

5 - Parte EDA & Limpieza - Sección NPCFP

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 Quinta Sección (NPCFP)

2.1 NPCFP1

2.1.1 1. ¿... está afiliado/a (cotizante o beneficiario/a) a alguna entidad de seguridad social en salud?

(Entidad Promotora de Salud (EPS), Entidad Promotora de Salud del régimen Subsidiado (EPS-S), entidades de regímenes de excepción o especial)

- 0. No
- 1. Sí
- 9. No sabe, no informa

Datos: 24536

```
[4]: data['NPCFP1'].count()
```

```
[4]: 24536
```

```
[5]: data['NPCFP1'].describe().apply("{0:.2f}".format)
```

```
[5]: count    24536.00
      mean      1.01
      std       0.71
      min       0.00
      25%       1.00
      50%       1.00
      75%       1.00
      max       9.00
      Name: NPCFP1, dtype: object
```

```
[6]: data = data.replace({'NPCFP1':{2:0}})
```

```
[7]: data['NPCFP1'].value_counts()
```

```
[7]: 1    23271
      0     1087
      9      178
      Name: NPCFP1, dtype: int64
```

```
[8]: data['NPCFP1'] = data['NPCFP1'].replace([0,1,9],["No","Si","No sabe, no
      ↪informa"])
```

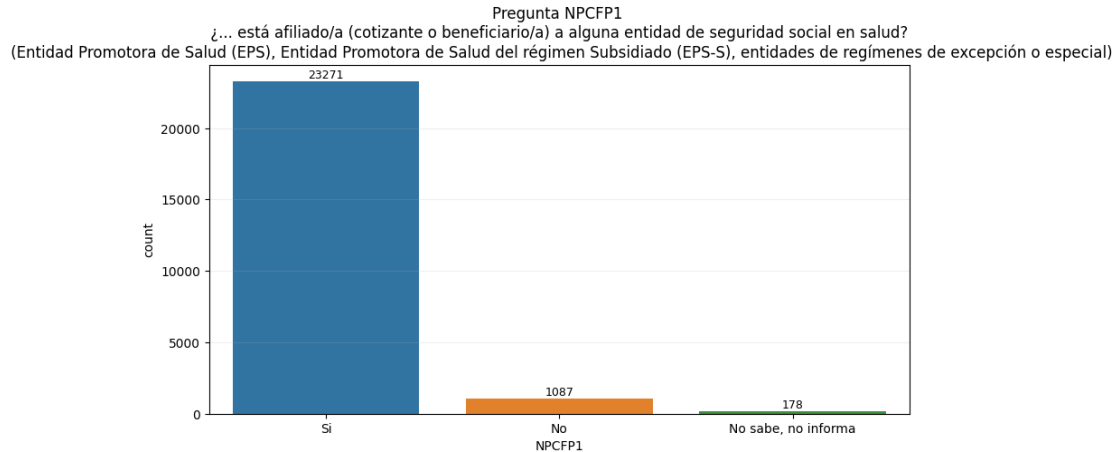
```
[9]: fig, ax = plt.subplots(figsize=(10, 5))

      g = sns.countplot(ax=ax, data = data, x ='NPCFP1')

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

      ax.set_title('Pregunta NPCFP1 \n ¿... está afiliado/a (cotizante o beneficiario/
      ↪a) a alguna entidad de seguridad social en salud? \n (Entidad Promotora de
      ↪Salud (EPS), Entidad Promotora de Salud del régimen Subsidiado (EPS-S),
      ↪entidades de regímenes de excepción o especial)')
      #ax.set_xticklabels(['No', 'Si', 'No sabe, no responde'])

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



2.2 NPCFP3

2.2.1 3. ¿Por qué razón principal no está afiliado/a?

1. Falta de dinero
2. Muchos trámites
3. No le interesa o descuido
4. No sabe que debe afiliarse o cómo afiliarse
5. No está vinculado/a laboralmente a una empresa o entidad
6. No tiene el puntaje requerido para acceder al régimen subsidiado
7. Por ser extranjero/a
8. Está en trámite de afiliación
9. Otra razón

Datos: 1087

```
[10]: data['NPCFP3'].count()
```

```
[10]: 1087
```

```
[11]: data['NPCFP3'].describe().apply("{0:.2f}".format)
```

```
[11]: count      1087.00
      mean        4.66
      std         2.66
      min         1.00
      25%         1.00
      50%         5.00
      75%         7.00
      max         9.00
      Name: NPCFP3, dtype: object
```

```
[19]: data['NPCFP3'].value_counts()
```

```
[19]: Falta de dinero                                309
      Por ser extranjero/a                            308
      No está vinculado/a laboralmente a una empresa o entidad  234
      Está en tramite de afiliación                    81
      No tiene el puntaje requerido para acceder al régimen subsidiado  44
      No le interesa o descuido                        37
      Otra razón                                       31
      Muchos trámites                                23
      No sabe que debe afiliarse o cómo afiliarse      20
      Name: NPCFP3, dtype: int64
```

