

Excercise 1, Part 2 (Aina Brichs Ralló, Balin Lin)

Take the file example1.txt (1.5 Gb) and fit the data by a simple regression model, estimating also the standard errors of the parameters and the variance of the errors. Investigate which is the number of chunks optimum to minimize the execution time.

```
[2]: # Importing the Python libraries
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
import sklearn.metrics as metrics
import matplotlib.pyplot as plt
from numpy.linalg import matrix_rank, inv
import sklearn.metrics as metrics
import math
%matplotlib inline

[3]: def regression_results(y_true, y_pred):

    # Regression metrics
    explained_variance = metrics.explained_variance_score(y_true, y_pred)
    r2 = metrics.r2_score(y_true, y_pred)
    mean_absolute_error = metrics.mean_absolute_error(y_true, y_pred)
    mse = metrics.mean_squared_error(y_true, y_pred)
    mean_squared_log_error = metrics.mean_squared_log_error(y_true, y_pred)
    median_absolute_error = metrics.median_absolute_error(y_true, y_pred)

    print('explained_variance: ', round(explained_variance, 4))
    print('r2: ', round(r2, 4))
    print('MAE: ', round(mean_absolute_error, 4))
    print('MSE: ', round(mse, 4))
    print('RMSE: ', round(np.sqrt(mse), 4))
    print('mean_squared_log_error: ', round(mean_squared_log_error, 4))
    print('median_absolute_error: ', round(median_absolute_error, 4))

[4]: df = pd.read_csv("example1.txt", sep=" ")

[5]: df.head()

[5]:      y      x
0  681.91  48.48
1  666.83  47.49
2  657.71  47.33
3  602.31  42.27
4  641.55  47.13

[6]: df.describe()

[6]:      y      x
count  1.100000e+08  1.100000e+08
```

```

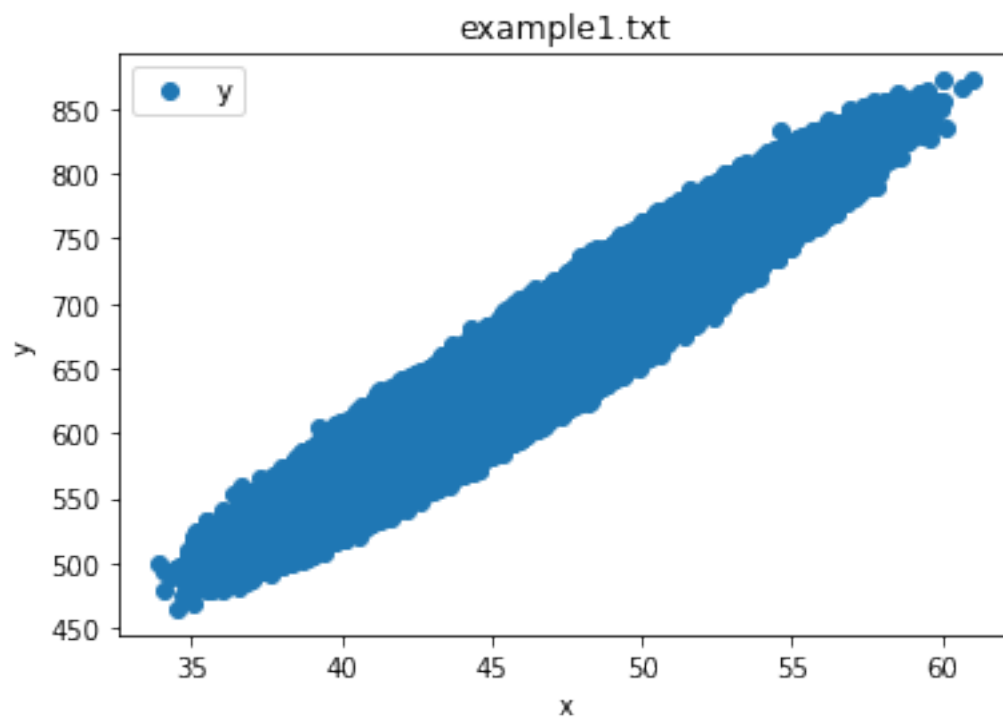
mean    6.396723e+02  4.545452e+01
std      3.789827e+01  2.462835e+00
min      4.650700e+02  3.395000e+01
25%      6.146000e+02  4.380000e+01
50%      6.356800e+02  4.525000e+01
75%      6.589900e+02  4.683000e+01
max      8.718700e+02  6.099000e+01

```

```

[7]: df.plot(x='x', y='y', style='o')
plt.title('example1.txt')
plt.xlabel('x')
plt.ylabel('y')
plt.show()

```



```

[5]: # Without Segmentation

```

```

x = df['x'].to_numpy().reshape((-1, 1))
y = df['y'].to_numpy()

```

```

[9]: model = LinearRegression().fit(x, y)

```

```

[10]: # coefficient of determination ([U+FFFD] [U+FFFD]2)
r_sq = model.score(x, y)
print('coefficient of determination:', r_sq)

```

```

coefficient of determination: 0.9214376422886941

```

```
[11]: # The attributes of model are .intercept_, which represents the coefficient,  $\rightarrow [U+FFFD][U+FFFD]_0$  and .coef_, which represents  $[U+FFFD][U+FFFD]_1$ :
# You can notice that .intercept_ is a scalar, while .coef_ is an array.
print('intercept:', model.intercept_)
print('slope:', model.coef_)
```

```
intercept: -31.74756121716905
slope: [14.77124576]
```

```
[12]: y_pred = model.predict(x)
print('predicted response:', y_pred, sep='\n')
```

```
predicted response:
[684.36243332 669.73890001 667.37550069 ... 711.83695044 677.1245229
 710.50753832]
```

```
[13]: y_pred = model.intercept_ + model.coef_ * x
print('predicted response:', y_pred, sep='\n')
```

```
predicted response:
[[684.36243332]
 [669.73890001]
 [667.37550069]
 ...
 [711.83695044]
 [677.1245229 ]
 [710.50753832]]
```

```
[14]: print('y:', y, y.shape)
print('y_pred:', y_pred, y_pred.shape)
```

```
y: [681.91 666.83 657.71 ... 726.61 682.3 724.07] (110000000,)
y_pred: [[684.36243332]
 [669.73890001]
 [667.37550069]
 ...
 [711.83695044]
 [677.1245229 ]
 [710.50753832]] (110000000, 1)
```

```
[15]: r = y_pred.reshape(-1) - y
print('r:', r)
```

```
r: [ 2.45243332  2.90890001  9.66550069 ... -14.77304956 -5.1754771
 -13.56246168]
```

```
[16]: n = len(df)
m = matrix_rank(x)
variance_pow2 = sum(r ** 2) / (n - m)
variance = math.sqrt(variance_pow2)
print('(n, m, variance_pow2, variance):', n, m, variance_pow2, variance)

(n, m, variance_pow2, variance): 110000000 1 112.83745877469549
10.622497765341986
```

```
[17]: # standard_error = math.sqrt(np.diag(variance_pow2 * inv(np.transpose(x) * x)))
```

```
[18]: regression_results(y, y_pred.reshape(-1))
```

```
explained_variance: 0.9214
r2: 0.9214
MAE: 8.4656
MSE: 112.8375
RMSE: 10.6225
mean_squared_log_error: 0.0003
median_absolute_error: 7.1433
```

```
[19]: from statsmodels.api import OLS
OLS(y, y_pred.reshape(-1)).fit().summary()
```

```
[19]: <class 'statsmodels.iolib.summary.Summary'>
"""
                                OLS Regression Results
=====
=====
Dep. Variable:                  y    R-squared (uncentered):
1.000
Model:                        OLS    Adj. R-squared (uncentered):
1.000
Method:                    Least Squares    F-statistic:
4.002e+11
Date:                Sun, 27 Feb 2022    Prob (F-statistic):
0.00
Time:                16:51:25    Log-Likelihood:
-4.1601e+08
No. Observations:        110000000    AIC:
8.320e+08
Df Residuals:            109999999    BIC:
8.320e+08
Df Model:                    1
Covariance Type:            nonrobust
=====
=====
                                coef    std err          t      P>|t|      [0.025    0.975]
-----
-----
```

x1	1.0000	1.58e-06	6.33e+05	0.000	1.000	1.000
=====						
Omnibus:		55895.835	Durbin-Watson:		1.851	
Prob(Omnibus):		0.000	Jarque-Bera (JB):		56058.017	
Skew:		0.054	Prob(JB):		0.00	
Kurtosis:		3.027	Cond. No.		1.00	

=====

Notes:

[1] R^2 is computed without centering (uncentered) since the model does not contain a constant.

