Appendix: Data Processing by Python

December 12, 2023

[1]:

```
import pandas as pd
import numpy as np
from plotnine import *
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
```

[2]:

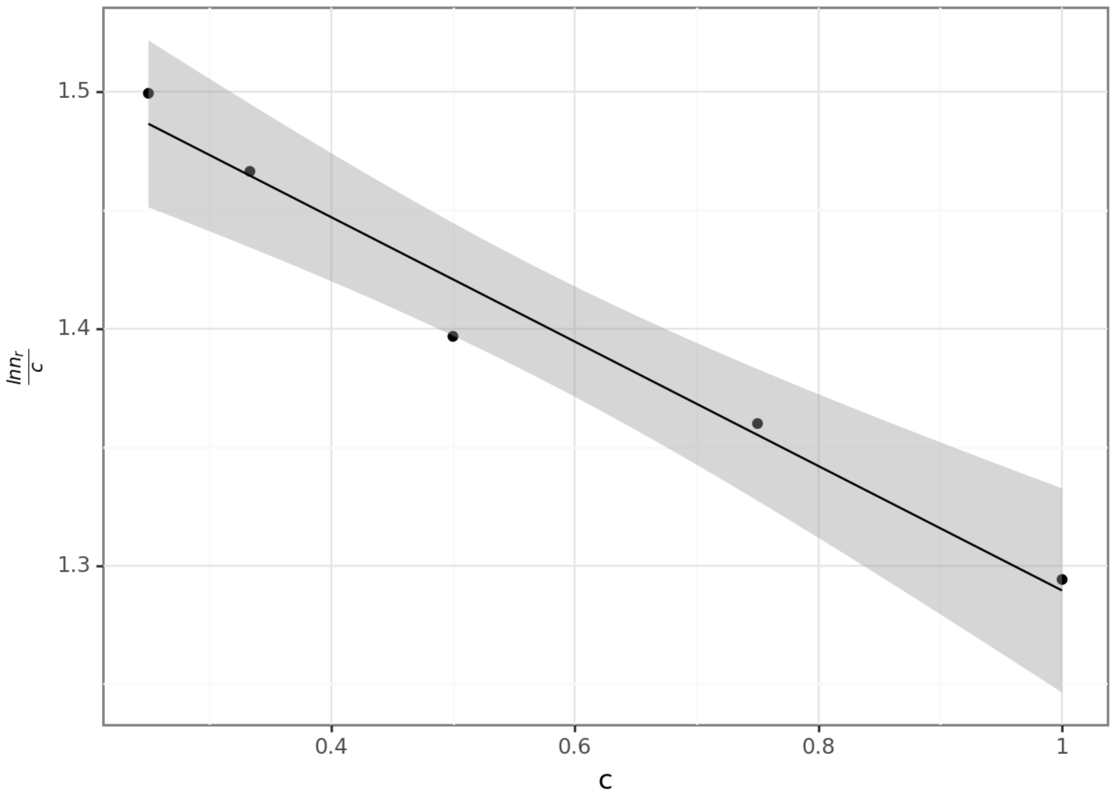
```
df = pd.read_csv("data.csv")
df['n_r'] = df['t'] / df['t'][5]
df['lnn_r'] = np.log(df['n_r'])
df['lnn_r/c'] = df['lnn_r'] / df['c']
df['n_sp'] = df['n_r'] - 1
df['n_sp/c'] = df['n_sp'] / df['c']
df.dropna(inplace = True)
df
```

[2]:

```
          t         c       n_r     lnn_r   lnn_r/c      n_sp    n_sp/c
0  315.9130  1.000000  3.647674  1.294090  1.294090  2.647674  2.647674
1  240.1600  0.750000  2.772996  1.019928  1.359904  1.772996  2.363994
2  174.1167  0.500000  2.010430  0.698349  1.396697  1.010430  2.020860
3  141.1967  0.333333  1.630321  0.488777  1.466330  0.630321  1.890962
4  125.9900  0.250000  1.454737  0.374825  1.499301  0.454737  1.818949
```

[3]:

```
p1 = ggplot(df, aes(x='c', y='lnn_r/c')) + geom_point() + geom_smooth(method =␣
 ↪'lm', size = 0.5) +\
    labs(y = '$\\frac{lnn_r}{c}$') + theme_bw()
p1
```



[3]: <Figure Size: (640 x 480)>

[4]:
```
import statsmodels.api as sm
import numpy as np

def lg(x, y):
    x = sm.add_constant(x)
    model = sm.OLS(y, x)
    results = model.fit()
    print(results.summary())
    intercept = results.params[0]
    slope = results.params[1]
    print(f'  (Intercept): {intercept}')
    print(f'  (Slope): {slope}')
```

[5]:
```
lg(df['t'], df['lnn_r/c'])
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                lnn_r/c   R-squared:                       0.941
Model:                            OLS   Adj. R-squared:                  0.921
```

