# ipynb

June 28, 2025

[1]:
```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt
```

[2]:
```
np.random.seed(42)

# Wiek, BMI, aktywność fizyczna, spożycie kalorii, liczba godzin snu
X = np.random.rand(1000, 5) * [80, 40, 2, 3000, 10]

# Poziom cukru, ciśnienie skurczowe, ciśnienie rozkurczowe
y = np.random.rand(1000, 3) * [200, 180, 120]
```

[3]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
  ↪random_state=42)
```

[4]:
```
model = LinearRegression()
model.fit(X_train, y_train)
```

[4]:
```
LinearRegression()
```

[5]:
```
y_pred = model.predict(X_test)
```

[6]:
```
mse = mean_squared_error(y_test, y_pred, multioutput='raw_values')
r2 = r2_score(y_test, y_pred, multioutput='raw_values')

print("Mean Squared Error (cukier, skurczowe, rozkurczowe):", mse)
print("R^2 Score (cukier, skurczowe, rozkurczowe):", r2)
```

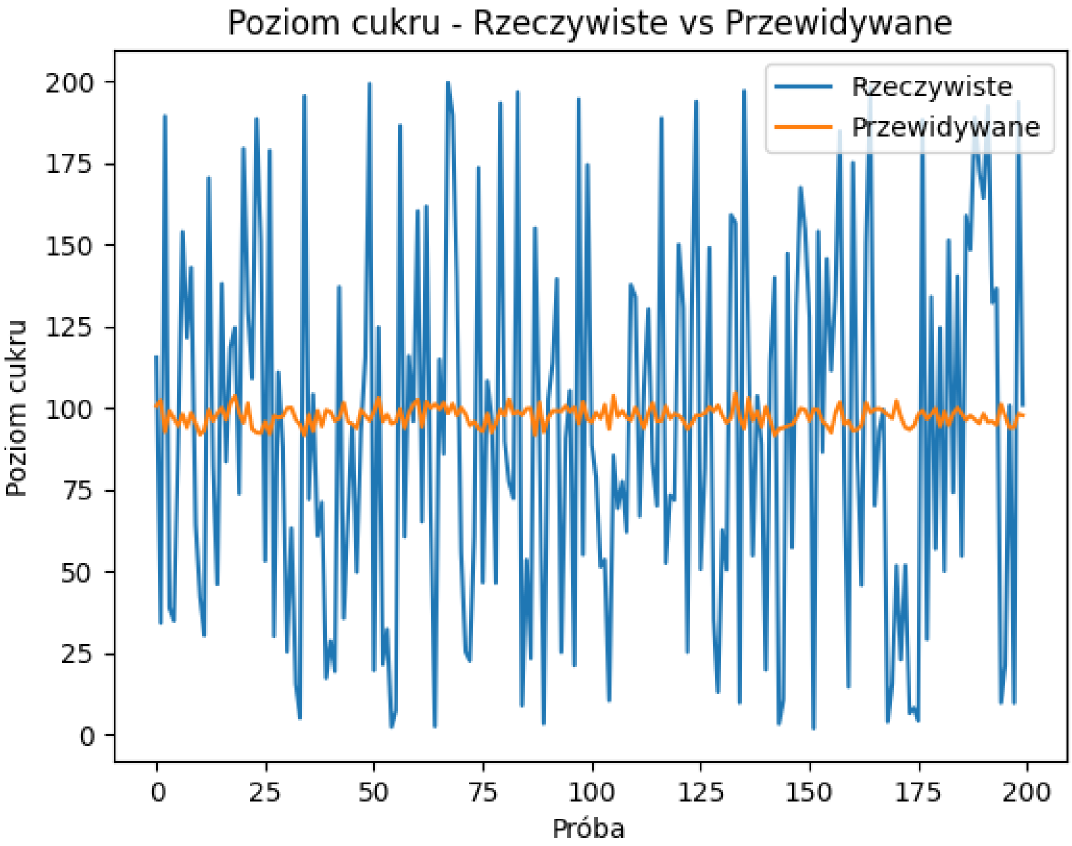
```
Mean Squared Error (cukier, skurczowe, rozkurczowe): [3352.40938387
2482.28127583 1236.10357925]
R^2 Score (cukier, skurczowe, rozkurczowe): [-0.00895378 -0.00739375
-0.02209181]
```

[7]:
```
print("Rzeczywiste wartości:\n", y_test[:10])
print("Przewidywane wartości:\n", y_pred[:10])
```

```
Rzeczywiste wartości:
 [[115.38722467  27.87145104  42.74327322]
 [ 34.153652    48.09778119  61.09360622]
 [189.21512361  44.04880746  15.7878368 ]
 [ 38.82575409  96.58935167 112.17864485]
 [ 34.89969728  68.166604   101.1773838 ]
 [ 89.14596418 150.5013508   31.79604739]
 [153.75283614  47.8057677   92.0680073 ]
 [121.39462325  33.9299735   96.59602007]
 [142.87751241 164.99575848  84.65959823]
 [ 64.19940913  94.69692239  82.53230476]]
Przewidywane wartości:
 [[100.57359063  89.22104702  62.65150398]
 [102.263723    87.49461266  60.3270513 ]
 [ 92.6587479   91.75116989  53.22685562]
 [ 99.08376285  86.62942943  60.47844493]
 [ 96.74246001  91.36821944  56.57134357]
 [ 94.40975983  88.65106883  57.24190241]
 [ 98.09898255  89.86724033  58.58506873]
 [ 93.90693479  90.32692639  57.40890739]
 [ 98.49166898  89.65707746  59.7225709 ]
 [ 95.0701386   94.98254489  51.82526185]]
```

[8]:

```python
plt.figure()
plt.plot(y_test[:, 0], label='Rzeczywiste')
plt.plot(y_pred[:, 0], label='Przewidywane')
plt.title('Poziom cukru - Rzeczywiste vs Przewidywane')
plt.xlabel('Próba')
plt.ylabel('Poziom cukru')
plt.legend()
plt.show()
```


Poziom cukru - Rzeczywiste vs Przewidywane
Rzeczywiste
Przewidywane
Poziom cukru
200
175
150
125
100
75
50
25
0
0
25
50
75
100
125
150
175
200
Próba