# HOKUGA 北海学園学術情報リポジトリ

学校法人北海学園 北 海 学 園 大 学 北 海 斎 科 大 学

タイトル	ディープラーニングの概要および北海学園大学工学部 コンピュータ実習室における機械学習環境
著者	串山, 繁; Kushiyama, Shigeru
引用	工学研究 : 北海学園大学大学院工学研究科紀要(18): 11-17
発行日	2018-09-30

### ディープラーニングの概要および北海学園大学工学部 コンピュータ実習室における機械学習環境

串山 繁\*

Outline of Deep Learning and Circumstances of Machine Learning in the Computer Room in Faculty of Engineering, Hokkai-gakuen University

Shigeru Kushiyama\*

### 要旨

最近のニュースでは、AI (Artificial Intelligence:人工知能)に関する話題が頻繁に取り挙げられ、様々な 分野において期待感が増大し、日本政府も若手データサイエンティストを育成するための予算を確保して漸 く支援の手を広げようとしつつある。AI が現在注目されているのは、機械学習とりわけディープラーニン グ(深層学習)が画像処理などの知覚問題において様々な成功を収めていることに起因する。

本資料では、ディープラーニングの概要、本学工学部計算機実習室における機械学習環境を紹介すると共 に TensorFlow を用いた簡単な回帰問題の事例を示す。

### 1. 序

AI を支える核心的な技術がディープラーニン グをはじめとする機械学習である.今日に至る経 緯を振り返ると,大きく以下の3つのステージに 分かれる様である.

第1ステージは、1980年代末を終わりとする Symbolic AI の時代<sup>1)</sup>である.それは人間が読め る記号表現を用いてルールを定義し、プログラム にそれを組込み実行するものであるが、ファジー で複雑な画像分類、言語翻訳等には対応が困難で あった.

第2ステージは,1990~2010年にかけて研究さ れた SVM (Suport Vector Machine)の時代<sup>2)</sup>で ある.これはカーネル法を用いてクラス分け問題 を高次元に写像して分離超平面を求めるもので, 単純な分類問題では高い性能を示すが,画像分類 などの知覚問題については不得手であった.

第3ステージは、2010年以降開花したディープ ラーニング (Deep Learning, DNN: Deep Neural Network とも呼ぶ) である. ニューラルネット ワーク自体は既に存在していたが、特徴量の重み やバイアスのパラメータ設定が専門家の芸術的な センスに依存し,現実問題に適用するには困難で あった.

ディープラーニングがブレークスルーした要因 として、以下の3つ;ハードウェア、データセッ ト、アルゴリズムの進化が指摘されている.参考 文献<sup>3)</sup>に拠れば、その詳細は以下の様である.

①1990年代と2000年代のボトルネックは, ハードウェアとデータであるが,1990~2010年の 間に CPU は約5000倍に高速化,ゲーム市場が契 機となって GPU (Graphics prossesing unit)と呼 ばれる高速な超並列チップが開発された.②デー タ面ではストレージハードウェアが飛躍的に発展 したことに加え,インターネットの台頭で機械学 習用のデータセットの収集,配布が容易となった. ③アルゴリズムの進化としては,誤差逆伝播法が 再評価され,損失値の最小化が確率的勾配降下法 (SGD: Stochastic Gradient Descent)を用いて効 率良く成される様になったことである.併せて活 性化関数やパラメータの調整を制御する最適化関 数の改善もある.以上主に3つの要因が相乗効果 の役割を果たし,今日のブレークスルーに至って

Graduate School of Engineering (Architecture and Building Eng.), Hokkai-Gakuen University

<sup>\*</sup>北海学園大学大学院工学研究科建設工学専攻(建築系)

いる.

応用事例としてはいずれも開発進展途上である が,医療診断,監視カメラ画像の自動識別,知的 なチャットロボット,Google 翻訳,Amazonの製 品推奨システムなど多岐に亘る.

### 2. ディープラーニング (DNN)の概要

### 2.1 DNN の仕組み

機械学習は教師あり,教師なし学習およびその 中間の強化学習に分けられるが,以下では教師あ り機械学習を例に説明する.教師あり機械学習で は,訓練用とテスト用のデータから,未知のデー タに対する答えを導くルールを学習する.ディー プラーニングのディープとは,図-1に示す隠れ層 (hidden layer)が連続し層数が多いことを意味す る.図中の○印はノードと呼ばれる.

