Lorenzo Rega

UNI: lr3103

Homework 1

Problem 1

a)

Model 2 uses NewPrescriptions as the dependent variable. Samples and Details are the independent variables of this linear regression model.

Let's run the linear regression model:

```
In [43]: import pandas as pd
         import statsmodels.formula.api as smf
         df_dov = pd.read_excel('DetailingData.xlsx', sheet_name="Original Variables")
         reg2 = smf.ols('NewPrescription ~ Samples + Details', data=df_dov)
         reg2_result = reg2.fit()
         reg2_result.summary()
```

Out[43]:

OLS Regression Results

Dep. Variable:	NewPrescription	**R-squared:**	0.199
Model:	OLS	**Adj. R-squared:**	0.198
Method:	Least Squares	**F-statistic:**	543.8
Date:	Tue, 11 Oct 2022	**Prob (F-statistic):**	1.02e-211
Time:	12:56:47	**Log-Likelihood:**	-7532.7
No. Observations:	4392	**AIC:**	1.507e+04
Df Residuals:	4389	**BIC:**	1.509e+04
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	11.1611	0.029	383.162	0.000	11.104	11.218
Samples	0.0596	0.004	13.363	0.000	0.051	0.068
Details	0.3844	0.016	24.610	0.000	0.354	0.415

Omnibus:	591.727	**Durbin-Watson:**	1.517
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	859.241
Skew:	1.015	**Prob(JB):**	2.62e-187
Kurtosis:	3.759	**Cond. No.**	8.16

The Samples coefficient is 0.0596 and the Details one is 0.3844. Since they are close to zero, we might wonder whether they have a statistically significant impact on New Prescriptions or not.

Each Beta has a Normal distribution; the numbers shown before are just single draws from the distribution. We want to know whether the true Beta is zero or not. That is why we do a hypothesis test on each Beta. Our null hypothesis is that Beta is zero and the alternative hypothesis is that Beta is different from 0. Under the Null hypothesis, Beta follows a T-student distribution with (N – p) degrees of freedom, where N is the number of data points and p is the number of variables in the regression.

We can see how both Samples and Details have p-values=0. This lets us reject the null hypothesis for both of the variables. We can also see from the confidence intervals that 0 is not included in them.

That is why we can conclude both Samples and Details have a statistically significant impact on New Prescriptions.

b)

The expected number of new prescriptions generated by a detailing visit is 0.3844. The 95% confidence interval is [0.354, 0.415]. Python has calculated it for us.

Let's now compute the expected $ margin generated by a detailing visit.

Each detailing visit costs 85$. The margin from a single new prescription is 115$, and that is the same for the additional refills. We also know that these margins do not account for the cost of detailing visits.

One detailing visit lets us have 0.3844 new prescriptions and 0.3844*2.5 additional refills.

Margin is revenues – costs. Since these margins do not account, so do not include, detailing costs, we must subtract 85$ from the total margin we get.

In conclusion, the expected $ margin is:

(0.3844*2.5*115+0.3844*115) -85=154.72$- 85$= **69.72$**

To find the 95% confidence interval we just do the same calculations for the extremes of [0.354, 0.415].

(0.354*2.5*115+0.354*115) -85=142.485 $- 85$= 57.485 $

(0.415*2.5*115+0.415*115) -85=167.0375 $- 85$= 82.0375 $

The 95% confidence interval of the expected margin is **[57.485, 82.0375].**

c)

Let’s run model 7 regression:

In [18]:
```
reg7 = smf.ols('NewPrescDiff ~ SamplesDiff + DetailsDiff', data=df_ddv)
reg7_result = reg7.fit()
reg7_result.summary()
```

Out[18]:

OLS Regression Results

Dep. Variable:	NewPrescDiff	**R-squared:**	0.613
Model:	OLS	**Adj. R-squared:**	0.613
Method:	Least Squares	**F-statistic:**	3335.
Date:	Mon, 10 Oct 2022	**Prob (F-statistic):**	0.00
Time:	12:51:07	**Log-Likelihood:**	202.92
No. Observations:	4209	**AIC:**	-399.8
Df Residuals:	4206	**BIC:**	-380.8
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	0.0799	0.004	22.473	0.000	0.073	0.087
SamplesDiff	0.0354	0.001	58.217	0.000	0.034	0.037
DetailsDiff	0.1115	0.003	41.334	0.000	0.106	0.117

Omnibus:	398.475	**Durbin-Watson:**	2.004
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	517.794
Skew:	0.825	**Prob(JB):**	3.65e-113
Kurtosis:	3.477	**Cond. No.**	6.05

In model 7, the coefficient is **0.1115**. The 95% confidence interval is **[0.106, 0.117]**.

