

7 - Parte EDA & Limpieza - Sección NPCHP

December 14, 2022

1 Análisis de la Encuesta Multiproposito

1.1 Importación de Paquetes y carga del archivo

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

```
[2]: data = pd.read_excel('Encuesta_Multiproposito_Suba.xlsx')
```

```
[3]: data.shape
```

```
[3]: (24536, 498)
```

Hay en total 24536 encuestados en la Localidad de Suba

2 Septima Sección (NPCHP)

2.1 NPCHP2

2.1.1 2. ¿ ...actualmente estudia (asiste al preescolar, escuela, colegio, o universidad)?

Datos: 23515

```
[4]: data['NPCHP2'].count()
```

```
[4]: 23515
```

```
[5]: data['NPCHP2'].value_counts()
```

```
[5]: 0.0    18009
     1.0     5506
     Name: NPCHP2, dtype: int64
```

```
[6]: data = data.replace({'NPCHP2':2},0)
```

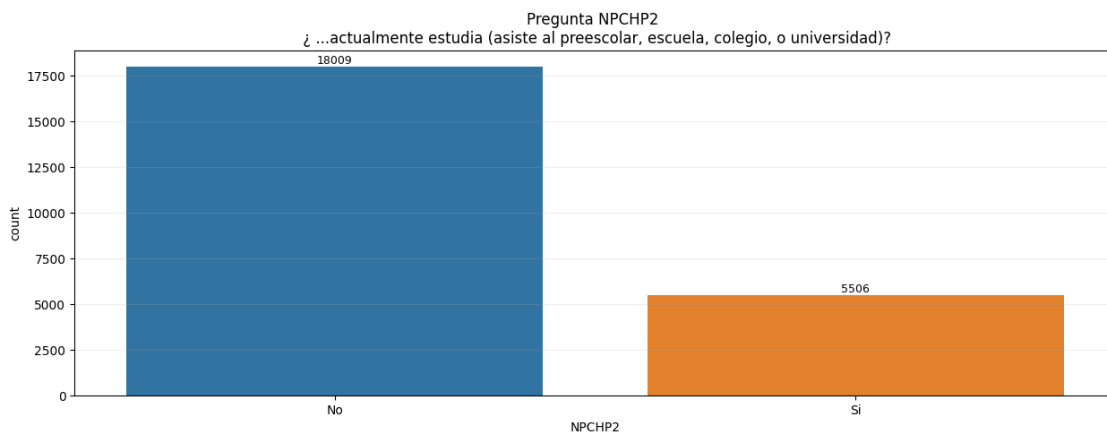
```
[7]: data['NPCHP2'] = data['NPCHP2'].replace([0, 1], ["No", "Si"])
```

```
[8]: fig, ax = plt.subplots(figsize=(15, 5))
g = sns.countplot(ax=ax, data = data, x = 'NPCHP2')

for bars in ax.containers:
    ax.bar_label(bars, fmt='%0f', fontsize=9)

ax.set_title('Pregunta NPCHP2 \n ¿ ...actualmente estudia (asiste al_
↳preescolar, escuela, colegio, o universidad)?')
#ax.set_xticklabels(['No', 'Si'])

plt.grid(alpha = 0.2, axis = 'y')
plt.show()
```



2.2 NPCHP3

2.3 3. ¿Cuál es la principal razón para que ...no estudie?

1. Considera que no está en edad de estudiar
2. Considera que ya terminó
3. Costos educativos elevados o falta de dinero
4. Labores de la finca donde vive (solo para rural)
5. Debe encargarse de las labores domésticas o del cuidado de niño/as y otras personas del hogar (adultos mayores, con discapacidad, etc.)
6. Necesita trabajar o buscar trabajo
7. No le gusta o no le interesa el estudio

8. Se casó o formó pareja
9. No existe centro educativo cercano
10. Otra razón

Datos: 4579

```
[8]: data['NPCHP3'].count()
```

```
[8]: 4579
```

```
[10]: data['NPCHP3'].value_counts()
```

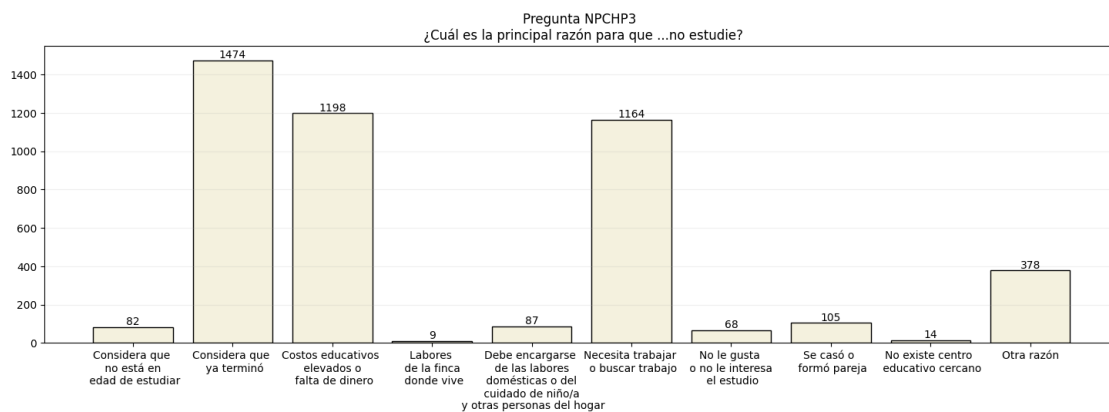
```
[10]: Considera que ya terminó
1474
Costos educativos elevados o falta de dinero
1198
Necesita trabajar o buscar trabajo
1164
Otra razón
378
Se casó o formó pareja
105
Debe encargarse de las labores domésticas o del cuidado de niño/a y otras
personas del hogar      87
Considera que no está en edad de estudiar
82
No le gusta o no le interesa el estudio
68
No existe centro educativo cercano
14
Labores de la finca donde vive
9
Name: NPCHP3, dtype: int64
```

```
[9]: data['NPCHP3'] = data['NPCHP3'].replace([1,2,3,4,5,6,7,8,9,10], ['Considera que
↪no está en edad de estudiar',
'Considera que ya terminó',
'Costos educativos elevados o falta de dinero',
'Labores de la finca donde vive',
'Debe encargarse de las labores domésticas o del cuidado de niño/a y otras
↪personas del hogar',
'Necesita trabajar o buscar trabajo',
'No le gusta o no le interesa el estudio',
'Se casó o formó pareja',
'No existe centro educativo cercano',
'Otra razón'])
```

```
[10]: data['NPCHP3'].describe().apply("{0:.2f}".format)
```

```
[10]: count      4579.00
      mean        4.22
      std         2.50
      min         1.00
      25%         2.00
      50%         3.00
      75%         6.00
      max         10.00
      Name: NPCHP3, dtype: object
```

```
[11]: plt.figure(figsize=(18,5))
      bars = plt.bar(data['NPCHP3'].value_counts().index.tolist(),data['NPCHP3'].
      ↪value_counts().tolist(), edgecolor = 'k', color = '#f4f1de')
      plt.xticks([1,2,3,4,5,6,7,8,9,10], ['Considera que \n no está en \n edad de_
      ↪estudiar',
      'Considera que \n ya terminó',
      'Costos educativos \n elevados o \n falta de dinero',
      'Labores \n de la finca \n donde vive',
      'Debe encargarse \n de las labores \n domésticas o del \n cuidado de niño/a \n_
      ↪y otras personas del hogar',
      'Necesita trabajar \n o buscar trabajo',
      'No le gusta \n o no le interesa \n el estudio',
      'Se casó o \n formó pareja',
      'No existe centro \n educativo cercano',
      'Otra razón'])
      plt.title('Pregunta NPCHP3 \n ¿Cuál es la principal razón para que ...no_
      ↪estudie?')
      plt.bar_label(bars)
      plt.grid(alpha = 0.2, axis = 'y')
      plt.show()
```



2.4 NPCHP4

2.4.1 4. ¿Cuál es el nivel educativo más alto alcanzado por ... y el último año o grado aprobado en este nivel?

