```
import pandas #for dataframe
from math import pi
import math #always useful
import numpy as np #always useful (linear algebra expecially)
import scipy #for various mathematical function you may want
import statsmodels.api as sm #The most complete python pluggin
#                             for statistical modeling in my sense
from statsmodels.formula.api import ols #For ordinary least square
#                                       "ols" not the same as above
anova_lm = sm.stats.anova_lm #The anova function from statsmodels

# Documentation
# https://www.statsmodels.org/devel/regression.html
# https://www.statsmodels.org/devel/anova.html
# https://www.statsmodels.org/devel/examples/notebooks/generated/interactions_anova.htm



#e.g. quadratic model

c1=["H     ",    "t-C4H9    ",    "i-C3H7    ",    "C2H5      ",
    "CH3       ",    "CH2OH     ",    "C6H5      ",    "CH=CH2    ",
    "NH2       ",    "NHCH3     ",    "N(CH3)2   ",    "OH        ",
    "CO2C2H5   ",   "OCH3      ",    "Br        ",    "Cl        ",
    "F         ",    "CN        ",    "NO2"]
c2=[0    ,    0    ,    0    ,    0    ,    0    ,    0.14 ,    0.1  ,
    0.06 ,    0.14 ,    0.12 ,    0.1  ,    0.3  ,    0.31 ,    0.28 ,
    0.45 ,    0.45 ,    0.44 ,    0.6  ,    0.65]
c3=[0,-0.07,-0.07,-0.07,-0.08,-0.06,-0.22,-0.15,-0.52,-0.58,-0.64,-0.38,
    0    ,-0.42,-0.15,-0.17,-0.25,0    ,0]
c4=[4390,4180,4060,3950,3850,3510,3350,3260,2600,2520,2250,2110,2020,
    1700,1290,1230,1060,1000,660]

df = pandas.DataFrame(data={'Amines': np.asarray(c1),
     'SigmaF': np.asarray(c2), 'SigmaR': np.asarray(c3),
     'K': np.asarray(c4)},)

mod_quad = ols('K ~ SigmaF*SigmaR + np.square(SigmaF) + np.square(SigmaR)', data=df)
res_quad=mod_quad.fit()# fitting of the model
print(res_quad.summary()) #returns the fitted model with : estimates, SE,
                          # t-value, p-value, confidence interval, and
                          # various other statistical test and information
table_quad = anova_lm(res_quad, typ=2) #type II anova
print(table_quad)
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                          K   R-squared:
0.975
Model:                                OLS   Adj. R-squared:
0.965
Method:                     Least Squares   F-statistic:
101.5
Date:                    Tue, 29 Mar 2022   Prob (F-statistic):
6.03e-10
Time:                            17:25:12   Log-Likelihood:
-126.44
No. Observations:                      19   AIC:
```

```
264.9
Df Residuals:                      13   BIC:
270.5
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================================
                        coef    std err          t      P>|t|
[0.025      0.975]
----------------------------------------------------------------------------------------------
Intercept          4194.9873    158.573     26.455      0.000
3852.411    4537.564
SigmaF            -6972.1093   1330.464     -5.240      0.000
-9846.401   -4097.817
SigmaR             1633.6887   1897.863      0.861      0.405
-2466.395    5733.772
SigmaF:SigmaR      -979.7089   3568.503     -0.275      0.788
-8688.991    6729.573
np.square(SigmaF)  2324.3059   2085.024      1.115      0.285
-2180.115    6828.727
np.square(SigmaR)  -203.0594   2426.333     -0.084      0.935
-5444.834    5038.715
==============================================================================
Omnibus:                        1.997   Durbin-Watson:
2.494
Prob(Omnibus):                  0.368   Jarque-Bera (JB):
1.132
Skew:                           0.244   Prob(JB):
0.568
Kurtosis:                       1.908   Cond. No.
90.2
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
                         sum_sq    df          F    PR(>F)
SigmaF             1.895683e+06   1.0  36.750588  0.000040
SigmaR             5.663795e+04   1.0   1.098010  0.313795
SigmaF:SigmaR      3.887974e+03   1.0   0.075374  0.787982
np.square(SigmaF)  6.410112e+04   1.0   1.242694  0.285137
np.square(SigmaR)  3.612824e+02   1.0   0.007004  0.934578
Residual           6.705709e+05  13.0        NaN       NaN
/home/thethemas/.local/lib/python3.9/site-
packages/scipy/stats/stats.py:1541: UserWarning: kurtosistest only
valid for n>=20 ... continuing anyway, n=19
  warnings.warn("kurtosistest only valid for n>=20 ... continuing "
```