NV (mera Sección (NVCBP) CBP1 vía de acceso a la edificación es: Gendero o camino en tierra	
• 2 • 3 \ • 4 \ Datos: data[' : count mean	Peatonal construida /ehicular destapada /ehicular pavimentada 24536 NVCBP1'].describe() 24536.000000 3.159235	
mean std min 25% 50% 75% max Name: plt.fi bars = plt.ti plt.xt	<pre>3.159235 1.068032 1.000000 2.000000 4.000000 4.000000 4.000000 NVCBP1, dtype: float64 gure(figsize=(10,5)) plt.bar(data['NVCBP1'].value_counts().index.tolist(),data['NVCBP1'].value_counts().tolist(), edgecolor = 'te('Pregunta NVCBP1 \n La vía de acceso a la edificación es') icks([1,2,3,4],['Sendero o camino en \n tierra','Peatonal \n construida','Vehícular \n destapada','Vehícula</pre>	
	Pregunta NVCBP1 La vía de acceso a la edificación es	
1000 800 600 400	7248	
NVO	Sendero o camino en Peatonal Vehícular Vehícular pavimentada CBP2	
1. Bu 2. Re 3. M	gular alo	
mean std min 25% 50% 75% max Name: plt.fi bars =	1.14215 0.40647 1.00000 1.00000 1.00000 3.00000 NVCBP2, dtype: float64 gure(figsize=(10,5)) plt.bar(data['NVCBP2'].value_counts().index.tolist(),data['NVCBP2'].value_counts().tolist(), edgecolor = 'r_label(bars)	'bla
plt.xt	icks([1,2,3], ['Bueno', 'Regular', 'Malo']) tle('Pregunta NVCBP2 \n ¿Cuál es el estado de la vía?') ow() Pregunta NVCBP2 ¿Cuál es el estado de la vía? 20083	
1500 1250 1000 7500		
0. No 1. Si Datos: data['	24536 NVCBP3'].value_counts() 1953	
Name:	2583 NVCBP3, dtype: int64 NVCBP3'].describe() 24536.000000	
data = data[' 1 2 0 Name: plt.fi	2.000000 NVCBP3, dtype: float64 data.replace({'NVCBP3':2},0) NVCBP3'].value_counts() 1953 2583 NVCBP3, dtype: int64 gure(figsize=(10,5))	
plt.ti plt.xt	Pregunta NVCBP3 ¿La edificación donde está ubicada la vivienda tiene andén? 21953	DI
1500		
	2583 No Si	
4. ¿L 0. No 1. Si Datos:		
: 1 1 2 Name: data[' count mean std min	5526 9010 NVCBP4, dtype: int64 NVCBP4'].describe() 24536.000000 1.367216 0.482056 1.000000	
25% 50% 75% max Name: data = data[' : 1 1	1.000000 1.000000 2.000000 2.000000 NVCBP4, dtype: float64 data.replace({'NVCBP4':2},0) NVCBP4'].value_counts()	
<pre>Name: plt.fi bars = plt.ti plt.xt plt.ba plt.sh</pre>	Pole (Pregunta NVCBP4) (atype: int64) gure(figsize=(10,5)) plt.bar(data['NVCBP4'].value_counts().index.tolist(),data['NVCBP4'].value_counts().tolist(), edgecolor = tle('Pregunta NVCBP4 \n ¿La edificación está ubicada en un conjunto residencial?') icks([0,1],['No','Si']) r_label(bars) ow() Pregunta NVCBP4 ¿La edificación está ubicada en un conjunto residencial?	'bl≀
1400 1200 1000 800	15526	
400 200		
NVC 5. La 1. Su 2. Ins	CBP5 iluminación de la vía de acceso a la edificación en las noches es: ciente suficiente o tiene	
Datos: data[' 1 2 2 3 Name:		
count mean std min 25% 50% 75% max Name:	24536.000000 1.140773 0.368506 1.000000 1.000000 1.000000 3.000000 NVCBP5, dtype: float64	
<pre>bars = plt.xt plt.ti</pre>	plt.bar(data['NVCBP5'].value_counts().index.tolist(),data['NVCBP5'].value_counts().tolist(), edgecolor = icks([1,2,3], ['Suficiente', 'Insuficiente', 'No tiene']) tle('Pregunta NVCBP5 \n La iluminación de la vía de acceso de la edificación en las noches es:') r_label(bars) ow() Pregunta NVCBP5 La iluminación de la vía de acceso de la edificación en las noches es: 21264	'bl
1750 1500 1250 1000		
	Suficiente Insuficiente No tiene	
plt.ti plt.sh	e(data['NVCBP5'].value_counts().tolist(), labels = ['Suficiente','Insuficiente','No Tiene'],autopct='%1.1f% tle('Pregunta NVCBP5 \n La iluminación de la vía de acceso de la edificación en las noches es:') ow() Pregunta NVCBP5 uminación de la vía de acceso de la edificación en las noches es:	%%'
	Suficiente 86.7% 0.7% No Tiene 12.6% Insuficiente	
6. ¿C Datos: data[' count	NVCBP6'].describe() 24536.000000	
	5.335915 4.409705 1.000000 3.000000 4.000000 6.000000 30.000000 NVCBP6, dtype: float64 ting a figure composed of two matplotlib.Axes objects (ax_box and ax_hist)	
sns.bo #ax_bo #sns.h ax_his ax_his ax_his ax_his ax_his ax_his ax_box ax_his ax_box plt.sh	uántos pisos tiene la edificación donde está ubicada la vivienda?	
sns.bo #ax_bo #sns.h ax_his ax_his ax_his ax_his ax_his ax_his ax_his ax_box ax_his ax_box plt.sh ¿Co 6000 4000 7. ¿L 0. No 1. Si Datos:	xxDox(x = data('NxCBPG'), ax-ax_box() xx.bxputo(data('NxCBPG'), yert = false) istplot(data-data, x='NxCBPG'), yert = false) istplot(data('NxCBPG'), bins = np.arange(32) - 0.5, edgecolor = 'k') t.axvline(data('NxCBPG'), mean(), color='r', linestyle='dashed', linewidth=1, label = 'Median') t.axvline(data('NxCBPG'), median(), color='b', linestyle='i', linestyle='linesty	
sns.bo #ax_bo #sns.h ax_his ax_his ax_his ax_his ax_his ax_his ax_his ax_box ax_his ax_box plt.sh ¿Co NVC 7. ¿L 0. No 1. Si Datos: data[' 1.0 2.0 Name: data[' 1.0 2.0 Name: data[' 1.0 2.0 Name:	xylot(x = data[wycser]), x=xx box) x	
### sns.bo #### #### sns.bo ### ### sns.bo #	Associate (and ("MCSBP"), as very, box) Associate (and ("MCSBP"), as very, box) Associate (and ("MCSBP"), bins = non-annue(32) = 0.5, reduction = "k") Lavida (data ("MCSBP"), bins = non-annue(32) = 0.5, reduction = "k") Lavida (data ("MCSBP"), bins = non-annue(32) = 0.5, reduction = "k") Lavida (data ("MCSBP"), bins = non-annue(32) = 0.5, reduction = "k") Lavida (data ("MCSBP"), bins = non-annue(32) = 0.5, reduction = "k") Lavida (data ("MCSBP"), bins = non-annue(32) = 0.5, reduction = "k") Lavida (data ("MCSBP"), cultification ("MCSBP"), cultification (data ("MCSBP"), cultification ("M	
### share sh	Assented (fatter) (1988), as-ease, 0x0 sectors (1987) as-ease, 0x1 sectors (1987) as-e	
### Sins. bo ### ### ### ### ### ### ### ### ###	constructions (commonly) over - micro-constructions (commonly) over the construction (commonly) over commonly) over commonly over the construction (commonly) over the construction (commonly) over commonly) over commonly over commonly over the construction (commonly) over commonly over the construction (commonly) over commonly over commonly over commonly over the construction (commonly) over commonly	
### share sh	colocity = serg("interface"), acces_pose_pose_pose_pose_pose_pose_pose_po	
### sns.bo #### #### ###########################	specificación donde está ubicada la vivienda tiene ascensor? se dificación donde está ubicada la vivienda tiene ascensor? La edificación donde está ubicada la vivienda tiene ascensor? La edificación donde está ubicada la vivienda tiene ascensor? 2. a descentida de la vivienda está dedicado a negocios de industria, comercio o servience con servi	cic
### Share in the state of the s	splancy and and posters (), where a post of the company of the of t	cic
### Share ### Sh	accessor of the Company of the Compa	cic
### Show ###	accessor of color (NORT) color (NO	cic
### Share ### Sh	sections - 1 (1997) 1997 1	cic
### Share sh	and a second content of the content	cic
sns.bo #ax_bo #ax_bo #ax_his ax_his a	Sept 1 - Sep	
sns.bo #ax_bo #ax_bo #ax_his ax_his a	place is a control of the property of the control o	
sns.bo #ax_bo #ax_bo #ax_bo #ax_his ax_his a	place a professional production of the control of t	
## Substance of the stance of	general control of the control of th	
sns.bo sns.so sns.so	give reparts of a law of the state of the st	
