exercise3 (Score: 12.0 / 12.0)

1. Task (Score: 12.0 / 12.0)

# Lab 5

1. 提交作業之前，建議可以先點選上方工具列的**Kernel**，再選擇**Restart & Run All**，檢查一下是否程式跑起來都沒有問題，最後記得儲存。
2. 請先填上下方的姓名(name)及學號(stduent_id)再開始作答，例如：

```
name = "我的名字"
student_id= "B06201000"
```

3. 演算法的實作可以參考lab-5 (https://yuanyuyuan.github.io/itcm/lab-5.html), 有任何問題歡迎找助教詢問。
4. **Deadline: 12/11(Wed.)**

In [1]:

```
name = "李澤諺"
student_id = "B05902023"
```

(Top)

# Exercise 3

### Analyse the convergence properties of the Jacobi and Gauss-Seidel methods for the solution of a linear system whose matrix is

### $$\left[\begin{matrix}

```
\alpha &&0 &&1\\
0 &&\alpha &&0\\
1 &&0 &&\alpha
\end{matrix}\right],
\quad \quad
\alpha \in \mathbb{R}.$$
```

令該矩陣為 $A$。
當 $|\alpha| > 1$ 時，$A$ 為strictly diagonally dominant，此時Jacobi method和Gauss–Seidel method皆會converge，且理論上Gauss-Seidel method的速度應不會慢於Jacobi method。
而當 $|\alpha| \leq 1$ 時，利用以下的程式來觀察Jacobi method和Gauss–Seidel method的convergence property。

### Import Library

In [2]:

```
import numpy as np
import matplotlib.pyplot as plt
```

### Implemetation of Jacobi Method and Gauss-Seidel Method

In [3]:

```
def Jacobi_method(A , b , iteration = 10):
    '''
    Return :
        estimation : list, estimation of solution to Ax = b at each iteration.
    '''
    n = A.shape[0]
    x = np.zeros(n)
    estimation = [x]
    for i in range(iteration):
        new_x = np.zeros(n)
        for j in range(n):
            s = 0
            for k in range(n):
                if (j != k):
                    s += A[j][k] * x[k]
            new_x[j] = (b[j] - s) / A[j][j]
        estimation.append(new_x)
        x = new_x
    return estimation
```

In [4]:

```
def Gauss_Seidel_method(A , b , iteration = 10):
    '''
    Return :
        estimation : list, estimation of solution to Ax = b at each iteration.
    '''
    n = A.shape[0]
    x = np.zeros(n)
    estimation = [x]
    for i in range(iteration):
        for j in range(n):
            s = 0
            for k in range(n):
                if (j != k):
                    s += A[j][k] * x[k]
            x[j] = (b[j] - s) / A[j][j]
        estimation.append(x)
    return estimation
```

### Prepare $\alpha$ ( $-1 \leq \alpha \leq 1$)

In [5]:

```
alpha = [(i + 1) / 10 for i in range(10)] + [(- i - 1) / 10 for i in range(10)]
alpha.sort(reverse = True)
print('alpha =' , alpha)
```

```
alpha = [1.0, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, -0.1, -0.2, -0.3, -0.4, -0.5, -0.6
, -0.7, -0.8, -0.9, -1.0]
```

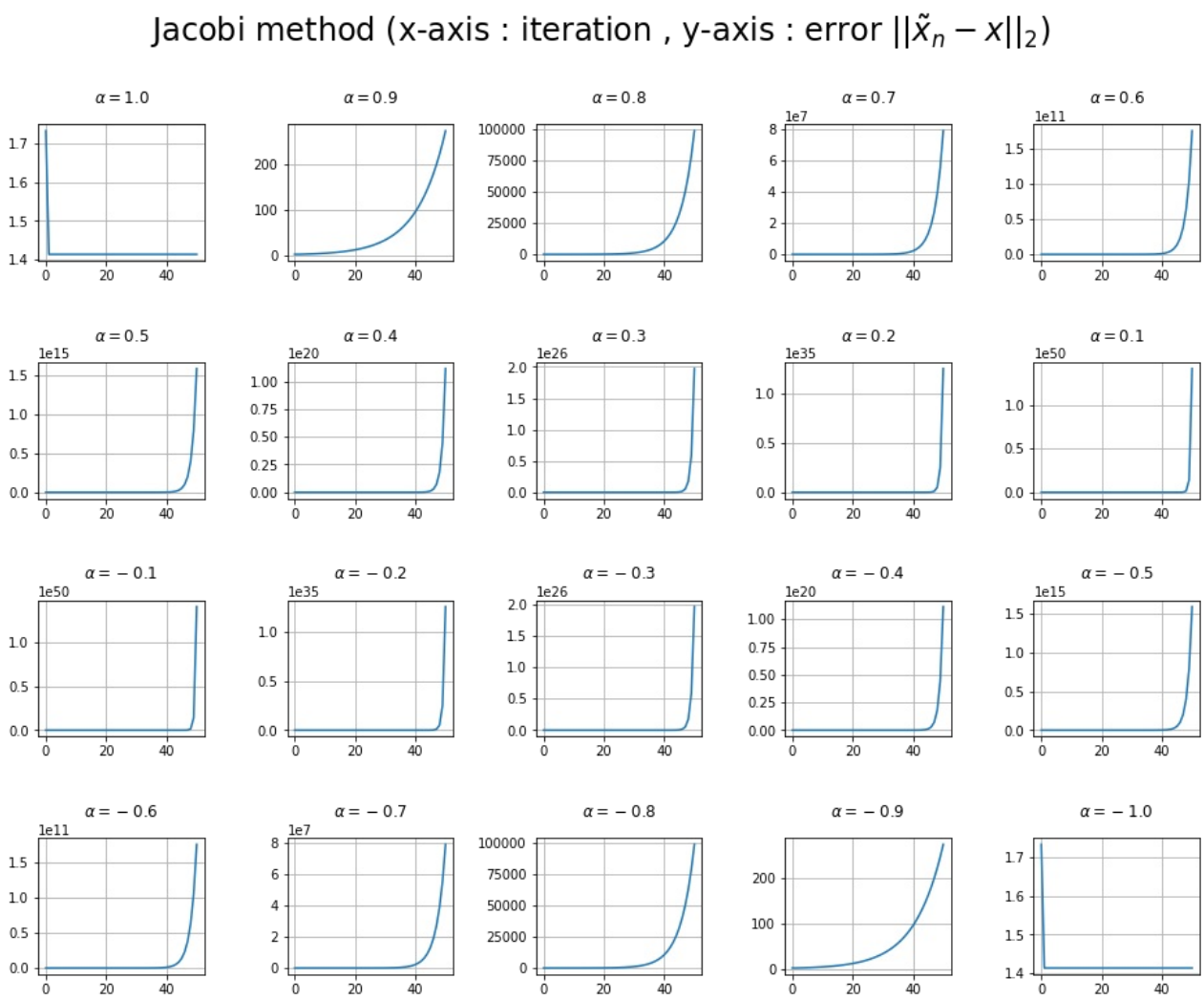
### Convergence Property of Jacobi Method

In [6]:

```
(fig , axes) = plt.subplots(4 , 5 , figsize = (16 , 12))
plt.subplots_adjust(wspace = 0.5 , hspace = 0.75)
for (i , ax) in enumerate(axes.flatten()):
    A = np.array([[alpha[i] , 0 , 1] , [0 , alpha[i] , 0] , [1 , 0 , alpha[i]]])
    x = np.ones(3)
    b = np.dot(A , x)
    estimation = Jacobi_method(A , b , iteration = 50)
    error = [np.linalg.norm(estimation[j] - x , 2) for j in range(len(estimation))]

    x_range = np.arange(-np.pi , np.pi , 0.01)
    ax.set_title(r'$\alpha = {}$'.format(alpha[i]) , y = 1.1)
    ax.plot(error)
    ax.grid(True)

plt.suptitle(r'Jacobi method (x-axis : iteration , y-axis : error $||\tilde{x}_n - x||_2)' , fontsize = 24)
plt.show()
```



