

データサイエンス基礎講座
2015秋
【機械学習・実践編】

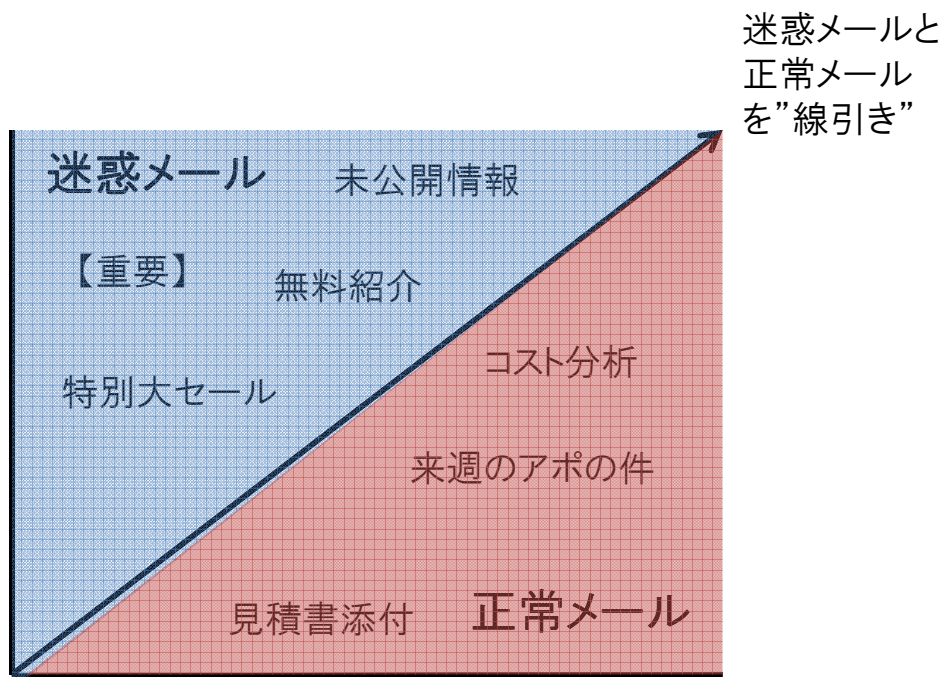
<< 特別編 >>

ニューラルネットワーク
と
ディープラーニング

ニューラルネットとディープラーニング

□はじめに、ニューラルネットワークで解きたい問題

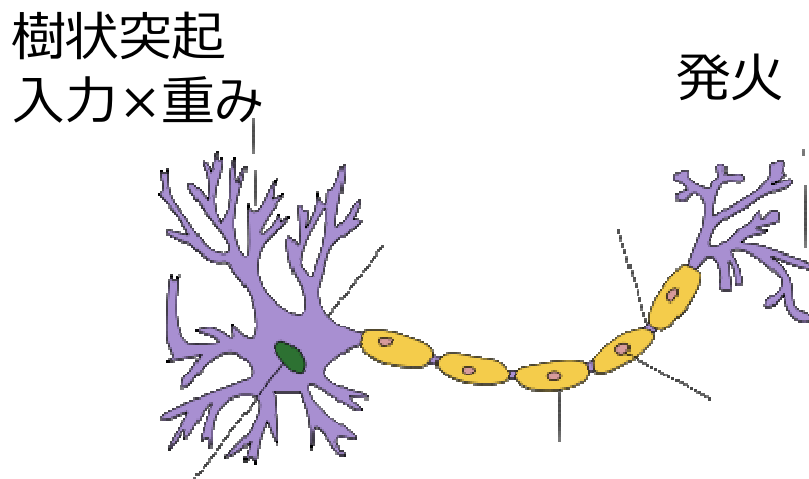
- 分類問題 - ラベルをつける



- 分類問題から発展して、ニューラルネットは畳込みネット（後述）などもあり、画像解析と相性が良い → ディープラーニングも高度な画像認識で利用される

ニューラルネットとディープラーニング

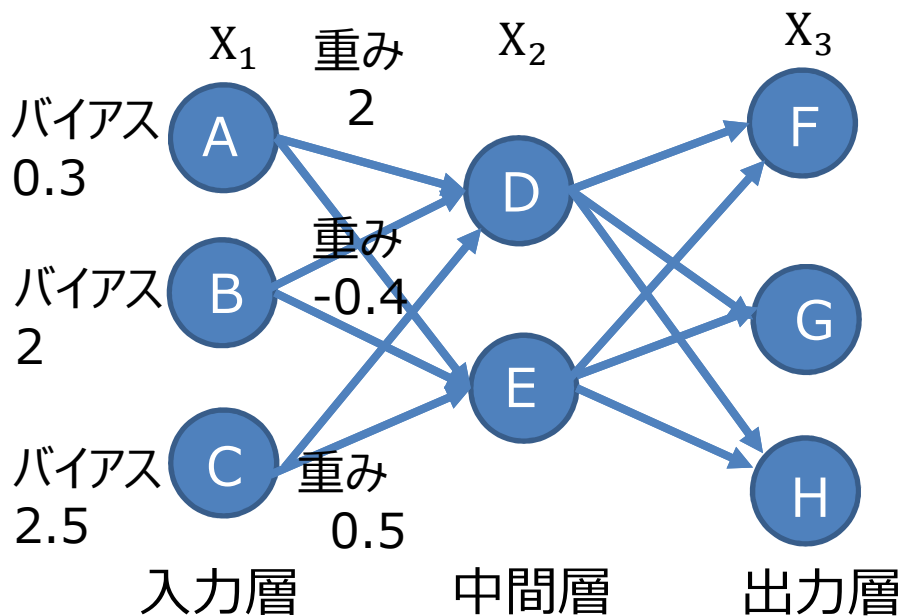
- ニューラルネットのざっくりとした考え方
