机器学习的本质：令计算机寻找一个函数，能够根据输入得到输出。

大致可以分类为两大任务：分类（垃圾邮件识别）与回归（价格预测）

Pytorch基础语法

1.使用确定值创建向量、矩阵与高维张量

|  |
| --- |
| a = torch.tensor([1,2])  b = torch.tensor([[1,2],[3,4]])  c = torch.tensor([[[1,2],[3,4]], [[1,2],[9,6]]]) |

2.使用函数创建张量：空张量、随机元素张量、全为0张量、全为1张量。Dtype可以指定元素类型。函数名后加\_like，并第一个参数为另一张量可以直接指定为这一张量的size。

|  |
| --- |
| d = torch.empty(3,4)  e = torch.rand(3,4)  f = torch.zeros(3,4,dtype=torch.long)  g = torch.ones(3,4,dtype=torch.double)  h = torch.rand\_like(g, dtype=torch.float) |

3 Tensor的基本运算。

|  |
| --- |
| i = j + k  i = torch.add(j,k)  j.add\_(k) # same to j = j + k  n = torch.tensor([1, 2])  p = torch.tensor([2, 3])  q = torch.dot(n,p)  n = torch.ones(2,3)  p = torch.ones(3,4)  q = torch.mm(n,p) # mm is alias to matmul  q = n @ p  n.mm(p)  n = torch.ones(2,3)  p = torch.ones(2,3)  q = torch.mul(n,p)  q = n \* p  u = torch.pow(p, 2)  u = q \*\* 2 |

4.一些张量单元运算符，包括根号、取对数、取整等。

|  |
| --- |
| r = torch.tensor([2,3])  o = torch.sqrt(r)  o = torch.log(r)  o = torch.log2(r)  o = torch.floor(r)  o = torch.ceil(r)  o = torch.round(r)  o = torch.trunc(r) |

5.一些张量单元运算符，包括根号、取对数、取整等。

|  |
| --- |
| u = torch.tensor([[1,2,3],[4,5,6]])  sz1, sz2 = u.shape, u.size() |

6. 判断是否有GPU，把数据放在GPU上。

|  |
| --- |
| if torch.cuda.is\_available():  device = torch.device('cuda')  else:  device = torch.device('cpu')  x = torch.tensor([1,2])  x.to(device)  x2 = torch.tensor([1,2], device=device) |

7.其他常用函数包括求均值、求和、求积、求最大最小值、取最大最小值下标等。

|  |
| --- |
| a = torch.tensor([[1., 2.], [3., 4.]])  n = torch.mean(a, dim = 1)  # result: tensor([1.5000, 3.5000])  n = torch.sum(a, dim = 1)  n = torch.prod(a, dim = 1)  n = torch.max(a, dim = 1)  n = torch.min(a, dim = 1)  n = torch.argmax(a, dim = 1)  n = torch.argmin(a, dim = 1) |