- Shin’s Lab -

# Python for Data Visualization

# Python for Data Visualization

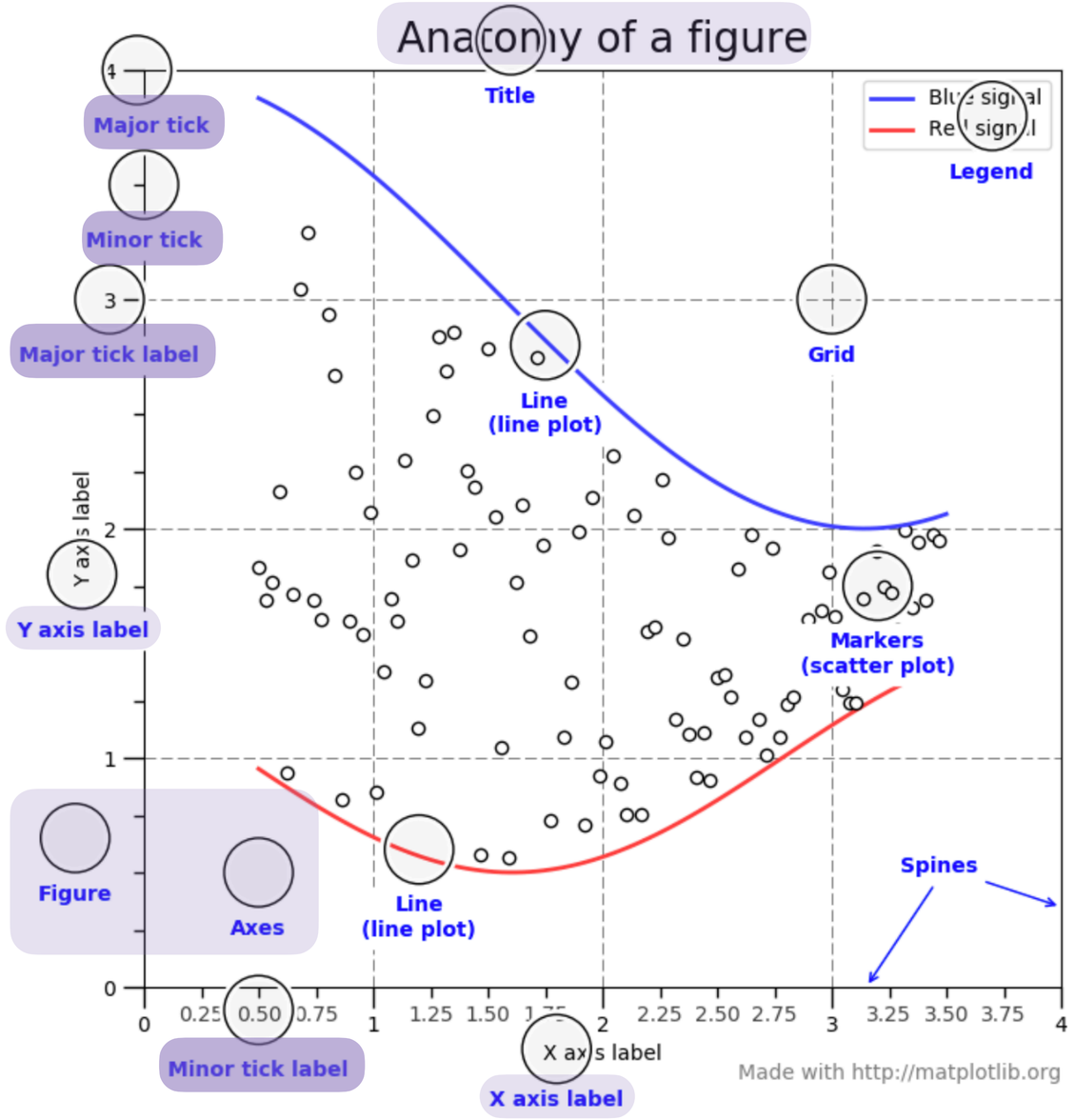
## -Chapter.1 Matplotlib Anatomy -

# Python for Data Visualization

## -Chapter.1 Matplotlib Anatomy -

### 1-05. Grid

1. ax.grid
2. Ticks and Grid
3. Grid Exercise


Anatomy of a figure
Title
Major tick
Minor tick
Major tick label
Line
(line plot)
Grid
Legend
Markers
(scatter plot)
Y axis label
Figure
Axes
Line
(line plot)
Spines
Minor tick label
X axis label
Made with http://matplotlib.org

# 1. ax.grid

## matplotlib.pyplot.grid

`matplotlib.pyplot.grid(b=None, which='major', axis='both', **kwargs)` [source]

Configure the grid lines.

**Parameters:**

**b** : bool or None, optional

Whether to show the grid lines. If any *kwargs* are supplied, it is assumed you want the grid on and *b* will be set to True.

If *b* is *None* and there are no *kwargs*, this toggles the visibility of the lines.

**which** : {'major', 'minor', 'both'}, optional

The grid lines to apply the changes on.

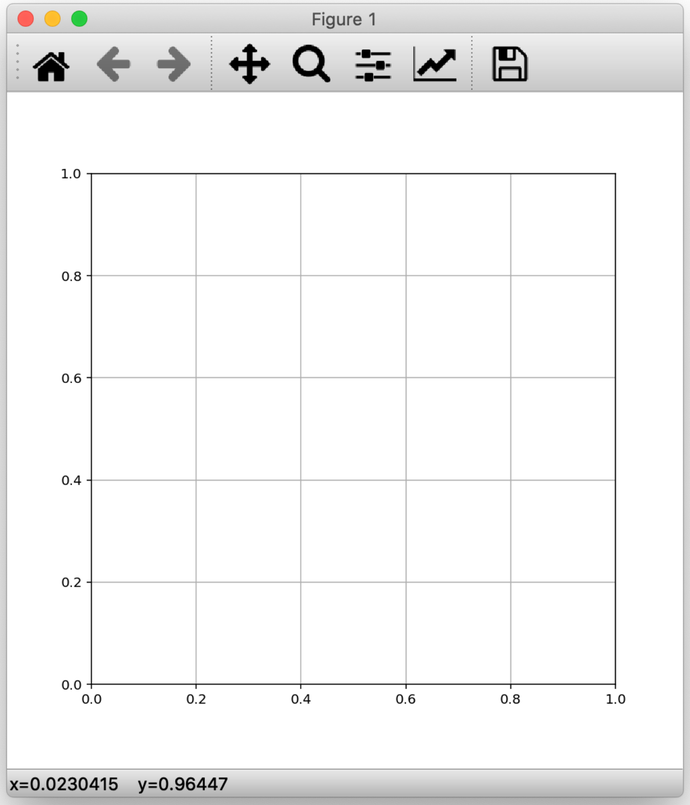
**axis** : {'both', 'x', 'y'}, optional

The axis to apply the changes on.

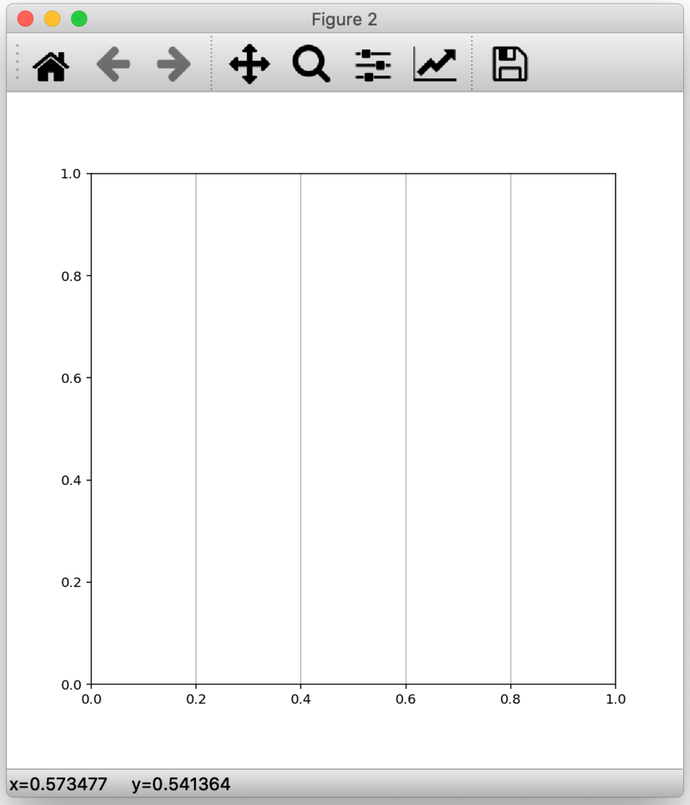
# 1. ax.grid(Axis Grid)

```
fig, ax = plt.subplots(figsize=(7, 7))
```

