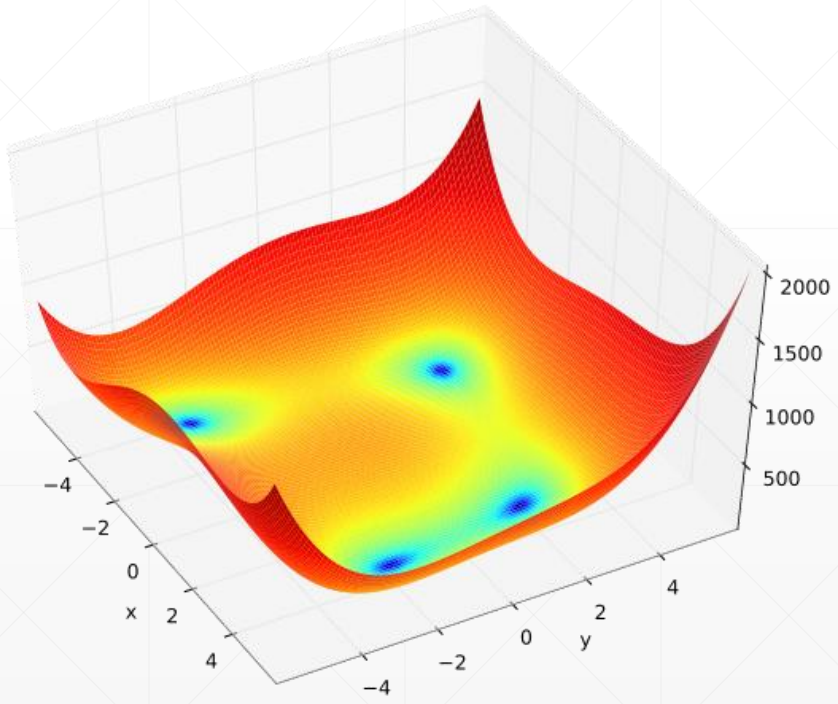
PyTorch

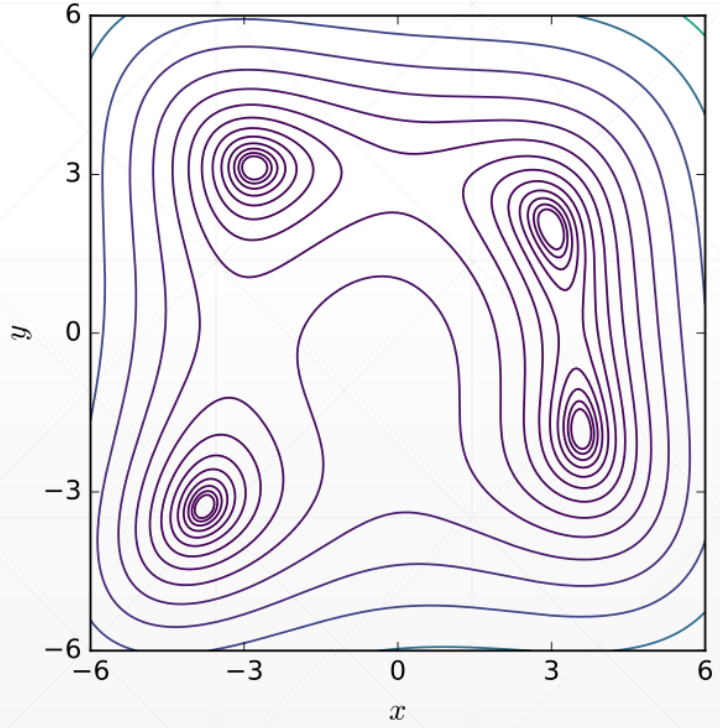
2D函数优化实例

主讲人：龙良曲

Himmelblau function

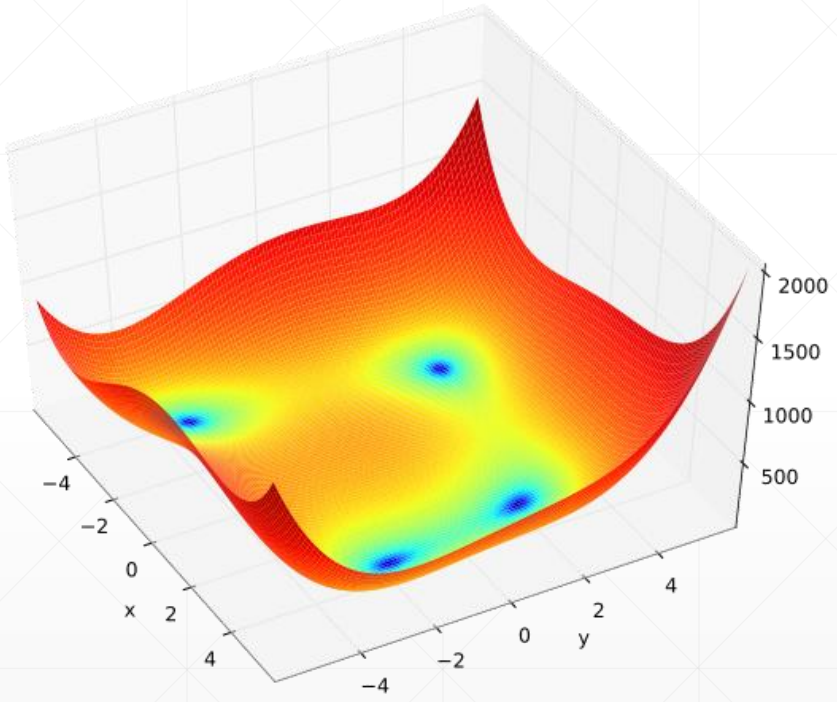
$$f(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2.$$





Minima

- $f(3.0, 2.0) = 0.0,$
- $f(-2.805118, 3.131312) = 0.0,$
- $f(-3.779310, -3.283186) = 0.0$
- $f(3.584428, -1.848126) = 0.0.$

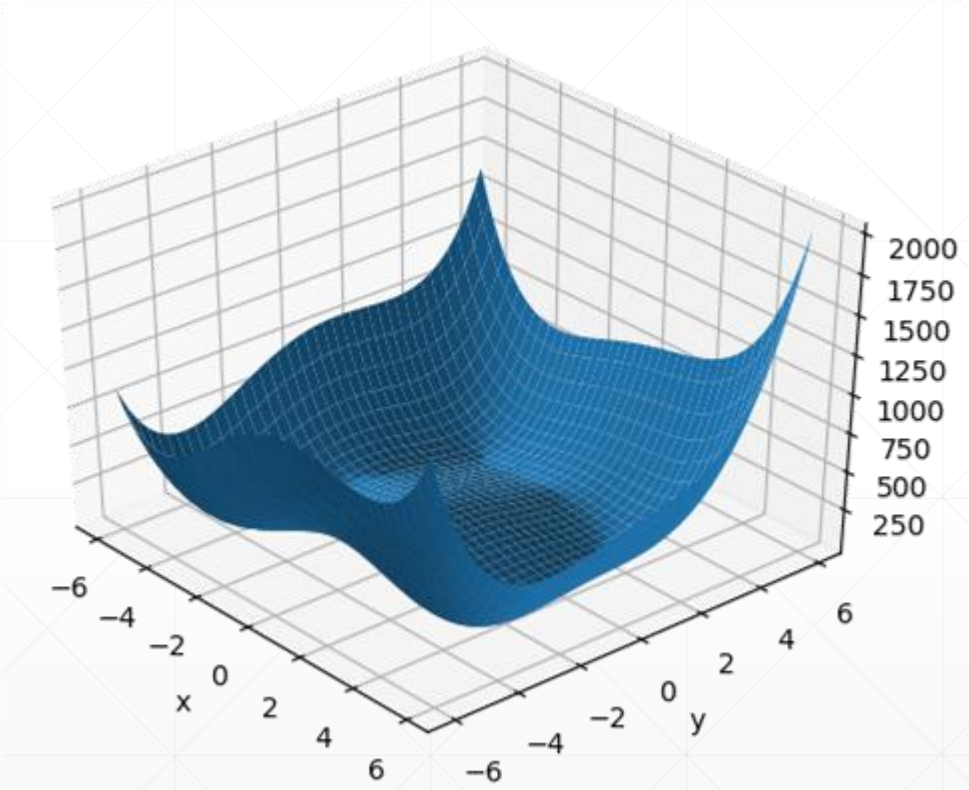


Plot

```python
def himmelblau(x):
    return (x[0] ** 2 + x[1] - 11) ** 2 + (x[0] + x[1] ** 2 - 7) ** 2

x = np.arange(-6, 6, 0.1)
y = np.arange(-6, 6, 0.1)
print('x,y range:', x.shape, y.shape)
X, Y = np.meshgrid(x, y)
print('X,Y maps:', X.shape, Y.shape)
Z = himmelblau([X, Y])

fig = plt.figure('himmelblau')
ax = fig.gca(projection='3d')
ax.plot_surface(X, Y, Z)
ax.view_init(60, -30)
ax.set_xlabel('x')
ax.set_ylabel('y')
plt.show()
```



Gradient Descent

迭代轮数

清零

移步梯度下降

```python
# [1., 0.], [-4, 0.], [4, 0.]
x = torch.tensor([0., 0.], requires_grad=True)
optimizer = torch.optim.Adam([x], lr=1e-3)
for step in range(20000):

    pred = himmelblau(x)

    optimizer.zero_grad()
    pred.backward()
    optimizer.step()

    if step % 2000 == 0:
        print ('step {}: x = {}, f(x) = {}'
               .format(step, x.tolist(), pred.item()))
```

下一课时

MNIST反向传播

Thank You.