Examples_Chapter5

October 6, 2021

1 Example 5.1 of the textbook

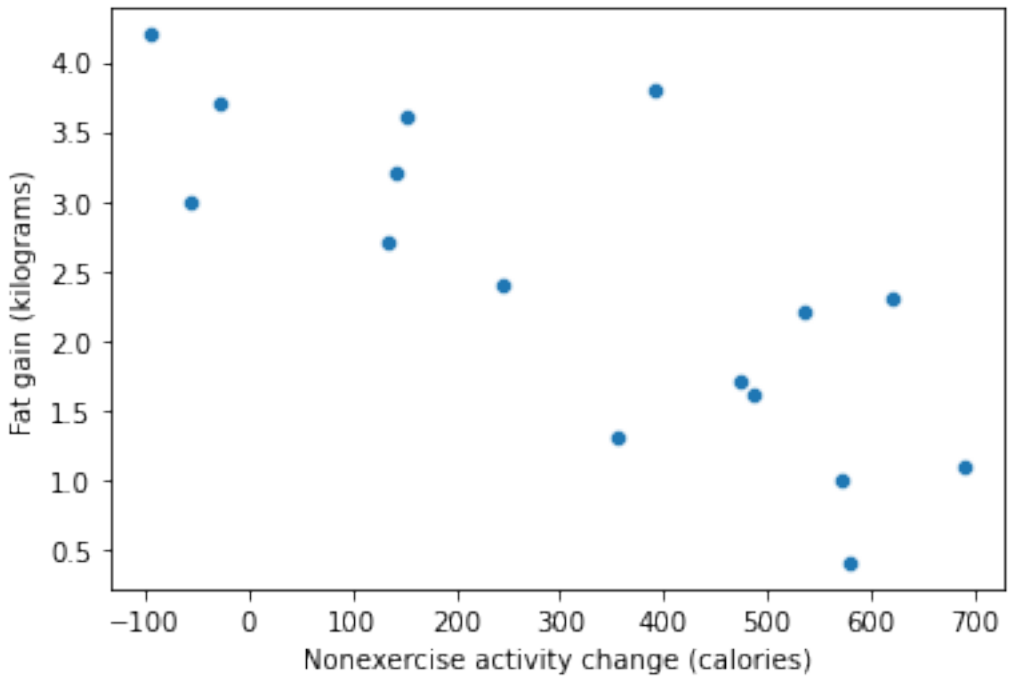
[1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from scipy import stats
```

[2]:
```
fatgain = pd.read_csv("eg05-01fatgain.csv")
fatgain.head()
```

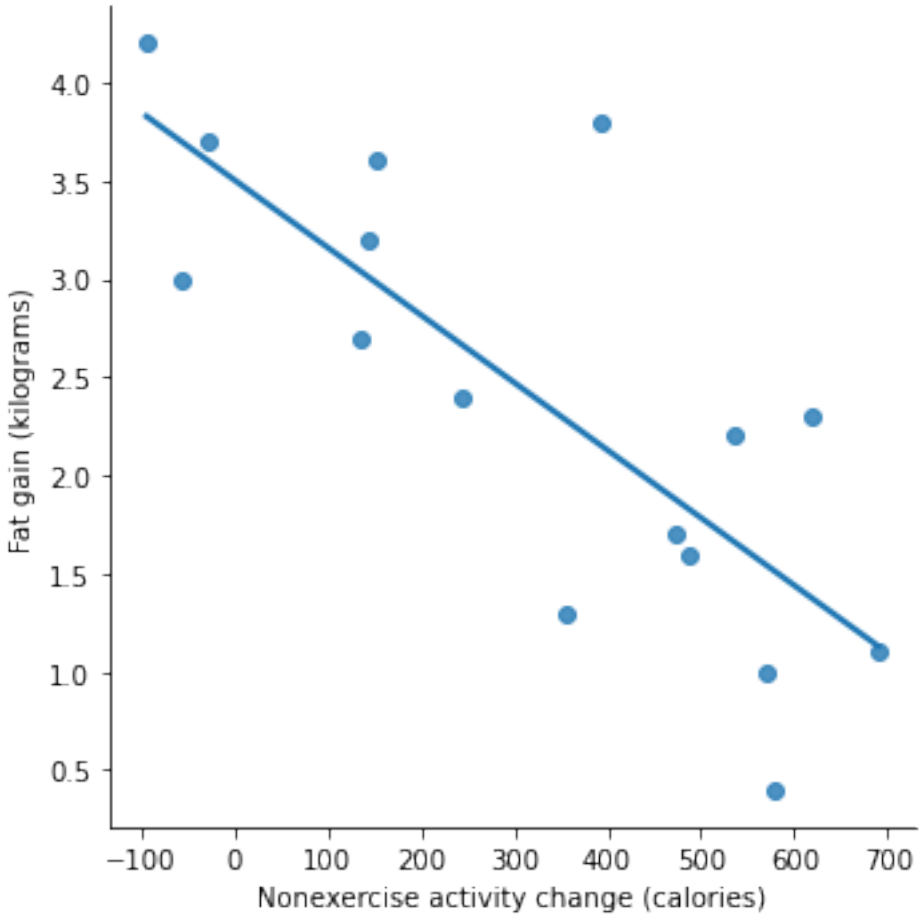
[2]:

	NEA	Fat
0	-94	4.2
1	-57	3.0
2	-29	3.7
3	135	2.7
4	143	3.2

[3]:
```
sns.scatterplot(x = "NEA", y = "Fat", data = fatgain)
plt.xlabel("Nonexercise activity change (calories)")
plt.ylabel("Fat gain (kilograms)")
plt.show()
```



[4]:
```
# sns.lmplot adds to a scatterplot the regression line that best fits the data␣
 ↪points
sns.lmplot(x = "NEA", y = "Fat", data = fatgain, ci=None)
plt.xlabel("Nonexercise activity change (calories)")
plt.ylabel("Fat gain (kilograms)")
plt.show()