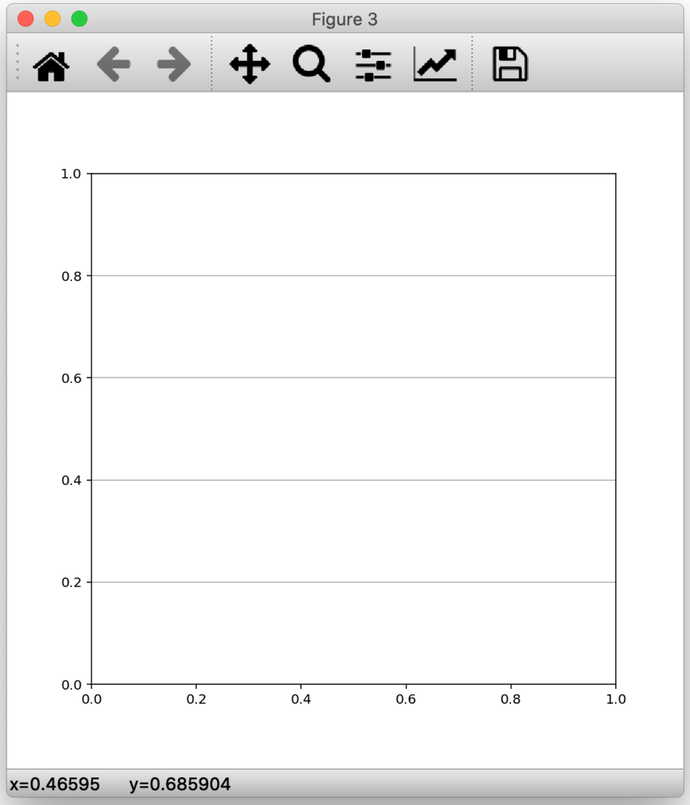
```
ax.grid()
```




```
ax.grid(axis='x')
```



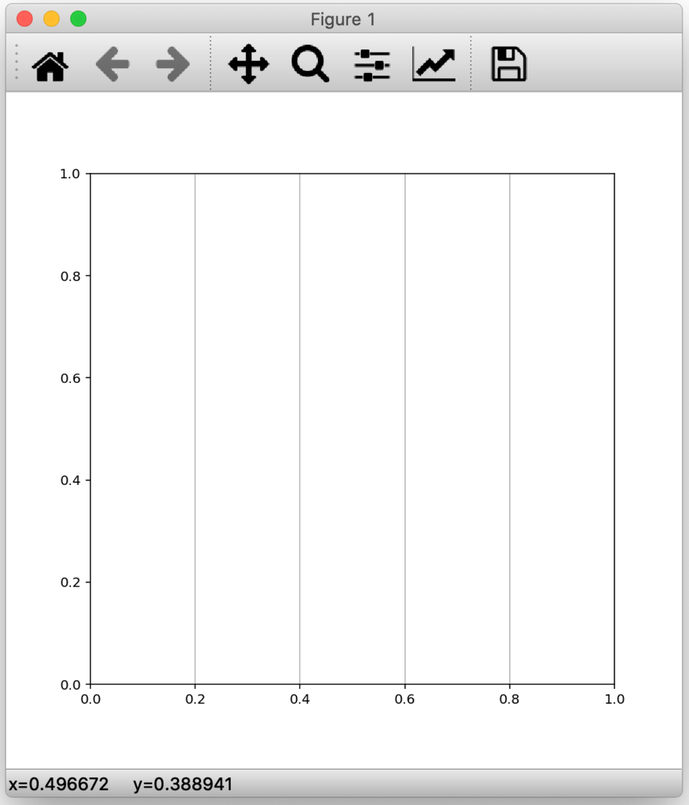

```
ax.grid(axis='y')
```



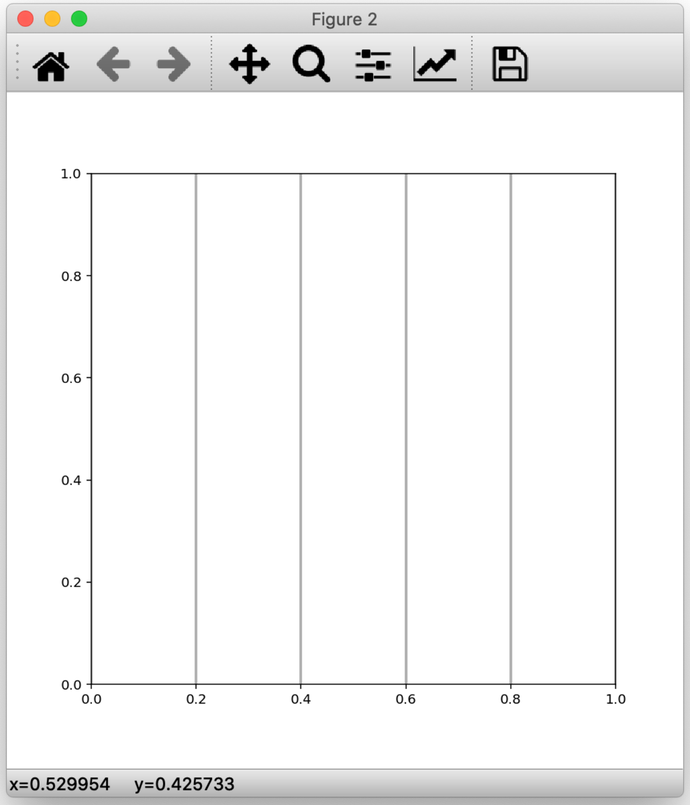
# 1. ax.grid(linewidth Argument)

```
fig, ax = plt.subplots(figsize=(7, 7))
```