```
[13]: data['NPCFP3'] = data['NPCFP3'].replace([1,2,3,4,5,6,7,8,9],['Falta de_
      ↪dinero','Muchos trámites','No le interesa o descuido',
      'No sabe que debe afiliarse o cómo afiliarse',
      'No está vinculado/a laboralmente a una empresa o entidad',
      'No tiene el puntaje requerido para acceder al régimen_
      ↪subsidiado',
      'Por ser extranjero/a','Está en tramite de_
      ↪afiliación','Otra razón'])
```

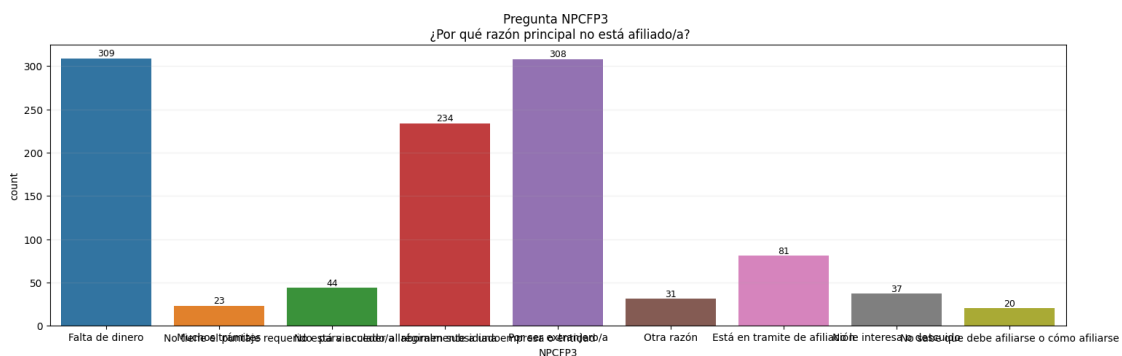
```
[27]: fig, ax = plt.subplots(figsize=(18, 5))

      g = sns.countplot(ax=ax, data = data, x = 'NPCFP3')

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

      ax.set_title('Pregunta NPCFP3 \n ¿Por qué razón principal no está afiliado/a?')
      #ax.set_xticklabels(fontsize=7)

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



2.3 NPCFP7

2.3.1 6. ¿... en los ÚLTIMOS 12 MESES ha participado en acciones informativas de cuidado personal, familiar o comunitario y del ambiente que promueven la salud, realizadas por la entidad de seguridad social en salud a la que está afiliado/a?

0. No

1. Si

Datos: 23126

```
[28]: data['NPCFP7'].count()
```

```
[28]: 23126
```

```
[29]: data['NPCFP7'].describe().apply("{0:.2f}".format)
```

```
[29]: count      23126.00
      mean        0.07
      std         0.26
      min         0.00
      25%         0.00
      50%         0.00
      75%         0.00
      max         1.00
      Name: NPCFP7, dtype: object
```

```
[30]: data = data.replace({'NPCFP7':2},0)
```

```
[31]: data['NPCFP7'].value_counts()
```

```
[31]: 0.0      21478
      1.0      1648
      Name: NPCFP7, dtype: int64
```

```
[32]: data['NPCFP7'] = data['NPCFP7'].replace([0,1],["No","Si"])
```

```
[33]: fig, ax = plt.subplots(figsize=(10, 5))

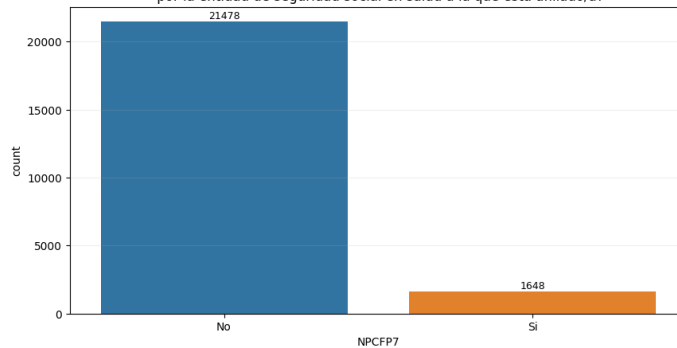
      g = sns.countplot(ax=ax, data = data, x ='NPCFP7')

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

      ax.set_title('Pregunta NPCFP7 \n ¿... en los ÚLTIMOS 12 MESES ha participado en
      ↳acciones informativas de cuidado personal, familiar o comunitario y del
      ↳ambiente que promueven la salud, realizadas \n por la entidad de seguridad
      ↳social en salud a la que está afiliado/a?')
      #ax.set_xticklabels(['No','Si'])
```

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

Pregunta NPCFP7
¿... en los ÚLTIMOS 12 MESES ha participado en acciones informativas de cuidado personal, familiar o comunitario y del ambiente que promueven la salud, realizadas por la entidad de seguridad social en salud a la que está afiliado/a?



2.4 NPCFP7{A1-A4}

2.4.1 6a. ¿... en los ÚLTIMOS 12 MESES ha participado en acciones preventivas realizadas por la entidad de seguridad social en salud a la que está afiliado/a, tales como:

Datos: 4252

```
[34]: salud_i = [i for i in data.columns if 'NPCFP7A' in str(i)]
```

```
[35]: salud_i
```

```
[35]: ['NPCFP7A1', 'NPCFP7A2', 'NPCFP7A3', 'NPCFP7A4']
```

```
[37]: for i in salud_i:
       print(i, data[i].count())
```

NPCFP7A1 4252

NPCFP7A2 4252

NPCFP7A3 4252

NPCFP7A4 4432

```
[38]: l_salud = ['Desparasitación', 'Vacunación', 'Enseñanza de cepillado y aplicación_
↳ de fluor o sellantes', 'Anticoncepción y prevención de enfermedades de_
↳ transmisión sexual']
```

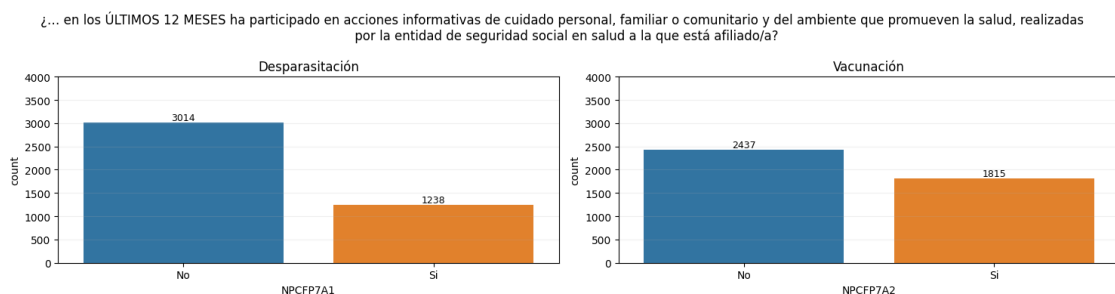
```
[22]: for i in salud_i:
       data = data.replace({i:2},0)
```