```



[5]:
```
lm = stats.linregress(x=fatgain['NEA'],y=fatgain['Fat'])
lm.intercept # intercept
```

[5]: 3.5051229156310724

[6]:
```
lm.slope # slope
```

[6]: -0.0034414870381249342

[7]:
```
lm.rvalue # correlation
```

[7]: -0.7785558457058473

[8]:
```
lm.rvalue**2 # r^2
```

[8]: 0.6061492048827472

[9]:
```
# what would be the fat gain for someone with 400 calories in NEA change?
yhat = lm.intercept + lm.slope*400
yhat
```

[9]:
```
2.1285281003810987
```

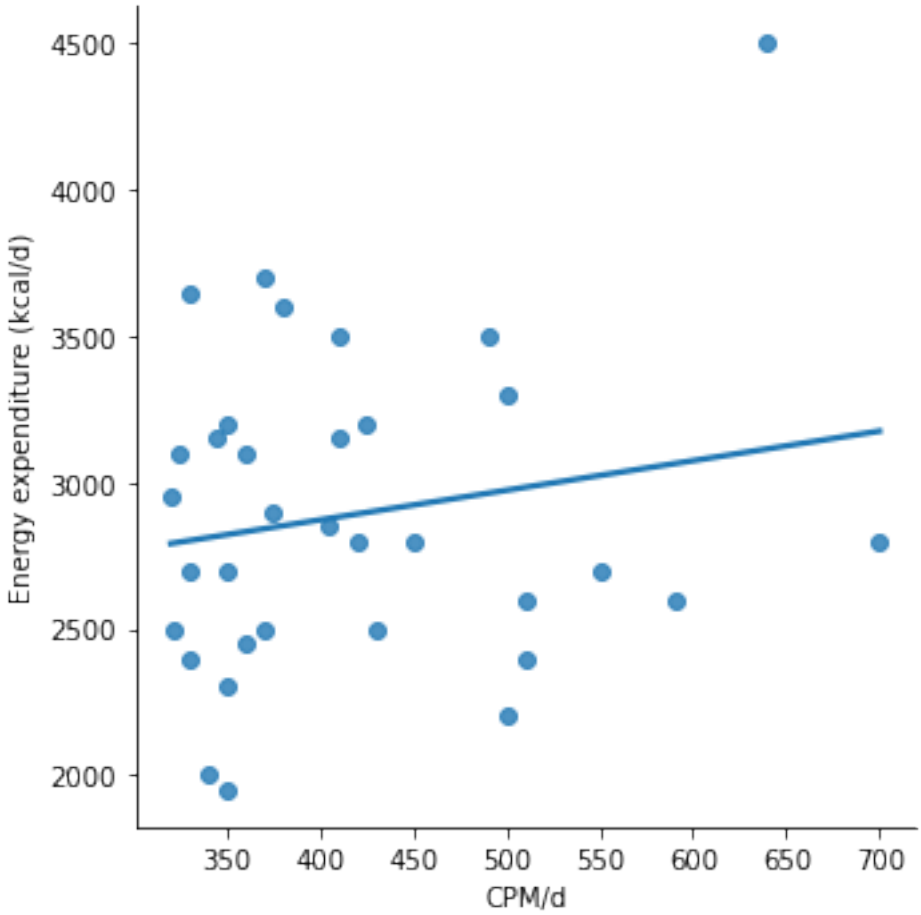
2 Example 5.5 of the textbook

[10]:
```
exercise = pd.read_csv("eg05-05exerc.csv")
exercise.head()
```

[10]:

	Subject	CPM/d	EnergyExpenditure
0	1	700	2800
1	2	640	4500
2	3	590	2600
3	4	550	2700
4	5	510	2400

[11]:
```
sns.lmplot(x = "CPM/d", y = "EnergyExpenditure", data = exercise,ci=None)