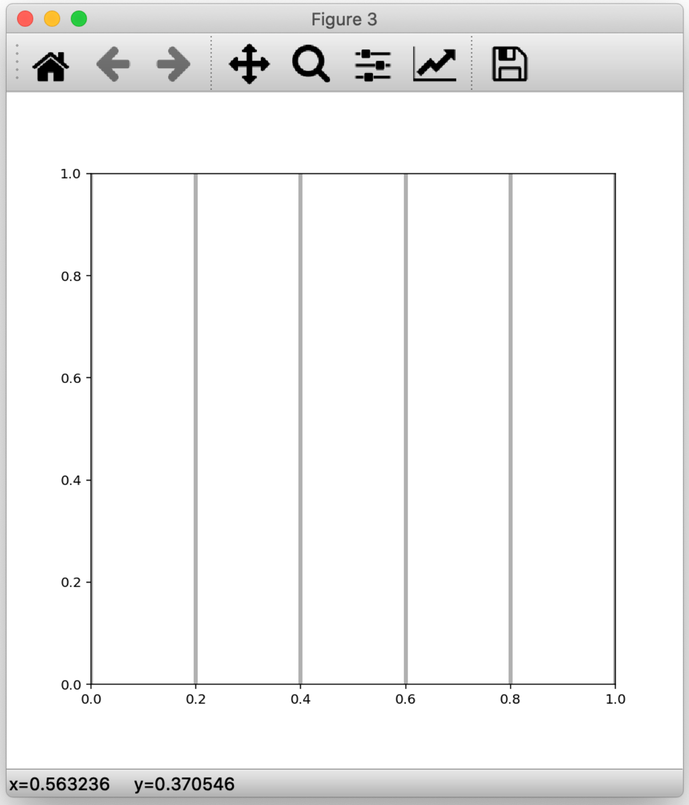
```
ax.grid(axis='x')
```



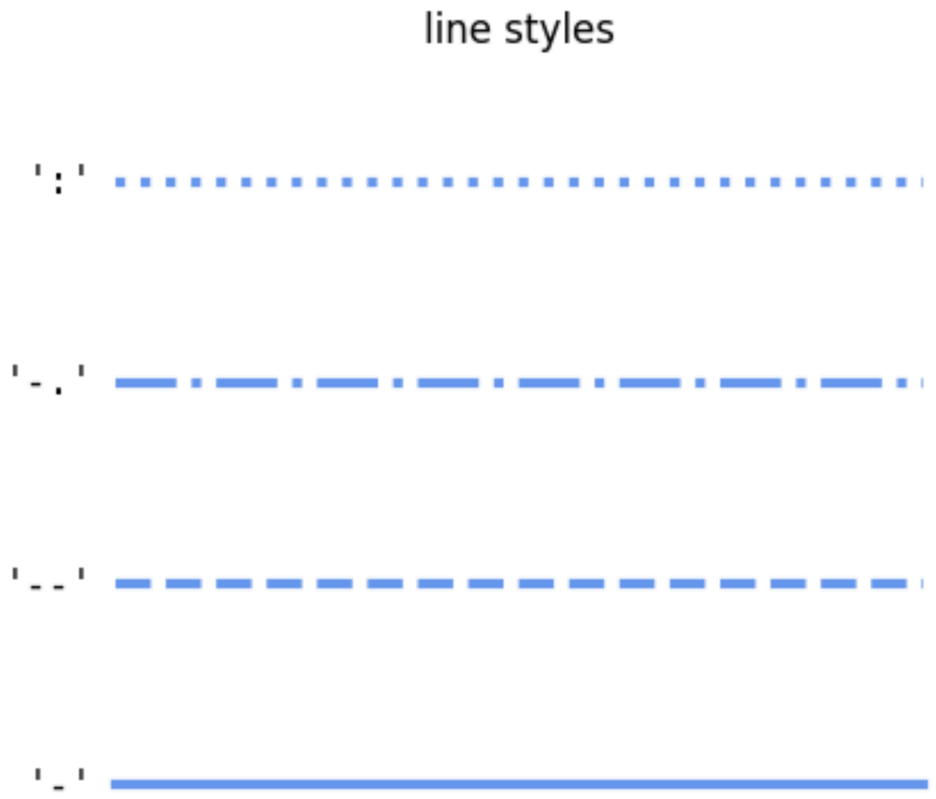

```
ax.grid(axis='x',
        linewidth=2)
```




```
ax.grid(axis='x',
        linewidth=3)
```

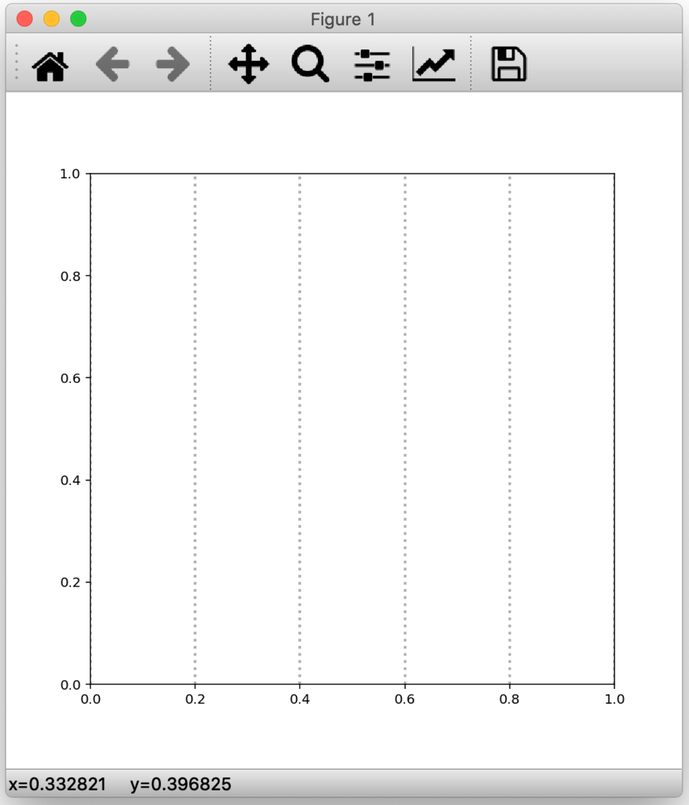


# 1. ax.grid(Line Styles in Matplotlib)

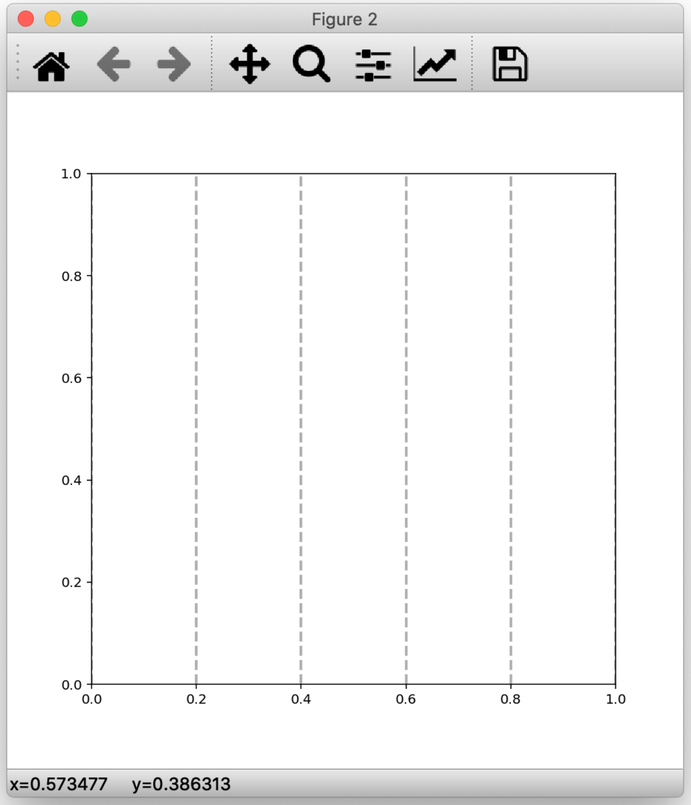


# 1. ax.grid(Line Styles in Matplotlib)

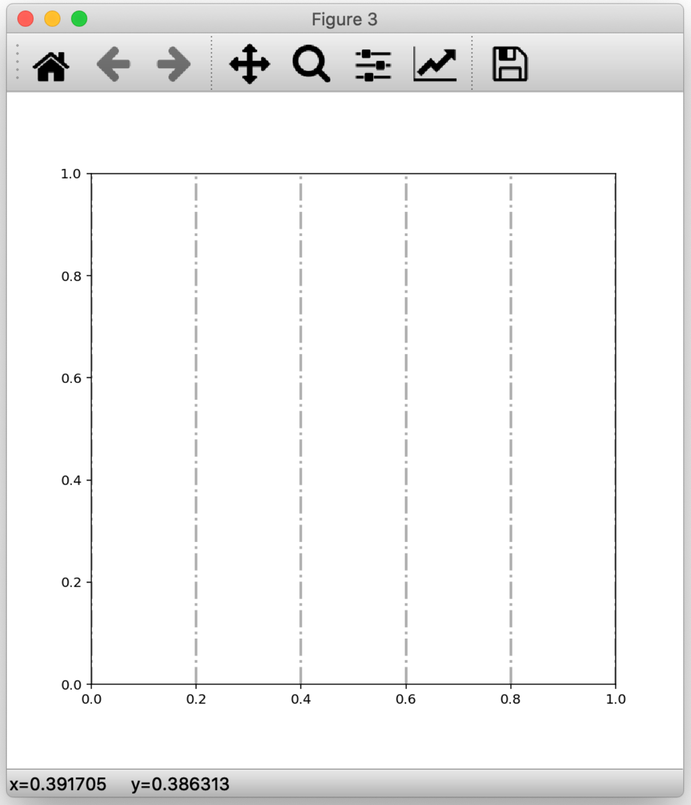
```
ax.grid(axis='x',
        linewidth=2,
        linestyle=':')
```




```
ax.grid(axis='x',
        linewidth=2,
        linestyle='--')
```