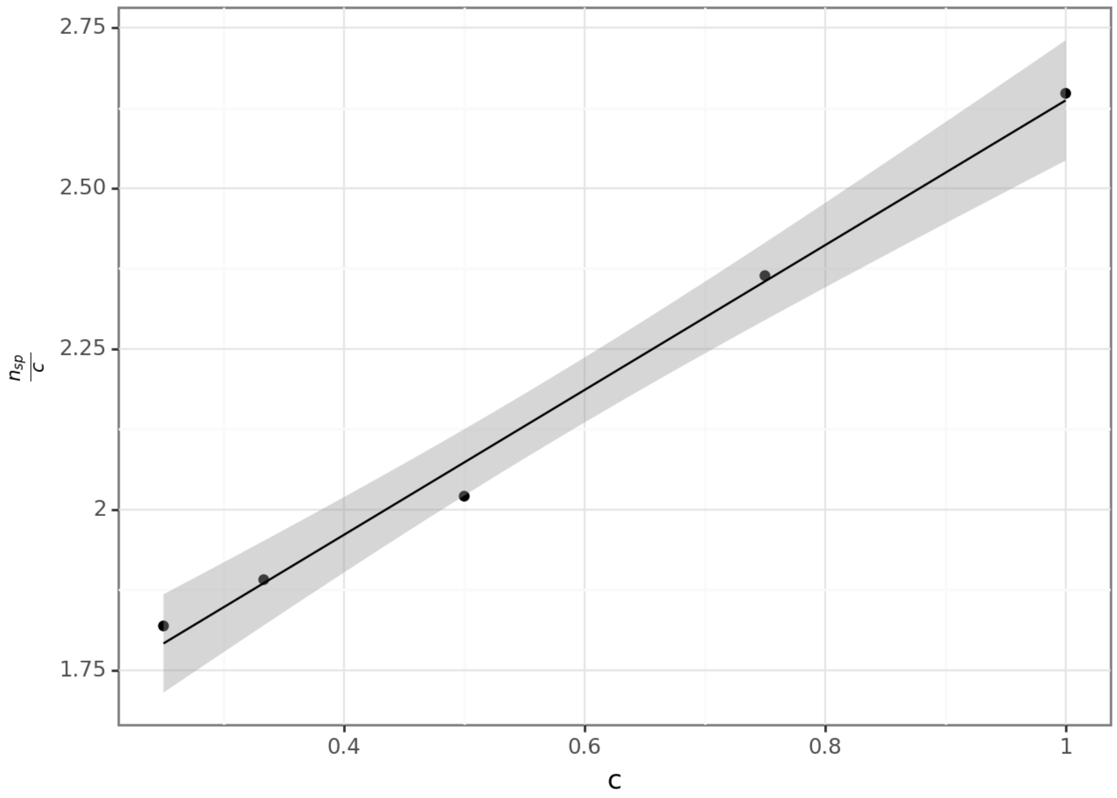
```
Method:                 Least Squares   F-statistic:                     47.56
Date:                Tue, 12 Dec 2023   Prob (F-statistic):            0.00625
Time:                        00:42:02   Log-Likelihood:                 13.016
No. Observations:                   5   AIC:                            -22.03
Df Residuals:                       3   BIC:                            -22.81
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          1.6059      0.031     51.556      0.000       1.507       1.705
t             -0.0010      0.000     -6.897      0.006      -0.001      -0.001
==============================================================================
Omnibus:                          nan   Durbin-Watson:                   1.657
Prob(Omnibus):                    nan   Jarque-Bera (JB):                0.586
Skew:                          -0.807   Prob(JB):                        0.746
Kurtosis:                       2.541   Cond. No.                         637.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
  (Intercept): 1.6058938305279733
  (Slope): -0.0010158113776368351
```

```
D:\conda\envs\datascience\lib\site-packages\statsmodels\stats\stattools.py:74:
ValueWarning: omni_normtest is not valid with less than 8 observations; 5
samples were given.
```

[6]:

```
p2 = ggplot(df, aes(x='c', y='n_sp/c')) + geom_point() + geom_smooth(method =␣
 ↪'lm', size = 0.5) + \
    labs(y = '$\\frac{n_{sp}}{c}$') + theme_bw()
p2
```



