# SLR - Penguins

April 3, 2024

[1]:
```
import pandas as pd
import seaborn as sns
```

[2]:
```
data = sns.load_dataset("penguins")
```

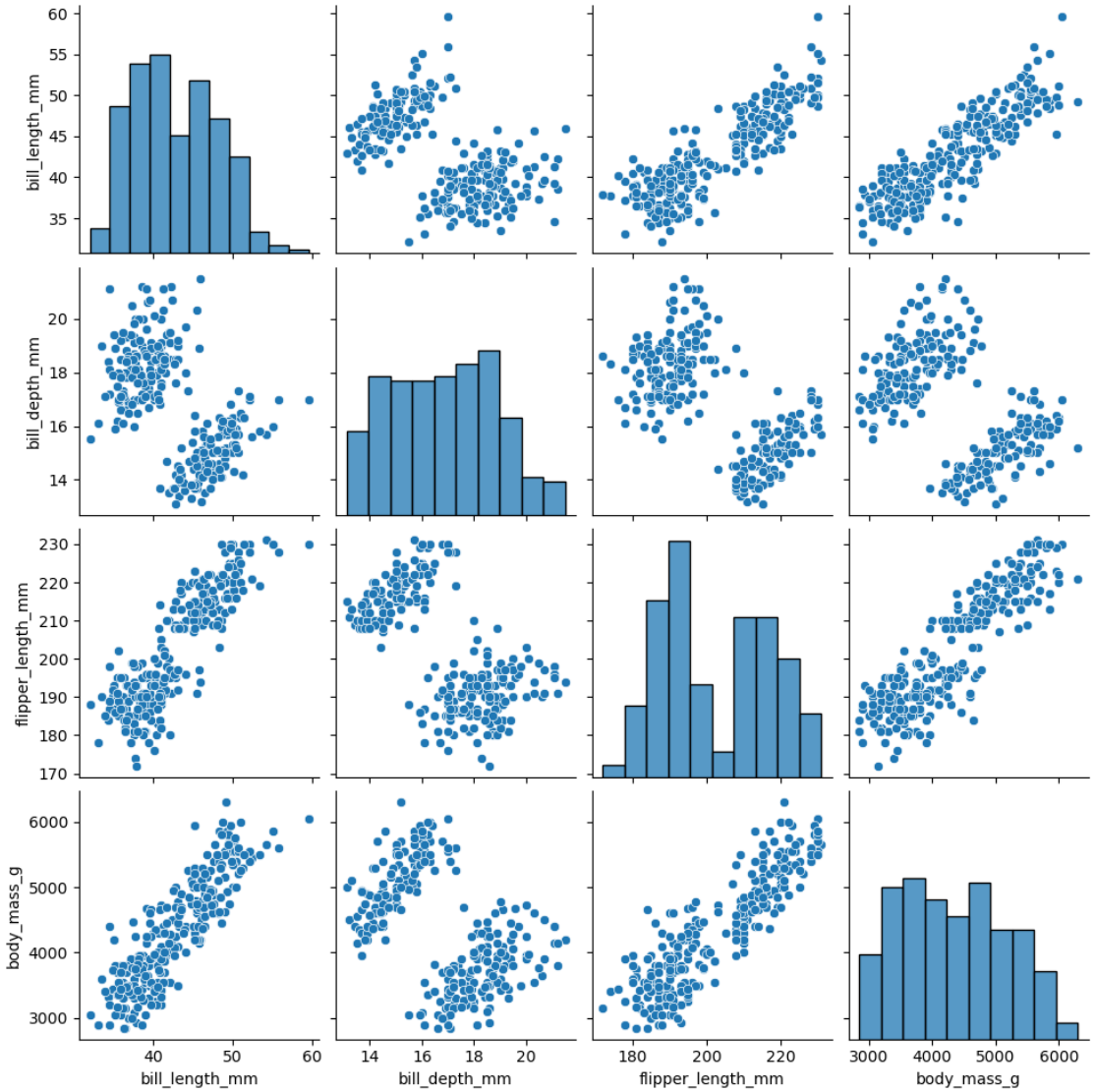
[3]:
```
data.head()
```

[3]:
```
  species     island  bill_length_mm  bill_depth_mm  flipper_length_mm  \
0  Adelie  Torgersen            39.1           18.7              181.0
1  Adelie  Torgersen            39.5           17.4              186.0
2  Adelie  Torgersen            40.3           18.0              195.0
3  Adelie  Torgersen             NaN            NaN                NaN
4  Adelie  Torgersen            36.7           19.3              193.0

   body_mass_g     sex
0       3750.0    Male
1       3800.0  Female
2       3250.0  Female
3          NaN     NaN
4       3450.0  Female
```

[4]:
```
# Keep Adelie and Gentoo penguins, drop NAs,

data_sub = data[data["species"] != "Chinstrap"]

data_final = data_sub.dropna()
data_final.reset_index(inplace = True, drop = True)
```

[5]:
```
#Scatterplot Matrix

sns.pairplot(data_final)
```

[5]:
```
<seaborn.axisgrid.PairGrid at 0x15ec117f0>
```




[6]:
```
ols_data = data_final[["bill_length_mm", "body_mass_g"]]
```

[7]:
```
ols_formula = "body_mass_g ~ bill_length_mm"