1. Ninguno
2. Preescolar
3. Básica primaria (1° - 5°)
4. Básica secundaria (6° - 9°)
5. Media (10° - 13°)
6. Técnico
7. Tecnológico
8. Universitaria incompleta (sin título)
9. Universitaria completa (con título)
10. Especialización incompleta (sin título)
11. Especialización completa (con título)
12. Maestría incompleta (sin título)
13. Maestría completa (con título)
14. Doctorado incompleto (sin título)
15. Doctorado completo (con título) ”

Datos: 18009

```
[12]: data['NPCHP4'].count()
```

```
[12]: 18009
```

```
[13]: data['NPCHP4'].value_counts()
```

```
[13]: 9.0      5322
      5.0      3506
      11.0     2313
      3.0      1560
      6.0      1554
      4.0      1021
      13.0      932
      7.0       745
      8.0       588
      1.0       211
      15.0        93
      10.0        69
      12.0        52
      2.0         31
      14.0         12
```

```
Name: NPCHP4, dtype: int64
```

```
[11]: data['NPCHP4'] = data['NPCHP4'].replace([1,2,3,4,5,6,7,8,9,10,11,12,13,14,15],
      ['Ninguno',
      'Preescolar',
```

```

'Básica primaria (1° - 5°)',
'Básica secundaria (6° - 9°)',
'Media (10° - 13°)',
'Técnico',
'Tecnológico',
'Universitaria incompleta (sin título)',
'Universitaria completa (con título)',
'Especialización incompleta (sin título)',
'Especialización completa (con título)',
'Maestría incompleta (sin título)',
'Maestría completa (con título)',
'Doctorado incompleto (sin título)',
'Doctorado completo (con título)']])

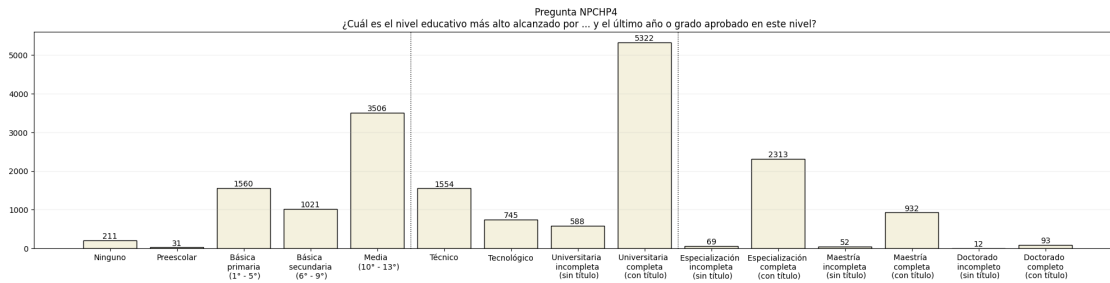
```

[]:

```

[15]: plt.figure(figsize=(25,5))
bars = plt.bar(data['NPCHP4'].value_counts().index.tolist(),data['NPCHP4'].
    ↳value_counts().tolist(), edgecolor = 'k', color = '#f4f1de')
plt.xticks([1,2,3,4,5,6,7,8,9,10,11,12,13,14,15],
    ['Ninguno',
    'Preescolar',
    'Básica \n primaria \n (1° - 5°)',
    'Básica \n secundaria \n (6° - 9°)',
    'Media \n (10° - 13°)',
    'Técnico',
    'Tecnológico',
    'Universitaria \n incompleta \n (sin título)',
    'Universitaria \n completa \n (con título)',
    'Especialización \n incompleta \n (sin título)',
    'Especialización \n completa \n (con título)',
    'Maestría \n incompleta \n (sin título)',
    'Maestría \n completa \n (con título)',
    'Doctorado \n incompleto \n (sin título)',
    'Doctorado \n completo \n (con título)'])
plt.title('Pregunta NPCHP4 \n ¿Cuál es el nivel educativo más alto alcanzado_
    ↳por ... y el último año o grado aprobado en este nivel?')
plt.axvline(5.5, color='k', linestyle=':', linewidth=1)
plt.axvline(9.5, color='k', linestyle=':', linewidth=1)
plt.bar_label(bars)
plt.grid(alpha = 0.2, axis = 'y')
plt.show()

```



2.5 NPCHP4A

2.5.1 4a. Último año o grado aprobado

Datos: 17798

```
[16]: data['NPCHP4A'].count()
```

```
[16]: 17798
```

```
[17]: data['NPCHP4A'].describe().apply("{0:.2f}".format)
```

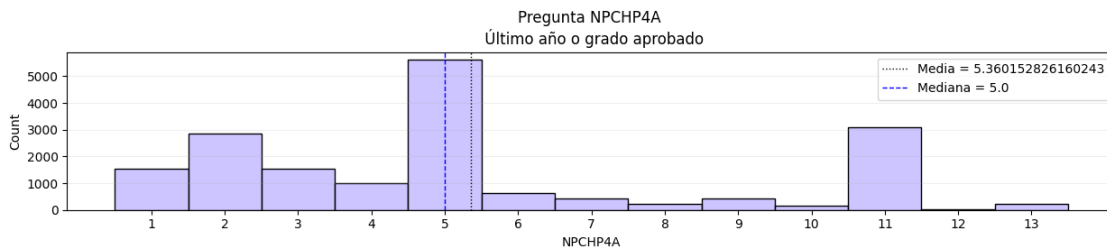
```
[17]: count      17798.00
      mean         5.36
      std          3.33
      min          1.00
      25%          3.00
      50%          5.00
      75%          7.00
      max          13.00
      Name: NPCHP4A, dtype: object
```

```
[18]: data['NPCHP4A'].value_counts()
```

```
[18]: 5.0      5615
      11.0     3101
      2.0     2868
      3.0     1553
      1.0     1541
      4.0      990
      6.0      630
      7.0      430
      9.0      422
      8.0      243
      13.0     222
      10.0     148
      12.0      35
```

Name: NPCHP4A, dtype: int64

```
[19]: plt.figure(figsize=(15,5))
plt.subplot(2,1,1)
#counts, edges, bars = plt.hist(data['NPCHP4A'], bins = np.arange(1,15)-0.5,
    ↪edgecolor = 'black', color = '#bdb2ff')
sns.histplot(data = data, x = 'NPCHP4A', bins = np.arange(1,15)-0.5, color =
    ↪'#bdb2ff')
#sns.kdeplot(data = data, x = 'NPCGP17B')
#ticklabels = [i for i in range(5)]
plt.xticks(range(1,14))
#plt.bar_label(bars)
plt.title('Pregunta NPCHP4A \n Último año o grado aprobado')
#plt.xlim([-1,10])
plt.axvline(data['NPCHP4A'].mean(), color='k', linestyle=':', linewidth=1,
    ↪label = f'Media = {data["NPCHP4A"].mean()}')
plt.axvline(data['NPCHP4A'].median(), color='b', linestyle='dashed',
    ↪linewidth=1, label = f'Mediana = {data["NPCHP4A"].median()}')
plt.legend()
plt.grid(alpha = 0.2, axis = 'y')
plt.show()
```



2.6 NPCHP7

2.6.1 7. ¿Cuántos años de estudios superiores (técnicos, tecnológicos, universitarios, de posgrado, etc.) ha realizado y aprobado?

Datos: 2153

```
[20]: data['NPCHP7'].count()
```

[20]: 2153

```
[21]: data['NPCHP7'].describe().apply("{0:.2f}".format)
```

```
[21]: count    2153.00
      mean      3.87
```



```

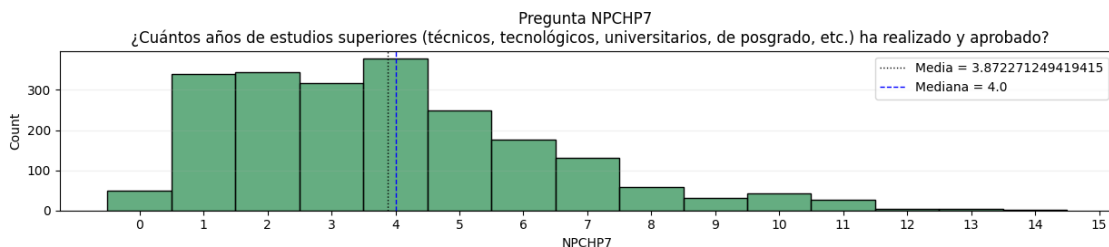
std          2.48
min          0.00
25%         2.00
50%         4.00
75%         5.00
max          15.00
Name: NPCHP7, dtype: object

```

```

[22]: plt.figure(figsize=(15,5))
plt.subplot(2,1,1)
#counts, edges, bars = plt.hist(data['NPCGP17B'] , bins = 50, edgecolor = 'black', color = '#bdb2ff')
sns.histplot(data = data, x = 'NPCHP7', bins = np.arange(16)-0.5, color = '#329257')
#sns.kdeplot(data = data, x = 'NPCGP17B')
#ticklabels = [i for i in range(5)]
plt.xticks(range(16))
#plt.bar_label(bars)
plt.title('Pregunta NPCHP7 \n ¿Cuántos años de estudios superiores (técnicos, tecnológicos, universitarios, de posgrado, etc.) ha realizado y aprobado?')
#plt.xlim([-1,10])
plt.axvline(data['NPCHP7'].mean(), color='k', linestyle=':', linewidth=1, label=f'Media = {data["NPCHP7"].mean()}')
plt.axvline(data['NPCHP7'].median(), color='b', linestyle='dashed', linewidth=1, label=f'Mediana = {data["NPCHP7"].median()}')
plt.legend()
plt.grid(alpha = 0.2, axis = 'y')
plt.show()

```



2.7 NPCHP18

2.7.1 17. En un trayecto normal a su sitio de estudio, ¿cuáles medios de transporte utiliza ...:

Datos: 5506

```
[13]: transporte = [i for i in data.columns if 'NPCHP18' in str(i)]  
print(transporte)
```

```
['NPCHP18AA', 'NPCHP18AB', 'NPCHP18AC', 'NPCHP18AD', 'NPCHP18AE', 'NPCHP18AF',  
'NPCHP18AG', 'NPCHP18AH', 'NPCHP18AI', 'NPCHP18AI1', 'NPCHP18AJ', 'NPCHP18AK',  
'NPCHP18AL', 'NPCHP18AM', 'NPCHP18AN', 'NPCHP18AO']
```

```
[14]: for i in transporte:  
print(i, data[i].count())
```

```
NPCHP18AA 5506  
NPCHP18AB 5506  
NPCHP18AC 5506  
NPCHP18AD 5506  
NPCHP18AE 5506  
NPCHP18AF 5506  
NPCHP18AG 5506  
NPCHP18AH 5506  
NPCHP18AI 5506  
NPCHP18AI1 5506  
NPCHP18AJ 5506  
NPCHP18AK 5506  
NPCHP18AL 5506  
NPCHP18AM 313  
NPCHP18AN 5506  
NPCHP18AO 5506
```

```
[15]: for i in transporte:  
data = data.replace({i:2},0)
```

```
[16]: for i in transporte:  
data[i] = data[i].replace([0,1], ['No', 'Si'])
```

```
[17]: list_tra = ['Transmilenio',  
                 'Buses del Sitp',  
                 'Bus buseta o colectivo',  
                 'Automovil de uso particular',  
                 'Taxi',  
                 'Motocicleta',  
                 'Bicicleta',  
                 'Ruta Escolar',  
                 'A pie',  
                 'Bus Intermunicipal',  
                 'Bicitaxi o mototaxi',  
                 'Aplicaciones',  
                 'Vehículo, patineta, bicicleta o moto eléctricos',  
                 'Caballo',  
                 'Otro','No se desplaza']
```

