Lab 1

January 16, 2024

[11]:
```
import numpy as np

# SECTION 3 EXERCISES

# 1

x = np.linspace(1,10,10) #closed, num steps
y = np.arange(1,11,1) #half-open, step size
print(x)
print(y)
```

```
[ 1.  2.  3.  4.  5.  6.  7.  8.  9. 10.]
[ 1  2  3  4  5  6  7  8  9 10]
```

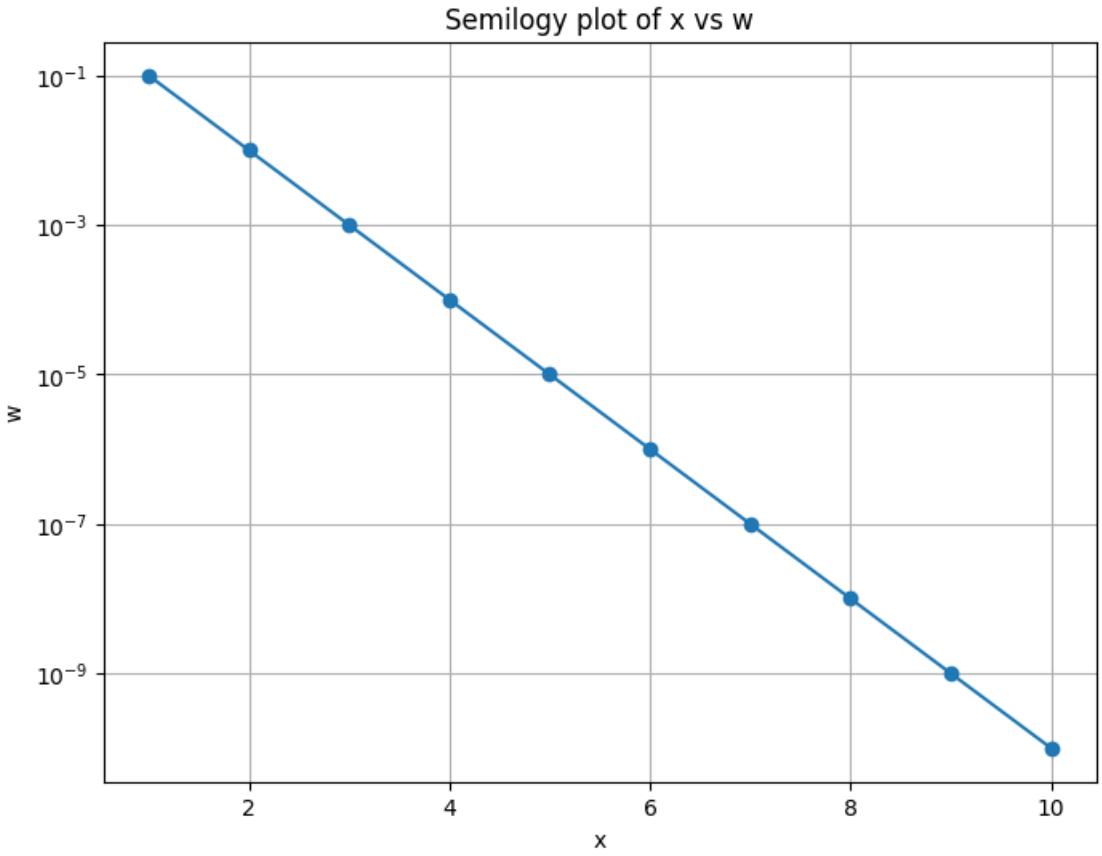
[13]:
```
# 2 and 3

print('The first three entries of x:',x[:3])
```

```
The first three entries of x: [1. 2. 3.]
```

[25]:
```
# 4

w = 10**(-np.linspace(1,10,10))
print(w)

# The entries are 1.e-01 1.e-02 1.e-03 1.e-04 1.e-05 1.e-06 1.e-07 1.e-08 1.
 ↪e-09 1.e-10

x = np.linspace(1,len(w),len(w))
print(x)

import matplotlib.pyplot as plt

plt.figure(figsize=(8, 6))
plt.semilogy(x, w, marker='o')
plt.xlabel('x')
plt.ylabel('w')
plt.title('Semilogy plot of x vs w')
```

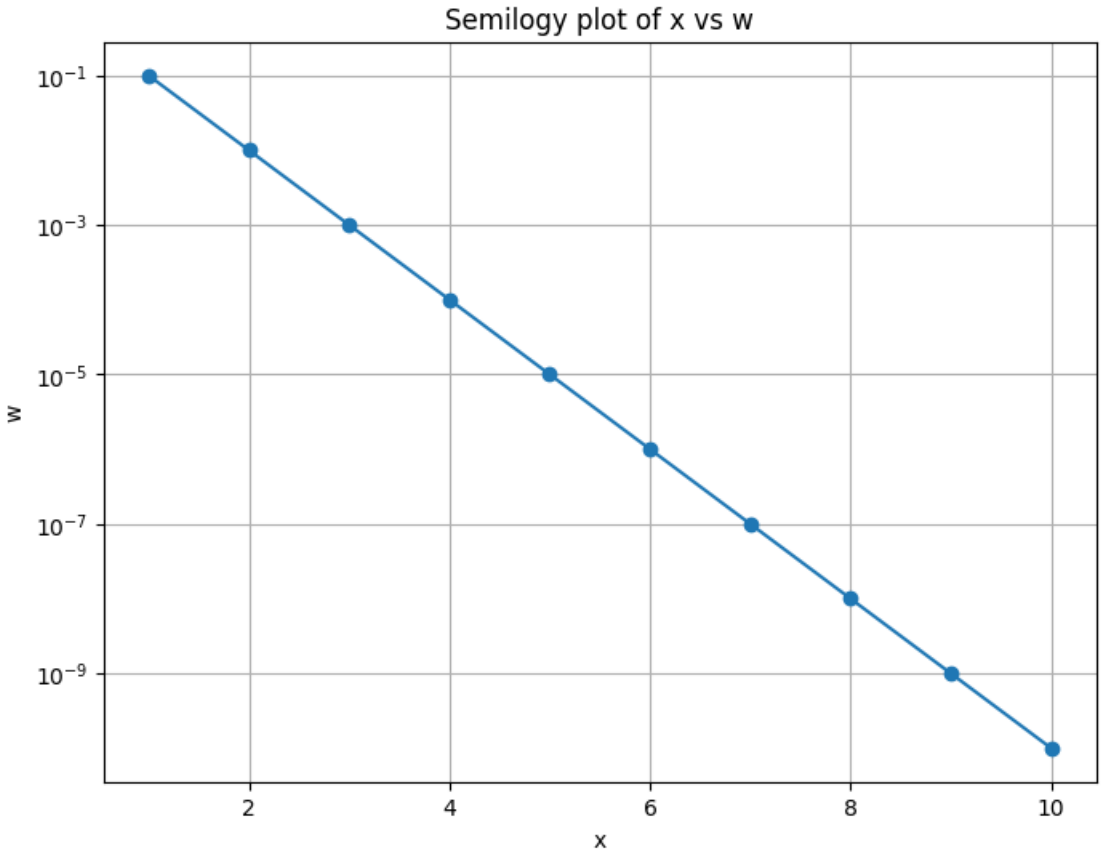
```
plt.grid(True)
plt.show()
```

```
[1.e-01 1.e-02 1.e-03 1.e-04 1.e-05 1.e-06 1.e-07 1.e-08 1.e-09 1.e-10]
[ 1.  2.  3.  4.  5.  6.  7.  8.  9. 10.]
```



[26]:
```
# 5

s = 3*w

print(s)

plt.figure(figsize=(8, 6))
plt.semilogy(x, w, marker='o')
plt.xlabel('x')
plt.ylabel('w')
plt.title('Semilogy plot of x vs w')
plt.grid(True)
plt.show()
```

```
[3.e-01 3.e-02 3.e-03 3.e-04 3.e-05 3.e-06 3.e-07 3.e-08 3.e-09 3.e-10]
```



[41]:
```
# SECTION 4 EXERCISES

# 1

import numpy.linalg as la
import math
def driver():

    n = 2
    y = [1,0]
    w = [0,1]

    # evaluate the dot product of y and w
    dp = dotProduct(y,w,n)
    # print the output
    print('the dot product is : ', dp)
    return
```

```
def dotProduct(x,y,n):
    dp = 0.

    for j in range(n):
        dp = dp + x[j]*y[j]

    return dp

driver()
```

```
the dot product is :  0.0
```

[42]:
```
# 2

def driver():

    n = 2
    mat = [[1,0],[0,1]]
    vec = [0,1]

    # evaluate the dot product of y and w
    prod = matVecMult(mat,vec,n)
    # print the output
    print('the product is : ', prod)
    return

def matVecMult(mat,vec,n):
    prod = [0.]*n

    for j in range(n):
        prod[j] = dotProduct(mat[j],vec,n)

    return prod

def dotProduct(x,y,n):
    dp = 0.

    for j in range(n):
        dp = dp + x[j]*y[j]

    return dp

driver()
```

```
the product is :  [0.0, 1.0]
```

[43]:
```
# 3

y = [1,0]
w = [0,1]
print(np.dot(y,w))

mat = [[1,0],[0,1]]
vec = [0,1]
print(np.matmul(mat,vec))

# The built-in functions are faster.
```

```
0
[0 1]
```

[ ]: