Stat-Ethics-Assignment3-Solutions

May 11, 2020

1 Assignment 3 - Probability and Statistics, Ethics and privacy

This assignment is worth 20% of your final grade.

It is due at **6pm on Friday 15 May**, and should be submitted as a single Jupyter Notebook, through the usual online ECS submission system (https://apps.ecs.vuw.ac.nz/submit/DATA201) .

The assignment makes use of the `scipy.stats` package and the `statsmodels.formula.api` package. It draws on material from Weeks 4 and 5, lectured by Richard Arnold.

[90]:
```
import numpy as np
import pandas as pd
import scipy.stats as stats
import statsmodels.formula.api as smf
import matplotlib.pyplot as pl
%matplotlib inline
```

1.1 Probability Distributions from data

(12 Marks)

Read the `EuropeanBirds.csv` data set of characteristics of 500 species of European Birds, which you can download from the course website.

Note: you will need to specify `encoding='latin1'` when you read the file in.

You will also need to consult the accompany information file `EuropeanBirds-Information.txt` in order to interpret the content of the data set.

Sourced from: Storchová, Lenka; Hořák, David (2018), Data from: Life-history characteristics of European birds, Dryad, Dataset, https://datadryad.org/stash/dataset/doi:10.5061/dryad.n6k3n

[91]:
```
birds = pd.read_csv("EuropeanBirds.csv", encoding='latin1')
# list its first few rows
birds.head()
```

[91]:
```
    ID           Order        Family             Species  LengthU_MEAN  \
0  1.0  Accipitriformes  Accipitridae  Accipiter brevipes          35.0
1  2.0  Accipitriformes  Accipitridae  Accipiter gentilis          55.0
2  3.0  Accipitriformes  Accipitridae     Accipiter nisus          33.0
3  4.0  Accipitriformes  Accipitridae   Aegypius monachus         105.0
```

```
4  5.0  Accipitriformes  Accipitridae    Aquila adalberti          80.0

   WingU_MEAN  WingM_MEAN  WingF_MEAN  TailU_MEAN  TailM_MEAN  …  \
0       227.5       220.0       235.0       160.0       154.0  …
1       332.5       312.0       353.0       239.5       223.0  …
2       221.5       203.0       240.0       164.5       149.0  …
3       786.5       772.0       801.0       373.0       365.0  …
4       610.5       600.0       621.0       298.5       291.0  …

   Folivore_B  Frugivore_B  Granivore_B  Arthropods_B  Other.invertebrates_B  \
0         0.0          0.0          0.0           1.0                    0.0
1         0.0          0.0          0.0           0.0                    0.0
2         0.0          0.0          0.0           0.0                    0.0
3         0.0          0.0          0.0           0.0                    0.0
4         0.0          0.0          0.0           0.0                    0.0

   Fish_B  Other.vertebrates_B  Carrion_B  Omnivore_B  \
0     0.0                  1.0        0.0         0.0
1     0.0                  1.0        0.0         0.0
2     0.0                  1.0        0.0         0.0
3     0.0                  0.0        1.0         0.0
4     0.0                  1.0        0.0         0.0

                                        Data.source
0  (1) Cramp, S. (2006) The Birds of the Western …
1  (1) Cramp, S. (2006) The Birds of the Western …
2  (1) Cramp, S. (2006) The Birds of the Western …
3  (1) Cramp, S. (2006) The Birds of the Western …
4  (1) Cramp, S. (2006) The Birds of the Western …

[5 rows x 86 columns]
```

1. Calculate the probabilities of the various types of nest building. What is the probability that a male is involved in nest building?

[92]:
```
qtab = birds["Nest.building"].value_counts().sort_index()
qual_dist = pd.DataFrame({'counts':qtab, 'probs':qtab/sum(qtab)})
qual_dist
```

[92]:

	counts	probs
B	207	0.417339
F	224	0.451613
M	27	0.054435
N	38	0.076613

[93]:
```
p = sum(qual_dist['probs'][{'B','M'}])
```

```
print('The probability that a male is invoved in nest building is␣
↪',round(100*p,1), '%', sep="")
```

```
The probability that a male is invoved in nest building is 47.2%
```