上図を用いて,回帰を例に DNN の仕組みを説 明する.

最初に学習のための特徴量(feature)および教 師値(target)から成る観測データを抽出し、3つ の特徴量を入力層(input layer)に与える.次い で、後掲する線形回帰式に非線形の活性化関数 (activation function: af)を作用させて2つの隠れ 層、出力層(output layer)のノード値を求める. 出力層の予測結果を教師値と比較して、誤差に相 当する損失値(loss)を損失関数(loss function) を基に算出し、損失値を最小化するために、確率 的勾配降下法に従って重み(weight)とバイアス (bias)を更新する.更新の制御は最適化関数 (optimizer)で行う.図-1の出力層へ向かう計算 (forward pass)および逆向きの計算(back pass) を充分損失値が最小化するまで指定回数繰り返



し、学習を終える、学習後、訓練データの損失値 とテストデータの損失値の大小関係を比較し、訓 練結果が過剰適合していないことを確認する.

DNN の目的, 即ち学習内容は, 重みとバイア スの最適なパラメータ値を得ることである. 上記 の学習結果を受けて未知の特徴量に対する予測値 を得る. なお, 中間の隠れ層についてはユーザー が層数と各隠れ層のノード数をソースコードで指 定する.

以下に,図-1を例に出力層へ向かう線形回帰式 を示す.図中の線分は重みを表し,各ノードに1 つのバイアスを考慮すると,第1層目の隠れ層は 次式で表される.

$$\label{eq:hamiltonian} \begin{split} _{2}H_{1} &= af\left(_{2}w_{11}*_{1}H_{1}+_{2}w_{21}*_{1}H_{2}+_{2}w_{31}*_{1}H_{3}+_{2}w_{41}*_{1}H_{4}+_{2}b_{1}\right)\\ _{2}H_{2} &= af\left(_{2}w_{12}*_{1}H_{1}+_{2}w_{22}*_{1}H_{2}+_{2}w_{32}*_{1}H_{3}+_{2}w_{42}*_{1}H_{4}+_{2}b_{2}\right)\\ _{2}H_{3} &= af\left(_{2}w_{13}*_{1}H_{1}+_{2}w_{23}*_{1}H_{2}+_{2}w_{33}*_{1}H_{3}+_{2}w_{43}*_{1}H_{4}+_{2}b_{3}\right) \end{split}$$

 $_{2}H_{4} = af(_{2}w_{14}*_{1}H_{1} + _{2}w_{24}*_{1}H_{2} + _{2}w_{34}*_{1}H_{3} + _{2}w_{44}*_{1}H_{4} + _{2}b_{4})$ 最後の出力層については、次の通りとなる。

Y'=af(<sub>3</sub>w<sub>11</sub>\*<sub>2</sub>H<sub>1</sub>+<sub>3</sub>w<sub>21</sub>\*<sub>2</sub>H<sub>2</sub>+<sub>3</sub>w<sub>31</sub>\*<sub>2</sub>H<sub>3</sub>+<sub>3</sub>w<sub>41</sub>\*<sub>2</sub>H<sub>4</sub>+<sub>3</sub>b) ただし, af:活性化関数,最終パラメータ 総数: (3\*4+4) + (4\*4+4) + (4\*1+1) = 41

### 2.2 DNN の学習手順

以下に全訓練データをミニバッチに分けて学習 する手順とそのフローを図-2に示す. 学習手順

1. 訓練データXと教師値(即ち目的値)Yを バッチデータとして抽出



図-2 DNN 訓練手順のフロー(参考文献<sup>3)</sup> 参照)

- X を付与し DNN を実行. 予測値 Y'を取
   得
- 3. 損失関数を用いて予測値 Y'と目的値 Y か ら当該バッチの損失値を計算
- 4. 確率的勾配降下法(stochastic gradient descent)を用いて損失関数の勾配(gradient) を計算
- W=W-step×gradient により勾配とは逆 方向に少し移動し、当該バッチの損失値を低 減する様、重みとバイアスを更新
- 6. 各バッチ毎に訓練データの学習を指定回数 (epoch 数)繰り返して重みとバイアスを最 適化. この制御は最適化関数で指定

## 工学部コンピュータ実習室における機械 学習環境

北海学園大学工学部計算機実習室には,2018年 3月にフリーでダウンロードできる以下の機械学 習関連ライブラリをインストールした.

