Convex Optimization

Tanmay Garg CS20BTECH11063

In [ ]:

In [ ]:
```
#Importing required Libraries
import numpy as np
import matplotlib.pyplot as plt
import cvxpy as cp
import math
```

In [ ]:
```
P = np.matrix('1 -0.5; -0.5 2')
Q = np.matrix('-1 0')
x = cp.Variable((2,1))
Obj = cp.Minimize(cp.quad_form(x, P) + Q@x)
cons = [
    np.matrix('1 -2; 1 4')@x <= np.matrix('-2 ;-3'),
    np.matrix('5 -76')@x <= 1
]
prob = cp.Problem(Obj, cons).solve()
print(prob)
print(x.value)
```

```
7.444444444444442
[[-2.33333333]
 [-0.16666667]]
```

In [ ]:
```
P = np.matrix('1 -0.5; -0.5 2')
Q = np.matrix('-1 0')
A = np.matrix('1 -2; 1 4; 5 -76')
b = np.matrix('-2 ;-3; 1')
lambd = cp.Variable((3,1))
x_opt = -0.5*np.linalg.inv(P)@(A.T@lambd+Q.T)
# print(x_opt.shape)
Obj = cp.Minimize(cp.quad_form(x_opt, P) + Q@x_opt)
cons = [
    # np.matrix('1 -2; 1 4; 5 -76')@x_opt <= np.matrix('-2 ;-3; 1'),
    A@x_opt <= b,
```

```
]
prob = cp.Problem(Obj, cons).solve()
print(lambd.value)
```

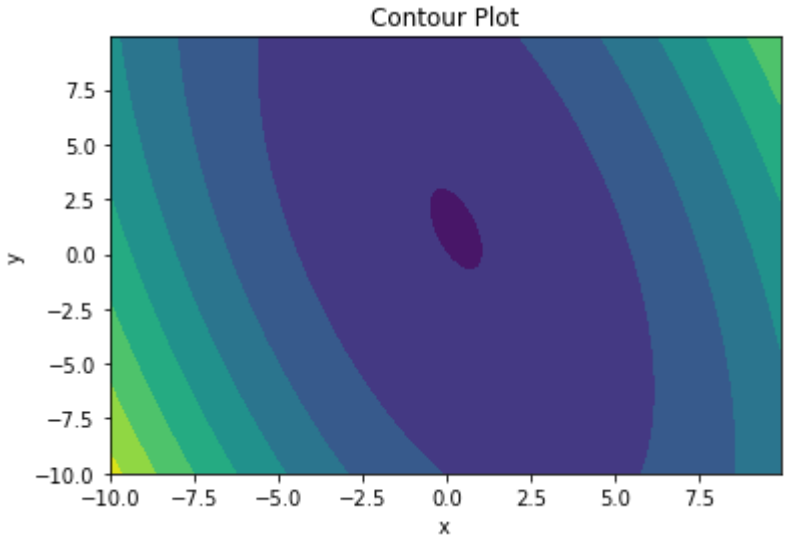
```
[[ 3.94444442e+00]
 [ 1.55555555e+00]
 [-2.19035271e-14]]
```

In [ ]:

```
x = np.arange(-10, 10, 0.1)
y = np.arange(-10, 10, 0.1)
[X, Y] = np.meshgrid(x, y)
fig, ax = plt.subplots(1, 1)
Z = (11*X**2+2*Y**2+5*X*Y-6*Y-12*X)/18
ax.contourf(X, Y, Z)

ax.set_title('Contour Plot')
ax.set_xlabel('x')
ax.set_ylabel('y')

plt.show()
```



In [ ]:

```
X = -2
Y = -3
Z1 = (11*X**2+2*Y**2+5*X*Y-6*Y-12*X)/18
print(Z1)

X = -2 + 0.001
Z2 = (11*X**2+2*Y**2+5*X*Y-6*Y-12*X)/18
```

```
X = -2
Y = -3 + 0.001
Z3 = (11*X**2+2*Y**2+5*X*Y-6*Y-12*X)/18

pd_y = (Z3-Z1)/0.001
pd_x = (Z2-Z1)/0.001
print(pd_x)
print(pd_y)
```

```
7.444444444444445
-3.9438333333334796
-1.5554444444445537
```