




Neural Network Consoleによるディープラーニング



ディープラーニングとは

動画視聴

- 動画視聴

ニューラルネットワークコンソール(NNC)を開発したソニーの小林さんの講義動画です。

- ディープラーニングとは？ 1 (38:32)

- 人工知能の歴史とディープラーニング

- ディープラーニングとは？ 2 (27:19)

- SONYにおけるAIへの取り組み
- ディープラーニングを用いたAI開発のポイント

➔ 講義ではニューラルネットワークコンソールを使って分析演習を実施



Neural Network Consoleに触れてみる

第8回講義内容

ディープラーニングで解く問題: 手書き文字の「4」と「9」を見分ける

x:image	y:9
1,28,28 	0
1,28,28 	1
1,28,28 	0
1,28,28 	1
1,28,28 	0

- すでに準備されている手書き文字画像の「4」と「9」の見分ける問題
- シンプルなモデルを使ってNeural Network Consoleの使い方を一通り(データの学習から予測)までやってみる.

Neural Network Consoleクラウド版にサインイン

Chromeブラウザを起動し、Neural Network ConsoleのWebからクラウドにサインイン。

The screenshot shows the website dl.sony.com/ja/ in a Chrome browser. The page features the Sony logo and the title "Neural Network Console". A search bar and navigation links for "サポート" (Support) and "プロジェクト" (Project) are visible. A prominent blue button labeled "サインイン" (Sign In) is highlighted with a red double-headed arrow. A blue callout box points to the address bar with the text "1. 'dl.sony.com/ja/' にアクセス" (Access 'dl.sony.com/ja/'). Another blue callout box points to the "サインイン" button with the text "2. 'サインイン' をクリック" (Click 'Sign In'). The main content area displays a large neural network diagram with various layers and nodes.

1. “dl.sony.com/ja/”にアクセス

2. “サインイン”をクリック

Neural Network Console

ドラッグ&ドロップで
簡単ニューラルネットワーク設計

コーディング無しでディープラーニングを用いた高度なAI開発を実現

または

無料で体験

Neural Network Consoleクラウド版にサインイン

Chromeブラウザを起動し、Neural Network ConsoleのWebからクラウドにサインイン。

Neural Network Console

ドラッグ&ドロップで
簡単ニューラルネットワーク設計

コーディング無しでディープラーニングを用いた高度なAI開発を実現

無料で体験

サインインアカウントの選択

Neural Network Consoleで利用するアカウントを選択してください。
Google、Sonyどちらのアカウントでも同様の機能が利用できます。

Google

Sony

1. “Google”をクリック

AI構築に不可欠なディープラーニングの開発基盤をご提供

詳細を見る

Deep Learningを利用したAI開発にこれから携わる方、
既に携わっている方、こんな悩みはありませんか？

Google にログイン

アカウントの選択

「sony.com」に移動

2. 認証に利用するGoogleアカウント
をクリック

Abcde
abcde@tcu.ac.jp

Abcde Fghijk

別のアカウントを使用

続行するにあたり、Google はあなたの名前、メールアドレス、言語設定、プロフィール写真を sony.com と共有します。このアプリを使用する前に、sony.com の [プライバシー ポリシー](#) と [利用規約](#) をご確認ください。

日本語 ヘルプ プライバシー 規約

Neural Network Consoleクラウド版にサインイン

「ダッシュボード」が表示される

「プロジェクト」
をクリック



The screenshot displays the Neural Network Console interface. On the left is a sidebar menu with the following items: 'ダッシュボード' (highlighted in blue), 'プロジェクト', 'データセット', 'ジョブ履歴', 'サンプルプロジェクト', '公開プロジェクト', '公開API', and 'サービス設定'. The main content area is divided into three sections: '最近のプロジェクト' (Recent Projects), '最近のジョブ' (Recent Jobs), and 'Information' (with an information icon). The 'Information' section is currently empty.