・Anaconda<sup>1</sup>を用いて, scikit-learn, Numpy, Scipy, matplotlib, Ipython, Jupyter Notebook なお, Pythonは3系である.

 ・ mglearn, TensorFlow, TensorBoard, pandas 上記は実習室 I ~ IIIの全コンピュータに配布す る都合上, TensorFlowの仮想環境で動作する.
 なお,実習室においては,必要に応じて上記の各 種ライブラリをインポートし, Python 3 系でソー スコードを記述する環境であり, R 言語はインス トールされていない.

scikit-learn は、機械学習初心者が最初に利用する良い学習材料と云われている.一方, TensorFlowは、Google Brain チームが開発し、

2015年11月に一般に公開した機械学習等に用い るソフトウェアライブラリで、アルゴリズムの実 装を簡単にする高レベル API (Application Programming Interface): Keras と組み合わせて 用いることが多い.通常, Jupyter Notebookを用 いてブラウザ上でプログラムを編集,分割して実 行する.また,markdown形式でソースコードの 解説も必要箇所に自由に記載可能で、LaTeXの 指定で数式も表記できる.なお,計算機実習室で の Jupyter Notebookの起動手順は、表-1に示す

#### 表-1 実習室での Jupyter Notebook 起動手順

1)スタート/Anaconda Prompt を選択
2)Prompt 起動画面にて,>以下を入力
# tensorflow の仮想環境に切り替え
>conda activate tensorflow
# Jupyter Notebook を起動
>jupyter notebook
3) Jupyter Notebook 起動画面にて,
新規ソースコード入力時:
画面右端 New を押下,Python 3 を選択
作成済ソースコード訂正時:
一覧から. ipynb の該当ファイルを選択
4) 編集済ソースコードは, File/Rename を選択して,
適当なファイル名で保存
5) ファイル保存先は下記の通り
デスクトップ/ユーザー名のフォルダ内

通りである.

各種ライブラリがある中で TensorFlow を加え た理由は, Keras に加えて可視化を補完するツー ル: TensorBoard が付属し,実装モデルの計算グ ラフや様々なデータログを取得することに拠り, 学習状況などを容易に可視化して実装モデルを検 討できるからである.当然ながら隠れ層の設定も 容易で最適化手法も各種取替え可能,パラメータ は自動チューニングされ,利用者の負担は軽い.

大規模な画像処理問題を実装する場合には、先 に述べた GPU がハードウエアに内蔵されている 必要がある.本格的には、GPU を複数備えた Workstation、Ubuntu 上で機械学習関連アプリ ケーションを開発することが理想と指摘されてい る<sup>3)</sup>.実習室のコンピュータは GPU を備えてお らず、Windows 上の CPU で処理可能な範囲に制 約されるが、scikit-learn の参考文献<sup>4)</sup>に記載され ている画像を扱うサンプル例の実行では殆ど支障 は無かった.

なお, Jupyter Notebook で使用するブラウザを Internet Explorer とした場合, 複数ページにまた がる編集・実行結果の印刷が2ページ以降出力さ れない. 原因は不明であるが, 他のブラウザを使 用すればこれを回避できる.

### 4. 回帰問題の事例

ここでは、TensorFlow と Keras を用いた回帰 事例として、建物の定量的安全性評価への実装可 能性についての検討事例を示す。

文献<sup>5)</sup> では、実在 RC 造 11 層建物を例に地震動

<sup>&</sup>lt;sup>1</sup>大規模データ処理,予測解析,科学技術計算向けの Pythonディストリビューション

を確定的,構造特性を不確定と仮定し,シミュレー ションベースの subset MCMC<sup>6</sup> を用いて建物の 破壊生起確率を定量的に評価することを試みた が,10<sup>6</sup>オーダーの破壊確率を求めるには subset MCMC を用いても15万回程度の応答計算を行 う必要があり,計算負荷は非現実的なレベルに近 かった.そこで本報では,上記応答計算のある一 定割合を深層学習の結果で置き換えることの可能 性について検討した.

