Data cleaning template for R modeling

- added the new 200 data points from Prolific 2
- to compare how the new data's mobility values are different from the previous
- final update: 15th May, included engine information

In [3]:
```
# importing libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.ticker import StrMethodFormatter
```

In [107…
```
# loading the csv file for the collected responses as of "date" - data 1

df = pd.read_csv('data\CO2-Rechner_February+27,+2024_20.14.csv', header = 0)

pd.set_option('display.max_columns', None)
df.head(9)
```

Out[107]:

	StartDate	EndDate	
0	Start Date	End Date	Res
1	{"ImportId":"startDate","timeZone":"Europe/Ber...	{"ImportId":"endDate","timeZone":"Europe/Berlin"}	{"ImportI
2	2024-01-26 17:17:08	2024-01-26 17:19:54	
3	2024-01-26 17:16:34	2024-01-26 17:26:53	
4	2024-01-26 17:18:24	2024-01-26 17:29:37	
5	2024-01-26 17:17:52	2024-01-26 17:29:44	
6	2024-01-26 17:30:04	2024-01-26 17:34:26	
7	2024-01-26 17:16:39	2024-01-26 17:36:41	
8	2024-01-26 17:35:17	2024-01-26 17:41:32	

In [108…

```
#old df_selected = df.iloc[1:, [25, 50, 51, 56, 61, 83, 86, 87, 88, 89, 90, 91, 92, 93

## adding engines: 26, 27, 29, 30, 32, 33, 35, 36, 38, 39

df_selected = df.iloc[2:, [25, 50, 51, 56, 87, 90, 91, 92, 93, 94, 95, 96, 97, 99, 106

df_selected.head(10)
```

Out[108]:

	Q1.1	Q5.2	Q6.1	Q4.4	Q1.7	Q3_1.1	Q3_2.1	Q3_3.1	Q3_4.1	
2	1	Männlich	36	2500	SPD	NaN	NaN	NaN	NaN	Beruf A
3	1	Männlich	60	1000	Bündnis 90/Die Grünen	NaN	NaN	NaN	Allgemeine oder fachgebundene Hochschulreife/A...	
4	2	Männlich	57	6500	7	NaN	NaN	NaN	NaN	Beruf A
5	2	Männlich	36	6500	Bündnis 90/Die Grünen	NaN	NaN	NaN	NaN	
6	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
7	2	Männlich	31	2500	AfD	NaN	NaN	NaN	Allgemeine oder fachgebundene Hochschulreife/A...	
8	0	Weiblich	63	1500	7	NaN	NaN	Realschulabschluss (Mittlere Reife) oder gleic...	NaN	
9	1	Männlich	45	1500	CDU/CSU	NaN	NaN	Realschulabschluss (Mittlere Reife) oder gleic...	NaN	Beruf A
10	1	Weiblich	41	1900	7	NaN	NaN	Realschulabschluss (Mittlere Reife) oder gleic...	NaN	
11	2	Weiblich	70	5400	Bündnis 90/Die Grünen	NaN	NaN	NaN	NaN	

In [109…

```
df_selected.columns.values
```

Out[109]:

```
array(['Q1.1', 'Q5.2', 'Q6.1', 'Q4.4', 'Q1.7', 'Q3_1.1', 'Q3_2.1',
       'Q3_3.1', 'Q3_4.1', 'Q3_5.1', 'Q3_6.1', 'Q3_7.1', 'Q3_8.1', 'Q5.4',
       'CO2_Wohnen', 'CO2_Strom', 'CO2_Wohnen_Strom', 'CO2_ Kreuzfahrt',
       'CO2_Flugreisen', 'CO2_ÖPNV', 'CO2_Auto1', 'CO2_Auto2',
       'CO2_Auto3', 'CO2_Auto4', 'CO2_Auto5', 'CO2_Auto_Gesamt',
       'CO2_Mobilität', 'CO2_Ernährung', 'CO2_Sonstiger_Konsum',
       'Öffentliche_emissionen', 'CO2_Gesamt',
       'Einschätzung_Wohnen_Strom', 'Einschätzung_Mobilität',
       'Einschätzung_Ernährung', 'Einschätzung_Sonstiger_Konsum',
       'Einschätzung_Gesamt', 'Q3.1', 'Q4.1', 'Q7.1', 'Q8', 'Q11', 'Q12',
       'Q15', 'Q16', 'Q19', 'Q20'], dtype=object)
```

In [110…
```
# Rename the columns

df_selected = df_selected.rename(columns={'Q1.1': 'no_cars', 'Q5.2': 'gender', 'Q6.1':
       'Q1.7': 'political_party',
       'Q3_1.1': 'education1', 'Q3_2.1': 'education2', 'Q3_3.1': 'education3',
       'Q3_4.1': 'education4', 'Q3_5.1': 'education5', 'Q3_6.1': 'education6', 'Q3_7.1
       'Q5.4': 'postal_code', 'CO2_Wohnen': 'CO2_housing', 'CO2_Strom' : 'CO2_electric
       'CO2_ Kreuzfahrt': 'CO2_cruise', 'CO2_Flugreisen': 'CO2_flight', 'CO2_ÖPNV': 'C
       'CO2_Auto2' : 'CO2_car2',
       'CO2_Auto3' : 'CO2_car3', 'CO2_Auto4': 'CO2_car4', 'CO2_Auto5': 'CO2_car5', 'CO
       'CO2_Mobilität': 'CO2_mobility', 'CO2_Ernährung': 'CO2_food', 'CO2_Sonstiger_Ko
       'Öffentliche_emissionen': 'public_emission', 'CO2_Gesamt': 'CO2_total',
       'Einschätzung_Wohnen_Strom': 'belief_housing_electricity', 'Einschätzung_Mobili
       'Einschätzung_Ernährung': 'belief_food', 'Einschätzung_Sonstiger_Konsum': 'beli
        'Q3.1': 'engine_1', 'Q4.1': 'engine_2', 'Q7.1': 'engine_3', 'Q8': 'engine_4',
       'Q15': 'engine_7', 'Q16': 'engine_8', 'Q19': 'engine_9', 'Q20': 'engine_10'})
```

In [111…
```
## Creating the first batch of the collected responses
data_1 = df_selected.copy()
```

In [112…
```
len(data_1)
```

Out[112]:
```
438
```

In [113…
```
## adding the batch info
df_selected['batch'] = 1
```

2nd dataset collected

In [114…
```
#### load the dataset

df_prolific = pd.read_csv('data\Carbon+Beliefs+-+Prolific_March+10,+2024_14.10.csv', h

df_prolific.head(10)
```

Out[114]:

	StartDate	EndDate	
0	Start Date	End Date	Res
1	{"ImportId":"startDate","timeZone":"Europe/Ber...	{"ImportId":"endDate","timeZone":"Europe/Berlin"}	{"ImportI
2	2024-03-06 14:30:30	2024-03-06 14:36:37	
3	2024-03-06 14:30:49	2024-03-06 14:39:54	
4	2024-03-06 14:29:34	2024-03-06 14:40:05	
5	2024-03-06 14:33:30	2024-03-06 14:41:42	
6	2024-03-06 14:30:41	2024-03-06 14:41:50	
7	2024-03-06 14:30:26	2024-03-06 14:42:28	
8	2024-03-06 14:31:59	2024-03-06 14:44:35	
9	2024-03-06 14:31:33	2024-03-06 14:45:31	

In [115…

```
# Select the columns with the variables of interests

## adding engines: 27, 29, 31, 33, 35, 37, 39, 41, 43, 45

df_prolific = df_prolific.iloc[2:, [26, 56, 57, 62, 94, 97, 98, 99, 100, 101, 102, 103
                                    122, 128, 134, 140, 146, 152, 153, 154, 156, 157, 1


df_prolific.head(10)
```

Out[115]:

	Q1.2	Q4.3	Q5.2	Q4.4	Q1.8	Q3_1.1	Q3_2.1	Q3_3.1	Q3_4.1	E
2	2	Männlich	39	4000	Einer anderen Partei	NaN	NaN	NaN	NaN	
3	0	Männlich	38	5000	SPD	NaN	NaN	NaN	NaN	
4	0	Männlich	45	2500	Bündnis 90/Die Grünen	NaN	NaN	NaN	NaN	
5	2	Männlich	22	4900	Bündnis Sarah Wagenknecht	NaN	NaN	NaN	Allgemeine oder fachgebundene Hochschulreife/A...	
6	2	Männlich	33	5000	Einer anderen Partei	NaN	NaN	NaN	NaN	
7	0	Männlich	29	2100	Bündnis 90/Die Grünen	NaN	NaN	NaN	Allgemeine oder fachgebundene Hochschulreife/A...	
8	1	Männlich	28	3000	Bündnis 90/Die Grünen	NaN	NaN	NaN	NaN	
9	0	Männlich	37	5000	AfD	NaN	NaN	Realschulabschluss (Mittlere Reife) oder gleic...	NaN	
10	1	Männlich	43	1900	Einer anderen Partei	NaN	NaN	NaN	NaN	
11	1	Männlich	33	6000	FDP	NaN	NaN	NaN	NaN	

In [116…

```
df_prolific.columns
```

Out[116]:

```
Index(['Q1.2', 'Q4.3', 'Q5.2', 'Q4.4', 'Q1.8', 'Q3_1.1', 'Q3_2.1', 'Q3_3.1',
       'Q3_4.1', 'Q3_5.1', 'Q3_6.1', 'Q3_7.1', 'Q3_8.1', 'Q5.4', 'CO2_Wohnen',
       'CO2_Strom', 'CO2_Wohnen_Strom', 'CO2_ Kreuzfahrt', 'CO2_Flugreisen',
       'CO2_ÖPNV', 'CO2_Auto1', 'CO2_Auto2', 'CO2_Auto3', 'CO2_Auto4',
       'CO2_Auto5', 'CO2_Auto_Gesamt', 'CO2_Mobilität', 'CO2_Ernährung',
       'CO2_Sonstiger_Konsum', 'Öffentliche_emissionen', 'CO2_Gesamt',
       'Einschätzung_Wohnen_Strom', 'Einschätzung_Mobilität',
       'Einschätzung_Ernährung', 'Einschätzung_Sonstiger_Konsum',
       'Einschätzung_Gesamt', 'Q3.1', 'Q5.1', 'Q8', 'Q10', 'Q13', 'Q15', 'Q18',
       'Q20', 'Q23', 'Q25'],
      dtype='object')
```

In [117…

```
# rename the columns
```

```
df_prolific = df_prolific.rename(columns={'Q1.2':'no_cars', 'Q4.3':'gender', 'Q5.2':'a
                                          'Q3_1.1':'education1', 'Q3_2.1':'education2'
       'Q3_4.1':'education4', 'Q3_5.1':'education5', 'Q3_6.1':'education6', 'Q3_7.1':'
       'Q5.4':'postal_code', 'CO2_Wohnen':'CO2_housing',
       'CO2_Strom':'CO2_electricity', 'CO2_Wohnen_Strom':'CO2_housing_electricity', 'C
       'CO2_ÖPNV':'CO2_public_transport', 'CO2_Auto1':'CO2_car1', 'CO2_Auto2':'CO2_car
       'CO2_Auto5':'CO2_car5', 'CO2_Auto_Gesamt':'CO2_car_total', 'CO2_Mobilität':'CO2
       'CO2_Sonstiger_Konsum':'CO2_other_consumption', 'Öffentliche_emissionen':'publi
       'Einschätzung_Wohnen_Strom':'belief_housing_electricity', 'Einschätzung_Mobilit
       'Einschätzung_Ernährung':'belief_food', 'Einschätzung_Sonstiger_Konsum':'belief
       'Einschätzung_Gesamt':'belief_total',
        'Q3.1': 'engine_1', 'Q4.1': 'engine_2', 'Q8': 'engine_3', 'Q10': 'engine_4', '
       'Q18': 'engine_7', 'Q20': 'engine_8', 'Q23': 'engine_9', 'Q25': 'engine_10'})
```

In [118…

```
## Creating the second batch of the collected responses
data_2 = df_prolific.copy()
len(data_2)
```

Out[118]:

```
200
```

In [119…

```
# adding the batch info
df_prolific['batch'] = 2
```

In [120…

```
# Concatenating the two dataframes

df_concat = pd.concat([df_selected, df_prolific], ignore_index=True)

df_concat.head()
```

Out[120]:

	no_cars	gender	age	income	political_party	education1	education2	education3	educat
0	1	Männlich	36	2500	SPD	NaN	NaN	NaN	
1	1	Männlich	60	1000	Bündnis 90/Die Grünen	NaN	NaN	NaN	Allgemeine fachgebun Hochschulreif
2	2	Männlich	57	6500	7	NaN	NaN	NaN	
3	2	Männlich	36	6500	Bündnis 90/Die Grünen	NaN	NaN	NaN	
4	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

In [121…

```
len(df_concat)
```

Out[121]:

```
638
```

Note 1: cleaning the education columns: 'education_clean'

In [122…
```
# function to clean the education columns: one highest degree remains

def clean_education(row):

    if (pd.notna(row['education8'])) and (pd.isna(row['education7'])) and (pd.isna(row
       (pd.isna(row['education5'])) and (pd.isna(row['education4'])) and (pd.isna(row[
       (pd.isna(row['education2'])) and (pd.isna(row['education1'])):
            return 'Anderer Abschluss'



    elif (pd.notna(row['education7'])):
            return 'Doktorgrad oder Habilitation'

    elif (pd.isna(row['education7'])) and (pd.notna(row['education6'])):
            return '(Fach-) Hochschulabschluss (Bachelor, Master, Magister, Diplom, St

    elif (pd.isna(row['education7'])) and (pd.isna(row['education6'])) and\
         (pd.notna(row['education4'])):
            return 'Allgemeine oder fachgebundene Hochschulreife/Abitur (Gymnasium bzw

    elif (pd.isna(row['education7'])) and (pd.isna(row['education6'])) and\
      (pd.isna(row['education4'])) and (pd.notna(row['education5'])):
            return 'Berufsausbildung, Lehre oder Ausbildung an einer Fachschule'

    elif (pd.isna(row['education7'])) and (pd.isna(row['education6'])) and\
      (pd.isna(row['education4'])) and (pd.isna(row['education5'])) and (pd.notna(row[
            return 'Realschulabschluss (Mittlere Reife) oder gleichwertiger Abschluss'

    elif (pd.isna(row['education7'])) and (pd.isna(row['education6'])) and\
        (pd.isna(row['education5'])) and (pd.isna(row['education4'])) and (pd.isna(row
        (pd.notna(row['education2'])):
            return 'Hauptschulabschluss (Volksschulabschluss) oder gleichwertiger Absc

    elif (pd.isna(row['education7'])) and (pd.isna(row['education6'])) and\
      (pd.isna(row['education5'])) and (pd.isna(row['education4'])) and (pd.isna(row['
        (pd.isna(row['education2'])) and (pd.notna(row['education1'])):
            return '(Noch) kein Abschluss'
    else: None
```

In [123…
```
df_concat['education_clean'] = df_concat.apply(clean_education, axis=1)
```

In [124…
```
df_concat = df_concat.drop(columns=['education1','education2', 'education3', 'educatic
```

In [125…
```
len(df_concat)
```

Out[125]:
```
638
```

In [126…
```
df_concat.head(10)
```

Out[126]:

	no_cars	gender	age	income	political_party	postal_code	CO2_housing	CO2_electrici
0	1	Männlich	36	2500	SPD	39106	1487.5	3
1	1	Männlich	60	1000	Bündnis 90/Die Grünen	30966	1400.4	
2	2	Männlich	57	6500	7	41812	1944	
3	2	Männlich	36	6500	Bündnis 90/Die Grünen	17034	607.95	3.
4	1	NaN	NaN	NaN	NaN	NaN	64.3076	26.
5	2	Männlich	31	2500	AfD	78054	1539.17575	32.666666666666666666
6	0	Weiblich	63	1500	7	42369	2026.5	47
7	1	Männlich	45	1500	CDU/CSU	01904	3024.864	4
8	1	Weiblich	41	1900	7	28237	1875.96	6
9	2	Weiblich	70	5400	Bündnis 90/Die Grünen	12209	2722.0275	42

3rd dataset collected

In [127…

```
#### load the dataset

df_prolific2 = pd.read_csv('data\Carbon+Beliefs+-+Prolific+V2_March+22,+2024_11.46.csv

df_prolific2.head(10)
```

Out[127]:

	StartDate	EndDate	
0	Start Date	End Date	Res
1	{"ImportId":"startDate","timeZone":"Europe/Ber...	{"ImportId":"endDate","timeZone":"Europe/Berlin"}	{"ImportI
2	2024-03-21 15:17:17	2024-03-21 15:30:17	
3	2024-03-21 15:20:20	2024-03-21 15:34:03	
4	2024-03-21 15:23:39	2024-03-21 15:34:55	
5	2024-03-21 15:18:51	2024-03-21 15:36:04	
6	2024-03-21 15:25:55	2024-03-21 15:37:17	
7	2024-03-21 15:21:09	2024-03-21 15:37:29	
8	2024-03-21 15:21:09	2024-03-21 15:39:08	
9	2024-03-21 15:21:54	2024-03-21 15:39:26	

In [128…

```
# adding engines: 27, 29, 31, 33, 35, 37, 39, 41, 43, 45

df_prolific2 = df_prolific2.iloc[2:, [26, 56, 57, 62, 95, 98, 100, 109, 110, 111, 112,
                                      116, 117, 123, 129, 135, 141, 147, 148, 149,
                                     27, 29, 31, 33, 35, 37, 39, 41, 43, 45]]

df_prolific2.head(3)
```

Out[128]:

	Q1.2	Q4.3	Q5.2	Q4.4	Q1.9	Q3.6	Q5.3	CO2_Wohnen	CO2_Strom	C
2	1	Männlich	40	8000	CDU/CSU	(Fach-) Hochschulabschluss (Bachelor, Master, ...	80803	1737	780	
3	1	Weiblich	22	1200	Bündnis Sarah Wagenknecht	Allgemeine oder fachgebundene Hochschulreife/A...	06406	2251.152	663	
4	0	Männlich	23	530	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	76131	340.452	663	

In [129…

```
df_prolific2.columns
```

Out[129]:

```
Index(['Q1.2', 'Q4.3', 'Q5.2', 'Q4.4', 'Q1.9', 'Q3.6', 'Q5.3', 'CO2_Wohnen',
       'CO2_Strom', 'CO2_Wohnen_Strom', 'CO2_ Kreuzfahrt', 'CO2_Flugreisen',
       'CO2_ÖPNV', 'CO2_Auto1', 'CO2_Auto2', 'CO2_Auto3', 'CO2_Auto4',
       'CO2_Auto5', 'CO2_Auto_Gesamt', 'CO2_Mobilität', 'CO2_Ernährung',
       'CO2_Sonstiger_Konsum', 'Öffentliche_emissionen', 'CO2_Gesamt',
       'Einschätzung_Wohnen_Strom', 'Einschätzung_Mobilität',
       'Einschätzung_Ernährung', 'Einschätzung_Sonstiger_Konsum',
       'Einschätzung_Gesamt_2', 'Q3.1', 'Q5.1', 'Q8', 'Q10', 'Q13', 'Q15',
       'Q18', 'Q20', 'Q23', 'Q25'],
      dtype='object')
```

In [130…

```
# recalculating the Einschätzung values to no. of people with CO2 footprint "higher th
# 100 people including the respondent

belief_columns = ['Einschätzung_Wohnen_Strom', 'Einschätzung_Mobilität','Einschätzung_
       'Einschätzung_Gesamt_2']

for col in belief_columns:
    df_prolific2[col] = df_prolific2[col].astype('float')
    df_prolific2[col] = 100 - df_prolific2[col]
```

In [131…

```
df_prolific2 = df_prolific2.rename(columns={'Q1.2':'no_cars', 'Q4.3':'gender', 'Q5.2':
                                            'Q3.6': 'education_clean', 'Q5.3': 'postal_
       'CO2_Strom': 'CO2_electricity', 'CO2_Wohnen_Strom': 'CO2_housing_electricity',
                                            'CO2_Flugreisen': 'CO2_flight', 'CO2_ÖPNV':
                                            'CO2_Auto2': 'CO2_car2', 'CO2_Auto3':'CO2_c
       'CO2_Auto5':'CO2_car5', 'CO2_Auto_Gesamt':'CO2_car_total', 'CO2_Mobilität':'CO2
       'CO2_Sonstiger_Konsum':'CO2_other_consumption', 'Öffentliche_emissionen':'publi
       'Einschätzung_Wohnen_Strom':'belief_housing_electricity', 'Einschätzung_Mobilit
       'Einschätzung_Ernährung':'belief_food', 'Einschätzung_Sonstiger_Konsum':'belief
       'Einschätzung_Gesamt_2':'belief_total',
        'Q3.1': 'engine_1', 'Q5.1': 'engine_2', 'Q8': 'engine_3', 'Q10': 'engine_4', '
       'Q18': 'engine_7', 'Q20': 'engine_8', 'Q23': 'engine_9', 'Q25': 'engine_10'})
```

