


▼ **Seatwork 6.1 Exploratory Data Analysis on Your Own Dataset**

Dataset: Food Waste

```
filepath = '/content/Food Waste data and research - by country.csv'
```

```
import pandas as pd
import numpy as np
```

```
FoodWaste = pd.read_csv(filepath)
FoodWaste
```



	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year)	Retail estimate (tonnes/year)	Food service estimate (kg/capita/year)	Food service estimate (tonnes/year)	Confidence in estimate
0	Afghanistan	126	82	3109153	16	594982	28	1051783	Country
1	Albania	127	83	238492	16	45058	28	79651	Country
2	Algeria	135	91	3918529	16	673360	28	1190335	Country
3	Andorra	123	84	6497	13	988	26	1971	Country
4	Angola	144	100	3169523	16	497755	28	879908	Country
...
209	Venezuela (Boliv. Rep. of)	116	72	2065461	16	445994	28	788407	Country
210	Viet Nam	120	76	7346717	16	1508689	28	2666991	Country
211	Yemen	148	104	3026946	16	456099	28	806270	Country
212	Zambia	122	78	1391729	16	279350	28	493822	Country
213	Zimbabwe	144	100	1458564	16	229059	28	404920	Country

214 rows × 12 columns

```
# Identify the column names
FoodWaste.columns

Index(['Country', 'combined figures (kg/capita/year)',
      'Household estimate (kg/capita/year)',
      'Household estimate (tonnes/year)', 'Retail estimate (kg/capita/year)',
      'Retail estimate (tonnes/year)',
      'Food service estimate (kg/capita/year)',
      'Food service estimate (tonnes/year)', 'Confidence in estimate',
      'M49 code', 'Region', 'Source'],
      dtype='object')
```

```
# Identify the data types of the data
FoodWaste.dtypes

Country                object
combined figures (kg/capita/year)  int64
Household estimate (kg/capita/year)  int64
Household estimate (tonnes/year)    int64
Retail estimate (kg/capita/year)    int64
Retail estimate (tonnes/year)       int64
```

```
Food service estimate (kg/capita/year)    int64
Food service estimate (tonnes/year)       int64
Confidence in estimate                   object
M49 code                                 int64
Region                                   object
Source                                   object
dtype: object
```

```
#Display the total number of records
print('The total number of records:', len(FoodWaste))
```

The total number of records: 214

```
#Display the first 20 records
FoodWaste.head(20)
```

	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year)
0	Afghanistan	126	82	3109153	16
1	Albania	127	83	238492	16
2	Algeria	135	91	3918529	16
3	Andorra	123	84	6497	13
4	Angola	144	100	3169523	16
5	Antigua and Barbuda	113	74	7178	13
6	Argentina	116	72	3243563	16
7	Armenia	137	93	275195	16
8	Aruba	113	74	7858	13
9	Australia	133	102	2563110	9
10	Austria	76	39	349249	9
11	Azerbaijan	137	93	934872	16
12	Bahamas	113	74	28792	13
13	Bahrain	171	132	216161	13
14	Bangladesh	84	65	10618233	16

```
# Display the last 20 records
FoodWaste.tail(20)
```

	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year)
194	Tunisia	135	91	1064407	16
195	Turkey	137	93	7762575	16
196	Turkmenistan	120	76	449895	16
197	Turks and Caicos Islands	113	74	2824	13
198	Tuvalu	120	76	878	16
199	Uganda	147	103	4546237	16
200	Ukraine	120	76	3344904	16
201	United Arab Emirates	134	95	923625	13
202	United Kingdom	98	77	5199825	4
203	United Rep. of Tanzania	163	119	6907649	16
204	United States of America	139	59	19359951	16
205	United States Virgin Islands	113	74	7732	13
206	Uruguay	113	74	255892	13
207	Uzbekistan	135	91	3001868	16
208	Vanuatu	135	91	27296	16
209	Venezuela (Bella Dora)	113	74	2225464	16

```
# Change the Confidence in estimate column to Certainty
FoodWaste.rename(columns = {'Confidence in estimate':'Certainty'}, inplace = True)
FoodWaste
```

	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year)	(t
0	Afghanistan	126	82	3109153	16	
1	Albania	127	83	238492	16	
2	Algeria	135	91	3918529	16	
3	Andorra	123	84	6497	13	
4	Angola	144	100	3169523	16	
...	
209	Venezuela (Boliv. Rep. of)	116	72	2065461	16	
210	Viet Nam	120	76	7346717	16	
211	Yemen	148	104	3026946	16	
212	Zambia	122	78	1391729	16	
213	Zimbabwe	144	100	1458564	16	

214 rows × 12 columns

```
# FoodWaste['Certainty'] contains the confidence levels
conditions = [
    (FoodWaste['Certainty'] == 'Very Low Confidence'),
    (FoodWaste['Certainty'] == 'Low Confidence'),
    (FoodWaste['Certainty'] == 'Medium Confidence'),
    (FoodWaste['Certainty'] == 'High Confidence')
]
values = [0, 1, 2, 3]

# Applying np.where with multiple conditions
FoodWaste['Classification'] = np.select(conditions, values, default=np.nan)
FoodWaste
```

	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year)	(t
0	Afghanistan	126	82	3109153	16	
1	Albania	127	83	238492	16	
2	Algeria	135	91	3918529	16	
3	Andorra	123	84	6497	13	
4	Angola	144	100	3169523	16	
...	
209	Venezuela (Boliv. Rep. of)	116	72	2065461	16	
210	Viet Nam	120	76	7346717	16	
211	Yemen	148	104	3026946	16	
212	Zambia	122	78	1391729	16	
213	Zimbabwe	144	100	1458564	16	

214 rows × 13 columns

```
# Create a new dataframe "foodwaste_data" that gathers data with Low Confidence
new_foodwaste_data = pd.DataFrame(FoodWaste)
new_foodwaste_data = FoodWaste[FoodWaste['Classification'] == 0].copy()
new_foodwaste_data
```

	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year) (t
0	Afghanistan	126	82	3109153	16
1	Albania	127	83	238492	16
2	Algeria	135	91	3918529	16
4	Angola	144	100	3169523	16
6	Argentina	116	72	3243563	16
...
209	Venezuela (Boliv. Rep. of)	116	72	2065461	16
210	Viet Nam	120	76	7346717	16
211	Yemen	148	104	3026946	16
212	Zambia	122	78	1391729	16
213	Zimbabwe	144	100	1458564	16

130 rows × 13 columns

```
#top 10 low confidence
top_10_foodwaste = new_foodwaste_data.head(10)
print('Top 10 records:', top_10_foodwaste)
```

Top 10 records:	Country	combined figures (kg/capita/year)	\
0	Afghanistan	126	
1	Albania	127	
2	Algeria	135	
4	Angola	144	
6	Argentina	116	
7	Armenia	137	
11	Azerbaijan	137	
14	Bangladesh	84	
16	Belarus	112	
18	Belize	97	
	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	\
0	82	3109153	
1	83	238492	
2	91	3918529	
4	100	3169523	
6	72	3243563	
7	93	275195	
11	93	934872	
14	65	10618233	
16	68	646356	
18	53	20564	
	Retail estimate (kg/capita/year)	Retail estimate (tonnes/year)	\
0	16	594982	
1	16	45058	
2	16	673360	
4	16	497755	
6	16	700380	
7	16	46259	
11	16	157148	
14	16	2550080	

```
4/9/24, 11:55 PM Marquez_Keith_Seatwork_6_1_Exploratory_Data_Analysis_on_Your_Own_Dataset.ipynb - Colaboratory
```

16	16	147838
18	16	6106

	Food service estimate (kg/capita/year)	\
0	28	
1	28	
2	28	
4	28	
6	28	
7	28	
11	28	
14	3	
16	28	
18	28	