### 4.1 解析仮定

入力地震動は、図-3 に示す JMA 神戸 NS の地 表面最大速度: PGA (Peak Ground Acceleration) を 50 kine に基準化したレベル 2 地震動を用い た.

解析モデルは文献<sup>5</sup> に同じ RC 造 11 層, X 方向 2スパン, Y 方向2スパン建物をせん断型架構に 置換した表-2 に示す構造階高,層重量等を有する 質点系モデルである.非線形応答解析を行う際の 仮定は,スケルトンカーブ:tri-linear型,履歴則: masing型,構造物の粘性減衰:5%の Rayleigh 減衰(2次迄考慮)と先の文献に同じである.な お,構造物の破壊指標は,最大応答塑性率μmaxと した.

以下の計算においては,建物各層の第1,第2 剛性:SK1,SK2と層降伏せん断力 V<sub>iv</sub>を不確定 とした場合および最弱層(本計算例については1



図-3 レベル2相当基準化入力地震動

階)のみの SK1, SK2, V<sub>iv</sub> を不確定とした場合に ついて考慮した. なお, これら構造特性を変動さ せる際の変動係数はいずれも 5 %, サンプルは, あらかじめ発生総数 600 に指定して均等にサンプ リングする Latin Hypercube Sampling 法で生成 し, 正規分布に従うと仮定した.

### 4.2 解析結果

解析では、やや耐震性の劣る X 方向フレーム についてのみ取り挙げ、先ず全層の構造特性が変 動するとした場合について検討した. 図-4 左上 の(a)図は、変動パラメータが多数ある内の最弱 層1階の第2剛性 SK2を x 軸、同じく層降伏せ ん断力  $V_{iy}$ を y 軸、最大応答塑性率  $\mu_{max}$ を z 軸に とり示したものである. 他の3つ(b)~(d)図は、 個々の構造特性と  $\mu_{max}$ の労布は面的ではなく 膨らみを有していることが分かる. これは、(a) 図で最弱層の x、y 座標値がほぼ同じであっても 上層階の構造特性が異なる場合が多数あり、それ に伴い異なる応答値  $\mu_{max}$ が生じたことに起因して いる. この様な分布形状の場合には、深層学習を 実行しても高い精度の結果を期待出来ない.

次に,最弱層1階のみSK1,SK2,V<sub>iv</sub>を変動す ると仮定した結果について述べる.これは,最弱 層にのみ損傷が集中して建物が崩壊に至る場合に 該当する.なお,DNNを実行する際には,試行 結果を踏まえた表-3に示す前提条件で行った.

先ず,3個の構造特性パラメータ(即ち,深層 学習における特徴量)をk分割交差検証の分割 (fold)毎に表-3に示す様に訓練とテストデータ に分割後,scikit-learnのStandardScalerで標準 化するTensorFlowのプログラムを記述,実行し た.

図-5(a) は, x, y, z 軸に夫々第2剛性, 層降伏 せん断力,  $\mu_{max}$ をとり, 4つの fold を平均した平

階	11F	10F	9F	8F	7F	6F	5F	4F	3F	2F	1F
構造階高 (cm)	295	295	295	295	295	295	295	295	295	295	490
層重量 (kN)	4022	4176	4282	4282	4311	4330	4426	4446	4486	4550	4724
初期剛性(kN/cm)	7193	9609	10604	11296	12083	13175	14441	15372	16617	18916	27463
第2剛性(kN/cm)	3205	4119	4368	5010	6010	8755	9629	10634	11393	10287	13560
層降伏せん断力(kN)	4089	6237	8296	9891	11267	12862	14122	15025	15824	16389	16869

表-2 解析建物の構造階高,層重量および構造特性





・特徴量3,観測データポイント数600
(=訓練データ 450 + テストデータ 150)
別に検証データ 100 個用意
<ul> <li>・隠れ層2,ノード数8</li> </ul>
∴パラメータ総数=(3*8+8)+(8*8+8)+(8+1)=113
・バッチサイズ:10,epoch 数:60
・活性化関数:隠れ層:'softplus',出力層:適用無し
・損失関数:'mean_squared_error'
・最適化関数:'adam'
・k 分割交差検証(k=4)を採用

均回帰曲面(構造特性~応答量関係)と最終 fold の各データ(即ち,ピンクの訓練データ,緑のテ ストデータ,青の検証データ)を重ねたものであ る.(a)図から概ね全てのデータが回帰曲面近傍 にあることが分かる.なお,個別の fold の回帰曲 面形状は,データが入れ替わるので,互いに微妙 に変化する.

一方, (b), (c) 図は, 夫々訓練とテストデータ

の損失値 (loss) の推移とテストデータの平均絶 対誤差 (MAE) の全 fold 平均値の推移である. (b) 図から損失値の推移は単調減少し,ロスの大 小関係はテストロス<訓練ロスであること,(c) 図から全 fold の MAE 平均値の推移が単調減少 していることが分かる.これより,過剰適合は認 められないと判断できる.なお最終 epoch=60 の MAE 平均値は 0.09454 であった.この値は, 回帰曲面に対する各点縦軸最大応答塑性率の平均 的な差を表し,分類問題の精度に該当する.

図-6(a), (b) は図-5(a) 図を Matlab で表示し 直したものである. 真横から見た (b) 図から, 平 均回帰曲面上に訓練, テストおよび検証データが 概ね綺麗に乗っていることが分かる.

図-7 は、TensorBoard に拠る本例の計算グラ フである. 図中の文字が小さく細部の視認が困難 であるが、そこには実装モデルの定義が示されて いる. 以上より、地震時応答計算の一定割合を学 習結果へ置き換えることは可能と推察される.



TensorBoard x	C You, Mr., Make,	
← → C ① kushiyama-	cc:6006/#graphs&_ignoreYOutliers=false&run=.	<b>☆</b> !
TensorBoard	SCALARS GRAPHS	INACTIVE 👻 🔿 🏚 🔞
Fit to screen  Converlead PNG  Run (1)  Session runs (0)  Upload Choose File Trace inputs Color  Structure Converte Structure Compute Sime Memory Compute Sime Memory Compute Sime Compute	Main Graph · · · · · · · · · · · · · · · · · · ·	Auxiliary Nodes
✓ Close legend.	dersa.3	dense_2
Comptriment of the explanation of the explanat	age 2	dense_1 v man dense_1 v man v m

図-7 TensorBoard による計算グラフの表示

### 5. 結語

現在急速に進化しつつある AI (人工知能) の中 核的な技術である Deep Learning (深層学習)の 概要と本学工学部計算機実習室の機械学習環境に ついて紹介した.大規模な問題を処理するには充 分な設備とは云えないが,機械学習を専攻する データサイエンス系学科の学生でなくとも scikitlearn, TensorFlow のサンプルコードを打ち込み 気軽に体験できる環境が整備されている.サンプ ルコードも GitHub 上に沢山用意されており,ま た個人所有の PC にも上記同様のライブラリを容 易にインストールできることを付記する.

本資料では回帰問題の事例として,Tensor Flowを用いて地震時応答計算の置き換え可能性 を調べ,特徴量の数を減らした特定層集中崩壊型 建物の応答計算に対して適用を見込めることを確 認した.最終目標は,建物の定量的安全性評価の 確率を得るための計算負荷低減である.

今後の課題としては、Matlab で作成した質点 系の動的非線形応答計算プログラムの必要箇所 で、Python3 系で記述された回帰問題処理コード を呼び出し、連結・実装することが挙げられる。

### 参考文献

- John Haugeland, "Artificial Intelligence: The Very Idea, Cambridge", MIT Press, 1985.
- Vladimir N. Vapnik, "Statistical Learning Theory", John Wiley & Sons, Inc., 1998.
- François Chollet, "Deep Learning with Python", Manning Publication, 2017.
   (邦訳)株式会社クイープ 訳, "Python と Keras によ るディープラーニング", マイナビ, 2018.
- 4) Andreas C. Müller and Sarah Guido, "Introduction to Machine Learning with Python", O'REILLY, 2017. (邦訳) 中田秀基 訳, "Python ではじめる機械学習", O'REILLY Japan, 2017.
- 5) Shigeru Kushiyama, "Seismic Damage Estimation of an Actual Reinforced Concrete Structure Using subset MCMC", Open Journal of Civil Engineering, September 2013, Vol.3, No.3.
- 6) S. K. Au and J. L. Beck: "Subset Simulation and its Application to Seismic Risk Based on Dynamics Analysis", Journal of Engineering Mechanics, 2003, pp. 901–917.