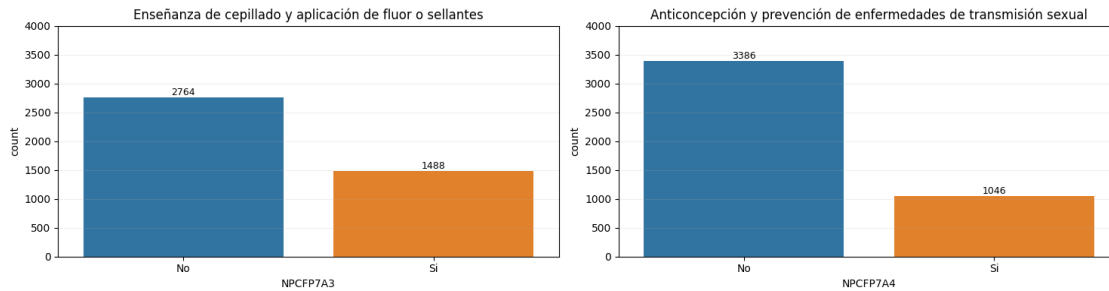
```
[39]: for i in salud_i:
      data[i] = data[i].replace([0,1],["No","Si"])
```

```
[40]: fig, axes = plt.subplots(1,2, figsize = (15,4))
fig.subplots_adjust(top=0.9)
fig.suptitle('¿... en los ÚLTIMOS 12 MESES ha participado en acciones_
↳informativas de cuidado personal, familiar o comunitario y del ambiente que_
↳promueven la salud, realizadas \n por la entidad de seguridad social en_
↳salud a la que está afiliado/a?')
for ax,cols,names in zip(axes,salud_i,l_salud):
    sns.countplot(x = cols, data = data, ax = ax)
    ax.grid(alpha = 0.2, axis = 'y')
    ax.set_title(f'{names}')
    #ax.set_xticklabels(['No','Si'])
    ax.set_ylim(0,4000)
    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()

fig, axes = plt.subplots(1,2, figsize = (15,4))
for ax,cols,names in zip(axes,salud_i[2:],l_salud[2:]):
    sns.countplot(x = cols, data = data, ax = ax)
    ax.grid(alpha = 0.2, axis = 'y')
    ax.set_title(f'{names}')
    #ax.set_xticklabels(['No','Si'])
    ax.set_ylim(0,4000)
    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()
```





2.5 NPCFP8

2.5.1 7. En general, considera que la calidad del servicio de la entidad de seguridad social en salud a la que ...está afiliado(a) es:

1. Muy buena
2. Buena
3. Regular
4. Mala
5. Muy mala
6. No sabe, no informa

Datos: 23126

```
[41]: data['NPCFP8'].count()
```

```
[41]: 23126
```

```
[42]: data['NPCFP8'].describe().apply("{0:.2f}".format)
```

```
[42]: count      23126.00
      mean         2.11
      std         1.13
      min         1.00
      25%         2.00
      50%         2.00
      75%         2.00
      max         9.00
      Name: NPCFP8, dtype: object
```

```
[43]: data['NPCFP8'].value_counts()
```

```
[43]: 2.0      16854
      1.0      3537
      3.0      1783
      9.0       447
      4.0       346
```



```
5.0      159
Name: NPCFP8, dtype: int64
```

```
[48]: data['NPCFP8'] = data['NPCFP8'].replace([1,2,3,4,5,9],['Muy
↪buena', 'Buena', 'Regular', 'Mala', 'Muy mala', 'No sabe, no informa'])
```

```
[49]: data['NPCFP8'].value_counts()
```

```
[49]: Buena      16854
Muy buena    3537
Regular      1783
No sabe, no informa  447
Mala         346
Muy mala     159
Name: NPCFP8, dtype: int64
```

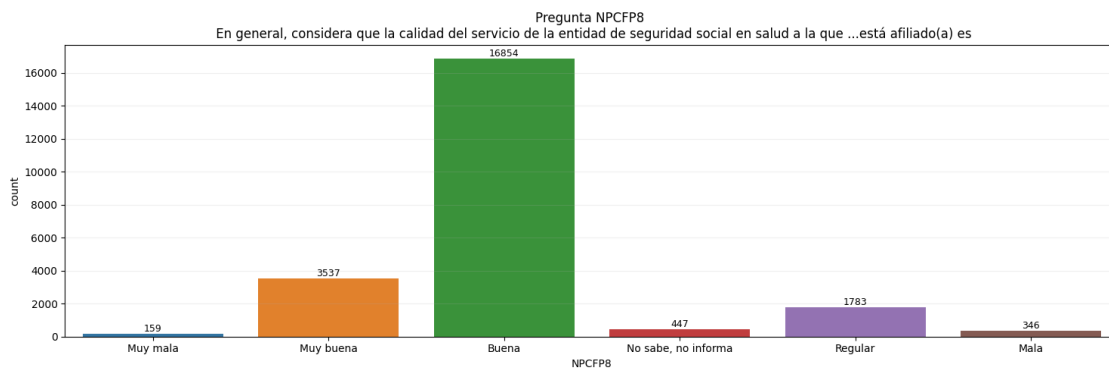
```
[50]: fig, ax = plt.subplots(figsize=(18, 5))

g = sns.countplot(ax=ax, data = data, x = 'NPCFP8')

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

ax.set_title('Pregunta NPCFP8 \n En general, considera que la calidad del
↪servicio de la entidad de seguridad social en salud a la que ...está
↪afiliado(a) es')
#ax.set_xticklabels(['Muy buena', 'Buena', 'Regular', 'Mala', 'Muy mala', 'No
↪sabe, \n no informa'])

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



2.6 NPCFP9

2.6.1 10. En promedio, ¿cuánto tiempo se demora en llegar ...desde su lugar de residencia a la IPS para sus consultas de medicina general? Minutos

Datos: 23126

Revisar caso de 1000

```
[28]: data['NPCFP9'].count()
```

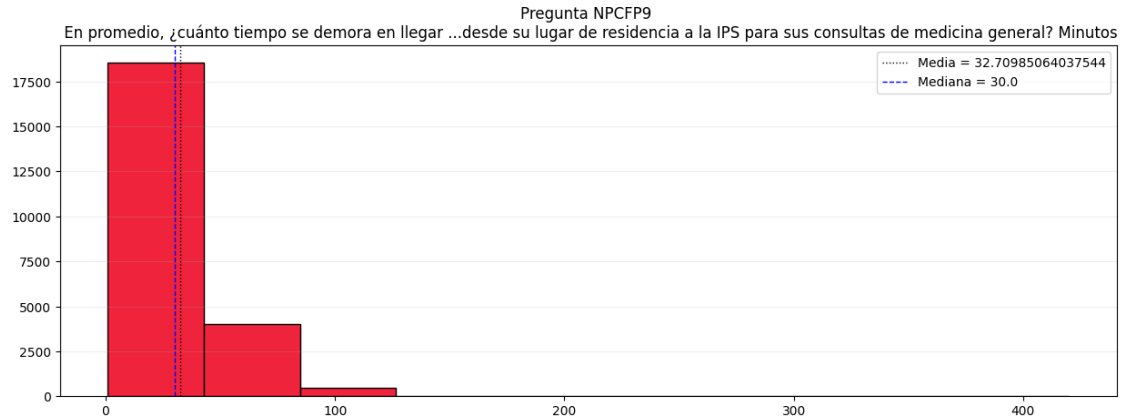
```
[28]: 23126
```

```
[53]: data['NPCFP9'].describe().apply("{0:.2f}".format)
```

```
[53]: count      23126.00
      mean       32.71
      std       18.71
      min        1.00
      25%       20.00
      50%       30.00
      75%       40.00
      max      420.00
      Name: NPCFP9, dtype: object
```

```
[51]: data['NPCFP9'] = data['NPCFP9'].replace(999,stats.trim_mean(data['NPCFP9'],0.1))
```

```
[58]: plt.figure(figsize=(15,5))
      plt.subplot(1,1,1)
      counts, edges, bars = plt.hist(data['NPCFP9'] , edgecolor = 'black', color = '#ef233c')
      #ticklabels = [i for i in range(5)]
      #plt.xticks(range(9))
      #plt.xticks(np.linspace(0, data['NPCEP8A'].max(),6),rotation = 45)
      #plt.bar_label(bars)
      plt.title('Pregunta NPCFP9 \n En promedio, ¿cuánto tiempo se demora en llegar ..
      ↪.desde su lugar de residencia a la IPS para sus consultas de medicina
      ↪general? Minutos')
      #plt.xlim([-1,10])
      plt.axvline(data['NPCFP9'].mean(), color='k', linestyle=':', linewidth=1, label=
      ↪ f'Media = {data["NPCFP9"].mean()}')
      plt.axvline(data['NPCFP9'].median(), color='b', linestyle='dashed',
      ↪linewidth=1, label = f'Mediana = {data["NPCFP9"].median()}')
      plt.legend()
      plt.grid(alpha = 0.2, axis = 'y')
      plt.show()
```



2.7 NPCFP13{A-F}

2.7.1 14. Sin estar enfermo/a y por prevención, ¿... por lo menos una vez al AÑO, consulta?

Datos: 24536

```
[54]: consulta = [i for i in data.columns if 'NPCFP13' in str(i)]
```

```
[55]: consulta
```

```
[55]: ['NPCFP13A', 'NPCFP13B', 'NPCFP13C', 'NPCFP13D', 'NPCFP13F']
```

```
[59]: for i in consulta:
        print(i, data[i].count())
```

```
NPCFP13A 24536
NPCFP13B 24536
NPCFP13C 24536
NPCFP13D 24536
NPCFP13F 24536
```

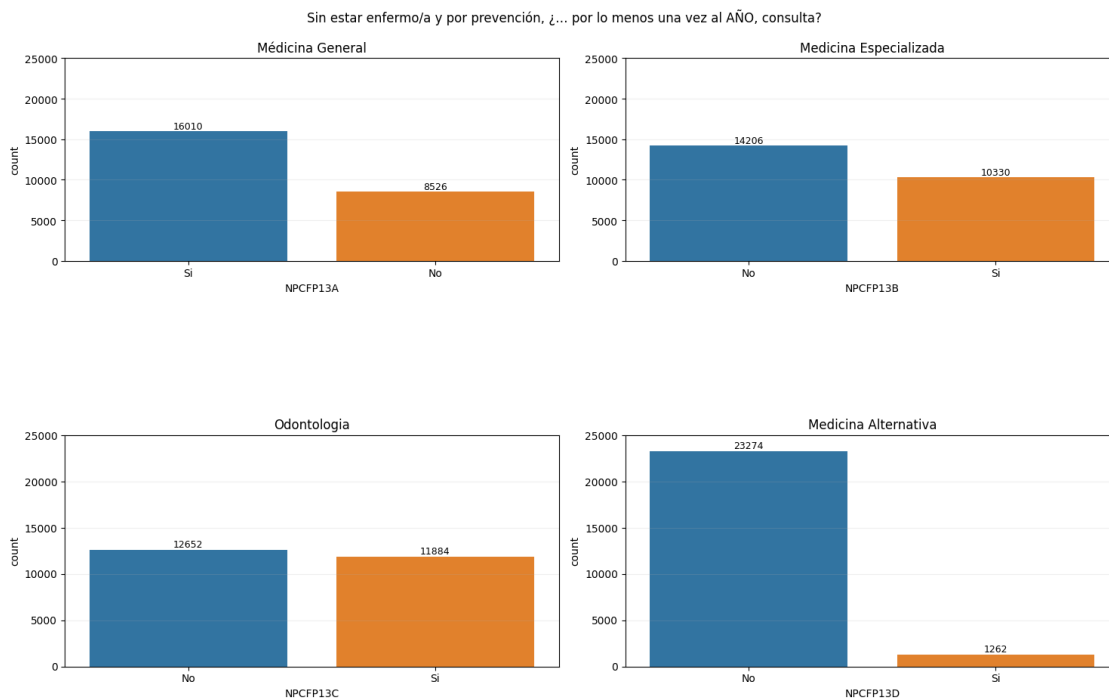
```
[60]: l_consulta = ['Medicina General',
                  'Medicina Especializada',
                  'Odontología',
                  'Medicina Alternativa',
                  'Psicología']
```

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

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

```
[63]: fig, axes = plt.subplots(1,2, figsize = (15,4))
fig.subplots_adjust(top=0.9)
fig.suptitle('Sin estar enfermo/a y por prevención, ¿... por lo menos una vez al_
↳AÑO, consulta?')
for ax,cols,names in zip(axes,consulta,l_consulta):
    sns.countplot(x = cols, data = data, ax = ax)
    ax.grid(alpha = 0.2, axis = 'y')
    ax.set_title(f'{names}')
    #ax.set_xticklabels(['No','Si'])
    ax.set_ylim(0,25000)
    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()

fig, axes = plt.subplots(1,2, figsize = (15,4))
for ax,cols,names in zip(axes,consulta[2:],l_consulta[2:]):
    sns.countplot(x = cols, data = data, ax = ax)
    ax.grid(alpha = 0.2, axis = 'y')
    ax.set_title(f'{names}')
    #ax.set_xticklabels(['No','Si'])
    ax.set_ylim(0,25000)
    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()
```



2.8 NPCFP14C

2.8.1 15. ¿A ...le han diagnosticado alguna de estas enfermedades o problemas de salud: Enfermedades mentales o del comportamiento (depresión, ansiedad, esquizofrenia, trastorno bipolar, alzheimer)?

Datos: 24536

```
[64]: data['NPCFP14C'].count()
```

```
[64]: 24536
```

```
[65]: data['NPCFP14C'].describe().apply("{0:.2f}".format)
```

```
[65]: count      24536.00
      mean        0.02
      std         0.14
      min         0.00
      25%         0.00
      50%         0.00
      75%         0.00
      max         1.00
      Name: NPCFP14C, dtype: object
```

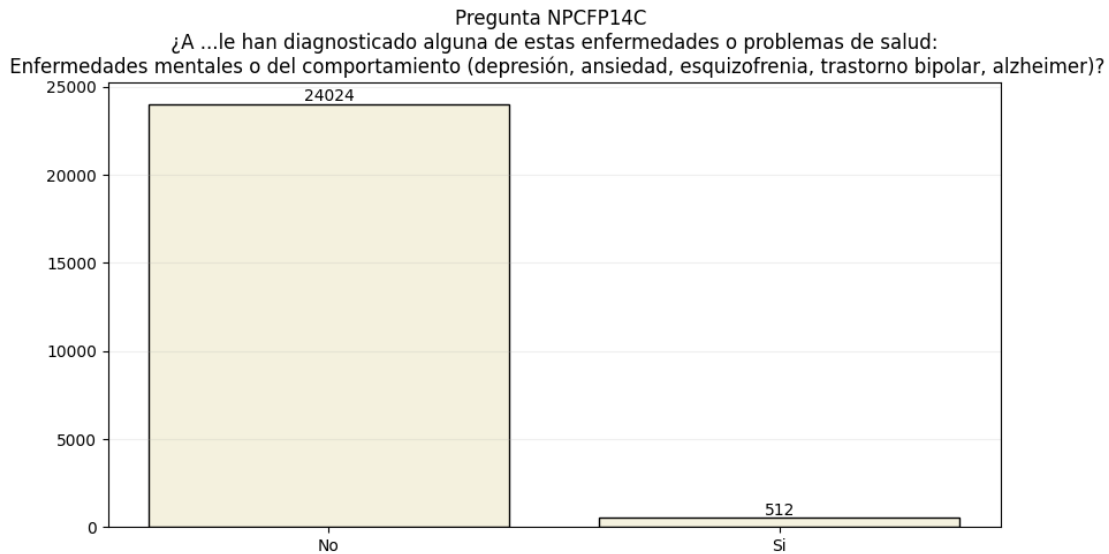
```
[66]: data = data.replace({'NPCFP14C':2},0)
```

```
[67]: data['NPCFP14C'].value_counts()
```

```
[67]: 0      24024
      1       512
      Name: NPCFP14C, dtype: int64
```

```
[68]: data['NPCFP14C'] = data['NPCFP14C'].replace([0,1],["No","Si"])
```

```
[69]: plt.figure(figsize=(10,5))
      bars = plt.bar(data['NPCFP14C'].value_counts().index.tolist(),data['NPCFP14C'].
      ↪value_counts().tolist(), edgecolor = 'k', color = '#f4f1de')
      #plt.xticks([0,1], ['No','Si'])
      plt.title('Pregunta NPCFP14C \n ¿A ...le han diagnosticado alguna de estas_
      ↪enfermedades o problemas de salud: \n Enfermedades mentales o del_
      ↪comportamiento (depresión, ansiedad, esquizofrenia, trastorno bipolar,_
      ↪alzheimer)?')
      plt.bar_label(bars)
      plt.grid(alpha = 0.2, axis = 'y')
      plt.show()
```



2.9 NPCFP24

2.9.1 23. ¿Quién se ocupa principalmente del cuidado de ...?

1. Una persona de este hogar, no remunerada
2. Una persona de otro hogar, no remunerada
3. Una persona contratada para ayudarlo
4. Una institución

Datos: 735

```
[70]: data['NPCFP24'].count()
```

```
[70]: 735
```

```
[71]: data['NPCFP24'].describe().apply("{0:.2f}".format)
```

```
[71]: count      735.00
      mean       1.38
      std        0.74
      min        1.00
      25%        1.00
      50%        1.00
      75%        1.00
      max        4.00
      Name: NPCFP24, dtype: object
```

```
[72]: data['NPCFP24'].value_counts()
```

```
[72]: 1.0    565
      3.0    97
      2.0    67
      4.0     6
      Name: NPCFP24, dtype: int64
```

```
[73]: data['NPCFP24'] = data['NPCFP24'].replace([1,2,3,4],['Una persona de este_
      ↪hogar, no remunerada','Una persona de otro hogar, no remunerada',
      ↪'Una persona contratada para ayudarle','Una institución'])
```

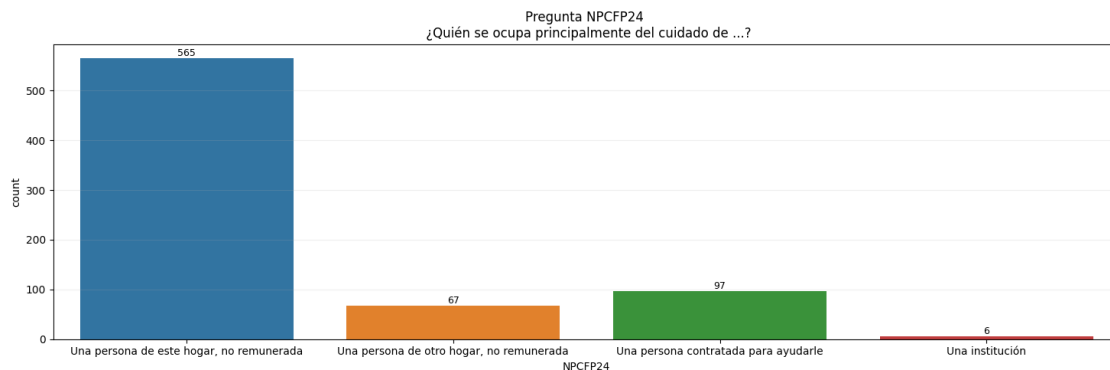
```
[76]: fig, ax = plt.subplots(figsize=(18, 5))

g = sns.countplot(ax=ax, data = data, x = 'NPCFP24')

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

ax.set_title('Pregunta NPCFP24 \n ¿Quién se ocupa principalmente del cuidado de_
      ↪...?')
#ax.set_xticklabels(['Una persona de \n este hogar, no remunerada','Una persona_
      ↪de \n otro hogar, no remunerada',
#                      'Una persona contratada \n para ayudarle','Una_
      ↪institución'])

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



2.10 NPCFP24A

2.10.1 23a. Una persona de este hogar, no remunerada. Núm. de orden

Datos: 565

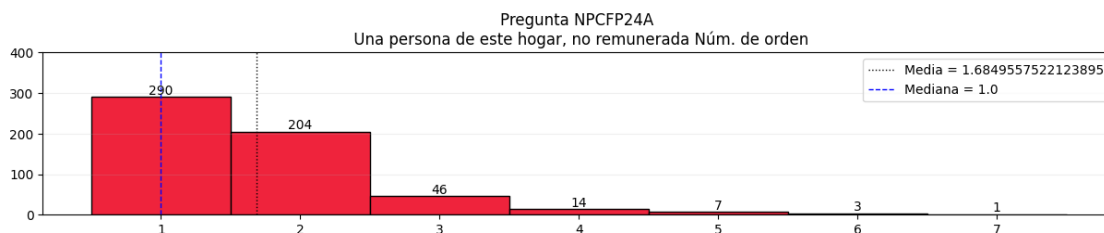
```
[77]: data['NPCFP24A'].count()
```

[77]: 565

```
[78]: data['NPCFP24A'].describe().apply("{0:.2f}".format)
```

```
[78]: count      565.00
      mean       1.68
      std       0.91
      min       1.00
      25%       1.00
      50%       1.00
      75%       2.00
      max       7.00
      Name: NPCFP24A, dtype: object
```

```
[79]: plt.figure(figsize=(15,5))
      plt.subplot(2,1,1)
      counts, edges, bars = plt.hist(data['NPCFP24A'] , bins = np.arange(1,9) - 0.5,
      ↪edgecolor = 'black', color = '#ef233c')
      #ticklabels = [i for i in range(5)]
      #plt.xticks(range(9))
      #plt.xticks(np.linspace(0, data['NPCEP8A'].max(),6),rotation = 45)
      plt.bar_label(bars)
      plt.title('Pregunta NPCFP24A \n Una persona de este hogar, no remunerada Núm. de orden')
      #plt.xlim([-1,10])
      plt.axvline(data['NPCFP24A'].mean(), color='k', linestyle=':', linewidth=1,
      ↪label = f'Media = {data["NPCFP24A"].mean()}')
      plt.axvline(data['NPCFP24A'].median(), color='b', linestyle='dashed',
      ↪linewidth=1, label = f'Mediana = {data["NPCFP24A"].median()}')
      plt.legend()
      plt.ylim((0,400))
      plt.grid(alpha = 0.2, axis = 'y')
      plt.show()
```



2.11 NPCFP24B

2.11.1 23b. Una persona de otro hogar no remunerada, o, una persona contratada, esta persona es:

1. Hombre
2. Mujer

Solo contestan 164

```
[80]: data['NPCFP24B'].count()
```

```
[80]: 164
```

```
[81]: data['NPCFP24B'].describe().apply("{0:.2f}".format)
```

