PythonとKerasによるディープラーニング
3章 入門：ニューラルネットワーク

19nd302h マブン

3.6回帰の例：住宅価格の予測

- 分類とは、入力データ点の離散的なラベルを1つ予測することでした。もう１種類の一般的な機械学習問題は回帰です。

- 回帰では、離散的なラベルではなく連続値を予測します。

　例えば、気象データに基づいて明日の気温を予測したり、ソフトウェアプロジェクトの仕様に基づいてプロジェクトの完了にかかる時間を予測したりします。

回帰とロジスティック回帰

- 紛らわしいことは、ロジスティック回帰は回帰アルゴリズムではなく、分類アルゴリズムです。

3.6.1 Boston Housing データセット

1970年代中頃のボストン近郊での住宅価格の中央値を予測します。犯罪発生率や地方財産税の税率など、当時のポストン近郊に関するデータ点を使用します。

リスト 3-24：Boston Housing データセットの読み込み

```
from keras.datasets import boston_housing

(train_data, train_targets), (test_data, test_targets) = \
    boston_housing.load_data()
```

このデータセットのデータを調べてみましょう。

```
>>> train_data.shape
(404, 13)
>>> test_data.shape
(102, 13)
```

```
>>>train_targets
array([15.2, 42.3, 50, ……19.4, 19.4, 29.1])
```

3.6.2 データの準備

それぞれ全く異なる供給するには、データ対処するためにベストプラクティスとして広く知られているのは、特徴量ごとの正規化です。入力データの特徴量ごと（入力データ行列の列）に、「特徴量の平均値を引き、標準偏差で割る」という処理を行います。そうすると、特徴量の中心が0になり、標準偏差が1になります。これをＮｕｍｐｙで実行するには簡単です。

- データの正規化

```
mean = train_data.mean(axis=0)
train_data _=  mean
std = train_data.std(axis=0)
train_data /= std

test_data -= mean
test_data / = std
```

テストデータの正規化に使用される値は、訓練データを使って計算されています。機械学習のワークフローでは、たとえデータの正規化のような単純なものであっても、テストデータを使って計算させた値は一切使用すべきではありません。

3.6.3ニューラルネットワークの構築

リスト 3-26：モデルの定義

```
from keras import models
from keras import layers

def build_model():
    # 同じモデルを複数回インスタンス化する必要があるため、
    # モデルをインスタンス化するための関数を使用
    model = models.Sequential()
    model.add(layers.Dense(64, activation='relu',
                           input_shape=(train_data.shape[1],)))
    model.add(layers.Dense(64, activation='relu'))
    model.add(layers.Dense(1))
    model.compile(optimizer='rmsprop', loss='mse', metrics=['mae'])
    return model
```

平均二乗誤差（mean squared error）
予測値と目的値との差の自乗であり、回帰問題の損失関数として広く使用されています。
平均絶対誤差（mean absolute error、以下MAE）
予測値と目的値との差の絶対値です。
例えば、この問題においてＭＡＥが0.5である場合、予測値は平均で500ドルずれていることになります。

3.6.4
k分割交差検証によるアプローチの検証

- **エポック数**

エポック数とは、「一つの訓練データを何回繰り返して学習させるか」の数のことです。

- **バリアンス**

予測値のばらつきの大きさです。

つまりバリアンスが大きいと学習データごとに予測値が大きく変化することを表し、バリアンスが小さいと学習データにかかわらず予測値がほぼ一定になるということを表します。

- **k分割交差検証**

利用可能なデータをＫ個のサブセットに分割し、全く同じモデルのインスタンスをＫ個作成します。そして、各モデルをＫ－１個のフォールドで訓練し、残りの1個のフォールドで評価します。そして、最後に、Ｋ個の検証スコアの平均を求めます。通常、Ｋの値は4か5になります。

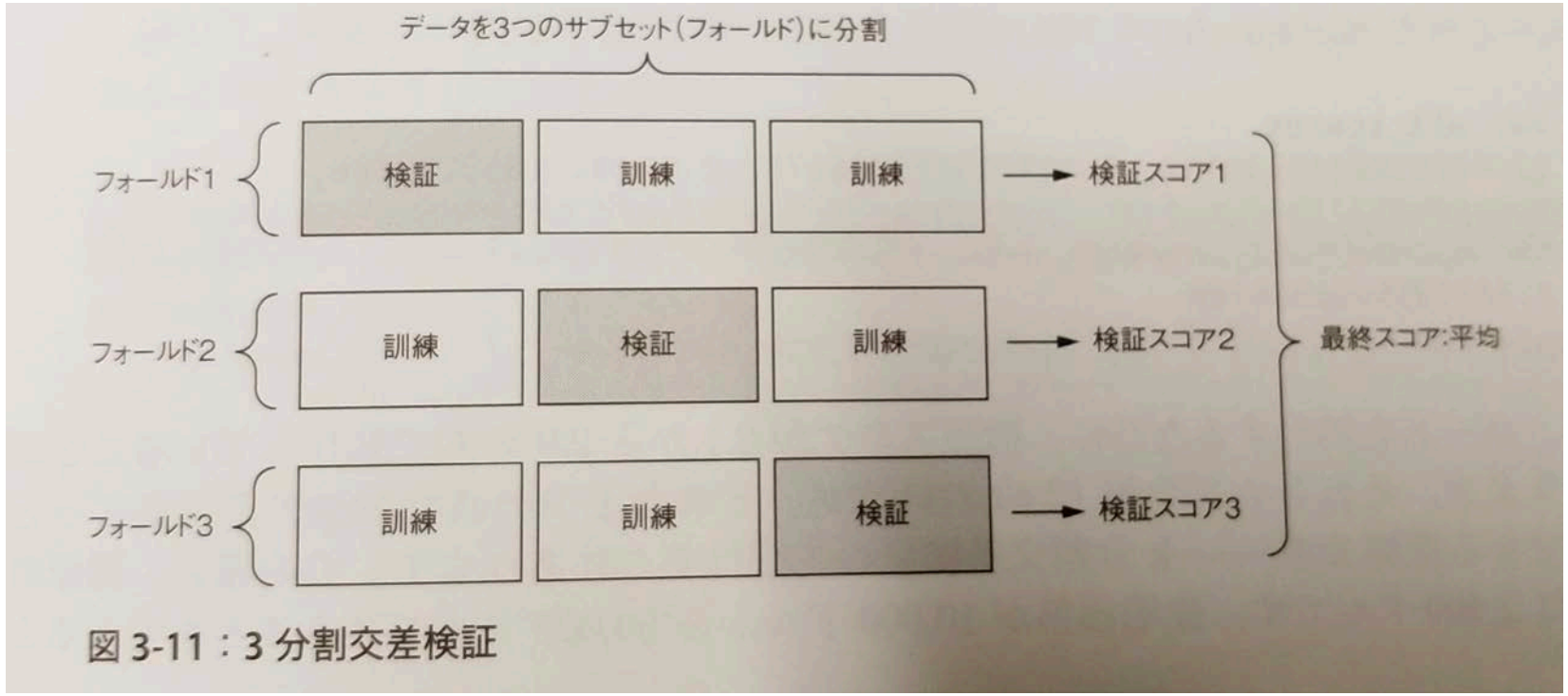

図 3-11：3 分割交差検証

3.6.5　まとめ

- 回帰に使用される損失関数は、分類に使用される損失関数とは異なる。回帰によく使用される損失関数は、平均二乗誤差（MAE）である。
- 同様に、回帰に使用される評価指標も、分類に使用される評価指標とは異なる。当然ながら、回帰には正解率の概念が適用されない。回帰の一般的な評価指標は、平均絶対誤差（ＭＡＥ）である。
- 入力データの特徴量がそれぞれ異なる範囲の値をとる場合は、前処理ステップとして各特徴量の尺度を個別に調整すべきである。
- 利用可能なデータが少ない場合、モデルを正確に評価するのに適した方法はｋ分割交差検証である。
- 利用可能な訓練データが少ない場合、深刻な過学習を回避するには、隠れ層の数が少ない（通常は１つか2つ）小さなネットワークを使用するのが望ましい。