Identifying Similarities and Differences in Biomarker Profiles between Alzheimer's Disease and Late Onset Bipolar Disease

DATA 690 Final Report

Jessica, Rohith, Rahul

December 2022

Identifying Similarities and Differences in MRI Profiles between Alzheimer’s Disease and Late Onset Bipolar Disease

Background

Ariadna Besga; Manuel Grana; Darya Chyzhyk created the dataset to analyze the clinical data between the patients to identify similarities and differences in disease types such as Alzheimer's and late onset bipolar disorder compared to a health control cohort. So using this dataset our objective of the study is to find how diseases may differ from each other from plasma biomarker measurements from patients. We are going to represent different types of hist plot using the condition column and the age column in which how many males or females got these three types of diseases and determine their count and after that, we are dividing the data frame into two subset dataframe like AD vs CRL & LOBD vs CRL so it will be easy to do the statistical method like logistic regression between the three diseases and by using the t-test to find the correlation coefficients and denoting the p values to compare the regression slopes between the disease of AD vs CRL and LOBD vs CRL.

Below are two hypothesis questions that we attempted to answer in our dataset. We will provide the results at the end.

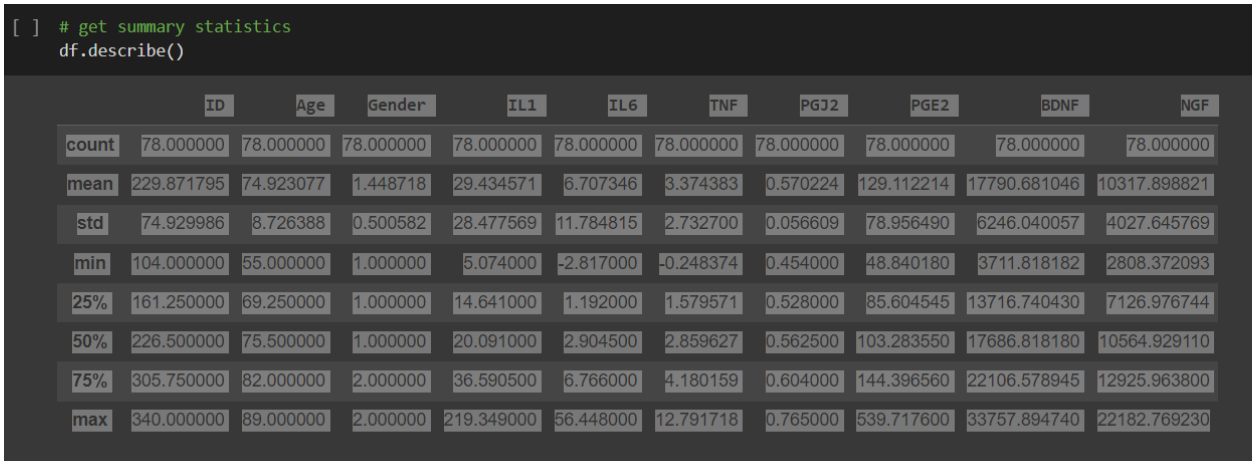
Hypothesis 1: The biomarker profiles for Alzheimer’s Disease (AD) and Late-Onset Bipolar Disorder (LOBD) will both be significantly different from biomarker profiles for the control cohort.

Hypothesis 2: The biomarker profiles for AD and LOBD will be significantly different from each other.

Data

Data is structured as follows. The condition column, ID, age, gender, measurements of particular bipolar patients, and treatment measurements are all merged into the file "clinical data all.csv.". In the condition column, "crl" stands for "control," "ea" for "Alzheimer's diseases," and "tb" for "Late Onset Bipolar Disorder." Measurements of particular biomarkers include, "IL1" stands for interleukin 1, "IL6" for interleukin 6, and "TNF" stands for tumor necrosis factor alpha, an inflammatory cytokine.Prostaglandin J2 is referred to as "PGJ2," while Prostaglandin E2 is referred to as "PGE2," both of which are groups of lipids produced at the site of injury or infection. "BDNF" stands for brain-derived neurotrophic factor, a protein that helps nerves survive, and "NGF" stands for nerve growth factor, which helps nerves grow; Measurements used during treatment include "Nitritos" and "MDA" = malondialdehyde columns. These two columns were dropped because they are irrelevant to our hypotheses.

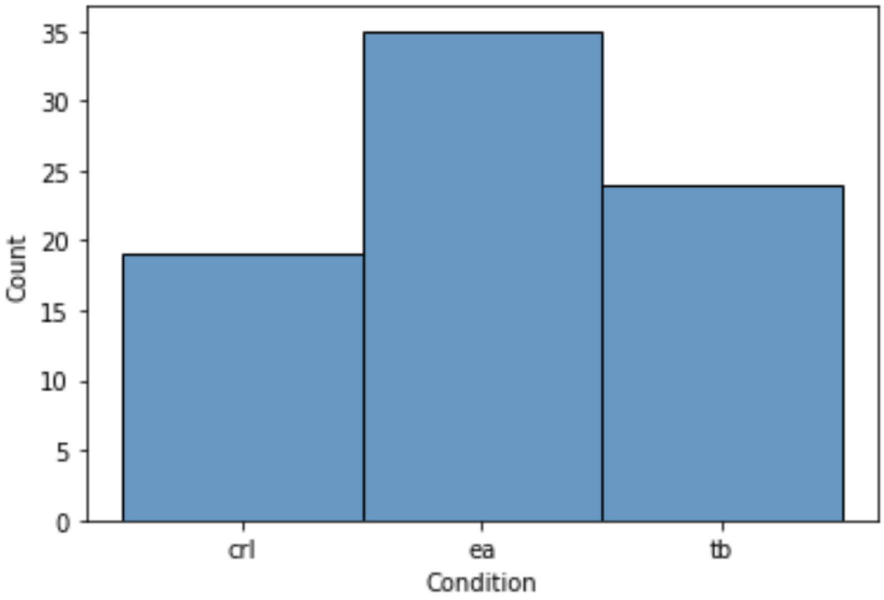
Below is a summary of statistical information per column of the dataset:



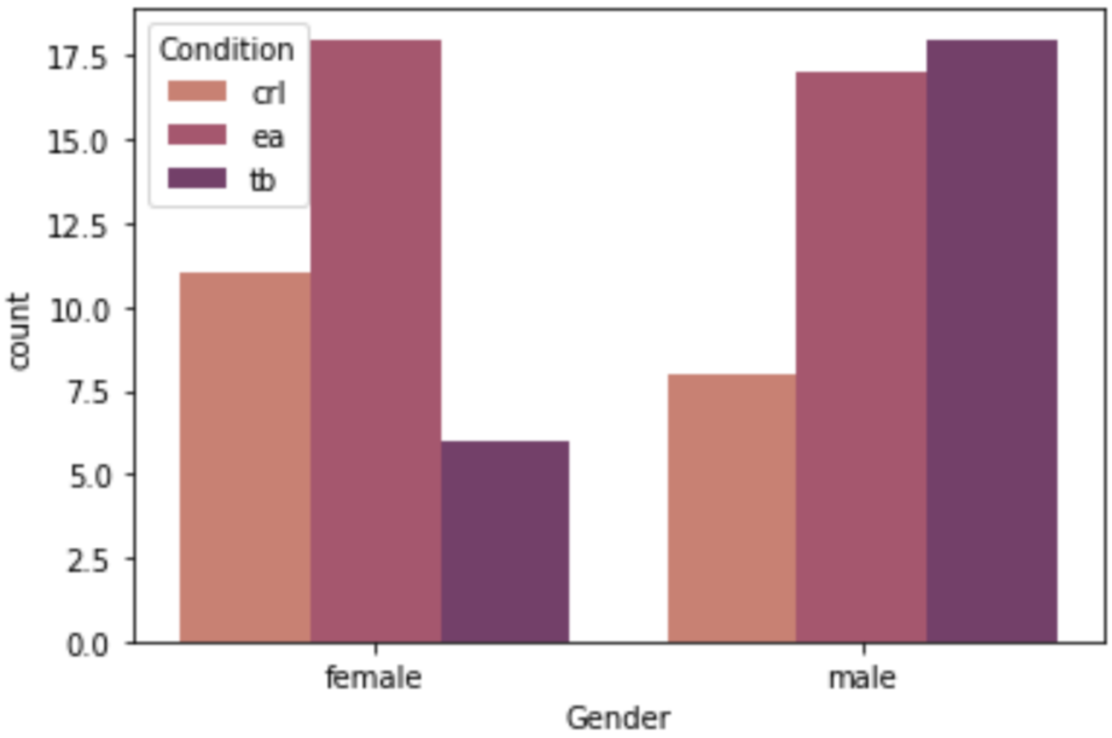
```
[ ]  # get summary statistics
     df.describe()
```