- 脳の神経細胞（ニューロン）
 - 入力×重みで閾値を越えると発火する



- 入力×重みの仕組みを人工的に表現→
ニューラルネットワーク

ニューラルネットとディープラーニング

- 入力層、中間層（隠れ層）、出力層からなる - **多層パーセプトロン**
- ある層のユニットと次の層のユニットはすべて結合（全結合）
- 各ユニットはバイアスをもち、重み行列をかけて、結合ユニットのバイアスを出力
- 教師あり学習の場合、入力層、出力層が与えられていて、中間層で重みとバイアスを最適化



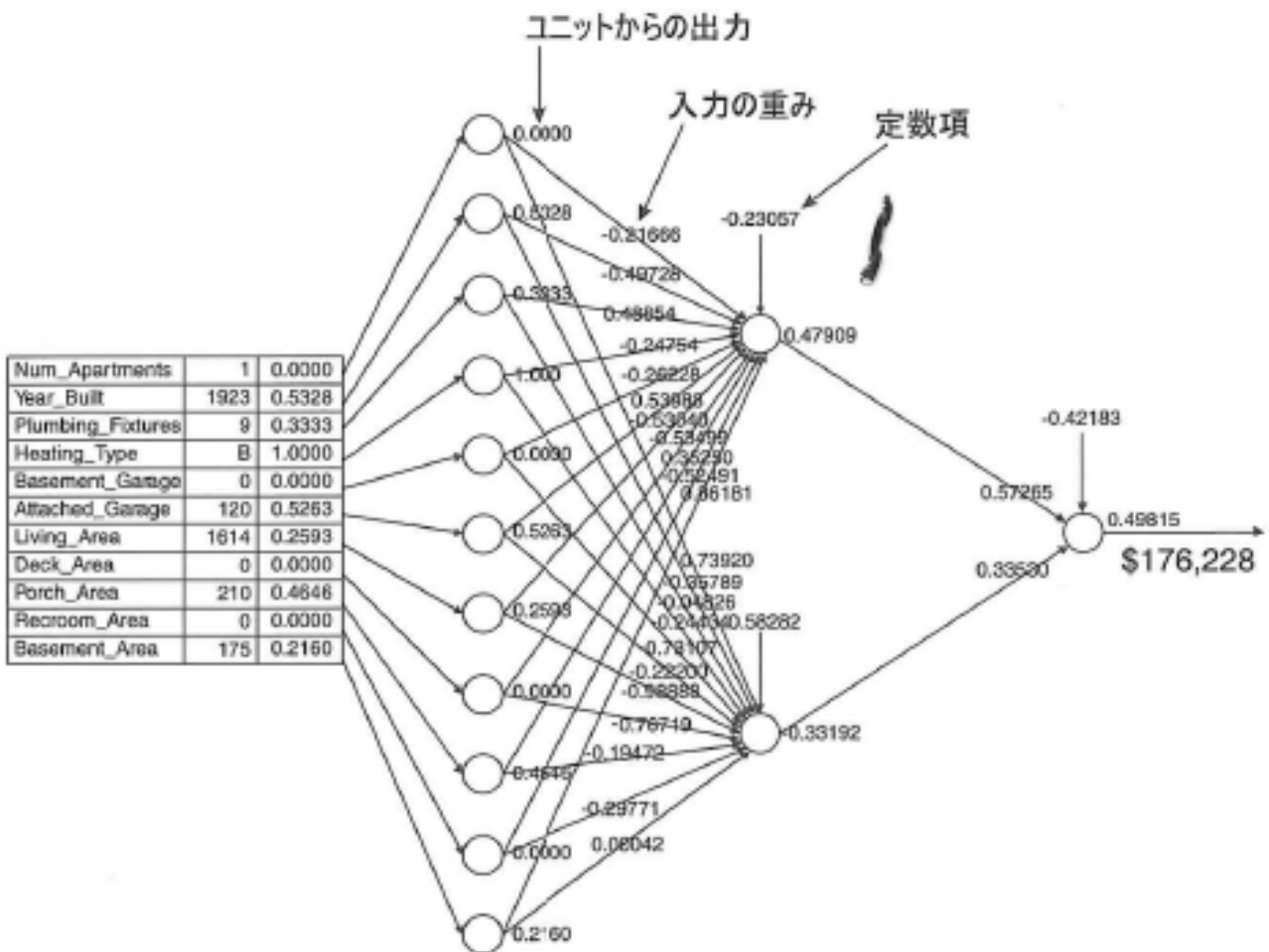
$$D \text{のバイアス} = 0.3 \times 2 + 2 \times -0.4 + 2.5 \times 0.5 = 1.05$$

