

トンネル崩壊予測のための機械学習の適用

東北工業大学 学生会員 ○荒川 玲博
 東北工業大学 学生会員 伊藤 綾一
 東北工業大学 正会員 権 永哲

1. はじめに

トンネル工事を進めるなかで、切羽を多角的視点から見て評価しなければならない場面が多くある。熟練した技術者であれば長年の経験に基づき正しく評価できるが、経験の浅い技術者であればそうはいかない。そのため、機械的に評価するプログラムがあれば客観的に判断できる。また、評価データを使用し事前にトンネル崩壊が予測できれば、人的被害の減少や費用の削減ができる。それを可能にするプログラムを機械学習により作成した。

機械学習のプログラムにとっては取り上げる問題がトンネル崩壊かそうでないかは関係なく、他の問題にも少しの変更で対応できるため、ベースとしての意義もある。

2. 機械学習

今回、ソースコードの開発に用いたのは、プログラム言語の「Python」と、Googleが開発した機械学習向けライブラリの「TensorFlow」である。この二つの言語はプログラミング知識の少ない者でも容易に開発することができる。また、少ないコード数で高度な数値計算を行うことができる。

機械学習とは既知のデータ(学習セット)を用いてプログラムに学習させ、未知のデータの入力から、対応する出力をするものである。学習セットが既知の特徴(X)とそれに対応するラベル(Y)からなり、その関係を学習させるものを「教師あり学習」という。また、出力が連続値であるものを回帰、離散値であるものを分類と呼ぶ。

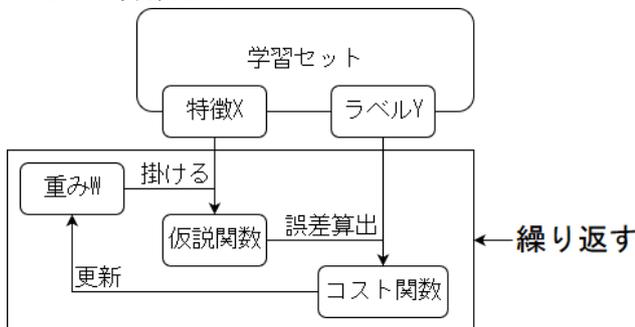


図1 学習の流れ

機械学習の実際の流れとしては、各特徴 X に重み W を掛けた仮説関数と、学習データのラベル Y との

誤差(コスト関数)を求めたのち、誤差が小さくなるように重みを少しずつらす。これを1ステップとし、終了まで何度も繰り返す。最終的にはコスト関数を最小になる時の重み W を求める。未知データの特徴 X に対しこの重み W を掛ければ、対応するラベル Y を予測することができる。

3. 作成したプログラム

今回作成したソースコードは、隠れ層が2層のニューラルネットワークを使用した。ノード数は入力層が13個、隠れ層1層目が16個、2層目は4個、出力層が崩壊と非崩壊の2個である。活性化関数は入力層と隠れ層1層目にシグモイド関数を、隠れ層2層目にソフトマックス関数を使用した。

ニューラルネットワークとは、「教師あり学習」において、入力層と出力層を人間の神経網のように接続する方法である。仮説関数の次数を容易に増やすことが可能で、より精度の高いパターンを学習できる。ニューラルネットワークは、複数の入力を受け取って出力をするノードによって構成されている。

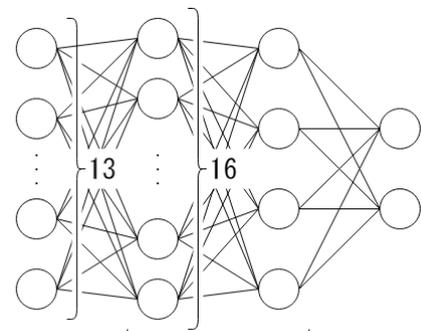


図2 ニューラルネットワーク

活性化関数とはあるノードが次のノードを渡す際値が大きすぎる値にならないように数値変換を行う関数である。現在主流であるのは ReLU 関数だが、比較的シンプルであるシグモイド関数を使用した。

4. 過学習の防止方法

学習結果が既知のデータだけでなく未知のデータに対しても対応できていることを汎化性能という。汎化性能が低い状態を「過学習」というが、この過学

習をいかに防ぐかが機械学習において重要である。学習データとは別に検証データを作成し正確度を確認する必要がある。過学習を防ぐいくつかの方法を最終的なプログラムの作成過程で実装した。

これらの方法はすべて1つの学習セットから複数のパターンを作成することにより過学習を防いでいる。

(1) ミニバッチ学習

最初に実装した方法としてミニバッチ学習というものがあるが、今回のモデルとの相性は悪く、最終的に完成したプログラムでは外している。

この方法は学習実行時に学習セットをすべて使用するのではなく、1ステップごとにランダム選出したデータを使用するというもの。

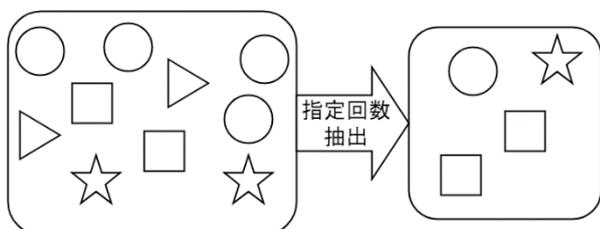


図3 ミニバッチ学習

(2) ドロップアウト

最終的にミニバッチ学習の代わりに実装したのがドロップアウトである。この方法は出力層を除く各層のノードを1ステップごとにランダムで脱落させるというものである。

今回のプログラムではそれぞれの層で35%のノードを不活性化している。実装直後は50%であったが、学習に時間がかかったため少しずつドロップアウト率を落とした。

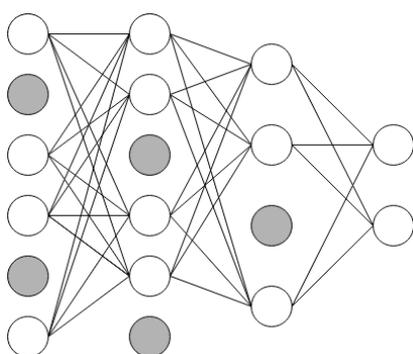


図4 ドロップアウト

5. 使用した学習データ

実際のトンネル崩壊事例を用い、学習セットは約700、検証データは23個を使用した。学習セットの特徴は、幾何構造、地質、不連続性、地下水、掘削、支持/強度の6種13カテゴリで、ラベルは崩壊か非崩壊である。学習セットのうち約650が非崩壊で、残りの50が崩壊のデータである。

ミニバッチ学習がうまくいかなかったのは、この崩壊のデータと非崩壊のデータに偏りがあることが原因である可能性がある。

図5は検証データのラベルと学習の出力結果の比較である。学習結果の今回の研究のモデルでは的中率が95.7% (22/23)であった。この研究の成果としては大きなものが得られた。

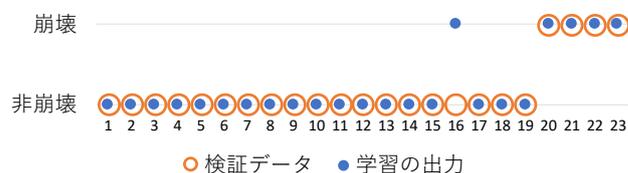


図5 検証データの正解数

6. まとめ

ニューラルネットワークを構築して学習させること自体は、モデルにあまり依存しないのでスムーズにできる。

機械学習の教師あり学習においては過学習を防ぐこと、データの収集、整理に時間を使うことが求められる。

また過学習を防ぐ際にそのモデルによって、有効な手法とそうではない手法があることを勘案してはならない。

今回のモデルでは、学習セットの数が少なく、ラベルの偏りがあった。2分類であったが、そのようなモデルであってもパラメータや使用する手法を調節することで、的中率の高い学習結果を出すことができる。

参考文献

- 1) TensorFlow : <https://www.tensorflow.org/>
- 2) ニューラルネットと TensorFlow 入門のためのオリジナルチュートリアル3 : <https://www.hellocybernetics.tech/entry/2017/07/08/152859>
- 3) TensorFlow を使ったディープラーニング : <https://knowledge.sakura.ad.jp/13725/>
- 4) 機械学習 | Coursera : <https://ja.coursera.org/learn/machine-learning>
- 5) Shin, H., Kwon, Y., Jung, Y., Bae, G. and Kim, Y.: Methodology for quantitative hazard assessment for tunnel collapses based on case histories in Korea, International Journal of Rock Mechanics & Mining Sciences 46 (2009)1072-1087