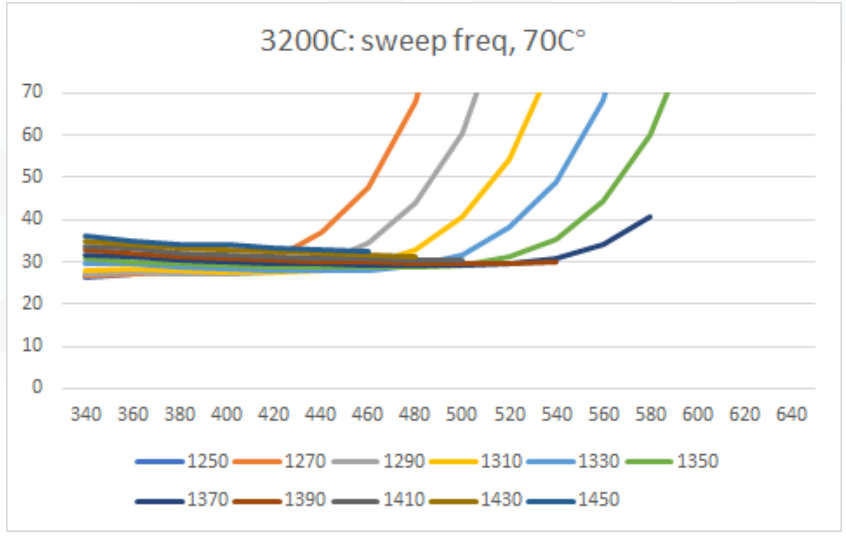
DIGWISE

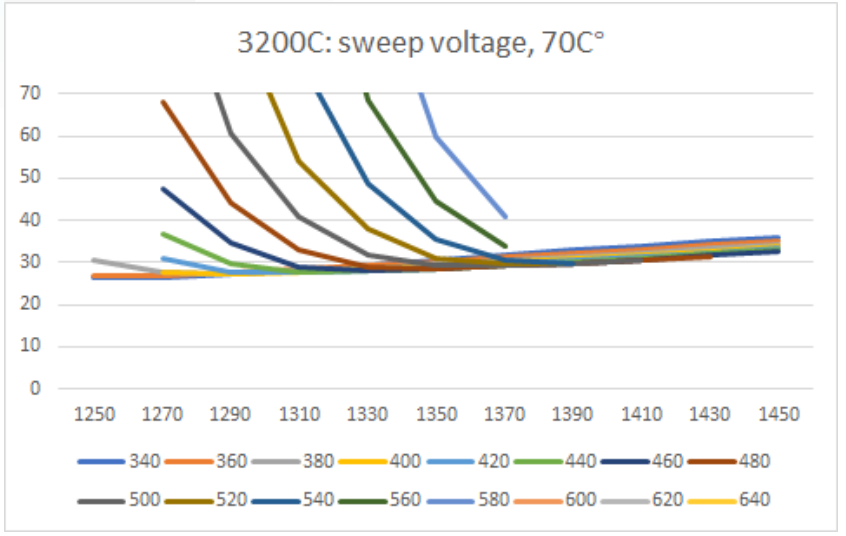
Interpolation Skill for Data Visualization