Let’s compute the expected extra profit between month t and month t-1:

(0.115*2.5*115+0.115*115) -85=46.2875 $- 85 $ = **-38.7125** $.

Let’s now compute the 95% confidence interval of the extra profit:

(0.106*2.5*115+0.106*115) -85=42.665$- 85$= -42.335$

(0.117*2.5*115+0.117*115) -85=47.0925 $- 85$=-37.9075 $

The interval is **[-42.335, -37.9075]**.

We get a different result from Model 2 both in values and meaning.

In model 2, an increase in details by 1 leads to an increase in new prescriptions by 0.3844, across all physicians on average. In model 7, an increase of 1, compared to the same physician's detailing last month, would lead to an increase in new prescriptions by 0.1115.

We have different results and that is because we are using different variables. We have differenced variables in model 7. That means we have the changes in the levels of these variables (except the dummy variable) from one month to the next.

Model 9 is the more informative. In fact, it takes into account the fact that we have different physicians in our dataset. We might have for example physician A that has on month t 20 visits and 200 new prescriptions and on month t+1 21 visits and 201 new prescriptions. The regression might think that each visit is worth approximately 200/20 =10 new prescriptions. If we carefully look at the data, we see how instead, an increase of one in the visits in the following month only increased prescriptions by one. Using model 2 we have biased data, whereas model 7 is trying to fix this bias.

We know from the case that physicians draw on their own knowledge and experience in deciding which drugs to prescribe. In our example, physician A might have 200 prescriptions just because it is his own knowledge that makes him do that, and not the visits he receives. That could be why increasing by 1 the visits in the following month did not make any real change. In this case, the prescriptions were not led by the visits.

d)

In my opinion, model 8 is the most reliable model.

```
In [19]: reg8 = smf.ols('NewPrescDiff ~ SamplesDiff + DetailsDiff + CompetitorNewDiff', data=df_ddv)
         reg8_result = reg8.fit()
         reg8_result.summary()
```

Out[19]:

OLS Regression Results

Dep. Variable:	NewPrescDiff	**R-squared:**	0.613
Model:	OLS	**Adj. R-squared:**	0.613
Method:	Least Squares	**F-statistic:**	2223.
Date:	Mon, 10 Oct 2022	**Prob (F-statistic):**	0.00
Time:	19:58:02	**Log-Likelihood:**	203.13
No. Observations:	4209	**AIC:**	-398.3
Df Residuals:	4205	**BIC:**	-372.9
Df Model:	3		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	0.0798	0.004	22.450	0.000	0.073	0.087
SamplesDiff	0.0354	0.001	58.206	0.000	0.034	0.037
DetailsDiff	0.1114	0.003	41.283	0.000	0.106	0.117
CompetitorNewDiff	0.0003	0.000	0.641	0.522	-0.001	0.001

Omnibus:	398.222	**Durbin-Watson:**	2.005
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	517.354
Skew:	0.825	**Prob(JB):**	4.55e-113
Kurtosis:	3.476	**Cond. No.**	8.91

First of all, because of what I explained in part c, I consider, in general, models with differenced variables more reliable. Models 7, 8, and 9 have the highest R^2 (61.3%) as well. I think model 9 uses too many variables and that could lead to an overfitting case, where regression will start capturing some spurious correlations in the data. Model 8 looks like the best one since it has the highest R^2 and not too many independent variables (only 3).

In model 8 the coefficient for the DetailsDiff is 0.1114 and the 95% confidence interval is [0.106, 0.117].

Let's compute the expected extra profit and the 95% confidence interval, just like we did in the previous cases.

