Report on the Experiment

No. 2

Subject Python による機械学習実験 2

Date 2021. 07. 13

Weather 腫れ Temp ℃ Wet %

Class E5
Group 4
Chief
Partner

No 14
Name 小畠 一泰

Kure National College of Technology

1　目的

データの品質と有益な情報の量は, 機械学習アルゴリズムの学習の効率を決定する上で重要な要因となる. このため, 機械学習アルゴリズムに入力する前にデータセットを精査し, 前処理を行うことがきわめて重要となる. 本実験では, 効率的な機械学習モデルを構築するのに不可欠な, データの前処理の手法について背景にある理論や特徴を解説した上で, Python プログラミングによる実装を行う. 今後, AI プログラミングの第一歩を踏み出すために, 解説を読み進めながら, 実際にソースコードを実行して結果を確認しつつ, 効率的な機械学習モデルを構築するための前処理について理解を深めることを本実験の目的とする.

2　考察

2.1　機械学習の専門家や技術者達から Python が広く支持されている理由を述べよ

AI を使ったソフトを効率よく開発するには, ライブラリーの利用が欠かせない. 具体的には Google が開発・公開する深層学習ライブラリー「TensorFlow」や Facebook が開発・公開する深層学習ライブラリー「PyTorch」, 機械学習のアルゴリズムを幅広くカバーするライブラリー「scikit-learn」などがよく使われている. これらはどれもオープンソースであり, 無料で利用できる. そしてほとんどのライブラリーで共通して使えるの言語が Python である. AI を使ったソフトの開発でエンジニアや研究者の誰もが Python を使うため, 情報の蓄積も Python 一色となっている. したがって Python を使うことがもっとも効率的であるからである.

2.2　Python のデータ構造である, リスト, タプル, ディクショナリについてまとめ, リストのスライスについて調べよ

- リスト
  - ミュータブル
  - インデックスで要素にアクセスする
  - スライス: `[start:stop]` や `[start:stop:step]` とすることで任意の値を取得できる
- タプル
  - イミュータブル
  - インデックスで要素にアクセスする
- ディクショナリ
  - ミュータブル
  - 要素へのアクセスは値に一意なキーで行う

2.3　機械学習で使用する, 訓練データセット, テストデータセット, 検証データセットの違いについて説明せよ

- 訓練データセット
  - 重みの学習に使用する
- テストデータセット
  - テストに使用する
- 検証データセット
  - 学習済モデルが汎用性があるのかどうかを検証するために使用する

2.4 考察 4

逐次選択アルゴリズムでは, k 近傍法を用いて正解率の変化を調べた. ここでは, 前回の実験で学習した, ロジスティック回帰, SVM, 決定木, ランダムフォレストを使って元のデータと 5 つのサブセットからなるデータを使って同様の実験を行い, 訓練データセットの正解率, テストデータセットの正解率を表にまとめよ（k 近傍法の結果も示すこと）. この表からわかることを考察せよ.

表 1: データセット, テストデータセットの正解率

サブセット	訓練データセットの正解率	テストデータセットの正解率
k 近傍法	0.95547	0.87288
ロジスティック回帰	0.92833	0.91334
SVM	0.96703	0.76516
決定木	0.94344	0.92832
ランダムフォレスト	0.90592	0.88513

訓練データ, テストデータの正答率がともに高いものは汎用的に使えそうだが, 訓練データが高くテストデータが低いものに関しては分類対象によって使えるかどうかが決まることがわかった.

2.5 考察 5

ランダムフォレストを使って重要度が 0.15 以上の特徴量を選択し, 3 つの特徴量を抽出した. ロジスティック回帰, SVM, 決定木, ランダムフォレスト, k 近傍法を使ってこの 3 つのサブセットからなるデータを使って同様の実験を行い, 訓練データセットの正解率, テストデータセットの正解率を表にまとめよ. この表からわかることを考察せよ.

表 2: データセット, テストデータセットの正解率

サブセット	訓練データセットの正解率	テストデータセットの正解率
k 近傍法	0.97329	0.95147
ロジスティック回帰	0.93654	0.90290
SVM	0.95816	0.92320
決定木	0.49370	0.43194
ランダムフォレスト	0.37430	0.33657

3 つの特徴量によって抽出したことで正答率が格段に下がってしまったものがある. 特徴量を減らすと学習効率が上がる反面, 精度を犠牲にする場合がありトレードオフであることがわかった.

2.6 2 回にわたって Python の scikit-learn ライブラリを用いて機械学習に関する実験を行った感想を述べよ

前回に引き続き今回はより実践的に学べたように感じた. 今後は AI などの機械学習が使用されるケースが増えてくると思うので, 自分の専門分野などに関わらず広い視野を持って学習していきたいと思った.

3 参考文献

1. https://xtech.nikkei.com/atcl/nxt/column/18/00628/030400001/
2. https://qiita.com/taro-ari/items/9f54536fe3c623813db1
3. https://note.nkmk.me/python-slice-usage/
4. https://shirakonotempura.hatenablog.com/entry/2019/01/18/031645