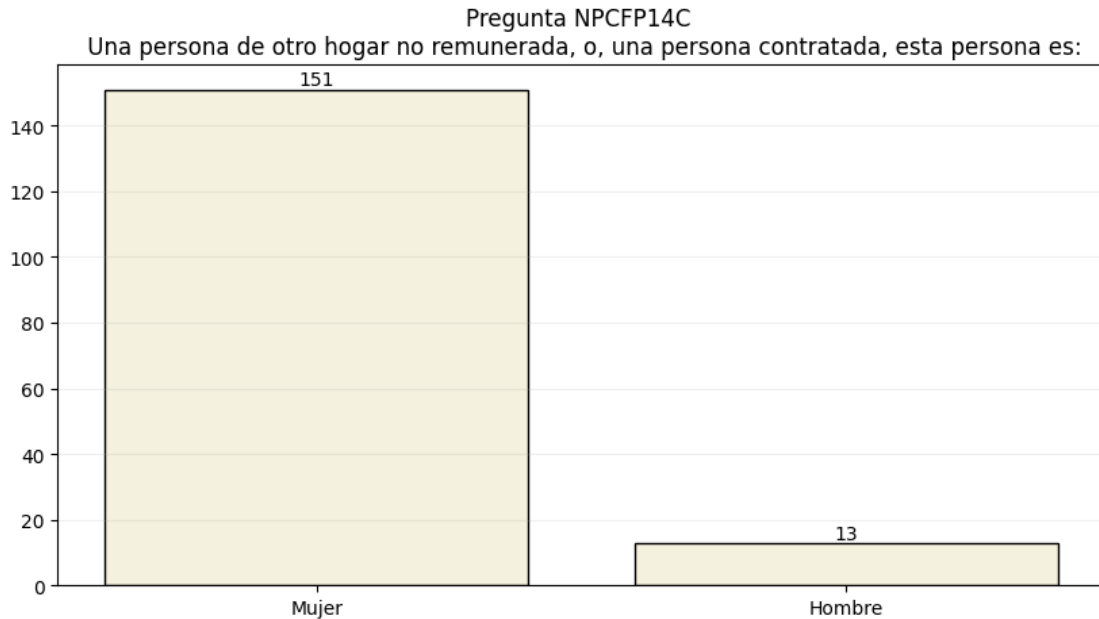
```
[81]: count      164.00
      mean       1.92
      std       0.27
      min       1.00
      25%       2.00
      50%       2.00
      75%       2.00
      max       2.00
      Name: NPCFP24B, dtype: object
```

```
[51]: data['NPCFP24B'].value_counts()
```

```
[51]: 2.0      151
      1.0      13
      Name: NPCFP24B, dtype: int64
```

```
[82]: data['NPCFP24B'] = data['NPCFP24B'].replace([1,2], ['Hombre', 'Mujer'])
```

```
[83]: plt.figure(figsize=(10,5))
      bars = plt.bar(data['NPCFP24B'].value_counts().index.tolist(),data['NPCFP24B'].
      ↪value_counts().tolist(), edgecolor = 'k', color = '#f4f1de')
      #plt.xticks([1,2], ['Hombre', 'Mujer'])
      plt.title('Pregunta NPCFP14C \n Una persona de otro hogar no remunerada, o, una
      ↪persona contratada, esta persona es:')
      plt.bar_label(bars)
      plt.grid(alpha = 0.2, axis = 'y')
      plt.show()
```



2.12 NPCFP33G

2.12.1 31.Durante los ÚLTIMOS 30 DÍAS, ...utilizó servicios de salud para: Servicios de psicología o psiquiatria?

1. Sí
2. No

Datos: 24536

```
[84]: data['NPCFP33G'].count()
```

```
[84]: 24536
```

```
[85]: data['NPCFP33G'].describe().apply("{0:.2f}".format)
```

```
[85]: count      24536.00
      mean         0.02
      std         0.14
      min         0.00
      25%         0.00
      50%         0.00
      75%         0.00
      max         1.00
      Name: NPCFP33G, dtype: object
```

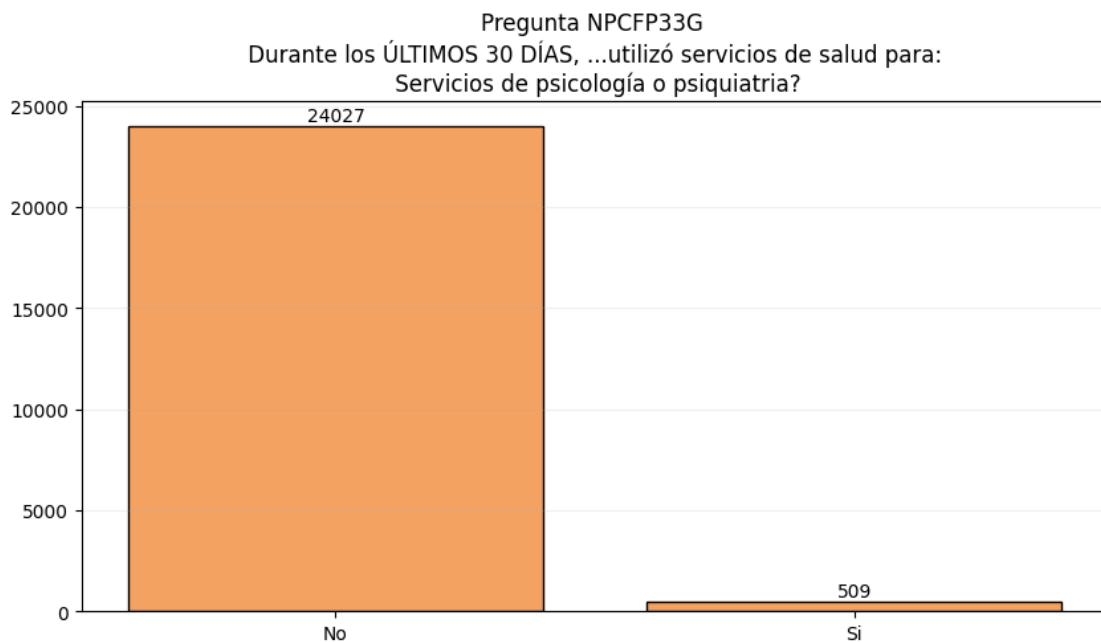
```
[86]: data = data.replace({'NPCFP33G':2},0)
```

```
[87]: data['NPCFP33G'].value_counts()
```

```
[87]: 0    24027
      1     509
      Name: NPCFP33G, dtype: int64
```

```
[88]: data['NPCFP33G'] = data['NPCFP33G'].replace([0,1], ['No','Si'])
```

```
[89]: plt.figure(figsize=(10,5))
      bars = plt.bar(data['NPCFP33G'].value_counts().index.tolist(),data['NPCFP33G'].
      ↪value_counts().tolist(), edgecolor = 'k', color = '#f4a261')
      #plt.xticks([0,1], ['No','Si'])
      plt.title('Pregunta NPCFP33G \n Durante los ÚLTIMOS 30 DÍAS, ...utilizó_
      ↪servicios de salud para: \n Servicios de psicología o psiquiatria?')
      plt.bar_label(bars)
      plt.grid(alpha = 0.2, axis = 'y')
      plt.show()
```



2.13 NPCFP36

2.13.1 34. En los ÚLTIMOS 30 DÍAS, ¿cuántas veces por semana ...practicó deporte o realizó actividad física por 30 minutos continuos o más?

1. 3 o más veces por semana
2. 1 a 2 veces por semana
3. Menos de una vez a la semana
4. No practicó deporte ni tuvo actividad física en el mes

Datos: 22213

```
[90]: data['NPCFP36'].count()
```

```
[90]: 22213
```

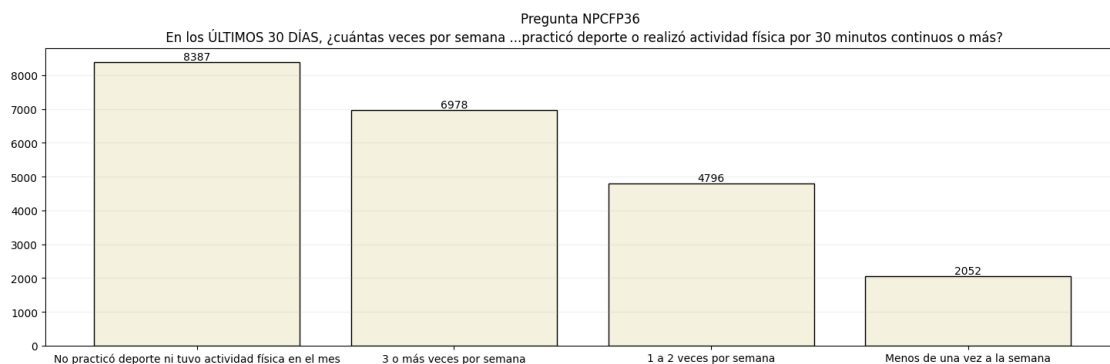
```
[91]: data['NPCFP36'].value_counts()
```

```
[91]: 4.0    8387
      1.0    6978
      2.0    4796
      3.0    2052
      Name: NPCFP36, dtype: int64
```

```
[92]: data['NPCFP36'] = data['NPCFP36'].replace([1,2,3,4], ['3 o más veces por se┐
      ↪mana',
      '1 a 2 veces por semana',
      'Menos de una vez a la semana',
      'No practicó deporte ni tuvo actividad física en el mes'])
```

```
[ ]:
```

```
[95]: plt.figure(figsize=(18,5))
      bars = plt.bar(data['NPCFP36'].value_counts().index.tolist(),data['NPCFP36'].
      ↪value_counts().tolist(), edgecolor = 'k', color = '#f4f1de')
      #plt.xticks([1,2,3,4], ['3 o más veces por semana',
      # '1 a 2 veces por semana',
      # 'Menos de una vez a la semana',
      # 'No practicó deporte ni \n tuvo actividad física en el mes'])
      plt.title('Pregunta NPCFP36 \n En los ÚLTIMOS 30 DÍAS, ¿cuántas veces por se┐
      ↪mana ...practicó deporte o realizó actividad física por 30 minutos┐
      ↪continuos o más?')
      plt.bar_label(bars)
      plt.grid(alpha = 0.2, axis = 'y')
      plt.show()
```



2.14 NPCFP37

2.14.1 35. ¿Cuál fue la razón principal para que ...no practicara deporte ni tuviera actividad física en los ÚLTIMOS 30 DÍAS?

1. Su salud no se lo permite
2. No le interesa o no le motiva
3. No tuvo tiempo
4. No tiene lugares donde hacerlo
5. No le alcanza el dinero para hacerlo
6. Otra razón

Datos: 8387

```
[96]: data['NPCFP37'].count()
```

```
[96]: 8387
```

```
[97]: data['NPCFP37'].value_counts()
```

```
[97]: 3.0    3514
      2.0    2721
      1.0    1668
      6.0     266
      4.0     187
      5.0      31
      Name: NPCFP37, dtype: int64
```

```
[98]: data['NPCFP37'] = data['NPCFP37'].replace([1,2,3,4,5,6],['Su salud no se lo
↳permite', 'No le interesa o no le motiva','No tuvo tiempo',
                        'No tiene lugares donde hacerlo','No le alcanza el dinero
↳para hacerlo', 'Otra razón'])
```

```
[64]: data['NPCFP37'].describe().apply("{0:.2f}".format)
```

```
[64]: count      8387.00
      mean        2.40
      std         1.04
      min         1.00
      25%         2.00
      50%         2.00
      75%         3.00
      max         6.00
      Name: NPCFP37, dtype: object
```

```
[99]: fig, ax = plt.subplots(figsize=(18, 5))
```

```

g = sns.countplot(ax=ax, data = data, x ='NPCFP37')

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

ax.set_title('¿Cuál fue la razón principal para que ...no practicara deporte ni
↳tuviera actividad física en los ÚLTIMOS 30 DÍAS?')
#ax.set_xticklabels(['Su salud no se lo permite', 'No le interesa o no le
↳motiva', 'No tuvo tiempo',
#
'No tiene lugares donde hacerlo', 'No le alcanza el
↳\ndinero para hacerlo', 'Otra razón'])

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

```



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

```
[ ]:
```