

Beta-Binomial Model

```
In [1]: import pandas as pd
import numpy as np
from scipy import special
from scipy import optimize
```

```
ichs_data = pd.read_csv('ICHS.txt', sep=' ')
ichs_data.head()
```

Out[1]:

	id	gender	height	cosv	sinv	xero	baseage	age	infect
1	121013	0	-3	-1	0	0	31	31	0
2	121013	0	-3	0	-1	0	31	34	0
3	121013	0	-2	1	0	0	31	37	0
4	121013	0	-2	0	1	0	31	40	0
5	121013	0	-2	-1	0	0	31	43	1

```
In [2]: ichs_data_grouped = pd.DataFrame({
    'baseage': ichs_data[['id', 'baseage']].groupby('id').mean()['baseage'],
    'infect': ichs_data[['id', 'infect']].groupby('id').sum()['infect'],
    'n': ichs_data.groupby('id').size()
})

ichs_data_grouped.head()
```

Out[2]:

	id	baseage	infect	n
	121013	31	1	6
	121113	-9	1	6
	121114	-26	2	6
	121140	-19	0	2
	121215	0	2	6

```
In [3]: def score_beta_binomial(X, y, n, gamma):
    X = np.column_stack((np.ones_like(y), X))

    mu = np.matmul(X, gamma[:-1])
    rho = gamma[-1]

    alpha = mu*(1-rho)/rho
    beta = (1 - mu)*(1 - rho)/rho

    return np.sum(np.array([
        special.digamma(alpha + y) - special.digamma(alpha + beta + n) - special.d
        igamma(alpha) + special.digamma(alpha + beta),
        special.digamma(beta + n - y) - special.digamma(alpha + beta + n) - specia
        l.digamma(beta) + special.digamma(alpha + beta),
    ]), axis=-1)
```

```
In [4]: mu, rho = optimize.root(  
        lambda x: score_beta_binomial(ichs_data_grouped[[]].values,  
                                       ichs_data_grouped['infect'].values,  
                                       ichs_data_grouped['n'].values,  
                                       x),  
        [0.1, 0.1])['x']  
mu, rho
```

```
Out[4]: (0.09015590102588235, 0.08221790596403289)
```

```
In [5]: special.logit(mu), special.logit(rho)
```

```
Out[5]: (-2.311732859660446, -2.4125868796604526)
```