children'])['charges'].agg(['min', 'max'])

    # Muestra las cuotas mínimas y máximas pagadas por género y número de hijos.
    print(min_max_charges)
```

```
                      min          max
sex    children
female 0         1607.51010  63770.42801
       1         2201.09710  58571.07448
       2         2801.25880  47305.30500
       3         4234.92700  46661.44240
```

```python
[ ] # ¿Cuáles son las cuotas mínimas y máximas que las personan pagan dependiendo
    # del género y del número de hijos?


    min_max_charges = df.groupby(['sex', 'children'])['charges'].agg(['min', 'max'])

    print(min_max_charges)
```

```
                      min          max
sex    children
female 0         1607.51010  63770.42801
       1         2201.09710  58571.07448
       2         2801.25880  47305.30500
       3         4234.92700  46661.44240
       4         4561.18850  36580.28216
       5         4687.79700  19023.26000
male   0         1121.87390  62592.87309
       1         1711.02680  51194.55914
       2         2304.00220  49577.66240
       3         3443.06400  60021.39897
       4         4504.66240  40182.24600
       5         4915.05985  14478.33015
```

```python
# ¿Cuál es el índice de masa corporal promedio para hombre y mujeres dependiendo
# región en la que viven y si son fumadores? ¿Impacta eso en la tarifa del
# seguro?

imc_and_charges_by_region_sex_smoker = df.groupby(['region', 'sex', 'smoker']).agg({'bmi': 'mean', 'charges': 'mean'})

print(imc_and_charges_by_region_sex_smoker)
```

```python
# ¿Cuál es el índice de masa corporal promedio para hombre y mujeres dependiendo
# región en la que viven y si son fumadores? ¿Impacta eso en la tarifa del
# seguro?

imc_and_charges_by_region_sex_smoker = df.groupby(['region', 'sex', 'smoker']).agg({'bmi': 'mean', 'charges': 'mean'})

print(imc_and_charges_by_region_sex_smoker)
```

```
                                bmi       charges
region    sex    smoker
northeast female no      29.777462   9640.426984
                 yes     27.261724  28032.046398
          male   no      28.861760   8664.042222
                 yes     29.560000  30926.252583
northwest female no      29.488704   8786.998679
                 yes     28.296897  29670.824946
          male   no      28.930379   8320.689321
                 yes     29.983966  30713.181419
southeast female no      32.780000   8440.205552
                 yes     32.251389  33034.820716
          male   no      34.129552   7609.003587
                 yes     33.650000  36029.839367
southwest female no      30.050355   8234.091260
                 yes     30.128571  31687.988430
          male   no      31.019841   7778.905534
                 yes     31.502703  32598.862854
```