$$h = f(Wx + b)$$

と表現

ニューラルネットとディープラーニング


□ ニューラルネットの例



Data Mining Techniques: For Marketing, Sales,
and Customer Relationship

ニューラルネットとディープラーニング

□ニューラルネットとディープラーニング

- 教師あり学習、教師なし学習いずれも対応
- 一般的に、中間層が1つのものをニューラルネットワーク、3つ以上の層をディープラーニング（深層学習）と呼ぶ
- おもに利用されるのは、画像認識分野、音声認識、翻訳など用途は広がりつつある
- 画像を入力し、中間層で認識処理をして、画像を認識する
- 例：手書き認識 
 - 入力：手書き画像
 - 出力：0～9までの数字
 - 複数のカテゴリに分類する問題は多クラス分類と呼ぶ
- とくに、畳込みニューラルネットなど近年進歩が著しく、認識精度も大幅に上昇

ニューラルネットとディープラーニング

□ディープラーニングの登場人物

- 全結合層 - 重み行列をかけてバイアスを足す
- 活性化関数 - 出力バイアスに非線形関数を適用
- 損失関数 - ネットワーク出力を評価する

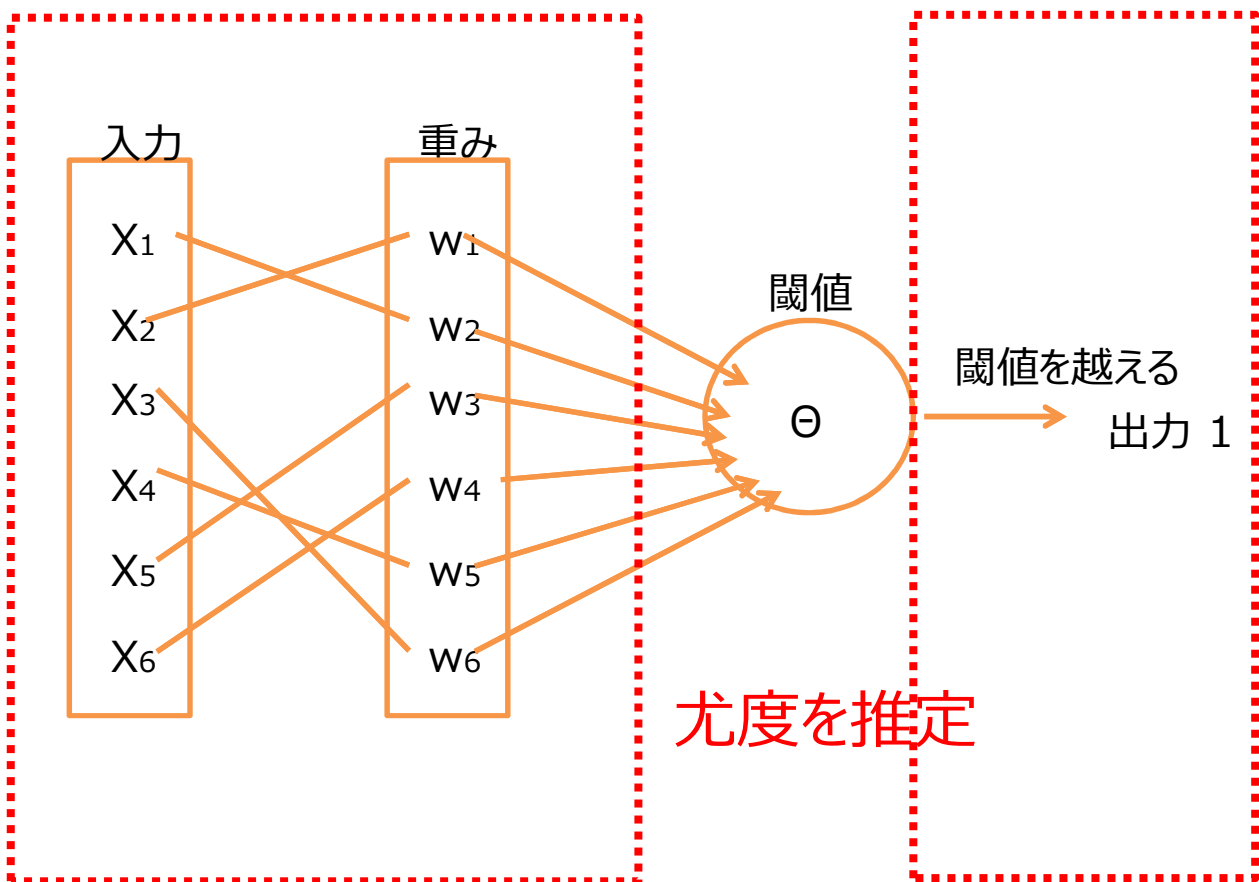
ニューラルネットとディープラーニング

□ 活性化関数

- どうやって入力から出力を求めるか？
- 最尤推定 - ベイズ推定的な考え方

事前分布

事後分布



- 出力は0か1なので、ロジスティクス関数 (シグモイド関数) になる
- 他にもReLU(ランプ関数) などが利用される

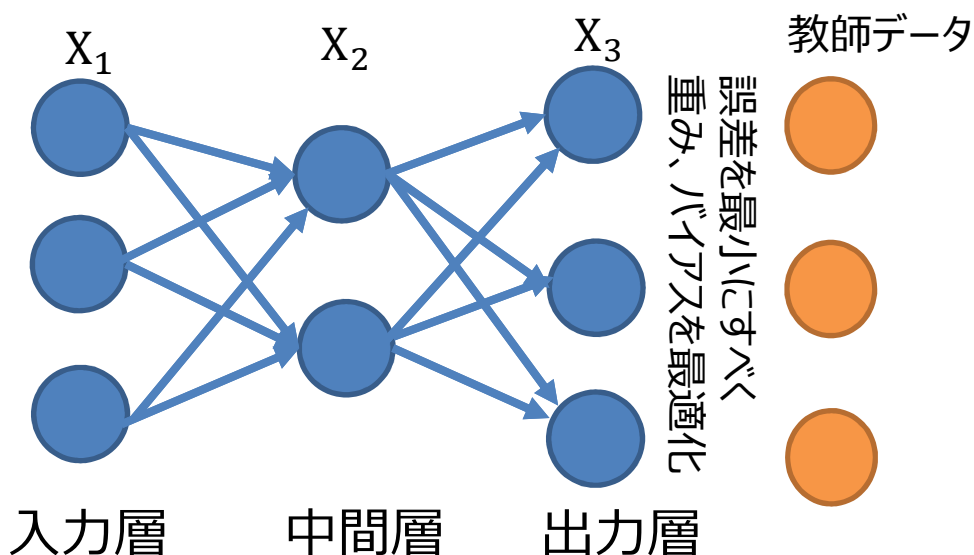
ニューラルネットとディープラーニング

□ 損失関数・誤差関数

- 出力した多層パーセプトロンによる出力と正解とどれくらい誤差があるかを返す関数
- 誤差を最小にするように学習する
 - 二乗誤差：モデルと出力の残差の二乗
 - エントロピー誤差：クラス分類で利用。確信の度合いを表示。ソフトマックス関数と一緒に利用される。