2022

Raw Data

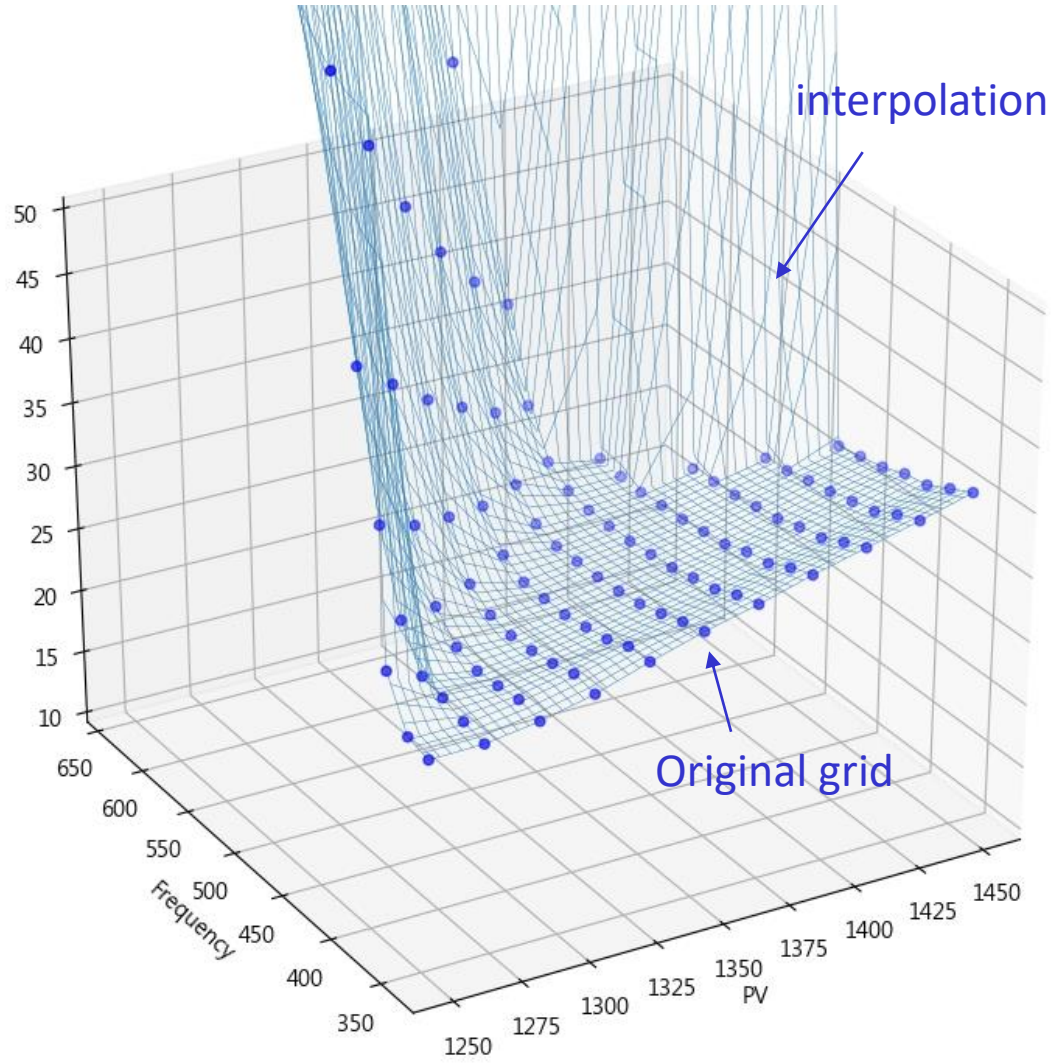
	1250	1270	1290	1310	1330	1350	1370	1390	1410	1430	1450
340	26.50509	26.59157	27.1862	28.14574	29.51405	30.73203	31.72863	32.89834	33.9263	34.94256	36.05177
360	26.95046	26.97829	27.55882	28.4125	29.42172	30.22486	31.20882	32.21728	33.12736	34.19985	35.16703
380	30.68099	27.51062	27.31884	27.93992	28.70529	29.59315	30.41023	31.34029	32.25423	33.25292	34.24931
400		27.88477	27.18547	27.65703	28.37764	29.09974	30.03478	30.9954	31.86741	32.81242	33.93663
420		30.91877	27.74744	27.54507	28.07242	28.83726	29.59269	30.37007	31.31482	32.37692	33.2635
440		36.94148	29.6405	27.87877	28.07848	28.71337	29.40735	30.13395	31.04852	32.05401	32.91569
460		47.65634	34.66393	29.0546	28.13498	28.67888	29.21368	29.8847	30.75339	31.64041	32.55779
480		67.91027	44.19169	33.036	29.02712	28.64713	29.17325	29.76799	30.60303	31.43807	
500		106.648	60.66285	40.86289	31.64315	29.17153	29.23021	29.59278	30.44172		
520		196.4025	91.5964	54.24083	38.13373	31.08903	29.57353	29.68012			
540		431.3268	156.7105	78.87226	48.84458	35.53458	30.66859	29.90678			
560		1276.016	317.8185	128.5704	68.29608	44.52476	33.93815				
580		5061.671	856.9483	237.6945	106.2133	60.00778	40.73559				
600			1904.178	493.333	175.2976	87.23597					
620				1298.176	365.9515						
640			550.5154								

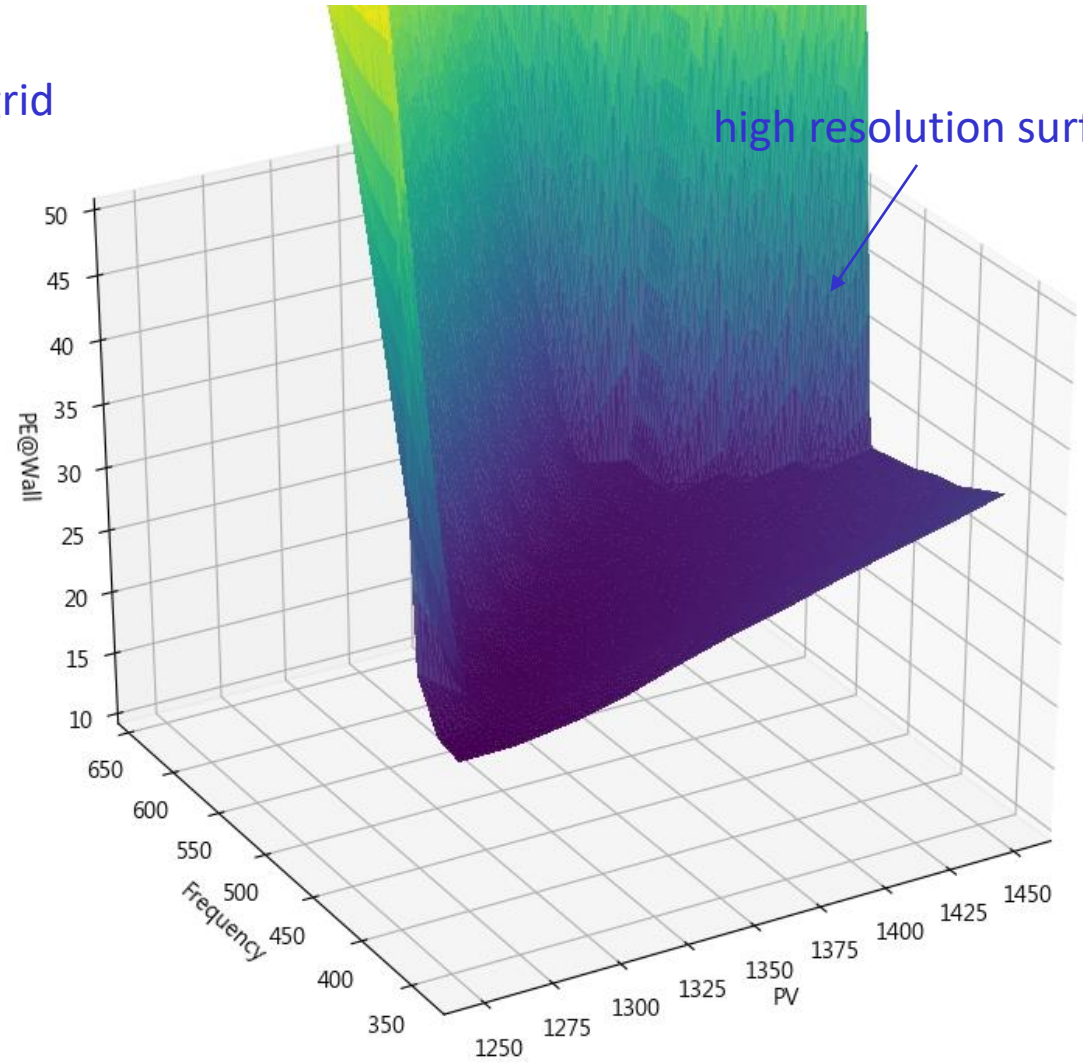




DIGWISE TECHNOLOGY
慧智科技有限公司

Resolution Enhancement





DIGWISE TECHNOLOGY
睿智科技有限公司

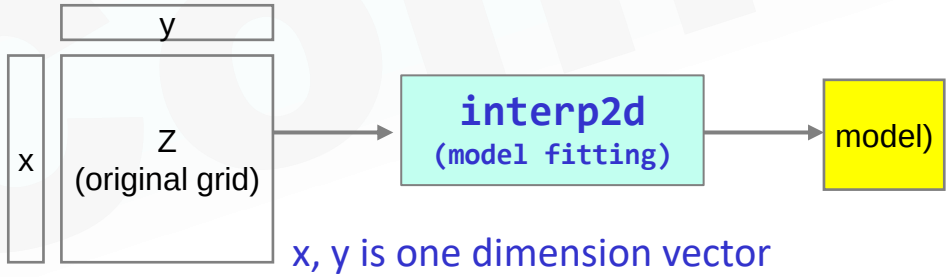
Skill

```
# A3200C Machine-level PE
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# load data
xlsx =  'PW_wall.xlsx'
df = pd.read_excel(xlsx,sheet_name='PE')
df = df.rename(columns={'Unnamed: 0':'Freq'}).set_index(['Freq'])

# 2D interpolation
from scipy.interpolate import interp2d
y,x,z = df.index.values.astype(int), \
        df.columns.values.astype(int), \
        df.values.astype(float)
gx,gy = np.meshgrid(x,y)
tx,ty = gx.ravel(),gy.ravel()
```

```
# model fitting
z = np.nan_to_num(z.clip(0,80),nan=80)
model = interp2d(x,y,z,kind='linear') # linear interpolation
```



x, y is one dimension vector
and z is 2-dimension vector

```
# interpolation grid
num = 100
ix,iy = np.linspace(x.min(),x.max(),num),np.linspace(y.min(),y.max(),num)
igx,igy = np.meshgrid(ix,iy)
itx,ity = igx.ravel(),igy.ravel()
pz = model(ix,iy) # prediction
```

```
# visualization
f = plt.figure(figsize=(8,7))
ax = Axes3D(f)
ax.scatter(tx,ty,z,c='b ') # original data point
#p = model(x,y) # fit original grid
#ax.plot_trisurf(tx,ty,p.ravel(),alpha=0.5,cmap=plt.cm.Blues)
ax.plot_trisurf(itx,ity,pz.ravel(),alpha=0.9,
    antialiased=False,linewidth=0,edgecolors='none',cmap=plt.cm.viridis)
#ax.plot_wireframe(igx,igy,pz,alpha=1,lw=0.3,cmap=plt.cm.Blues)
ax.set_xlabel('PV')
ax.set_ylabel('Frequency')
ax.set_zlabel('PE@Wall')
ax.set_zlim(top=50,bottom=10)
ax.view_init(30,240)
plt.show()
```