2. How many bird species have a solely monogamous mating system?

[94]:
```
mtab = birds["Mating.system"].value_counts().sort_index()
mating_dist = pd.DataFrame({'counts':mtab, 'probs':mtab/sum(mtab)})
mating_dist
```

[94]:

	counts	probs
M	435	0.871743
M,PA	7	0.014028
M,PG	28	0.056112
M,PG,PA	3	0.006012
M,PG,PM	2	0.004008
M,PM	6	0.012024
PG	5	0.010020
PG,PA	1	0.002004
PG,PM	4	0.008016
PM	8	0.016032

[95]:
```
nn = mating_dist['counts']['M']
print('The number of species with an exclusively monogamous mating system is␣
↪',nn, sep="")
```

```
The number of species with an exclusively monogamous mating system is 435
```

3. After monogamy only, what is the next most common mating system?

Monogamy + Polyandrous

4. What is the probability that a species is Sedentary (lives in the same area in both the breeding and non-breeding season)?

[96]:
```
stab = birds["Sedentary"].value_counts().sort_index()
sedentary_dist = pd.DataFrame({'counts':stab, 'probs':stab/sum(stab)})
sedentary_dist
```

[96]:

	counts	probs
0.0	313	0.627255
1.0	186	0.372745

[97]:
```
p = sedentary_dist['probs'][1]
print('The probability that a species is Sedentary is ',round(100*p,1), '%',␣
↪sep="")
```

```
The probability that a species is Sedentary is 37.3%
```

5. What is the probability that a species is Sedentary **and** occupies human settlements in its breeding area?

[98]:
```
shtab = pd.crosstab(birds['Sedentary'], birds['Human.settlements'],␣
↪normalize='all')
shtab
```

[98]:
```
Human.settlements       0.0       1.0
Sedentary
0.0                0.599198  0.028056
1.0                0.306613  0.066132
```

[99]:
```
p = shtab[1][1]
print('The probability that a species is Sedentary and breeds in human␣
↪settlements is ',round(100*p,1), '%', sep="")
```

```
The probability that a species is Sedentary and breeds in human settlements is
6.6%
```

6. What is the probability that a Sedentary species occupies human settlements in its breeding area?

[100]:
```
shtab = pd.crosstab(birds['Sedentary'], birds['Human.settlements'],␣
↪normalize='index')
shtab
```

[100]:
```
Human.settlements       0.0       1.0
Sedentary
0.0                0.955272  0.044728
1.0                0.822581  0.177419
```

[101]:
```
p = shtab[1][1]
print('The probability that a Sedentary sepecies breeds in human settlements is␣
↪',round(100*p,1), '%', sep="")
```

```
The probability that a Sedentary sepecies breeds in human settlements is 17.7%
```

7. What is the probability that a species is Sedentary, given that it occupies human settlements in its breeding area?

[102]:
```
shtab = pd.crosstab(birds['Sedentary'], birds['Human.settlements'],␣
↪normalize='columns')
shtab
```

[102]:
```
Human.settlements       0.0       1.0
Sedentary
0.0                0.661504  0.297872
1.0                0.338496  0.702128
```

[103]:
```python
p = shtab[1][1]
print('The probability that a species is Sedentary, given that it breeds in␣
↪human settlements is ',round(100*p,1), '%', sep="")
```

```
The probability that a species is Sedentary, given that it breeds in human
settlements is 70.2%
```

8. A test for Coronavirus is 70% likely to detect the infection if it is present, and 99.1% likely to return a negative test if the infection is absent. If the prevalence of the disease (the proportion of people who have the disease) is 0.1%, then what is the probability that a person who tests positive actually has the disease?

ANS: If S is the disease status (D=infected, H=healthy) and T is the test result (P=positive, N=negative), then

$$\begin{aligned}\Pr(S = D) &= 0.001 \\ \Pr(T = P|S = D) &= 0.70 \\ \Pr(T = N|S = H) &= 0.991\end{aligned}$$

and so the probabilities of the four outcomes are

$$\begin{aligned}\Pr(S = D, T = P) &= \Pr(S = D)\Pr(T = P|S = D) = 0.001 \times 0.70 = 0.0007 \\ \Pr(S = D, T = N) &= \Pr(S = D)\Pr(T = N|S = D) = 0.001 \times 0.30 = 0.0003 \\ \Pr(S = H, T = P) &= \Pr(S = H)\Pr(T = P|S = H) = 0.999 \times 0.009 = 0.0090 \\ \Pr(S = H, T = N) &= \Pr(S = H)\Pr(T = N|S = H) = 0.999 \times 0.991 = 0.9900\end{aligned}$$