□ どうやって誤差を最小にするか？

- 勾配降下法
- 確率的勾配降下法
- 誤差逆伝播法



ニューラルネットとディープラーニング

□ 勾配降下法

- 目的関数：正解とモデルの誤差を最も小さくする
- 最も小さな点（極小点）をもとめるアプローチ
- どれだけ小さくするかは、学習率 $\epsilon \nabla E$ というパラメータで調整
- 損失関数のすべての勾配を計算する
→ **バッチ最適化**

$$w^{(t+1)} = w^{(t)} - \epsilon \nabla E$$

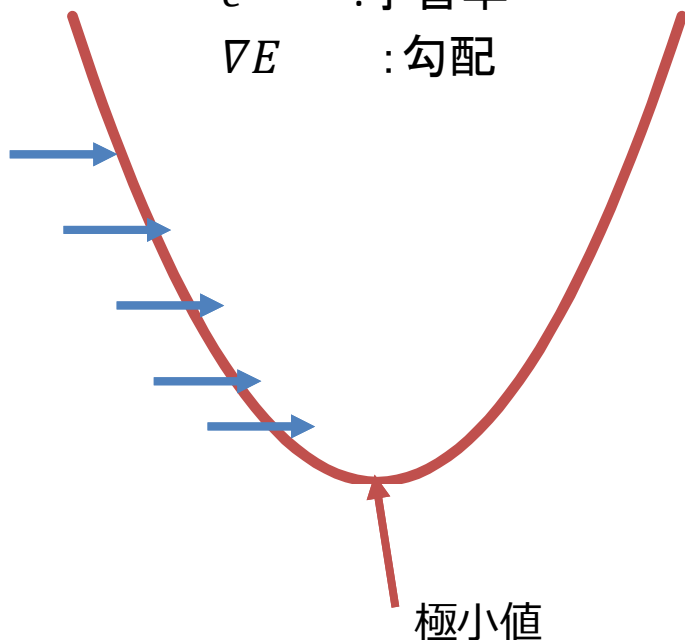
$w^{(t+1)}$: t+1 の重み

$w^{(t)}$: tの重み

ϵ : 学習率

∇E : 勾配

ある地点での
勾配(傾き)を求める
傾きが正であれば
増加関数なので、
値を小さくして極小値
を求める



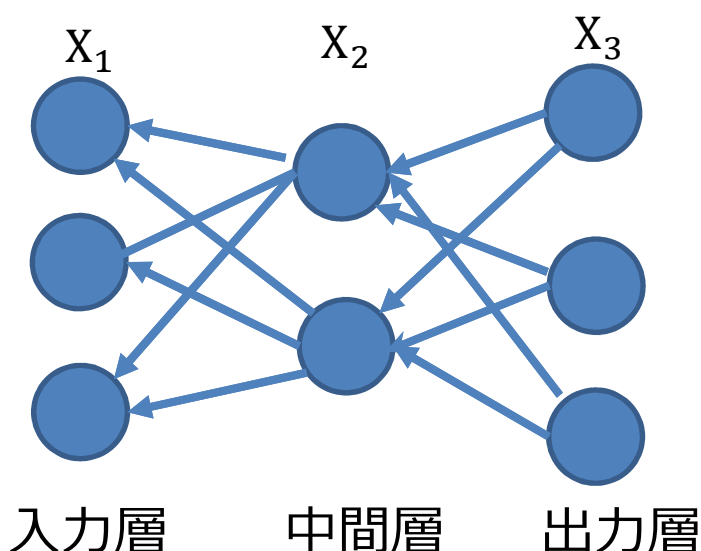
ニューラルネットとディープラーニング

- 確率的勾配降下法 (SGD: Stochastic Gradient Descent)
- 教師データすべてについて勾配降下法でバッチ最適化すると、計算量・コストかかる
- 実際には、一部の教師データの一部を使って勾配を計算しても正しく最適化できる
- ランダムに一部のデータ(ミニバッチ)を選んで勾配を計算 → 確率的勾配降下法 (SGD)

ニューラルネットとディープラーニング

□ 誤差逆伝播法

- 多層パーセプトロンにおいて、勾配を求めるアルゴリズムとして誤差逆伝播法を利用するケースが一般的
- 勾配降下法ならびに確率的勾配降下法いずれも利用
- 通常は、入力層→中間層→出力層と計算が進む（順伝播ネットワーク）
- 誤差逆伝播法（Error Backpropagation）では、出力側から入力側に向かって逆に向かって勾配を求める
- 具体的には、合成関数の微分（連鎖率）を利用
- 多層パーセプトロンは、順伝播、逆伝播いずれも可能



$$\frac{df}{dx} = \frac{df}{dg} \cdot \frac{dg}{dx}$$

連鎖率:
 f, g が微分可能
であれば、合成関数
は分離可能

ニューラルネットとディープラーニング

□ディープラーニングの最適化

- クラウドの利用による計算リソースの増加にくわえて、ここ数年で新しい最適化手法が続々と登場
- こうした最適化手法が、ディープラーニングの性能向上に寄与
- モーメンタム法：勾配を直接利用するのではなく、勾配の移動平均を使うアルゴリズム
- Adam：学習率を自動的に調整するアルゴリズム
- ドロップアウト：多層パーセプトロンをランダムに無効化して、有効なユニットからなる部分ネットワークを利用する手法。パラメータの間の依存関係を取り除く。

ニューラルネットとディープラーニング

□ 畳込みネット(Convolution Neural Network: CNN)

- 画像認識に应用される順伝播型ネットワーク
- これまでの例では全結合であったものの、畳込みネットでは、特定のユニットのみが結合する特別の層をもつ
- 生物の脳の視角野に関する神経科学の知見をヒントに開発
- 外界から眼に取り込んだ光は網膜で電気信号に変換、その後、脳の視角野に伝達
- 神経細胞のうち、特定のパターンに反応することが知られている
- 特定のパターンに反応するニューロンをつくる（おばあさん細胞）
- 視覚野を畳込み層、プーリング層を交互に配置（20層くらい）
- 画像から物体のカテゴリを認識する物体カテゴリ認識において最も重要な技術

ニューラルネットとディープラーニング

□ 入力（画像） → 出力（カテゴリラベル）

□ 畳込み層

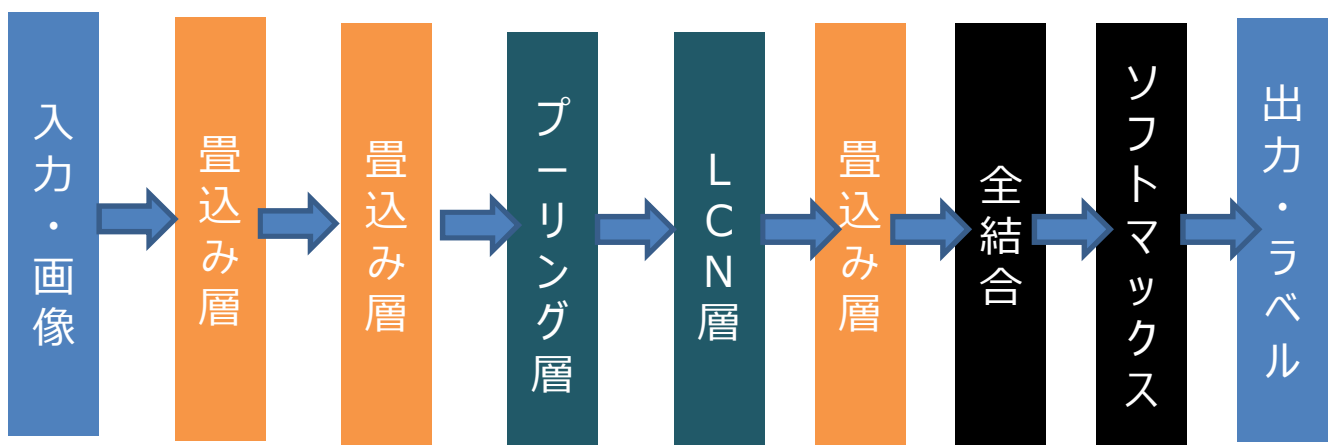
- フィルタと呼ばれる小さな画像を通して特徴抽出をおこなう

□ プーリング層

- 画像の解像度を下げる
- 最終的には、カテゴリ数（例：10個）にしないといけないので、次元を落とす必要がある

□ LCN層：

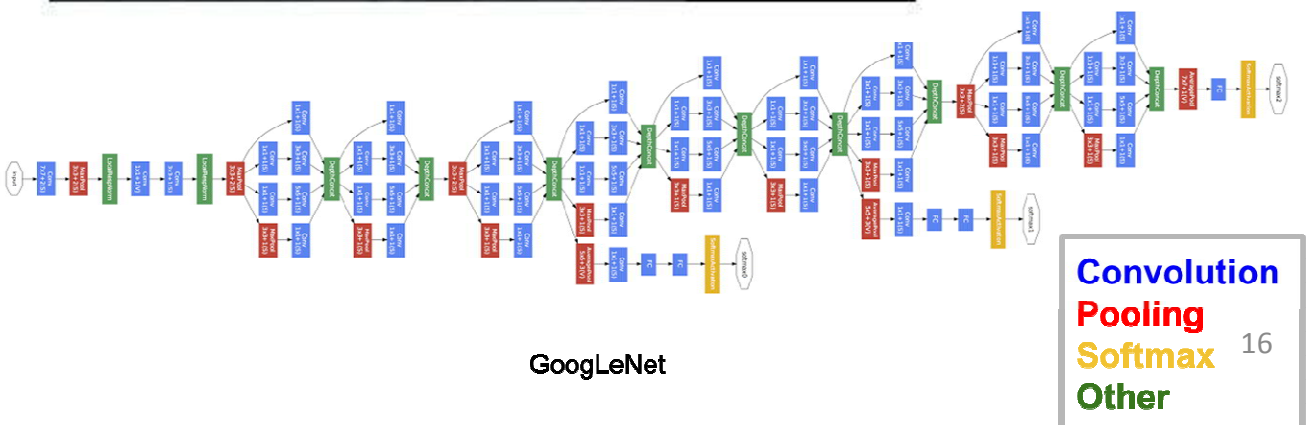
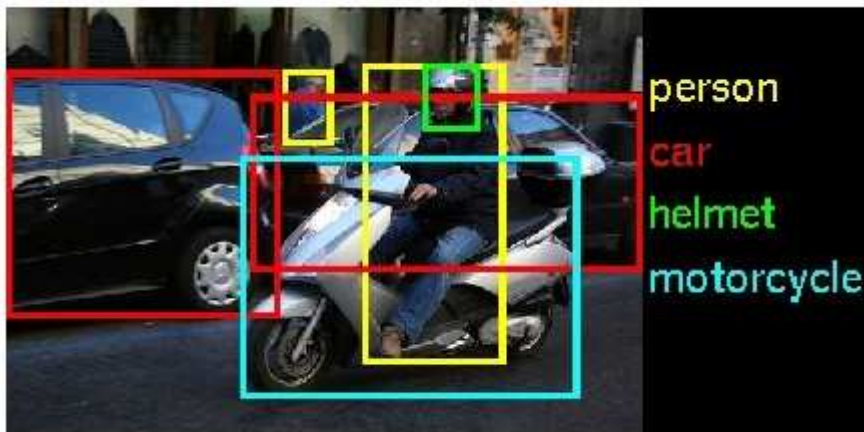
- 局所コントラスト正規化
- 画像の濃淡を正規化する



ニューラルネットとディープラーニング

□ 畳込みネット

- どのように畳込み層、プーリング層を重ねるかは設計次第
- ただし、カテゴリを出力するので、ソフトマックス層は必要
- GoogLeNetの場合
 - 画像認識コンテスト ILSVRC2014
 - 200のうち142で最高の検出性能
 - 誤検知率は6.7%
 - 22層の畳込みネット
 - 全結合はなし



