

```

import pandas

df = pandas.read_csv('http://pogo.software/me4ml/xray.csv')

x = np.array(df['Distance (mm)'][:])
y = np.array(df['Total absorption'][:])

---

# read in the CSV file
df = pandas.read_csv('http://pogo.software/me4ml/hdpeVel.csv')
# set the 'index' column as the one containing the temperature values
df = df.set_index('T/C f/MHz')

# extract the frequency values (and scale since they are MHz)
freq = df.columns.values.astype(np.float) * 1e6
# extract the temperature values
temp = df.index.values.astype(np.float)

# extract the main part - the velocity values
vel = df.to_numpy()
# calculate the total number of values
tot_values = len(freq) * len(temp)

---

x1grid, x2grid = np.meshgrid(freq, temp)
Xgrid = np.concatenate([x1grid.reshape([tot_values, 1]),
                        x2grid.reshape([tot_values, 1])], axis=1)
ygrid = vel.reshape([tot_values, 1])

---

from sklearn.linear_model import LinearRegression

reg = LinearRegression()
reg.fit(Xgrid, ygrid)

---

from mpl_toolkits.mplot3d import Axes3D

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.scatter(Xgrid[:, 0], Xgrid[:, 1], ygrid, marker='x', color='#000000')
ax.scatter(Xgrid[:, 0], Xgrid[:, 1], y_lin, marker='o', color='#ff0000')

---

import plotly.graph_objects as go

fig = go.Figure()
fig.add_trace(go.Scatter3d(x=Xgrid[:, 0], y=Xgrid[:, 1], z=ygrid[:, 0],
mode='markers',
                                marker=dict(size=2, color='#000000',
symbol='x'))))

fig.add_trace(go.Scatter3d(x=Xgrid[:, 0], y=Xgrid[:, 1], z=y_lin[:, 0],
mode='markers',
                                marker=dict(size=3, color='#ff0000',
symbol='circle'))))

---

```

```
from sklearn.preprocessing import PolynomialFeatures
...
poly = PolynomialFeatures(degree = 2)
X_poly = poly.fit_transform(Xgrid)

print(X_poly.shape)
print(poly.powers_)
---

reg_poly = LinearRegression()
reg_poly.fit(X_poly, ygrid)
```