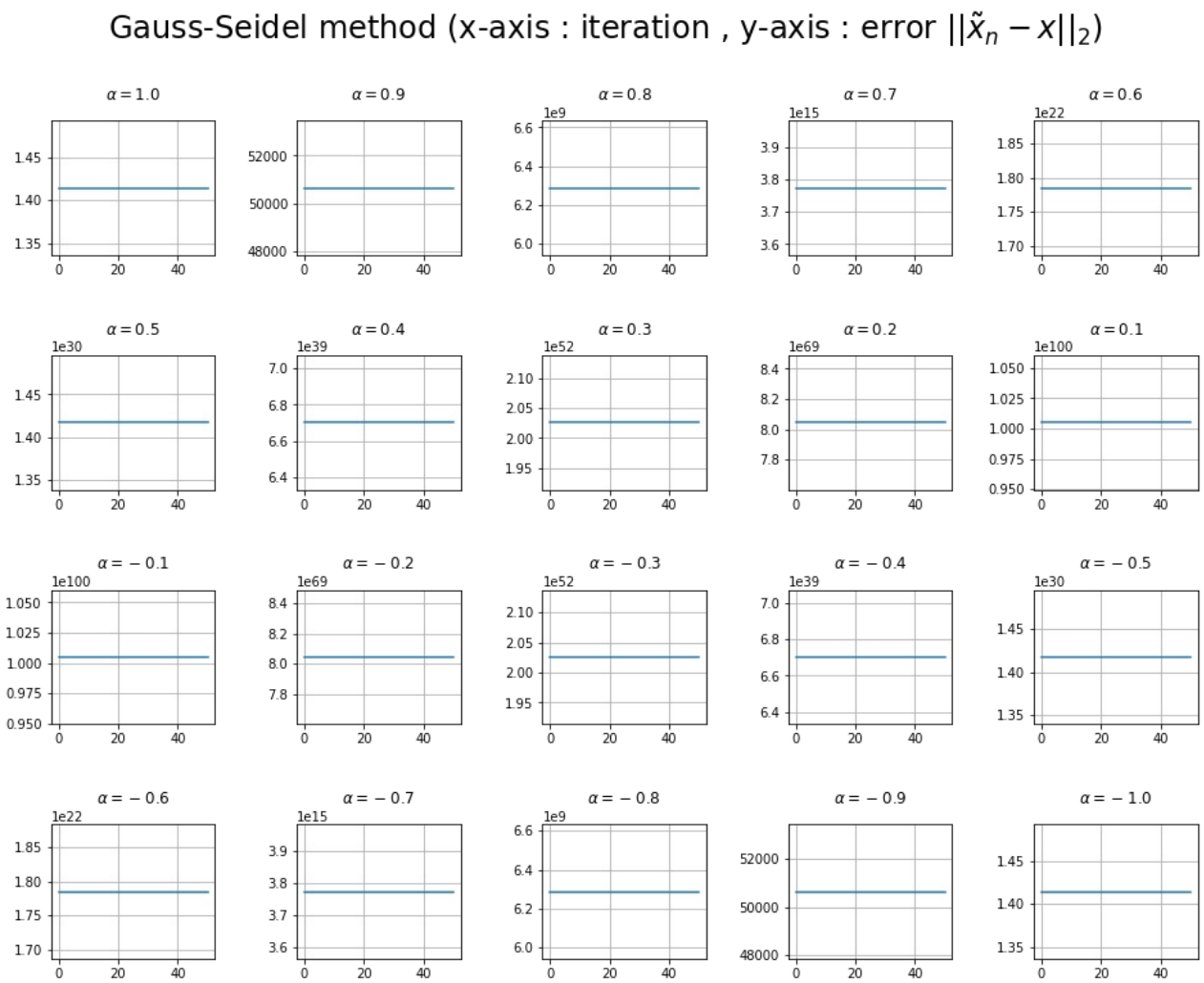

## Convergence Property of Gauss-Seidel Method

In [7]:

```
(fig , axes) = plt.subplots(4 , 5 , figsize = (16 , 12))
plt.subplots_adjust(wspace = 0.5 , hspace = 0.75)
for (i , ax) in enumerate(axes.flatten()):
    A = np.array([[alpha[i] , 0 , 1] , [0 , alpha[i] , 0] , [1 , 0 , alpha[i]]])
    x = np.ones(3)
    b = np.dot(A , x)
    estimation = Gauss_Seidel_method(A , b , iteration = 50)
    error = [np.linalg.norm(estimation[j] - x , 2) for j in range(len(estimation))]

    x_range = np.arange(-np.pi , np.pi , 0.01)
    ax.set_title(r'$\alpha = {}$'.format(alpha[i]) , y = 1.1)
    ax.plot(error)
    ax.grid(True)

plt.suptitle(r'Gauss-Seidel method (x-axis : iteration , y-axis : error $||\tilde{x}_n - x||_2)' , fontsi
ze = 24)
plt.show()
```




在以上程式中，先取 $x$ 為所有元素皆為 1 的vector，並計算 $b = Ax$，接著用Jacobi method和Gauss-Seidel method計算 $Ax = b$ 的近似解 $\tilde{x}$，並觀察真正的解 $x$ 和估計值 $\tilde{x}$ 之間的誤差 $||\tilde{x} - x||_2$。
由上圖大致可以看出，當 $|\alpha| \leq 1$ 時，Jacobi method和Gauss-Seidel method可能不會converge到真正的解。

In [ ]:

```
```