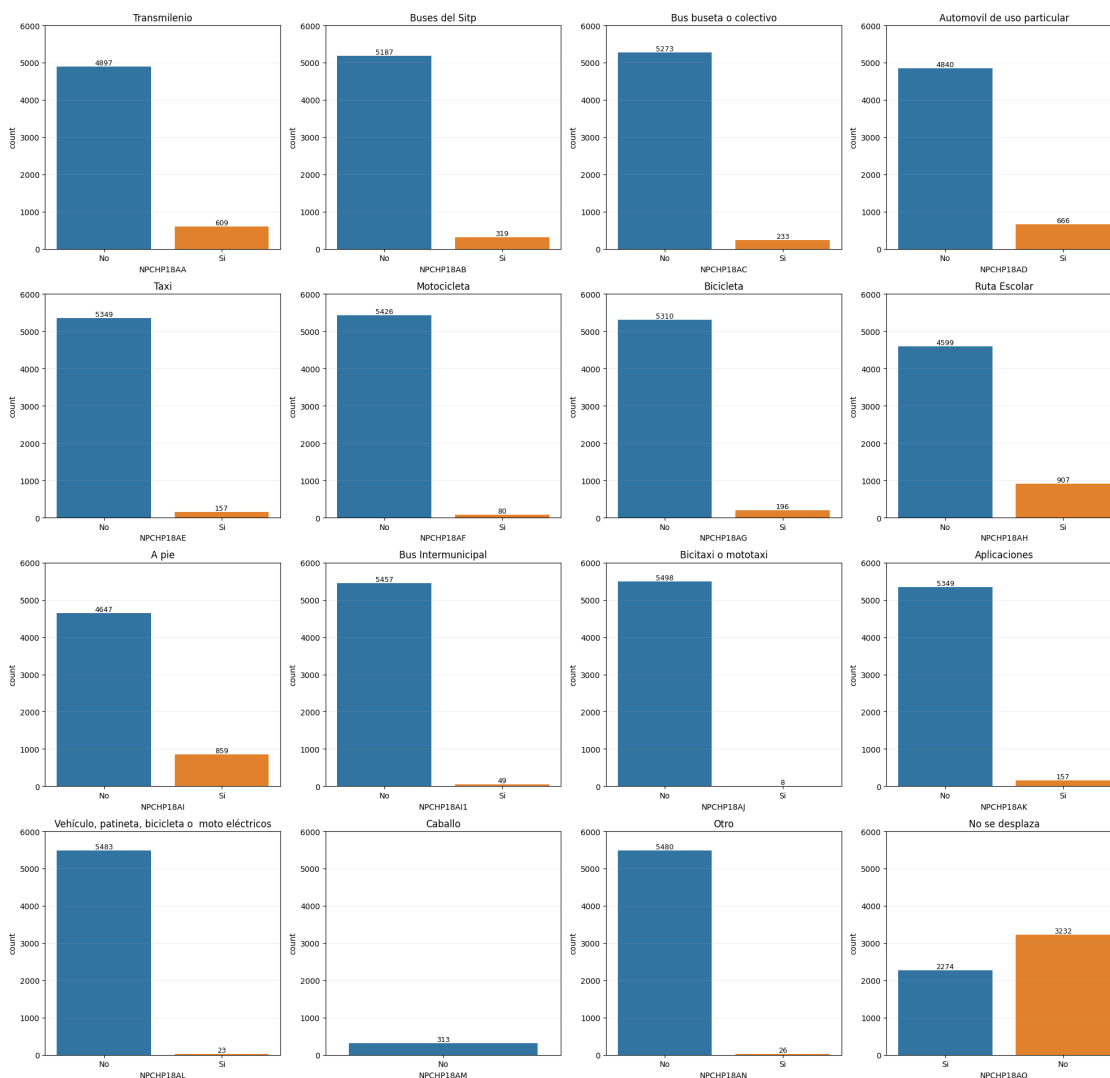
```

[18]: fig, axes = plt.subplots(4,4, figsize = (20,20), squeeze=False)
axli = axes.flatten()
fig.subplots_adjust(top=0.9)
fig.suptitle('En un trayecto normal a su sitio de estudio, ¿cuáles medios de
↳ transporte utiliza ...:', fontsize=24)
for ax,cols,names in zip(axli,transporte,list_tra):
    sns.countplot(x = cols, data = data, ax = ax)
    ax.grid(alpha = 0.2, axis = 'y')
    ax.set_title(f'{names}')
    #ax.set_xticks([0,1],['No','Si'])
    ax.set_ylim(0,6000)
    ax.margins(y=0.1) # make room for the labels
    for bars in ax.containers:
        ax.bar_label(bars, fmt='%.0f', fontsize=9)
plt.tight_layout()
plt.subplots_adjust(top=0.94)

plt.show()

```

En un trayecto normal a su sitio de estudio, ¿cuáles medios de transporte utiliza ...:



2.8 NPCHP19

2.8.1 18. ¿Cuánto tiempo se demora ... en su viaje de ida a su sitio de estudio (incluya tiempo de espera del medio de transporte y el tiempo en todos los medios utilizados)? Minutos

Datos: 3232

```
[28]: data['NPCHP19'].count()
```

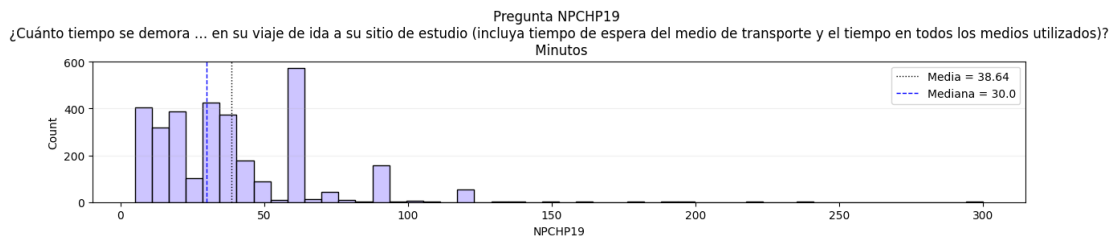
```
[28]: 3232
```

```
[29]: data['NPCHP19'].describe().apply("{0:.2f}".format)
```

```
[29]: count    3232.00
      mean      60.03
      std      144.21
      min       5.00
      25%      20.00
      50%      30.00
      75%      60.00
      max     999.00
      Name: NPCHP19, dtype: object
```

```
[19]: data['NPCHP19'] = data['NPCHP19'].replace(999,stats.trim_mean(data['NPCHP19'],0.
      ↪1))
```

```
[20]: plt.figure(figsize=(15,5))
      plt.subplot(2,1,1)
      #counts, edges, bars = plt.hist(data['NPCHP19'], bins = np.arange(1,15)-0.5,
      ↪edgecolor = 'black', color = '#bdb2ff')
      sns.histplot(data = data, x = 'NPCHP19', bins = 50, color = '#bdb2ff')
      #sns.kdeplot(data = data, x = 'NPCHP19')
      #ticklabels = [i for i in range(5)]
      #plt.xticks(range(1,14))
      #plt.bar_label(bars)
      plt.title('Pregunta NPCHP19 \n ¿Cuánto tiempo se demora ... en su viaje de ida a
      ↪su sitio de estudio (incluya tiempo de espera del medio de transporte y el
      ↪tiempo en todos los medios utilizados)? \n Minutos')
      #plt.xlim([-1,10])
      plt.axvline(data['NPCHP19'].mean(), color='k', linestyle=':', linewidth=1,
      ↪label = f'Media = {data["NPCHP19"].mean():.2f}')
      plt.axvline(data['NPCHP19'].median(), color='b', linestyle='dashed',
      ↪linewidth=1, label = f'Mediana = {data["NPCHP19"].median():.2f}')
      plt.legend()
      plt.grid(alpha = 0.2, axis = 'y')
      plt.show()
```



2.9 NPCHP28

2.9.1 25. Durante este AÑO ESCOLAR, ¿ ...ha recibido crédito educativo?