In [132…

```
## Creating the third batch of the collected responses

data_3 = df_prolific2.copy()
len(data_3)
```

Out[132]:

```
201
```

In [133…
```
# adding the batch info
df_prolific2['batch'] = 3
```

In [134…
```
# unioning the 3rd dataset
df_concat = pd.concat([df_concat, df_prolific2], ignore_index=True)
len(df_concat)
```

Out[134]:
```
839
```

In [135…
```
df_concat.head()
```

Out[135]:

	no_cars	gender	age	income	political_party	postal_code	CO2_housing	CO2_electricity	CO2_h
0	1	Männlich	36	2500	SPD	39106	1487.5	390	
1	1	Männlich	60	1000	Bündnis 90/Die Grünen	30966	1400.4	39	
2	2	Männlich	57	6500	7	41812	1944	42	
3	2	Männlich	36	6500	Bündnis 90/Die Grünen	17034	607.95	3.85	
4	1	NaN	NaN	NaN	NaN	NaN	64.3076	26.32	

Note 2: mapping urban/rural classifications with the PLZ column

In [136…
```
### loading the cleaned urban/rural classification table - EUROSTAT

df_urban_class = pd.read_csv('classification_urban_by_postal_code.csv', encoding='cp12
df_urban_class.head()
```

Out[136]:

	Unnamed: 0	NUTS3_ID	NUTS1_NAME	NUTS2_NAME	NUTS3_NAME	POSTAL_CODE	CLASSIFICATIOI
0	0	DEA1D	Nordrhein-Westfalen	Düsseldorf	Rhein-Kreis Neuss	41363	PI
1	1	DEA1E	Nordrhein-Westfalen	Düsseldorf	Viersen	41366	PI
2	2	DEA1E	Nordrhein-Westfalen	Düsseldorf	Viersen	41748	PI
3	3	DEA1E	Nordrhein-Westfalen	Düsseldorf	Viersen	41749	PI
4	4	DEA1E	Nordrhein-Westfalen	Düsseldorf	Viersen	41751	PI

In [137…

```
# change the datatype and join the classification table

df_urban_class['POSTAL_CODE'] = df_urban_class['POSTAL_CODE'].astype('str')
df_final = pd.merge(df_concat, df_urban_class, left_on = ['postal_code'], right_on = [
df_final.head()
```

Out[137]:

	no_cars	gender	age	income	political_party	postal_code	CO2_housing	CO2_electricity	CO2_h
0	1	Männlich	36	2500	SPD	39106	1487.5	390	
1	1	Männlich	60	1000	Bündnis 90/Die Grünen	30966	1400.4	39	
2	2	Männlich	57	6500	7	41812	1944	42	
3	2	Männlich	36	6500	Bündnis 90/Die Grünen	17034	607.95	3.85	
4	1	NaN	NaN	NaN	NaN	NaN	64.3076	26.32	

In [138…

```
df_final = df_final.rename(columns={'education_clean' : 'education', 'CLASSIFICATION':
```

In [139…

```
## loading the new classification - RLK and KTU

df_urban_class_new = pd.read_csv('new_classification_urban_by_postal_code.csv', encodi
df_urban_class_new.head()
```

Out[139]:

	Unnamed: 0	POSTAL_CODE	NUTS3_ID	NUTS1_NAME	NUTS2_NAME	NUTS3_NAME	RLK2022	KTU
0	0	41363	DEA1D	Nordrhein-Westfalen	Düsseldorf	Rhein-Kreis Neuss	sehr zentral	Städt
1	1	41366	DEA1E	Nordrhein-Westfalen	Düsseldorf	Viersen	sehr zentral	Städt
2	2	41748	DEA1E	Nordrhein-Westfalen	Düsseldorf	Viersen	sehr zentral	Städt
3	3	41749	DEA1E	Nordrhein-Westfalen	Düsseldorf	Viersen	sehr zentral	Städt
4	4	41751	DEA1E	Nordrhein-Westfalen	Düsseldorf	Viersen	sehr zentral	Städt

In [140…
```
len(df_urban_class_new)
```

Out[140]:
```
8319
```

In [141…
```
df_urban_class_new['POSTAL_CODE'] = df_urban_class_new['POSTAL_CODE'].astype('str')
```

In [142…
```
df_final2 = pd.merge(df_final, df_urban_class_new, left_on = ['postal_code'], right_or
df_final2.head()
```

Out[142]:

	no_cars	gender	age	income	political_party	postal_code	CO2_housing	CO2_electricity	CO2_h
0	1	Männlich	36	2500	SPD	39106	1487.5	390	
1	1	Männlich	60	1000	Bündnis 90/Die Grünen	30966	1400.4	39	
2	2	Männlich	57	6500	7	41812	1944	42	
3	2	Männlich	36	6500	Bündnis 90/Die Grünen	17034	607.95	3.85	
4	1	NaN	NaN	NaN	NaN	NaN	64.3076	26.32	

In [143…
```
df_final2.columns
```

Out[143]:
```
Index(['no_cars', 'gender', 'age', 'income', 'political_party', 'postal_code',
       'CO2_housing', 'CO2_electricity', 'CO2_housing_electricity',
       'CO2_cruise', 'CO2_flight', 'CO2_public_transport', 'CO2_car1',
       'CO2_car2', 'CO2_car3', 'CO2_car4', 'CO2_car5', 'CO2_car_total',
       'CO2_mobility', 'CO2_food', 'CO2_other_consumption', 'public_emission',
       'CO2_total', 'belief_housing_electricity', 'belief_mobility',
       'belief_food', 'belief_other_consumption', 'belief_total', 'engine_1',
       'engine_2', 'engine_3', 'engine_4', 'engine_5', 'engine_6', 'engine_7',
       'engine_8', 'engine_9', 'engine_10', 'batch', 'Q5.1', 'education',
       'Unnamed: 0_x', 'NUTS3_ID_x', 'NUTS1_NAME_x', 'NUTS2_NAME_x',
       'NUTS3_NAME_x', 'POSTAL_CODE_x', 'EUROSTAT', 'Unnamed: 0_y',
       'POSTAL_CODE_y', 'NUTS3_ID_y', 'NUTS1_NAME_y', 'NUTS2_NAME_y',
       'NUTS3_NAME_y', 'RLK2022', 'KTU2022'],
      dtype='object')
```

In [144…
```
df_final2 = df_final2[['no_cars', 'gender', 'age', 'income', 'political_party', 'educa
                       'EUROSTAT', 'RLK2022', 'KTU2022', 'NUTS1_NAME_x', 'NUTS2_NAME_x',
                       'CO2_housing', 'CO2_electricity', 'CO2_housing_electricity', 'CO2
       'CO2_car3', 'CO2_car4', 'CO2_car5', 'CO2_car_total', 'CO2_mobility',
       'CO2_food', 'CO2_other_consumption', 'public_emission', 'CO2_total', 'belief_hc
                       'belief_mobility', 'belief_food','belief_other_consumption', 'bel
                        'engine_1',
       'engine_2', 'engine_3', 'engine_4', 'engine_5', 'engine_6', 'engine_7',
       'engine_8', 'engine_9', 'engine_10']]
```

In [145…
```
df_final2 = df_final2.rename(columns={'NUTS1_NAME_x':'NUTS1_NAME', 'NUTS2_NAME_x':'NUT
```

In [146…
```
df_final2.head(10)
```

Out[146]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK2(
0	1	Männlich	36	2500	SPD	(Fach-) Hochschulabschluss (Bachelor, Master, ...	39106	IN	zer
1	1	Männlich	60	1000	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	30966	PU	s zer
2	2	Männlich	57	6500	7	Berufsausbildung, Lehre oder Ausbildung an ein...	41812	PU	s zer
3	2	Männlich	36	6500	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	17034	NaN	s perip
4	1	NaN	NaN	NaN	NaN	None	NaN	NaN	N
5	2	Männlich	31	2500	AfD	Allgemeine oder fachgebundene Hochschulreife/A...	78054	IN	zer
6	0	Weiblich	63	1500	7	Realschulabschluss (Mittlere Reife) oder gleic...	42369	PU	s zer
7	1	Männlich	45	1500	CDU/CSU	Berufsausbildung, Lehre oder Ausbildung an ein...	01904	NaN	N
8	1	Weiblich	41	1900	7	Realschulabschluss (Mittlere Reife) oder gleic...	28237	PU	s zer
9	2	Weiblich	70	5400	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	12209	PU	s zer

Note 3: remove the unfinished surveys with null values

In [147…
```
len(df_final2)
```

Out[147]:
```
839
```

In [148…
```
## counting the number of missing values of the urban_rural_class
len(df_final2[df_final2['EUROSTAT'].isna()])
```

Out[148]:
```
201
```

In [149…
```
len(df_final2[df_final2['RLK2022'].isna()])
```

Out[149]:
```
179
```

In [150…
```
len(df_final2[df_final2['KTU2022'].isna()])
```

Out[150]:
```
179
```

In [151…
```
df_count = df_final2.dropna(subset = ['CO2_housing_electricity', 'CO2_mobility', 'CO2_
                                      'CO2_total', 'belief_housing_electricity', 'belie
       'belief_other_consumption', 'belief_total'], how = 'any')

len(df_count)
```

Out[151]:
```
709
```

In [152…
```
df_final2.columns
```

Out[152]:
```
Index(['no_cars', 'gender', 'age', 'income', 'political_party', 'education',
       'postal_code', 'EUROSTAT', 'RLK2022', 'KTU2022', 'NUTS1_NAME',
       'NUTS2_NAME', 'NUTS3_NAME', 'CO2_housing', 'CO2_electricity',
       'CO2_housing_electricity', 'CO2_cruise', 'CO2_flight',
       'CO2_public_transport', 'CO2_car1', 'CO2_car2', 'CO2_car3', 'CO2_car4',
       'CO2_car5', 'CO2_car_total', 'CO2_mobility', 'CO2_food',
       'CO2_other_consumption', 'public_emission', 'CO2_total',
       'belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total', 'batch', 'engine_1',
       'engine_2', 'engine_3', 'engine_4', 'engine_5', 'engine_6', 'engine_7',
       'engine_8', 'engine_9', 'engine_10'],
      dtype='object')
```

In [153…
```
### dropping the values for

df_final3 = df_final2.dropna(subset = ['age', 'income', 'political_party', 'education'
       'EUROSTAT', 'RLK2022', 'KTU2022','CO2_housing','CO2_electricity','CO2_housing_e
       'CO2_food', 'CO2_other_consumption', 'public_emission', 'CO2_total',
       'belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total'], how = 'any')
```