[6]: `<Figure Size: (640 x 480)>`

[7]:
```
lg(df['c'], df['n_sp/c'])
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 n_sp/c   R-squared:                       0.992
Model:                            OLS   Adj. R-squared:                  0.990
Method:                 Least Squares   F-statistic:                     388.7
Date:                Tue, 12 Dec 2023   Prob (F-statistic):           0.000285
Time:                        00:42:02   Log-Likelihood:                 10.908
No. Observations:                   5   AIC:                            -17.82
Df Residuals:                       3   BIC:                            -18.60
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          1.5100      0.036     41.921      0.000       1.395       1.625
c              1.1267      0.057     19.714      0.000       0.945       1.309
==============================================================================
Omnibus:                          nan   Durbin-Watson:                   2.041
Prob(Omnibus):                    nan   Jarque-Bera (JB):                1.204
Skew:                          -1.202   Prob(JB):                        0.548
Kurtosis:                       2.941   Cond. No.                         4.86
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
  (Intercept): 1.5100197850895833
  (Slope): 1.1267084319031693
```

```
D:\conda\envs\datascience\lib\site-packages\statsmodels\stats\stattools.py:74:
ValueWarning: omni_normtest is not valid with less than 8 observations; 5
samples were given.
```

[8]:

```python
df2 = df[['t', 'n_sp/c', 'lnn_r/c']]
df2 = pd.melt(df2, id_vars = 't')
df2
```

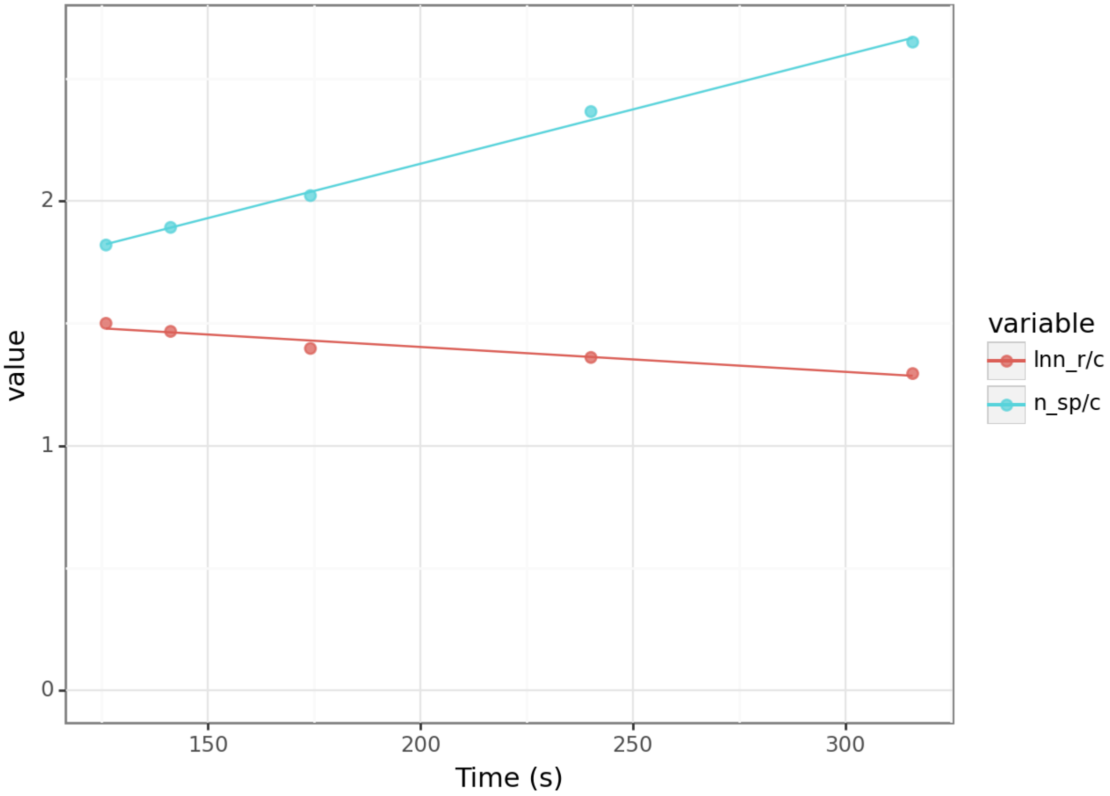
[8]:

```
          t variable     value
0  315.9130   n_sp/c  2.647674
1  240.1600   n_sp/c  2.363994
2  174.1167   n_sp/c  2.020860
3  141.1967   n_sp/c  1.890962
4  125.9900   n_sp/c  1.818949
5  315.9130  lnn_r/c  1.294090
6  240.1600  lnn_r/c  1.359904
7  174.1167  lnn_r/c  1.396697
8  141.1967  lnn_r/c  1.466330
9  125.9900  lnn_r/c  1.499301
```

[9]:

```python
plot = (ggplot(df2, aes(x='t', y='value', color='variable')) +
        geom_point(size = 2, alpha = 0.75) +
        geom_smooth(method='lm', se=False, size = 0.5, fullrange = True) +
        theme_bw() +
        labs(x = 'Time (s)', y = 'value') +
         expand_limits(y=(0, None))
        )

print(plot)
```


value
2
1
0
150
200
250
300
Time (s)
variable
lnn_r/c
n_sp/c