ニューラルネットとディープラーニング

□ディープラーニングの実装

名称	開発元	開発言語	特徴
Caffee	UCバークレー	Protocol Buffer (Python)	単一GPUで最高速度、コミュニティが充実
Torch7	ニューヨーク大学等	Lua/C++	企業利用が多い、Luaスクリプトで記述
Pylearn2	LISA Lab	Python	Theanoという自動微分フレームワークをもとにPylearn2で設定
Cuda-convnet2	Alex Krizhevsky	C++	マルチGPU下での並列実行に注力
Chainer	Preferred Networks	Python	後発フレームワーク、柔軟性重視

- どのフレームワークも基本的にはGPU (Graphics Processing Unit)利用が前提
- Rは、あまり利用されない
- 全結合・勾配降下・誤差逆伝播などいずれも計算量が膨大
- 畳込みネットのフィルタ、プーリングは行列計算、GPUは行列計算が得意

ニューラルネットとディープラーニング

□ Chainerのインストール (Windows編)

- 開発言語 Pythonのインストール
- <http://www.python.jp/>
- Python 3.4.3 → ダウンロード
- File → Windowsx86MSIInstaller
をクリック&ダウンロード

Python 3.4.3
» 日本語ドキュメント
» ダウンロード



ダウンロードをクリック

Windows help file	Windows	
Windows x86-64 MSI installer	Windows	for AMD64/EM64T/x64, not Itanium processors
Windows x86 MSI installer	Windows	

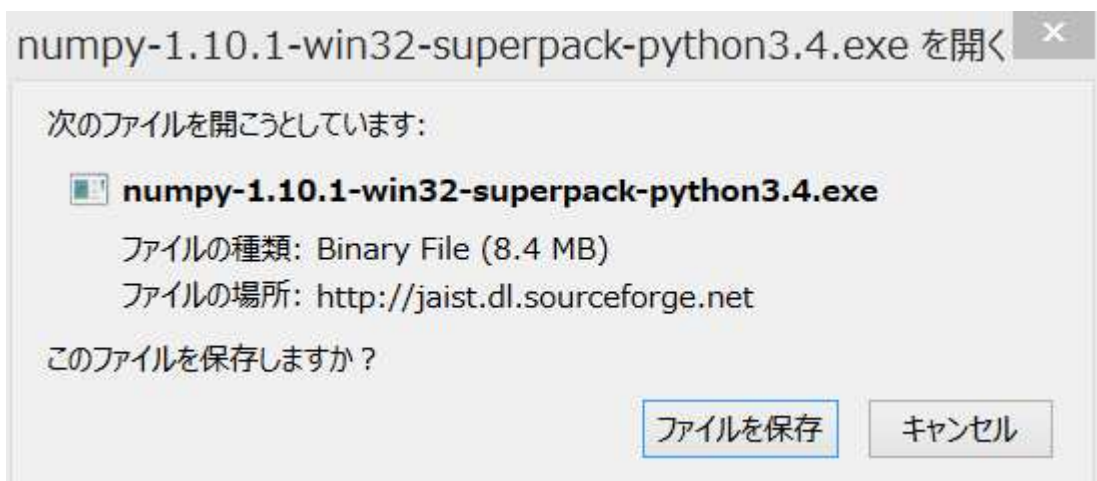


Windows x86 MSI Installerをクリック

ニューラルネットとディープラーニング

□Chainerのインストール

- Numpyのインストール
- 以下にアクセス
- <http://sourceforge.net/projects/numpy/files/NumPy/1.10.1/numpy-1.10.1-win32-superpack-python3.4.exe/download>
- 自動的にダウンロードが始まるので、インストール



ニューラルネットとディープラーニング

□ さらにディープラーニングを勉強 したい場合



深層学習 (機械学習プロフェッショナルシリーズ)
単行本 (ソフトカバー) - 2015/4/8
岡谷 貴之 (著)

数学的に結構難しく、とっつきにくいものの、
今回取り上げた順伝播、確率的勾配降下法
畳込みネットの数理的な説明。くわえて、
今回取り上げていない制限ボルツマンマシン
(RBM)などもカバー。

【機械学習】ディープラーニング フレームワークChainerを試しながら解説してみる。

<http://qiita.com/kenmatsu4/items/7b8d24d4c5144a686412>

Chainer Tutorial

<http://docs.chainer.org/en/stable/tutorial/index.html>