In [154…
```
len(df_final3)
```

Out[154]:
```
619
```

Note 4: calclating the gap between the "belief" and "actual"

In [155…
```
def flatten_extend(matrix):
...     flat_list = []
...     for row in matrix:
...         flat_list.extend(row)
...     return flat_list
```

In [156…
```
# Removing any rows that contains 'Invalid Expression'

columns_to_check = ['no_cars', 'gender','age', 'income', 'political_party', 'educatior
       'postal_code', 'EUROSTAT', 'RLK2022', 'KTU2022', 'NUTS1_NAME',
       'NUTS2_NAME', 'NUTS3_NAME', 'CO2_housing', 'CO2_electricity',
       'CO2_housing_electricity', 'CO2_cruise', 'CO2_flight',
       'CO2_public_transport', 'CO2_car1', 'CO2_car2', 'CO2_car3', 'CO2_car4',
       'CO2_car5', 'CO2_car_total', 'CO2_mobility', 'CO2_food',
       'CO2_other_consumption', 'public_emission', 'CO2_total',
```

```
       'belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total']

indices = []

for col in columns_to_check:
    indices.append(df_final3[df_final3[col].str.contains('Invalid Expression', na=Fals


row_to_remove = list(set(flatten_extend(indices)))
row_to_remove
```

Out[156]:
```
[616, 560, 576]
```

In [157…
```
df_final4 = df_final3.drop(row_to_remove)
```

In [158…
```
# Removing rows with invalid values for belief variable

columns_to_check = ['belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total']

indices = []

for col in columns_to_check:
    df_final4[col] = df_final4[col].astype('float')
    indices.append(df_final4[(df_final4[col]<0) | (df_final4[col].astype('float')>100)

row_to_remove2 = list(set(flatten_extend(indices)))
row_to_remove2
```

Out[158]:
```
[600]
```

In [159…
```
df_final4 = df_final4.drop(row_to_remove2)
```

In [160…
```
# Removing rows with invalid values for CO2 footprint

columns_to_check = ['CO2_housing','CO2_electricity','CO2_food']

indices = []

for col in columns_to_check:
    df_final4[col] = df_final4[col].astype('float')
    indices.append(df_final4[df_final4[col]==0].index.tolist())

row_to_remove3 = list(set(flatten_extend(indices)))
row_to_remove3
```

Out[160]:
```
[230]
```

In [161…
```
df_final4 = df_final4.drop(row_to_remove3)
```

In [162…
```
df_final4.columns
```

Out[162]:
```
Index(['no_cars', 'gender', 'age', 'income', 'political_party', 'education',
       'postal_code', 'EUROSTAT', 'RLK2022', 'KTU2022', 'NUTS1_NAME',
       'NUTS2_NAME', 'NUTS3_NAME', 'CO2_housing', 'CO2_electricity',
       'CO2_housing_electricity', 'CO2_cruise', 'CO2_flight',
       'CO2_public_transport', 'CO2_car1', 'CO2_car2', 'CO2_car3', 'CO2_car4',
       'CO2_car5', 'CO2_car_total', 'CO2_mobility', 'CO2_food',
       'CO2_other_consumption', 'public_emission', 'CO2_total',
       'belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total', 'batch', 'engine_1',
       'engine_2', 'engine_3', 'engine_4', 'engine_5', 'engine_6', 'engine_7',
       'engine_8', 'engine_9', 'engine_10'],
      dtype='object')
```

In [163…
```
# change the datatype to number
# only the following columns should be datatype object: gender,'political_party', educ

columns_to_change = ['no_cars', 'age', 'income',
       'CO2_housing', 'CO2_electricity', 'CO2_housing_electricity',
       'CO2_cruise', 'CO2_flight', 'CO2_public_transport', 'CO2_car1', 'CO2_car2',
       'CO2_car3', 'CO2_car4', 'CO2_car5', 'CO2_car_total', 'CO2_mobility',
       'CO2_food', 'CO2_other_consumption', 'public_emission', 'CO2_total',
       'belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total']


for col in columns_to_change:
    if col == 'age':
        df_final4[col] = df_final4[col].astype('Int64')

    else:
        df_final4[col] = df_final4[col].astype('float')
```

In [164…
```
# calculate the rank in the target areas
# tied groups: tied groups are given the lowest number as a rank

target_areas = ['CO2_housing_electricity', 'CO2_mobility', 'CO2_food', 'CO2_other_cons
target_areas2 = ['belief_housing_electricity', 'belief_mobility', 'belief_food','belie

# finding the rank in the total numbber of respondents
for col in target_areas:
    new_col = 'actual_rank_' + col + '1'
    df_final4[new_col] = df_final4[col].rank(ascending=False, method='min')

# calculating how many people have higher CF and scaling it for 101 people (101 people
# calculaing the rank in a group of 101 and subtract 1 to calculate the no. of people
for col in target_areas:
    old_col = 'actual_rank_' + col + '1' # rank in the total group
    new_col = 'actual_rank_' + col + '2' # it is called rank but actually how many pec
    #old: df_final4[new_col] = round(((df_final4[old_col]-1)/len(df_final4)) * 101, 0)

    # scaling the rank to the range 1 - 101, subtract 1 to find the no.of people with
    df_final4[new_col] = round(100*(df_final4[old_col] - df_final4[old_col].min())/ (d

# difference between the belief and the actual number of people who have higher number
for index, col in enumerate(target_areas):
    actual_col = 'actual_rank_' + col + '2'
    estimated_col = target_areas2[index]
    new_col = 'final_' + estimated_col
```

```
    df_final4[new_col] = df_final4[actual_col] - df_final4[estimated_col]
```

scaling formula used:

$$x_{normalized} = (b-a)\frac{x - min(x)}{max(x) - min(x)} + a$$

In [165…
```
df_final4.tail(5)
```

Out[165]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK
834	1.00	Weiblich	22	5000.00	SPD	Allgemeine oder fachgebundene Hochschulreife/A...	34233	IN	ze
835	1.00	Weiblich	26	7000.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	55252	PU	ze
836	0.00	Weiblich	30	1500.00	Die Linke	(Fach-) Hochschulabschluss (Bachelor, Master, ...	67454	IN	ze
837	3.00	Weiblich	20	6000.00	SPD	Allgemeine oder fachgebundene Hochschulreife/A...	35075	IN	ze
838	1.00	Weiblich	29	6000.00	FDP	(Fach-) Hochschulabschluss (Bachelor, Master, ...	82152	IN	ze

In [166…
```
# checking the range of the calculated values for rank and differences
df_final4.describe()
```

Out[166]:

	no_cars	age	income	CO2_housing	CO2_electricity	CO2_housing_electricity	CO2_cruise
count	614.00	614.00	614.00	614.00	614.00	614.00	614.00
mean	0.98	35.06	17439.94	1431.53	362.95	1794.48	1038.10
std	0.86	13.72	322879.45	1255.94	692.02	1502.52	7749.29
min	0.00	18.00	0.00	7.35	0.03	34.77	0.00
25%	0.00	25.00	1800.00	729.54	35.00	980.53	0.00
50%	1.00	31.00	3000.00	1146.42	381.52	1466.85	0.00
75%	1.00	42.00	4500.00	1906.34	487.50	2221.69	0.00
max	5.00	100.00	8000000.00	15030.96	11700.00	16188.51	98915.00

In [167…
```
df_final4.columns
```

Out[167]:
```
Index(['no_cars', 'gender', 'age', 'income', 'political_party', 'education',
       'postal_code', 'EUROSTAT', 'RLK2022', 'KTU2022', 'NUTS1_NAME',
       'NUTS2_NAME', 'NUTS3_NAME', 'CO2_housing', 'CO2_electricity',
       'CO2_housing_electricity', 'CO2_cruise', 'CO2_flight',
       'CO2_public_transport', 'CO2_car1', 'CO2_car2', 'CO2_car3', 'CO2_car4',
       'CO2_car5', 'CO2_car_total', 'CO2_mobility', 'CO2_food',
       'CO2_other_consumption', 'public_emission', 'CO2_total',
       'belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total', 'batch', 'engine_1',
       'engine_2', 'engine_3', 'engine_4', 'engine_5', 'engine_6', 'engine_7',
       'engine_8', 'engine_9', 'engine_10',
       'actual_rank_CO2_housing_electricity1', 'actual_rank_CO2_mobility1',
       'actual_rank_CO2_food1', 'actual_rank_CO2_other_consumption1',
       'actual_rank_CO2_total1', 'actual_rank_CO2_housing_electricity2',
       'actual_rank_CO2_mobility2', 'actual_rank_CO2_food2',
       'actual_rank_CO2_other_consumption2', 'actual_rank_CO2_total2',
       'final_belief_housing_electricity', 'final_belief_mobility',
       'final_belief_food', 'final_belief_other_consumption',
       'final_belief_total'],
      dtype='object')
```

In [168…
```
df_final4.head()
```

Out[168]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK2
25	1.00	Weiblich	65	3000.00	CDU/CSU	(Fach-) Hochschulabschluss (Bachelor, Master, ...	66440	PU	zer
26	2.00	Weiblich	59	800.00	8	Allgemeine oder fachgebundene Hochschulreife/A...	65933	PU	zer
27	0.00	Weiblich	60	1750.00	8	Berufsausbildung, Lehre oder Ausbildung an ein...	95028	IN	peri
28	1.00	Männlich	73	2500.00	SPD	Realschulabschluss (Mittlere Reife) oder gleic...	63741	IN	zer
30	0.00	Männlich	43	2500.00	7	Berufsausbildung, Lehre oder Ausbildung an ein...	13059	PU	zer

Note 5: clean the political party variables

In [169…
```
df_final4.columns
```

Out[169]:
```
Index(['no_cars', 'gender', 'age', 'income', 'political_party', 'education',
       'postal_code', 'EUROSTAT', 'RLK2022', 'KTU2022', 'NUTS1_NAME',
       'NUTS2_NAME', 'NUTS3_NAME', 'CO2_housing', 'CO2_electricity',
       'CO2_housing_electricity', 'CO2_cruise', 'CO2_flight',
       'CO2_public_transport', 'CO2_car1', 'CO2_car2', 'CO2_car3', 'CO2_car4',
       'CO2_car5', 'CO2_car_total', 'CO2_mobility', 'CO2_food',
       'CO2_other_consumption', 'public_emission', 'CO2_total',
       'belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total', 'batch', 'engine_1',
       'engine_2', 'engine_3', 'engine_4', 'engine_5', 'engine_6', 'engine_7',
       'engine_8', 'engine_9', 'engine_10',
       'actual_rank_CO2_housing_electricity1', 'actual_rank_CO2_mobility1',
       'actual_rank_CO2_food1', 'actual_rank_CO2_other_consumption1',
       'actual_rank_CO2_total1', 'actual_rank_CO2_housing_electricity2',
       'actual_rank_CO2_mobility2', 'actual_rank_CO2_food2',
       'actual_rank_CO2_other_consumption2', 'actual_rank_CO2_total2',
       'final_belief_housing_electricity', 'final_belief_mobility',
       'final_belief_food', 'final_belief_other_consumption',
       'final_belief_total'],
      dtype='object')
```

In [170…
```
## change the column names for the differences between the actual ranks and the estima

df_final4 = df_final4.rename(columns={'NUTS1_NAME':'federal_state',
       'final_belief_housing_electricity':'belief_diff_housing_electricity',
       'final_belief_mobility':'belief_diff_mobility',
       'final_belief_food':'belief_diff_food', 'final_belief_other_consumption':'belie
       'final_belief_total':'belief_diff_total'})
```

In [171…
```
df_final4['political_party'].unique()
```

Out[171]:
```
array(['CDU/CSU', '8', 'SPD', '7', 'FDP', 'Bündnis 90/Die Grünen', 'AfD',
       'Die Linke', 'Einer anderen Partei', 'Bündnis Sarah Wagenknecht'],
      dtype=object)
```

In [172…
```
## clean the variables for political party value 7 and 8

df_final4['political_party'].replace('7', 'Einer anderen Partei', inplace=True)
df_final4['political_party'].replace('8', 'Keine Angabe', inplace=True)
df_final4['political_party'].unique()
```

Out[172]:
```
array(['CDU/CSU', 'Keine Angabe', 'SPD', 'Einer anderen Partei', 'FDP',
       'Bündnis 90/Die Grünen', 'AfD', 'Die Linke',
       'Bündnis Sarah Wagenknecht'], dtype=object)
```

In [173…
```
df_final4.head()
```

Out[173]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK2
25	1.00	Weiblich	65	3000.00	CDU/CSU	(Fach-) Hochschulabschluss (Bachelor, Master, ...	66440	PU	zer
26	2.00	Weiblich	59	800.00	Keine Angabe	Allgemeine oder fachgebundene Hochschulreife/A...	65933	PU	zer
27	0.00	Weiblich	60	1750.00	Keine Angabe	Berufsausbildung, Lehre oder Ausbildung an ein...	95028	IN	peri
28	1.00	Männlich	73	2500.00	SPD	Realschulabschluss (Mittlere Reife) oder gleic...	63741	IN	zer
30	0.00	Männlich	43	2500.00	Einer anderen Partei	Berufsausbildung, Lehre oder Ausbildung an ein...	13059	PU	zer

In [174…
```
len(df_final4)
```

Out[174]:
614

In [175…
```
pd.options.display.float_format = '{:.0f}'.format
df_final4.describe()
```

Out[175]:

	no_cars	age	income	CO2_housing	CO2_electricity	CO2_housing_electricity	CO2_cruise	CO2
count	614	614	614	614	614	614	614	
mean	1	35	17440	1432	363	1794	1038	
std	1	14	322879	1256	692	1503	7749	
min	0	18	0	7	0	35	0	
25%	0	25	1800	730	35	981	0	
50%	1	31	3000	1146	382	1467	0	
75%	1	42	4500	1906	488	2222	0	
max	5	100	8000000	15031	11700	16189	98915	

Note 6: removing the outliers in the income column

In [176…
```
df_final4.sort_values(by=['income'], ascending=False).head(10)
```

Out[176]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RL
363	3	Männlich	19	8000000	AfD	Realschulabschluss (Mittlere Reife) oder gleic...	90587	PU	z
450	3	Männlich	27	194267	Bündnis Sarah Wagenknecht	(Fach-) Hochschulabschluss (Bachelor, Master, ...	50126	PU	z
723	1	Männlich	33	150000	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	59065	PU	z
456	3	Männlich	19	100000	SPD	Allgemeine oder fachgebundene Hochschulreife/A...	33102	IN	z
727	0	Männlich	27	60000	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	40221	PU	z
736	1	Weiblich	21	60000	Einer anderen Partei	Allgemeine oder fachgebundene Hochschulreife/A...	10589	PU	z
237	2	Weiblich	27	50000	Einer anderen Partei	Berufsausbildung, Lehre oder Ausbildung an ein...	19053	IN	pe
665	1	Männlich	35	40000	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	20249	PU	z
719	3	Männlich	19	30000	CDU/CSU	Allgemeine oder fachgebundene Hochschulreife/A...	85135	IN	pe
554	2	Männlich	18	20000	FDP	Allgemeine oder fachgebundene Hochschulreife/A...	47800	PU	z

In [177…

```
# checking the outliers in the bottom values
pd.options.display.float_format = '{:.2f}'.format
df_final4.sort_values(by=['income'], ascending=True).head(30)
```

Out[177]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK
591	2.00	Männlich	21	0.00	Bündnis Sarah Wagenknecht	Allgemeine oder fachgebundene Hochschulreife/A...	75031	IN	z
102	1.00	Männlich	47	0.00	Einer anderen Partei	Berufsausbildung, Lehre oder Ausbildung an ein...	74594	PR	pe
269	3.00	Männlich	54	0.00	CDU/CSU	(Fach-) Hochschulabschluss (Bachelor, Master, ...	73557	IN	z
293	0.00	Weiblich	23	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	35041	IN	z
98	2.00	Männlich	23	0.00	FDP	(Fach-) Hochschulabschluss (Bachelor, Master, ...	76593	IN	z
222	1.00	Weiblich	68	0.00	SPD	Allgemeine oder fachgebundene Hochschulreife/A...	44795	PU	z
95	1.00	Weiblich	53	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	79312	IN	z
582	0.00	Weiblich	25	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	35037	IN	z
213	2.00	Männlich	58	0.00	CDU/CSU	Realschulabschluss (Mittlere Reife) oder gleic...	97070	IN	z
525	0.00	Männlich	21	0.00	CDU/CSU	Allgemeine oder fachgebundene Hochschulreife/A...	64409	PU	z
270	1.00	Weiblich	18	0.00	Einer anderen Partei	Realschulabschluss (Mittlere Reife) oder gleic...	29225	IN	pe
34	2.00	Männlich	62	0.00	Keine Angabe	(Fach-) Hochschulabschluss (Bachelor, Master, ...	46149	PU	z
463	3.00	Männlich	23	0.00	Die Linke	(Fach-) Hochschulabschluss (Bachelor, Master, ...	71334	PU	z
208	1.00	Weiblich	22	1.20	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	99089	IN	z
349	1.00	Weiblich	29	2.00	CDU/CSU	Berufsausbildung, Lehre oder Ausbildung an ein...	45307	PU	z

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK
478	1.00	Weiblich	23	3.00	Die Linke	Allgemeine oder fachgebundene Hochschulreife/A...	48485	IN	z
479	0.00	Weiblich	32	6.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	80639	PU	z
607	1.00	Weiblich	26	12.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	86153	IN	z
563	1.00	Weiblich	30	90.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	80993	PU	z
66	1.00	Weiblich	67	100.00	Keine Angabe	(Fach-) Hochschulabschluss (Bachelor, Master, ...	22337	PU	z
730	0.00	Männlich	35	100.00	Einer anderen Partei	Allgemeine oder fachgebundene Hochschulreife/A...	97859	PR	pe
200	0.00	Weiblich	20	200.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	73098	IN	z
568	0.00	Männlich	38	200.00	Einer anderen Partei	Allgemeine oder fachgebundene Hochschulreife/A...	66709	IN	z
781	0.00	Weiblich	20	400.00	CDU/CSU	Allgemeine oder fachgebundene Hochschulreife/A...	34125	IN	z
815	0.00	Weiblich	24	400.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	97070	IN	z
749	0.00	Weiblich	22	400.00	Einer anderen Partei	Doktorgrad oder Habilitation	80804	PU	z
802	0.00	Weiblich	21	500.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	20535	PU	z
204	2.00	Weiblich	42	500.00	Einer anderen Partei	Realschulabschluss (Mittlere Reife) oder gleic...	26605	IN	pe
587	0.00	Weiblich	18	520.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	10243	PU	z
640	0.00	Männlich	23	530.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss	76131	PU	z

In [178…
```
## delete the rows where the income answered were 1.2, 2, 3, 6, 12 as these answers do
## the row indices of those data points are as follows: 208, 349, 478, 479, 607

df_final4 = df_final4.drop([208, 349, 478, 479, 607])
```

In [179…
```
## checking if the rows are dropped
pd.options.display.float_format = '{:.2f}'.format
df_final4.sort_values(by=['income'], ascending=True).head(30)
```

Out[179]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK
98	2.00	Männlich	23	0.00	FDP	(Fach-) Hochschulabschluss (Bachelor, Master, ...	76593	IN	z
222	1.00	Weiblich	68	0.00	SPD	Allgemeine oder fachgebundene Hochschulreife/A...	44795	PU	z
269	3.00	Männlich	54	0.00	CDU/CSU	(Fach-) Hochschulabschluss (Bachelor, Master, ...	73557	IN	z
270	1.00	Weiblich	18	0.00	Einer anderen Partei	Realschulabschluss (Mittlere Reife) oder gleic...	29225	IN	per
293	0.00	Weiblich	23	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	35041	IN	z
102	1.00	Männlich	47	0.00	Einer anderen Partei	Berufsausbildung, Lehre oder Ausbildung an ein...	74594	PR	per
591	2.00	Männlich	21	0.00	Bündnis Sarah Wagenknecht	Allgemeine oder fachgebundene Hochschulreife/A...	75031	IN	z
95	1.00	Weiblich	53	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	79312	IN	z
582	0.00	Weiblich	25	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	35037	IN	z
525	0.00	Männlich	21	0.00	CDU/CSU	Allgemeine oder fachgebundene Hochschulreife/A...	64409	PU	z
213	2.00	Männlich	58	0.00	CDU/CSU	Realschulabschluss (Mittlere Reife) oder gleic...	97070	IN	z
34	2.00	Männlich	62	0.00	Keine Angabe	(Fach-) Hochschulabschluss (Bachelor, Master, ...	46149	PU	z
463	3.00	Männlich	23	0.00	Die Linke	(Fach-) Hochschulabschluss (Bachelor, Master, ...	71334	PU	z
563	1.00	Weiblich	30	90.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	80993	PU	z
730	0.00	Männlich	35	100.00	Einer anderen Partei	Allgemeine oder fachgebundene Hochschulreife/A...	97859	PR	per

