notebook

September 17, 2021

0.1 Energy saved from recycling

Did you know that recycling saves energy by reducing or eliminating the need to make materials from scratch? For example, aluminum can manufacturers can skip the energy-costly process of producing aluminum from ore by cleaning and melting recycled cans. Aluminum is classified as a non-ferrous metal.

Singapore has an ambitious goal of becoming a zero-waste nation. The amount of waste disposed of in Singapore has increased seven-fold over the last 40 years. At this rate, Semakau Landfill, Singapore's only landfill, will run out of space by 2035. Making matters worse, Singapore has limited land for building new incineration plants or landfills.

The government would like to motivate citizens by sharing the total energy that the combined recycling efforts have saved every year. They have asked you to help them.

You have been provided with three datasets. The data come from different teams, so the names of waste types may differ.

```
<div style="font-size:16px"><b>datasets/wastestats.csv - Recycling statistics per waste type fo
</div>
<div>Source: <a href="https://www.nea.gov.sg/our-services/waste-management/waste-statistics-and
```

waste_type: The type of waste recycled.

waste_disposed_of_tonne: The amount of waste that could not be recycled (in metric tonnes).

total_waste_recycle_tonne: The amount of waste that could be recycled (in metric tonnes).

total_waste_generated: The total amount of waste collected before recycling (in metric tonnes).

recycling_rate: The amount of waste recycled per tonne of waste generated.

year: The recycling year.

```
</div>
```

```
<div style="font-size:16px"><b>datasets/2018_2019_waste.csv - Recycling statistics per waste ty
</div>
<div> Source: <a href="https://www.nea.gov.sg/our-services/waste-management/waste-statistics-an
```

Waste Type: The type of waste recycled.

Total Generated: The total amount of waste collected before recycling (in thousands of metric tonnes).

Total Recycled: The amount of waste that could be recycled. (in thousands of metric tonnes).

Year: The recycling year.

<b>datasets/energy_saved.csv - Estimations of the amount of energy

material: The type of waste recycled.

energy_saved: An estimate of the energy saved (in kiloWatt hour) by recycling a metric tonne of waste.

crude_oil_saved: An estimate of the number of barrels of oil saved by recycling a metric tonne of waste.

[1]:
```
#Import Necessary Libraries
import os
from pathlib import Path
import re
import pandas as pd
import numpy as np
from itertools import combinations
```

[2]:
```
#Find data
find_data = [*os.scandir()]
find_data
```

[2]:
```
[<DirEntry '.ipynb_checkpoints'>,
 <DirEntry 'datasets'>,
 <DirEntry 'notebook.ipynb'>]
```

[3]:
```
datas = [*os.scandir('datasets')]
print(datas)
print(type(datas))
```

```
[<DirEntry '2018_2019_waste.csv'>, <DirEntry 'energy_saved.csv'>, <DirEntry
'wastestats.csv'>]
<class 'list'>
```

[4]:
```
#Import Data
waste_stats_older = pd.read_csv('datasets/wastestats.csv')
waste_stats_newer = pd.read_csv('datasets/2018_2019_waste.csv')
energy_stats = pd.read_csv('datasets/energy_saved.csv')
tuple_df = (waste_stats_older, waste_stats_newer, energy_stats)
```

[5]:
```
for df in tuple_df:
    print(df.head())
    print(df.shape)
    print(df.columns[df.isna().any()].tolist())
    print(df.columns)
```

```
           waste_type  waste_disposed_of_tonne  total_waste_recycled_tonne  \
0                Food                   679900                       111100.0
1     Paper/Cardboard                   576000                       607100.0
2             Plastics                  762700                        59500.0
3                 C&D                     9700                      1585700.0
4  Horticultural waste                  111500                       209000.0

   total_waste_generated_tonne  recycling_rate  year
0                       791000            0.14  2016
1                      1183100            0.51  2016
2                       822200            0.07  2016
3                      1595400            0.99  2016
4                       320500            0.65  2016
(225, 6)
[]
Index(['waste_type', 'waste_disposed_of_tonne', 'total_waste_recycled_tonne',
       'total_waste_generated_tonne', 'recycling_rate', 'year'],
      dtype='object')
                Waste Type  Total Generated ('000 tonnes)  \
0  Construction& Demolition                          1440
1             Ferrous Metal                          1278
2           Paper/Cardboard                          1011
3                  Plastics                           930
4                      Food                          7440

   Total Recycled ('000 tonnes)  Year
0                          1434  2019
1                          1270  2019
2                           449  2019
3                            37  2019
4                           136  2019
(30, 4)
[]
Index(['Waste Type', 'Total Generated ('000 tonnes)',
       'Total Recycled ('000 tonnes)', 'Year'],
      dtype='object')
  The table gives the amount of energy saved in kilowatt hour (kWh) and the
amount of crude oil (barrels) by recycling 1 metric tonne (1000 kilogram)  per
waste type  \
0    1 barrel oil is approximately 159 litres of oil
1                                                NaN
2                                           material
3                                      energy_saved
4                                    crude_oil saved

   Unnamed: 1 Unnamed: 2     Unnamed: 3         Unnamed: 4   Unnamed: 5
0         NaN        NaN            NaN                NaN          NaN
1         NaN        NaN            NaN                NaN          NaN
```

```
2     Plastic      Glass  Ferrous Metal  Non-Ferrous Metal        Paper
3    5774 Kwh     42 Kwh        642 Kwh          14000 Kwh     4000 kWh
4  16 barrels        NaN    1.8 barrels         40 barrels  1.7 barrels
(5, 6)
['The table gives the amount of energy saved in kilowatt hour (kWh) and the
amount of crude oil (barrels) by recycling 1 metric tonne (1000 kilogram)  per
waste type', 'Unnamed: 1', 'Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4', 'Unnamed:
5']
Index(['The table gives the amount of energy saved in kilowatt hour (kWh) and
the amount of crude oil (barrels) by recycling 1 metric tonne (1000 kilogram)
per waste type',
       'Unnamed: 1', 'Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4', 'Unnamed: 5'],
      dtype='object')
```

[6]:
```
#Convert '000 tonnes in newer stats data to tonnes
convert_cols = waste_stats_newer.columns[1:3]
waste_stats_newer[convert_cols] = waste_stats_newer[convert_cols]*1000
```

[7]:
```
stats_dict = {'waste_type':'Waste Type',
              'waste_disposed_of_tonne':'Waste Disposed of (tonnes)',
              'recycling_rate': 'Recycling Rate',
             'total_waste_generated_tonne':'Total Generated (\'000 tonnes)',
             'total_waste_recycled_tonne':'Total Recycled (\'000 tonnes)',
             'year':'Year'}
waste_stats_older = waste_stats_older.rename(columns = stats_dict)
print(waste_stats_older['Waste Type'].unique())
print(len(waste_stats_older['Waste Type'].unique()))
print(waste_stats_newer['Waste Type'].unique())
print(len(waste_stats_newer['Waste Type'].unique()))
waste_stats_older.head()
```

```
['Food' 'Paper/Cardboard' 'Plastics' 'C&D' 'Horticultural waste' 'Wood'
 'Ferrous metal' 'Non-ferrous metal' 'Used slag' 'Ash & Sludge' 'Glass'
 'Textile/Leather' 'Scrap tyres' 'Others (stones, ceramics & rubber etc.)'
 'Total' 'Others (stones, ceramics & rubber etc)' 'Food waste'
 'Construction Debris' 'Wood/Timber' 'Horticultural Waste' 'Ferrous Metal'
 'Non-ferrous Metals' 'Used Slag' 'Sludge' 'Scrap Tyres' 'Ferrous Metals'
 'Others' 'Construction debris' 'Non-ferrous metals' 'Ash and sludge'
 'Plastic' 'Others (stones, ceramic, rubber, etc.)']
32
['Construction& Demolition' 'Ferrous Metal' 'Paper/Cardboard' 'Plastics'
 'Food' 'Wood' 'Horticultural' 'Ash & Sludge' 'Textile/Leather'
 'Used Slag' 'Non-Ferrous Metal' 'Glass' 'Scrap Tyres'
 'Others (stones, ceramic, rubber, ect)' 'Overall']
15
```

[7]:

