

しおん研-ゼロつくゼミ

第4回

今日やること（実装多め）

- ▶ 多次元配列において適用される数値微分アルゴリズムの実装
- ▶ 精度と損失の可視化
- ▶ クラス定義
- ▶ ニューラルネットワークの実装（できなかった分は宿題にします）

学習のフレームワーク（復習）

- ▶ 目的は、推論に最適な（＝もっとも適切な出力を返す）パラメータを見つけること！
- ▶ そのためにはどうすればいい？
 - ある推論に対して、それがどれくらい正しいか評価する関数を定義し、その評価が最大になるようにパラメータを定めれば良いのでは？

学習データ X → 出力データ $Y = f(X, W)$

正解データ t



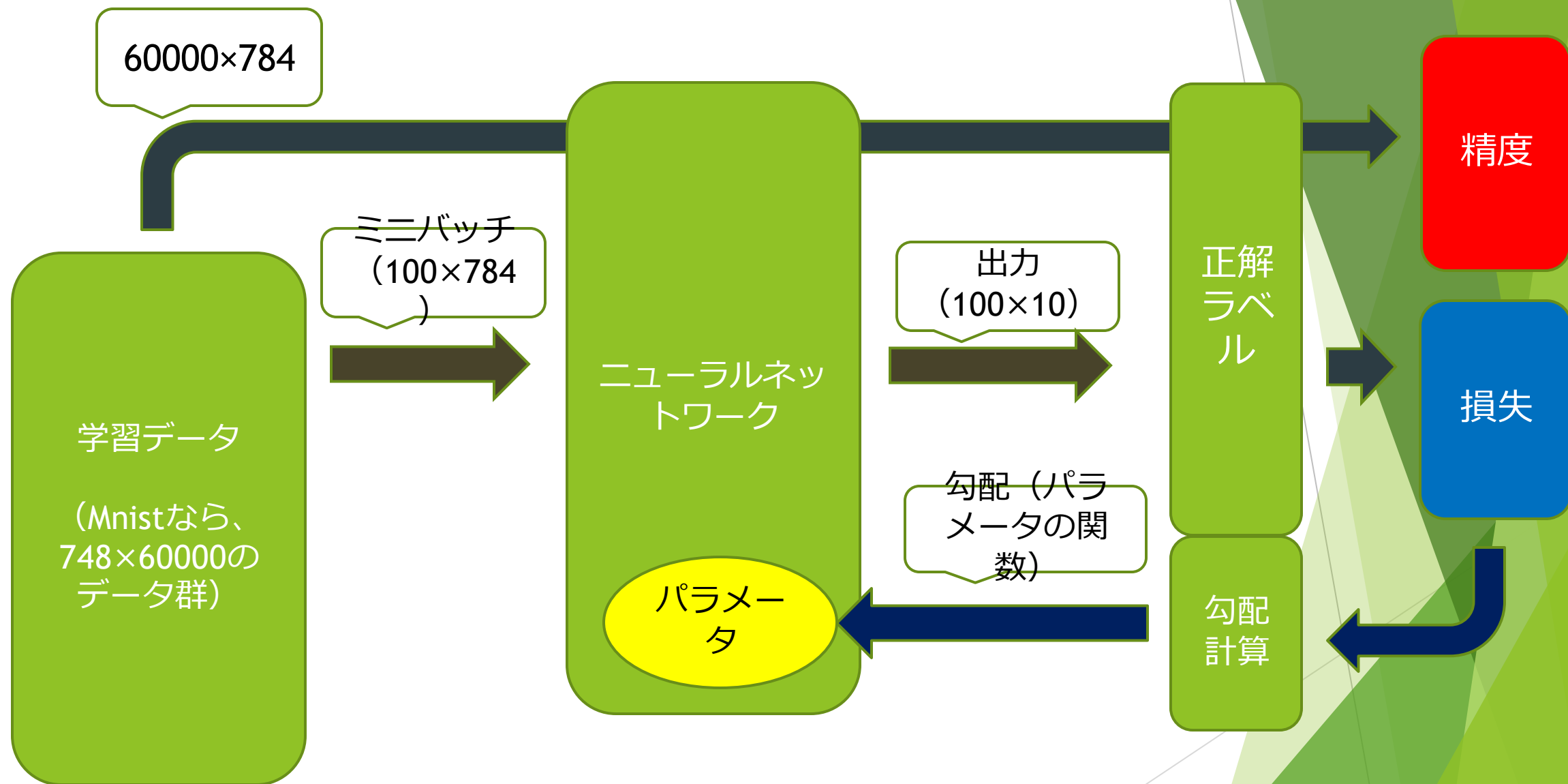
評価関数 $L(W)$
= 損失関数

勾配降下法の3ステップ（復習）

- ▶ 1、 $L(W)$ の勾配（grad）を求める
- ▶ 2、勾配の値に応じてパラメータを更新する
- ▶ 1,2を繰り返す

（質問）ステップ1について

「どんな関数でも、ある値における微分値が必ず求まる」方法とは？

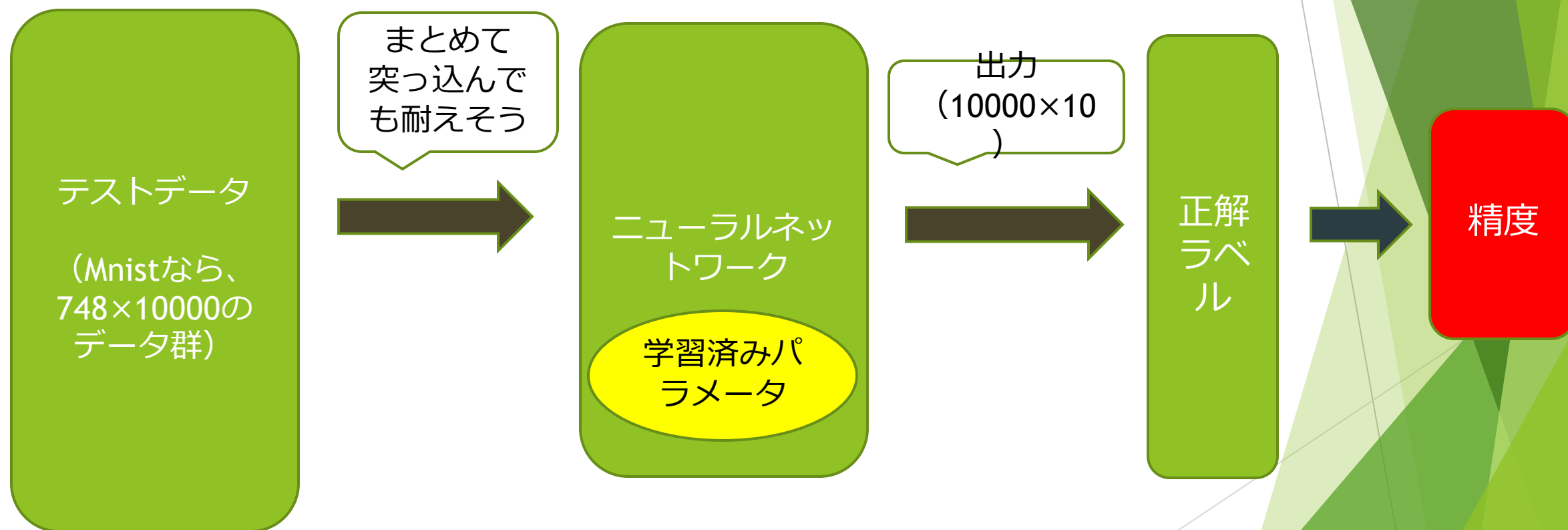


学習のフレームワーク

どうしてバッチ学習を行うのか

- ▶ メモリを減らすため
- ▶ データに「揺らぎ」を持たせることで、局所最適解を脱する確率を上げるため
- ▶ これを確率的勾配降下法（Stochastic Gradient Descent）という！

テストのフレームワーク



多次元配列における勾配計算 (数値微分)

精度と損失の可視化

▶ 学習の経過を可視化したい！

→epochごとに「精度」「損失」をリストに格納

・エポックとは？

ミニバッチによって全訓練データが被りなく抽出されるという仮定のもとで、全訓練データをちょうど1回ずつ学習したと考えられる学習回数。

$\text{Iters_per_epoch} = \text{訓練データの総数} / \text{バッチサイズ}$

▶ 知りたいのは

「訓練データの損失」「訓練データの精度」「テストデータの精度」

▶ (復習) 訓練データとテストデータの役割

クラス定義

- ▶ クラスとは：

「そのクラスに属するオブジェクトについて何ができるか、というのを一つのまとまりとして定義したもの」

- ▶ 例えば・・・

クラス定義の練習



2層ニューラルネットワークの実装 (ネットワークそのものの実装)


- ▶ まず、ニューラルネットワークのクラスを作成してください。

(注意すること)

- ・まず、ハイパーパラメータとして必要なものを確認してください。
- ・インスタンスとしてパラメータ用Dictを作成し、初期値を格納してください。
(中間層ノードは50個と考えてください)
- ・メソッドとしては、
 - 「入力行列から出力を予測する関数」 (predict)
 - 「入力行列と正解ラベルを渡すと損失を返す関数」 (loss)
 - 「入力行列と正解ラベルを渡すと精度を返す関数」 (accuracy)
 - 「入力行列と正解ラベルを渡すと、パラメータについての勾配を計算する関数」 (numerical_gradient)を作成してください。

2層ニューラルネットワークの実装 (学習・可視化)

- ▶ まず、Mnist訓練データ・テストデータをロードしてください。
- ▶ 学習時に必要なハイパーパラメータを定義してください。
今回はepoch = 10、learning_rate = 0.1、batch_size = 100
として計算してください。
- ▶ for文（学習過程）の中身を記述してください。
 - ・ ミニバッチ抽出（今回は100枚）
 - ・ 勾配を計算してください。
 - ・ パラメータを更新してください。
 - ・ epochsごとに、損失及び訓練精度、テスト精度を格納してください。



それでは早速、
やってみようー！