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK
66	1.00	Weiblich	67	100.00	Keine Angabe	(Fach-) Hochschulabschluss (Bachelor, Master, ...	22337	PU	z
568	0.00	Männlich	38	200.00	Einer anderen Partei	Allgemeine oder fachgebundene Hochschulreife/A...	66709	IN	z
200	0.00	Weiblich	20	200.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	73098	IN	z
781	0.00	Weiblich	20	400.00	CDU/CSU	Allgemeine oder fachgebundene Hochschulreife/A...	34125	IN	z
815	0.00	Weiblich	24	400.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	97070	IN	z
749	0.00	Weiblich	22	400.00	Einer anderen Partei	Doktorgrad oder Habilitation	80804	PU	z
802	0.00	Weiblich	21	500.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	20535	PU	z
204	2.00	Weiblich	42	500.00	Einer anderen Partei	Realschulabschluss (Mittlere Reife) oder gleic...	26605	IN	pe
587	0.00	Weiblich	18	520.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	10243	PU	z
640	0.00	Männlich	23	530.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	76131	PU	z
817	0.00	Weiblich	27	538.00	FDP	(Fach-) Hochschulabschluss (Bachelor, Master, ...	69181	PU	z
559	0.00	Männlich	66	560.00	Bündnis Sarah Wagenknecht	(Fach-) Hochschulabschluss (Bachelor, Master, ...	46348	IN	z
614	0.00	Weiblich	25	570.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	72072	IN	z
373	0.00	Weiblich	22	600.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	15230	IN	pe
32	1.00	Weiblich	57	600.00	CDU/CSU	Realschulabschluss (Mittlere Reife) oder gleic...	78244	IN	z

In [180…
```
df_final4['income'].median()
```

Out[180]:
```
3000.0
```

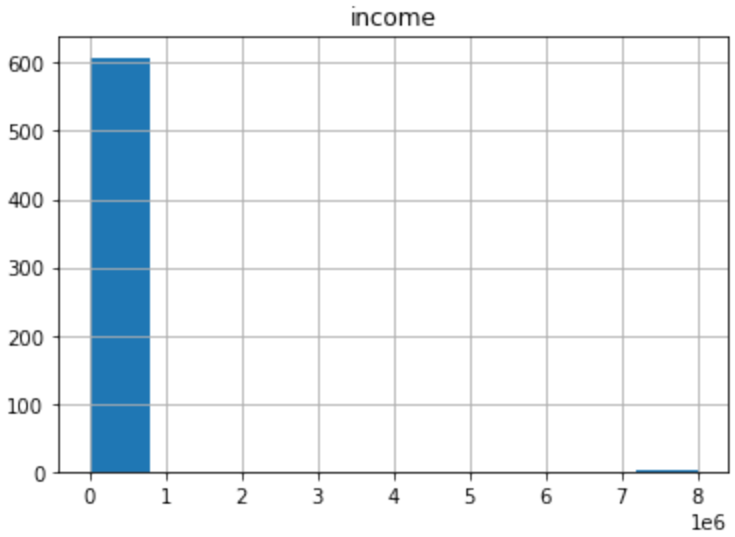
In [181…
```
# we will check the top 1%

df_final4['income'].quantile(.99)
```

Out[181]:
```
49199.99999999959
```

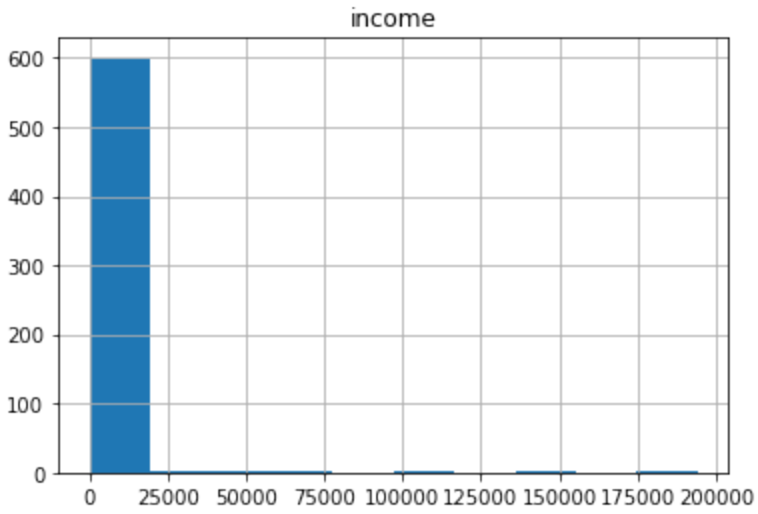
In [182…
```
df_final4.hist(column='income')
```

Out[182]:
```
array([[<AxesSubplot:title={'center':'income'}>]], dtype=object)
```



In [183…
```
df_var_income_ol = df_final4[df_final4['income']<1*(10**6)]
df_var_income_ol.hist(column='income')
```

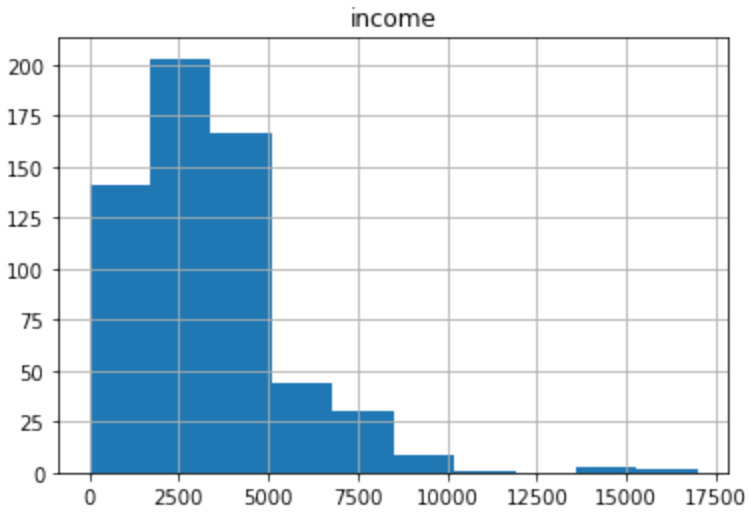
Out[183]:
```
array([[<AxesSubplot:title={'center':'income'}>]], dtype=object)
```



In [184…
```
# right-skewed distribution

df_var_income_ol2 = df_final4[df_final4['income']<20000]
df_var_income_ol2.hist(column='income')
```

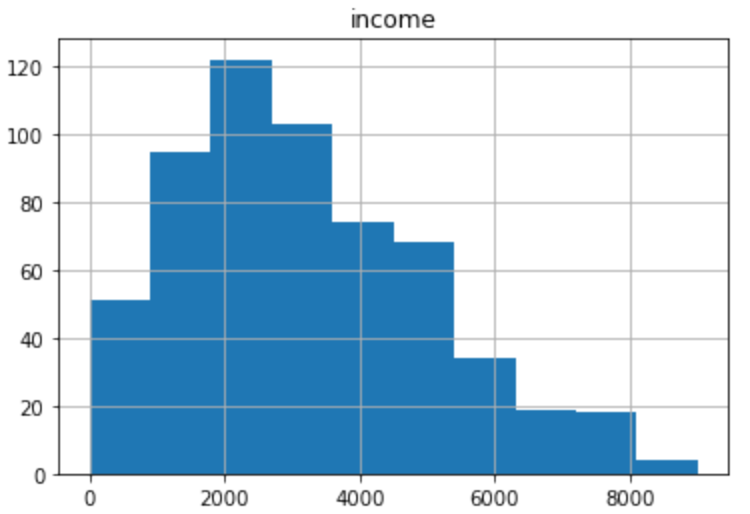
Out[184]:
```
array([[<AxesSubplot:title={'center':'income'}>]], dtype=object)
```



In [185…
```
df_var_income_ol3 = df_final4[df_final4['income']<10000]

df_var_income_ol3.hist(column='income')
```

Out[185]:
```
array([[<AxesSubplot:title={'center':'income'}>]], dtype=object)
```



In [186…
```
# high income class, outliers of more than EUR 10,000 net monthly income will be omitt

df_final5 = df_final4[df_final4['income']<10000]
df_final5.sort_values(by=['income'], ascending=False).head()
```

Out[186]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK
673	1.00	Weiblich	27	9000.00	FDP	(Fach-) Hochschulabschluss (Bachelor, Master, ...	80805	PU	ze
579	0.00	Weiblich	34	9000.00	Einer anderen Partei	Realschulabschluss (Mittlere Reife) oder gleic...	10557	PU	ze
471	1.00	Weiblich	42	8500.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	81375	PU	ze
294	3.00	Weiblich	22	8500.00	FDP	(Fach-) Hochschulabschluss (Bachelor, Master, ...	46562	PU	ze
634	1.00	Weiblich	19	8000.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	72124	IN	ze

In [187…

```
df_final5.sort_values(by=['income'], ascending=True).head(30)
```

Out[187]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK
102	1.00	Männlich	47	0.00	Einer anderen Partei	Berufsausbildung, Lehre oder Ausbildung an ein...	74594	PR	per
222	1.00	Weiblich	68	0.00	SPD	Allgemeine oder fachgebundene Hochschulreife/A...	44795	PU	z
463	3.00	Männlich	23	0.00	Die Linke	(Fach-) Hochschulabschluss (Bachelor, Master, ...	71334	PU	z
213	2.00	Männlich	58	0.00	CDU/CSU	Realschulabschluss (Mittlere Reife) oder gleic...	97070	IN	z
525	0.00	Männlich	21	0.00	CDU/CSU	Allgemeine oder fachgebundene Hochschulreife/A...	64409	PU	z
98	2.00	Männlich	23	0.00	FDP	(Fach-) Hochschulabschluss (Bachelor, Master, ...	76593	IN	z
582	0.00	Weiblich	25	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	35037	IN	z
591	2.00	Männlich	21	0.00	Bündnis Sarah Wagenknecht	Allgemeine oder fachgebundene Hochschulreife/A...	75031	IN	z
95	1.00	Weiblich	53	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	79312	IN	z
293	0.00	Weiblich	23	0.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	35041	IN	z
270	1.00	Weiblich	18	0.00	Einer anderen Partei	Realschulabschluss (Mittlere Reife) oder gleic...	29225	IN	per
269	3.00	Männlich	54	0.00	CDU/CSU	(Fach-) Hochschulabschluss (Bachelor, Master, ...	73557	IN	z
34	2.00	Männlich	62	0.00	Keine Angabe	(Fach-) Hochschulabschluss (Bachelor, Master, ...	46149	PU	z
563	1.00	Weiblich	30	90.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	80993	PU	z
66	1.00	Weiblich	67	100.00	Keine Angabe	(Fach-) Hochschulabschluss (Bachelor, Master, ...	22337	PU	z

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK
730	0.00	Männlich	35	100.00	Einer anderen Partei	Allgemeine oder fachgebundene Hochschulreife/A...	97859	PR	per
568	0.00	Männlich	38	200.00	Einer anderen Partei	Allgemeine oder fachgebundene Hochschulreife/A...	66709	IN	z
200	0.00	Weiblich	20	200.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	73098	IN	z
815	0.00	Weiblich	24	400.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	97070	IN	z
781	0.00	Weiblich	20	400.00	CDU/CSU	Allgemeine oder fachgebundene Hochschulreife/A...	34125	IN	z
749	0.00	Weiblich	22	400.00	Einer anderen Partei	Doktorgrad oder Habilitation	80804	PU	z
204	2.00	Weiblich	42	500.00	Einer anderen Partei	Realschulabschluss (Mittlere Reife) oder gleic...	26605	IN	per
802	0.00	Weiblich	21	500.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	20535	PU	z
587	0.00	Weiblich	18	520.00	Bündnis 90/Die Grünen	Allgemeine oder fachgebundene Hochschulreife/A...	10243	PU	z
640	0.00	Männlich	23	530.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	76131	PU	z
817	0.00	Weiblich	27	538.00	FDP	(Fach-) Hochschulabschluss (Bachelor, Master, ...	69181	PU	z
559	0.00	Männlich	66	560.00	Bündnis Sarah Wagenknecht	(Fach-) Hochschulabschluss (Bachelor, Master, ...	46348	IN	z
614	0.00	Weiblich	25	570.00	Bündnis 90/Die Grünen	(Fach-) Hochschulabschluss (Bachelor, Master, ...	72072	IN	z
199	3.00	Männlich	22	600.00	FDP	Allgemeine oder fachgebundene Hochschulreife/A...	67227	PU	z
32	1.00	Weiblich	57	600.00	CDU/CSU	Realschulabschluss (Mittlere Reife) oder gleic...	78244	IN	z

In [188…
```
len(df_final5)
```

Out[188]:
```
588
```

In [189…
```
df_final5.columns
```

Out[189]:
```
Index(['no_cars', 'gender', 'age', 'income', 'political_party', 'education',
       'postal_code', 'EUROSTAT', 'RLK2022', 'KTU2022', 'federal_state',
       'NUTS2_NAME', 'NUTS3_NAME', 'CO2_housing', 'CO2_electricity',
       'CO2_housing_electricity', 'CO2_cruise', 'CO2_flight',
       'CO2_public_transport', 'CO2_car1', 'CO2_car2', 'CO2_car3', 'CO2_car4',
       'CO2_car5', 'CO2_car_total', 'CO2_mobility', 'CO2_food',
       'CO2_other_consumption', 'public_emission', 'CO2_total',
       'belief_housing_electricity', 'belief_mobility', 'belief_food',
       'belief_other_consumption', 'belief_total', 'batch', 'engine_1',
       'engine_2', 'engine_3', 'engine_4', 'engine_5', 'engine_6', 'engine_7',
       'engine_8', 'engine_9', 'engine_10',
       'actual_rank_CO2_housing_electricity1', 'actual_rank_CO2_mobility1',
       'actual_rank_CO2_food1', 'actual_rank_CO2_other_consumption1',
       'actual_rank_CO2_total1', 'actual_rank_CO2_housing_electricity2',
       'actual_rank_CO2_mobility2', 'actual_rank_CO2_food2',
       'actual_rank_CO2_other_consumption2', 'actual_rank_CO2_total2',
       'belief_diff_housing_electricity', 'belief_diff_mobility',
       'belief_diff_food', 'belief_diff_other_consumption',
       'belief_diff_total'],
      dtype='object')
```

In [190…
```
### Columns in the final data for R
df_final_save = df_final5[['age', 'income', 'political_party', 'education',
                     'EUROSTAT', 'RLK2022', 'KTU2022', 'federal_state',
                     'CO2_housing', 'CO2_electricity', 'CO2_housing_electricity', 'CO2
       'CO2_car3', 'CO2_car4', 'CO2_car5', 'CO2_car_total', 'CO2_mobility',
       'CO2_food', 'CO2_other_consumption', 'public_emission', 'CO2_total',
                          'belief_diff_housing_electricity', 'belief_diff_mobility',
       'belief_diff_food', 'belief_diff_other_consumption',
       'belief_diff_total', 'batch', 'engine_1',
       'engine_2', 'engine_3', 'engine_4', 'engine_5', 'engine_6', 'engine_7',
       'engine_8', 'engine_9', 'engine_10']]
```

In [191…
```
df_final_save.head()
```

Out[191]:

	age	income	political_party	education	EUROSTAT	RLK2022	KTU2022	feder
25	65	3000.00	CDU/CSU	(Fach-) Hochschulabschluss (Bachelor, Master, ...	PU	zentral	Städtischer Kreis	S
26	59	800.00	Keine Angabe	Allgemeine oder fachgebundene Hochschulreife/A...	PU	sehr zentral	kreisfreie Großstadt	
27	60	1750.00	Keine Angabe	Berufsausbildung, Lehre oder Ausbildung an ein...	IN	peripher	Ländlicher Kreis mit Verdichtungsansätzen	
28	73	2500.00	SPD	Realschulabschluss (Mittlere Reife) oder gleic...	IN	sehr zentral	Städtischer Kreis	
30	43	2500.00	Einer anderen Partei	Berufsausbildung, Lehre oder Ausbildung an ein...	PU	sehr zentral	kreisfreie Großstadt	

In [192…

```
### save the cleaned data for Regression

df_final_save.to_csv('data_cleaned_R_final_batch_engine.csv')
```

In [193…

```
#### Continue with the exploratory data analysis with: df_final2

df_final5.head()
```

Out[193]:

	no_cars	gender	age	income	political_party	education	postal_code	EUROSTAT	RLK2
25	1.00	Weiblich	65	3000.00	CDU/CSU	(Fach-) Hochschulabschluss (Bachelor, Master, ...	66440	PU	zer
26	2.00	Weiblich	59	800.00	Keine Angabe	Allgemeine oder fachgebundene Hochschulreife/A...	65933	PU	zer
27	0.00	Weiblich	60	1750.00	Keine Angabe	Berufsausbildung, Lehre oder Ausbildung an ein...	95028	IN	peri
28	1.00	Männlich	73	2500.00	SPD	Realschulabschluss (Mittlere Reife) oder gleic...	63741	IN	zer
30	0.00	Männlich	43	2500.00	Einer anderen Partei	Berufsausbildung, Lehre oder Ausbildung an ein...	13059	PU	zer

In [194…

```
df_final5.to_csv('data_cleaned_descriptive_analysis_final_batch_engine.csv')
```

Comparing the 1, 2, 3 batches

- columns to compare: 'CO2_mobility', 'CO2_flight', 'CO2_car_total'

In [195…
```
df_final5.groupby(['batch'])['batch'].count()
```

Out[195]:
```
batch
1    254
2    163
3    171
Name: batch, dtype: int64
```

In [196…
```
round(100*171/588,2)
```

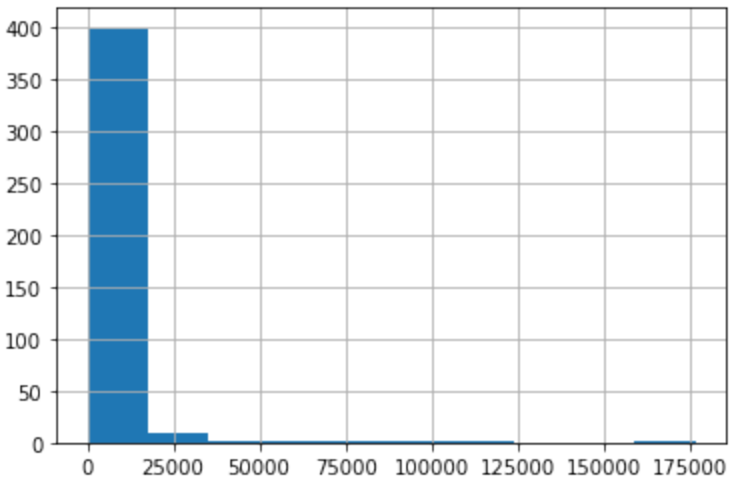
Out[196]:
```
29.08
```

In [197…
```
batch12 = df_final5[(df_final5['batch'] == 1) | (df_final5['batch'] == 2)]
batch3 = df_final5[df_final5['batch'] == 3]
print(len(batch12))
print(len(batch3))
```

```
417
171
```

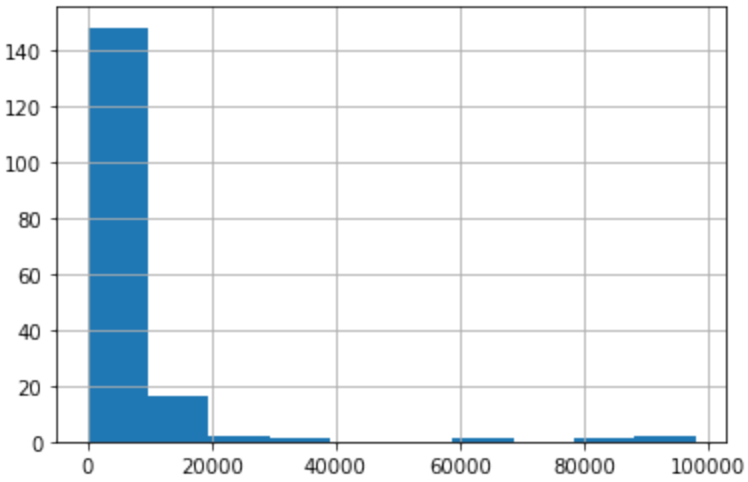
In [198…
```
batch12['CO2_mobility'].hist()
```

Out[198]:
```
<AxesSubplot:>
```



In [199…
```
batch3['CO2_mobility'].hist()
```

Out[199]:
```
<AxesSubplot:>
```



In [200…
```
batch12['CO2_mobility'].describe()
```

Out[200]:
```
count      417.00
mean      6344.51
std      15647.53
min          0.00
25%       1113.36
50%       3215.81
75%       6207.80
max     176807.10
Name: CO2_mobility, dtype: float64
```

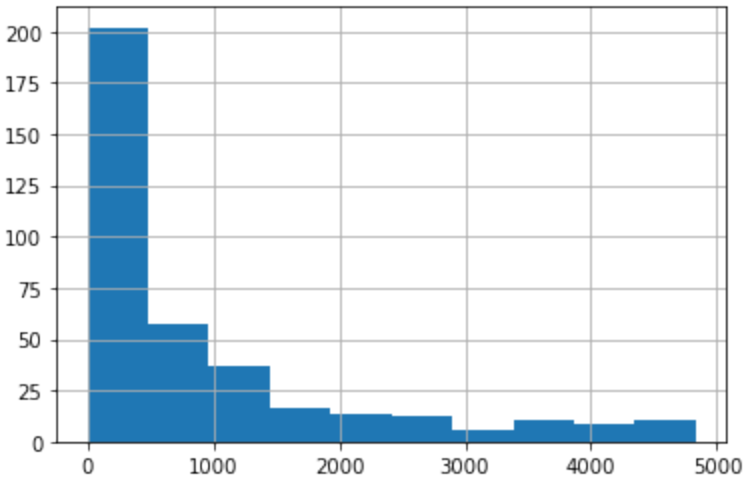
In [201…
```
batch3['CO2_mobility'].describe()
```

Out[201]:
```
count     171.00
mean     6127.61
std     13118.98
min         0.00
25%      1035.31
50%      3366.98
75%      5733.45
max     97951.30
Name: CO2_mobility, dtype: float64
```

Flights

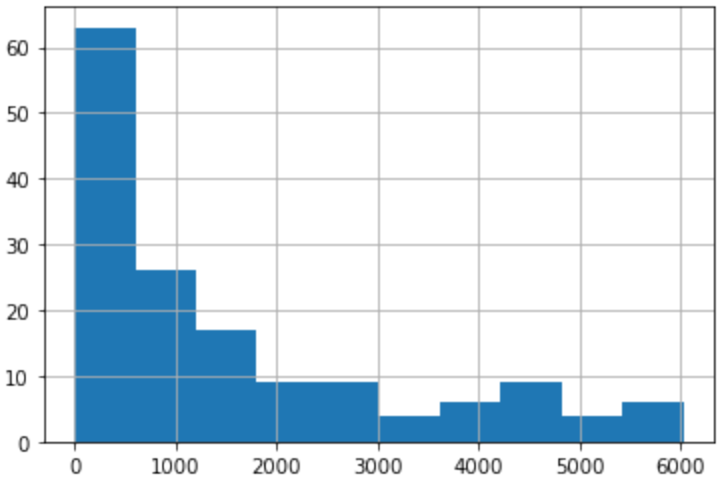
In [202…
```
hist_data = batch12[batch12['CO2_flight'] < batch12['CO2_flight'].quantile(0.9)]
hist_data['CO2_flight'].hist()
```

Out[202]:
```
<AxesSubplot:>
```



In [203…
```
hist_data = batch3[batch3['CO2_flight'] < batch3['CO2_flight'].quantile(0.9)]
hist_data['CO2_flight'].hist()
```

Out[203]:
```
<AxesSubplot:>
```



In [204…
```
# Creating the two different dataset

data1 = batch12[batch12['CO2_flight'] < batch12['CO2_flight'].quantile(0.9)]
data2 = batch3[batch3['CO2_flight'] < batch3['CO2_flight'].quantile(0.9)]

batch12_flight = data1['CO2_flight'] # work + personal

batch3_flight = data2['CO2_flight'] #only work related
```

In [205…
```
batch12_flight.describe()
```

Out[205]:
```
count    374.00
mean     852.60
std     1247.85
min        0.00
25%        0.00
50%      266.00
75%     1204.00
max     4832.00
Name: CO2_flight, dtype: float64
```

In [206…
```
batch3_flight.describe()
```

Out[206]:
```
count    153.00
mean    1498.18
std     1742.59
min        0.00
25%        0.00
50%      798.00
75%     2408.00
max     6031.00
Name: CO2_flight, dtype: float64
```

one-way ANOVA

In [207…
```
## ONE-way ANOVA
## H0 (null hypothesis): μ1 = μ2 = μ3 = … = μk (It implies that the means of all the p
## H1 (null hypothesis): It states that there will be at least one population mean tha

from scipy.stats import f_oneway
f_oneway(batch12_flight, batch3_flight) ### reject H0,
```

Out[207]:
```
F_onewayResult(statistic=22.791980956316962, pvalue=2.3443052774260043e-06)
```

In [208…
```
# fail to reject H0: the means of all the population are equal
f_oneway(batch12['CO2_flight'], batch3['CO2_flight'])
```

Out[208]:
```
F_onewayResult(statistic=2.015368436415039, pvalue=0.15624520160985503)
```

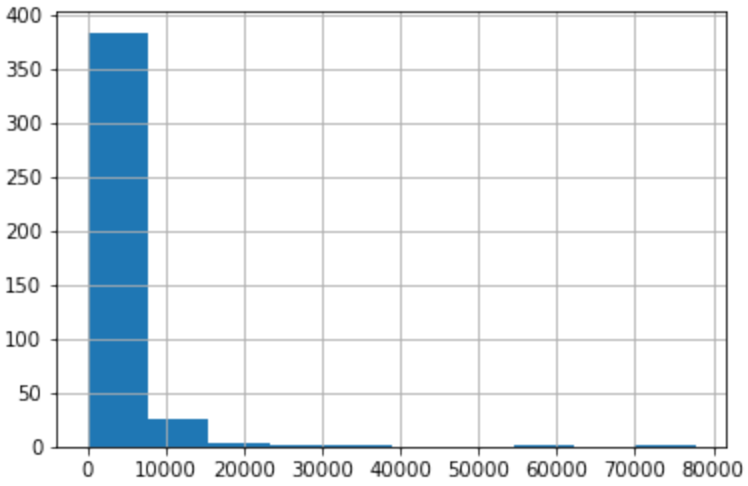
In [209…
```
# fail to reject H0: the means of all the population are equal
f_oneway(batch12['CO2_car_total'], batch3['CO2_car_total'])
```

Out[209]:
```
F_onewayResult(statistic=2.2739860571069284, pvalue=0.13210005471590572)
```

In [210…
```
##### Cars
```

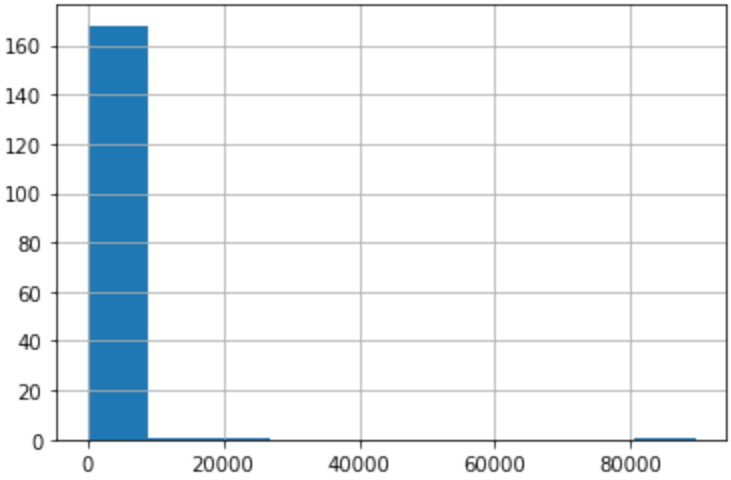
In [211…
```
batch12['CO2_car_total'].hist()
```

Out[211]:
```
<AxesSubplot:>
```



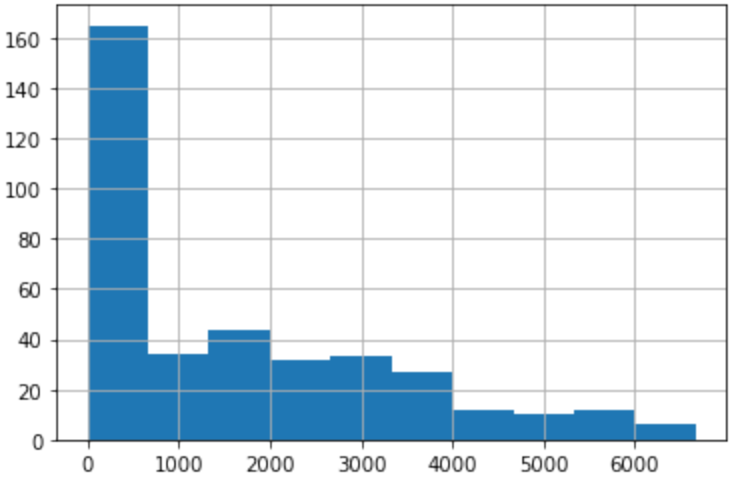
In [212…
```
batch3['CO2_car_total'].hist()
```

Out[212]:
```
<AxesSubplot:>
```



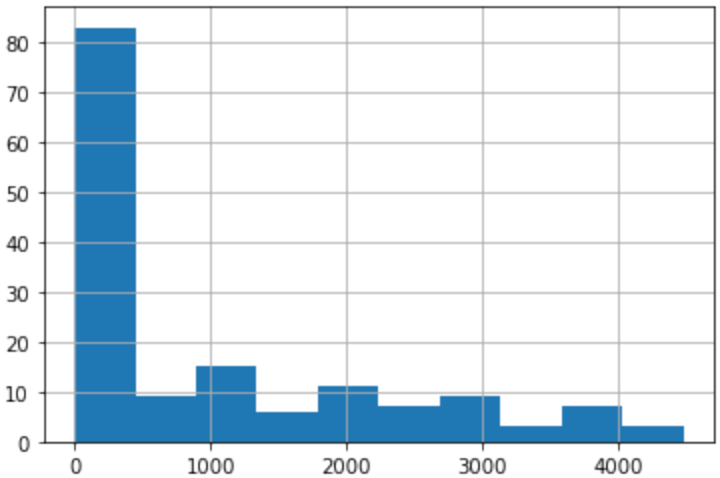
In [213…
```
hist_data = batch12[batch12['CO2_car_total'] < batch12['CO2_car_total'].quantile(0.9)]
hist_data['CO2_car_total'].hist()
```

Out[213]: <AxesSubplot:>



In [214…
```
hist_data = batch3[batch3['CO2_car_total'] < batch3['CO2_car_total'].quantile(0.9)]
hist_data['CO2_car_total'].hist()
```

Out[214]: <AxesSubplot:>



In [215…
```
# Creating the two different dataset

data1 = batch12[batch12['CO2_car_total'] < batch12['CO2_car_total'].quantile(0.9)]
data2 = batch3[batch3['CO2_car_total'] < batch3['CO2_car_total'].quantile(0.9)]

batch12_car = data1['CO2_car_total'] # work + personal

batch3_car = data2['CO2_car_total'] #only work related
```

In [216…
```
f_oneway(batch12_car, batch3_car)
```

Out[216]:
```
F_onewayResult(statistic=17.259753671897645, pvalue=3.804454793665776e-05)
```

In [ ]:
```

```