パターン認識 2021 最終レポート

澤 祐里（21–1–037–0801）

2021 年 8 月 12 日

目次

1 目的

以下の内容についてレポートにまとめる。

1.1 課題 1

多くの従業員が勤務する工場の入退室管理をカメラで撮った画像中の顔認識で行うことになった。貴方はエンジニアとして、その設計から運用開始までの責任者となった。学習データの収集、識別器のチューニングなどを行い、十分な精度が得られると判断して運用を開始したが、正しく従業員を認識しないというクレームが寄せられた。それに関して以下のような点について考察して 200 字以上で述べよ。

- 運用前にチューニングを行って十分な精度が得られていた識別器で何が起こっていると言えるか
- それを回避するにはどのような方策が考えられるか

1.2 課題 2

「もはや SVM（サポートベクトルマシン）を使う理由はなく、使っている人は最新の情報をキャッチできていない」という趣旨の若干煽り気味のツイートであるが、一理あるという反応もある。　このツイートに関して、以下のような点について調査・考察して 200 字以上で述べよ。

- 「もはや SVM を使用する理由はない」という理由
- それでもこの講義において SVM を取り扱った理由

1.3 課題 3

1.3.1 課題 3-1

softmax 手法で、学習率と割引率を変化させて、更新回数と方策の比較を行う。別途配布しているスクリプト（ipynb ファイル）を Google Colaboratory で実行して 20 Newsgroups dataset の識別を行うためにどの識別手法が一番適しているかを考察せよ。ここでは、識別手法として、

- k 最近傍法：kNN
- 線形サポートベクトルマシン：linearSVM
- 非線形サポートベクトルマシン：nonlinearSVM
- ランダムフォレスト：randomForest

を候補とする。スクリプトでは他の手法も選択できるが、まずは上記のみを対象とすること。考察の中では、

- 比較対象の識別手法の評価結果（表などで分かりやすくまとめる）
- その結果から判断できる最良の識別手法とその理由

を述べること。ここでは第 10 回のスライドで説明した識別手法の選択指針に従って最良の手法を判断すること。精度が 0。05 程度違うものは同じ精度であるとし、汎化性能が高くなることが見込まれるものを選択してその理由と共に述べること。

1.3.2 課題 3-2

次に、汎化性能のことは忘れて、スクリプトで得られる精度が少しでも高くなる結果を目指すことにする。配布しているスクリプトでは

- k 最近傍法：kNN
- ロジスティック回帰：logistic
- 線形サポートベクトルマシン：linearSVM
- 非線形サポートベクトルマシン：nonlinearSVM
- 決定木：decisionTree
- アダブースト：adaBoost
- ランダムフォレスト：randomForest

が選択できる。これらすべてに対して、精度を評価して、どれが最良の精度となるか示せ。なお、それぞれの識別手法のパラメータは標準的な設定としているが、どれか一つの手法でよいので、標準的なパラメータから変更してみること。レポートでは以下の項目について述べること。

- どの手法のどのパラメータを変更したか（可能であればその理由も）
- 比較対象の識別手法の評価結果（表などで分かりやすくまとめる）
- どれが最良の精度になったか

さらに、Python のプログラミング知識があれば上記以外の識別手法を利用・実装しても構わない（ただし、深層学習を利用することは禁止する）。その場合は、どのような識別手法を利用したかをレポートで述べること。　最後に、様々な手法・パラメータを試した結果、最良と判断した識別手法とそのパラメータ設定を提出すること。提出された識別手法・パラメータを利用してある student_id で得られたデータで性能評価を実施し、提出者全体の中で上位の結果となったものには若干加点する。提出方法は以下で説明する

2 本論

2.1 課題 1

- 運用前にチューニングを行って十分な精度が得られていた識別器で何が起こっていると言えるか
  → 運用前にチューニングを行って十分な精度が得られていた識別器が、本番環境で十分な精度を得られなかったのは、過学習が起きていたからだと考えられる。過学習とは、学習データに存在する外れ値 (同じクラスの他の値から大きく離れた値) なども含めて、識別器が過剰に適合してしまい、データの変化に識別器が対応できず、精度が落ちてしまうことである。例えば、識別境界が線形ベクトルの場合、データを 2 分して判別することしか出来ないため、識別機の自由度は低い。そのため、データに外れ値が存在していたとしても、外れ値を識別するためには、識別境界を外れ値に合わせる必要があり、その場合には他の多くの値が識別できなくなるため、そのような識別境界は選択されない。その結果として、外れ値の識別境界への影響は少ないといえる。しかし、識別境界が非線形ベクトルの場合、識別境界のベクトルの次元数を上げることで、どのような曲線でも描くことができるようになり、識別器の自由度は高くなる。そのため、他の多くのデータを識別したまま、学習データの外れ値なども識別することができる (学習時の精度は高い)。しかし、学習データの数は有限であるため、外れ値の周りに存在する識別空間の全てにデータが存在することはない。そのため、空白である周りの識別空間は、外れ値の定義より、誤識別されている可能性が高く、その結果として本番環境で、学習時には空白だった識別空間のデータが与えられた際に精度が落ちてしまう。
- それを回避するにはどのような方策が考えられるか
  - 単純な識別機を使う。‥線形サポートベクトルマシンなどの単純モデルは、AdaBoost や非線形サポートベクトルマシンなどに比べて、識別器の自由度が低いため、過学習を抑えることができる。
  - 学習データを大量に用意する。‥識別機の自由度が高いほど大量のデータが必要となり、データが多いほど誤識別の可能性は低くなる。
  - Dropout や正則化などを組み込む。‥学習に使用するデータを工夫して使うことで過学習を抑える。

2.2 課題 2

- 「もはや SVM を使用する理由はない」という理由
  → SVM を使う必要がない理由として、SVM は、データ量に対するパフォーマンスが悪く、計算量が多くなること、解釈性に難があることなどが考えられ、また、ディープラーニングなどのパフォーマンスが良い手法が発展していることなども考えられる。カーネル SVM では、グラム行列と呼ばれるデータ数*データ数の行列を計算する必要があるため、データ数が膨大になるほど、計算が爆発的に増えて、多くのメモリが必要となるなど、ビッグデータを扱う現代では、致命的な弱点を持っている。その上で、ディープラーニングでは、同じような非線形問題を解くことができる上に、計算負荷もデータ数に応じて線形に近い増加、並列処理とも相性がいいなどの利点を持っている。
- それでもこの講義において SVM を取り扱った理由
- → SVM は、今では使われることが少なくなったものの、今でも SVM が良い選択であるような問題も存在する。そして、様々な問題を解決する際には、それぞれ最適となる手法が存在する。その際に、多種多様な手法を知っていることは、それだけ解決するための選択肢が多いということであり、また、どの問題に、どの手法が適しているかを理解することができることなどが考えられる。

2.3 課題 3

参考文献

[1] https://drive.google.com/file/d/1pqyeJ20TixzuxkGSjkr2cp0MEKh6jj6w/view?usp=drive_web&authuser=0