Datos: 5506

```
[31]: data['NPCHP28'].count()
```

```
[31]: 5506
```

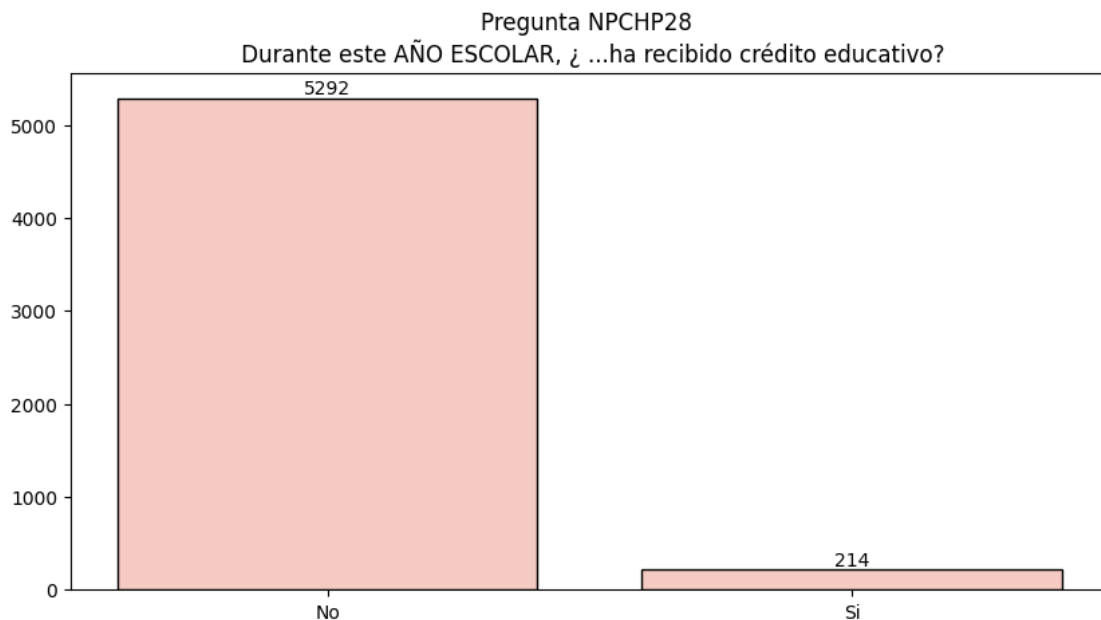
```
[32]: data['NPCHP28'].value_counts()
```

```
[32]: 2.0    5292  
     1.0     214  
     Name: NPCHP28, dtype: int64
```

```
[33]: data = data.replace({'NPCHP28':2},0)
```

```
[21]: data['NPCHP28'] = data['NPCHP28'].replace([0,1],["No", "Si"])
```

```
[34]: plt.figure(figsize=(10,5))  
     bars = plt.bar(data['NPCHP28'].value_counts().index.tolist(),data['NPCHP28'].  
           ↳value_counts().tolist(), edgecolor = 'black', color = '#f5cac3')  
     plt.xticks([0,1], ['No','Si'])  
     plt.title('Pregunta NPCHP28 \n Durante este AÑO ESCOLAR, ¿ ...ha recibido_↳  
           ↳crédito educativo?')  
     plt.bar_label(bars)  
     plt.show()
```



2.10 NPCHP28A

25a. Valor Datos: 214

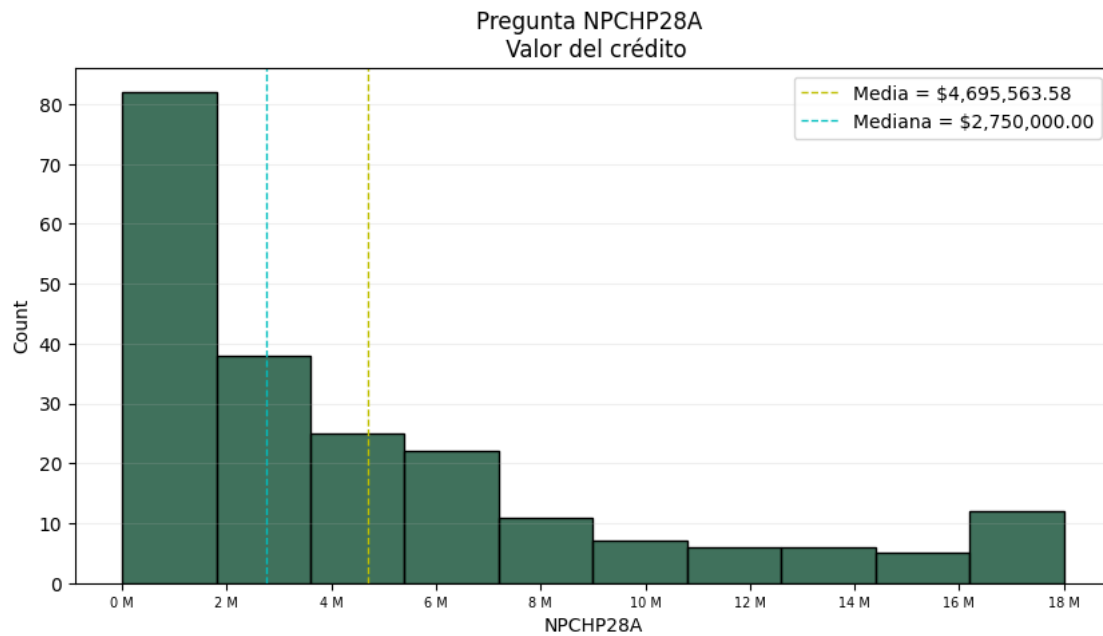
```
[35]: data['NPCHP28A'].count()
```

```
[35]: 214
```

```
[36]: data['NPCHP28A'].describe().apply("{0:.2f}".format)
```

```
[36]: count          214.00
      mean         4695563.58
      std          5015230.21
      min           99.00
      25%          750000.00
      50%          2750000.00
      75%          6500000.00
      max          18000000.00
      Name: NPCHP28A, dtype: object
```

```
[37]: plt.figure(figsize=(10,5))
      sns.histplot(data = data, x = 'NPCHP28A', color = '#004225', fill = True)
      plt.title('Pregunta NPCHP28A \n Valor del crédito')
      plt.grid(alpha = 0.2, axis = 'y')
      plt.axvline(data['NPCHP28A'].mean(), color='y', linestyle='dashed',
      ↪linewidth=1, label = f'Media = ${data["NPCHP28A"].mean():.2f}')
      plt.axvline(data['NPCHP28A'].median(), color='c', linestyle='dashed',
      ↪linewidth=1, label = f'Mediana = ${data["NPCHP28A"].median():.2f}')
      plt.xticks(range(0,20000000,2000000),[f'{(i / 1000000):.0f} M' for i in
      ↪range(0,20000000,2000000)], fontsize=7)
      plt.legend()
      plt.show()
```



2.11 NPCHP28B

2.11.1 25b. Frecuencia:

1. Mensual
2. Bimestral
3. Semestral
4. Anual

Datos: 214

```
[38]: data['NPCHP28B'].count()
```

```
[38]: 214
```

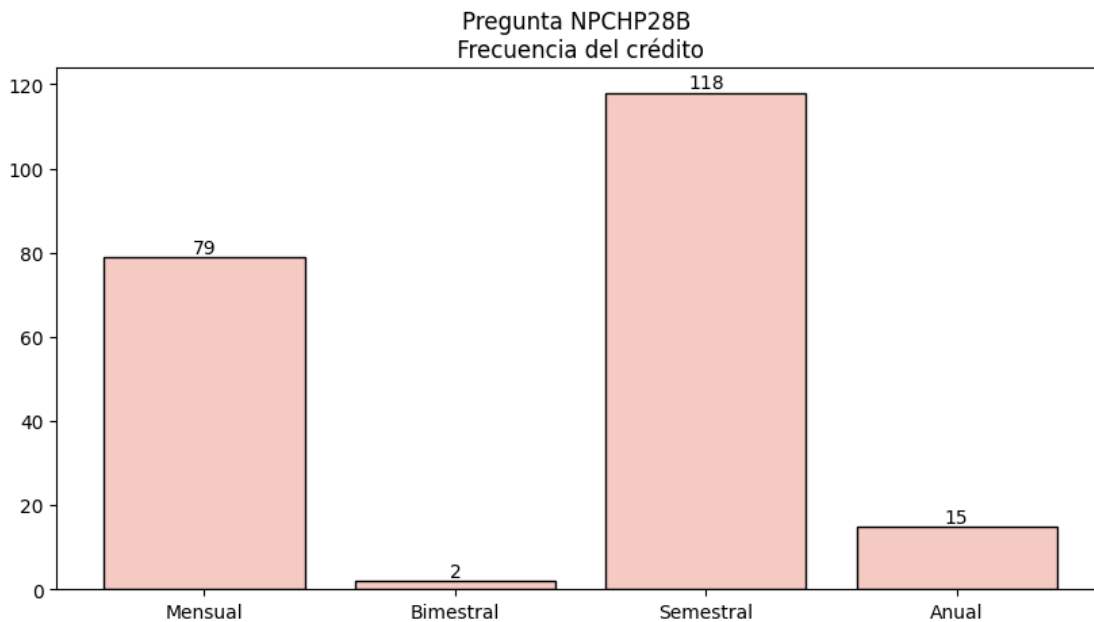
```
[39]: data['NPCHP28B'].value_counts()
```

```
[39]: 3.0    118
      1.0     79
      4.0     15
      2.0      2
      Name: NPCHP28B, dtype: int64
```

```
[22]: data['NPCHP28B'] = data['NPCHP28B'].replace([1,2,3,4],
      ↪ ['Mensual', 'Bimestral', 'Semestral', 'Anual'])
```



```
[40]: plt.figure(figsize=(10,5))
bars = plt.bar(data['NPCHP28B'].value_counts().index.tolist(),data['NPCHP28B'].
↪value_counts().tolist(), edgecolor = 'black', color = '#f5cac3')
plt.xticks([1,2,3,4], ['Mensual','Bimestral','Semestral', 'Anual'])
plt.title('Pregunta NPCHP28B \n Frecuencia del crédito ')
plt.bar_label(bars)
plt.show()
```



2.12 NPHCP29

2.12.1 26. De quién recibió el crédito educativo?

```
[23]: credito = [i for i in data.columns if 'NPCHP29' in str(i)]
print(credito)
```

```
['NPCHP29A', 'NPCHP29B', 'NPCHP29C', 'NPCHP29D', 'NPCHP29E', 'NPCHP29F',
'NPCHP29G', 'NPCHP29H', 'NPCHP29I']
```

```
[24]: for i in credito:
print(i, data[i].count())
```

```
NPCHP29A 61
NPCHP29B 104
NPCHP29C 26
NPCHP29D 1
NPCHP29E 2
NPCHP29F 3
NPCHP29G 7
```

NPCHP29H 3
NPCHP29I 11

```
[26]: for i in credito:  
      data = data.replace({i:2},0)
```

```
[27]: for i in credito:  
      data[i] = data[i].replace([0,1], ["No","Si"])
```

```
[28]: list_credito= ['Del mismo establecimiento educativo',  
                   'Del Icetex',  
                   'De un banco o corporación',  
                   'Del gobierno distrital o municipal',  
                   'De otra entidad pública',  
                   'De la empresa pública donde usted o un/a familiar trabajan',  
                   'De la empresa privada donde usted o un/a familiar trabajan',  
                   'De un/a familiar',  
                   'De otra entidad']
```

2.13 NPCHP32

2.13.1 29.¿Con quién permanece ...después de asistir al establecimiento educativo o durante la mayor parte del tiempo entre semana?

1. Una persona de este hogar
2. Una persona de otro hogar
3. Una institución
4. Permanece solo

Datos: 3522

```
[45]: data['NPCHP32'].count()
```

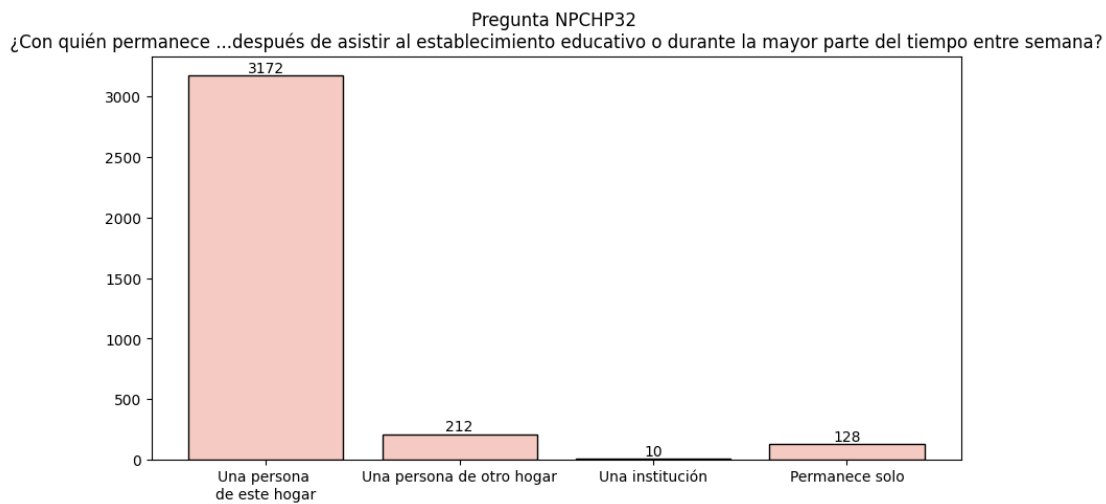
```
[45]: 3522
```

```
[46]: data['NPCHP32'].value_counts()
```

```
[46]: 1.0    3172  
      2.0    212  
      4.0    128  
      3.0     10  
      Name: NPCHP32, dtype: int64
```

```
[29]: data['NPCHP32'] = data['NPCHP32'].replace([1,2,3,4], ['Una persona de este_␣  
                  ↪hogar',  
                  'Una persona de otro hogar',  
                  'Una institución',  
                  'Permanece solo'])
```

```
[47]: plt.figure(figsize=(10,5))
bars = plt.bar(data['NPCHP32'].value_counts().index.tolist(),data['NPCHP32'].
    ↳value_counts().tolist(), edgecolor = 'black', color = '#f5cac3')
plt.xticks([1,2,3,4], ['Una persona \nde este hogar',
    'Una persona de otro hogar',
    'Una institución',
    'Permanece solo'])
plt.title('Pregunta NPCHP32 \n ¿Con quién permanece ...después de asistir al_
    ↳establecimiento educativo o durante la mayor parte del tiempo entre semana?_
    ↳')
plt.bar_label(bars)
plt.show()
```



2.13.2 NPCHP32B

2.13.3 29b. Una persona de otro hogar

1. Hombre
2. Mujer

Datos: 212

```
[48]: data['NPCHP32B'].count()
```

```
[48]: 212
```

```
[49]: data['NPCHP32B'].value_counts()
```

```
[49]: 2.0    201
      1.0    11
      Name: NPCHP32B, dtype: int64
```

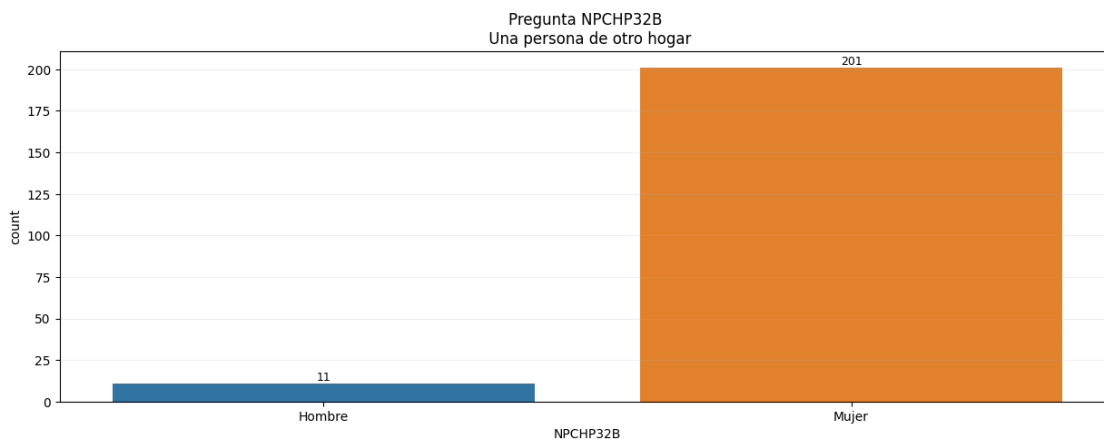
```
[30]: data['NPCHP32B'] = data['NPCHP32B'].replace([1,2],["Hombre","Mujer"])
```

```
[50]: fig, ax = plt.subplots(figsize=(15, 5))
g = sns.countplot(ax=ax, data = data, x = 'NPCHP32B')

for bars in ax.containers:
    ax.bar_label(bars, fmt='%.0f', fontsize=9)

ax.set_title('Pregunta NPCHP32B \n Una persona de otro hogar')
ax.set_xticklabels(['Hombre', 'Mujer'])

plt.grid(alpha = 0.2, axis = 'y')
plt.show()
```



2.14 NPCHP35

2.14.1 32. Durante el PRESENTE AÑO, ... ha sido discriminado/a, molestado/a, o le han hecho sentir mal por alguno de los siguientes motivos:

Datos: 19993

```
[31]: disc = [i for i in data.columns if 'NPCHP35' in str(i)]
print(disc)
```

```
['NPCHP35A', 'NPCHP35B', 'NPCHP35C', 'NPCHP35D', 'NPCHP35E', 'NPCHP35F',
'NPCHP35I', 'NPCHP35J', 'NPCHP35K']
```

```
[32]: for i in disc:
    print(i, data[i].count())
```

```
NPCHP35A 19993
NPCHP35B 19993
NPCHP35C 19993
```

```

NPCHP35D 19993
NPCHP35E 19993
NPCHP35F 19993
NPCHP35I 19993
NPCHP35J 19993
NPCHP35K 19993

```

```

[33]: for i in disc:
      data = data.replace({i:2},0)

```

```

[34]: for i in disc:
      data[i] = data[i].replace([0,1], ["No", "Si"])

```

```

[35]: list_disc = ['Por su raza u origen étnico',
                  'Por ser hombre o mujer',
                  'Por ser de los sectores LGBTI',
                  'Por sus creencias religiosas',
                  'Por su peso, su tamaño o apariencia física',
                  'Por sentirse identificado con algún grupo como metaleros, skinhead,
↪emos, entre otros',
                  'Por su condición económica y social',
                  'Por su origen campesino',
                  'Por ser extranjero']

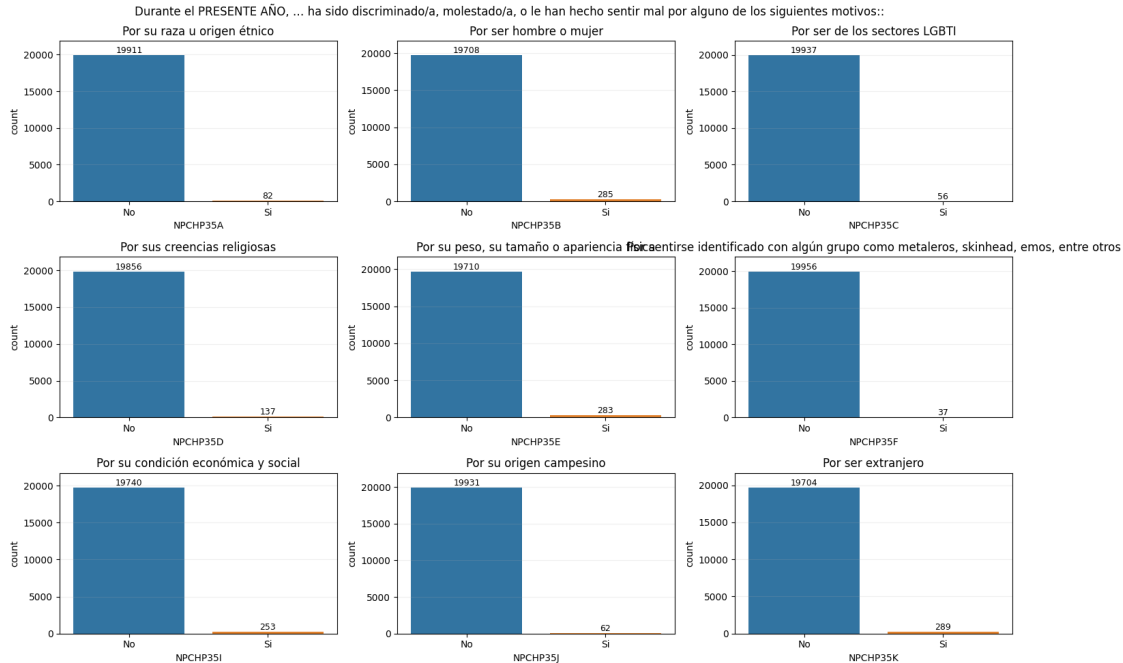
```

```

[36]: fig, axes = plt.subplots(3,3, figsize = (15,10), squeeze=False)
      axli = axes.flatten()
      fig.subplots_adjust(top=0.9)
      fig.suptitle('Durante el PRESENTE AÑO, ... ha sido discriminado/a, molestado/a, o
↪le han hecho sentir mal por alguno de los siguientes motivos:: ')
      for ax,cols,names in zip(axli,disc,list_disc):
          sns.countplot(x = cols, data = data, ax = ax)
          ax.grid(alpha = 0.2, axis = 'y')
          ax.set_title(f'{names}')
          #ax.set_xticks([0,1],['No', 'Si'])
          #ax.set_ylim(0,3200)
          ax.margins(y=0.1) # make room for the labels
          for bars in ax.containers:
              ax.bar_label(bars, fmt='%.0f', fontsize=9)
      plt.tight_layout()

      plt.show()

```



2.15 NPCHP36

2.15.1 33. ¿En los ÚLTIMOS 12 MESES ha tomado cursos virtuales (on-line) a través de internet (Open English, NextU, SENA virtual, entre otros)?

1. Sí
2. No

Datos: 23515

```
[56]: data['NPCHP36'].count()
```

```
[56]: 23515
```

```
[57]: data['NPCHP36'].value_counts()
```

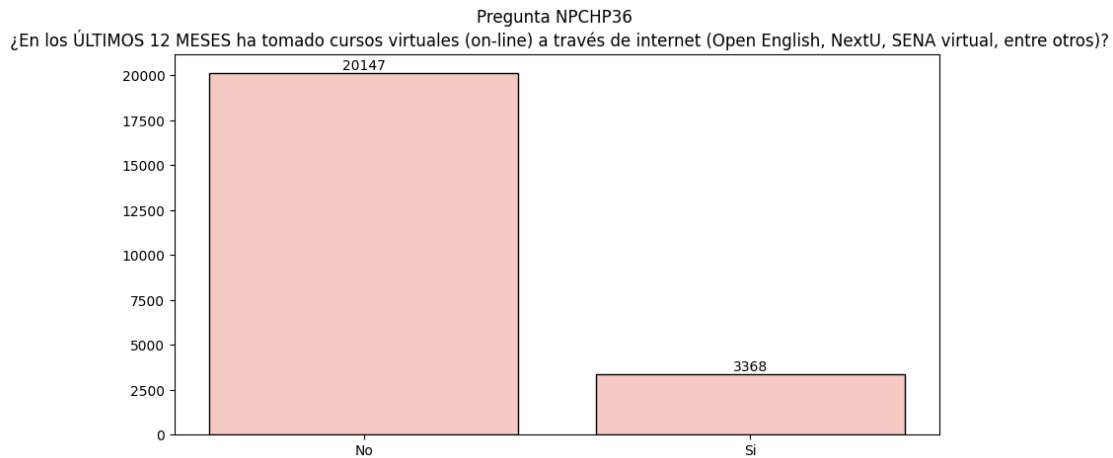
```
[57]: 2.0    20147
      1.0     3368
      Name: NPCHP36, dtype: int64
```

```
[37]: data['NPCHP36'] = data['NPCHP36'].replace([0,1], ["No","Si"])
```

```
[58]: data = data.replace({'NPCHP36':2},0)
```

```
[38]: plt.figure(figsize=(10,5))
      bars = plt.bar(data['NPCHP36'].value_counts().index.tolist(),data['NPCHP36'].
      ↪value_counts().tolist(), edgecolor = 'black', color = '#f5cac3')
```

```
#plt.xticks([0,1], ['No', 'Si'])
plt.title('Pregunta NPCHP36 \n ¿En los ÚLTIMOS 12 MESES ha tomado cursos_
↳virtuales (on-line) a través de internet (Open English, NextU, SENA virtual,
↳entre otros)?')
plt.bar_label(bars)
plt.show()
```



2.16 NPCHP37

2.16.1 34. En comparación a la formación que normalmente recibía antes de las medidas tomadas por efecto de la pandemia del coronavirus (COVID-19), ¿considera que la calidad de la educación en el año 2020?:

- 1. Mejoró
- 2. Se mantuvo igual
- 3. Empeoró
- 9. No sabe

Datos: 2972

```
[61]: data['NPCHP37'].count()
```

```
[61]: 2972
```

```
[62]: data['NPCHP37'].value_counts()
```

```
[62]: 3.0    1427
      2.0    1168
      1.0     269
      9.0     108
      Name: NPCHP37, dtype: int64
```

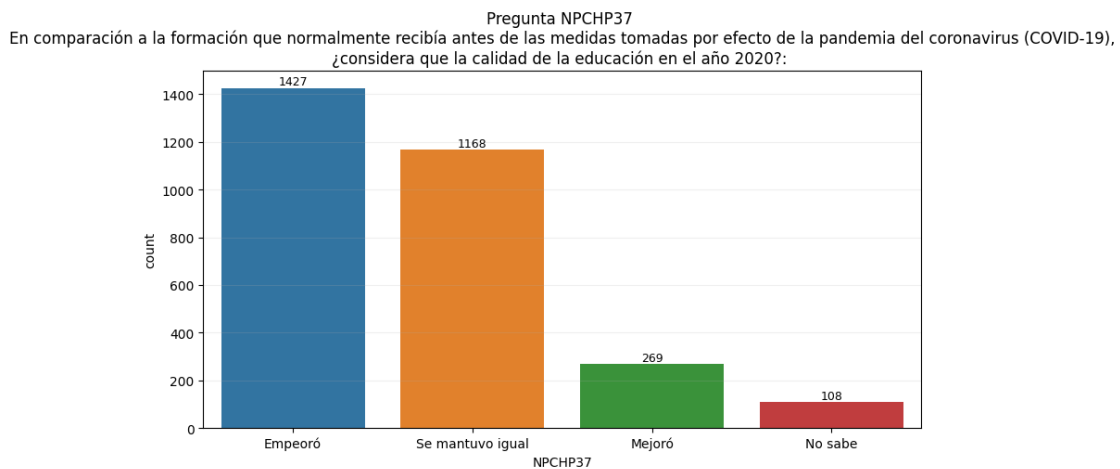
```
[41]: data['NPCHP37'] = data['NPCHP37'].replace([1,2,3,9], ['Mejoró','Se mantuvo igual', 'Empeoró', 'No sabe'])
```

```
[48]: fig, ax = plt.subplots(figsize=(10, 5))
g = sns.countplot(ax=ax, data = data, x = 'NPCHP37')

for bars in ax.containers:
    ax.bar_label(bars, fmt='%.0f', fontsize=9)

ax.set_title('Pregunta NPCHP37
En comparación a la formación que normalmente recibía antes de las medidas
tomadas por efecto de la pandemia del coronavirus (COVID-19),
¿considera que la calidad de la educación en el año 2020?: ')
#ax.set_xticklabels(['Mejoró', 'Se mantuvo igual', 'Empeoró', 'No sabe'])

plt.grid(alpha = 0.2, axis = 'y')
plt.show()
```



```
[45]: data.to_excel('Encuesta_Multiproposito_Suba.xlsx', index = False)
```

```
[ ]:
```