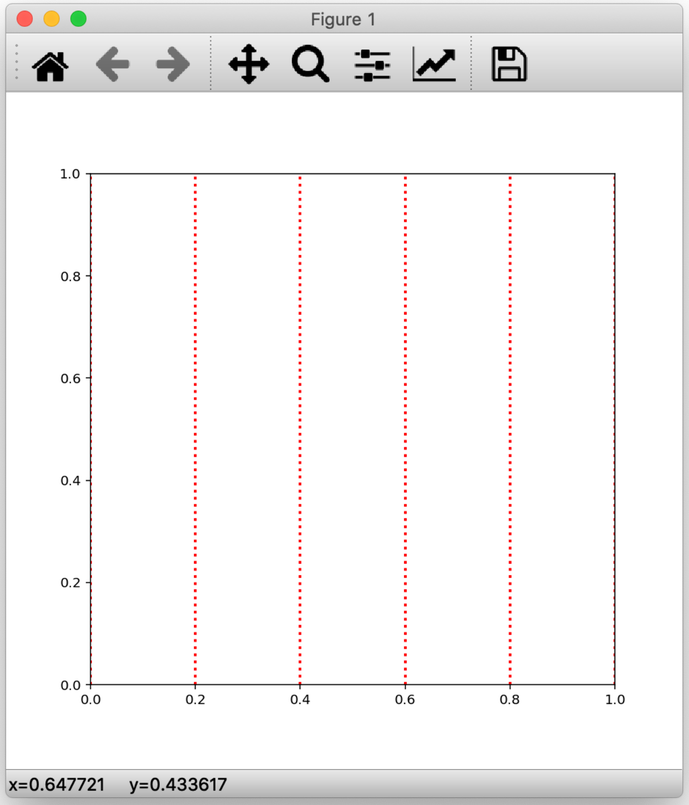



```
ax.grid(axis='x',
        linewidth=2,
        linestyle='-.')
```

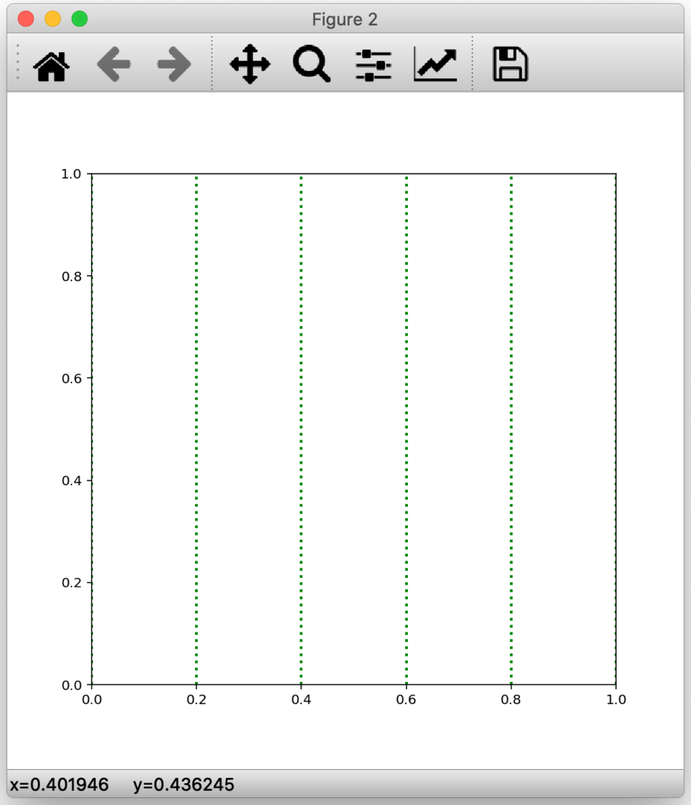


# 1. ax.grid(Grid Color)

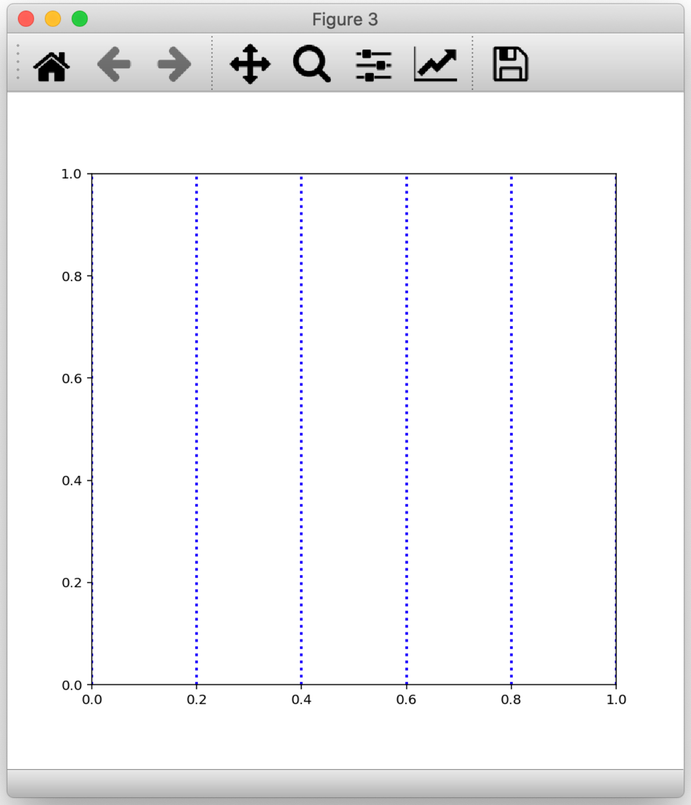
```
ax.grid(axis='x',
        linewidth=2,
        linestyle=':',
        color='r')
```




```
ax.grid(axis='x',
        linewidth=2,
        linestyle=':',
        color='g')
```



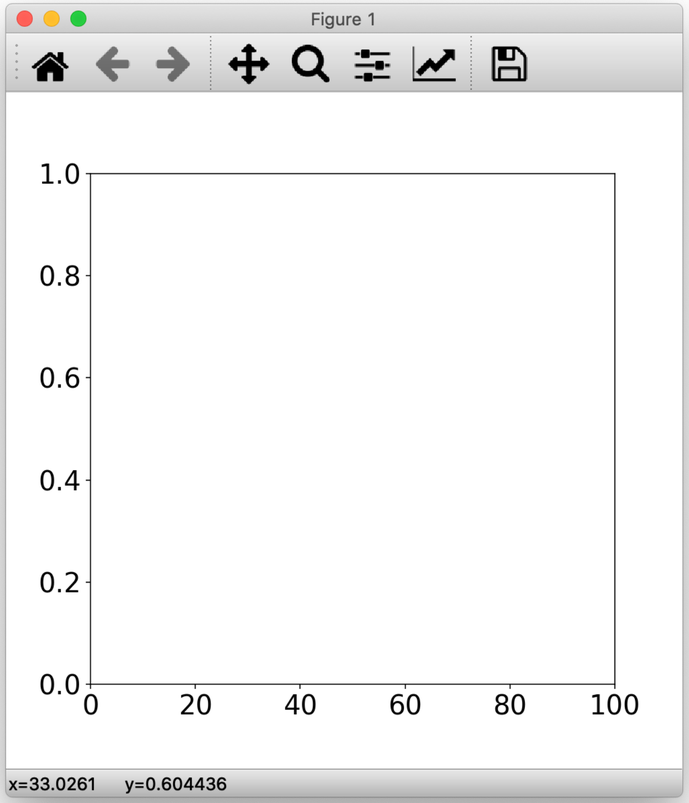

```
ax.grid(axis='x',
        linewidth=2,
        linestyle=':',
        color='b')
```