サンプルプロジェクトを選択

サンプルとして用意された画像認識のプロジェクトを選択

Neural Network Console

目的から探す テクニックから探す

Dashboard

Project

Dataset

Job History

Sample Project

Public Project

Lemon Orange Apple

画像認識

1. “Sample Project”を
クリック

2. “画像認識”を
クリック

Neural Network Console

目的から探す テクニックから探す

目的から探す > 画像認識

I Input
Dataset : x 1,28,28

A Affine w b 1

S Sigmoid 1

B BinaryCrossEntropy T.Dataset : y 1

tutorial.basics.01_logistic_regression

7601 55

最もシンプルな1層のニューラルネットワークによる画像の2クラス分類

Cloudで開く Download 解説を読む

3. 左上にある
”tutorial.basics.01_logistic_regression”の
“Cloudで開く”をクリック

サンプルプロジェクトを開く

プロジェクト名を指定し開く。

The screenshot shows the Neural Network Console interface. A dialog box is open, prompting the user to create a new project from a selected sample project. The dialog box contains the following text:

選択したサンプルプロジェクトを元に新しいプロジェクトを作成します。プロジェクト名を入力してください。1~255文字以内で以下の文字は使用できません。
(*, /, :, *, ?, ", <, >, |, ;)

The input field contains the text "tutorial.basics.01_logistic_regression". There are "Cancel" and "OK" buttons at the bottom of the dialog box.

1. 半角英数字でプロジェクト名を入力し"OK"をクリック

The screenshot shows the Neural Network Console interface. The "Project" tab is selected, and a list of projects is displayed. The first project, "tutorial.basics.01_logistic_regression", is highlighted. A callout box points to this project with the following text:

2. 作成されたプロジェクトをクリック

ネットワーク編集画面の確認

ネットワークの編集画面が表示されることを確認。

The screenshot displays the Neural Network Console interface for a project named "tutorial.basics.01_logistic_regression". The interface is divided into several sections:

- Navigation:** Home, EDIT (active), TRAINING, EVALUATION.
- Dataset:** tutorial.basics.01_logistic_regression
- Tools:** DATASET, CONFIG, and various utility icons.
- Components Panel (Left):** Search bar, IO (Input), Loss (SquaredError, HuberLoss, AbsoluteError), and Layer Property.
- Main Canvas:** A diagram of a neural network with four layers:
 - I Input:** Dataset: x, size 1,28,28.
 - A Affine:** Weights (w) and bias (b), size 1.
 - S Sigmoid:** size 1.
 - B BinaryCrossEntropy:** T.Dataset: y, size 1.
- Controller Panel (Right):** Run button, Profile, Train (with Structure Search checkbox), Evaluate, and a list of hardware configurations:

Standard	ABCI
<input checked="" type="radio"/> CPU 10 H Free	x 1
<input type="radio"/> NVIDIA® Tesla® K80 GPU	x 1
<input type="radio"/> NVIDIA® Tesla® V100 GPU	x 1
<input type="radio"/> NVIDIA® Tesla® V100 GPU	x 4
<input type="radio"/> NVIDIA® Tesla® V100 GPU	x 8
<input type="radio"/> NVIDIA® Tesla® V100 GPU+	x 8

At the bottom of the Controller panel, there is a note: "CPUの無料利用枠は残り 10 時間です。GPUのご利用は有料です。" and an "Enter credit card" button.

レイヤー説明: Input

ネットワーク全体の入力層にあたるレイヤーである。

パラメータとして入力データのサイズを入力する必要がある。

指定したデータサイズと入力データのサイズが合わない場合、学習時にエラーが発生する。

データサイズの例

表データ

入力サイズ: 100

個人属性情報 100カラム

予測値

ID	性別	年齢	居住地	収入	...	購入頻度
00001	男性	20代	東京	700万		高
00002	男性	30代	埼玉	600万		低
00003	女性	20代	東京	300万		中
00004	女性	10代	東京	NA		中
00005	男性	20代	千葉	400万		高

RGB画像

入力サイズ: (3, 256, 128)



256

128

白黒画像

入力サイズ: (1, 28, 28)