The probability of a postive test is

$$\Pr(T = P) = \Pr(S = D, T = P) + \Pr(S = H, T = P) = 0.0007 + 0.0090 = 0.0097$$

and so by Bayes' Rule the probability of having the disease given a positive test is

$$\begin{aligned}\Pr(S = D|T = P) &= \frac{Pr(T = P|S = D)\Pr(S = D)}{\Pr(T = P)} \\ &= \frac{0.7 \times 0.001}{0.0097} \\ &= \frac{0.0007}{0.0097} \\ &= 0.072\end{aligned}$$

There is a 7.2% chance that a person who tests positive actually has the disease.

9. How would your answer above change if the probability of a false positive test was zero?

ANS: *If there was no possibility of a false positive test, then every positive test is a true positive, so if a person tests positive for the disease there is a 100% chance that they do have the disease.*

1.2 Theoretical Probability Distributions

(5 Marks)

10. A Poisson random variable is often used to model counts of customer arrivals in a shop. Assume that the number of customers to arrive in a particular hour follows a Poisson(5) distribution. Compute and plot the probabililty distribution of a Poisson(5) distribution. (Plot the distribution over the range 0 to 15.)

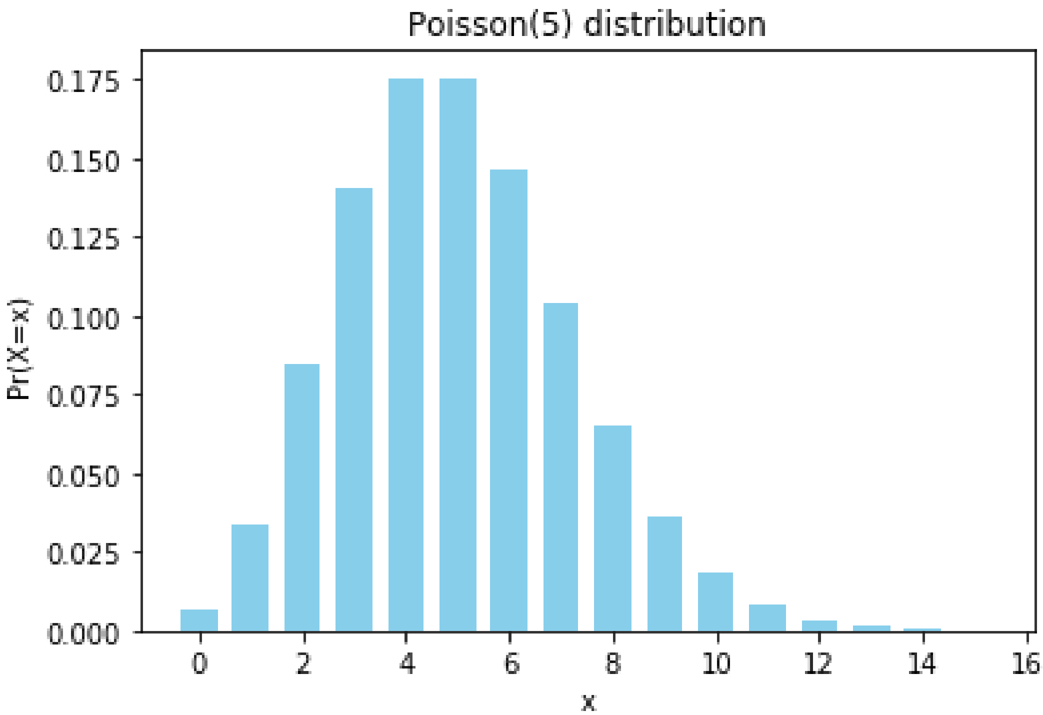
[104]:
```
# Set the parameters
lam = 5
# Generate an array from 0 to 15
nmax = 15
x = np.arange(0, nmax+1)
```