(0.1114*2.5*115+0.1114*115) -85= 44.8385 $- 85 $ = **-40.1615** $

(0.106*2.5*115+0.106*115) -85=42.665$- 85$= -42.335$

(0.117*2.5*115+0.117*115) -85=47.0925 $- 85$=-37.9075 $

The interval is [-42.335, -37.9075].

The model shows us that every visit costs too much if compared to what it makes the company earn. That is why we have negative profits.

e)

Model 3:

Dep. Variable:	NewPrescription	**R-squared:**	0.353
Model:	OLS	**Adj. R-squared:**	0.352
Method:	Least Squares	**F-statistic:**	796.6
Date:	Mon, 10 Oct 2022	**Prob (F-statistic):**	0.00
Time:	21:03:47	**Log-Likelihood:**	-7064.0
No. Observations:	4392	**AIC:**	1.414e+04
Df Residuals:	4388	**BIC:**	1.416e+04
Df Model:	3		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	-15.1468	0.815	-18.593	0.000	-16.744	-13.550
Samples	0.0707	0.004	17.562	0.000	0.063	0.079
Details	0.3979	0.014	28.326	0.000	0.370	0.425
MedicalCPI	0.1031	0.003	32.310	0.000	0.097	0.109

Omnibus:	200.423	**Durbin-Watson:**	1.880
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	231.408
Skew:	0.522	**Prob(JB):**	5.63e-51
Kurtosis:	3.419	**Cond. No.**	1.14e+04

Model 4:

Dep. Variable:	NewPrescription	**R-squared:**	0.360
Model:	OLS	**Adj. R-squared:**	0.359
Method:	Least Squares	**F-statistic:**	492.9
Date:	Mon, 10 Oct 2022	**Prob (F-statistic):**	0.00
Time:	21:03:55	**Log-Likelihood:**	-7039.6
No. Observations:	4392	**AIC:**	1.409e+04
Df Residuals:	4386	**BIC:**	1.413e+04
Df Model:	5		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	-14.8137	0.814	-18.197	0.000	-16.410	-13.218
Samples	0.0687	0.004	17.079	0.000	0.061	0.077
Details	0.3905	0.014	27.857	0.000	0.363	0.418
MedicalCPI	0.1015	0.003	31.735	0.000	0.095	0.108
Psych	-0.8338	0.144	-5.810	0.000	-1.115	-0.552
CompetitorNew	0.0042	0.001	3.583	0.000	0.002	0.006

Omnibus:	197.675	**Durbin-Watson:**	1.893
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	228.625
Skew:	0.515	**Prob(JB):**	2.26e-50
Kurtosis:	3.434	**Cond. No.**	1.15e+04

Model 5:

Dep. Variable:	NewPrescription	**R-squared:**	0.358
Model:	OLS	**Adj. R-squared:**	0.357
Method:	Least Squares	**F-statistic:**	611.3
Date:	Mon, 10 Oct 2022	**Prob (F-statistic):**	0.00
Time:	21:03:51	**Log-Likelihood:**	-7046.1
No. Observations:	4392	**AIC:**	1.410e+04
Df Residuals:	4387	**BIC:**	1.413e+04
Df Model:	4		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	-15.0917	0.811	-18.598	0.000	-16.683	-13.501
Samples	0.0698	0.004	17.399	0.000	0.062	0.078
Details	0.3937	0.014	28.106	0.000	0.366	0.421
MedicalCPI	0.1029	0.003	32.396	0.000	0.097	0.109
Psych	-0.8618	0.143	-6.005	0.000	-1.143	-0.580

Omnibus:	194.833	**Durbin-Watson:**	1.891
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	224.714
Skew:	0.512	**Prob(JB):**	1.60e-49
Kurtosis:	3.426	**Cond. No.**	1.14e+04

MedicalCPI appears to have a statistically significant effect if we look at models 3, 4, and 5. In all three models, the coefficient is different from zero and its p-value is zero, showing that we must reject the null hypothesis that the coefficient is equal to 0.