28

28

レイヤー説明: Affine

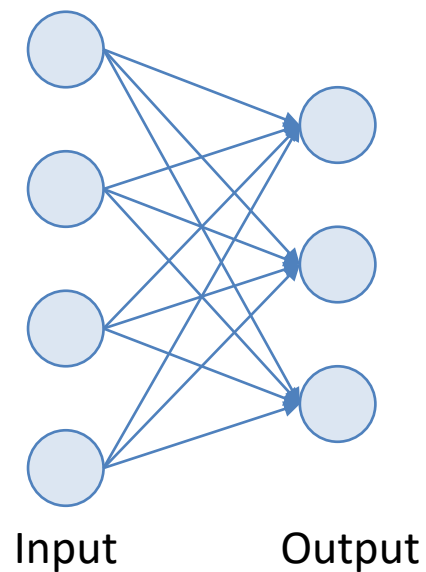
入力層の全ての値を用いた線形変換を行うレイヤーである。

Outputのサイズを任意に指定でき、レイヤーのサイズを自在に調整できる。

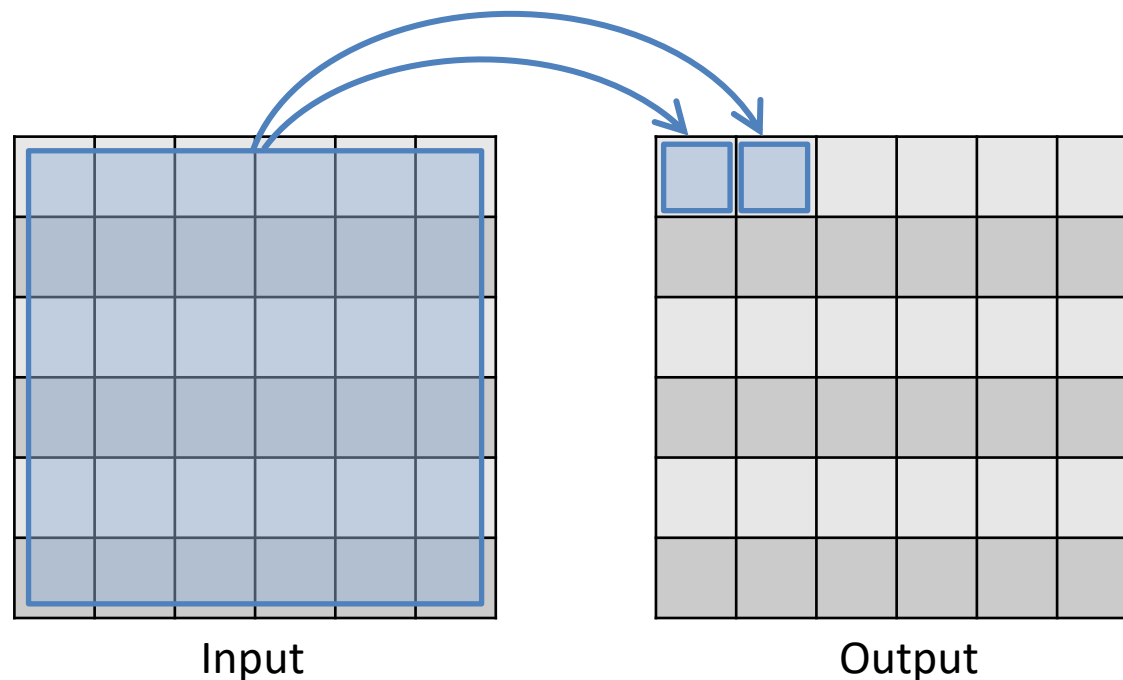
Outputの各セルはInputの全ての値が入力値として利用される。

$$y = Wx + b$$

1次元→1次元の例

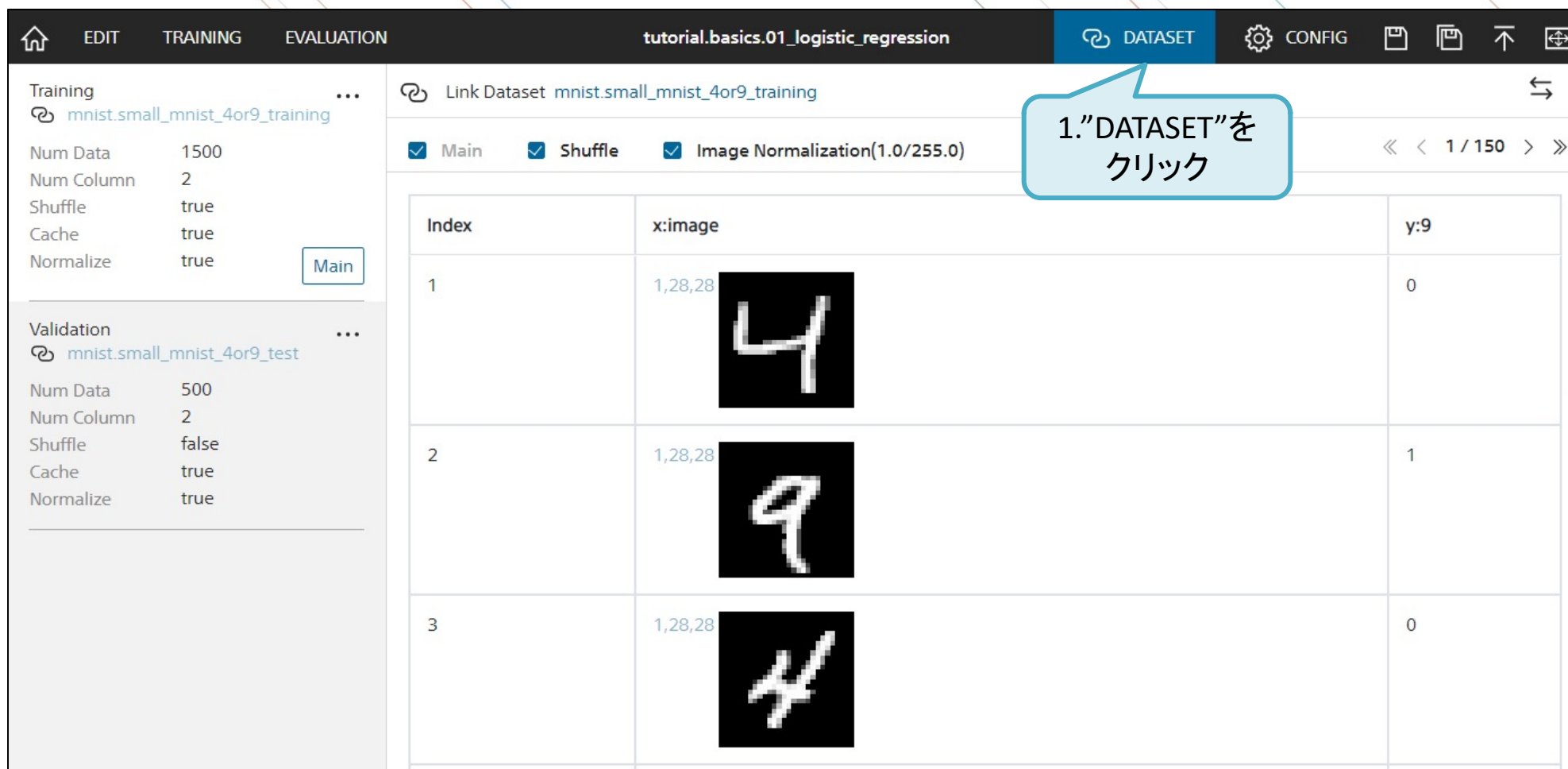


2次元→2次元の例

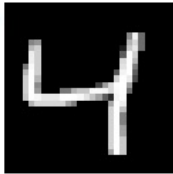




データセット設定画面の確認

設定されているサンプルデータセットの確認。



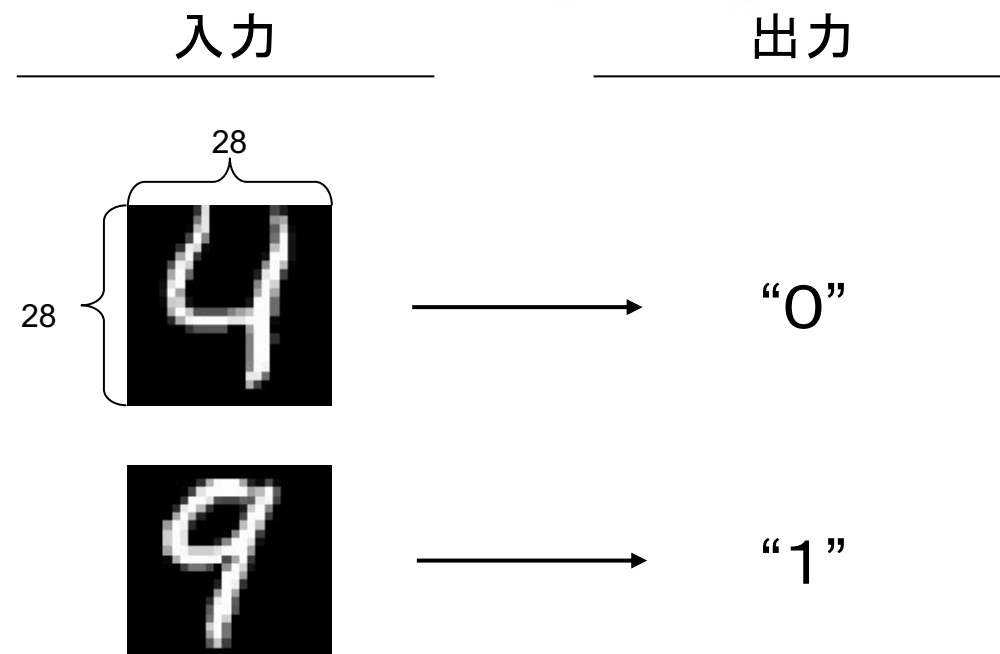
The screenshot shows the Neural Network Console interface for a tutorial project named 'tutorial.basics.01_logistic_regression'. The 'DATASET' tab is selected, showing the configuration for the training dataset 'mnist.small_mnist_4or9_training'. The training dataset has 1500 samples, 2 columns, and is configured with 'Shuffle: true' and 'Image Normalization(1.0/255.0)'. The validation dataset 'mnist.small_mnist_4or9_test' has 500 samples, 2 columns, and is configured with 'Shuffle: false' and 'Image Normalization: true'. A table displays the first three samples of the training dataset, showing handwritten digits 4, 9, and 4 with their corresponding indices and labels (0, 1, 0).

Index	x:image	y:9
1	1,28,28 	0
2	1,28,28 	1
3	1,28,28 	0

データセットの説明

0～9の手書き文字画像から数字を判別する。ハンズオンでは4と9のみをピックアップしている。
アメリカ国勢調査局が収集した手書き文字データから、数字を抽出したデータ(MNIST)である。

MNISTデータセット(手書き数字認識)



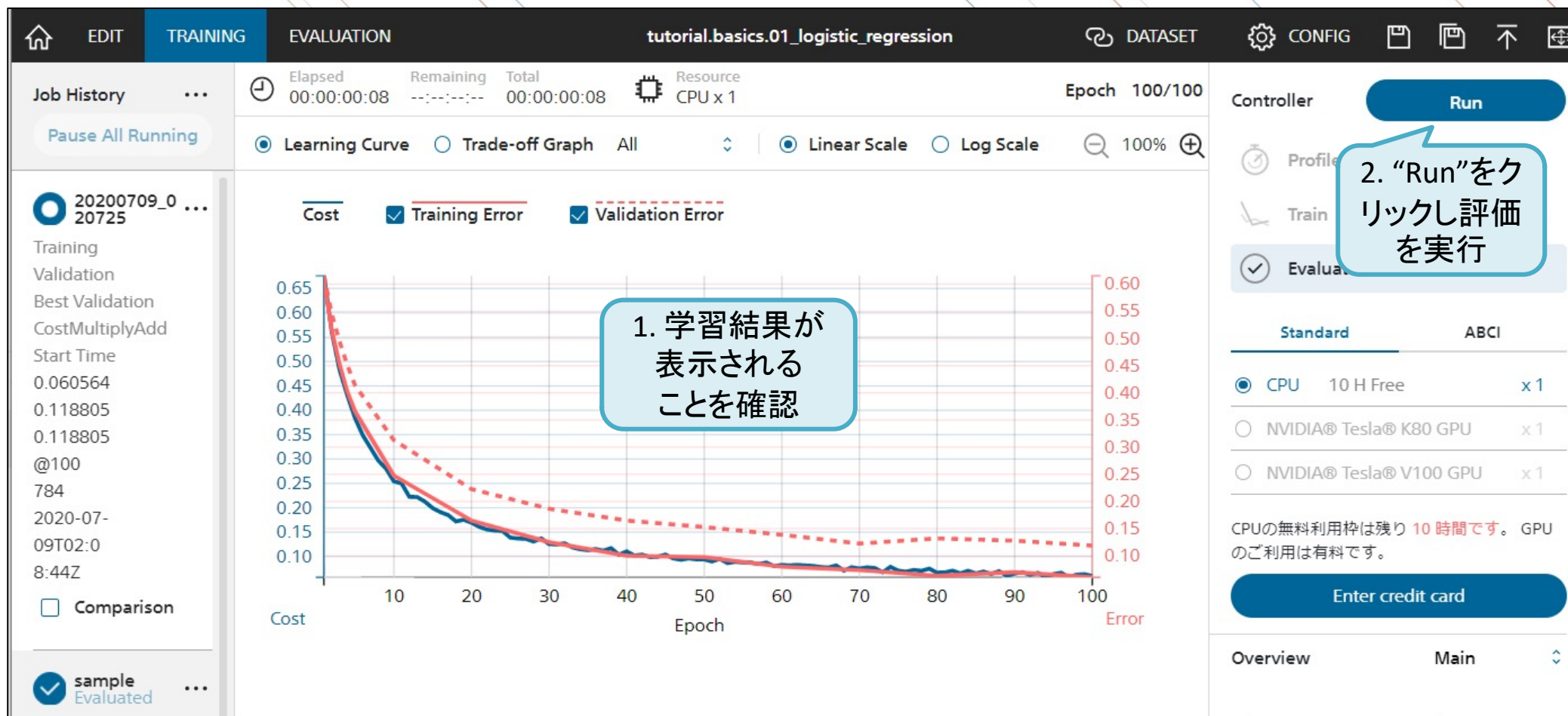
学習の実行

”Run”をクリックするとクラウド上の計算用インスタンスが起動し学習を開始。

The screenshot displays the Neural Network Console interface for a tutorial project named "tutorial.basics.01_logistic_regression". The main workspace shows a neural network architecture with four layers: "Input" (Dataset: x, size 1,28,28), "Affine" (weights w, bias b), "Sigmoid", and "BinaryCrossEntropy" (Target Dataset: y, size 1). The right-hand panel, titled "Controller", features a prominent blue "Run" button. A light blue callout box with a speech bubble points to this button, containing the text: "1. 'Run'をクリックし学習を実行". Below the "Run" button, there are options for "Standard" and "ABC1" configurations, with "CPU 10 H Free x 1" selected. Other options include various NVIDIA GPU configurations. At the bottom of the controller panel, a note states: "CPUの無料利用枠は残り 10 時間です。GPUのご利用は有料です。" and an "Enter credit card" button is visible.

学習結果の表示と評価の実行

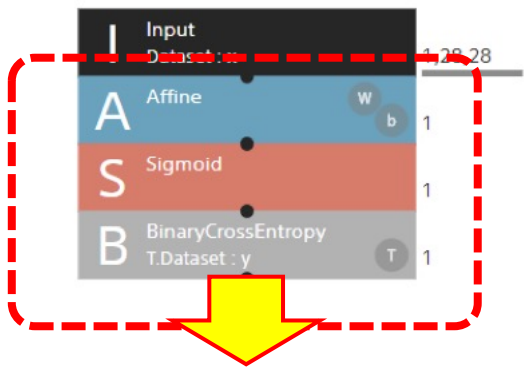
学習結果が表示されることを確認し、評価を行うため再度”Run”をクリック。



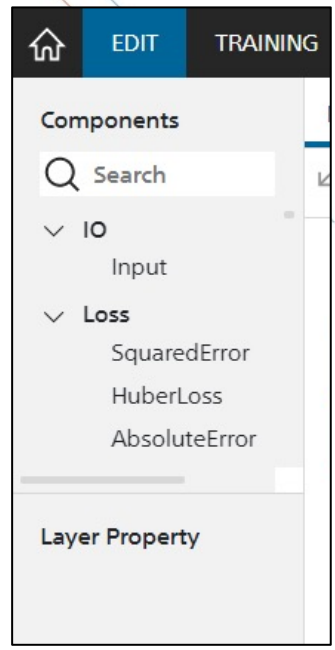
CNNの実装

Convolution層とAffine-ReLU層を「Input」と「Affine」の間に挿入してCNNを実装してみる。

① [ASB]の周りをドラッグで選択

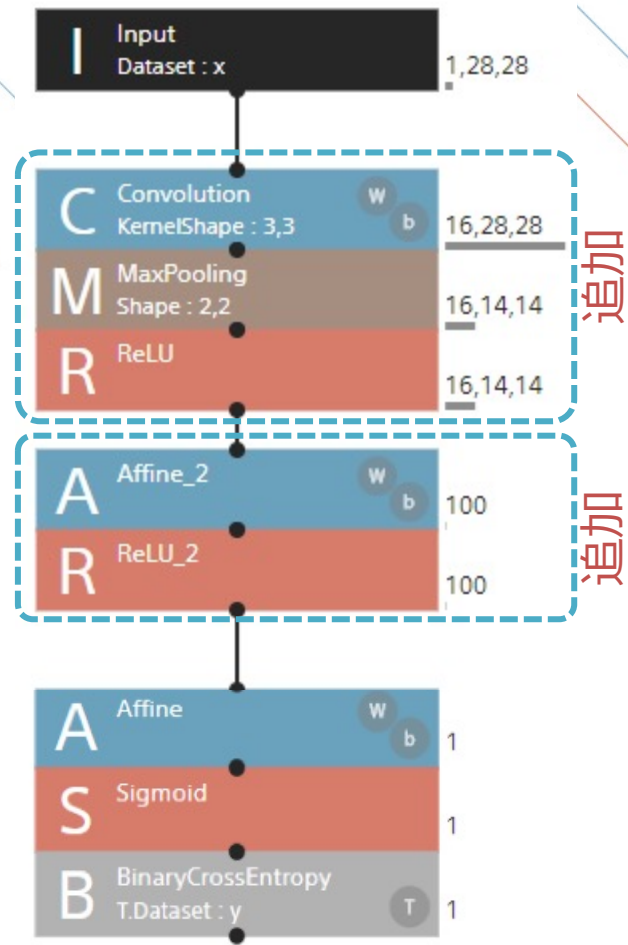


② [ASB]のセットをマウスでドラッグして「Input」と「Affine」の間に隙間を作る

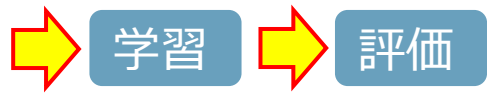


③ 左の「Components」の検索窓に「Convolution」、「MaxPooling」等と入力して、レイヤーをダブルクリック
右図のようになるよう配置していく。

④ レイヤーの●間をマウスでドラッグして接続する。
接続を消したい場合は、消したい接続をクリックして選択し、右上の「アクション」ボタンから「削除」を選択する。



⑤ ネットワークができたら、編集で「実行」→評価で「実行」
→ 混同行列[Confusion Matrix]で精度を確認

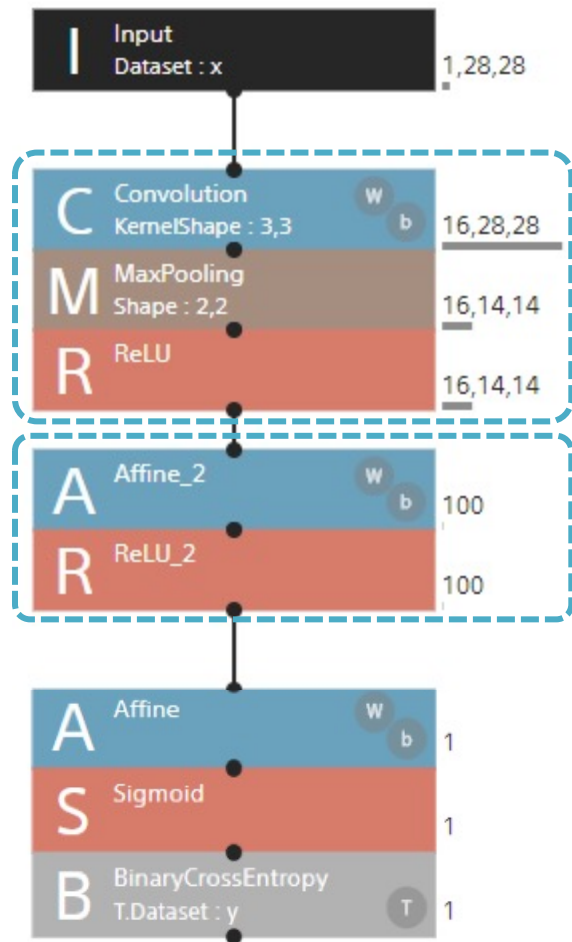


CNNのパラメタの変更

Convolution層とAffine-ReLU層の組み合わせとConvolutionのパラメータ(OutMaps, KernelShape)及び、Affineのサイズを変更し、試行錯誤することができます。

各レイヤー **C Convolution** **A Affine** をクリックすると
左下に設定されている
パラメタが表示されます。

Layer Property	
C Convolution	
Name	Convolution
Input	1,28,28
OutMaps	16
KernelShape	3,3
BorderMode	same
Padding	1,1
Strides	1,1
Dilation	1,1



- Convolution層の数
- Convolutionのパラメータ (OutMaps, KernelShape)

- Affine-ReLU層の数
- Affineのサイズ

※前回の履歴も自動的に残るので、精度を比較してみてください