	ID	Age	Gender	IL1	IL6	TNF	PGJ2	PGE2	BDNF	NGF
count	78.000000	78.000000	78.000000	78.000000	78.000000	78.000000	78.000000	78.000000	78.000000	78.000000
mean	229.871795	74.923077	1.448718	29.434571	6.707346	3.374383	0.570224	129.112214	17790.681046	10317.898821
std	74.929986	8.726388	0.500582	28.477569	11.784815	2.732700	0.056609	78.956490	6246.040057	4027.645769
min	104.000000	55.000000	1.000000	5.074000	-2.817000	-0.248374	0.454000	48.840180	3711.818182	2808.372093
25%	161.250000	69.250000	1.000000	14.641000	1.192000	1.579571	0.528000	85.604545	13716.740430	7126.976744
50%	226.500000	75.500000	1.000000	20.091000	2.904500	2.859627	0.562500	103.283550	17686.818180	10564.929110
75%	305.750000	82.000000	2.000000	36.590500	6.766000	4.180159	0.604000	144.396560	22106.578945	12925.963800
max	340.000000	89.000000	2.000000	219.349000	56.448000	12.791718	0.765000	539.717600	33757.894740	22182.769230

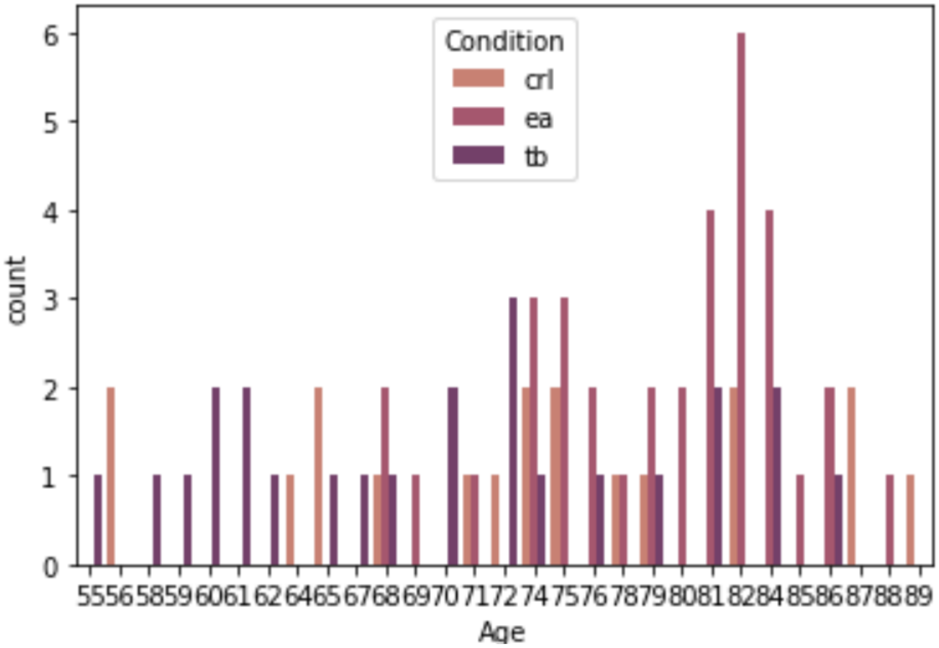
The counts plot below shows there are 19 people in the control group, 35 people in the Alzheimer’s group, and 24 people in the LOBD group.



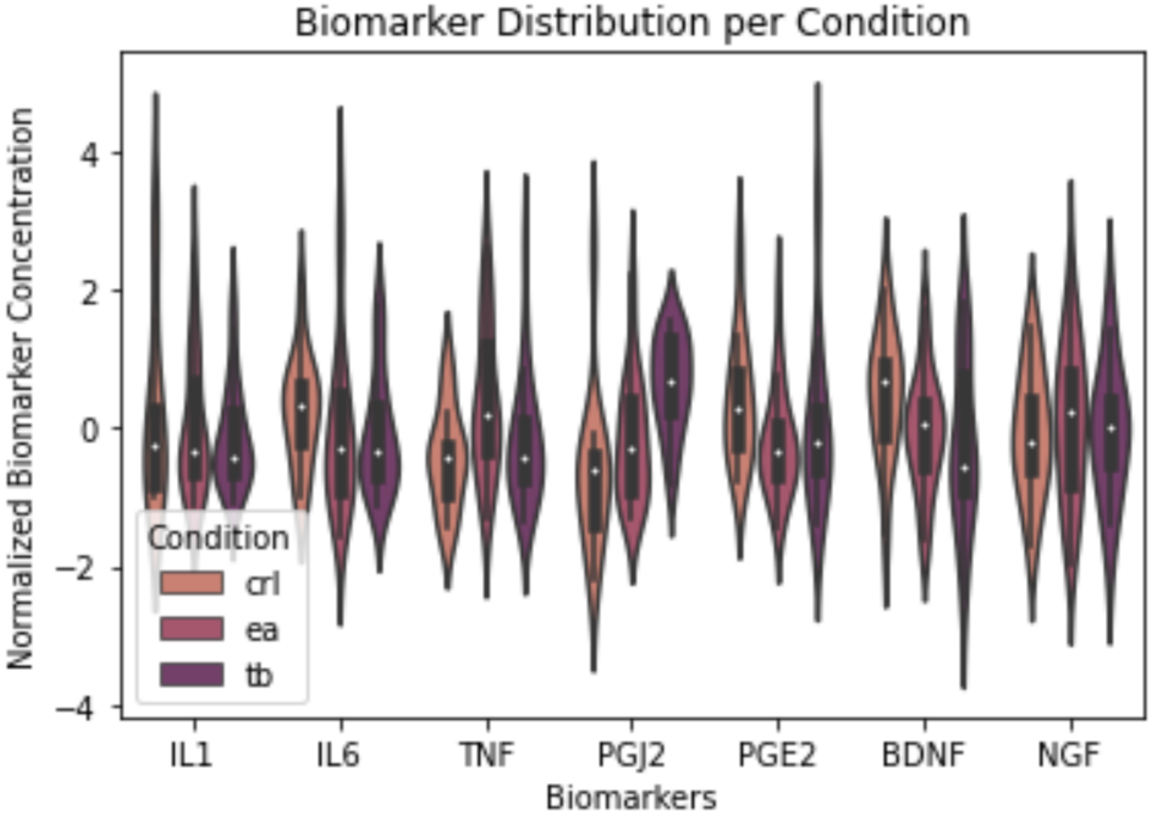
The counts plot below shows how many females and males have each condition in the dataset. For the control group, there are 11 females and 8 males. For AD, there are 18 females and 17 males. For LOBD, there are 6 females and 18 males.



The counts plot below shows the distribution of ages for each condition within the study. The people in this study are from age 55 to 90. There is a spike towards around 80-84 years of age for both AD and LOBD patients. The distribution is left-skewed.



Below are violin plots showing the distribution of biomarker values for each biomarker per condition. The distributions of biomarker across patients appear mostly even and similar. Biomarker PGJ2, PGE2, and BDNF appear have the largest difference in biomarker profiles between conditions. However this is not enough information to draw any conclusions about biomarker ranges correlating to condition.



Methods

We performed logistic regressions for control vs AD and control vs LOBD using the statsmodel package, specifically using the statsmodel.formula.api object to instantiate the model and fit() to train the model. Statsmodel uses the “newton” solver to complete the regression. For comparing control to AD and control to LOBD diagnoses, we computed this logistic regression with the formula:

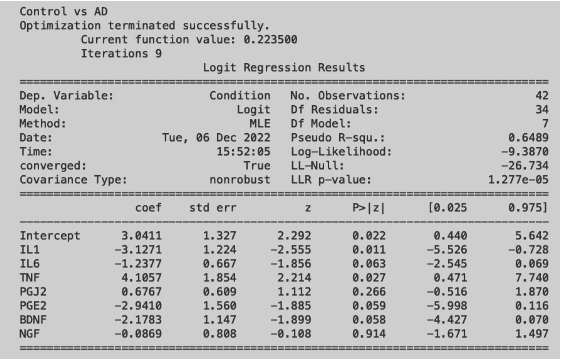
$$Regmodel = Condition \sim IL1 + IL6 + TNF + PGJ2 + PGE2 + BDNF + NGF$$

Then, we were interested in comparing the slopes of the control vs AD regression and the control vs LOBD regression to discover if the two conditions had significantly different biomarker profiles. In order to calculate the significance of the difference between the regression slopes, we used a t-value equation (Soper) to compare correlation coefficients. We accomplished this using the scipy package, specifically scipy.stats.t.sf(). This produced p-values for each biomarker in each set (control vs AD and control vs LOBD) that determine the significance of the difference between the correlations of each biomarker towards each diagnosis.

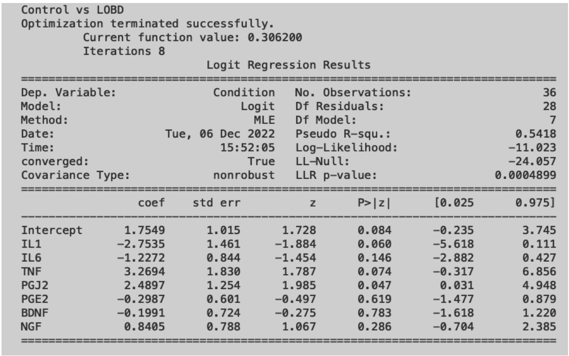
Finally, we plotted the logistic regression model fit of each biomarker. This allowed us to visualize how much each biomarker affected the condition in each dataset. To achieve this, we plotted with the regplot() function of the seaborn package.