```

[8]:
```
#importing ols function

from statsmodels.formula.api import ols
```

[9]:
```
OLS = ols(formula = ols_formula, data = ols_data)
model = OLS.fit()
```

[10]:
```
model.summary()
```

[10]:

| | | | |
|---|---|---|---|
| **Dep. Variable:** | body_mass_g | **R-squared:** | 0.769 |
| **Model:** | OLS | **Adj. R-squared:** | 0.768 |
| **Method:** | Least Squares | **F-statistic:** | 874.3 |
| **Date:** | Sat, 16 Mar 2024 | **Prob (F-statistic):** | 1.33e-85 |
| **Time:** | 16:28:09 | **Log-Likelihood:** | -1965.8 |
| **No. Observations:** | 265 | **AIC:** | 3936. |
| **Df Residuals:** | 263 | **BIC:** | 3943. |
| **Df Model:** | 1 | | |
| **Covariance Type:** | nonrobust | | |

| | **coef** | **std err** | **t** | **P> \|t\|** | **[0.025** | **0.975]** |
|---|---|---|---|---|---|---|
| **Intercept** | -1707.2919 | 205.640 | -8.302 | 0.000 | -2112.202 | -1302.382 |
| **bill_length_mm** | 141.1904 | 4.775 | 29.569 | 0.000 | 131.788 | 150.592 |

| | | | |
|---|---|---|---|
| **Omnibus:** | 2.060 | **Durbin-Watson:** | 2.067 |
| **Prob(Omnibus):** | 0.357 | **Jarque-Bera (JB):** | 2.103 |
| **Skew:** | 0.210 | **Prob(JB):** | 0.349 |
| **Kurtosis:** | 2.882 | **Cond. No.** | 357. |