### ### #### #########################	Septiments of the control of the con	
## ## ## ## ## ## ## ## ## ## ## ## ##	SPOT STATE OF THE	
## ## ## ## ## ## ## ## ## ## ## ## ##	The second secon	
### ### ### ### ### ### ### ### ### ##	Services and servi	
### ### ### ### ### ### ### ### ### ##	Sept of the control o	
## ## ## ## ## ## ## ## ## ## ## ## ##	Septimination of the septimina	sel
### ### ### ### ### ### ### ### ### ##	Service of the control of the contro	sel
## A GO A CONTROL OF A CONTROL	Service of the control of the contro	sel
### ### ### ### ### ### ### ### ### ##	And the second of the second o	s e
## A C C C C C C C C C C C C C C C C C C	And the second of the second o	sel
## A CONTRIBUTED TO THE PROPERTY OF THE PROPER	Figure 2 and the control of the cont	
### ### ### ### ### ### ### ### ### ##	The second control of	se!
## A CONTROL OF A	BET A CONTROL OF THE	se!
### ### ### ### ### ### ### ### ### ##	The control of the co	se!

80]: (80]: 3	25000 - 24518
80]: (80]: :	15000 - 15000 - 18
81]:	NVCBP11B 11B. ¿Cuenta con el servicio público de Acueducto? Datos: 24536 Interpreta de la contenta del la contenta de la co
92]:	<pre>lata = data.replace({'NVCBP11B':2},0)</pre> lata['NVCBP11B'].value_counts() 24323 213 lame: NVCBP11B, dtype: int64 Fig, ax = plt.subplots(figsize=(15, 5))
	<pre>ing, ax = pic.subplots(\(\text{Igsize}(15, 5)\) is = sns.countplot(ax=ax, data = data, x ='NVCBP11B') for bars in ax.containers: ax.bar_label(bars, fmt='%.0f', fontsize=9) ix.set_title('Pregunta NVCBP11B \n ¿Cuenta con el servicio público de Acueducto?') ix.set_xticklabels(['No','Si']) it.grid(alpha = 0.2, axis = 'y') it.show() Pregunta NVCBP11B</pre>
	¿Cuenta con el servicio público de Acueducto? 25000 20000 15000 10000
	5000 - 213 NO NVCBP11B
9]: (Inc. ¿Cuenta con el servicio público de Alcantarillado? Datos: 24536 Data['NVCBP11C'].count() Data = data.replace({'NVCBP11C':2},0)
3]:	<pre>lata['NVCBP11C'].value_counts() 23530 1006 lame: NVCBP11C, dtype: int64 Fig, ax = plt.subplots(figsize=(15, 5)) g = sns.countplot(ax=ax, data = data, x ='NVCBP11C') For bars in ax.containers: ax.bar_label(bars, fmt='%.0f', fontsize=9)</pre>
;	ax.bar_label(bars, tmt= %.01°, tontsize=9) ax.set_title('Pregunta NVCBP11C \n ¿Cuenta con el servicio público de Alcantarillado?') ax.set_xticklabels(['No','Si']) att.grid(alpha = 0.2, axis = 'y') att.show() Pregunta NVCBP11C ¿Cuenta con el servicio público de Alcantarillado? 23530
	20000 - 15000 - 10000 -
	NVCBP11C NVCBP11C NVCBP11C Si NVCBP11C NVCBP11C
94]: : 95]: (96]: (<pre>lata = data.replace({'NVCBP11D':2},0)</pre> lata['NVCBP11D'].value_counts()
98]:	<pre>lame: NVCBP11D, dtype: int64 Fig, ax = plt.subplots(figsize=(15, 5)) g = sns.countplot(ax=ax, data = data, x ='NVCBP11D') For bars in ax.containers: ax.bar_label(bars, fmt='%.0f', fontsize=9) ax.set_title('Pregunta NVCBP11D \n ¿Cuenta con el servicio público de Recolección de Basuras?') ax.set_xticklabels(['No','Si']) alt.grid(alpha = 0.2, axis = 'y')</pre>
	Pregunta NVCBP11D ¿Cuenta con el servicio público de Recolección de Basuras? 25000 20000 15000
	10000 - 161 - 161 - Si NVCBP11D
	1. (NVCBP14A) Fábricas o Industrias 2. (NVCBP14B) Basureros o botaderos de basuras 3. (NVCBP14C) Plazas de Mercado o Mataderos 4. (NVCBP14D) Terminales de Buses 5. (NVCBP14E) Bares o discotecas 6. (NVCBP14L) Prostíbulos
	7. (NVCBP14F) Expendios de droga (ollas) 8. (NVCBP14G) Lotes baldíos o sitios oscuros y peligrosos 9. (NVCBP14H) Lineas de alta tensión 10. (NVCBP14I) Caños de aguas resiguales 11. (NVCBP14J) Zona de riesgo de incendio forestal 12. (NVCBP14K) Talleres de mecánica, servitecas o estaciones de gasolina Datos: 24536
101	<pre>problemas = [i for i in data.columns if ('NVCBP14') in i] for i in problemas: print(i, data[i].count()) IVCBP14A 24536 IVCBP14B 24536 IVCBP14C 24536 IVCBP14D 24536 IVCBP14E 24536 IVCBP14E 24536 IVCBP14E 24536 IVCBP14F 24536</pre>
I I I I I I	<pre>IVCBP14G 24536 IVCBP14H 24536 IVCBP14I 24536 IVCBP14J 24536 IVCBP14K 24536 I</pre>
	'Terminales de Buses', 'Bares o discotecas', 'Prostíbulos', 'Expendios de droga (ollas)', 'Lotes baldíos o sitios oscuros y peligrosos', 'Lineas de alta tensión', 'Caños de aguas resiguales', 'Zona de riesgo de incendio forestal', 'Talleres de mecánica, servitecas o estaciones de gasolina'] Fig, axes = plt.subplots(4,3, figsize = (15,15), squeeze=False) Fig.subplots_adjust(top=0.9)
	<pre>ig.subplots_adjust(top=0.9) ixli = axes.flatten() ig.suptitle('La vivienda está cerca de:') for ax,cols,names in zip(axli,problemas,list_pro): 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.margins(y=0.1) # make room for the labels for bars in ax.containers: ax.bar_label(bars, fmt='%.0f', fontsize=9) ilt.tight_layout()</pre>
	Expression () Fábricas o Industrias La vivienda está cerca de: Basureros o botaderos de basuras Plazas de Mercado o Mataderos
	2044 No N
	1758 No N
	2354 No No No NVCBP14F Caños de aguas resiguales Zona de riesgo de incendio forestal 20000 20000 15000 20000 15000 21361 20000 15000 20000 15000 20000 15000 21361 20000 21361
	NVCBP15 Siguientes problemas presenta el entorno cercano a su vivienda? 1. (NVCBP15A) Ruido
	 (NVCBP15B) Exceso de anuncios publicitarios (NVCBP15C) Inseguridad (NVCBP15D) Contaminación del aire (NVCBP15E) Malos Olores (NVCBP15F) Disposición inadecuada de las basuras. (NVCBP15G) Invasión del espacio público (NVCBP15H) Presencia de insectos, roedores o animales que causen molestia (NVCBP15I) Contaminación de Cuerpos de Agua (NVCBP15J) Abandono de escombros
L08	11. (NVCBP15K) Disposición inadecuada de residuos hospitalarios 12. (NVCBP15L) Arboles que ponen en riesgo las viviendas o sus habitantes 13. (NVCBP15M) Lugares con presencia de orina o excremento humano Patos: 24536 Problemas15 = [i for i in data.columns if ('NVCBP15') in i] Print(problemas15) Print(problemas15) Print(problemas15, 'NVCBP15B', 'NVCBP15C', 'NVCBP15D', 'NVCBP15E', 'NVCBP15F', 'NVCBP15G', 'NVCBP15H', 'NVCBP15I', 'NVCBP15I', 'NVCBP15K', 'NVCBP15L', 'NVCBP15M']
	For i in problemas15: print(i,data[i].count()) IVCBP15A 24536 IVCBP15B 24536 IVCBP15C 24536 IVCBP15C 24536 IVCBP15D 24536 IVCBP15F 24536 IVCBP15F 24536 IVCBP15F 24536 IVCBP15F 24536 IVCBP15F 24536 IVCBP15F 24536
110	<pre>IVCBP15I 24536 IVCBP15J 24536 IVCBP15K 24536 IVCBP15L 24536 IVCBP15M 24536 I</pre>
114	Disposición inadecuada de las basuras', Invasión del espacio público', Presencia de insectos, roedores o animales que causen molestia', Contaminación de Cuerpos de Agua', Abandono de escombros', Disposición inadecuada de residuos hospitalarios',' Árboles que ponen en riesgo las viviendas o sus habitantes', Lugares con presencia de orina o excremento humano'] Fig. axes = plt.subplots(4,3, figsize = (15,15), squeeze=False) Fig.subplots_adjust(top=0.9) ixli = axes.flatten() Fig.suptitle('¿Cuales de los siguientes problemas presenta el entorno cercano a su vivienda?')
	<pre>for ax,cols,names in zip(axli,problemas15,list_problemas): 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(0,25000) 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()</pre>
	<pre>clt.show() Fig, ax = plt.subplots(figsize=(5, 5)) (x = sns.countplot(ax=ax, data = data, x ='NVCBP15M')) For bars in ax.containers: ax.bar_label(bars, fmt='%.0f', fontsize=9) ax.set_title('Lugares con presencia de orina o excremento humano') ax.set_xticklabels(['No','Si']) ax.set_ylim(0,25000)</pre>
	olt.grid(alpha = 0.2, axis = 'y') olt.show() Cuales de los siguientes problemas presenta el entorno cercano a su vivienda? Exceso de anuncios publicitarios 25000 20000 15000 15000 10000 10000 11202
	No NVCBP15A NVCBP15B NVCBP15C Contaminación del aire 25000 Malos Olores Disposición inadecuada de las basuras 25000 19856 20000 19918 20000 15
	No N
	3383 No N
	Lugares con presencia de orina o excremento humano 23660 20000
	15000 -
	NVCBP15M NVCBP15M NVCBP16 I6. En total : cuántos grupos de personas (hogares) preparan los alimentos por separado e
	<pre>lata['NVCBP16'].value_counts()</pre>
116 (16]:	2
116 (16]: 117 (17]:	<pre>blt.figure(figsize=(10,5)) counts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#ED1C24' cticklabels = [i for i in range(5)] cplt.xticks(range(5), ticklabels) color:xticks(range(7))</pre>
116 (16]: 117 (17]:	Name: NVCBP16, dtype: int64 Plt.figure(figsize=(10,5)) counts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#ED1C24' tticklabels = [i for i in range(5)] tplt.xticks(range(5), ticklabels)
116 (16]: 117 (17]:	Name: NVCBP16, dtype: int64 plt.figure(figsize=(10,5)) nounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#ED1C24' plt.xtickabeLs = [i for i in range(5)] plt.xticks(range(5), ticklabeLs) plt.xticks(range(7)) plt.bar_label(bars) plt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta violt.xlim([0,7]) plt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience
116 (16]: 117 (17]:	Name: NVCBP16, dtype: int64 Not.figure(figsize=(10,5)) Nounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#ED1C24' ##ED1C24' ##
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 (.17 (.17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 (.17 (.17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 (.17 (.17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 (.17 (.17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 (.17 (.17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 [.17 [.17]: .21 [lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 [.17 [.17]: .21 [lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 [.17 [.17]: .21 [lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
.16 [.17 [.17]: .21 [lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322
116 (16]: 117 (17]:	lame: NVCBP16, dtype: int64 ilt.figure(figsize=(10,5)) ounts, edges, bars = plt.hist(data['NVCBP16'], bins = np.arange(1,8) - 0.5, edgecolor = 'black', color = '#EDIC24' tit.klabels = [i for i in range(5)] plt.xticks(range(5), ticklabels) llt.xticks(range(7)) llt.bar_label(bars) llt.title('Pregunta NVCBP16 \n ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vi lt.xlim(e,7]) llt.show() Pregunta NVCBP16 ¿Cuántos grupos de personas (hogares) preparan los alimentos por separado en esta vivience 25000 - 24322 20000 - 15000 - 24322