

Python for Visualization

In Solid Tumor Studies

Figures to be Discussed

- ▶ Spider Plot
- ▶ Waterfall Plot
- ▶ Swimmer Plot
- ▶ Forest Plot for ORR

Variables Required in ADSL

- ▶ SUBJID
- ▶ TRTDURM
- ▶ DTHDY
- ▶ EOTSTT

Variables Required in ADTR

- ▶ SUBJID
- ▶ PARAMCD
- ▶ AVAL
- ▶ PCHG
- ▶ ABLFL
- ▶ ADY
- ▶ ANL01FL
- ▶ PARQUAL
- ▶ TRTP

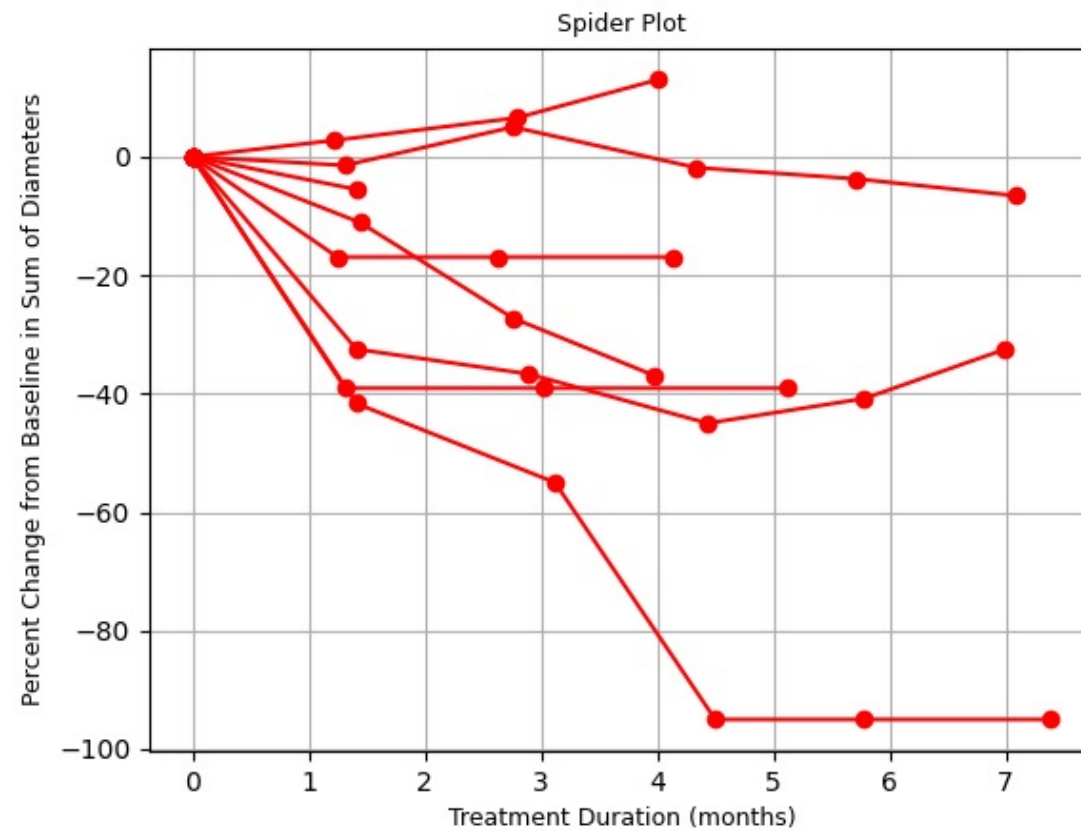
Variables Required in ADRS

- ▶ SUBJID
- ▶ PARAMCD
- ▶ AVALC
- ▶ ADY
- ▶ PARQUAL

Python Program Flow

- ▶ Import the Python Modules
- ▶ Read the ADaM datasets
- ▶ Create the variables required for the plot
- ▶ Do the plot

Spider Plot



Spider Plot Code

```
#spider.py
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_sas('adtr.sas7bdat')
df2=df[(df["PARQUAL"]==b'INVESTIGATOR') & (df['PARAMCD']==b'SUMDIAM') &
(df['ANL01FL']==b'Y')]
df2=df2.assign(ADY2=df2['ADY']/(365.25/12))

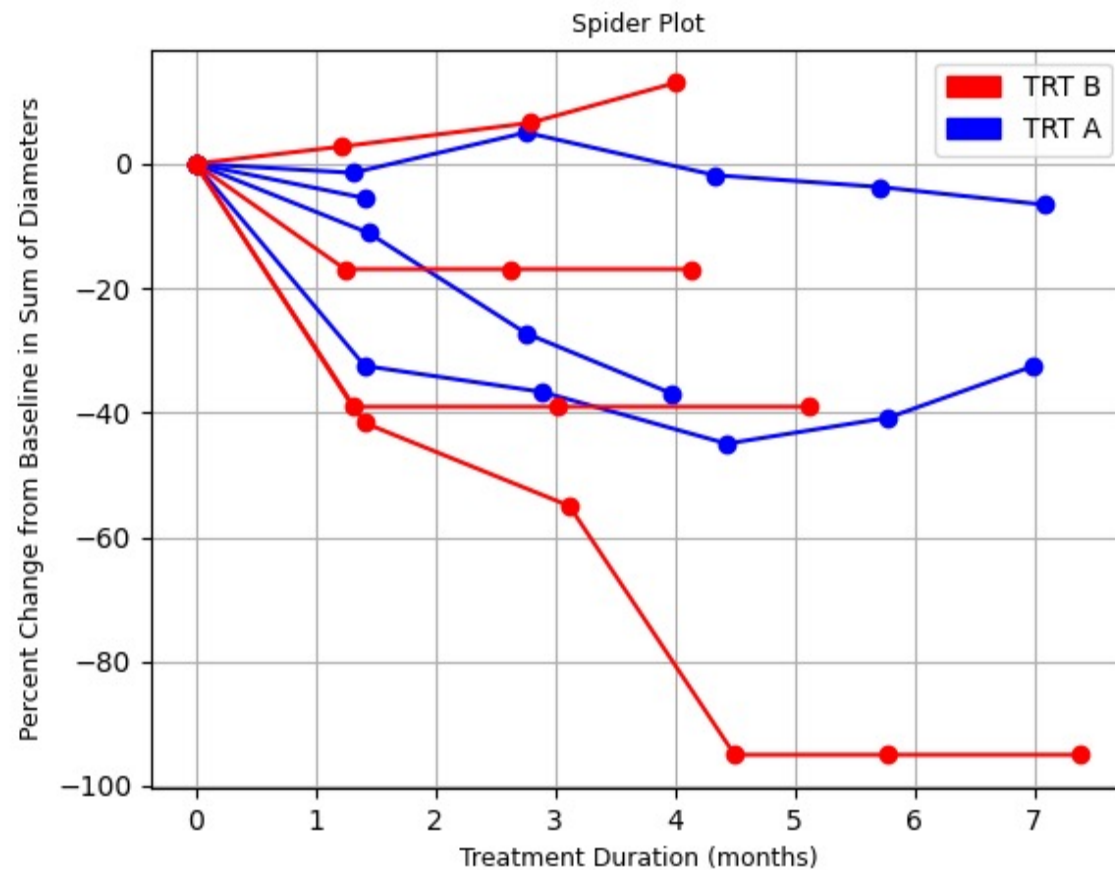
df2.loc[df2['ABLFL']==b'Y','ADY2']=0
df2.loc[df2['ABLFL']==b'Y','PCHG']=0
```

Spider Plot Code, Continued

```
df2.set_index('ADY2', inplace=True)
df2.groupby('SUBJID')['PCHG'].plot(color='red', marker='o')

plt.title('Spider Plot', fontsize=9)
plt.xlabel('Treatment Duration (months)', fontsize=9)
plt.ylabel('Percent Change from Baseline in Sum of
Diameters', fontsize=9)
plt.grid(True)
plt.show()
```

Spider Plot for Two Treatments



Spider plot for Two Treatments, Code

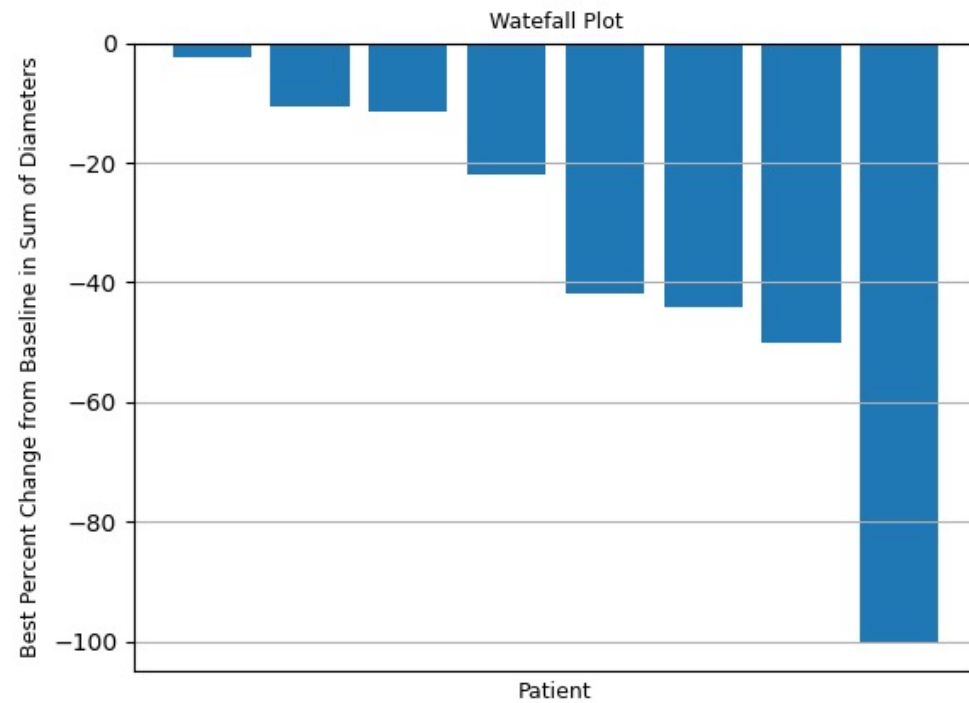
```
df2=df[... & (df['TRTP']==b'TRT A')]
```

```
df3=df[... & (df['TRTP']==b'TRT B')]
```

```
df2.groupby('SUBJID')['PCHG'].plot(color='blue', marker='o')
```

```
df3.groupby('SUBJID')['PCHG'].plot(color='red', marker='o')
```

Waterfall Plot



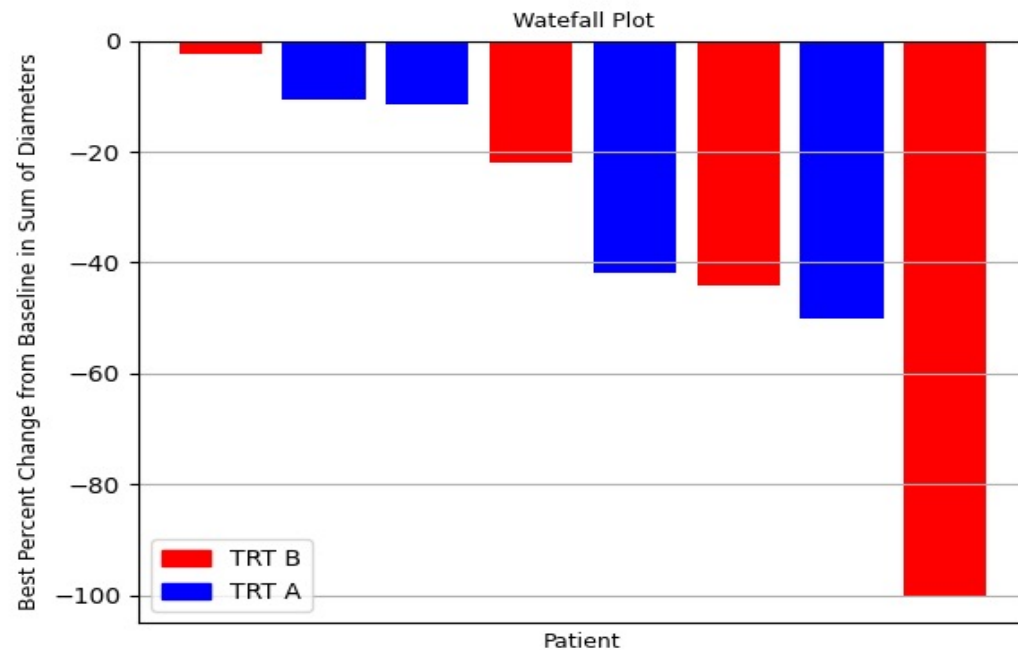
Waterfall Plot Code

```
df2=df[(df["PARQUAL"]==b'INVESTIGATOR') &  
(df['PARAMCD']==b'BESTPCHG')]
```

```
df_sorted=df2.sort_values('AVAL', ascending=False)
```

```
plt.bar('SUBJID', 'AVAL', data=df_sorted)
```

Waterfall Plot for Two Treatments



Waterfall Plot for Two Treatments, Code

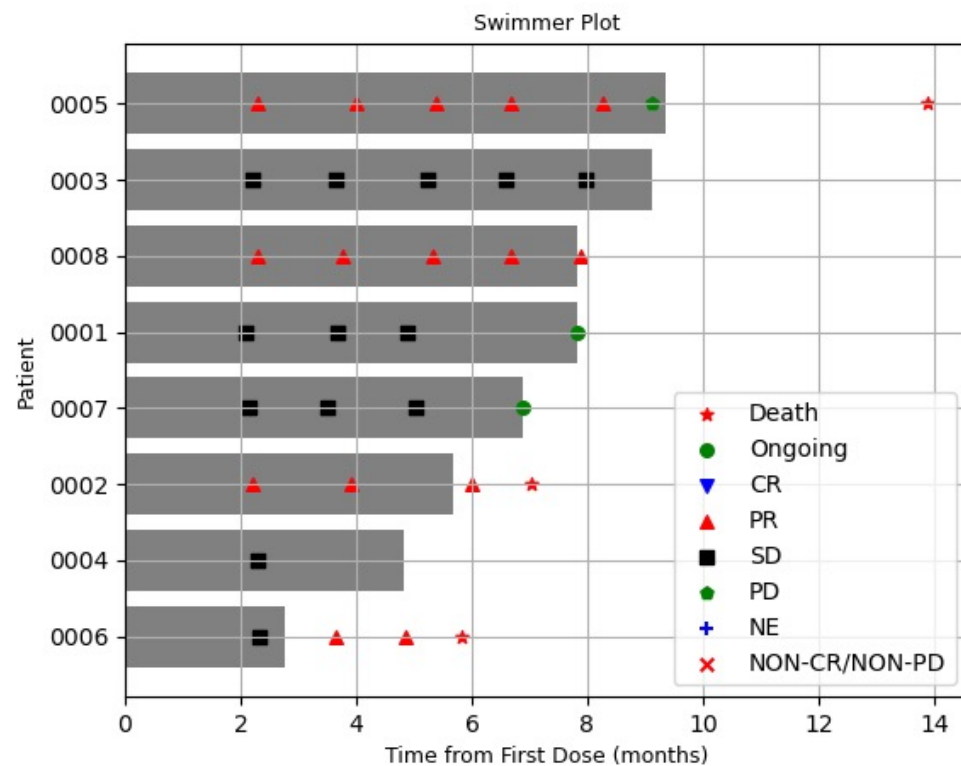
```
df2=df2.assign(COL='red')
```

```
df2.loc[df2['TRTP']==b'TRT A','COL']='blue'
```

```
df_sorted=df2.sort_values('AVAL', ascending=False)
```

```
plt.bar('SUBJID', 'AVAL', data=df_sorted, color='COL')
```

Swimmer Plot



Swimmer Plot Code

```
fig, ax=plt.subplots()  
plt.subplots_adjust( top=0.9, bottom=0.1)  
df.sort_values('TRTDURM', inplace=True)  
ax.barh('SUBJID', 'TRTDURM', color="gray", zorder=-1 , data=df)
```

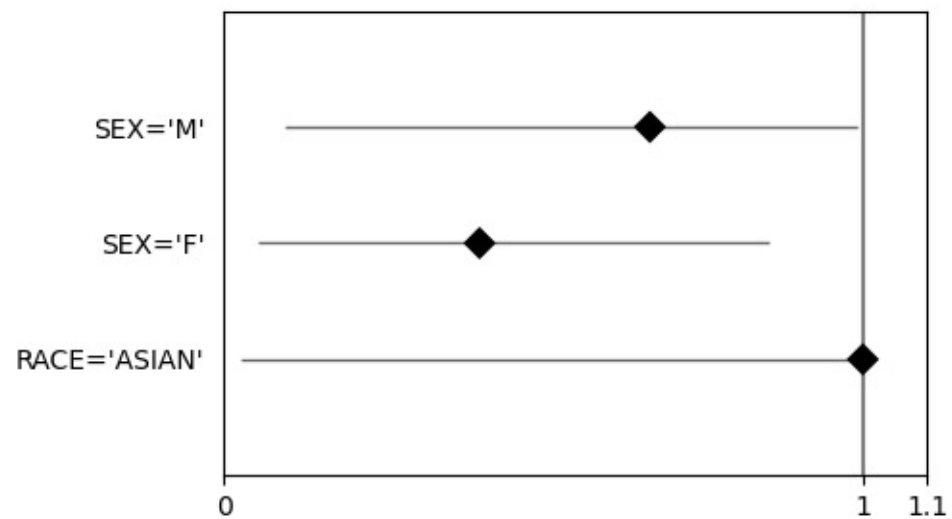
Swimmer Plot Code, Continued

```
ax.scatter('DTHDM','SUBJID', data=df, marker='*',color='red', s=30,  
zorder=1, label='Death')
```

```
ax.scatter('TRTDURM','SUBJID', data=ongo, marker='o',color='green',  
s=30, zorder=1, label='Ongoing')
```

```
ax.scatter('ADM','SUBJID', data=cr, marker='v',color='blue', s=30,  
zorder=1, label='CR')
```

Forest Plot



Forest plot for ORR

ORR	95% CI
0.667	(0.094, 0.992)
0.4	(0.053, 0.853)
1.0	(0.025, 1.0)

Forest Plot Code

```
q_, ci_low, ci_upp=confint(df['SEX']==b'M')  
q2_, ci_low2, ci_upp2=confint(df['SEX']==b'F')  
q3_, ci_low3, ci_upp3=confint(df['RACE']==b'ASIAN')
```

```
labs = ["SEX='M'", "SEX='F'", "RACE='ASIAN'"]  
measure = [q_, q2_, q3_]   
lower = [ci_low, ci_low2, ci_low3]  
upper = [ci_upp, ci_upp2, ci_upp3]
```

Forest Plot Code

```
p = EffectMeasurePlot(label=labs, effect_measure=measure,  
lcl=lower, ucl=upper)
```

```
p.labels(effectmeasure='ORR')
```

```
p.colors(pointshape="D")
```

```
ax=p.plot(figsize=(7,3), t_adjuster=0.02, max_value=1.1,  
min_value=0 )
```

About the Author

- ▶ Hengwei Liu is associate director, statistical programming at Daiichi Sankyo Inc. in Basking Ridge, New Jersey.

He has been working in the pharmaceutical industry for 18 years.

- ▶ Contact: hengwei_liu@yahoo.com