Results

In order to test our first hypothesis that biomarkers for both AD and LOBD will significantly differ from control (those without a condition), we performed logistic regressions for both control vs AD and control vs LOBD diagnoses. The regressions gave us correlation coefficients for each biomarker and a p-value indicating if the null hypothesis (that the correlation coefficient is 0) can be accepted or rejected. For AD, two out of the seven biomarkers were statistically significant. Biomarker IL1 and TNF had correlation coefficients of -3.12 and 4.10 and p-values of 0.011 and 0.027, respectively. This means that for these biomarkers we can reject the null hypothesis, and say that these biomarkers have a significant effect on marking Alzheimer's. For LOBD, the PGJ2 biomarker was statistically significant with correlation coefficient of 2.48 a p-value of 0.047. Once again, we can reject the null hypothesis and say that this biomarker has a significant effect on marking late onset Bipolar Disorder. Because each condition was proven to have statistically significant independent variables (biomarkers), we can conclude that our initial hypothesis is correct. We can reject the null hypothesis and accept our alternate hypothesis. Biomarkers for AD and LOBD do significantly differ from control. The LLR p-value for the AD model is $1.277e^{-5}$ and its pseudo R-squared value is 0.648. For the LOBD model, the LLR p-value is 0.0004 and the pseudo R-squared value is 0.541. For both models, the p-values confirm that the models are statistically significant, and the quality of the fit is decent but not great. The details of these two logistic regressions are below.

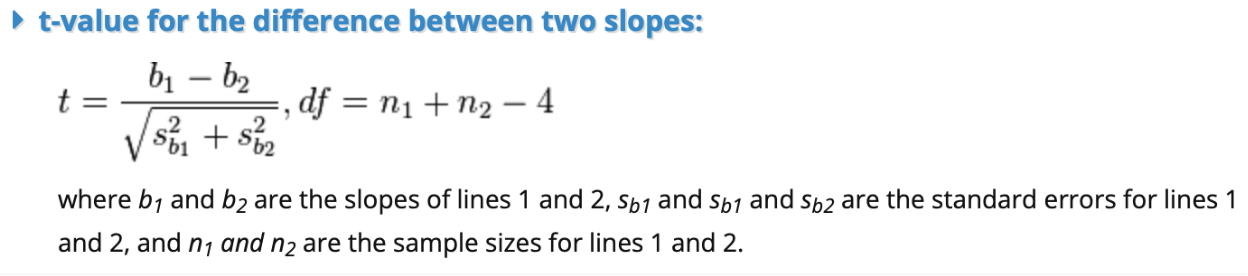


```
Control vs AD
Optimization terminated successfully.
         Current function value: 0.223500
         Iterations 9
                           Logit Regression Results
==============================================================================
Dep. Variable:              Condition   No. Observations:                   42
Model:                          Logit   Df Residuals:                       34
Method:                           MLE   Df Model:                            7
Date:                Tue, 06 Dec 2022   Pseudo R-squ.:                  0.6489
Time:                        15:52:05   Log-Likelihood:                -9.3870
converged:                       True   LL-Null:                       -26.734
Covariance Type:            nonrobust   LLR p-value:                 1.277e-05
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      3.0411      1.327      2.292      0.022       0.440       5.642
IL1           -3.1271      1.224     -2.555      0.011      -5.526      -0.728
IL6           -1.2377      0.667     -1.856      0.063      -2.545       0.069
TNF            4.1057      1.854      2.214      0.027       0.471       7.740
PGJ2           0.6767      0.609      1.112      0.266      -0.516       1.870
PGE2          -2.9410      1.560     -1.885      0.059      -5.998       0.116
BDNF          -2.1783      1.147     -1.899      0.058      -4.427       0.070
NGF           -0.0869      0.808     -0.108      0.914      -1.671       1.497
==============================================================================
```



```
Control vs LOBD
Optimization terminated successfully.
         Current function value: 0.306200
         Iterations 8
                           Logit Regression Results
==============================================================================
Dep. Variable:              Condition   No. Observations:                   36
Model:                          Logit   Df Residuals:                       28
Method:                           MLE   Df Model:                            7
Date:                Tue, 06 Dec 2022   Pseudo R-squ.:                  0.5418
Time:                        15:52:05   Log-Likelihood:                -11.023
converged:                       True   LL-Null:                       -24.057
Covariance Type:            nonrobust   LLR p-value:                 0.0004899
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      1.7549      1.015      1.728      0.084      -0.235       3.745
IL1           -2.7535      1.461     -1.884      0.060      -5.618       0.111
IL6           -1.2272      0.844     -1.454      0.146      -2.882       0.427
TNF            3.2694      1.830      1.787      0.074      -0.317       6.856
PGJ2           2.4897      1.254      1.985      0.047       0.031       4.948
PGE2          -0.2987      0.601     -0.497      0.619      -1.477       0.879
BDNF          -0.1991      0.724     -0.275      0.783      -1.618       1.220
NGF            0.8405      0.788      1.067      0.286      -0.704       2.385
==============================================================================
```

Our second hypothesis was that the biomarker profile between AD and LOBD would also differ significantly from each other. To test this hypothesis we performed a t-test. We used a t-value formula to compare the slopes of our two regressions (control vs AD and control vs LOBD) and tested for their significance (Soper).



▸ **t-value for the difference between two slopes:**

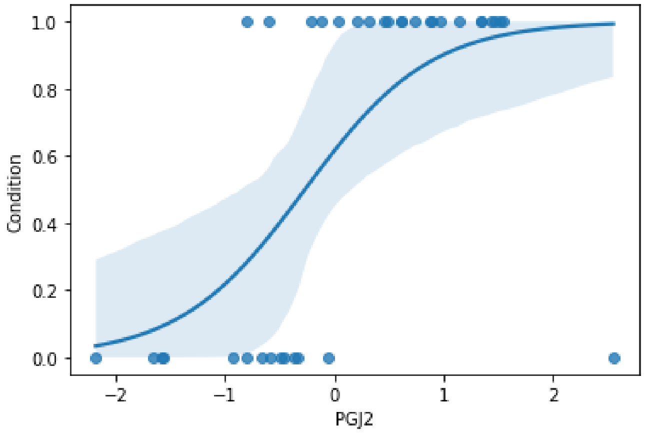
$$t = \frac{b_1 - b_2}{\sqrt{s_{b1}^2 + s_{b2}^2}}, df = n_1 + n_2 - 4$$

where b_1 and b_2 are the slopes of lines 1 and 2, s_{b1} and s_{b1} and s_{b2} are the standard errors for lines 1 and 2, and n_1 *and* n_2 are the sample sizes for lines 1 and 2.

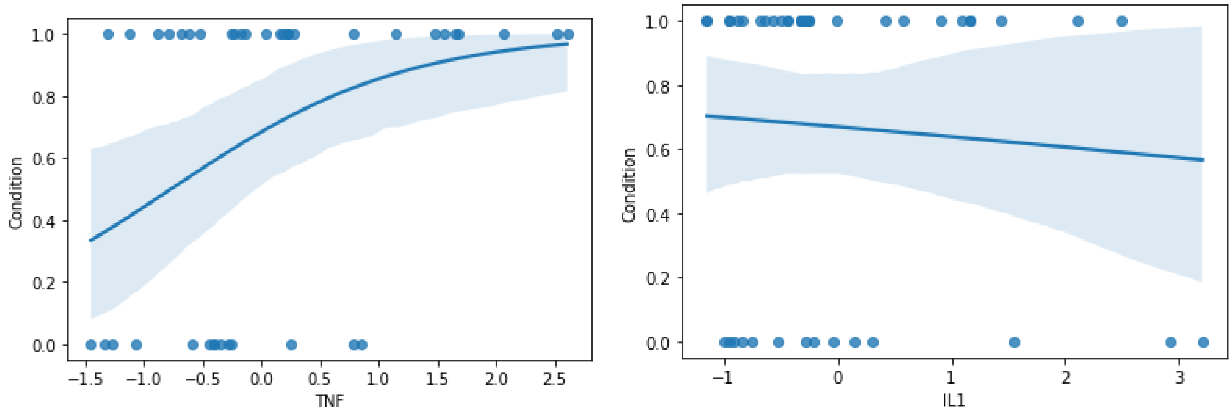
Screenshot of formula from Soper.

We calculated the p value for the difference in correlation coefficient for each biomarker in each regression. For biomarkers IL1, IL6, TNF, PGJ2, PGE2, BDNF, and NGF, their p-values were 0.42, 0.49, 0.37, 0.09, 0.05, 0.07, and 0.20, respectively. No biomarker correlation comparison received a p-value below 0.05. Therefore, the null hypothesis cannot be rejected. The biomarker profiles between AD and LOBD are not statistically significant and our second alternate hypothesis is rejected.

Then, using the two trained regression models, we plotted the regression for each biomarker per condition to evaluate the fit. The more sinusoidal the line of fit is, the more separable the conditions are based on that biomarker. Below are the regression fit plots for the significant biomarkers.



The above plot shows the LOBD regression fit for biomarker PGJ2.



The two plots above show the AD regression fit for biomarkers TNF and IL1.

Conclusion

We tested our two hypotheses about how AD and LOBD biomarker profiles differ from control, and we found that AD biomarkers are significantly different from control - according to the regression, 2 biomarkers had p-values of 0.011 and 0.027. Similarly, for the LOBD regression, one biomarker had a p-value of 0.047. We reject the null hypothesis and accept that biomarker profiles for both AD and LOBD differ significantly from control. For the second hypothesis, we compared the slopes of each regression and found they were not significantly different with p-values below 0.05, even though different biomarkers were significant for each condition. In this case, we concluded that the AD biomarker profile is not significantly different from LOBD biomarker profile.

Discussion

A limitation of this dataset is that it is small. There are only 19 people in the healthy control cohort, when at least 30 samples is desired when fitting data to a normal curve. This makes our analysis a little underpowered. In the future, we could try to find a dataset with more samples of biomarker profiles to make our regression results more robust. A limitation of our analysis is that we did not include age or gender in our logistic regression. In the future, we could include these parameters in our model or even try to find another dataset perhaps more diverse in age.

We found that the fit of the logistic regression between AD and control and LOBD and control were not significantly different from one another. This could suggest that these diseases work in similar pathways, or perhaps have similar effects on the blood. In the future, we could look for other biological measurements that may be able to discriminate between AD and LOBD when combined with biomarker profiles.

References

Ariadna Besga, Manuel Graña, & Darya Chyzhyk. (2020). Alzheimer's Disease versus Bipolar Disorder versus Health Control MRI data and processed results [Data set]. Zenodo. https://doi.org/10.5281/zenodo.3935636

Soper, D. (n.d.). *Formulas: Significance of the difference between two slopes*. Significance of the Difference between Two Slopes Formulas - Free Statistics Calculators. Retrieved December 7, 2022, from https://www.danielsoper.com/statcalc/formulas.aspx?id=103

Appendix

In [1]:

```
"""
Rohith, Rahul, and Jessica
Dataset from https://zenodo.org/record/3935636#.Y3vv-i-B1ZJ
"""
```

Out[1]:

```
'\nRohith, Rahul, and Jessica\nDataset from https://zenodo.org/record/393563
6#.Y3vv-i-B1ZJ\n'
```

In [2]:

```
import os
import numpy as np
import pandas as pd
import seaborn as sns
import scipy.stats
from scipy.stats import zscore
import statsmodels.stats.weightstats as sms
import statsmodels.formula.api as smf
import matplotlib.pyplot as plt
```

In [3]:

```
# Set up file stream access
from google.colab import drive
drive.mount("/content/drive", force_remount=True)

person = input("Enter your name to set up correct file stream access \n")

# edit the path under your name to the path where you copied the final proje
# then run this cell, enter your name when prompted, and the data will be mo
if person == 'Jessica':
  os.chdir("/content/drive/My Drive/DATA690/data")
elif person == 'Rahul':
  os.chdir("/content/drive/My Drive/data/690")
elif person == 'Rohith':
  os.chdir("/content/drive/My Drive/")
else:
  print('Whoops! Make sure you have a user identified')
```

```
Mounted at /content/drive
Enter your name to set up correct file stream access
Jessica
```

In [4]:

```
"""
Combine files for ease of analysis
"""
! paste -d "," clinical_data_id_age_gender.csv inflammation.csv stress.csv >
! head clinical_data_all.csv
```

```
Subject,id,Age,gender,IL1,IL6,TNF,PGJ2,PGE2,BDNF,NGF,Nitritos,MDA
crl,104,79,2,16.678,3.931,2.048844,0.482,224.1214,16132.89474,8924.651163,2.
1932,2.455
crl,105,72,2,12.583,11.266,2.11641,0.553,130.97452,23047.36842,7738.604651,4
.5203,2.455
crl,106,82,1,76.747,6.279,4.819018,0.708,98.36448,16001.31579,14240.93023,0.
3833,3.98
crl,111,65,2,11.218,2.464,0.15701736,0.486,170.48108,21698.68421,15264.18605
,-0.9096,7.031
crl,112,68,1,28.282,5.692,2.11641,0.454,86.50378,21928.94737,9664.186047,-0.
3925,2.455
crl,117,89,2,20.774,-0.469,1.711018,0.54,104.82334,19955.26316,8381.176471,-
1.6853,2.455
crl,118,87,2,39.204,54.981,2.994758,0.531,68.65228,23330.26316,6161.860465,-
2.9781,1.693
crl,122,74,2,73.334,10.093,12.791718,0.51,161.37676,9902.727273,14912,-2.719
6,1.693
crl,124,75,2,81.525,3.345,4.95415,0.487,76.55644,17238.15789,4045.581395,0.1
247,2.455
```

In [5]:

```
"""
subject column - crl is "control, ea is "Alzheimers Disease", tb is "Late On
Il1, Il6, TNF, PGJ2, PGE2, BDNF, NGF columns - measurements of specific biom
Nitritos, MDA columns - treatment measurements, IGNORE
"""
# create dataframe from large combined file
filepath = "clinical_data_all.csv"
df = pd.read_csv(filepath, sep=",", header=0, engine="c")
df
```

Out[5]:

	Subject	id	Age	gender	IL1	IL6	TNF	PGJ2	PGE2	BDNF
0	crl	104	79	2	16.678	3.931	2.048844	0.482	224.12140	16132.894740
1	crl	105	72	2	12.583	11.266	2.116410	0.553	130.97452	23047.368420
2	crl	106	82	1	76.747	6.279	4.819018	0.708	98.36448	16001.315790
3	crl	111	65	2	11.218	2.464	0.157017	0.486	170.48108	21698.684210
4	crl	112	68	1	28.282	5.692	2.116410	0.454	86.50378	21928.947370
...	...	...	...	...	...	...	...	...	...	...
73	tb	334	76	1	26.543	5.503	1.490615	0.654	102.68020	9850.909091
74	tb	336	55	1	14.641	0.676	0.475693	0.643	243.25520	12291.818180
75	tb	338	81	1	14.641	1.641	4.197074	0.577	333.28680	6969.090909
76	tb	339	60	1	31.303	6.469	3.046830	0.597	126.37240	6437.272727
77	tb	340	72	1	219.349	9.607	3.114490	0.538	105.04102	8428.181818

78 rows × 13 columns

In [6]:
```
# rename some columns for clarity and drop some columns we will not use
df.rename(columns={"Subject":"Condition", "id":"ID", "gender":"Gender"}, inp
df.drop(columns=["Nitritos", "MDA"], inplace=True)
df
```

Out[6]:

	Condition	ID	Age	Gender	IL1	IL6	TNF	PGJ2	PGE2	BDN
0	crl	104	79	2	16.678	3.931	2.048844	0.482	224.12140	16132.8947
1	crl	105	72	2	12.583	11.266	2.116410	0.553	130.97452	23047.3684
2	crl	106	82	1	76.747	6.279	4.819018	0.708	98.36448	16001.3157
3	crl	111	65	2	11.218	2.464	0.157017	0.486	170.48108	21698.6842
4	crl	112	68	1	28.282	5.692	2.116410	0.454	86.50378	21928.9473
...	...	...	...	...	...	...	...	...	...	
73	tb	334	76	1	26.543	5.503	1.490615	0.654	102.68020	9850.9090
74	tb	336	55	1	14.641	0.676	0.475693	0.643	243.25520	12291.8181
75	tb	338	81	1	14.641	1.641	4.197074	0.577	333.28680	6969.0909
76	tb	339	60	1	31.303	6.469	3.046830	0.597	126.37240	6437.2727
77	tb	340	72	1	219.349	9.607	3.114490	0.538	105.04102	8428.1818

78 rows × 11 columns

In [7]:
```
# get summary statistics
df.describe()
```

Out[7]:

	ID	Age	Gender	IL1	IL6	TNF	PGJ2	
count	78.000000	78.000000	78.000000	78.000000	78.000000	78.000000	78.000000	7
mean	229.871795	74.923077	1.448718	29.434571	6.707346	3.374383	0.570224	
std	74.929986	8.726388	0.500582	28.477569	11.784815	2.732700	0.056609	7
min	104.000000	55.000000	1.000000	5.074000	-2.817000	-0.248374	0.454000	
25%	161.250000	69.250000	1.000000	14.641000	1.192000	1.579571	0.528000	8
50%	226.500000	75.500000	1.000000	20.091000	2.904500	2.859627	0.562500	1
75%	305.750000	82.000000	2.000000	36.590500	6.766000	4.180159	0.604000	14
max	340.000000	89.000000	2.000000	219.349000	56.448000	12.791718	0.765000	5

In [8]:
```
# check for null values
print(df.isna().sum())
```

```
Condition    0
ID           0
Age          0
Gender       0
IL1          0
IL6          0
TNF          0
PGJ2         0
PGE2         0
BDNF         0
NGF          0
dtype: int64
```

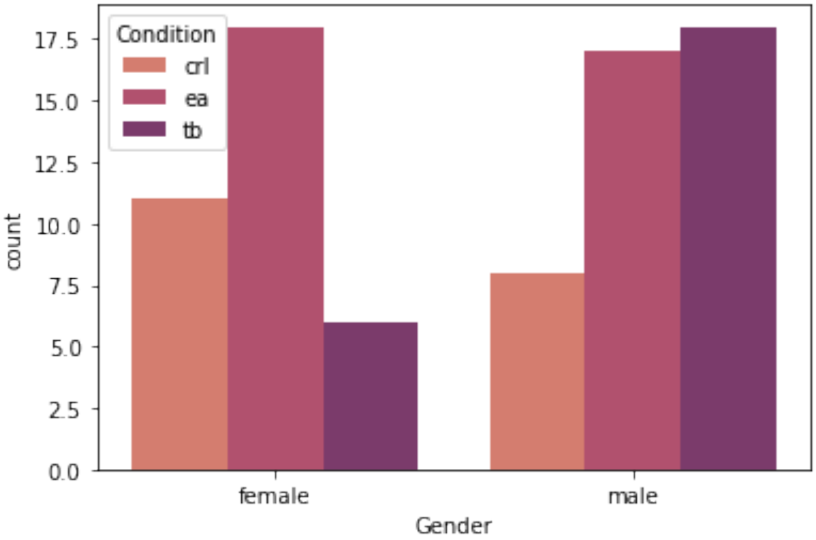
In [9]:

```
# get value counts for gender param and a histogram
gen = {
    1: "male",
    2: "female"
}
df["Gender"] = df["Gender"].replace(to_replace=gen)
print(df["Gender"].value_counts())
sns.countplot(x=df["Gender"], hue=df["Condition"], palette="flare")
```

```
male      43
female    35
Name: Gender, dtype: int64
```

Out[9]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff8f9ba5550>
```

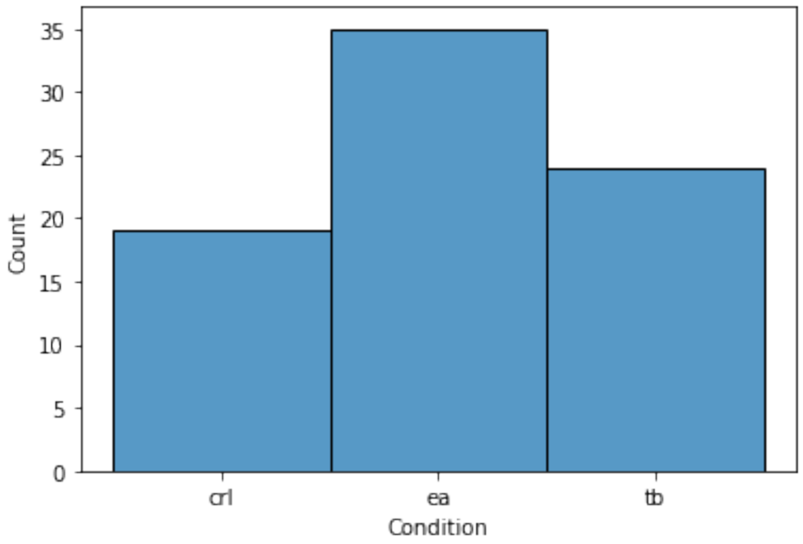


In [10]:

```
# get value counts for condition param and a histogram
print(df["Condition"].value_counts())
sns.histplot(data=df, x="Condition")
```

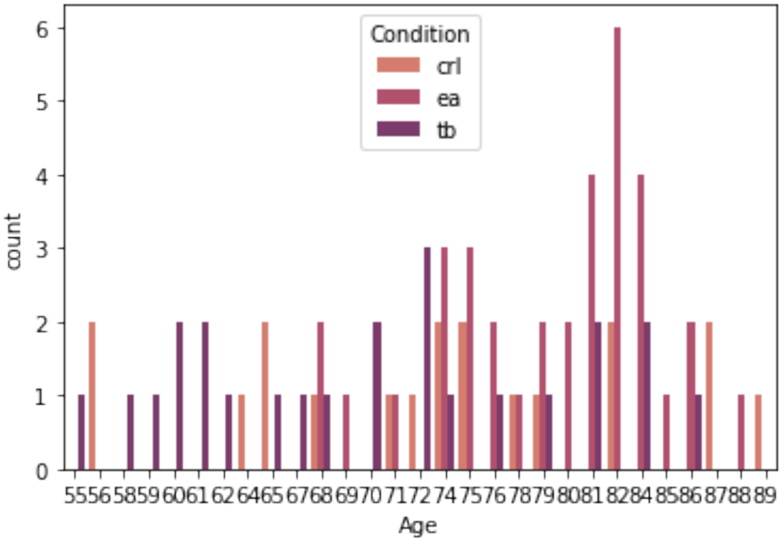
```
ea     35
tb     24
crl    19
Name: Condition, dtype: int64
```

```
Out[10]: <matplotlib.axes._subplots.AxesSubplot at 0x7ff8f9a6daf0>
```



```
In [11]: print(df["Age"].nunique())
         colors = ["lightpink", "blue", "purple"]
         #sns.histplot(x=df["Age"], hue=df["Condition"], bins=27, kde=True, palette=c
         sns.countplot(x=df["Age"], hue=df["Condition"], palette="flare")#, kde=True,
```

```
         29
Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x7ff8f95c8640>
```



```
In [12]: # exclude biomakers that are outliers (marker = 2.5std+avg)
         biomarkers = ["IL1", "IL6", "TNF", "PGJ2", "PGE2", "BDNF", "NGF"]
         for i in biomarkers:
           upper_outlier = df[i].mean() + (2.5*df[i].std())
           lower_outlier = df[i].mean() - (2.5*df[i].std())
           df = df.loc[(df[i]<upper_outlier)]
           df = df.loc[(df[i]>lower_outlier)]
         print(df.shape)
```

```
(64, 11)
```

In [13]:

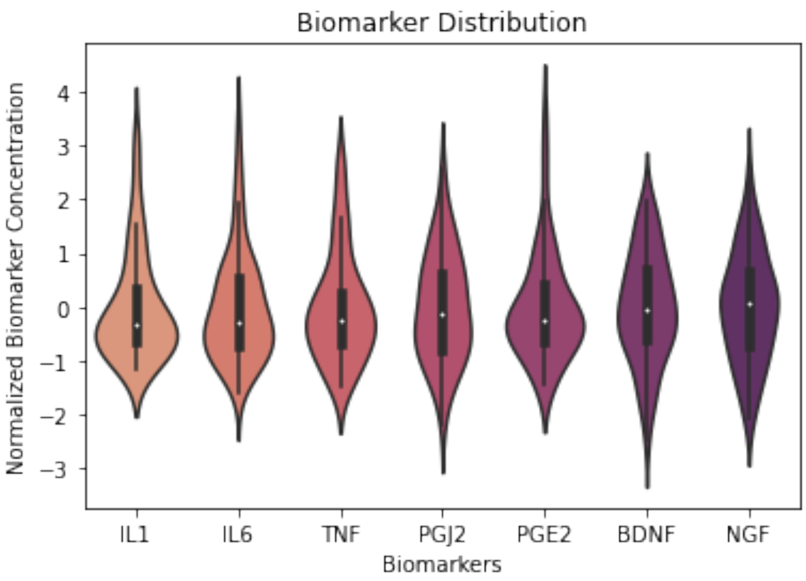
```
# normalize biomarker data by z-scoring
df[biomarkers] = df[biomarkers].apply(zscore)
```

In [14]:

```
# get distribution plots for each biomarker
# these aren't very helpful because they lump all the conditions together
sns.violinplot(data=df[biomarkers], inner="box", palette="flare")
plt.ylabel("Normalized Biomarker Concentration")
plt.xlabel("Biomarkers")
plt.title("Biomarker Distribution")
```

Out[14]:

```
Text(0.5, 1.0, 'Biomarker Distribution')
```

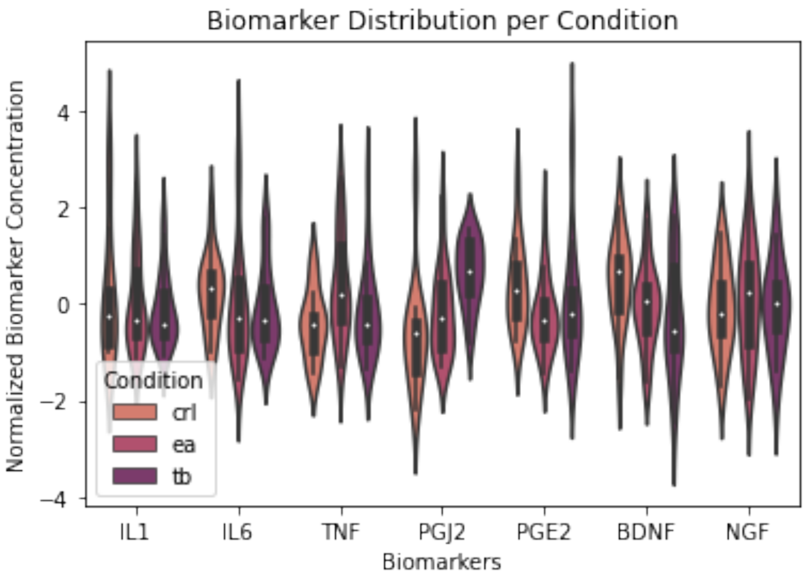


In [15]:

```
# let's get distribution plots for each biomarker per condition
# first transform data from wide to long format so we can use the "hue" para
melted= pd.melt(df, id_vars=["Condition"], value_vars=biomarkers)
sns.violinplot(data=melted, x="variable", y="value", hue="Condition", palett
plt.ylabel("Normalized Biomarker Concentration")
plt.xlabel("Biomarkers")
plt.title("Biomarker Distribution per Condition")
```

Out[15]:

```
Text(0.5, 1.0, 'Biomarker Distribution per Condition')
```



In [16]:

```
# create a subset dataframe that includes only alzheimers and control
alz = df[["Condition", "IL1", "IL6", "TNF", "PGJ2", "PGE2", "BDNF", "NGF"]]
alz = alz[alz.Condition != "tb"]

# create a subset dataframe that includes only late onset bipolar and contro
lobd = df[["Condition", "IL1", "IL6", "TNF", "PGJ2", "PGE2", "BDNF", "NGF"]]
lobd = lobd[lobd.Condition != "ea"]

# convert crl/ea to indicator variables
df_one = pd.get_dummies(alz["Condition"])
df_two = pd.concat((df_one, alz), axis=1)
df_two = df_two.drop(["Condition", "crl"], axis=1)
binary_ad = df_two.rename(columns={"ea": "Condition"})
binary_ad.head()

# convert crl/tb to indicator variables
df_one = pd.get_dummies(lobd["Condition"])
df_two = pd.concat((df_one, lobd), axis=1)
df_two = df_two.drop(["Condition", "crl"], axis=1)
binary_bp = df_two.rename(columns={"tb": "Condition"})
binary_bp.head()
```

Out[16]:

	Condition	IL1	IL6	TNF	PGJ2	PGE2	BDNF	NGF
0	0	-0.521543	0.102536	-0.440248	-1.662412	2.564718	-0.305992	-0.328404
1	0	-0.756831	1.936854	-0.410368	-0.338194	0.438504	0.891055	-0.663266
2	0	2.929876	0.689718	0.784841	2.552703	-0.305868	-0.328772	1.172566
3	0	-0.835261	-0.264328	-1.276895	-1.587808	1.340299	0.657568	1.461466
4	0	0.145195	0.542922	-0.410368	-2.184638	-0.576606	0.697432	-0.119608

In [17]:

```
# logistic regression - control vs AD
print("Control vs AD")
log_reg1 = smf.logit("Condition ~ IL1 + IL6 + TNF + PGJ2 + PGE2 + BDNF + NGF
print(log_reg1.summary())
```

```
Control vs AD
Optimization terminated successfully.
         Current function value: 0.223500
         Iterations 9
                           Logit Regression Results
============================================================================
==
Dep. Variable:              Condition   No. Observations:
42
Model:                          Logit   Df Residuals:
34
Method:                           MLE   Df Model:
7
Date:                Mon, 12 Dec 2022   Pseudo R-squ.:                  0.64
89
Time:                        14:43:53   Log-Likelihood:                -9.38
70
converged:                       True   LL-Null:                       -26.7
34
Covariance Type:            nonrobust   LLR p-value:                 1.277e-
05
============================================================================
==
                 coef    std err          z      P>|z|      [0.025      0.97
5]
----------------------------------------------------------------------------
--
Intercept      3.0411      1.327      2.292      0.022       0.440       5.6
42
IL1           -3.1271      1.224     -2.555      0.011      -5.526      -0.7
28
IL6           -1.2377      0.667     -1.856      0.063      -2.545       0.0
69
TNF            4.1057      1.854      2.214      0.027       0.471       7.7
40
PGJ2           0.6767      0.609      1.112      0.266      -0.516       1.8
70
PGE2          -2.9410      1.560     -1.885      0.059      -5.998       0.1
16
BDNF          -2.1783      1.147     -1.899      0.058      -4.427       0.0
70
NGF           -0.0869      0.808     -0.108      0.914      -1.671       1.4
97
============================================================================
==

Possibly complete quasi-separation: A fraction 0.12 of observations can be
perfectly predicted. This might indicate that there is complete
quasi-separation. In this case some parameters will not be identified.
```

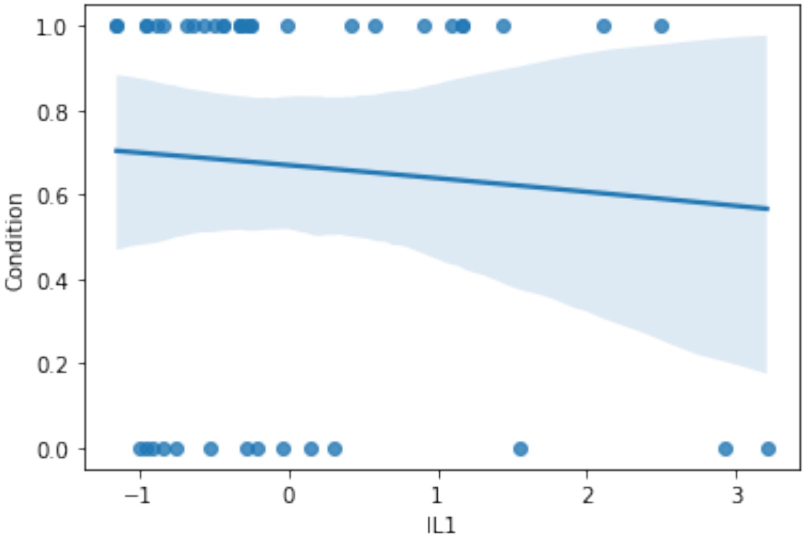
In [18]:

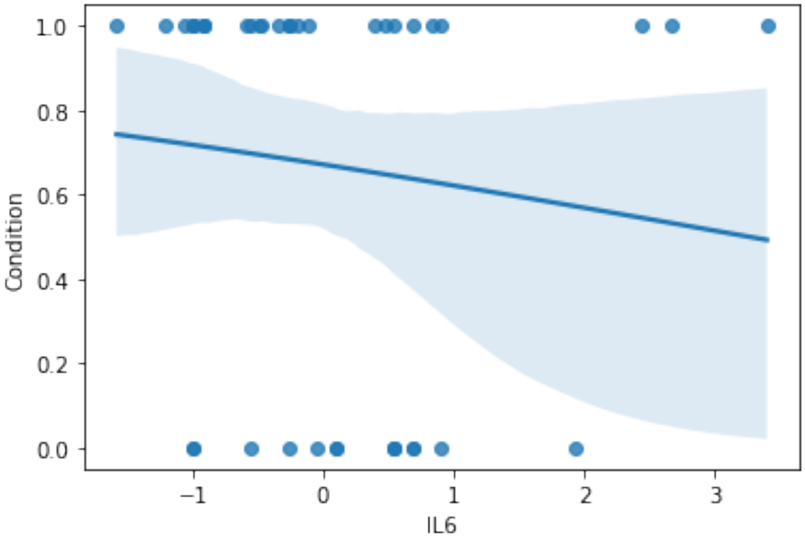
```
# logistic regression - control vs LOBD
print("Control vs LOBD")
log_reg2 = smf.logit("Condition ~ IL1 + IL6 + TNF + PGJ2 + PGE2 + BDNF + NGF
print(log_reg2.summary())
```

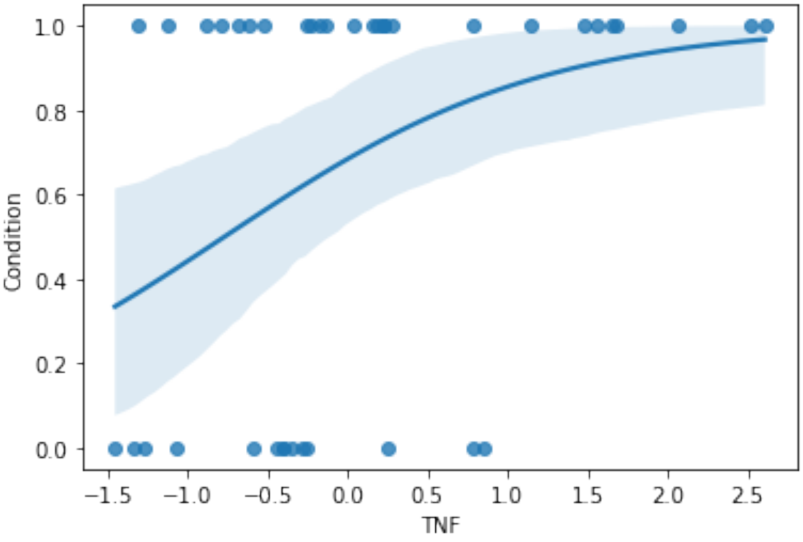
```
Control vs LOBD
Optimization terminated successfully.
         Current function value: 0.306200
         Iterations 8
                           Logit Regression Results
============================================================================
==
Dep. Variable:              Condition   No. Observations:
36
Model:                          Logit   Df Residuals:
28
Method:                           MLE   Df Model:
7
Date:                Mon, 12 Dec 2022   Pseudo R-squ.:                  0.54
18
Time:                        14:43:53   Log-Likelihood:                -11.0
23
converged:                       True   LL-Null:                       -24.0
57
Covariance Type:            nonrobust   LLR p-value:                 0.00048
99
============================================================================
==
                 coef    std err          z      P>|z|      [0.025      0.97
5]
----------------------------------------------------------------------------
--
Intercept      1.7549      1.015      1.728      0.084      -0.235       3.7
45
IL1           -2.7535      1.461     -1.884      0.060      -5.618       0.1
11
IL6           -1.2272      0.844     -1.454      0.146      -2.882       0.4
27
TNF            3.2694      1.830      1.787      0.074      -0.317       6.8
56
PGJ2           2.4897      1.254      1.985      0.047       0.031       4.9
48
PGE2          -0.2987      0.601     -0.497      0.619      -1.477       0.8
79
BDNF          -0.1991      0.724     -0.275      0.783      -1.618       1.2
20
NGF            0.8405      0.788      1.067      0.286      -0.704       2.3
85
============================================================================
==
```

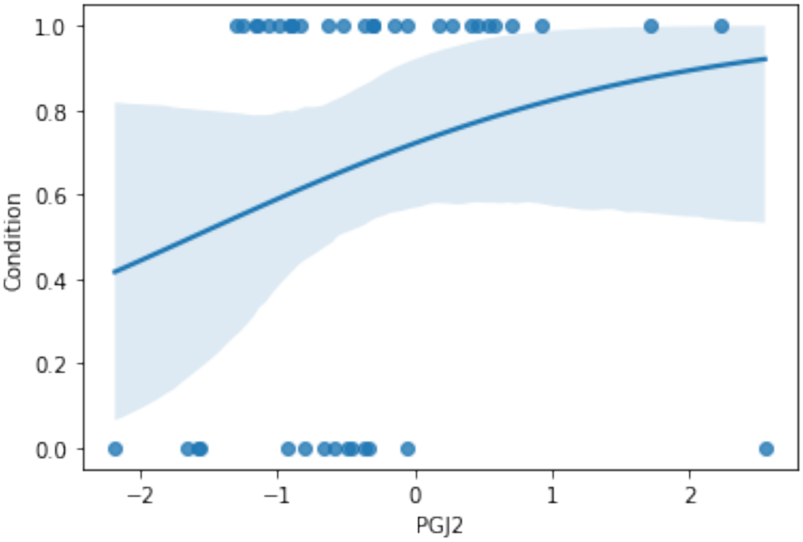
In [19]:

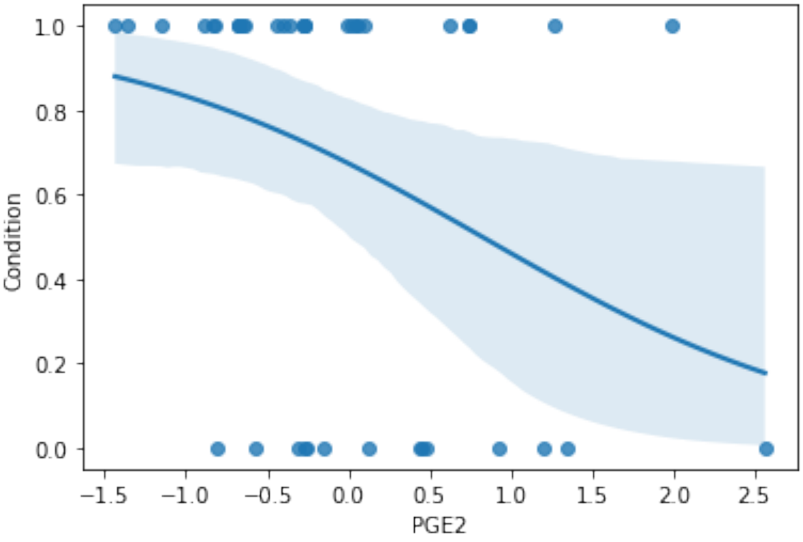
```
# logistic regression for control vs AD per biomarkers
# more sinusoidal shapes indicate that the biomarker has a more significant
for i in biomarkers:
  sns.regplot(y="Condition", x=i, data=binary_ad, logistic=True)
  plt.show()
```

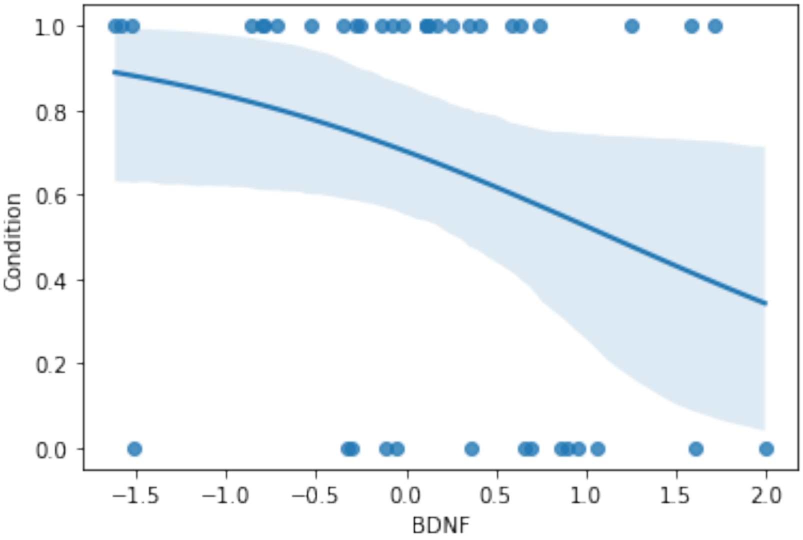


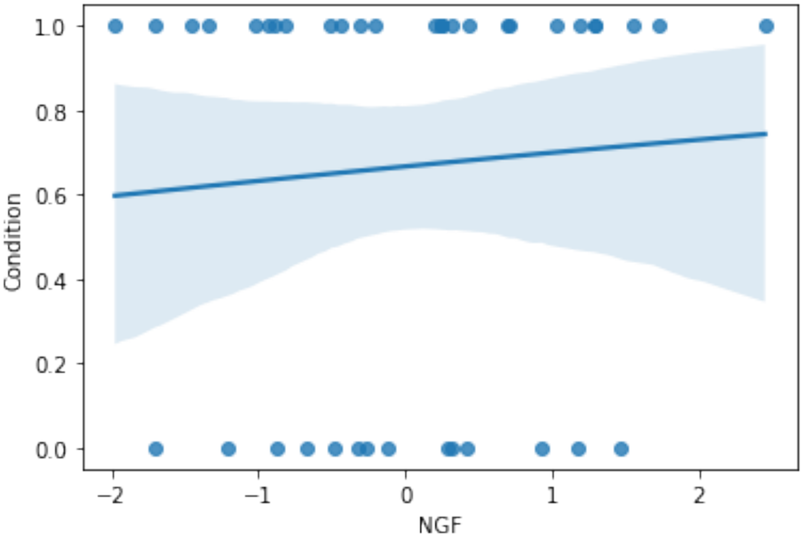



1.0
0.8
0.6
0.4
0.2
0.0
Condition
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
TNF


1.0
0.8
0.6
0.4
0.2
0.0
Condition
−2
−1
0
1
2
PGJ2

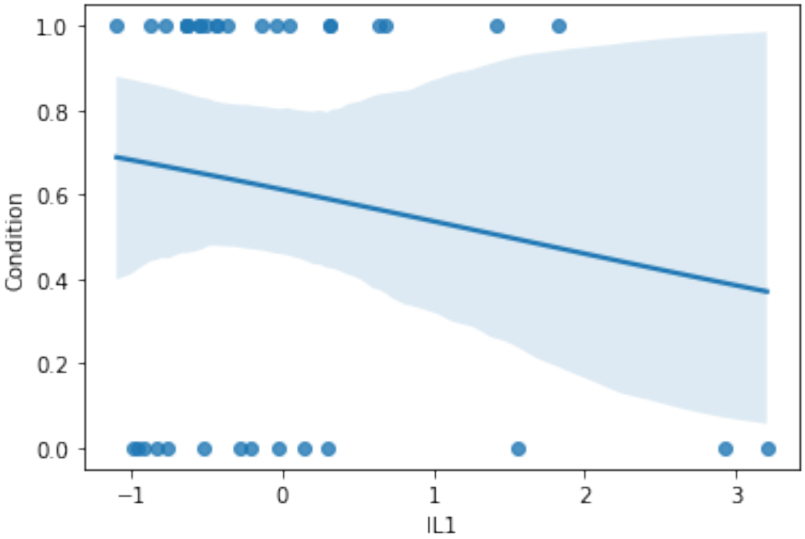

1.0
0.8
0.6
0.4
0.2
0.0
Condition
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
PGE2

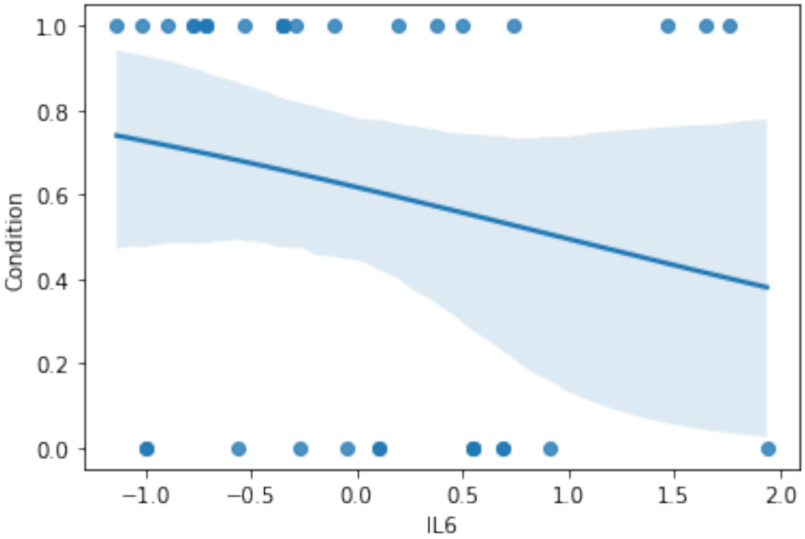


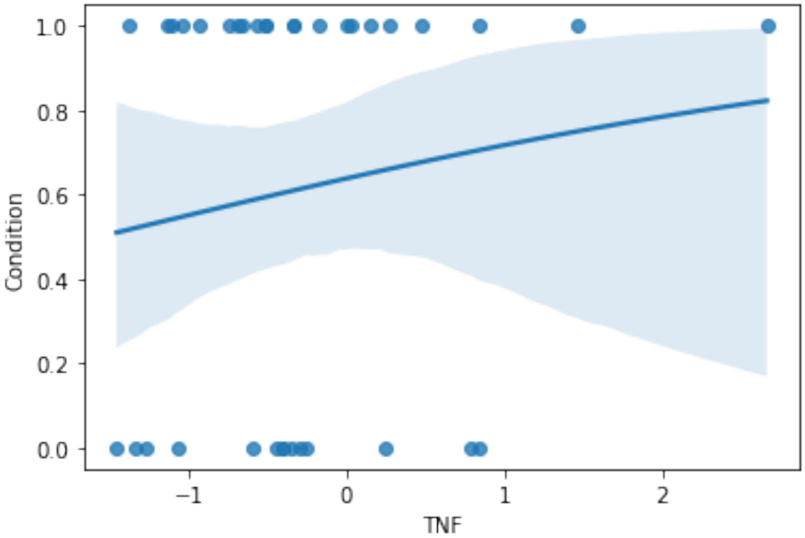


In [20]:

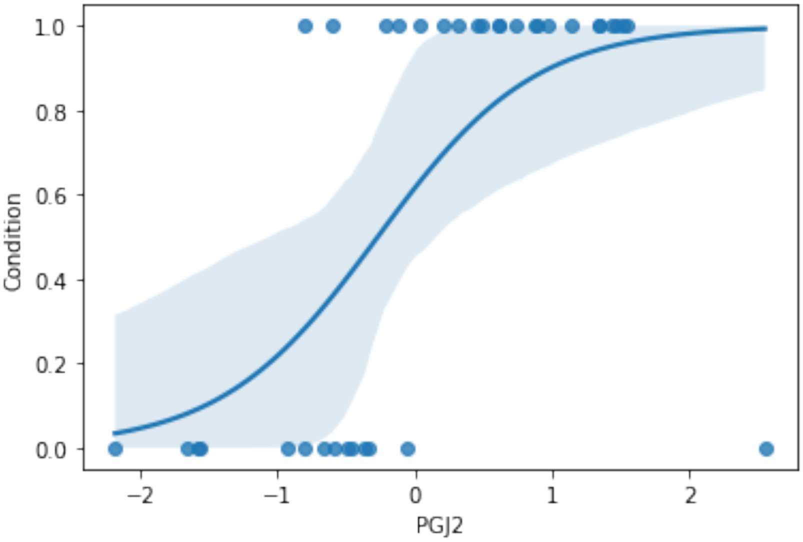
```
# logistic regression for control vs LOBD per biomarkers
# more sinusoidal shapes indicate that the biomarker has a more significant
for i in biomarkers:
  sns.regplot(y="Condition", x=i, data=binary_bp, logistic=True)
  plt.show()
```

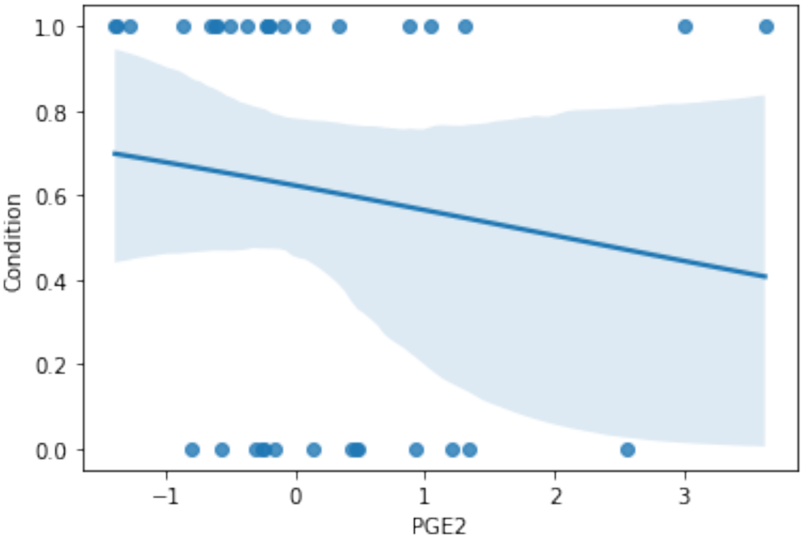


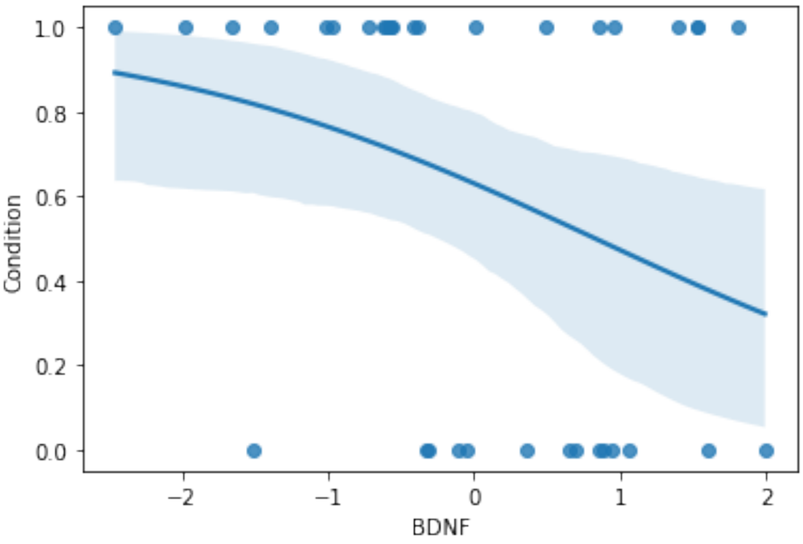


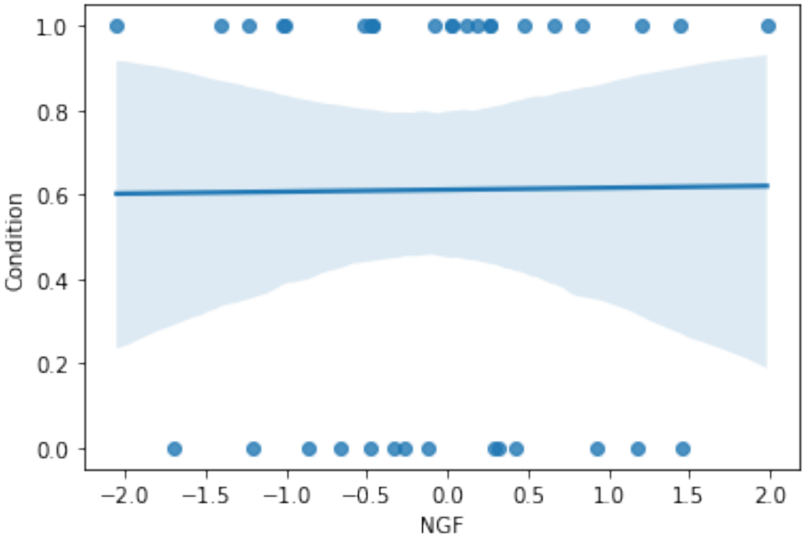


```
/usr/local/lib/python3.8/dist-packages/statsmodels/genmod/families/links.py:
188: RuntimeWarning: overflow encountered in exp
  t = np.exp(-z)
```


Condition
1.0
0.8
0.6
0.4
0.2
0.0
−2
−1
0
1
2
PGJ2


Condition
1.0
0.8
0.6
0.4
0.2
0.0
−1
0
1
2
3
PGE2


Condition
1.0
0.8
0.6
0.4
0.2
0.0
−2
−1
0
1
2
BDNF



In [21]:

```
# compare slopes of biomarkers between AD logistic regression and LOBD logis
# if pval < 0.05, the slope is significantly different, meaning that the spe
# if pval > 0.05, the slope is not significantly different, meaning that the
t_value = lambda b1, b2, s1, s2 : (b1-b2)/np.sqrt(s1**2 + s2**2)
df = log_reg1.df_model + log_reg1.df_resid + log_reg2.df_model + log_reg2.df
p_values = {}

for bio in biomarkers:
  temp = t_value(log_reg1.params[bio], log_reg2.params[bio], log_reg1.bse[bi
  p_values[bio] = scipy.stats.t.sf(abs(temp), df)

p_values
```

Out[21]:

```
{'IL1': 0.4225927140003683,
 'IL6': 0.4961241247889039,
 'TNF': 0.3745672824417241,
 'PGJ2': 0.09883158130506635,
 'PGE2': 0.05918011319320679,
 'BDNF': 0.07445322118263474,
 'NGF': 0.20697574845164324}
```