[105]:
```
# Evaluate the probability mass function at these locations
probs = stats.poisson.pmf(x, lam)
pd.DataFrame({"x":x,"P(X=x)":probs})
```

[105]:

	x	P(X=x)
0	0	0.006738
1	1	0.033690
2	2	0.084224
3	3	0.140374
4	4	0.175467
5	5	0.175467
6	6	0.146223
7	7	0.104445
8	8	0.065278
9	9	0.036266
10	10	0.018133
11	11	0.008242
12	12	0.003434
13	13	0.001321
14	14	0.000472
15	15	0.000157

[106]:
```
# Plot these probabilities
width = 0.70  # the width of the bars
fig, ax = pl.subplots(1,1)
rects = ax.bar(x, probs, width, color='SkyBlue')
ax.set(xlabel='x', ylabel='Pr(X=x) ', title='Poisson(5) distribution');
```



11. Find out

(a) The mean and variance of the distribution

(b) The probability that two customers arrive in a particular hour

(c) The probability fewer than 10 arrive

(d) The probability that no more than 10 arrive

(e) The probability that more than 15 arrive

[107]:
```
# (a) The mean and variance of the distribution
# The mean and variance are both 5
```

[108]:
```
# (b) The probability that two customers arrive in a particular hour
prob = stats.poisson.pmf(2, lam)
prob
```

[108]: 0.08422433748856832

[109]:
```
# (c) The probability that fewer than 10 arrive
prob = stats.poisson.cdf(9, lam)
prob
```

[109]: 0.9681719426937951

[110]:
```
# (d) The probability that no more than 10 arrive
prob = stats.poisson.cdf(10, lam)
prob
```

[110]:
```
0.9863047314016171
```

[111]:
```
# (e) The probability that more than 15 arrive
prob = 1-stats.poisson.cdf(15, lam)
prob
```

[111]:
```
6.900824185562815e-05
```

1.3 Model fitting

(9 Marks)

Use the European Birds data set from above. before you start, ensure that you rename the column `Sexual.Dimorphism` as `SexualDimorphism` - since the '.' in its name causes a problem for the ols fitting command. Use the `rename()` command to do this.

[112]:
```
birds = pd.read_csv("EuropeanBirds.csv", encoding='latin1')
# list its first few rows
birds.head()
```

[112]:
```
    ID            Order        Family              Species  LengthU_MEAN  \
0  1.0  Accipitriformes  Accipitridae  Accipiter brevipes          35.0
1  2.0  Accipitriformes  Accipitridae  Accipiter gentilis          55.0
2  3.0  Accipitriformes  Accipitridae     Accipiter nisus          33.0
3  4.0  Accipitriformes  Accipitridae   Aegypius monachus         105.0
4  5.0  Accipitriformes  Accipitridae    Aquila adalberti          80.0

   WingU_MEAN  WingM_MEAN  WingF_MEAN  TailU_MEAN  TailM_MEAN  …  \
0       227.5       220.0       235.0       160.0       154.0  …
1       332.5       312.0       353.0       239.5       223.0  …
2       221.5       203.0       240.0       164.5       149.0  …
3       786.5       772.0       801.0       373.0       365.0  …
4       610.5       600.0       621.0       298.5       291.0  …

   Folivore_B  Frugivore_B  Granivore_B  Arthropods_B  Other.invertebrates_B  \
0         0.0          0.0          0.0           1.0                    0.0
1         0.0          0.0          0.0           0.0                    0.0
2         0.0          0.0          0.0           0.0                    0.0
3         0.0          0.0          0.0           0.0                    0.0
4         0.0          0.0          0.0           0.0                    0.0

   Fish_B  Other.vertebrates_B  Carrion_B  Omnivore_B  \
0     0.0                  1.0        0.0         0.0
1     0.0                  1.0        0.0         0.0
```

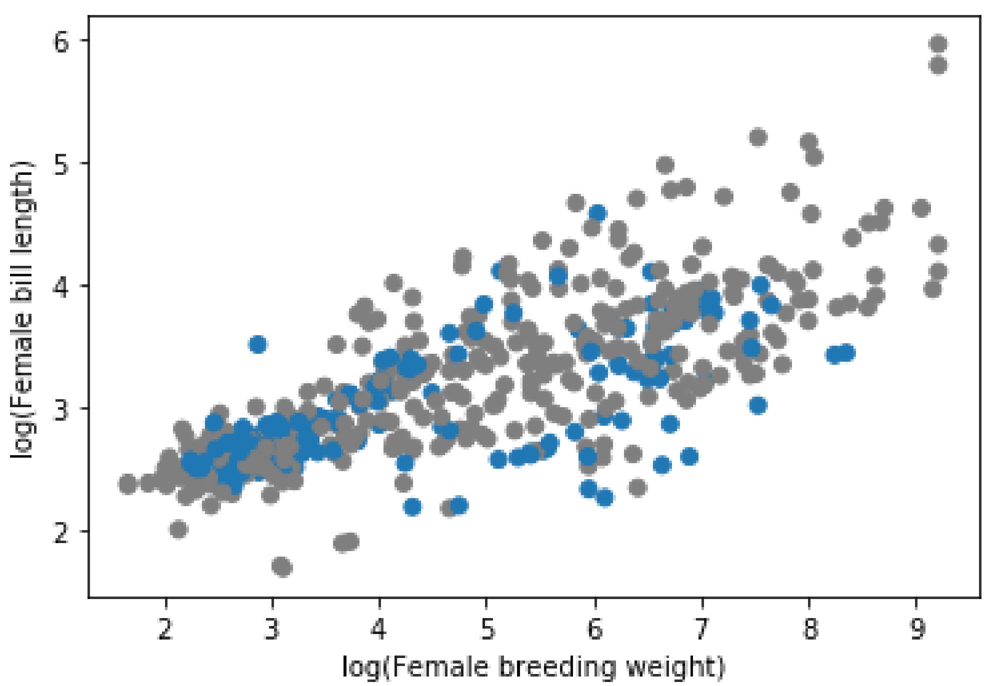
```
2      0.0                         1.0         0.0           0.0
3      0.0                         0.0         1.0           0.0
4      0.0                         1.0         0.0           0.0

                                            Data.source
0  (1) Cramp, S. (2006) The Birds of the Western …
1  (1) Cramp, S. (2006) The Birds of the Western …
2  (1) Cramp, S. (2006) The Birds of the Western …
3  (1) Cramp, S. (2006) The Birds of the Western …
4  (1) Cramp, S. (2006) The Birds of the Western …

[5 rows x 86 columns]
```

12. Draw a scatter plot of the log of female bill length against the log of female breeding weight. Distinguish using a plot symbol species that are or are not sexually dimorphic (with a difference between males and females in size/colour).

[113]:
```
birds['logWeightF_MEAN'] = np.log(birds['WeightF_MEAN'])
birds['logBillF_MEAN'] = np.log(birds['BillF_MEAN'])
birds.rename(columns = {'Sexual.dimorphism':'SexualDimorphism'}, inplace = True)
cols = np.where(birds['SexualDimorphism']==0,'tab:grey','tab:blue')
fig, ax = pl.subplots(1,1)
pl.scatter(birds['logWeightF_MEAN'], birds['logBillF_MEAN'], marker='o',␣
↪c=cols);
ax.set(xlabel='log(Female breeding weight)', ylabel='log(Female bill length)');
```



13. Fit a regression model for log female bill length as predicted by log female breeding weight.

(a) Print out a summary of the model fit

(b) Plot the fitted curve onto the data

(c) Draw a scatter plot of the residuals and comment on them

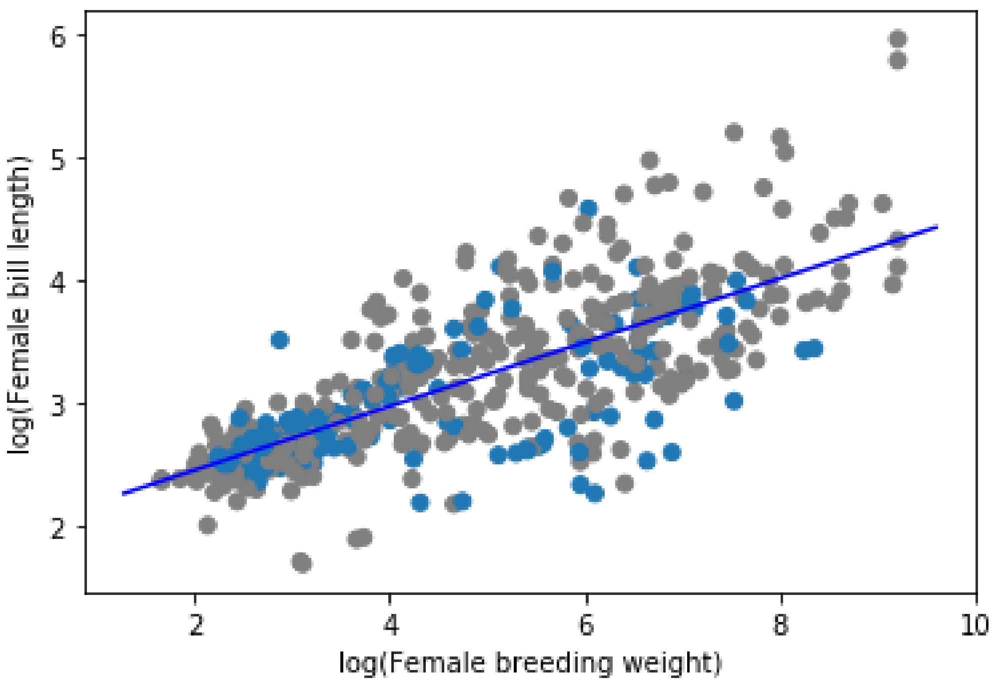
[114]:
```
fittedmodel = smf.ols("logBillF_MEAN ~ logWeightF_MEAN", data=birds).fit()
predictions = fittedmodel.predict(birds)
residuals = birds['logBillF_MEAN'] - predictions
```

[115]:
```
# (a) Print out the detail of the model fit
fittedmodel.summary()
```

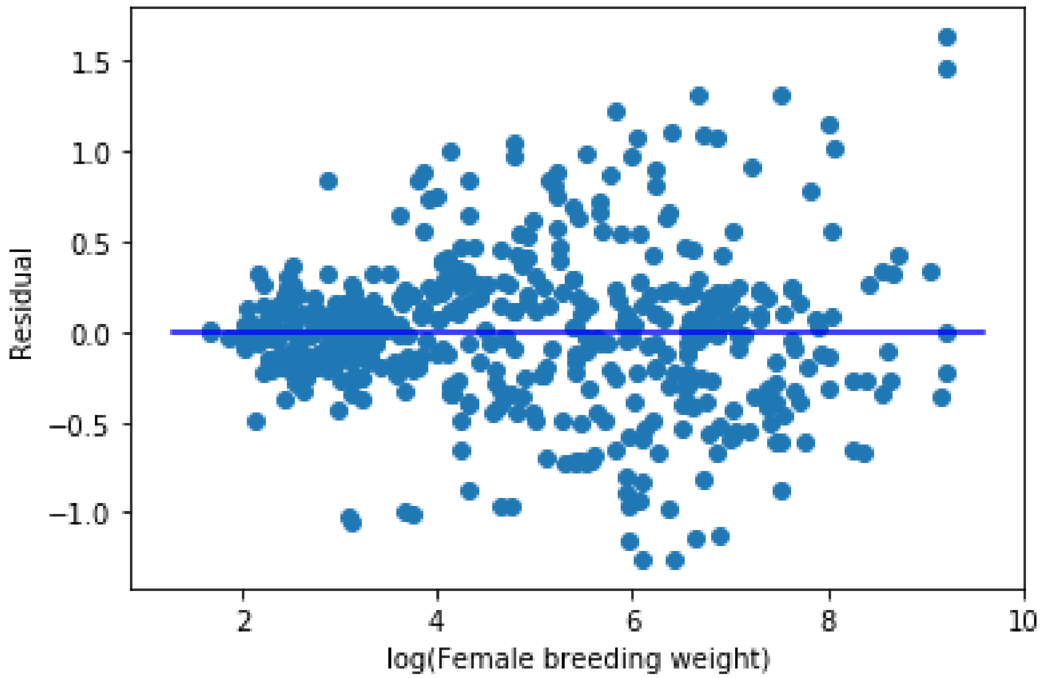
[115]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:          logBillF_MEAN   R-squared:                       0.557
Model:                            OLS   Adj. R-squared:                  0.556
Method:                 Least Squares   F-statistic:                     620.5
Date:                Tue, 21 Apr 2020   Prob (F-statistic):           2.51e-89
Time:                        15:55:07   Log-Likelihood:                -281.26
No. Observations:                 496   AIC:                             566.5
Df Residuals:                     494   BIC:                             574.9
Df Model:                           1
Covariance Type:            nonrobust
================================================================================
===
                      coef    std err          t      P>|t|      [0.025
0.975]
--------------------------------------------------------------------------------
---
Intercept           1.9272      0.054     35.584      0.000       1.821
2.034
logWeightF_MEAN     0.2609      0.010     24.911      0.000       0.240
0.281
==============================================================================
Omnibus:                       22.890   Durbin-Watson:                   0.546
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               45.270
Skew:                           0.264   Prob(JB):                     1.48e-10
Kurtosis:                       4.383   Cond. No.                         15.1
==============================================================================

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

[116]:
```
# (b) Plot the fitted curve onto the data
cols = np.where(birds['SexualDimorphism']==0,'tab:grey','tab:blue')
fig, ax = pl.subplots(1,1)
pl.scatter(birds['logWeightF_MEAN'], birds['logBillF_MEAN'], marker='o',␣
 ↪c=cols);
ax.set(xlabel='log(Female breeding weight)', ylabel='log(Female bill length)');
xmin, xmax = ax.get_xbound() # get the plot bounds
xp = np.linspace(xmin, xmax, 101)
xpmat = pd.DataFrame({'logWeightF_MEAN':xp})
yp = fittedmodel.predict(xpmat)
pl.plot(xp, yp, 'b-');
```



[117]:
```
# (c) Draw a scatter plot of the residuals and comment on them
fig, ax = pl.subplots(1,1)
pl.scatter(birds['logWeightF_MEAN'], residuals, marker='o');
ax.set(xlabel='log(Female breeding weight)', ylabel='Residual');
xmin, xmax = ax.get_xbound() # get the plot bounds
xp = [xmin,xmax] # store these as a vector
yp = [0, 0]
pl.plot(xp, yp, 'b-');
```



ANS: The residuals show non-constant variance: there is a tight group of small bird species, and then a wider scatter among larger species.

14. Now add Sexual Dimorphism as a covariate, and see if that improves the model by inspecting the residual scatter plot.

[118]:
```
fittedmodel2 = smf.ols("logBillF_MEAN ~ logWeightF_MEAN*C(SexualDimorphism)",
                       data=birds).fit()
predictions2 = fittedmodel2.predict(birds)
residuals2 = birds['logBillF_MEAN'] - predictions2
fittedmodel2.summary()
```

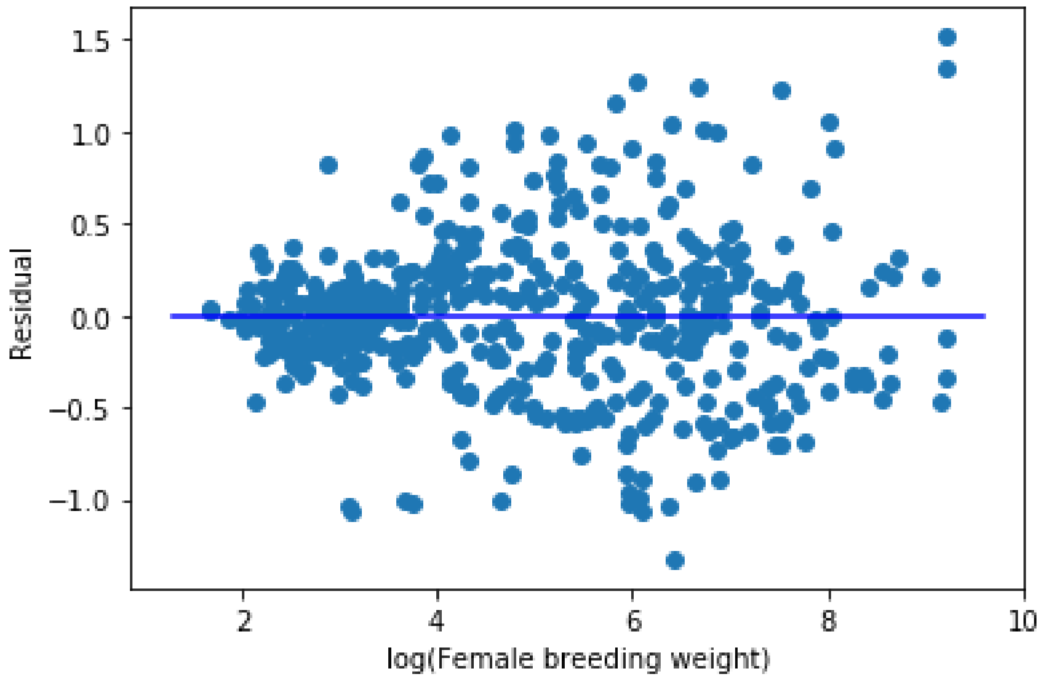
[118]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
Dep. Variable:          logBillF_MEAN   R-squared:                       0.577
Model:                            OLS   Adj. R-squared:                  0.574
Method:                 Least Squares   F-statistic:                     223.6
Date:                Tue, 21 Apr 2020   Prob (F-statistic):           1.74e-91
Time:                        15:55:08   Log-Likelihood:                -269.74
No. Observations:                 496   AIC:                             547.5
Df Residuals:                     492   BIC:                             564.3
Df Model:                           3
Covariance Type:            nonrobust
```

```
================================================================================
==============================
                                                  coef    std err          t
P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
------------------------------
Intercept                                       1.8765      0.065     28.726
0.000       1.748       2.005
C(SexualDimorphism)[T.1.0]                      0.2512      0.114      2.203
0.028       0.027       0.475
logWeightF_MEAN                                 0.2793      0.012     22.887
0.000       0.255       0.303
logWeightF_MEAN:C(SexualDimorphism)[T.1.0]     -0.0829      0.023     -3.592
0.000      -0.128      -0.038
==============================================================================
Omnibus:                       17.659   Durbin-Watson:                   0.603
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               29.160
Skew:                           0.251   Prob(JB):                     4.66e-07
Kurtosis:                       4.077   Cond. No.                         36.1
==============================================================================

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

[119]:

```
# Draw a scatter plot of the residuals and comment on them
fig, ax = pl.subplots(1,1)
pl.scatter(birds['logWeightF_MEAN'], residuals2, marker='o');
ax.set(xlabel='log(Female breeding weight)', ylabel='Residual');
xmin, xmax = ax.get_xbound() # get the plot bounds
xp = [xmin,xmax] # store these as a vector
yp = [0, 0]
pl.plot(xp, yp, 'b-');
```



ANS: *There is no significant improvement in the appearance of the plot: Sexual Dimorphism has a significant presence in the model (p-values are small in the fitted model summary), but does not alter the non-constant variance apparent in the residual plot.*

1.4 Ethical and privacy issues

(5 Marks)

15. During the Coronavirus pandemic, many countries are investigating the electronic collection of contact data as a means of identifying close contacts of people who are diagnosed with Covid-19. One possibility is an app on a smartphone which broadcasts an identifier which other devices record when in close proximity. If a person is identified as a case, then the data from their app is used to identify the personal devices to which they have been close in the previous weeks.
    Write a short discussion, of about 250-300 words, about the issues of **privacy**, **security** and **confidentiality** that you see with a plan of this sort.

ANS: *Answers should include the discussion of the following sort:*

1. Discussion of whether the scheme is voluntary, and if it is not then that there is a significant invasion of privacy. It will mean that it is discoverable who a person is associating with, independent of the need to trace close contacts. This may be used for improper purposes, and it is difficult to be confident that the data will be used only for that purpose.

2. It is not going to be easy to have confidence that the data will be kept suitably secure. If a system such as this is put together at short order without a chance for rigorous testing. Given recent experiences with software such as Zoom, it is clear that software always incl, des

risks.

3. Even when the data **are** used for a purpose in line with the goal of tracing contacts, it risks revealing relationships and activities that are legitimate (not illegal), but which a person might reasonably wish to remain private.
4. Balanced against these considerations are the concerns of the community benefit of providing these data for the public good. This balance is the sacrifice of personal privacy (which is complementary to the sacrifice implied by loss of movement and in some cases employment in the lockdown).