Table of contents

Kulstof

1 Case

The purpose of this analysis is to get knowledge and collect data regarding carbon in the soil.

Here we are using three different data sources:

- Mark Analysis Online (MAO) which is our own.
- Polygon data that is provided by Rita.
- `Forsøg DB (FSDB)` a experiment.

We want to combine the data sources to get an idea on how it has evolv over time and get information from the MAO and FSDB.

2 Analysis

2.1 MAO

MAO contains **36771** data observation and **51** columns.

We have tree columns of interst:

- `humus`
- `kulstof`
- `totalkulstofpct`

There can be some observation where one of the columns not contains a value or is 0. It is agreed with Henrik to keep all observation.

2.2 FSDB

FSDB has 823 rows and 122 columns.

2.3 Mark polygon

It contains 574040 rows and 70 columns.

2.4 The combined data

After we have made the combination of the data we get a table like this

id_new	dataset	date	humus	ler	geometry	polygon	kulstof	totalkulstofpct
1342	MAO	2016-01-01	1.1	1.0	10, 58	10, 57	0	0

- `id_new` is a unique key that is different from MOA and FSDB. In MAO it is a unique key made of Rita and is called `id_rth`. In FSDB it is a combination of `PLANNR` and `LBNR`.
- `dataset` is an identifier for showing if the data point is from MAO or FSDB.
- `date` is a date for the experiment.
- `humus` in both FSDB and MAO we have a humus column that we just append on top of each other.
- `ler` the same goes for ler.
- `geometry` and `polygon` just specify where the mark and point is located.
- `kulstof` and `totalkulstofpct` are columns in MAO that is of interest but is not in FSDB.

When combining the data we actually get a few more observation ex for MAO we get 36774 where we original had 36771. It is because the point lays on the edges of two mark polygons:

2.5 MAO and FSDB on same polygon.

There are some points that lay in the same polygon

```
import sys
print(sys.executable)
```

```
C:\Users\lucb\AppData\Local\Programs\Python\Python310\pythonw.exe
```

```
# Data wrangling
import pandas as pd
import numpy as np

# Geo data wrangling
import geopandas as gpd
#from pyjanitor import cLean_names

import matplotlib.pyplot as plt

# Style
plt.rc('figure', figsize = (10,10))
pd.set_option("display.max_columns", 500)
```

```
df = pd.read_excel('to henrik v3.xlsx')
```

```
df.head(2)
```

	Unnamed: 0	id_new	dataset	date	humus	ler	geometry	polygon	kulstof	totalkulstofpct	id	origid	cvr_left	farmid	aar_udtagning	manedudtagning	høstar	fi
0	0	2016#57.3616#10.2209	MAO	2016-01-26	0.0	0.0	POINT (10.22092 57.361597)	POLYGON ((10.2204650073341 57.35973901357593, ...	0.0	2.03	311590.0	98958888	25057120.0	8275.0	2016.0	1.0	2016.0	2:
1	1	2017#55.1257#9.31971	MAO	2017-11-24	0.0	0.0	POINT (9.3197149218175	POLYGON ((9.328257032238005	0.0	1.32	477090.0	14664471	16265241.0	59431.0	2017.0	11.0	2017.0	6

Unnamed: 0	id_new	dataset	date	humus	ler	geometry	polygon	kulstof	totalkulstofpct	id	origid	cvr_left	farmid	aar_udtagning	manedudtagning	høstar	fi
						55.12571865252)	55.12373604215482,...										

2.6 Aggregation

2.6.1 Size of MAO and FSDB

```
(
    df
    .groupby('dataset')
    .agg({'dataset': 'size'})
)
```

	dataset
dataset	
FSDB	824
MAO	36774

```
(
    df
    .pivot_table(
      aggfunc = np.size,
      index = 'dataset',
      columns = 'marknr'
)
)
```

marknr	001-0	0010	0017	0025	0027	003-0	0037	0049	005-0	0050	006-0	007-0	008-0	009-0	010-0	010-22	011-22	012-0	012-22	013-0	013-22	014-22	015-22	02	1	1-0	1-05	1-0a	1-0b	1-0c	1-1	1-10	1-11	1-2	1-2a	1-2b	1-3	1-
dataset																																						
FSDB	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	73.0	NaN	NaN	NaN	NaN	2.0	NaN	NaN	2.0	NaN	NaN	NaN	Na
MAO	2.0	2.0	2.0	2.0	3.0	1.0	1.0	2.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	7.0	1.0	1.0	1.0	4.0	1.0	1.0	1.0	15.0	1549.0	1.0	14.0	7.0	2.0	241.0	4.0	7.0	81.0	1.0	1.0	28.0	7.0

2 rows × 376516 columns

2.6.2 Missing and Zeros value in `ler`, `humus` and `kulstof`

```
(
  df
  [['humus', 'ler', 'kulstof', 'totalkulstofpct']]
  .isnull()
  .sum()
)
```

```
humus               0
ler                 0
kulstof           824
totalkulstofpct   824
dtype: int64
```

So we are only missing value in kulstof and totalkulstofpct which is unique columns.

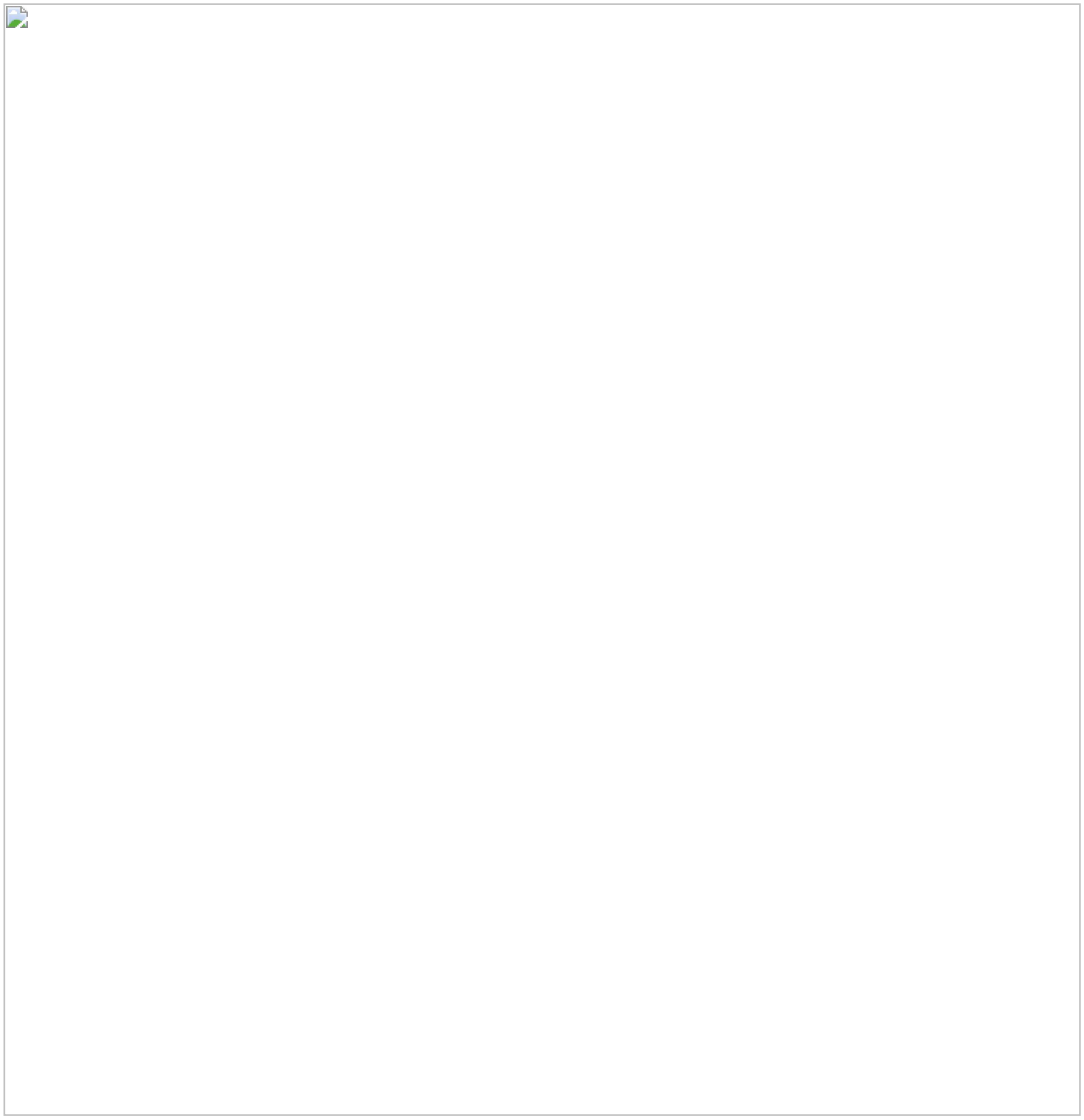
```
for cn in df[['humus', 'ler', 'kulstof', 'totalkulstofpct']]:
  col = df[cn]
  count = (col == 0).sum()
  print('Count of zeros in column', cn, 'is:', count)
```

```
Count of zeros in column humus is: 32630
Count of zeros in column ler is: 32169
Count of zeros in column kulstof is: 36774
Count of zeros in column totalkulstofpct is: 4129
```

2.6.3 Number of observation for each year

```
(
  df
  .assign(
    year = lambda x: x.date.dt.year.astype(str)
  )
  .groupby('year')
  .agg({'year': 'size'})
  .plot(kind = 'bar')
)
```

```
<AxesSubplot: xlabel='year'>
```



So a lot of observatoin from 2011 which is from MAO mostly and it dosent have any data point before. But let os have a look:

```
(
  df
  .assign(
    year = lambda x: x.date.dt.year.astype(str)
  )
  .groupby(['year', 'dataset'])
  .agg({'year': 'size'})

)
```

		year
year	**dataset**	
1996.0	**FSDB**	13
1997.0	**FSDB**	31
1998.0	**FSDB**	32
1999.0	**FSDB**	33
2000.0	**FSDB**	39
2001.0	**FSDB**	39
2002.0	**FSDB**	35
2003.0	**FSDB**	28
2004.0	**FSDB**	15
	MAO	21
2005.0	**FSDB**	26
2006.0	**FSDB**	31
2007.0	**FSDB**	6
2008.0	**FSDB**	30
2009.0	**FSDB**	28
2010.0	**FSDB**	6
2011.0	**FSDB**	17
	MAO	477
2012.0	**FSDB**	25
	MAO	485
2013.0	**FSDB**	36
	MAO	981
2014.0	**FSDB**	47
	MAO	2313
2015.0	**FSDB**	51
	MAO	2531
2016.0	**FSDB**	34
	MAO	2830

		year
year	**dataset**	
2017.0	**FSDB**	28
	MAO	2932
2018.0	**FSDB**	46
	MAO	4460
2019.0	**FSDB**	34
	MAO	4733
2020.0	**FSDB**	10
	MAO	6568
2021.0	**MAO**	4917
2022.0	**MAO**	3526
nan	**FSDB**	104

Surprisingly there is 104 observation from FSDB that hasn’t a date.