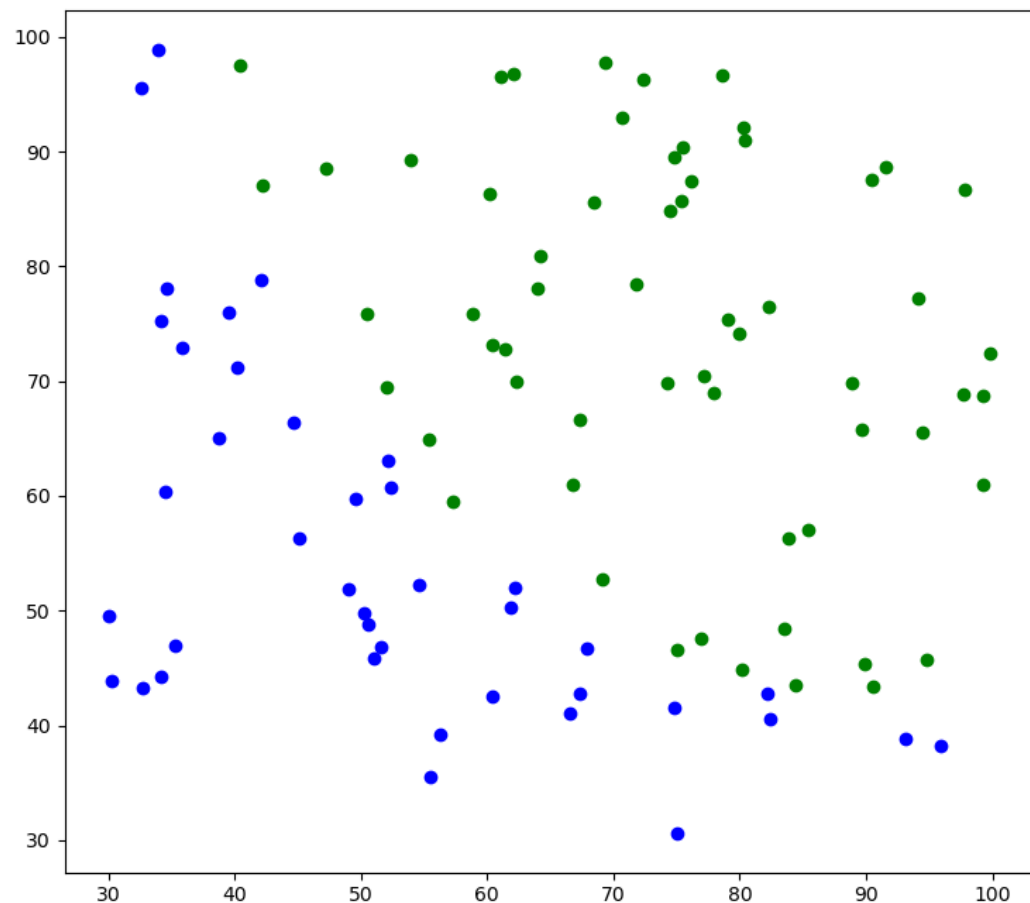


# 神经网络

# 逻辑回归

```
1 34.62365962451697,78.0246928153624,0
2 30.28671076822607,43.89499752400101,0
3 35.84740876993872,72.90219802708364,0
4 60.18259938620976,86.30855209546826,1
5 79.0327360507101,75.3443764369103,1
6 45.08327747668339,56.3163717815305,0
7 61.10666453684766,96.51142588489624,1
8 75.02474556738889,46.55401354116538,1
9 76.09878670226257,87.42056971926803,1
10 84.43281996120035,43.53339331072109,1
```

- 考虑上图和右图展示的数据集
  - 横坐标和纵坐标分别表示两门课成绩
  - 绿色代表录取，蓝色代表不录取
  - 判断录取标准？
- 是否可以用一条直线来分类？



“大学生录取预测分类”

## 实现细节

- 梯度下降方式
  - 随机梯度下降
  - 全批量：一次性计算
  - 小批量样本：设置batch size
- 学习率的选取？
- 何时终止
  - 设置迭代次数
  - 梯度阈值
  - .....

## 实验任务

- 在数据集 **data1.txt** 上完成大学生录取预测分类训练：
  - 选择合适的损失函数
  - 利用训练集完成网络训练
  - 画出数据可视化图、loss 曲线图，计算模型收敛后的分类准确率。
- Deadline
  - **无需提交**
- 选做题（学有余力同学可做）
  - 在 data2.txt 完成分类。
  - 在情感分类数据集完成情感预测分类。

## 附录

矩阵求导: <https://zhuanlan.zhihu.com/p/137702347>

矩阵运算库Numpy教程:

<https://www.runoob.com/numpy/numpy-tutorial.html>

Matplotlib可视化教程:

<https://www.runoob.com/matplotlib/matplotlib-tutorial.html>

```
>>> import numpy as np
>>> x = np.random.rand(2, 3)
>>> x
array([[0.35179327, 0.84536476, 0.35445497],
       [0.65371529, 0.76534139, 0.0040888 ]])
>>> y = np.random.random((2, 3))
>>> y
array([[0.66791057, 0.25198801, 0.10348534],
       [0.37049378, 0.92085611, 0.51674214]])
>>>
```

```
>>> x.T
array([[0.35179327, 0.65371529],
       [0.84536476, 0.76534139],
       [0.35445497, 0.0040888 ]])
>>> x[1]
array([0.65371529, 0.76534139, 0.0040888 ])
>>> x[1][2]
0.004088804618508801
>>> x[1, 2]
0.004088804618508801
>>> x.T[1]
array([0.84536476, 0.76534139])
>>>
```

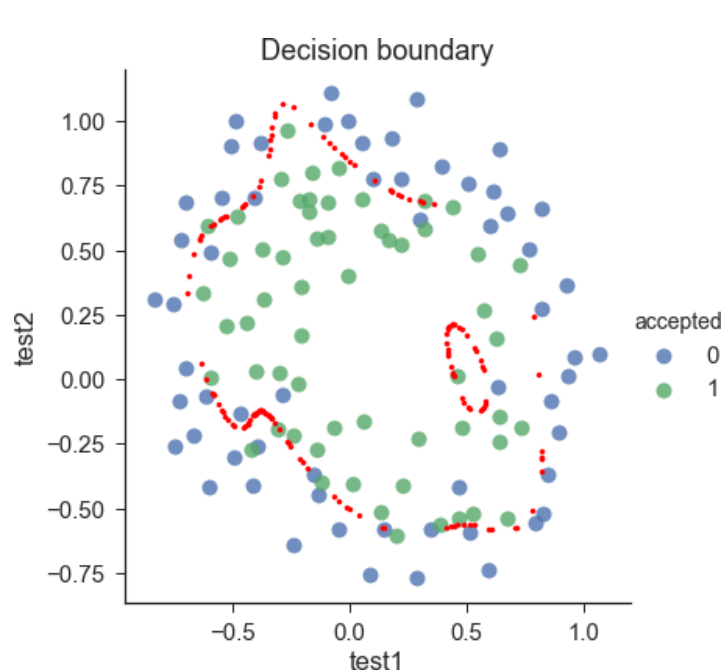
```
>>> x
array([[0.35179327, 0.84536476, 0.35445497],
       [0.65371529, 0.76534139, 0.0040888 ]])
>>> y
array([[0.66791057, 0.25198801, 0.10348534],
       [0.37049378, 0.92085611, 0.51674214]])
>>> np.vdot(x, y)
1.4337487219538598
>>> np.dot(x, y.T)
array([[0.48466912, 1.09195834],
       [0.62990333, 0.9490796 ]])
>>>
```

- np.random.rand
- np.vdot
- np.dot

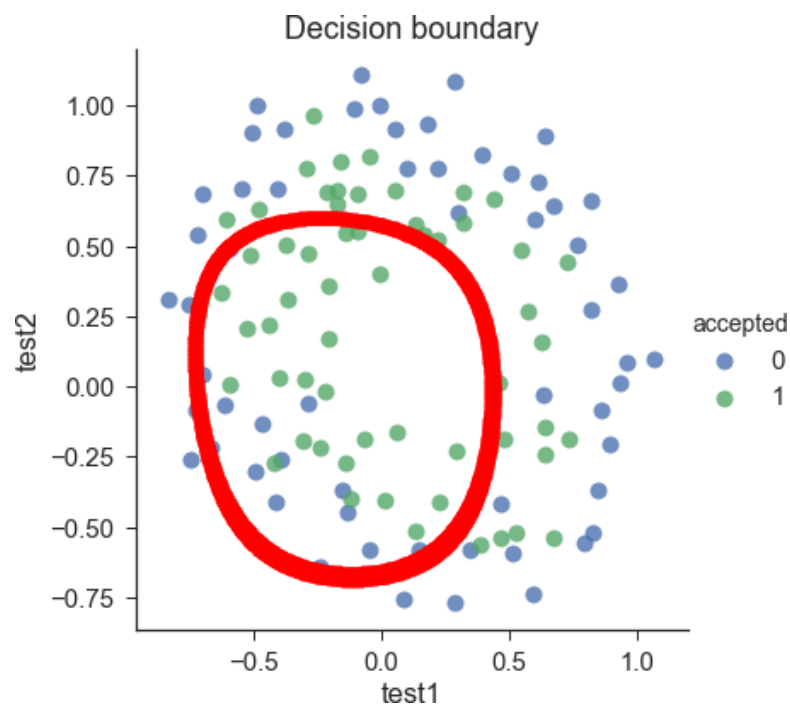
## 正则化处理

✓ 逻辑回归求解—正则化，防止过拟合

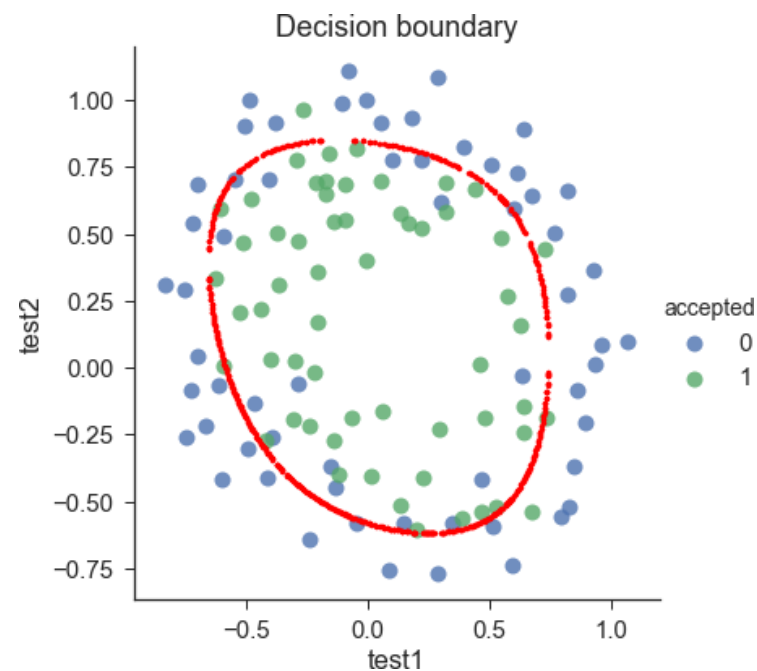
$$J(w) = -\frac{1}{m} l(w) = -\frac{1}{m} \sum_{i=1}^m (y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})) + \frac{\lambda}{2m} \sum_{j=1}^n w_j^2$$



没有正则化，过拟合



正则化过度，欠拟合



适当的正则化

Thanks!