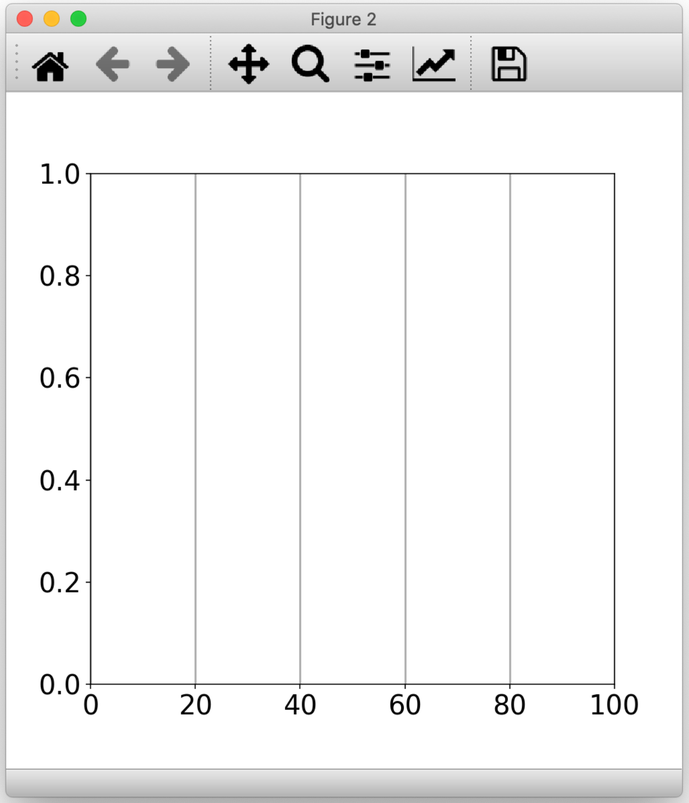


# 2. Ticks and Grid

```
major_xticks = [i for i in range(0, 101, 20)]

ax.set_xticks(major_xticks)
ax.tick_params(labelsize=20)
```

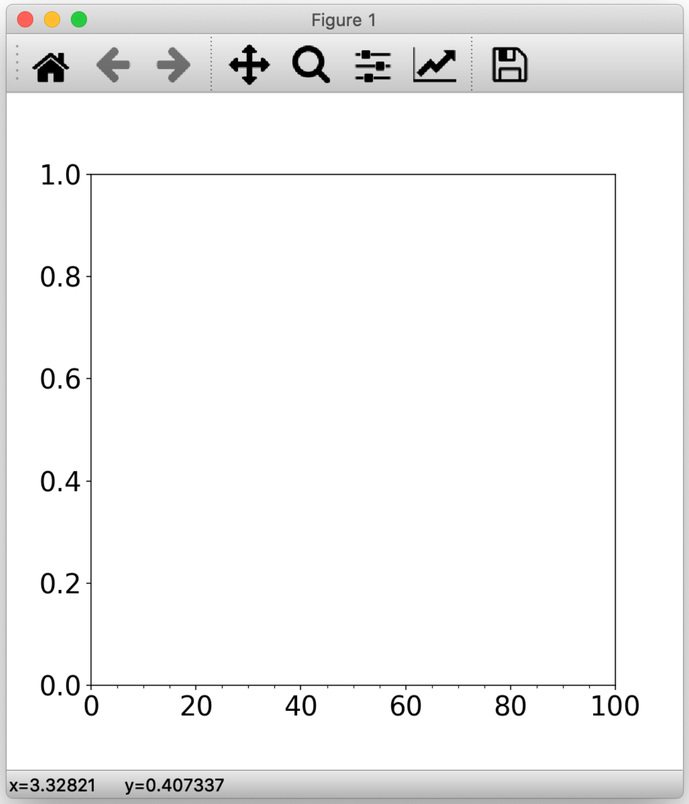
```
ax.set_xticks(major_xticks)
ax.tick_params(labelsize=20)

ax.grid(axis='x', which='major',
        linewidth=1.5)
```

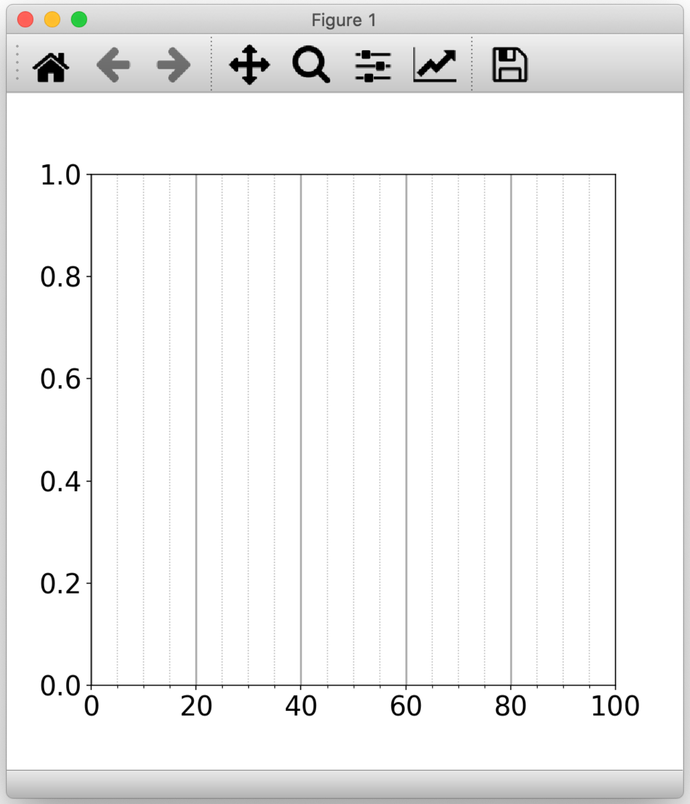


# 2. Ticks and Grid(Major and Minor Grid)

```
fig, ax = plt.subplots(figsize=(7, 7))

major_xticks = [i for i in range(0, 101, 20)]
minor_xticks = [i for i in range(0, 101, 5)]

ax.set_xticks(major_xticks)
ax.set_xticks(minor_xticks, minor=True)

ax.tick_params(labelsize=20)

ax.grid(axis='x', which='major',
        linewidth=1.5)
ax.grid(axis='x', which='minor',
        linestyle=':')
```

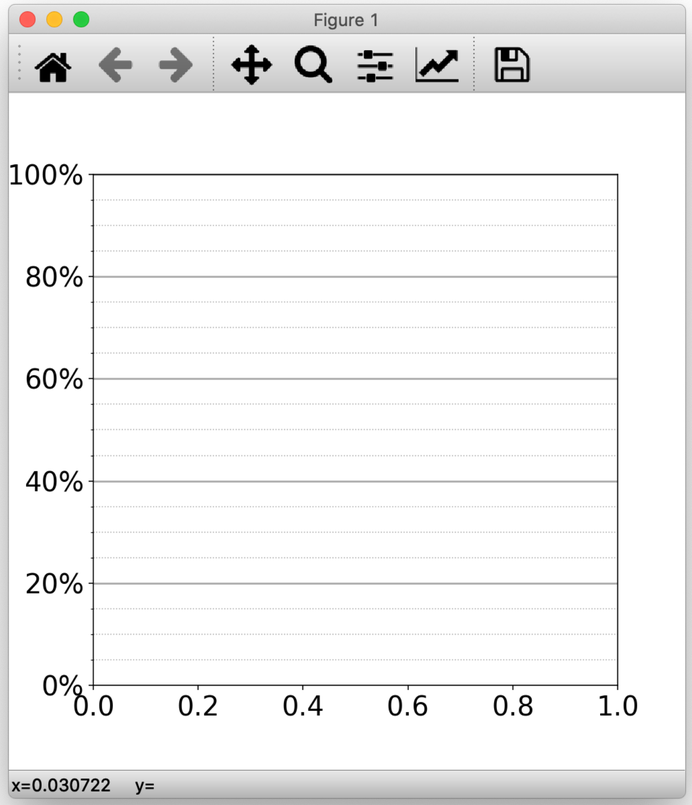



after ax.grid()