	Waste Type	Waste Disposed of (tonnes)	\
0	Food	679900	

```
1      Paper/Cardboard                       576000
2              Plastics                      762700
3                   C&D                        9700
4  Horticultural waste                       111500

   Total Recycled ('000 tonnes)  Total Generated ('000 tonnes)  \
0                      111100.0                          791000
1                      607100.0                         1183100
2                       59500.0                          822200
3                     1585700.0                         1595400
4                      209000.0                          320500

   Recycling Rate  Year
0            0.14  2016
1            0.51  2016
2            0.07  2016
3            0.99  2016
4            0.65  2016
```

[8]:
```
print(waste_stats_newer['Waste Type'].value_counts())
print(waste_stats_older['Waste Type'].value_counts())
```

```
Others (stones, ceramic, rubber, ect)    2
Ash & Sludge                             2
Overall                                  2
Construction& Demolition                 2
Paper/Cardboard                          2
Used Slag                                2
Ferrous Metal                            2
Plastics                                 2
Textile/Leather                          2
Scrap Tyres                              2
Wood                                     2
Horticultural                            2
Non-Ferrous Metal                        2
Food                                     2
Glass                                    2
Name: Waste Type, dtype: int64
Paper/Cardboard                          15
Textile/Leather                          15
Glass                                    15
Total                                    15
Plastics                                 14
Others (stones, ceramics & rubber etc)   12
Scrap Tyres                              11
Sludge                                   11
Construction Debris                      11
Horticultural Waste                      11
```

```
Non-ferrous Metals                          11
Wood/Timber                                 11
Food waste                                  11
Used Slag                                   11
Ferrous Metal                                7
Food                                         4
Horticultural waste                          4
Scrap tyres                                  4
Wood                                         4
Ferrous Metals                               4
Ferrous metal                                4
Used slag                                    4
C&D                                          3
Ash & Sludge                                 3
Non-ferrous metal                            3
Others                                       1
Non-ferrous metals                           1
Ash and sludge                               1
Plastic                                      1
Others (stones, ceramic, rubber, etc.)       1
Others (stones, ceramics & rubber etc.)      1
Construction debris                          1
Name: Waste Type, dtype: int64
```

[9]:
```
#Regex syntax to pull for considered values of Waste Type attribute
## Non-Ferrous Metals
# (?i)(non).(ferrous).\w*\b
## Ferrous Metal
#(?i)^(ferrous).(metal)\s
## plastic
# (?:lastic)
regx_values = {r'(?i)(non).(ferrous).\w*\b':'Non-Ferrous Metal',
               r'(?i)^(ferrous).\w*\b':'Ferrous Metal',
               r'(?i)^.*(plastic).?\b': 'Plastics'
               }
```

[10]:
```
f_waste_stats_older = waste_stats_older.replace(regx_values, regex = True)
print(f_waste_stats_older['Waste Type'].unique())
print(f_waste_stats_older['Waste Type'].value_counts())
```

```
['Food' 'Paper/Cardboard' 'Plastics' 'C&D' 'Horticultural waste' 'Wood'
 'Ferrous Metal' 'Non-Ferrous Metal' 'Used slag' 'Ash & Sludge' 'Glass'
 'Textile/Leather' 'Scrap tyres' 'Others (stones, ceramics & rubber etc.)'
 'Total' 'Others (stones, ceramics & rubber etc)' 'Food waste'
 'Construction Debris' 'Wood/Timber' 'Horticultural Waste' 'Used Slag'
 'Sludge' 'Scrap Tyres' 'Others' 'Construction debris' 'Ash and sludge'
 'Others (stones, ceramic, rubber, etc.)']
Non-Ferrous Metal                           15
```

```
Plastics                                    15
Ferrous Metal                               15
Glass                                       15
Textile/Leather                             15
Total                                       15
Paper/Cardboard                             15
Others (stones, ceramics & rubber etc)      12
Horticultural Waste                         11
Used Slag                                   11
Sludge                                      11
Construction Debris                         11
Scrap Tyres                                 11
Wood/Timber                                 11
Food waste                                  11
Used slag                                    4
Wood                                         4
Scrap tyres                                  4
Food                                         4
Horticultural waste                          4
Ash & Sludge                                 3
C&D                                          3
Construction debris                          1
Others (stones, ceramic, rubber, etc.)       1
Others                                       1
Ash and sludge                               1
Others (stones, ceramics & rubber etc.)      1
Name: Waste Type, dtype: int64
```

[11]:
```
total_waste_stats = pd.concat((f_waste_stats_older, waste_stats_newer))
recycled_waste_stats = total_waste_stats[total_waste_stats['Waste Type'].
 ↪isin(['Glass','Plastics', 'Ferrous Metal', 'Non-Ferrous Metal'])]
recycled_waste_stats=recycled_waste_stats.reset_index(drop = True)
recycled_waste_stats.sort_values('Total Recycled (\'000 tonnes)', ascending =␣
 ↪False).head()
```

[11]:

	Waste Type	Waste Disposed of (tonnes)	Total Recycled ('000 tonnes)	Total Generated ('000 tonnes)	Recycling Rate	Year
9	Ferrous Metal	57000.0	1388900.0	1445900	0.96	2014
56	Ferrous Metal	7800.0	1371000.0	1378800	0.99	2017
13	Ferrous Metal	46800.0	1369200.0	1416000	0.97	2013
1	Ferrous Metal	6000.0	1351500.0	1357500	0.99	2016
5	Ferrous Metal	15200.0	1333300.0			

```
5                         1348500            0.99  2015
```

[12]:
```
recycled_waste_stats['Waste Type']
```

[12]:
```
0              Plastics
1         Ferrous Metal
2     Non-Ferrous Metal
3                 Glass
4              Plastics
             …
63                Glass
64        Ferrous Metal
65             Plastics
66    Non-Ferrous Metal
67                Glass
Name: Waste Type, Length: 68, dtype: object
```

[13]:
```
energy_stats=energy_stats.transpose()
energy_stats.head()
```

[13]:
```
                   0  \
The table gives the amount of energy saved in k…  1 barrel oil is
approximately 159 litres of oil
Unnamed: 1
NaN
Unnamed: 2
NaN
Unnamed: 3
NaN
Unnamed: 4
NaN

                                                    1                  2  \
The table gives the amount of energy saved in k…  NaN           material
Unnamed: 1                                        NaN            Plastic
Unnamed: 2                                        NaN              Glass
Unnamed: 3                                        NaN      Ferrous Metal
Unnamed: 4                                        NaN  Non-Ferrous Metal

                                                          3  \
The table gives the amount of energy saved in k…  energy_saved
Unnamed: 1                                          5774 Kwh
Unnamed: 2                                            42 Kwh
Unnamed: 3                                           642 Kwh
Unnamed: 4                                         14000 Kwh

                                                          4
```

```
The table gives the amount of energy saved in k…  crude_oil saved
Unnamed: 1                                                16 barrels
Unnamed: 2                                                       NaN
Unnamed: 3                                              1.8 barrels
Unnamed: 4                                               40 barrels
```

[14]:
```
f_energy_stats = energy_stats.iloc[0:5,2:].reset_index(drop = True)
f_energy_stats.columns = list(f_energy_stats.iloc[0])


f_energy_stats = f_energy_stats.iloc[1:]
f_energy_stats.head()
print(f_energy_stats.head())

#Replace 'Plastic' to 'Plastics'
f_energy_stats = f_energy_stats.replace({'Plastic': 'Plastics'})
print(f_energy_stats['material'].value_counts())
```

```
           material energy_saved crude_oil saved
1           Plastic     5774 Kwh      16 barrels
2             Glass       42 Kwh             NaN
3      Ferrous Metal      642 Kwh     1.8 barrels
4  Non-Ferrous Metal    14000 Kwh      40 barrels
Plastics             1
Non-Ferrous Metal    1
Glass                1
Ferrous Metal        1
Name: material, dtype: int64
```

[15]:
```
f_energy_stats['energy_saved'] = f_energy_stats['energy_saved'].str.split(r'\s.
 ↪*?\Z',expand = True)
f_energy_stats['energy_saved'] = f_energy_stats['energy_saved'].astype(int)
```

[16]:
```
recycled_waste_stats['Total Recycled (\'000 tonnes)'] =␣
 ↪recycled_waste_stats['Total Recycled (\'000 tonnes)'].astype(int)

print(f_energy_stats['energy_saved'].head())
print(type(f_energy_stats['energy_saved']))
print(recycled_waste_stats['Total Recycled (\'000 tonnes)'].head())
print(type(recycled_waste_stats['Total Recycled (\'000 tonnes)']))
```

```
1     5774
2       42
3      642
4    14000
Name: energy_saved, dtype: int32
<class 'pandas.core.series.Series'>
0      59500
```

```
1    1351500
2      95900
3      14700
4      57800
Name: Total Recycled ('000 tonnes), dtype: int32
<class 'pandas.core.series.Series'>
```

[17]:
```
rec_waste_stats = recycled_waste_stats.merge(f_energy_stats, left_on='Waste␣
↪Type', right_on='material')
rec_waste_stats =rec_waste_stats.rename(columns = {'Year':'year'})
rec_waste_stats['total_energy_saved'] = rec_waste_stats['Total Recycled (\'000␣
↪tonnes)'].values * rec_waste_stats['energy_saved'].values
rec_waste_stats.head()
```

[17]:

	Waste Type	Waste Disposed of (tonnes)	Total Recycled ('000 tonnes)	\
0	Plastics	762700.0	59500	
1	Plastics	766800.0	57800	
2	Plastics	789000.0	80000	
3	Plastics	741100.0	91100	
4	Plastics	721300.0	82100	

	Total Generated ('000 tonnes)	Recycling Rate	year	material	\
0	822200	0.07	2016	Plastics	
1	824600	0.07	2015	Plastics	
2	869000	0.09	2014	Plastics	
3	832200	0.11	2013	Plastics	
4	803400	0.10	2012	Plastics	

	energy_saved	crude_oil saved	total_energy_saved
0	5774	16 barrels	343553000
1	5774	16 barrels	333737200
2	5774	16 barrels	461920000
3	5774	16 barrels	526011400
4	5774	16 barrels	474045400

[18]:
```
rec_waste_stats.sort_values(by = ['material','year'], ascending = False)
rec_waste_stats=rec_waste_stats.loc[rec_waste_stats['year'].
↪isin(list(range(2015,2020)))]
rec_waste_stats['Waste Type'].value_counts()
```

[18]:
```
Plastics            5
Non-Ferrous Metal   5
Glass               5
Ferrous Metal       5
Name: Waste Type, dtype: int64
```

[19]:
```
annual_energy_savings = rec_waste_stats[['total_energy_saved', 'year','Waste␣
↪Type']]
annual_energy_savings
```

[19]:

	total_energy_saved	year	Waste Type
0	343553000	2016	Plastics
1	333737200	2015	Plastics
14	299093200	2017	Plastics
15	213638000	2019	Plastics
16	236734000	2018	Plastics
17	867663000	2016	Ferrous Metal
18	855978600	2015	Ferrous Metal
31	880182000	2017	Ferrous Metal
32	815340000	2019	Ferrous Metal
33	80892000	2018	Ferrous Metal
34	1342600000	2016	Non-Ferrous Metal
35	-2049367296	2015	Non-Ferrous Metal
48	1290800000	2017	Non-Ferrous Metal
49	1736000000	2019	Non-Ferrous Metal
50	-1914967296	2018	Non-Ferrous Metal
51	617400	2016	Glass
52	613200	2015	Glass
65	520800	2017	Glass
66	462000	2019	Glass
67	504000	2018	Glass

[20]:
```
annual_energy_savings = annual_energy_savings.groupby(['year']).sum()
annual_energy_savings.sort_values('total_energy_saved')
annual_energy_savings
```

[20]:

year	total_energy_saved
2015	-8.590383e+08
2016	2.554433e+09
2017	2.470596e+09
2018	-1.596837e+09
2019	2.765440e+09

[21]:
```
annual_energy_savings
```

[21]:

year	total_energy_saved
2015	-8.590383e+08
2016	2.554433e+09
2017	2.470596e+09
2018	-1.596837e+09
2019	2.765440e+09

[ ]: