



実装して理解する機械学習の手法： ディープラーニング

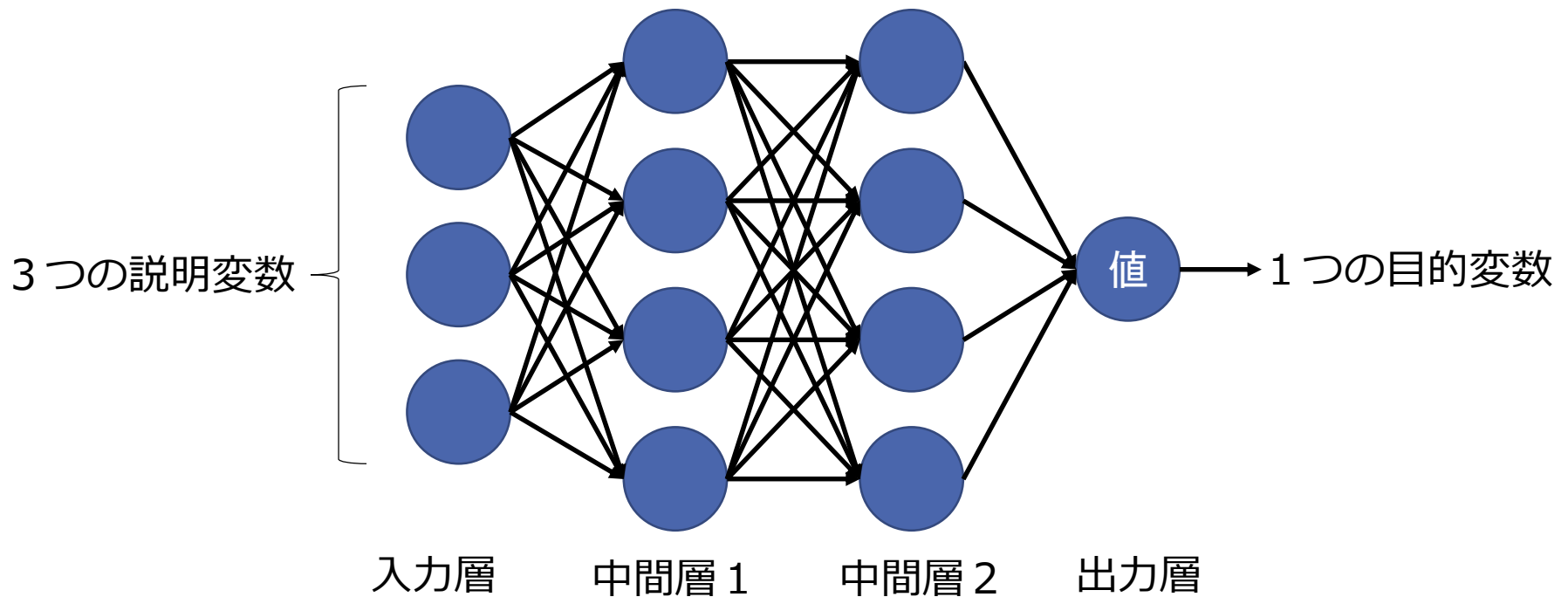
2019.8.9
@福岡システムLSI総合開発センター

我々は技術者です。技術者にとっての基礎とは何でしょう？
私は「**動作原理を理解する事**」であると考えています。

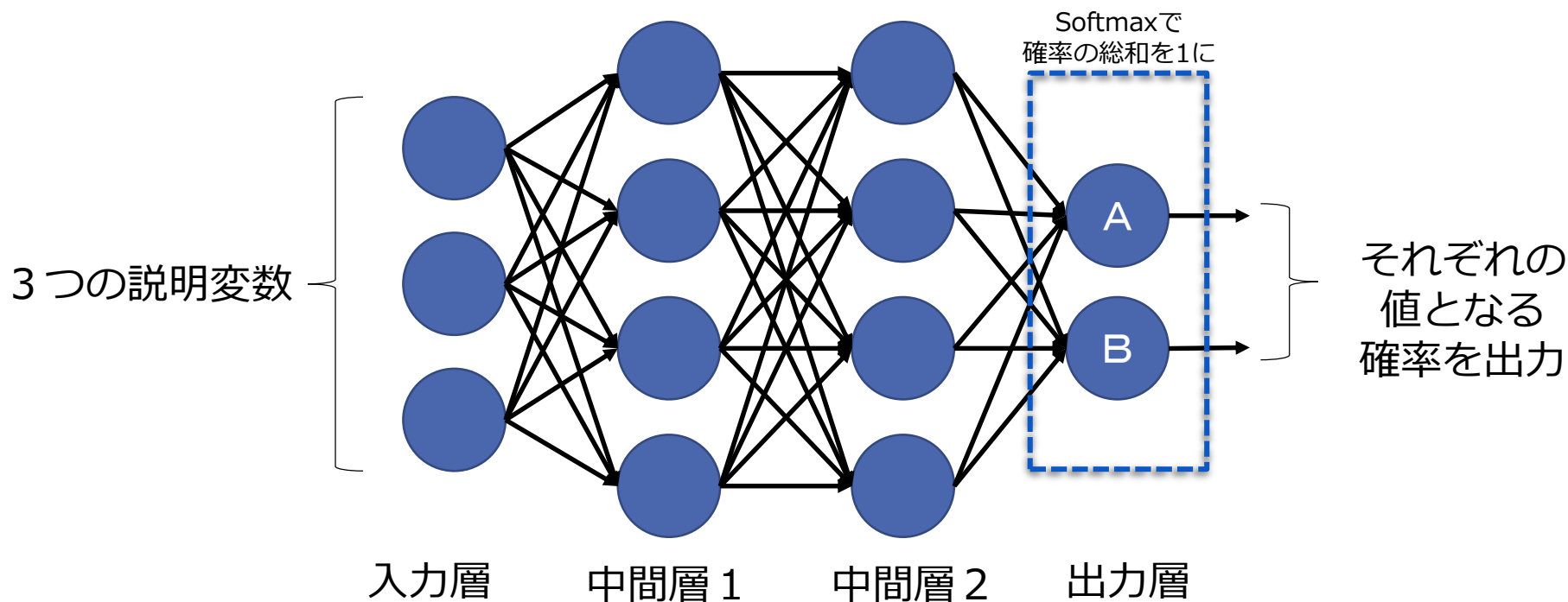
技術者は自分が作り出したものを改修し、さらにいいものを作っていく役目があると思います。このとき、動作原理をしらない部分は改修ができません。

そこで今日は、あえて**1手法の実装を行う**事により、その動作原理の理解の一助にできればと考えております。

出力層のノードを1つとすることで、ディープラーニングをつかって回帰課題を解決する事ができます。



出力層のノードを2つ以上とすることで、ディープラーニングをつかって分類課題を解決することができます。



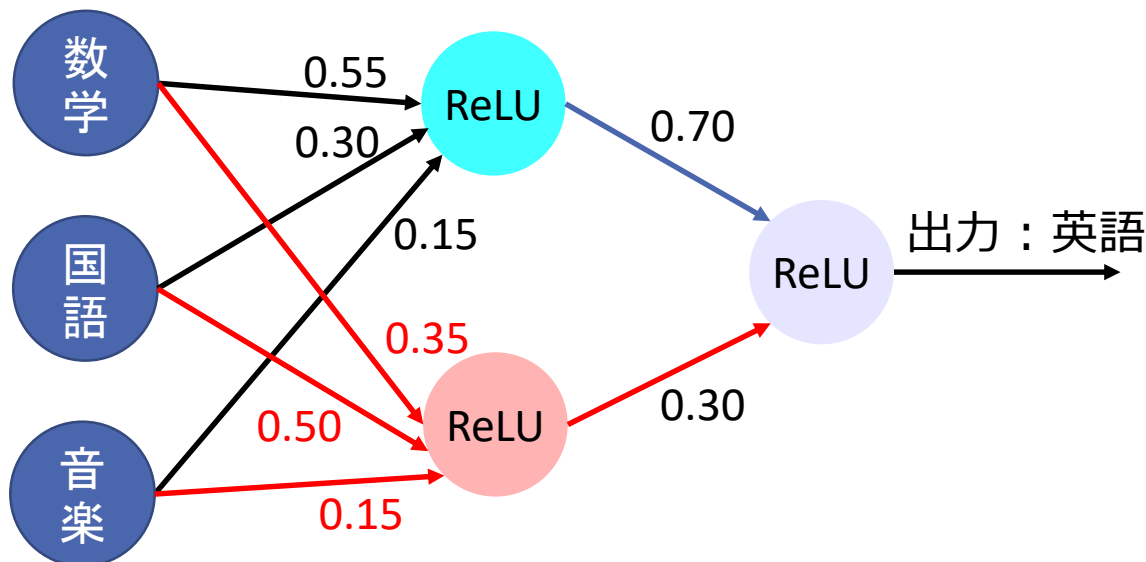
ニューラルネットワークの構築とは…

① ニューロンの配置を定義する。

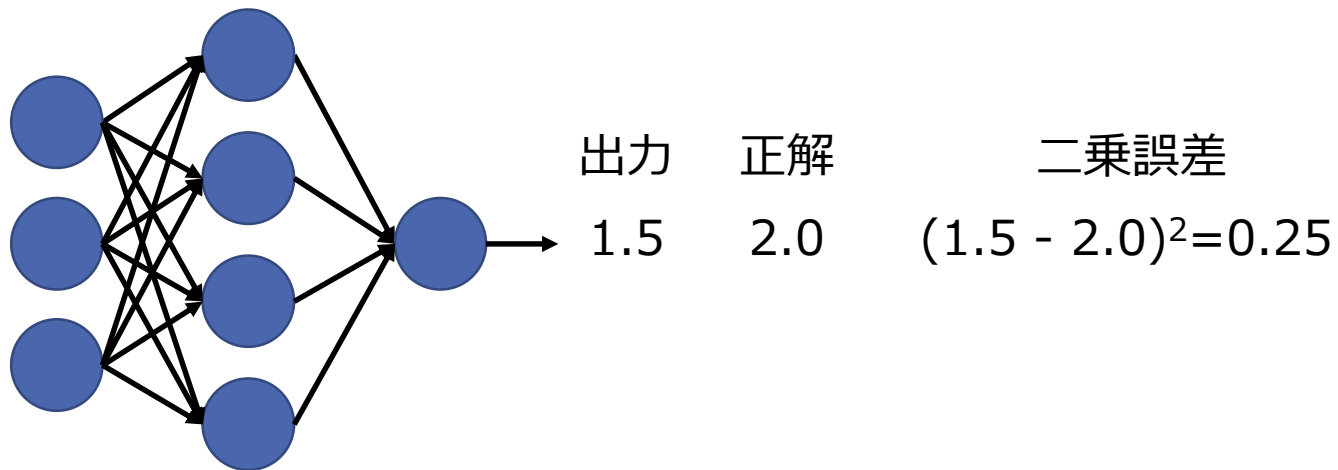
② 予測の精度が高まるように重みを決定する。

…という事にほかなりません。実際には、①は人間が好きに決めます。

②は過去データを教師として、コンピュータに計算させます。

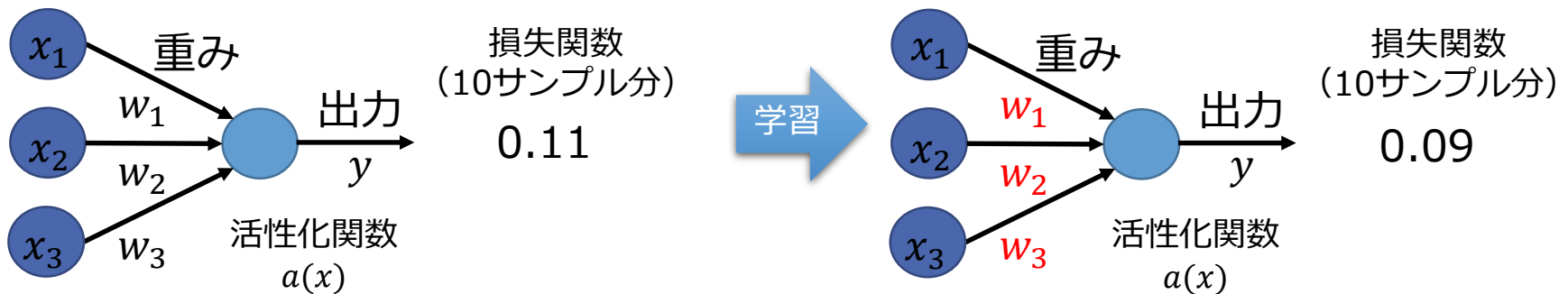


- **損失関数**はニューラルネットの出力が正解とどれだけ異なっているかをあらわす指標です。
- ニューラルネットは損失関数になるべく小さくなるように学習します。
- タスクの種類によっていくつかの種類**の損失関数を使い分けていきます**。回帰課題では**平均二乗誤差**、二値分類課題では**クロスエントロピー**、多クラス分類課題では**カテゴリカルクロスエントロピー**が使われることが多い。



1000個のサンプルをもつ学習用データを使ってニューラルネットの訓練をすると想定する。

1. 適当に重み (w_1 、 w_2 、 w_3 ...) を設定し、1000個のうち10個のサンプルを使って損失関数を計算する。
2. 1よりも損失関数が小さくなるように重みを更新する。



- 計算に用いる**サンプル (上記の場合10個) をミニバッチ**という。
- **1回のパラメータ更新 (学習) を1エポック**という。
- この手順による損失最小化を**「バックプロパゲーション」**という。

今回の課題：MNISTデータセット

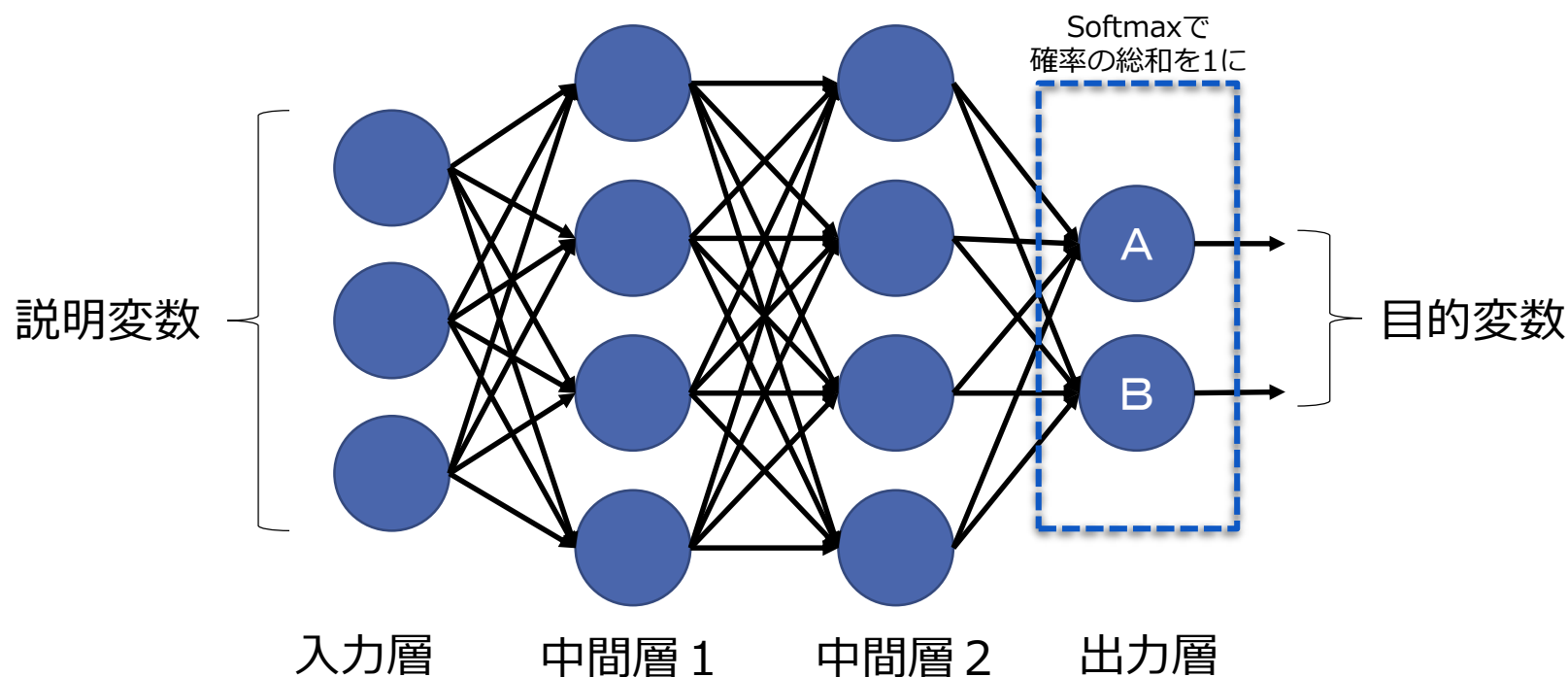
抜粋版

- **MNIST**という手書き数字の画像データセットはしばしば分類課題のベンチマークとして使用される。
- MNISTは0~9のグレースケール手書き数字画像データ。画像サイズは28x28。訓練用画像が6万枚、テスト用画像が1万枚。



実際にディープラーニングを実装してみましょう。

02.DLearn_Using_myClass.ipynb を参照しながら説明します。



※ 本資料は抜粋版となります。上記ソースについては公開をご容赦ください。