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

#### y = intercept + slope * x

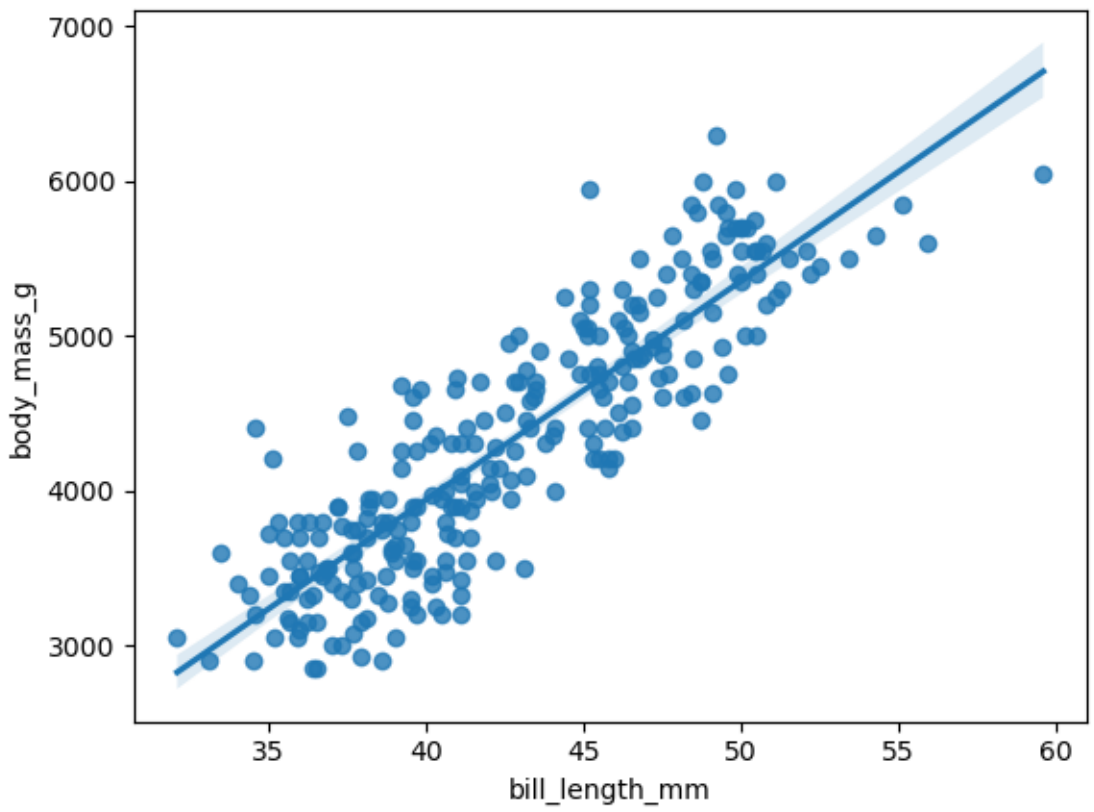
**Body mass(g) = -1707.2919 + 141.1904 * bill length (mm)**

[11]:
```
# Subset x variable

x = ols_data["bill_length_mm"]

#Getting predictions from the model
fitted_values = model.predict(x)
```

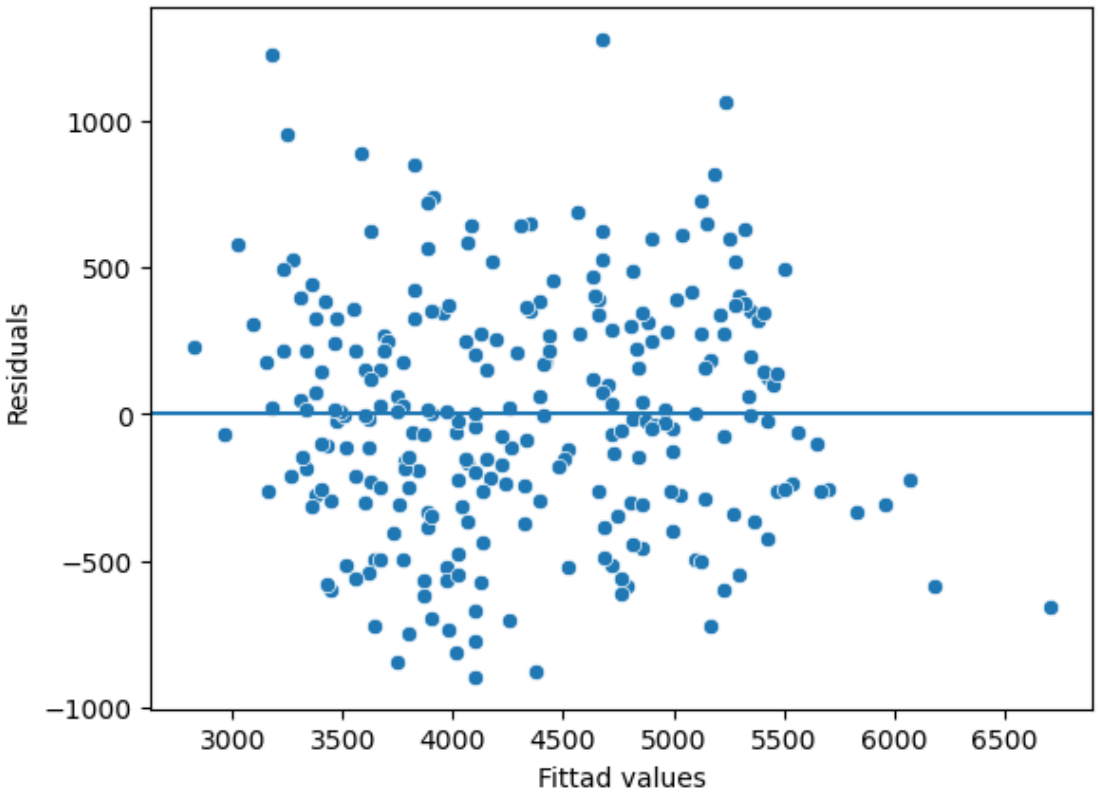
[12]:
```
#Calculate residuals

residuals = model.resid
```

