

# 情報リテラシー（第12回 後期）

## 深層学習 –ニューラルネットワークで手書き数字認識–

# 今日のねらい

- 深層学習の基本概念を理解できる
- ニューラルネットワークの仕組みを知る
- 簡単な深層学習モデルを実装できる
- 深層学習の可能性と応用例を知る

# 深層学習とは

## 定義

ニューラルネットワークを多層化した機械学習

## 特徴

- 人間の脳の神経細胞（ニューロン）を模倣
- 層を深くすることで複雑なパターンを学習
- 大量のデータとパラメータで高精度を実現

**深層学習 = ディープラーニング（Deep Learning）**

# 機械学習と深層学習の違い

## 従来の機械学習

- 特徴量を人間が設計
- 例：「顔認識」→「目の位置」「鼻の形」を手動で定義
- データ量が少なくても動作

## 深層学習

- **特徴量を自動で学習**
- 例：「顔認識」→ 画像から自動で重要な特徴を発見
- 大量のデータで真価を発揮

# ニューラルネットワークの基本 構造

入力層 → 隠れ層 → 出力層

## 各要素

- **ノード**: 計算を行う単位
- **重み**: ノード間の結合の強さ（学習で調整する数値）
- **層**: ノードの集まり

## 今日のモデル

入力784 → 隠れ層128 → 隠れ層64 → 出力10

# 活性化関数

## 役割

計算結果を次の層に渡す前に変換する

## ReLU（レルー）

### マイナスの値を0にする

- 隠れ層でよく使われる
- 計算が速い
- 式： $f(x) = \max(0, x)$

## Softmax（ソフトマックス）

### 確率に変換する（合計が1になる）

- 出力層でよく使われる
- 各数字（0-9）の確率を出力

# 損失関数と最適化

## 損失関数

### 予測と正解のズレを数値化

- 予測が正解に近い → 損失が小さい
- 予測が外れている → 損失が大きい
- 学習の目標：損失を小さくする

## 最適化

### 損失を小さくするように重みを調整

- 少しずつ重みを変更
- どう変更すればよいかを自動計算
- 手法：Adam（アダム）など

# 学習の仕組み

## バックプロパゲーション（誤差逆伝播）

間違いを後ろから前に伝えて修正

### 流れ

1. 予測する（前向きに計算）
2. 間違いを見つける（正解と比較）
3. 原因を探る（後ろから前に）
4. 重みを修正（少しずつ調整）
5. 繰り返す

### 結果

繰り返すほど精度が上がる



# 深層学習の応用例：画像認識

## 顔認識

- スマホのロック解除
- 空港の入国審査
- SNSの自動タグ付け

## 医療画像診断

- レントゲン画像から病変を検出
- CT画像からがんを発見
- 診断の補助として活用

## 自動運転

- 歩行者や車両の検出
- 信号や標識の認識

# 深層学習の他の応用例

## 自然言語処理

- ChatGPT、Claude（会話AI）
- 機械翻訳（Google翻訳）
- 文章生成

## 音声認識

- Siri
- 音声入力
- リアルタイム翻訳

## 画像生成

- Stable Diffusion
- テキストから画像を生成

# 今日の実習：MNIST手書き数字認識

## MNIST（エムニスト）

手書き数字（0-9）の画像データセット

### データ

- 訓練データ：60,000枚
- テストデータ：10,000枚
- 画像サイズ：28×28ピクセル

### 目標

ニューラルネットワークで手書き数字を認識

# 今日のColabの流れ

**セル1:** データ読み込みと可視化

**セル2:** モデル構築

**セル3:** 学習と学習曲線

**セル4:** 予測と評価、混同行列

**セル5:** 正解/不正解の例

# モデルの構造

## 今日作るニューラルネットワーク

入力層: 784ノード (28×28ピクセル)  
↓  
隠れ層1: 128ノード (ReLU)  
↓  
隠れ層2: 64ノード (ReLU)  
↓  
出力層: 10ノード (Softmax、0-9の確率)

**総パラメータ数: 109,386個**

# 学習の様子

## エポック（学習の繰り返し回数）

訓練データ全体を1回学習すること

## 5エポックの学習

Epoch 1/5 → 精度: 約92%  
Epoch 2/5 → 精度: 約96%  
Epoch 3/5 → 精度: 約97%  
Epoch 4/5 → 精度: 約98%  
Epoch 5/5 → 精度: 約98%

学習曲線でエポックごとの改善を確認

# まとめ

## 深層学習の基本

- **ニューラルネットワーク**: 層・ノード・重み
- **活性化関数**: ReLU、Softmax
- **学習**: 間違いを見つけて少しずつ修正

## 今日の体験

- MNISTで手書き数字認識
- 約98%の精度を実現
- 深層学習の威力を実感

## 応用分野

画像認識、自然言語処理、音声認識、画像生成など

### 振り返り:

Teamsに今日の感想を書いてください