

武汉纺织大学计算机与人工智能学院

深度学习基础

3. 神经网络入门

吴晓堃

xkun.wu at gmail dot com

2021/03/29

Outline

本章内容

神经网络剖析。IMDB 电影评论分类：二分类问题。路透社新闻分类：多分类问题。
波士顿预测房价：标量回归问题。实践：三类基本问题。

重点：层、神经网络拓扑、监督学习的三类基本使用场景；

难点：监督学习中三类基本问题的 Keras 基础实现。

3

学习目标

4 / 13

- 理解神经网络的基本构成单元（层）及其拓扑结构；
- 理解监督学习的三类基本使用场景：二分类问题、多分类问题和标量回归问题；
- 掌握监督学习中三类基本问题的 Keras 基础实现。

4

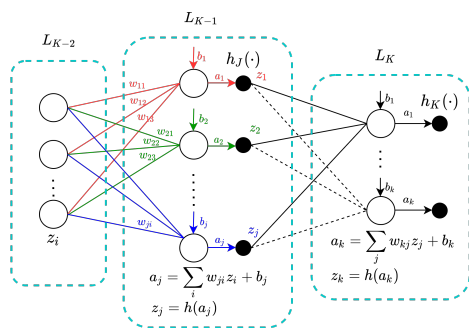
概念

层

层是数据处理单元，将输入张量转换为输出张量。

模型

层构成的网络（有向无环图）；网络的拓扑结构定义了一个假设空间。例如：密集连接网络假设输入特征中没有特定结构。



5

监督学习的三大基本问题

二分类

- 概率输出：sigmoid 激活；
- 损失函数：二元交叉熵 (binary_crossentropy)。

多分类

- 概率输出：softmax 激活；
- 损失函数：分类交叉熵 (categorical_crossentropy)。

回归

- 标量输出：无激活；
- 损失函数：均方误差 (MSE, mean squared error)。

6

二分类问题

```
from keras import models
from keras import layers

model = models.Sequential()
model.add(layers.Dense(32, activation='relu', input_shape=(num_input_features,)))
model.add(layers.Dense(32, activation='relu'))
model.add(layers.Dense(1, activation='sigmoid'))

model.compile(optimizer='rmsprop', loss='binary_crossentropy')
```

7

单标签多分类问题

```
from keras import models
from keras import layers

model = models.Sequential()
model.add(layers.Dense(32, activation='relu', input_shape=(num_input_features,)))
model.add(layers.Dense(32, activation='relu'))
model.add(layers.Dense(num_classes, activation='softmax'))

model.compile(optimizer='rmsprop', loss='categorical_crossentropy')
```

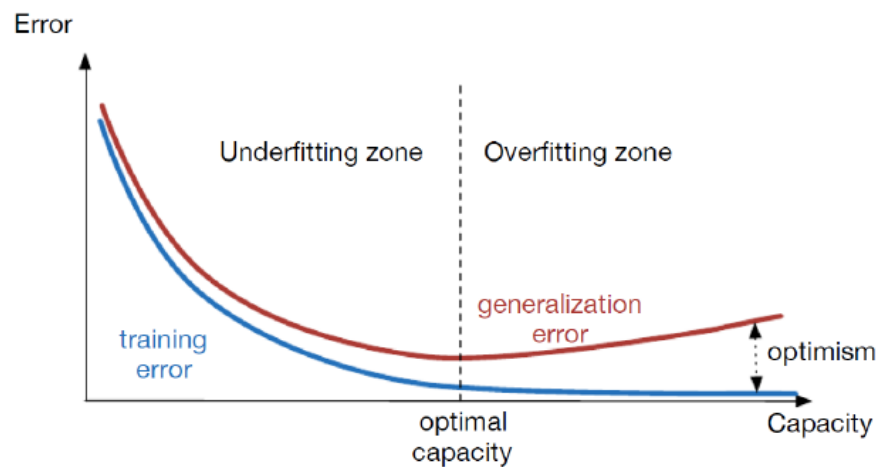
8

```
from keras import models
from keras import layers

model = models.Sequential()
model.add(layers.Dense(32, activation='relu', input_shape=(num_input_features,)))
model.add(layers.Dense(32, activation='relu'))
model.add(layers.Dense(num_classes, activation='sigmoid'))

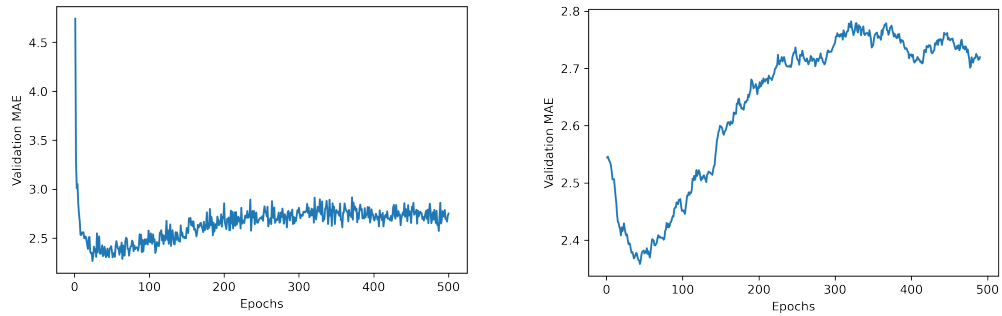
model.compile(optimizer='rmsprop', loss='binary_crossentropy')
```

9



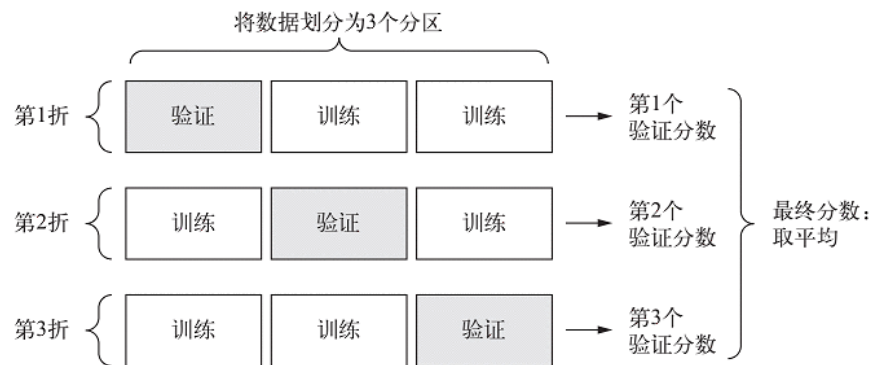
10

过拟合与欠拟合：实际曲线



11

K-fold 交叉验证



12

```
from keras import models
from keras import layers

model = models.Sequential()
model.add(layers.Dense(32, activation='relu', input_shape=(num_input_features,)))
model.add(layers.Dense(32, activation='relu'))
model.add(layers.Dense(num_values))

model.compile(optimizer='rmsprop', loss='mse')
```