Model 9:

Dep. Variable:	NewPrescDiff	**R-squared:**	0.613
Model:	OLS	**Adj. R-squared:**	0.613
Method:	Least Squares	**F-statistic:**	1333.
Date:	Mon, 10 Oct 2022	**Prob (F-statistic):**	0.00
Time:	09:42:04	**Log-Likelihood:**	203.24
No. Observations:	4209	**AIC:**	-394.5
Df Residuals:	4203	**BIC:**	-356.4
Df Model:	5		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	0.0813	0.010	8.154	0.000	0.062	0.101
SamplesDiff	0.0354	0.001	58.126	0.000	0.034	0.037
DetailsDiff	0.1114	0.003	41.181	0.000	0.106	0.117
CompetitorNewDiff	0.0003	0.000	0.637	0.524	-0.001	0.001
MedicalCPIDiff	-0.0020	0.011	-0.176	0.860	-0.024	0.020
DRGDiff	5.921e-05	0.000	0.385	0.701	-0.000	0.000

Omnibus:	398.617	**Durbin-Watson:**	2.005
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	517.979
Skew:	0.826	**Prob(JB):**	3.33e-113
Kurtosis:	3.475	**Cond. No.**	99.3

We have understood from previous analyses that the data used in these models are biased. That is why we should check what happens in model 9 as well.

In model 9 the coefficient of this independent variable is -0.0020, and we see that the p-value is 0.86, which makes us accept the null hypothesis that the coefficient is zero. We can also see how zero is included in the 95% confidence interval [-0.024, 0.020].

f) CompetitorNew coefficient in model 5 is 0.0042 and its p-value is zero. That means it is statistically significant. An increase of one of the new prescriptions written by the physician in that month for the main competitor to Xuris leads to an increase in new prescriptions of Xuris by 0.0042.

The Psych coefficient is -0.8338 and its p-value is 0. It is statistically significant as well.

That means that if you are a psychiatrist, you are going to prescribe 0.8338 units of Xuris less than someone who's not a psychiatrist.

Problem 2

a)

We first run a linear regression model with feed as the only independent variable and eggs as the dependent one.

```
In [44]: df_ep = pd.read_csv('egg_production.csv')
         regEgg = smf.ols('eggs ~ feed', data=df_ep)
         regEgg_result = regEgg.fit()
         regEgg_result.summary()
```

Out[44]:

OLS Regression Results

Dep. Variable:	eggs	**R-squared:**	0.176
Model:	OLS	**Adj. R-squared:**	0.175
Method:	Least Squares	**F-statistic:**	331.1
Date:	Tue, 11 Oct 2022	**Prob (F-statistic):**	3.38e-67
Time:	13:07:32	**Log-Likelihood:**	-1190.7
No. Observations:	1552	**AIC:**	2385.
Df Residuals:	1550	**BIC:**	2396.
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	3.8328	0.114	33.635	0.000	3.609	4.056
feed	-0.0891	0.005	-18.195	0.000	-0.099	-0.080

Omnibus:	0.504	**Durbin-Watson:**	2.111
Prob(Omnibus):	0.777	**Jarque-Bera (JB):**	0.484
Skew:	0.043	**Prob(JB):**	0.785
Kurtosis:	3.005	**Cond. No.**	201.

The coefficient of the feed variable is -0.0891. Its p-value is zero, so we can assume it is statistically significant. What is outstanding in this regression model is that the feed coefficient is negative. I would have personally expected to find the opposite. Intuitively, the more you feed your chicken, the more eggs you expect to get back. We are in an analytics course though and we can't just base ourselves on intuition. This is what comes from the data: for every more unit of feed given to the chicken, we get 0.0891 eggs less. It looks like we should not feed the chickens to get the largest number of eggs. We can also note how the R^2 is equal to 17.6%. This means not much variability is explained by the linear regression model so it would be interesting to add another variable into the model to see if the feed coefficient remains negative or not.

b)

We now run another regression model, similar to the one we did before. This time we are going to have two independent variables: feed and temperature.

```
In [9]: regEgg2 = smf.ols('eggs ~ feed + temperature', data=df_ep)
        regEgg2_result = regEgg2.fit()
        regEgg2_result.summary()
```

Out[9]:

OLS Regression Results

Dep. Variable:	eggs	**R-squared:**	0.176
Model:	OLS	**Adj. R-squared:**	0.175
Method:	Least Squares	**F-statistic:**	165.6
Date:	Tue, 11 Oct 2022	**Prob (F-statistic):**	6.63e-66
Time:	10:52:04	**Log-Likelihood:**	-1190.5
No. Observations:	1552	**AIC:**	2387.
Df Residuals:	1549	**BIC:**	2403.
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	3.8449	0.116	33.137	0.000	3.617	4.072
feed	-0.0891	0.005	-18.190	0.000	-0.099	-0.079
temperature	-0.0006	0.001	-0.557	0.577	-0.003	0.002

Omnibus:	0.513	**Durbin-Watson:**	2.111
Prob(Omnibus):	0.774	**Jarque-Bera (JB):**	0.489
Skew:	0.043	**Prob(JB):**	0.783
Kurtosis:	3.007	**Cond. No.**	278.

This time we get -0.0891 as the coefficient for the feed variables and -0.0006 as the coefficient for the temperature variable. The coefficient of the feed variable is the same as we found before. It looks like there actually might be a negative correlation between feed and the number of eggs, even though it seems still strange to me. It is interesting how the R^2 is still 17.6% even though we added another variable. If we look at the F-test we see how we must reject the hypothesis that every coefficient is equal to zero. If we look at the temperature coefficient p-value we must accept the null hypothesis though. We can it as well from its 95% confidence interval, where zero is included. Temperature doesn't have a statistically significant impact on the number of eggs. This kind of goes against the farmer's suspects.

c)

We suspect that the amount of feed given to each chicken depends on the temperature. We plot the data on a graph to understand whether this is true or not. We use the temperature as our independent variable and the feed as our dependent variable.

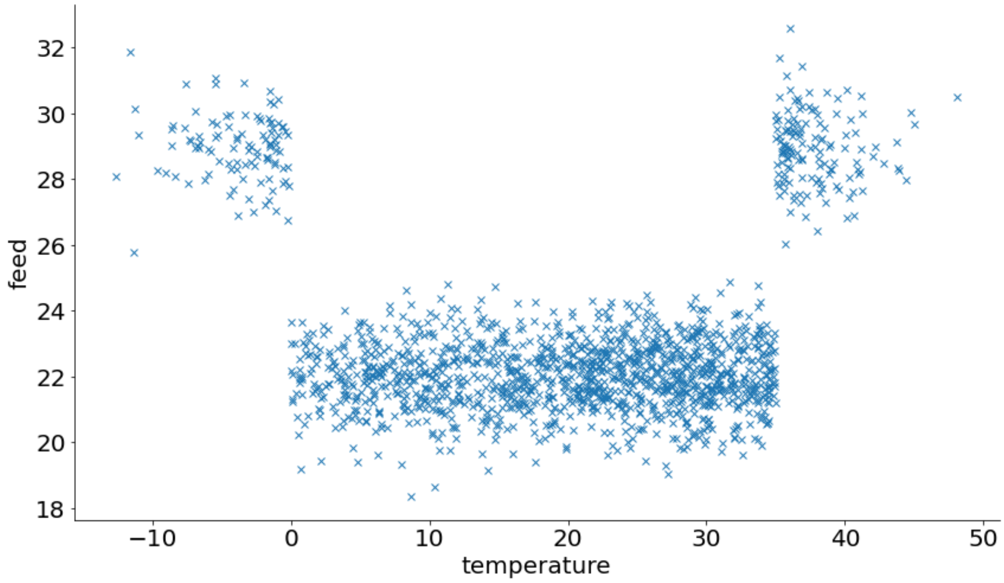
```
In [26]: plt.figure(figsize=(14, 8))

plt.plot(df_ep.temperature, df_ep.feed, linewidth=0, marker='x')

plt.xticks(fontsize=20)
plt.yticks(fontsize=20)

plt.xlabel("temperature", fontsize=20)
plt.ylabel('feed', fontsize=20)

sns.despine()
```



What we can see from the graph is that when the temperature is between 0 and 35 the feed given is in the range [20,24] approximately. We can also note two other areas that appear to be rarer. That is when temperatures become very low or very high. The farmer tends to give much more feed to his chickens when that happens.

To capture this phenomenon, we can create a binary variable that is equal to 1 when we have a temperature below 0 or higher than 35.

```
In [31]: def is_temperature_extreme(value):
    if float(value) < 0 or float(value)>35:
        return 1
    else:
        return 0


df_ep['extreme_temperature'] = df_ep['temperature'].map(is_temperature_extreme)
display(df_ep.head())
```

	eggs	feed	temperature	extreme_temperature
0	1.944645	28.521682	-3.920247	1
1	2.367084	20.810192	7.489837	0
2	1.361380	29.259575	-5.425451	1
3	1.763221	22.245235	1.486627	0
4	2.003410	23.331641	9.976938	0

d)

The linear regression with three independent variables (feed, temperature, extreme_temperature) makes us conclude something different from what we saw before.

In [32]:
```
regEgg3 = smf.ols('eggs ~ feed + temperature + extreme_temperature', data=df_ep)
regEgg3_result = regEgg3.fit()
regEgg3_result.summary()
```

Out[32]:

OLS Regression Results

Dep. Variable:	eggs	**R-squared:**	0.236
Model:	OLS	**Adj. R-squared:**	0.234
Method:	Least Squares	**F-statistic:**	159.0
Date:	Tue, 11 Oct 2022	**Prob (F-statistic):**	7.96e-90
Time:	12:00:21	**Log-Likelihood:**	-1132.5
No. Observations:	1552	**AIC:**	2273.
Df Residuals:	1548	**BIC:**	2294.
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	1.0529	0.278	3.786	0.000	0.507	1.598
feed	0.0388	0.013	3.081	0.002	0.014	0.063
temperature	-0.0007	0.001	-0.695	0.487	-0.003	0.001
extreme_temperature	-1.0276	0.094	-10.966	0.000	-1.211	-0.844

Omnibus:	0.478	**Durbin-Watson:**	2.108
Prob(Omnibus):	0.787	**Jarque-Bera (JB):**	0.390
Skew:	0.023	**Prob(JB):**	0.823
Kurtosis:	3.062	**Cond. No.**	724.

First of all, the R^2 has increased to 23.6%, so it looks like a better model than the ones used before. The F-test still tells us that not all the coefficients are zero.

The feed coefficient is now 0.0388. It is now positive, and it goes in the same direction as common knowledge: the more we feed the chicken, the more eggs we get. The temperature coefficient has a high p-value, so we can confirm that the temperature data, used as we did before, is just useless. What is interesting though is that the coefficient of the extreme_temperature variable is -1.0276 with a p-value of zero. What is statistically significant is not every temperature we have but knowing if we are in extreme conditions or not. If we are in these extreme conditions, we are, in fact, going to have 1.0276 fewer eggs.

What the previous regression models were probably seeing was that when there was a lot of feeding, we had fewer eggs. That is why the coefficient of the feed variable was negative. The thing is though, that the farmer fed the chickens a lot only in extreme temperatures and we now know that with extreme temperatures the chickens make fewer eggs. In conclusion, it is the extreme temperature that makes a chicken produce fewer eggs and not the amount of feed received.

e)

We can find the 90% confidence interval for the prediction of the number of eggs that were produced if the feed was 25 and the temperature was −1 using python.

In [42]:
```
new_data = pd.DataFrame({"feed":[25], 'temperature':[-1], "extreme_temperature":[1]})
predictions = regEgg3_result.get_prediction(new_data)
predictions.summary_frame(alpha=0.1)
```

Out[42]:

	mean	mean_se	mean_ci_lower	mean_ci_upper	obs_ci_lower	obs_ci_upper
0	0.994741	0.062953	0.891131	1.098351	0.16108	1.828403

We are looking for a specific observation in this case. We can see how the confidence interval for the single observation is wider than the one of the mean.

If we were asked which was the number of eggs that we usually have with temp=-1 and feed=25 then we would have used the mean_ci. We are looking for instead specifically how many eggs we get under these conditions. Since our request is specific, we are also going to have a larger error and a larger confidence interval.