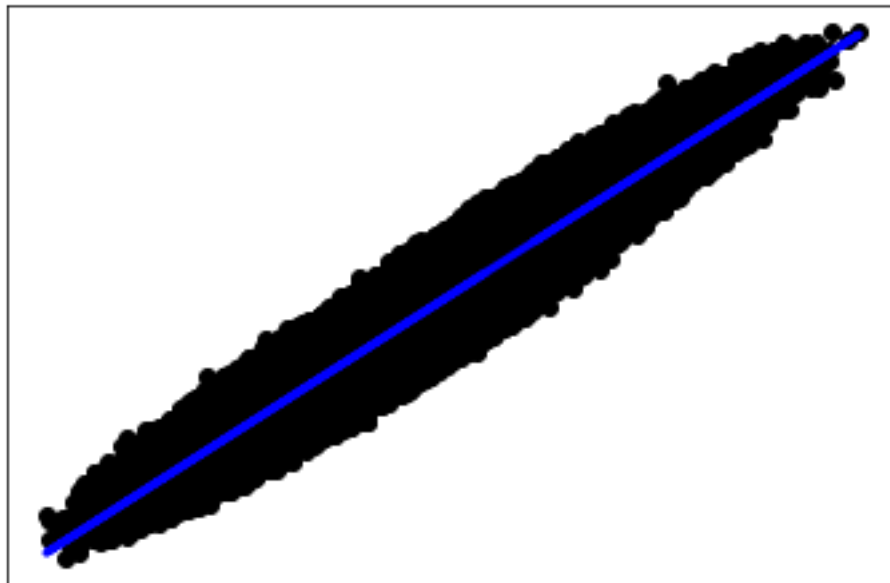
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.

""

```
[20]: # Plot outputs
plt.scatter(x, y, color="black")
plt.plot(x, y_pred, color="blue", linewidth=3)

plt.xticks(())
plt.yticks(())

plt.show()
```



```
[56]: # With Segmentation
```

```
import time
```

```

y_plot = []
x_plot = []

for i in range(1, 100, 1):
    start_time = time.time()
    div = math.ceil(len(df) / i)
    xtx = 0
    xty = 0
    x_all = []
    y_all = []
    for j in range(0, len(df), div):
        minidx = min(len(df), j + div)

        x_div = df['x'][j:minidx].to_numpy().reshape((-1, 1))
        y_div = df['y'][j:minidx].to_numpy()

        model = LinearRegression().fit(x_div, y_div)

        matrix = np.ones((len(x_div), 2))
        matrix[:, 1] = x_div.reshape(-1)

        xtx = xtx + np.transpose(matrix) @ matrix
        xty = xty + np.transpose(matrix) @ y_div

        x_all.append(x_div)
        y_all.append(y_div)

    xtx_inv = inv(xtx)
    beta = xtx_inv @ xty

    rss_div = 0
    for idx in range(len(x_all)):
        y_pred_div = model.predict(x_all[idx])
        r_div = y_all[idx] - y_pred_div
        rss_div += np.transpose(r_div) @ r_div

    n = len(df)
    m = matrix_rank(df['x'])
    variance_pow2_div = rss_div / (n - m)
    variance_div = math.sqrt(variance_pow2_div)
    standard_error = np.sqrt(np.diag(variance_pow2_div * xtx_inv))

    print("--- i = {0}, {1} seconds ---".format(i, time.time() - start_time))
    print("beta:", beta)
    print('(n, m, variance_pow2_div, variance_div):', n, m, variance_pow2_div,
    ↪variance_div)
    print('standard_error:', standard_error)

```

```
x_plot.append(i)
y_plot.append(time.time() - start_time)
```

```
--- i = 1, 18.46971082687378 seconds ---
beta: [-31.7475612  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 112.837458774905
10.622497765351849
standard_error: [0.01872012 0.00041124]
--- i = 2, 11.155757188796997 seconds ---
beta: [-31.74756125  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 114.55619148776523
10.703092613248062
standard_error: [0.01886216 0.00041436]
--- i = 3, 6.7759459018707275 seconds ---
beta: [-31.7475612  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 117.10231816399846
10.821382451609335
standard_error: [0.01907062 0.00041894]
--- i = 4, 6.731182813644409 seconds ---
beta: [-31.7475612  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 119.48741438507001
10.931029886752208
standard_error: [0.01926385 0.00042318]
--- i = 5, 6.756405591964722 seconds ---
beta: [-31.74756121  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 121.76144335841987
11.034556781240463
standard_error: [0.0194463  0.00042719]
--- i = 6, 6.746172904968262 seconds ---
beta: [-31.74756121  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 124.25040351587161
11.146766504949838
standard_error: [0.01964405 0.00043154]
--- i = 7, 5.0918238162994385 seconds ---
beta: [-31.74756121  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 127.51072322807053
11.292064613172851
standard_error: [0.01990011 0.00043716]
--- i = 8, 4.958791017532349 seconds ---
beta: [-31.74756119  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 132.7893315511929
11.523425339333478
standard_error: [0.02030784 0.00044612]
--- i = 9, 4.903936147689819 seconds ---
beta: [-31.74756121  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 143.37842190126722
11.974072903622528
```

```

standard_error: [0.02110202 0.00046356]
--- i = 10, 4.928837776184082 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 170.8394184040048
13.070555397687
standard_error: [0.02303436 0.00050601]
--- i = 11, 4.8861308097839355 seconds ---
beta: [-31.74756121 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.561001561932
16.660162110913927
standard_error: [0.02936035 0.00064498]
--- i = 12, 4.930521726608276 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.5292372333453
16.65920878173226
standard_error: [0.02935867 0.00064495]
--- i = 13, 4.902444124221802 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.3900441498971
16.65503059588595
standard_error: [0.02935131 0.00064478]
--- i = 14, 4.35237979888916 seconds ---
beta: [-31.74756121 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.42544355326555
16.656093286039965
standard_error: [0.02935318 0.00064482]
--- i = 15, 4.3148651123046875 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.4071513721267
16.65554416319463
standard_error: [0.02935221 0.0006448 ]
--- i = 16, 4.2975077629089355 seconds ---
beta: [-31.74756121 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.30854478422157
16.65258372698428
standard_error: [0.029347 0.00064469]
--- i = 17, 4.262519121170044 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.38313400786285
16.65482314549941
standard_error: [0.02935094 0.00064478]
--- i = 18, 4.281831741333008 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.325709064241
16.65309908288067
standard_error: [0.0293479 0.00064471]
--- i = 19, 4.28321099281311 seconds ---
beta: [-31.74756122 14.77124576]

```



```

(n, m, variance_pow2_div, variance_div): 110000000 1 277.1867217981996
16.648925544857228
standard_error: [0.02934055 0.00064455]
--- i = 20, 4.2891881465911865 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.18229342619463
16.648792551599488
standard_error: [0.02934032 0.00064454]
--- i = 21, 4.316877126693726 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2233312297709
16.65002496183627
standard_error: [0.02934249 0.00064459]
--- i = 22, 4.291399955749512 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.248143642764
16.65077006155463
standard_error: [0.0293438 0.00064462]
--- i = 23, 4.353016138076782 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2939025541317
16.652144082794013
standard_error: [0.02934622 0.00064467]
--- i = 24, 4.338202953338623 seconds ---
beta: [-31.74756121 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.18268443346886
16.648804294407118
standard_error: [0.02934034 0.00064454]
--- i = 25, 4.346697092056274 seconds ---
beta: [-31.74756121 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2807366290592
16.651748755883247
standard_error: [0.02934553 0.00064466]
--- i = 26, 4.33784294128418 seconds ---
beta: [-31.74756121 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.17342605121297
16.648526242620186
standard_error: [0.02933985 0.00064453]
--- i = 27, 4.351030111312866 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2270783197726
16.650137486512616
standard_error: [0.02934269 0.00064459]
--- i = 28, 4.298534870147705 seconds ---
beta: [-31.74756121 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2008233886285
16.649349037984294
standard_error: [0.0293413 0.00064456]

```

```

--- i = 29, 4.298187971115112 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.1631906962037
16.648218844555224
standard_error: [0.0293393  0.00064452]
--- i = 30, 4.309473037719727 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.1426149688351
16.647600877268626
standard_error: [0.02933822 0.0006445 ]
--- i = 31, 4.299461126327515 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.155029943375
16.64797374887932
standard_error: [0.02933887 0.00064451]
--- i = 32, 4.297496795654297 seconds ---
beta: [-31.74756121  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.14248444243816
16.647596956991666
standard_error: [0.02933821 0.0006445 ]
--- i = 33, 4.2761640548706055 seconds ---
beta: [-31.74756121  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.13588379623263
16.647398709595222
standard_error: [0.02933786 0.00064449]
--- i = 34, 4.288546085357666 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.21196729500696
16.649683699548376
standard_error: [0.02934189 0.00064458]
--- i = 35, 4.3027119636535645 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.16896057255605
16.648392131751223
standard_error: [0.02933961 0.00064453]
--- i = 36, 4.317367792129517 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.19448259480527
16.64915861522153
standard_error: [0.02934096 0.00064456]
--- i = 37, 4.319252014160156 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.16597328887747
16.648302414627068
standard_error: [0.02933945 0.00064452]
--- i = 38, 4.313910007476807 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.17398057728764

```

```

16.648542896520635
standard_error: [0.02933988 0.00064453]
--- i = 39, 4.356665134429932 seconds ---
beta: [-31.74756121 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.21119986797146
16.649660653237696
standard_error: [0.02934185 0.00064458]
--- i = 40, 4.345142126083374 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.22174958249104
16.649977464924422
standard_error: [0.0293424 0.00064459]
--- i = 41, 4.348558187484741 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.20003903541635
16.649325482896188
standard_error: [0.02934125 0.00064456]
--- i = 42, 4.355313062667847 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.22052516157913
16.6499406954373
standard_error: [0.02934234 0.00064459]
--- i = 43, 4.401844263076782 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2683906986933
16.651378042032835
standard_error: [0.02934487 0.00064464]
--- i = 44, 4.3970701694488525 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.3034075716773
16.65242947955875
standard_error: [0.02934672 0.00064468]
--- i = 45, 4.3805131912231445 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.1948695420279
16.64917023584142
standard_error: [0.02934098 0.00064456]
--- i = 46, 4.391687870025635 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.17722980095385
16.64864047905876
standard_error: [0.02934005 0.00064454]
--- i = 47, 4.422831773757935 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.201000628798
16.64935436071916
standard_error: [0.02934131 0.00064456]
--- i = 48, 4.407738208770752 seconds ---

```

```

beta: [-31.74756121  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.1781830745404
16.648669108206228
standard_error: [0.0293401  0.00064454]
--- i = 49, 4.416971921920776 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2280556732209
16.650166836197798
standard_error: [0.02934274 0.0006446 ]
--- i = 50, 4.425450086593628 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.18568595276156
16.648894436351068
standard_error: [0.02934049 0.00064455]
--- i = 51, 4.45092511177063 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.08668288342415
16.64592090824128
standard_error: [0.02933525 0.00064443]
--- i = 52, 4.448307991027832 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.0291538647658
16.644192797031817
standard_error: [0.02933221 0.00064436]
--- i = 53, 4.46961522102356 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.06783444248475
16.645354740662174
standard_error: [0.02933426 0.00064441]
--- i = 54, 4.459146976470947 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.1058052852254
16.64649528535137
standard_error: [0.02933627 0.00064445]
--- i = 55, 4.46758508682251 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.19501324568324
16.64917455148102
standard_error: [0.02934099 0.00064456]
--- i = 56, 4.407857894897461 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.19828169781556
16.649272707773623
standard_error: [0.02934116 0.00064456]
--- i = 57, 4.316066026687622 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.1707099744249
16.648444671332662

```

```

standard_error: [0.0293397  0.00064453]
--- i = 58, 4.322872161865234 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.23996842285516
16.650524568999476
standard_error: [0.02934337 0.00064461]
--- i = 59, 4.335929870605469 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.27748670488626
16.65165117052619
standard_error: [0.02934535 0.00064465]
--- i = 60, 4.298932075500488 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.30608602904795
16.652509901785013
standard_error: [0.02934687 0.00064469]
--- i = 61, 4.302288055419922 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.30861711365185
16.652585898702096
standard_error: [0.029347  0.00064469]
--- i = 62, 4.2961390018463135 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.3029126342782
16.652414618735573
standard_error: [0.0293467  0.00064468]
--- i = 63, 4.308485269546509 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.33464979773373
16.65336752124728
standard_error: [0.02934838 0.00064472]
--- i = 64, 4.209300994873047 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2547605070878
16.650968755813814
standard_error: [0.02934415 0.00064463]
--- i = 65, 4.2056567668914795 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2890280047774
16.651997718135124
standard_error: [0.02934596 0.00064467]
--- i = 66, 4.203208923339844 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.24181864587064
16.650580129409025
standard_error: [0.02934347 0.00064461]
--- i = 67, 4.214802026748657 seconds ---
beta: [-31.74756122  14.77124576]

```

```

(n, m, variance_pow2_div, variance_div): 110000000 1 277.33140157110375
16.653269996343173
standard_error: [0.02934821 0.00064472]
--- i = 68, 4.2030699253082275 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.3864143432119
16.654921625249756
standard_error: [0.02935112 0.00064478]
--- i = 69, 4.221116065979004 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.4365822707645
16.65642765633629
standard_error: [0.02935377 0.00064484]
--- i = 70, 4.216807126998901 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.5016460465236
16.658380654989354
standard_error: [0.02935721 0.00064491]
--- i = 71, 4.224998950958252 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.37830452514856
16.654678157357125
standard_error: [0.02935069 0.00064477]
--- i = 72, 4.2193310260772705 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.32400219460106
16.65304783499408
standard_error: [0.02934781 0.00064471]
--- i = 73, 4.220609903335571 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.3273835345032
16.653149357839293
standard_error: [0.02934799 0.00064471]
--- i = 74, 4.227190017700195 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.30837248067473
16.652578553505602
standard_error: [0.02934699 0.00064469]
--- i = 75, 4.222815990447998 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.31064477039814
16.652646779728386
standard_error: [0.02934711 0.00064469]
--- i = 76, 4.233040809631348 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.32006118883777
16.652929507712383
standard_error: [0.02934761 0.0006447 ]

```

```

--- i = 77, 4.232987880706787 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.2082317848993
16.649571519558673
standard_error: [0.02934169 0.00064457]
--- i = 78, 4.246238946914673 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.147086033124
16.647735162271292
standard_error: [0.02933845 0.0006445 ]
--- i = 79, 4.264038801193237 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.170627689426
16.648442200080645
standard_error: [0.0293397  0.00064453]
--- i = 80, 4.252377033233643 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.189931420955
16.649021935866234
standard_error: [0.02934072 0.00064455]
--- i = 81, 4.256382942199707 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.18999887842824
16.64902396173506
standard_error: [0.02934072 0.00064455]
--- i = 82, 4.257929801940918 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.0908133642224
16.646044976637015
standard_error: [0.02933547 0.00064444]
--- i = 83, 4.262908220291138 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.9782665789326
16.642664046928683
standard_error: [0.02932952 0.0006443 ]
--- i = 84, 4.264907121658325 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.9851044462949
16.642869477535864
standard_error: [0.02932988 0.00064431]
--- i = 85, 4.271947145462036 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.9621069783977
16.642178552653426
standard_error: [0.02932866 0.00064429]
--- i = 86, 4.265393018722534 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.9753208536524

```

```

16.6425755474822
standard_error: [0.02932936 0.0006443 ]
--- i = 87, 4.267472982406616 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.00654177189335
16.643513504422476
standard_error: [0.02933101 0.00064434]
--- i = 88, 4.273285865783691 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.98199328079085
16.642776008851133
standard_error: [0.02932971 0.00064431]
--- i = 89, 4.276233911514282 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.96248266623707
16.642189839868944
standard_error: [0.02932868 0.00064429]
--- i = 90, 4.284337997436523 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.9075807003503
16.640540276696257
standard_error: [0.02932577 0.00064422]
--- i = 91, 4.293184995651245 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.9187396725956
16.640875568088227
standard_error: [0.02932636 0.00064424]
--- i = 92, 4.27630615234375 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 277.02941654026415
16.644200687935246
standard_error: [0.02933222 0.00064436]
--- i = 93, 4.282340049743652 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.9187979600365
16.640877319421488
standard_error: [0.02932637 0.00064424]
--- i = 94, 4.2920989990234375 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.82899593242144
16.63817886466008
standard_error: [0.02932161 0.00064413]
--- i = 95, 4.294445037841797 seconds ---
beta: [-31.74756122 14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.6759888888754
16.633580158488893
standard_error: [0.02931351 0.00064395]
--- i = 96, 4.315028190612793 seconds ---