# 2. Ticks and Grid(x, y Grid)

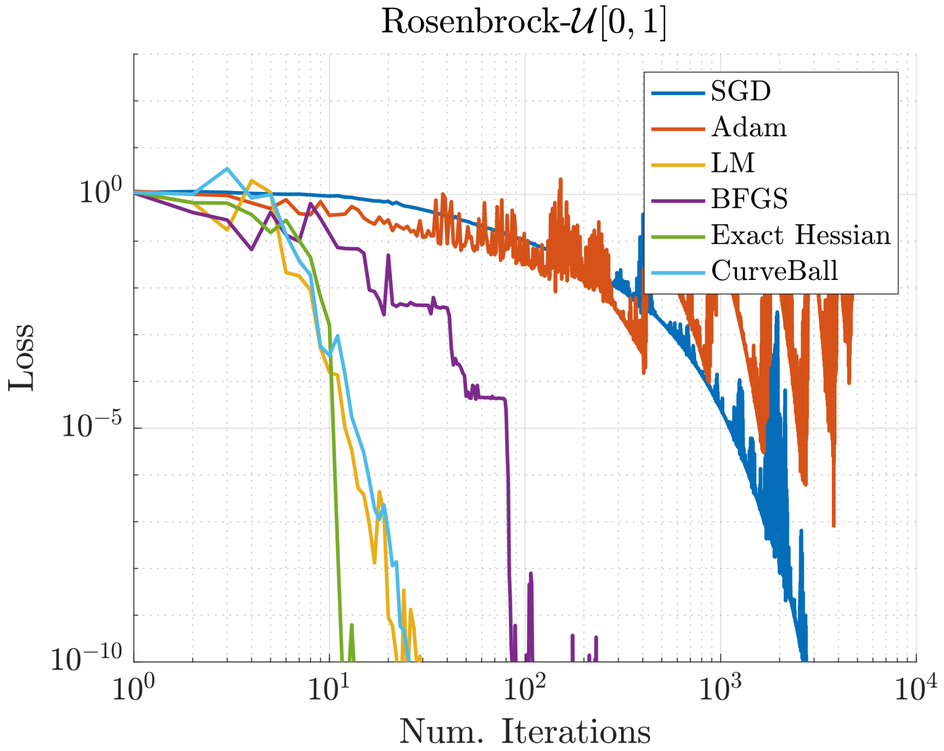
```
fig, ax = plt.subplots(figsize=(7, 7))

major_yticks = [i for i in range(0, 101, 20)]
minor_yticks = [i for i in range(0, 101, 5)]
ytick_labels = [str(i)+'%' for i in major_yticks]

ax.set_yticks(major_xticks)
ax.set_yticks(minor_xticks, minor=True)
ax.set_yticklabels(ytick_labels)

ax.tick_params(labelsize=20)

ax.grid(axis='y', which='major',
        linewidth=1.5)
ax.grid(axis='y', which='minor',
        linestyle=':')
```



# 3. Grid Exercise

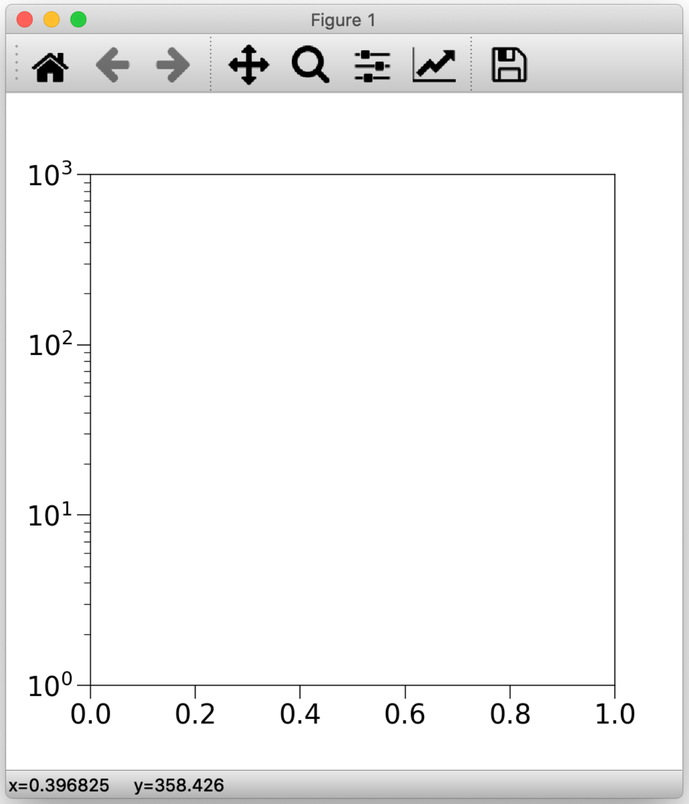
**Small Steps and Giant Leaps: Minimal Newton Solvers for Deep Learning**

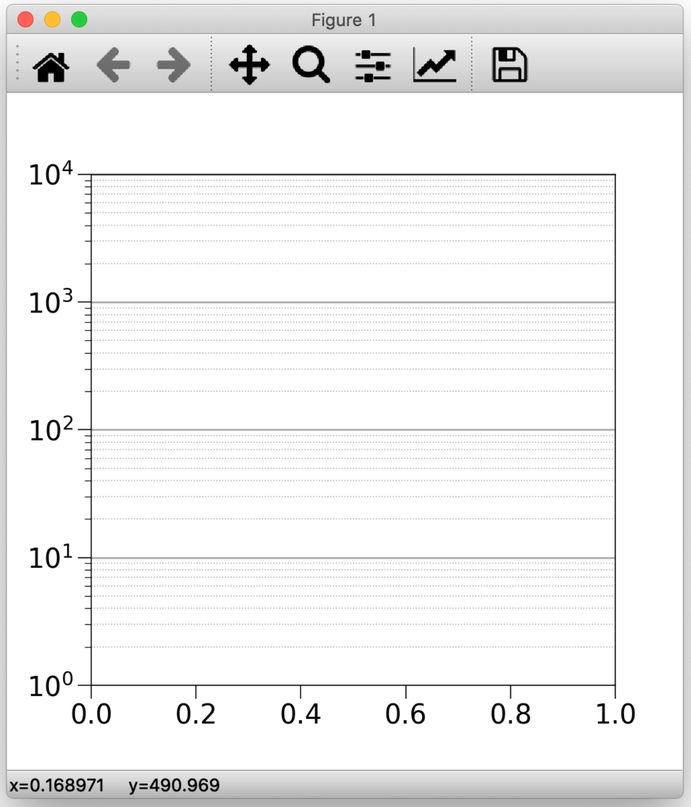
João F. Henriques  Sebastien Ehrhardt  Samuel Albanie  Andrea Vedaldi
Visual Geometry Group, University of Oxford
{joao,hyenal,albanie,vedaldi}@robots.ox.ac.uk



# 3. Grid Exercise

```
fig, ax = plt.subplots(figsize=(7, 7))

ax.set_yscale('log')
ax.set_ylim([1, 1E3])

ax.tick_params(labelsize=20,
               length=10)
ax.tick_params(which='minor',
               length=5)

ax.grid(axis='y', which='major',
        linewidth=1.5)
ax.grid(axis='y', which='minor',
        linestyle=':')
```



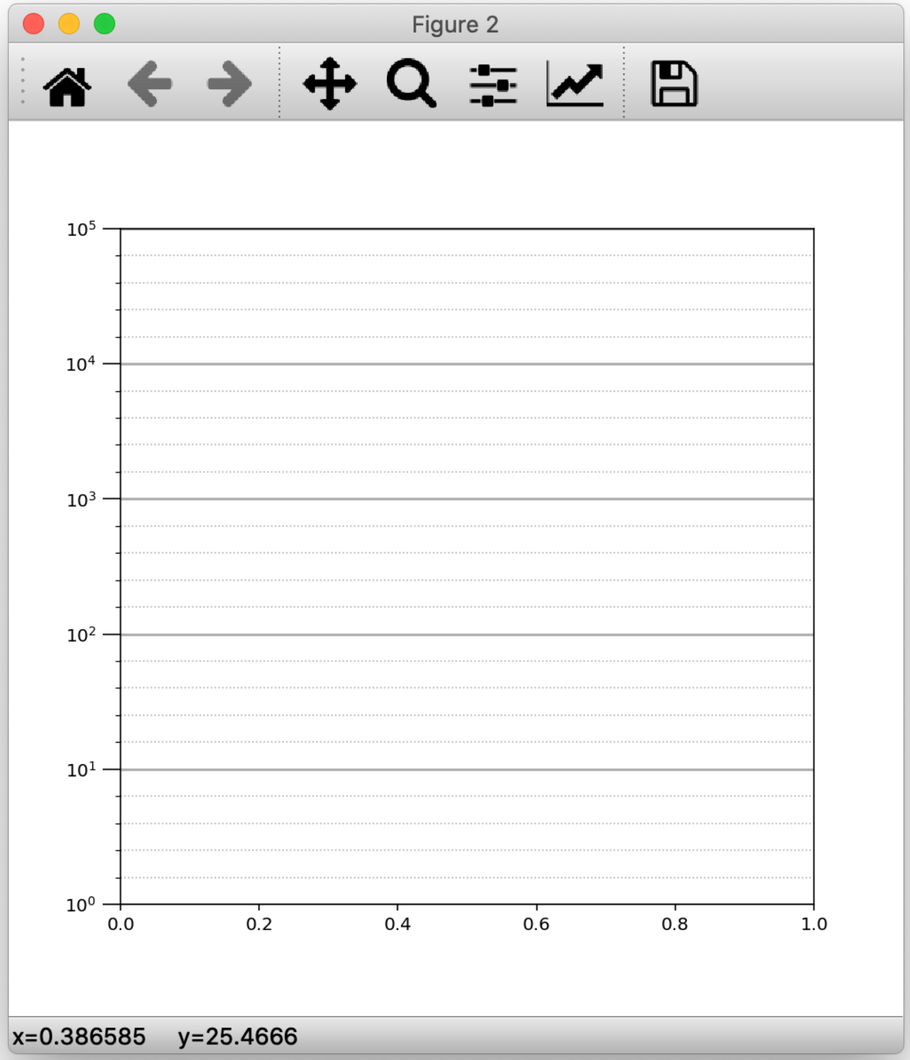




after ax.grid()

# 3. Grid Exercise

```
m_exp, M_exp = 0, 5
n_inter_yticks = 4

n_major_yticks = M_exp - m_exp + 1
n_minor_yticks = (n_major_yticks-1)*(n_inter_yticks+1) + 1

major_yticks = np.logspace(m_exp, M_exp, n_major_yticks)
minor_yticks = np.logspace(m_exp, M_exp, n_minor_yticks)


fig, ax = plt.subplots(figsize=(7, 7))
ax.set_yscale('log')
ax.set_ylim([10**m_exp, 10**M_exp])

ax.set_yticks(major_yticks)
ax.set_yticks(minor_yticks,
              minor=True)

ax.tick_params(axis='y',
               which='major',
               length=10)
ax.tick_params(axis='y',
               which='minor',
               length=3)
```

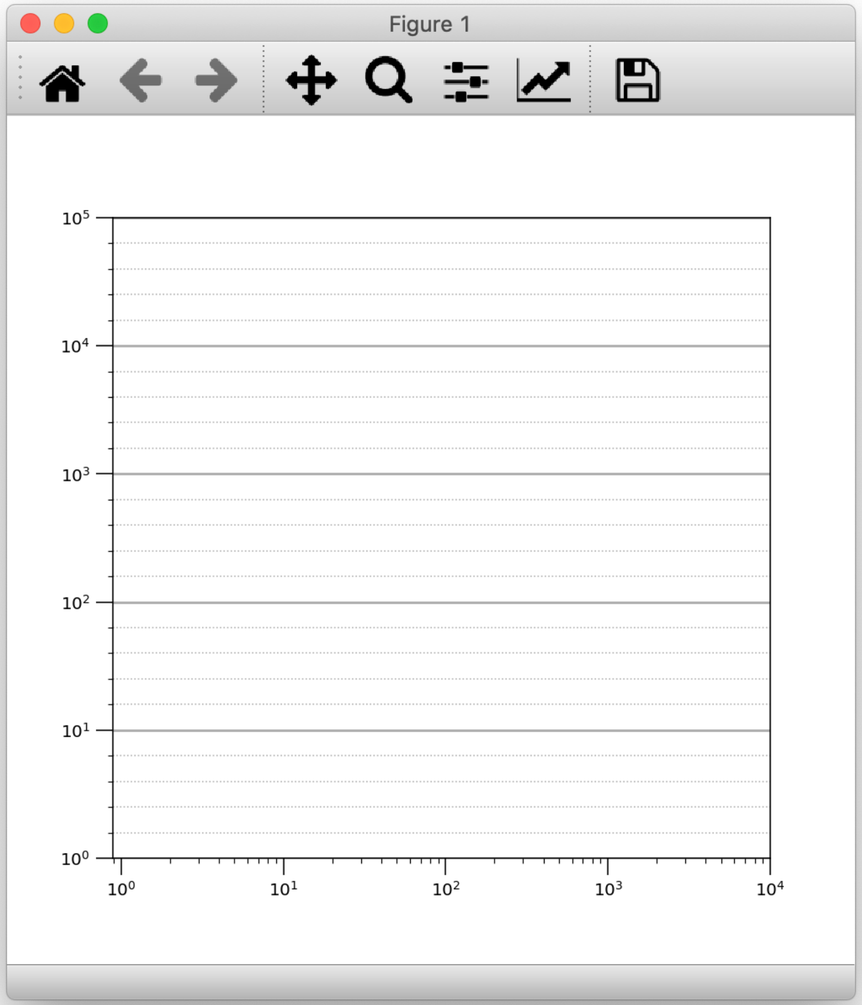
```
ax.grid(axis='y',
        which='major',
        linewidth=1.5)
ax.grid(axis='y',
        which='minor',
        linestyle=':')
```



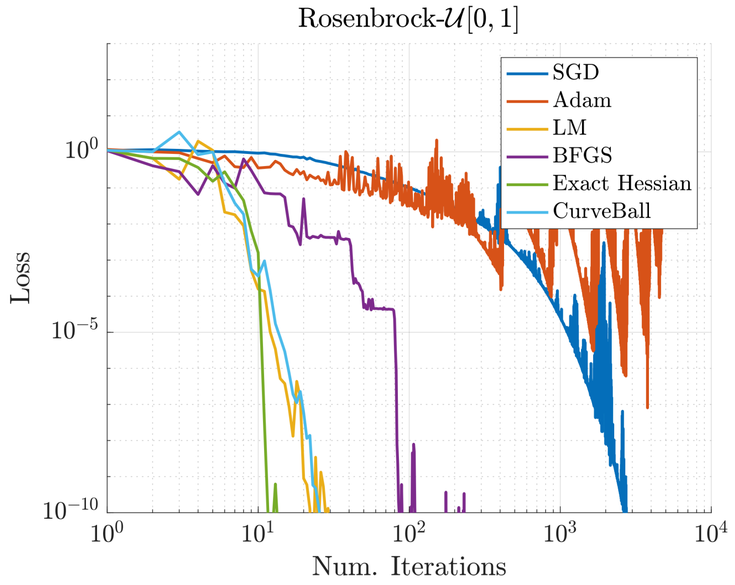
# 3. Grid Exercise

```
m_exp, M_exp = 0, 5
n_inter_yticks = 4

n_major_yticks = M_exp - m_exp + 1
n_minor_yticks = (n_major_yticks-1)*(n_inter_yticks+1) + 1

major_yticks = np.logspace(m_exp, M_exp, n_major_yticks)
minor_yticks = np.logspace(m_exp, M_exp, n_minor_yticks)
major_xticks = np.logspace(0, 4, 5)


fig, ax = plt.subplots(figsize=(7, 7))
ax.set_yscale('log')
ax.set_xscale('log')
ax.set_ylim([10**m_exp, 10**M_exp])

ax.set_yticks(major_yticks)
ax.set_yticks(minor_yticks,
              minor=True)
ax.set_xticks(major_xticks)
```

```
ax.tick_params(axis='y',
               which='major',
               length=10)
ax.tick_params(axis='y',
               which='minor',
               length=3)

ax.grid(axis='y',
        which='major',
        linewidth=1.5)
ax.grid(axis='y',
        which='minor',
        linestyle=':')
```

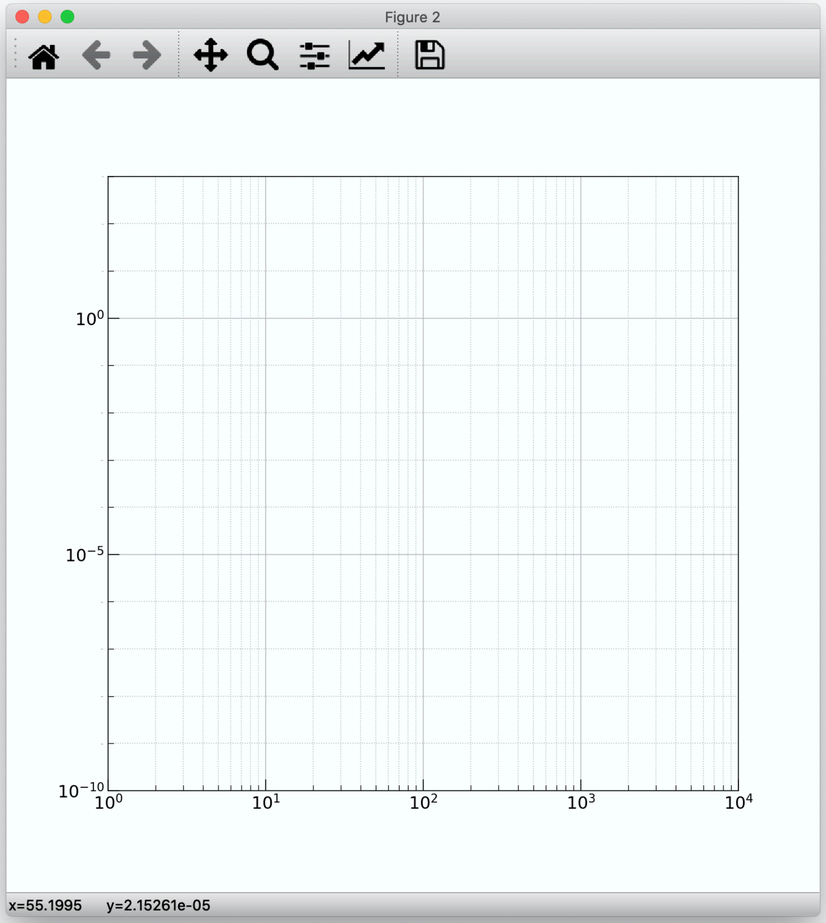


# 3. Grid Exercise

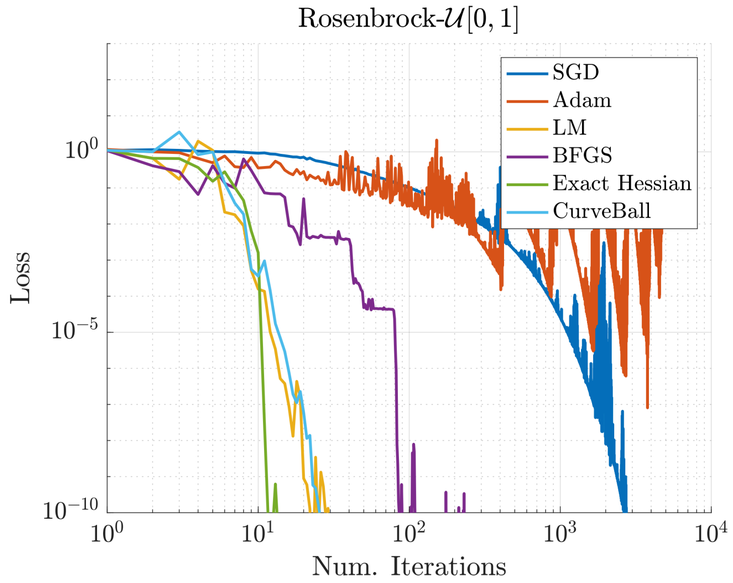



```
fig, ax = plt.subplots(figsize=(10, 10))
ax.set_yscale('log')
ax.set_xscale('log')

ax.set_ylim([10**-10, 10**3])
ax.set_xlim([10**0, 10**4])

major_yticks = [10**i for i in [-10, -5, 0]]
minor_yticks = [10**i for i in range(-10, 4)]
major_xticks = [10**i for i in range(0, 5)]
```

```
ax.set_yticks(major_yticks)
ax.set_yticks(minor_yticks,
              minor=True)
ax.set_xticks(major_xticks)

ax.tick_params(which='major',
               direction='in',
               length=10,
               labelsize=15)
ax.tick_params(which='minor',
               direction='in',
               length=5,
               labelsize=0)

ax.grid(which='major',
        color='silver')
ax.grid(which='minor',
        linestyle=':',
        color='silver')
```

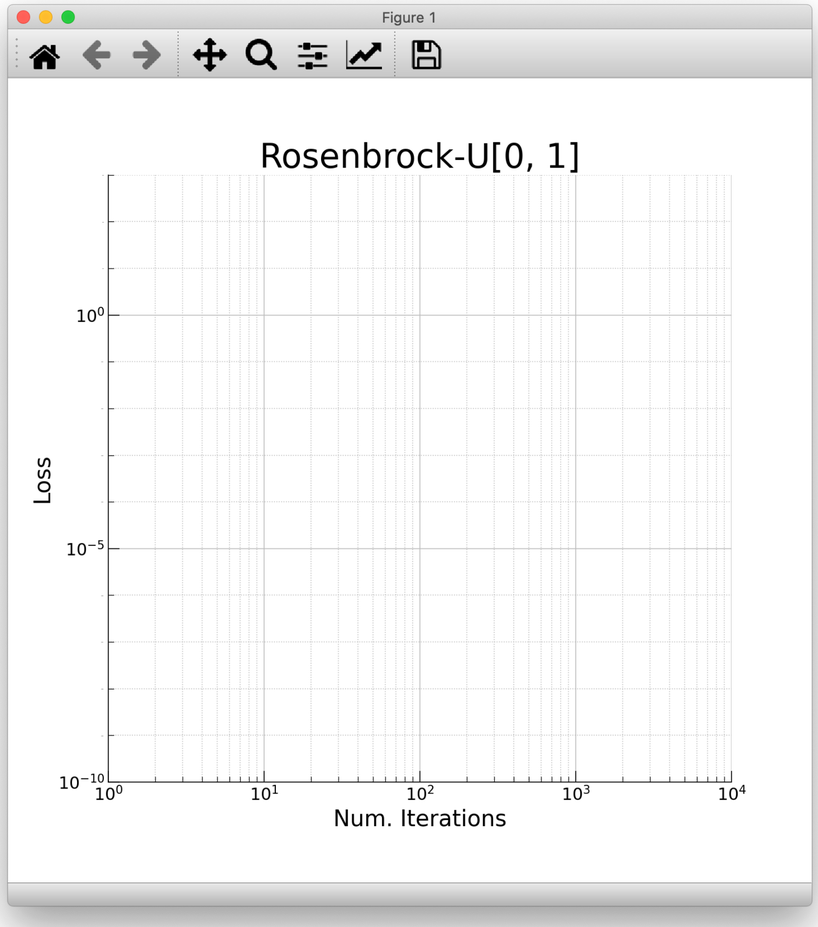


# 3. Grid Exercise




```
fig, ax = plt.subplots(figsize=(10, 10))
ax.set_yscale('log')
ax.set_xscale('log')

ax.set_ylim([10**-10, 10**3])
ax.set_xlim([10**0, 10**4])

major_yticks = [10**i for i in [-10, -5, 0]]
minor_yticks = [10**i for i in range(-10, 4)]
major_xticks = [10**i for i in range(0, 5)]

ax.set_yticks(major_yticks)
ax.set_yticks(minor_yticks,
              minor=True)
ax.set_xticks(major_xticks)
```

```
ax.tick_params(which='major',
               direction='in',
               length=10,
               labelsize=15)
ax.tick_params(which='minor',
               direction='in',
               length=5,
               labelsize=0)
ax.grid(which='major',
        color='silver')
ax.grid(which='minor',
        linestyle=':',
        color='silver')


ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

ax.set_title("Rosenbrock-U[0, 1]",
             fontsize=30)
ax.set_xlabel("Num. Iterations",
              fontsize=20)
ax.set_ylabel("Loss",
              fontsize=20)
```



# Python for Data Visualization

## -Chapter.1 Matplotlib Anatomy -

### 1-05. Grid

1. ax.grid
2. Ticks and Grid
3. Grid Exercise