test

In [ ]:

```
import sqlite3
import numpy as np
import pandas as pd
import scipy.stats as st
import matplotlib.pyplot as plt
import statsmodels.formula.api as sm
import statsmodels.api as sm_api

%matplotlib inline

db_file = 'elderspeak_detect.db'
con = sqlite3.connect(db_file)

df = pd.read_sql_query('''
    select
        td.elderspeak,
       pr.spraaksnelheid,
       pr.geluidsniveau,
           pr.toonhoogte,
        tr.cilt,
        tr.woordlengteratio,
        tr.aantal_collectieve_voornaamwoorden,
        tr.aantal_verkleinwoorden,
        tr.aantal_herhalingen,
        tr.textcat_elderspeak_score,
        id.geslacht,
        id.leeftijd,
        id.moedertaal,
        id.student_zorg,
        id.werk_zorg
    from praat_resultaten pr
    join tekst_resultaten tr on pr.audio_id = tr.audio_id
    join teksten t on t.tekst_id = tr.tekst_id
    join test_data td on td.audio_id = tr.audio_id
```

```
    left join input_data id on tr.audio_id in (id.leeftijdsgenoot_opname, id.oudere_opname)
    where t.methode = 'GOOGLE_ENKEL_NL_BE'
    and pr.spraaksnelheid > 0''', con)
con.close()

print(f"Aantal kolommen: { len(df.columns) }")
print(f"Aantal rijen: { len(df.index) }")
df.tail()
```

```
Aantal kolommen: 15
Aantal rijen: 47
```

Out[ ]:

	elderspeak	spraaksnelheid	geluidsniveau	toonhoogte	cilt	woordlengteratio	aantal_collectiev
42	0	3.159722	73.272019	111.466943	55.39	0.166667	0
43	1	2.892109	71.059838	194.761772	65.78	0.114754	2
44	0	3.540215	75.012906	129.553227	68.45	0.044944	2
45	0	2.429387	33.696677	98.056544	-1.00	-1.000000	0
46	1	1.281776	66.549338	237.226846	65.39	0.111111	0

In [ ]:

```
reg = sm.ols(formula="elderspeak ~ spraaksnelheid + geluidsniveau + toonhoogte", data=df).fi
reg.params
```

Out[ ]:

```
Intercept        -0.682542
spraaksnelheid   -0.154075
geluidsniveau     0.016376
toonhoogte        0.002186
dtype: float64
```

In [ ]:

```
reg = sm.ols(formula="elderspeak ~ cilt + woordlengteratio + aantal_collectieve_voornaamwoor
reg.params
```

Out[ ]:

```
Intercept                             0.641273
cilt                                 -0.004738
woordlengteratio                      0.603489
aantal_collectieve_voornaamwoorden   -0.045982
aantal_verkleinwoorden               -0.033458
aantal_herhalingen                    0.070305
dtype: float64
```

In [ ]:

```
reg = sm.ols(formula="elderspeak ~ spraaksnelheid + geluidsniveau + toonhoogte + cilt + woor
reg.params
```

Out[ ]:

```
Intercept                           -0.338043
spraaksnelheid                      -0.113442
geluidsniveau                        0.011385
toonhoogte                           0.001475
cilt                                -0.004189
woordlengteratio                    -0.120754
aantal_collectieve_voornaamwoorden  -0.032410
aantal_verkleinwoorden               0.032736
aantal_herhalingen                   0.020968
textcat_elderspeak_score             0.631851
dtype: float64
```

In [ ]:

```
df['reg_model_praat'] = -0.682542 -0.154075 * df['spraaksnelheid'] + 0.016376 * df['geluidsn
df['reg_model_tekst'] = 0.641273 -0.004738 * df['cilt'] + 0.603489 * df['woordlengteratio']
df['reg_model_volledig'] = -0.338043 -0.113442 * df['spraaksnelheid'] + 0.011385 * df['gelui

df[['elderspeak', 'textcat_elderspeak_score', 'reg_model_praat', 'reg_model_tekst', 'reg_moo
```

Out[ ]:

	elderspeak	textcat_elderspeak_score	reg_model_praat	reg_model_tekst	reg_model_volledig
0	0	1.492628e-10	0.470329	0.359381	0.127765
1	1	5.497069e-06	0.740600	0.429100	0.423556
2	0	6.963545e-07	0.308622	0.363723	0.089986
3	0	2.806736e-05	0.180474	0.717499	0.212633
4	0	7.241415e-06	0.467125	0.362286	0.131557
5	1	6.058949e-05	0.205637	0.278221	0.158393
6	0	9.893542e-10	0.505150	0.267387	0.141166
7	0	1.449370e-02	0.492853	0.429835	0.156388
8	1	1.946689e-03	0.588510	0.431145	0.339568
9	1	9.999999e-01	0.641194	0.581487	1.087465
10	0	1.799652e-07	0.336121	0.331631	0.104039
11	1	1.000000e+00	0.561459	0.386736	0.962166
12	0	1.909542e-03	0.455485	0.348453	0.100614
13	1	7.815228e-04	0.392451	0.507617	0.337252
14	1	1.000000e+00	0.563268	0.433406	0.910897
15	0	1.333404e-03	0.354037	0.257563	0.103856
16	0	2.874842e-07	0.578989	0.254221	0.228866

	elderspeak	textcat_elderspeak_score	reg_model_praat	reg_model_tekst	reg_model_volledig
17	0	8.234085e-11	0.147196	0.572319	0.032711
18	0	6.649797e-06	0.461301	0.513211	0.279516
19	1	9.982306e-01	0.345245	0.259101	0.687906
20	0	9.992041e-01	0.435058	0.448492	0.817198
21	0	3.851894e-06	0.289190	0.229405	0.087322
22	0	1.233303e-11	0.311159	0.257718	0.034395
23	0	2.324891e-06	0.071862	0.254421	-0.074185
24	1	1.000000e+00	0.298876	0.458302	0.959450
25	1	9.931636e-01	0.462966	0.270767	0.745433
26	0	3.488684e-04	0.495693	0.632381	0.473252
27	0	6.657302e-10	0.173289	0.238267	0.021565
28	1	1.000000e+00	0.590950	0.413713	0.923128
29	0	2.514670e-08	0.215697	0.569761	0.165777
30	0	8.560129e-08	0.400407	0.158562	0.140721
31	0	1.360664e-09	0.300849	0.284071	0.200330
32	0	2.888697e-09	0.265324	0.371749	0.010589
33	0	9.995882e-01	0.408590	0.334147	0.756422
34	1	9.995782e-01	0.258781	0.496022	0.802164
35	1	9.999998e-01	0.296625	0.501338	0.758773
36	0	7.209262e-03	0.427324	0.325899	0.154071
37	0	3.539525e-05	0.462169	0.312699	0.295817
38	1	1.000000e+00	0.523853	0.384143	0.768908
39	0	1.292956e-08	0.349928	0.182970	0.111815
40	1	3.924295e-04	0.266798	0.437544	0.007674
41	1	9.999995e-01	0.445832	0.329136	0.873847
42	0	7.054659e-10	0.274193	0.415890	0.201885
43	1	1.000000e+00	0.461281	0.698657	0.952766
44	0	2.404300e-06	0.283614	0.399504	0.163294
45	0	0.000000e+00	-0.290681	0.042522	0.039576
46	1	1.000000e+00	0.728358	0.468815	0.989607

In [ ]:

```
def cohend(d1: pd.Series, d2: pd.Series) -> float:
    # calculate the size of samples
    n1, n2 = len(d1), len(d2)

    # calculate the variance of the samples
    s1, s2 = np.var(d1, ddof=1), np.var(d2, ddof=1)

    # calculate the pooled standard deviation
    s = np.sqrt(((n1 - 1) * s1 + (n2 - 1) * s2) / (n1 + n2 - 2))
```

```
    # calculate the means of the samples
    u1, u2 = np.mean(d1), np.mean(d2)

    # the effect size
    return (u1 - u2) / s
```

In [ ]:

```
elderspeak = df[(df['elderspeak'] == 1)]
not_elderspeak = df[(df['elderspeak'] == 0)]

cd_dict = {
    'kenmerk': [],
    'd': [],
    'effect': []
}

for c in ['spraaksnelheid', 'geluidsniveau', 'toonhoogte', 'cilt', 'woordlengteratio', 'aant
    d = cohend(elderspeak[c], not_elderspeak[c])
    abs_d = np.abs(d)

    if abs_d <= 0.01:
        effect = 'zeer klein'
    elif abs_d <= 0.2:
        effect = 'klein'
    elif abs_d <= 0.5:
        effect = 'middelmatig'
    elif abs_d <= 0.8:
        effect = 'groot'
    elif abs_d <= 1.2:
        effect = 'zeer groot'
    else:
        effect = 'reusachtig'

    cd_dict['kenmerk'].append(c)
    cd_dict['d'].append(d)
    cd_dict['effect'].append(effect)

df_cd = pd.DataFrame.from_dict(cd_dict)
df_cd
```

Out[ ]:

	kenmerk	d	effect
0	spraaksnelheid	-0.061360	klein
1	geluidsniveau	0.412075	middelmatig

	kenmerk	d	effect
2	toonhoogte	0.488279	middelmatig
3	cilt	0.190854	klein
4	woordlengteratio	0.227493	middelmatig
5	aantal_collectieve_voornaamwoorden	-0.109957	klein
6	aantal_verkleinwoorden	0.092747	klein
7	aantal_herhalingen	0.353327	middelmatig
8	textcat_elderspeak_score	1.872398	reusachtig
9	leeftijd	0.322459	middelmatig
10	reg_model_praat	0.788455	groot
11	reg_model_tekst	0.587246	groot
12	reg_model_volledig	2.103016	reusachtig