[19]:
```
import matplotlib.pyplot as plt
```

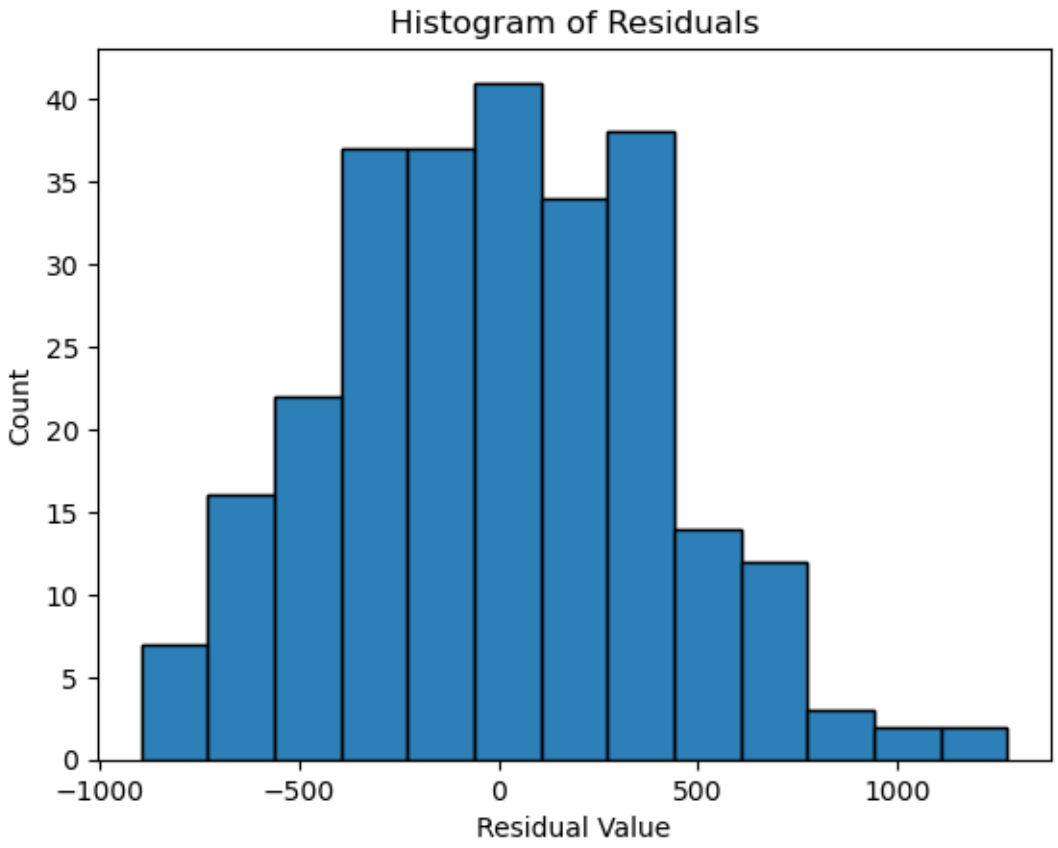
[23]:
```
sns.regplot(x = "bill_length_mm", y = "body_mass_g", data = ols_data)


plt.show()
```



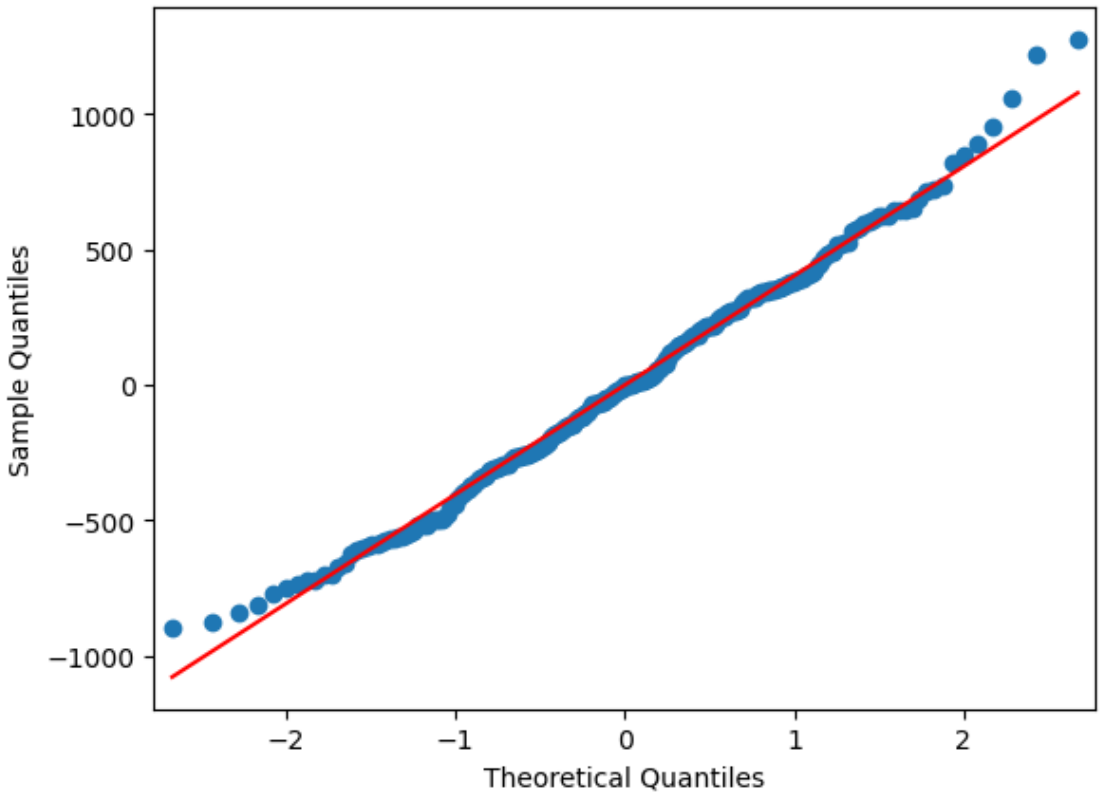

[25]:
```
fig = sns.scatterplot(x = fitted_values, y = residuals)
fig.axhline(0)
fig.set_xlabel("Fittad values")
fig.set_ylabel("Residuals")
plt.show()
```




[27]:
```
fig = sns.histplot(residuals)
fig.set_xlabel("Residual Value")
fig.set_title("Histogram of Residuals")
plt.show()
```




[28]:
```
import statsmodels.api as sm
fig = sm.qqplot(model.resid, line = 's')
plt.show()
```


1000
500
0
−500
−1000
Sample Quantiles
−2
−1
0
1
2
Theoretical Quantiles

[ ]: