Hands on Machine Learning with Scikit-Learn &TensolFlow

Chapter 6

Decision trees

Created by Yusuke FUJIMOTO

はじめに

- この資料は「Hands-On Machine Learning with Scikit-Learn and TensorFlow - O'Reilly Media」を読んだ際の（主にソースコードに関する）簡単な解説を残したものです。
- 全部を解説したわけではないので注意
- 余裕があればソースコード周りの背景知識もまとめたい
- 何かあったら yukkyo12221222@gmail.com まで

Chapter 6

Decision trees

今回のポイント

- Gini 係数で良し悪しを判断している
- 分類だけでなく回帰もできる
-

Training and visualizing a decision tree

以下のようにして作成した木を見せることができる。
すごく便利。

```
from sklearn.tree import export_graphviz

export_graphviz(
        tree_clf,
        out_file=image_path("iris_tree.dot"),
        feature_names=iris.feature_names[2:],
        class_names=iris.target_names,
        rounded=True,
        filled=True
    )
```

Making Predictions（How training）

Gini 係数。クラスが純粋ではない度合い（不純度）を表すスコア。クラスが綺麗に分かれていると下がる。

$$G_i = 1 - \sum_{k=1}^{n} {p_{i,k}}^2$$

Scikit-Learn では CART アルゴリズムで Gini 係数が下がるようにモデル（のパラメータ）を学習する。

「Model Interpretation: White Box VS Black Box」

説明のしやすさが変わる。

- White Box model
  - 学習や判断の内訳が目に見えやすいモデル
  - 決定木
  - 線形回帰系など
- Black Box model
  - 学習や判断の内訳が目に見えづらいモデル
  - ニューラルデットワーク（Deep learning）
  - ランダムフォレスト

The CART Training Algorithm

CART アルゴリズムにおけるコスト関数（最小化したい対象）。これを greedy に解いていく。

$$J(k, t_k) = \frac{m_{\text{left}}}{m} G_{\text{left}} + \frac{m_{\text{right}}}{m} G_{\text{right}}$$

- k : ある1つの特徴
- t_k : その特徴 k のしきい値（分岐点）
- G_* : 左右の各集合の不純度
- m_* : 左右の各集合内のデータ数
  → この関数が小さいほどきれいに分かれている

Computional Complexity

- オーダー: 計算の複雑さ（かかる時間やスケール）を表す式。プログラミングなどよく出る。データが増えたときにどんな風に計算時間が変わるかなどを表す。$O(n)$ などで表される、
- 例: $O(n^2)$ のアルゴリズムの場合、データの数が 2 倍になると処理にかかる時間は $2^2 = 4$ 倍に膨れ上がる。
- $O(n^2)$ はまだ可愛い方。
- $O(a^n)$ はかなりやばい。

- 決定木の場合、だいたいノード（分岐点）はだいたい $O(log_2(m))$ 個できる。よって予測にかかる時間も $O(log_2(m))$。
- しかし、トレーニングには $O(n \times m\ log(m))$ 必要。データ数が多かったり特徴次元が多いと時間がかかるようになる。
- m はデータの数、n は特徴の種類数？（特徴次元？）

$P \neq NP$ 予想（問題）

- P : 多項式時間で判定できる（解くことができる）問題
  - 例: 簡単な線形問題等の最適値を探す
- NP : 多項式時間で正当性の判定（検証ができる）問題
  - 例: 探した解が正しいかを確認（当てはめる）
- NP-Hard problem : めっちゃ解くのが難しい（解くのに時間がかかる）問題のこと
- もし $P = NP$ だとめっちゃ難しそうに見えた問題も簡単に解けることになる　→　色々困る

Gini Impurity or Entropy

複雑さを表す尺度の１つ。

$$H_i = -\sum_{k=1,\ p_{i,k}\neq 0}^{n} p_{i,k}\,\log(p_{i,k})$$

- Gini 不純度の方が計算が僅かに速いので基本はこっち
- ただし、Gini 不純度より エントロピーの方が若干バランス良く木を作る傾向がある

Regularization Hyperparameters

Regression

Instability