```



```

beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.7354514486843
16.63536748763562
standard_error: [0.02931666 0.00064402]
--- i = 97, 4.29867696762085 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.79008350924215
16.63700945209932
standard_error: [0.02931955 0.00064409]
--- i = 98, 4.307709217071533 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.685280534555
16.633859459985675
standard_error: [0.029314  0.00064396]
--- i = 99, 4.316742181777954 seconds ---
beta: [-31.74756122  14.77124576]
(n, m, variance_pow2_div, variance_div): 110000000 1 276.58678758566913
16.63089858022317
standard_error: [0.02930878 0.00064385]

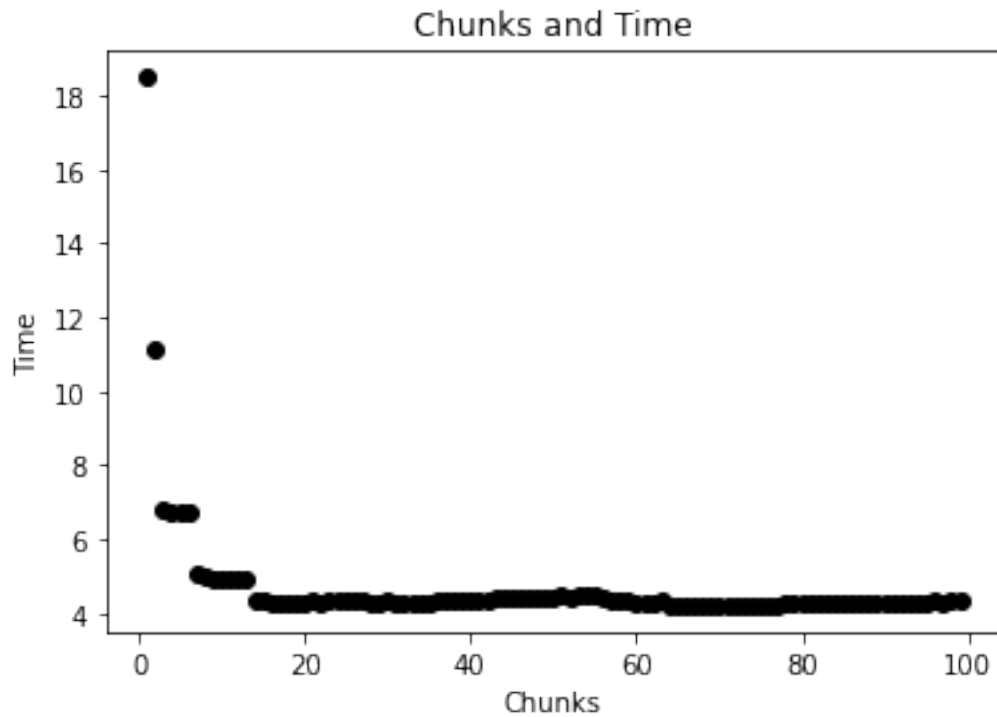
```

```

[57]: # Plot outputs
plt.scatter(x_plot, y_plot, color="black")
plt.title('Chunks and Time')

plt.xlabel('Chunks')
plt.ylabel('Time')
plt.show()

```



```
[62]: y_plot.index(min(y_plot)) + 1
```

```
[62]: 66
```

Answer:

Without Segmentation:

- Standard errors of the parameters: [0.01872012 0.00041124]
- Variance of the errors: 112.837458774905

With Segmentation:

- Investigate which is the number of chunks optimum to minimize the execution time: 66
- Standard errors of the parameters: [0.02934596 0.00064467]
- Variance of the errors: 277.2890280047774