	Food service estimate (tonnes/year)	Certainty	M49 code	\
0	1051783	Very Low Confidence	4	
1	79651	Very Low Confidence	8	
2	1190335	Very Low Confidence	12	
4	879908	Very Low Confidence	24	
6	1238100	Very Low Confidence	32	
7	81775	Very Low Confidence	51	
11	277800	Very Low Confidence	31	
14	544436	Very Low Confidence	50	
16	261341	Very Low Confidence	112	

```
# Create a new dataframe "foodwaste_data" that gathers data with High Confidence
new_foodwaste_data = pd.DataFrame(FoodWaste)
new_foodwaste_data = FoodWaste[FoodWaste['Classification'] == 3].copy()
print('The total number of records:', len(new_foodwaste_data))
new_foodwaste_data
```

The total number of records: 10

	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year)	Retail estimate (tonnes/year)	Food service estimate (kg/capita/year)	Food service estimate (tonnes/year)	Certifi
9	Australia	133	102	2563110	9	238248	22	546340	Confic
10	Austria	76	39	349249	9	77289	28	254191	Confic
54	Denmark	132	81	469449	30	172003	21	119134	Confic
74	Germany	102	75	6263775	6	498244	21	1718433	Confic
96	Italy	97	67	4059806	4	219552	26	1548291	Confic
136	New Zealand	90	61	291769	3	14923	26	122306	Confic
167	Saudi Arabia	151	105	3594080	20	673502	26	876260	Confic
185	Sweden	112	81	812948	10	100364	21	205746	Confic
202	United Kingdom	98	77	5199825	4	283627	17	1114248	Confic
204	United States of America	139	59	19359951	16	5151313	64	20934827	Confic

```
#top 10 high confidence
top_10_foodwaste = new_foodwaste_data.head(10)
print('Top 10 records:', top_10_foodwaste)
```

Top 10 records:

	Country	combined figures (kg/capita/year)	\
9	Australia	133	
10	Austria	76	
54	Denmark	132	
74	Germany	102	
96	Italy	97	
136	New Zealand	90	
167	Saudi Arabia	151	
185	Sweden	112	

202	United Kingdom	98
204	United States of America	139

	Household estimate (kg/capita/year)	Household estimate (tonnes/year) \
9	102	2563110
10	39	349249
54	81	469449
74	75	6263775
96	67	4059806
136	61	291769
167	105	3594080
185	81	812948
202	77	5199825
204	59	19359951

	Retail estimate (kg/capita/year)	Retail estimate (tonnes/year) \
9	9	238248
10	9	77289
54	30	172003
74	6	498244
96	4	219552
136	3	14923
167	20	673502
185	10	100364
202	4	283627
204	16	5151313

	Food service estimate (kg/capita/year) \
9	22
10	28
54	21
74	21
96	26
136	26
167	26
185	21
202	17
204	64

	Food service estimate (tonnes/year)	Certainty	M49 code \
9	546340	High Confidence	36
10	254191	High Confidence	40
54	119134	High Confidence	208
74	1718433	High Confidence	276
96	1548291	High Confidence	380
136	122306	High Confidence	554
167	876260	High Confidence	682
185	205746	High Confidence	752
202	1114248	High Confidence	826

```
#Create a new dataframe "foodwaste_data" that gathers data with age less than 150
new_foodwaste_data = pd.DataFrame(FoodWaste)
new_foodwaste_data = FoodWaste[FoodWaste['combined figures (kg/capita/year)'] <= 150].copy()
print('The total number of records:', len(new_foodwaste_data))
new_foodwaste_data
```


The total number of records: 204

	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year)	Retail estimate (tonnes/year)	Food service estimate (kg/capita/year)	Food service estimate (tonnes/year)	Coi
0	Afghanistan	126	82	3109153	16	594982	28	1051783	V Coi
1	Albania	127	83	238492	16	45058	28	79651	V Coi
2	Algeria	135	91	3918529	16	673360	28	1190335	V Coi
3	Andorra	123	84	6497	13	988	26	1971	Coi
4	Angola	144	100	3169523	16	497755	28	879908	V Coi
...	
209	Venezuela (Boliv. Rep. of)	116	72	2065461	16	445994	28	788407	V Coi
210	Viet Nam	120	76	7346717	16	1508689	28	2666991	V Coi
211	Yemen	148	104	3026946	16	456099	28	806270	V Coi
212	Zambia	122	78	1391729	16	279350	28	493822	V Coi
213	Zimbabwe	144	100	1458564	16	229059	28	404920	V Coi

204 rows × 13 columns

```
#Create a new dataframe "foodwaste_data" that gathers data with age greater than 150
new_foodwaste_data = pd.DataFrame(FoodWaste)
new_foodwaste_data = FoodWaste[FoodWaste['combined figures (kg/capita/year)'] >= 150].copy()
print('The total number of records:', len(new_foodwaste_data))
new_foodwaste_data
```

The total number of records: 10

	Country	combined figures (kg/capita/year)	Household estimate (kg/capita/year)	Household estimate (tonnes/year)	Retail estimate (kg/capita/year)	(tonnes/year)
13	Bahrain	171	132	216161	13	

```
#Use numpy to get the mean Household Capital and Household Tonnes value.
mean_HCapital = np.mean(FoodWaste['Household estimate (kg/capita/year)'])
mean_HTonnes = np.mean(FoodWaste['Household estimate (tonnes/year)'])
print('Average Household estimate (kg/capita/year):', mean_HCapital)
print('Average Household estimate (tonnes/year):', mean_HTonnes)
```

```
Average Household estimate (kg/capita/year): 84.29439252336448
Average Household estimate (tonnes/year): 2658895.6542056073
```

```
#Use numpy to get the mean Retail Capital and Retail Tonnes value.
mean_RCapital = np.mean(FoodWaste['Retail estimate (kg/capita/year)'])
mean_RTonnes = np.mean(FoodWaste['Retail estimate (tonnes/year)'])
print('Average Retail estimate (kg/capita/year):', mean_RCapital)
print('Average Retail estimate (tonnes/year):', mean_RTonnes)
```

```
Average Retail estimate (kg/capita/year): 15.116822429906541
Average Retail estimate (tonnes/year): 552045.4299065421
```

```
#Use numpy to get the mean Food Capital and Food Tonnes value.
mean_FCapital = np.mean(FoodWaste['Food service estimate (kg/capita/year)'])
mean_FTonnes = np.mean(FoodWaste['Food service estimate (tonnes/year)'])
print('Average Food service estimate (kg/capita/year):', mean_FCapital)
print('Average Food service estimate (tonnes/year):', mean_FTonnes)
```

```
Average Food service estimate (kg/capita/year): 27.38317757009346
Average Food service estimate (tonnes/year): 1138859.2476635515
```

```
#Use numpy to get the median Household Capital and Household Tonnes value.
median_HCapital = np.median(FoodWaste['Household estimate (kg/capita/year)'])
median_HTonnes = np.median(FoodWaste['Household estimate (tonnes/year)'])
print('Median Household estimate (kg/capita/year):', median_HCapital)
print('Median Household estimate (tonnes/year):', median_HTonnes)
```

```
Median Household estimate (kg/capita/year): 80.0
Median Household estimate (tonnes/year): 520508.0
```

```
#Use numpy to get the median Retail Capital and Retail Tonnes value.
median_RCapital = np.median(FoodWaste['Retail estimate (kg/capita/year)'])
median_RTonnes = np.median(FoodWaste['Retail estimate (tonnes/year)'])
print('Median Retail estimate (kg/capita/year):', median_RCapital)
print('Median Retail estimate (tonnes/year):', median_RTonnes)
```

```
Median Retail estimate (kg/capita/year): 16.0
Median Retail estimate (tonnes/year): 100650.0
```

```
#Use numpy to get the median Food Capital and Food Tonnes value.
median_FCapital = np.median(FoodWaste['Food service estimate (kg/capita/year)'])
```