plt.xlabel("CPM/d")
plt.ylabel("Energy expenditure (kcal/d)")
plt.show()
```

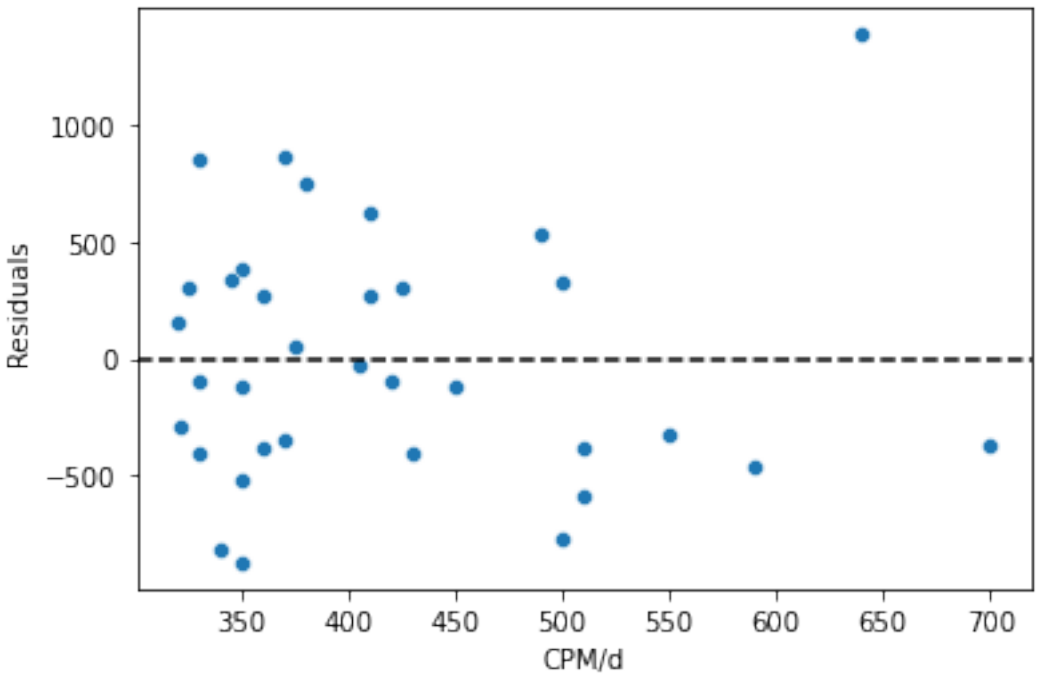


[12]:
```
lm2 = stats.linregress(x=exercise['CPM/d'],y=exercise['EnergyExpenditure'])
print(lm2.intercept) # intercept
print(lm2.slope) #slope
```

```
2467.549357864118
1.011082124022258
```

[13]:
```
#calculating the residuals
x = exercise['CPM/d']
y_hat = lm2.intercept + lm2.slope*x ## predicted energy expenditures
y = exercise['EnergyExpenditure'] ## observed energy expenditures
residuals = y - y_hat
```

[14]:
```
sns.scatterplot(x = exercise['CPM/d'], y = residuals)
plt.xlabel('CPM/d')
plt.ylabel("Residuals")
plt.axhline(y=0,color="black",linestyle="--")
plt.show()
```


1000
500
0
−500
Residuals
350
400
450
500
550
600
650
700
CPM/d

[ ]: