

# Neural Network Console クラウド版

## スターターガイド

### -詳細機能説明編-

ソニーネットワークコミュニケーションズ株式会社

# 概要

本ドキュメントはDeep Learning(DL)モデル作成の一連の流れに沿って、Neural Network Console(NNC)の機能を細部まで紹介する内容になっております。すでにDLモデル作成の経験がある方にNNCの機能や使い方をご理解いただくことを想定しております。これからDLをはじめようという方は、ご自身のテーマに近い以下のスターターガイドをご覧ください。

- [スターターガイド-画像分類編-](#)
- [スターターガイド-表データ分類編](#)
- [スターターガイド-時系列データ異常検知編-](#)
- [スターターガイド-物体検出編-](#)
- [スターターガイド-領域抽出編-](#)

# 目次

1

アカウントサインイン

2

データセットのアップロード

3

モデル作成

4

推論実行

# サインインページへの移動

Chromeを利用して、<https://dl.sony.com/ja>に移動します。

ページの右上にある「無料で体験」をクリックします。

サインインするためのアカウントをSONYアカウントまたはGoogleアカウントから選択します。どちらを選択してもこのドキュメントの内容は進めることが可能です。

## サインインページへの移動方法



- ※アカウントがない場合には選択後に作成可能です
- ※どちらのアカウントを選択いただいても、機能に違いはありません

# SONYアカウントでのサインイン

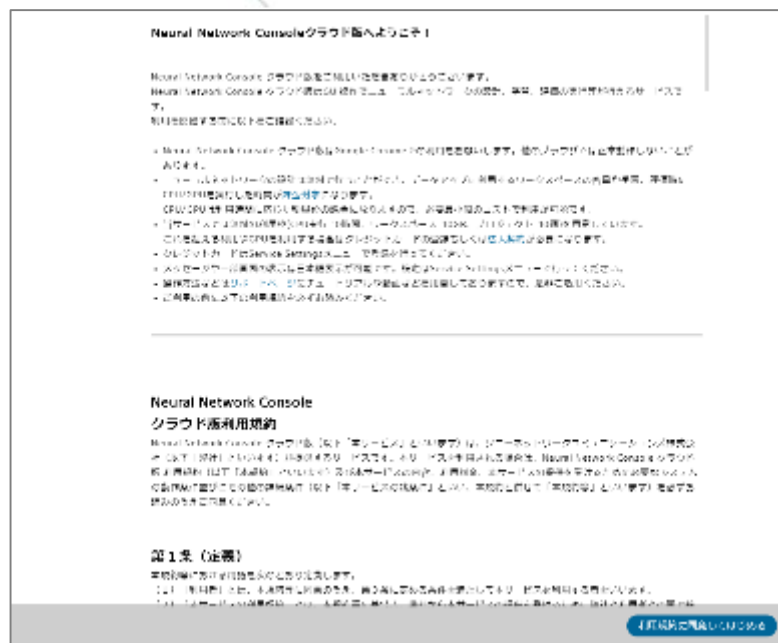
SONYアカウントに登録しているメールアドレス・パスワードを入力し、ログインを行います。  
※アカウントをお持ちでない方は、「新しいアカウントの作成」から新規作成を行ってください。  
詳細は、Appendixの[SONYアカウントの取得方法](#)に記載があります。

## 1. メールアドレスの入力



The screenshot shows the Sony login interface. At the top is the Sony logo. Below it, the text 'サインイン' (Sign In) is centered. Underneath, it says '一つのアカウントで、Sonyグループの複数サービスへアクセスもっと詳しく' (Access multiple Sony Group services with one account, learn more). The 'サインインID' (Sign In ID) field contains 'youraddress@example.com'. Below it is a checkbox labeled 'サインインIDを記憶する' (Remember Sign In ID). The 'パスワード' (Password) field is masked with dots. A blue 'サインイン' (Sign In) button is at the bottom. Below the button, it says 'サインインでお困りですか?' (Having trouble signing in?). At the very bottom is a button for '新しいアカウントの作成' (Create new account).

## 2. 利用規約への同意



The screenshot shows the 'Neural Network Console クラウド版へようこそ！' (Welcome to Neural Network Console Cloud Edition!) page. It contains a detailed introduction to the service and its terms of use. The text is in Japanese and includes a list of terms. At the bottom, there is a blue button labeled '利用規約に同意して利用する' (I agree to the terms of use and use the service).



# Googleアカウントでのサインイン

Googleのメールアドレス・パスワードを入力し、ログインを行います。

## 1. メールアドレスの入力


Google にログイン

---

ログイン

[| sony.com |](#)
に移動

メールアドレスまたは電話番号

xxxxx@gmail.com

[メールアドレスを忘れた場合](#)





音声または画面上のテキストを入力

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。

[アカウントを作成](#)

[次へ](#)

## 2. パスワードの入力

### 3. 利用規約への同意

[illegible]

# Dashboardページ

サインインするとDashboardページが表示されます。

画面上では直近の学習や評価のジョブ実行の進捗を確認できるほか、データ管理やモデル作成などそれぞれのページに移動することができます。

モデル作成のプロジェクト管理ページ  
データセット管理ページ  
過去の学習や評価の履歴管理ページ

NNCのサンプルプロジェクト一覧ページ  
NNCの公開プロジェクト一覧ページ

言語変更や利用料確認などの設定ページ

サポートページへの移動やログアウトなど

Neural Network Console

Dashboard

Project


Dataset

Job History

Sample Project

Public Project




Service Settings

 12345678901234567890

Recent Projects

image_recognition.ILSVRC2012.vgg.vgg-11	2020-06-17
image_recognition.ILSVRC2012.residual netwo...	2020-05-11
classification.iris.iris	2020-05-11

Recent Jobs

 20200501_003901 tutorial.basics.01_logistic_regression Training, Finished	2020-05-01
 20200115_113149 image_recognition.ILSVRC2012.residual netwo...	2020-01-15
 sample tutorial.object_detection.synthetic_image_obje...	2020-01-07

① Information

直近のプロジェクトを表示

直近の学習や評価などの履歴を表示

# ユーザ設定

Service Settingsからユーザ名や言語表示の変更、クレジットカード登録※<sup>1</sup>などが可能です。  
グループ機能を利用される場合には、ユーザ名を設定することで作業者が明確になり便利です※<sup>2</sup>。

The screenshot shows the 'Neural Network Console' interface. On the left is a sidebar menu with options: Dashboard, Project, Dataset, Job History, Sample Project, Public Project, and Service Settings (highlighted with a red box and callout 1). The main content area shows user information: 'No name' (with a 'set' button highlighted by callout 2) and 'ID: 123456789012'. Below this is a language selection section with '言語:' and buttons for 'English' and '日本語' (highlighted by callout 3). Further down are usage statistics for '実行時間' (10H, 0H used) and 'ワークスペース容量' (10GB, 1.3GB used). A section of Japanese text explains the free trial limits. At the bottom of the main area is an 'Enter credit card' button (highlighted by callout 4). The footer shows the user ID '12345678901234567890' and a link to 'Neural Network Consoleクラウド版の退会'.

1. “Service Settings”をクリック

2. “set”をクリックし、ポップアップからユーザ名を変更

3. 言語表示を選択

4. “Enter credit card”をクリックし、ポップアップからクレジットカードを登録

※<sup>1</sup> GPUなど有料のサービスを利用される場合には、クレジットカード登録ないしは法人契約が必要になります。

※<sup>2</sup> すでにグループ登録をされている方は、画面上に表示されるタブを、Personalに変更ください。(本紙には記載がありません)



# 目次

1

アカウントサインイン

2

データセットのアップロード

3

モデル作成

4

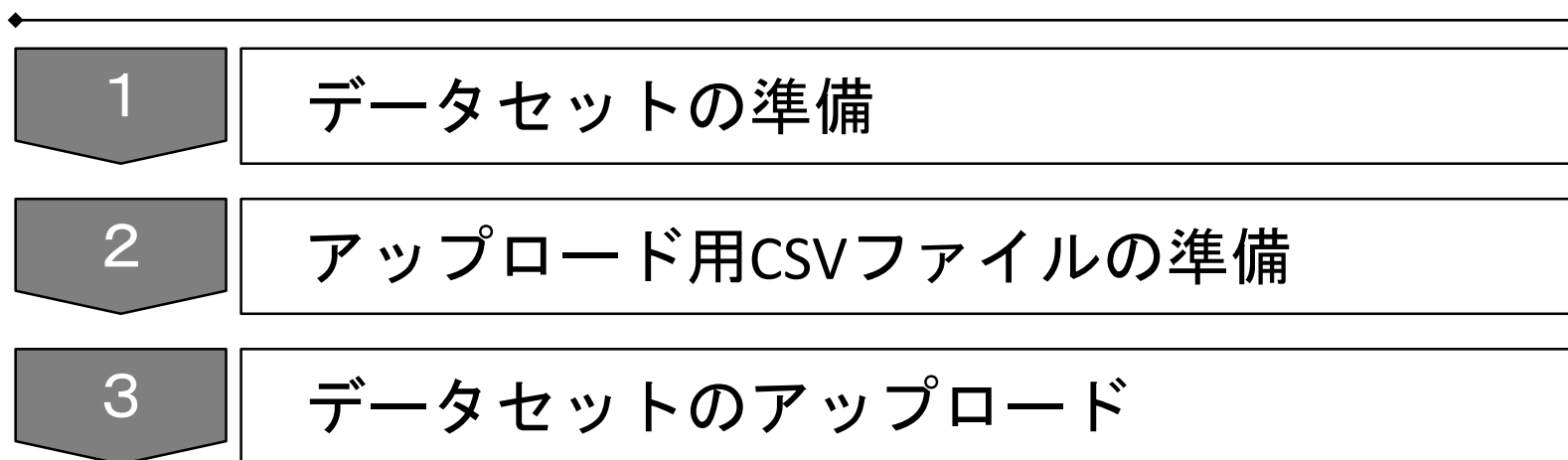
推論実行

# データアップロードのステップ

NNCはクラウドサービスのため、モデルを作成するために必要なデータセットをあらかじめクラウドにアップロードする必要があります。

お手持ちのPCにデータセットを準備し、以下のステップでクラウドへのアップロードを行います。

## データアップロードのステップ



# データセットの準備

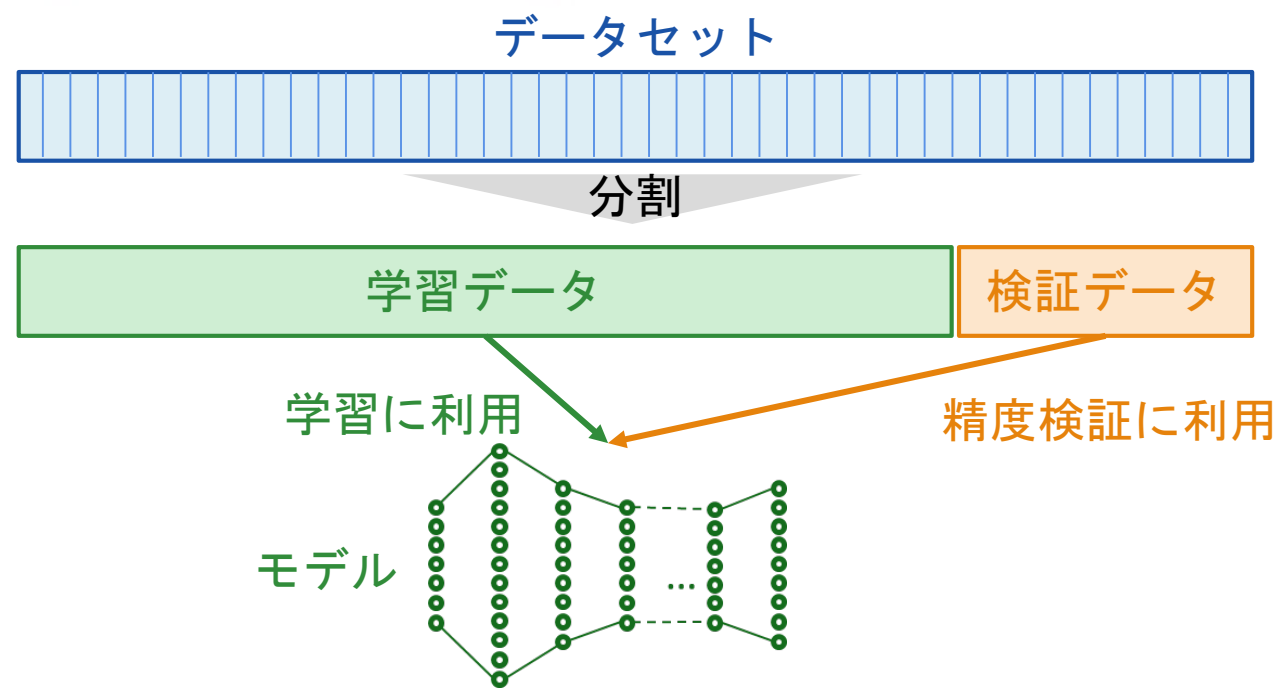
データセット  
の準備

CSVファイル  
の準備

データ  
アップロード

モデル作成には、モデルを学習させるためのデータセット（学習データ、Training Data）と、モデルの精度を検証するためのデータセット（検証データ、Validation Data）の2つが必要になります。

作成されたモデルの精度を正しく検証するためには、学習に利用していないデータで検証データを準備する必要があるため、あらかじめ準備したデータセットを学習データと検証データに分割しておきます。



# アップロード用CSVファイルの準備

データセット  
の準備

CSVファイル  
の準備

データ  
アップロード

データセットはNNC所定のアップロード用CSVファイルで準備する必要があります。

画像や時系列データは別途ファイルで準備いただき、ファイルの相対パスをアップロード用CSVファイルに記入します。

## NNC所定のデータセットフォーマットの例

	A	B
ヘッダ	x:image	y
データ (2行目以降)	./training/5/0.png	5
	./training/0/1.png	0
	./training/4/2.png	4
	./training/1/3.png	1
	./training/9/4.png	9
	./training/2/5.png	2
	./training/1/6.png	1
	./training/3/7.png	3
	./training/1/8.png	1
	./training/4/9.png	4

## ヘッダの書き方

変数名[\_\_次元Index]:[コメント]

- 変数名はNNC上でデータセットのカラムを指定する際に利用します。英数字で入力ください。
- 多次元の場合には変数名の後に、「\_\_」をつけ、0から順にインデックスをつけ、記載します。
- 「:」を入力することで、「:」以下をコメントアウトすることができます。  
後ろにデータセットの説明などのコメントを追加するなどして、利用ください。

# 表データのデータセット作成例

データセット  
の準備

CSVファイル  
の準備

データ  
アップロード

表データを利用する場合には、あらかじめ全てのデータを数値化しておく必要があります。  
入力データについては、“\_\_{数字}”の形式でベクトル形式で準備することで、1つの入力レイヤーで入力データを表現することができます。

## 表データ分類のデータセット作成例(退会予測の例)

### 数値化後のデータ

入力データ					出力データ	
入会後経過日数	性別	年齢	購入回数	購入総額	最終購入後経過日数	継続・退会
1386	1	36	12	17280	30	0
1142	1	39	8	18000	100	1
2004	2	43	4	42000	671	1
2221	2	37	1	21600	239	0
1613	0	46	35	20100	16	0
1239	0	38	11.5	28800	165	0
2336	1	28	18	42000	244	0

※ カテゴリ変数などはダミー変数化するなど事前に数値化する必要があります

### アップロード用CSVファイル

入力データをx、出力データをyにカラム名を変更  
複数カラムは“\_\_{数字}”でベクトル表現

x__0	x__1	x__2	x__3	x__4	x__5	y
1386	1	36	12	17280	30	0
1142	1	39	8	18000	100	1
2004	2	43	4	42000	671	1
2221	2	37	1	21600	239	0
1613	0	46	35	20100	16	0
1239	0	38	11.5	28800	165	0
2336	1	28	18	42000	244	0

※ ベクトル表現を利用しない場合、ネットワーク構築時に多数のインプットレイヤーを作成する必要があります



# 画像データのデータセット作成例

データセット  
の準備

CSVファイル  
の準備

データ  
アップロード

画像データを利用する場合には、画像データは相対パスを記載します。  
画像データは事前に画像サイズとカラーかモノクロかを統一する必要があります。

## 画像分類のデータセット作成例 (犬と猫の2分類のモデル作成時の例)



カラム名のxが入力で、yが出力

trainData.csv

x	y
data/train/dog/001.jpg	0
data/train/dog/002.jpg	0
data/train/dog/003.jpg	0
⋮	
data/train/cat/001.jpg	1
data/train/cat/002.jpg	1
data/train/cat/003.jpg	1
⋮	

画像は相対パスを記載

testData.csv

x	y
data/test/dog/001.jpg	0
data/test/dog/002.jpg	0
data/test/dog/003.jpg	0
⋮	
data/test/cat/001.jpg	1
data/test/cat/002.jpg	1
data/test/cat/003.jpg	1
⋮	

ラベルは事前に数値化

この例ではdogを0、catを1  
と定義し、データを準備

# 時系列データのデータセット作成例

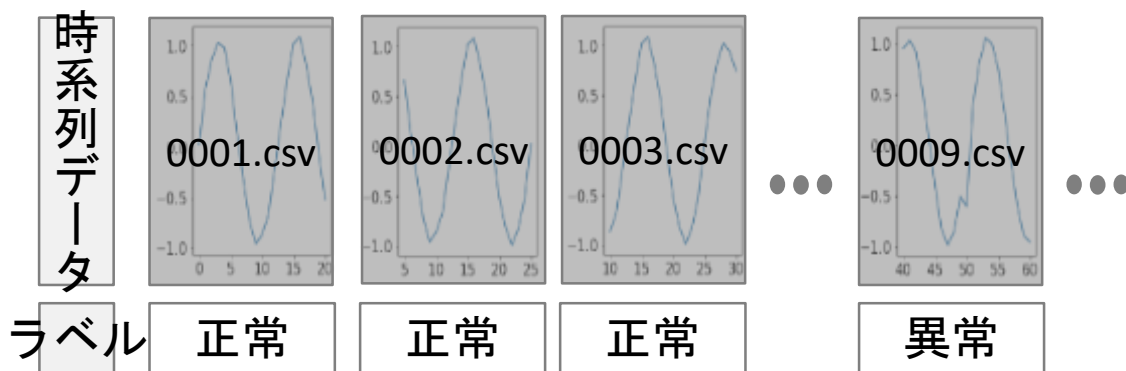
データセット  
の準備

CSVファイル  
の準備

データ  
アップロード

時系列データを利用する場合には、時系列データはCSVファイルの相対パスを記載します。  
時系列データは事前に入力数とその長さを統一する必要があります。

## 時系列データ分類のデータセット作成例(時系列異常検知の例)



カラム名のxが入力で、yが出力  
training\_data.csv    0001.csv

x	y	
0001.csv	0	0.02355
0002.csv	0	0.57766
0003.csv	0	0.64312
⋮	⋮	0.93555
⋮	⋮	0.83247
0009.csv	1	⋮
⋮	⋮	-0.3254

ラベルは事前に数値化  
この例では正常を0、異常を  
1と定義し、データを準備

- ✓カラム名や時刻情報は入れず、  
数値データのみを記載
- ✓複数の時系列データを入力に  
する場合には、列を増やして  
記載

# データセットのアップロード

データセット  
の準備

CSVファイル  
の準備

データ  
アップロード

NNCへのデータアップロードには専用のアップローダを利用します。  
アップローダ上で準備したCSVファイルを指定し、データをアップロードします。

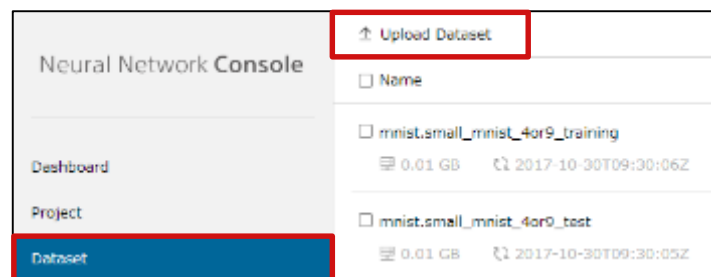
## 1. アップローダの取得

- ✓ 以下のリンクからアップローダをダウンロード

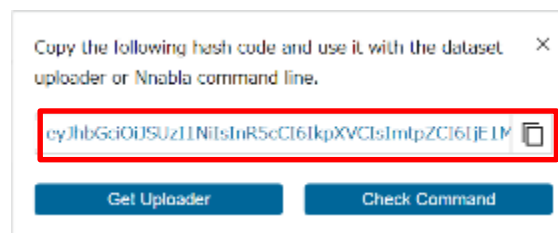
<https://support.dl.sony.com/docs-ja/データセットアップロードツールのダウンロード/>

## 2. アップロードキーの取得

- ✓ NNCにログインし、Datasetタブの中のUpload Datasetをクリック

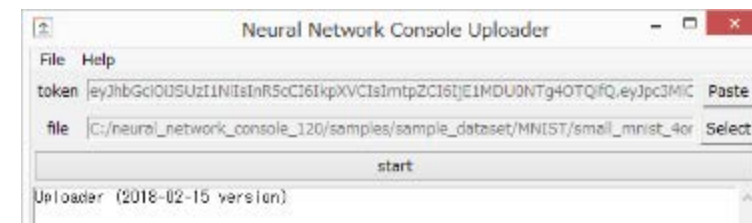


- ✓ ポップアップ画面に表示されるアップロードキーをコピー



## 3. アップローダの実行

- ✓ 1 で取得したアップローダを起動
- ✓ tokenに2で取得したアップロードキーを貼り付け
- ✓ fileに作成したCSVファイルを指定
- ✓ Startをクリックし、アップロードを実行



※画像データはCSVに含まれるパスを参照し自動的にアップロードされます。

# アップロード先のデータセット確認

データセット  
の準備

CSVファイル  
の準備

データ  
アップロード

アップロード後はDatasetタブの一覧にデータセットが追加されます。

アップロード時のCSVファイルのファイル名がデータ名として一覧に表示され、選択することで中身を確認することができます。表示の際に、画像や時系列データなどはサムネイルの形で確認できます。

## アップロード後のデータセットの例

### アップロード時のCSVファイル

#### mnist.mnist\_training.csv

x:image	y:label
data/train/5/0001.jpg	5
data/train/0/0001.jpg	0
data/train/4/0001.jpg	4
data/train/1/0001.jpg	1
⋮	⋮

### データセット一覧に追加されたデータセット

Neural Network Console

Group Personal

Preview

60000 Rows 2 Cols

CSVのファイル名がデータ名になります

mnist.mnist\_training

mnist.mnist\_test




mnist.mnist\_training\_100

mnist.mnist\_unlabeled

iris\_flower\_dataset.iris\_flower\_dataset\_training\_delo

iris\_flower

画像や時系列データはサムネイルで表示されます

	x:image	y:label
1		5
2		0
3		4
4		1



# 目次

1

アカウントサインイン

2

データセットのアップロード

3

モデル作成

4

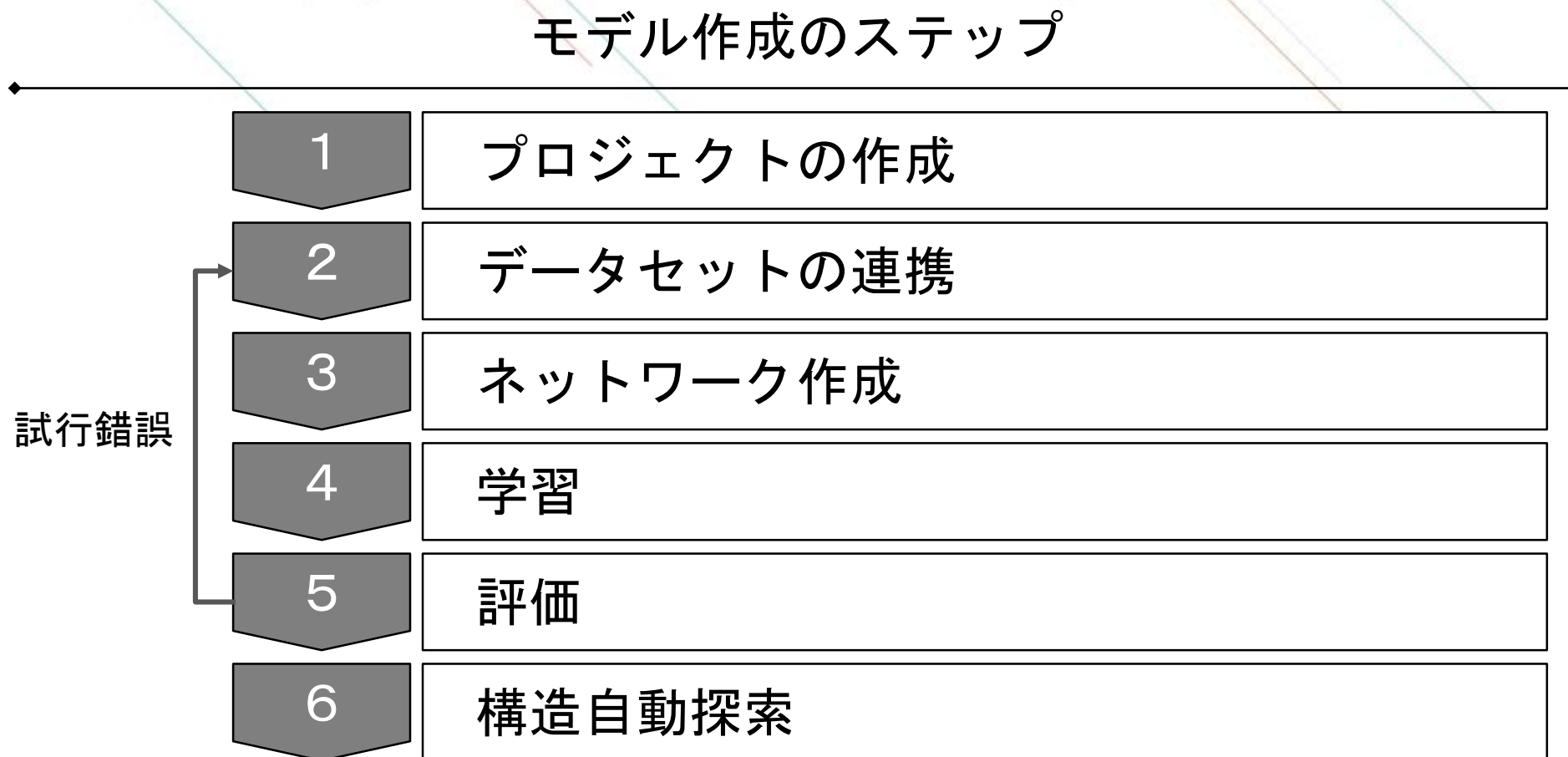
推論実行



# モデル作成のステップ

NNCではモデル作成の一連のステップをプロジェクトの中で実施します。

プロジェクト内で試行錯誤の履歴が一元管理されるため、最も良いモデルを簡単に見つけることができます。



# プロジェクトの作成



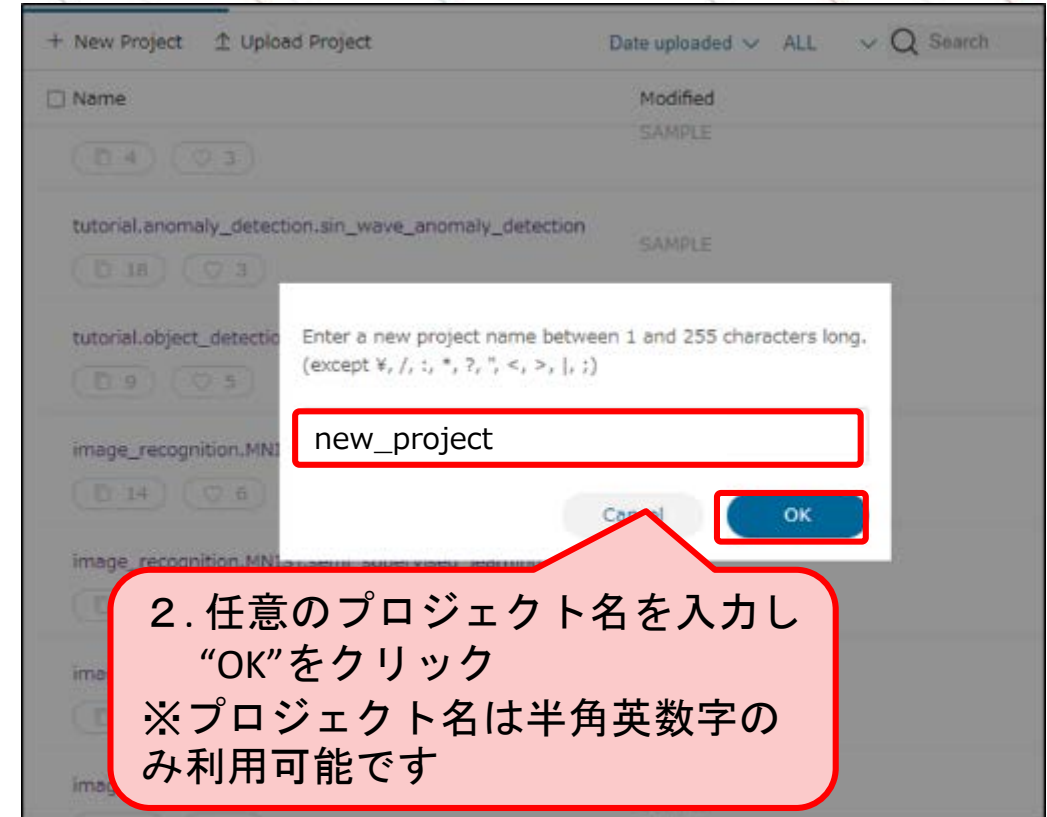
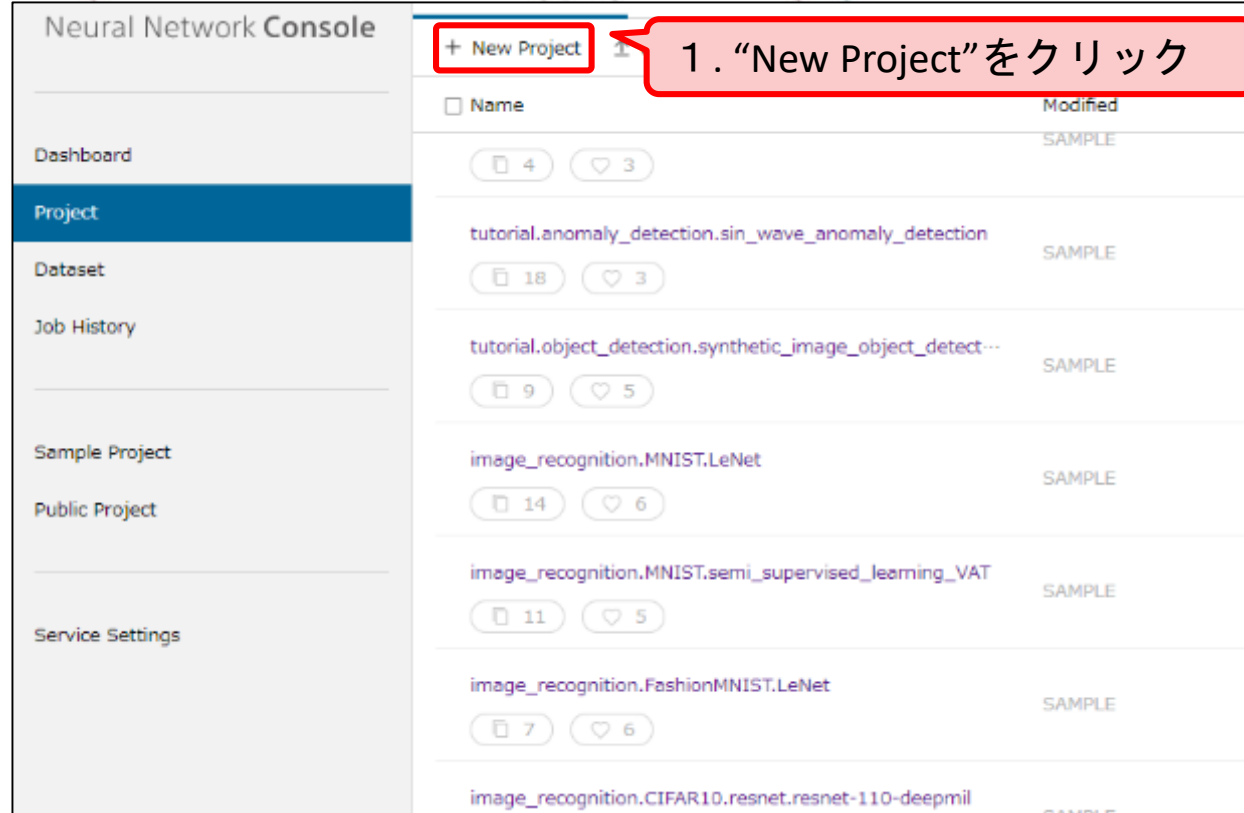
プロジェクトの作成方法は以下の 4 種類があります。用途に合わせて選択をしてください。

プロジェクトの作成方法		利用シーン
1	新規プロジェクトの作成	ゼロからモデルを作成
2	既存プロジェクトのコピー	NNC上で作成したプロジェクトをコピーし、それをベースにして作成
3	サンプルプロジェクトのコピー	NNC上の関連するサンプルプロジェクトをもとにモデルを作成
4	他DLフレームワーク等のモデル利用	他DLフレームワーク等の関連するDLモデルをもとに、NNCでモデルを作成

# プロジェクトの作成

## 新規プロジェクトの作成

Projectタブから新規プロジェクトを作成します。

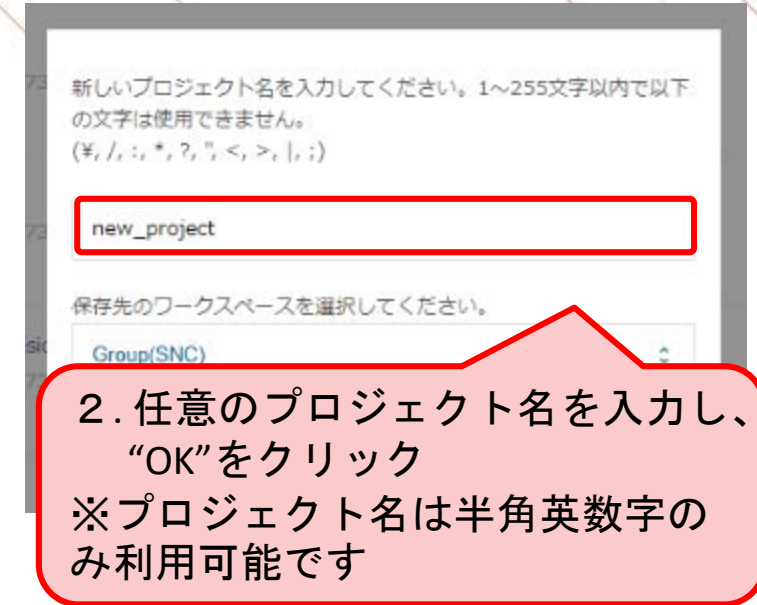
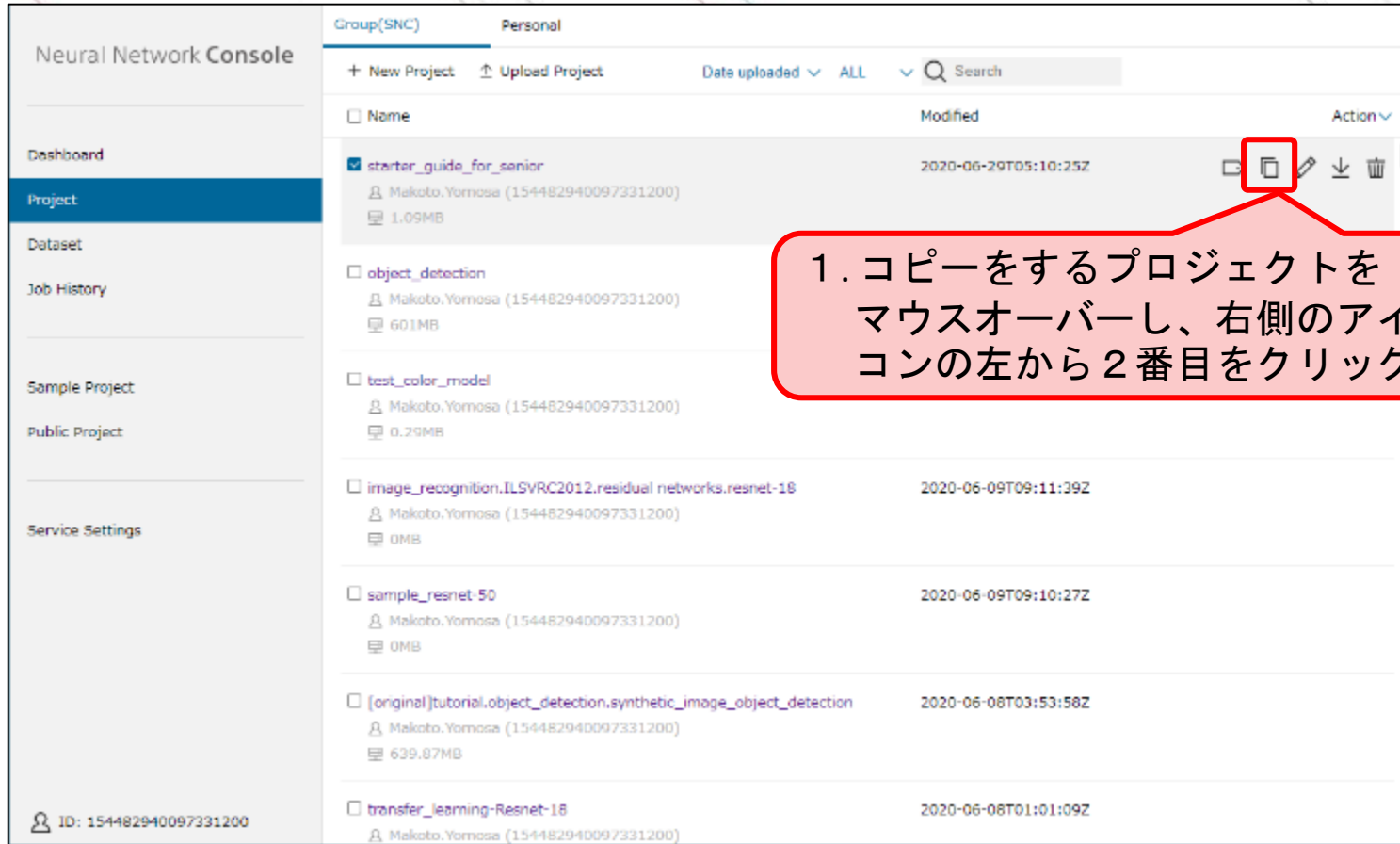


# プロジェクトの作成

## 既存プロジェクトのコピー



Projectタブから既存プロジェクトのコピーを作成します。



## プロジェクトの作成

# サンプルプロジェクトのコピー

サンプルプロジェクトはSample Projectタブに目的別やテクニック別などに整理されており、ここからプロジェクトをコピーします。

Neural Network Console

Dashboard  
Project  
Dataset  
Job History  
Sample Project  
Public Project  
Service Settings

目的から探す

テクニックから探す

データセットから探す

人気ランキング

目的から探す > 画像認識

I Input  
Dataset : x

A Affine

S Sigmoid

B BinaryCrossEntropy  
T.Dataset : y

1,28,28  
1  
1  
1

tutorial.basics.01\_logistic\_regression

7627 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

I Input  
Dataset : x

C Convolution2D

M MaxPooling2D

T Flatten

C Convolution2D

M MaxPooling2D

A Flatten

T Flatten

A Flatten

S Softmax

B Softmax

1,28,28  
16,256,24  
16,12,12  
16,12,12  
8,8,8  
8,8,4  
10  
10  
10  
10  
10

tutorial.basics.02\_binary\_cnn

1341 26

4層Convolutional Neural Networkによる画像の2クラス分類

Cloudで開く Download

I Input  
Dataset : x

A Flatten

T Flatten

A Flatten

S Softmax

B Softmax

1,28,28  
256  
256  
256  
10  
10

tutorial.basics.10\_deep\_mlp

397 18

Deep Neural Networks

Cloudで開く

1. “Cloudで開く”をクリック

2. ポップアップで好きな名前  
※を入力してOKをクリック  
※プロジェクト名は半角英数字のみ利用可能です

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。  
(¥, /, :, \*, ?, ", <, >, |, ;)

new\_project

Cancel OK

※サンプルプロジェクトのプロジェクト名がわかっている場合には、Projectタブからコピーすることも可能です。



# プロジェクトの作成 他DLフレームワーク等のモデル利用



以下に示す 5 つの移植元からネットワークモデルなどをロードすることで、プロジェクトを作成できます。

	移植元	対応フォーマット
1	NNC※1	.sdcproj
2	DLフレームワーク NNabla※2	.nntxt、.nnp
3	DLフレームワーク ONNX	.onnx
4	DLフレームワーク Caffe	.prototxt
5	DLフレームワーク TensorFlow※3	.pb

※ 1 NNC Windows版やクラウド版の他ユーザからのモデル転用を想定しています。

※ 2 [NNabla](#)とはソニーが開発したDLフレームワークで、NNCでもバックエンドではNNablaを利用しています。

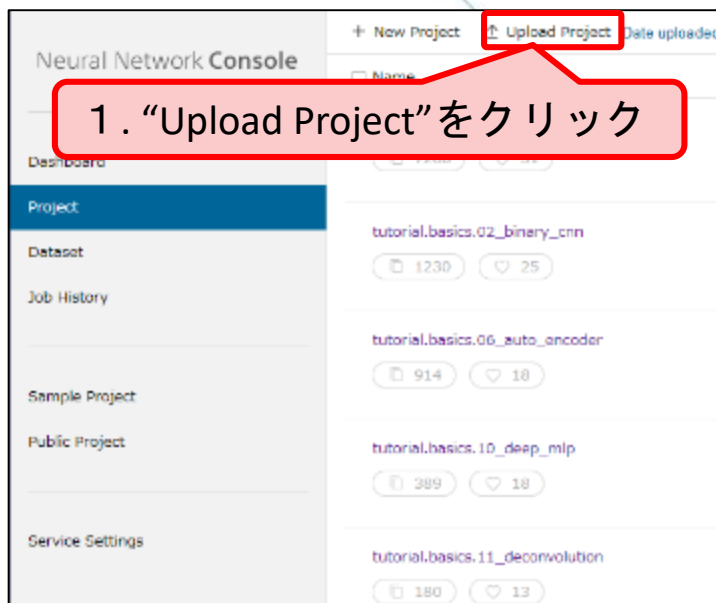
※ 3 TensorFlowの互換状況は[ウェブページ](#)でご確認ください。

# プロジェクトの作成 他DLフレームワーク等のモデル利用

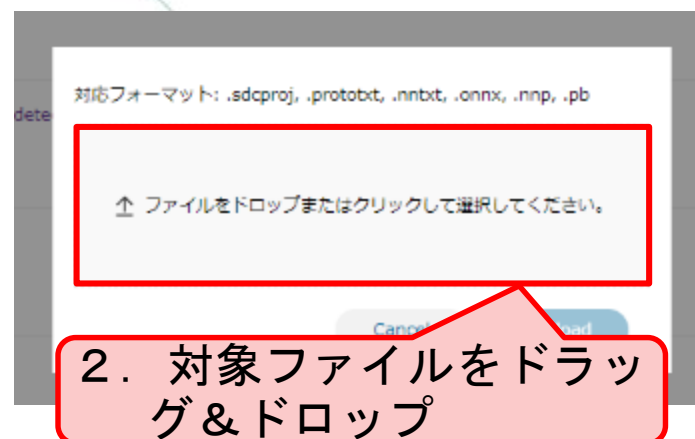


Projectタブから、以下のステップで他DLフレームワークなどのモデルをロードします。

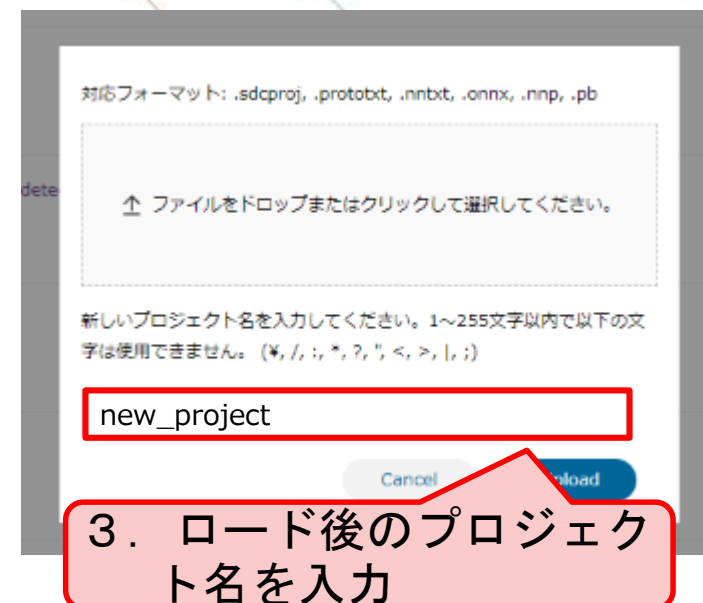
## 1. Upload Projectを選択



## 2. 対象ファイルの指定



## 3. プロジェクト名の指定



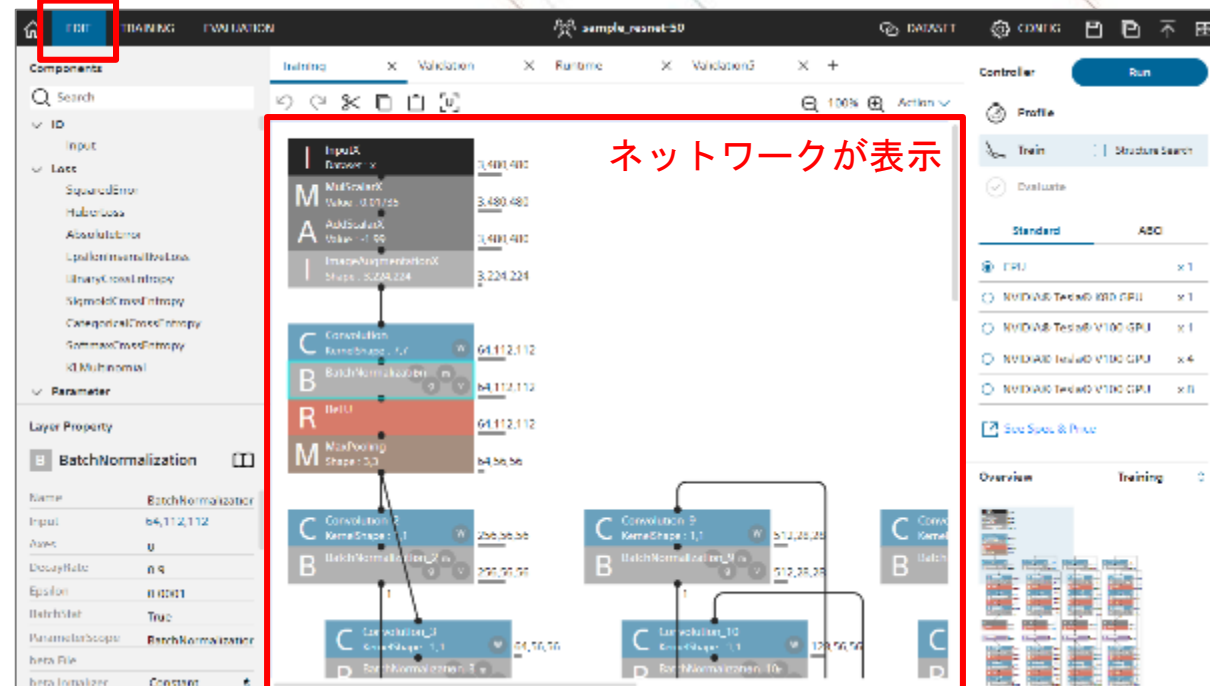
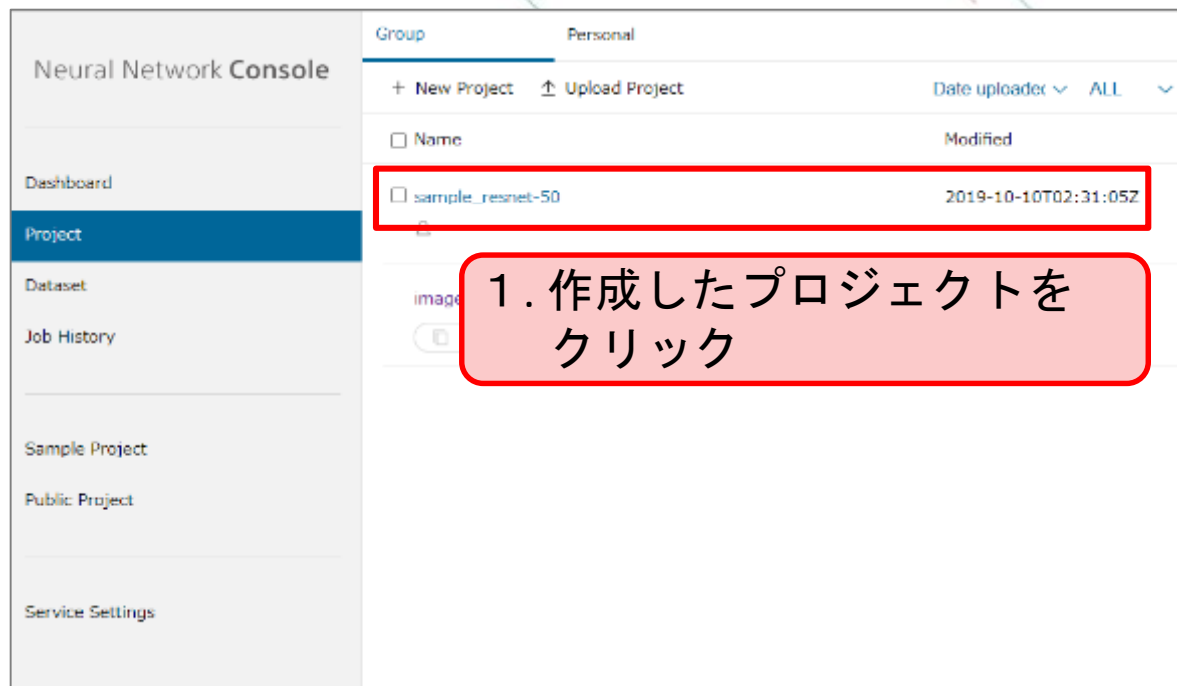
# プロジェクトの起動



作成したプロジェクトをクリックし、プロジェクトを起動します。

※「新規プロジェクトの作成」を選択された場合には、自動で空のプロジェクトが起動されます。

Editタブ: ネットワークを作成するページ



# データセットの連携



プロジェクトに事前にアップロードしたデータセットを紐づけます。

1. Datasetタブをクリック

2. Trainingをクリック

3. Not Setをクリック

- Suffle  
エポックごとにデータの順序を入れ替え
- Image Normalization  
入力データが画像の場合、入力時に1/255倍  
TrainingとValidationで同じになるように注意

6. Validationをクリックし、3～5と同様の手順を実施

4. 一覧から学習に用いるデータセットを選択

選択したデータセット名

mnist.small\_mnist\_4or9\_test

500 Rows 2 Cols 2017-10-30T09:30:05Z

5. リンクマークをクリック※2

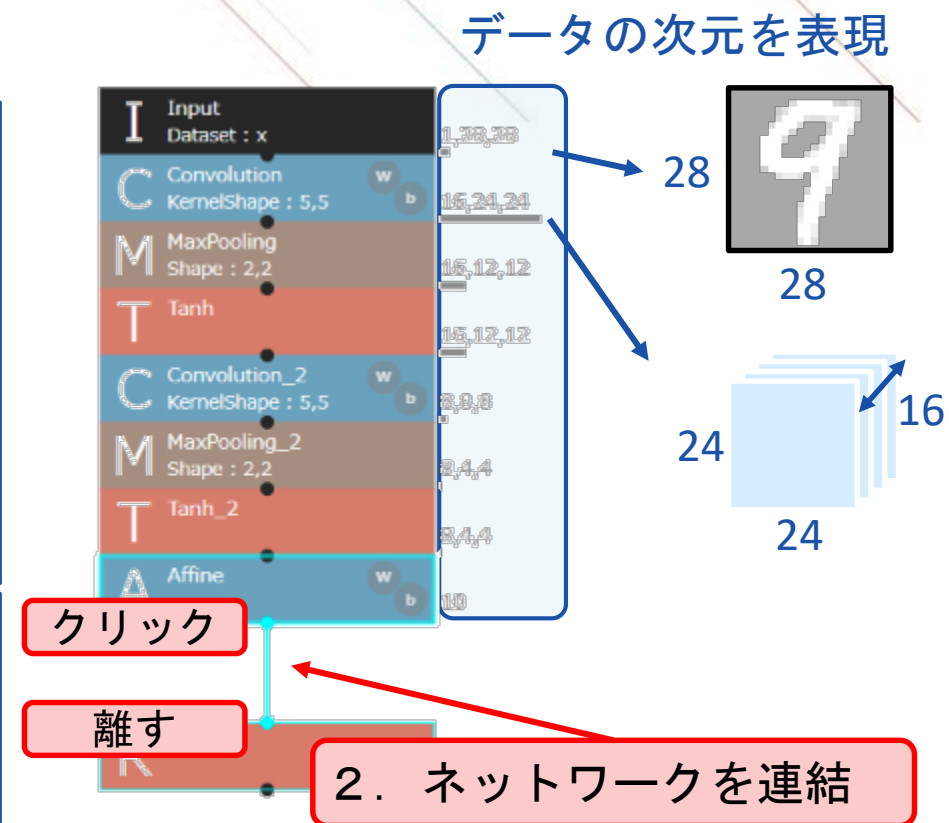
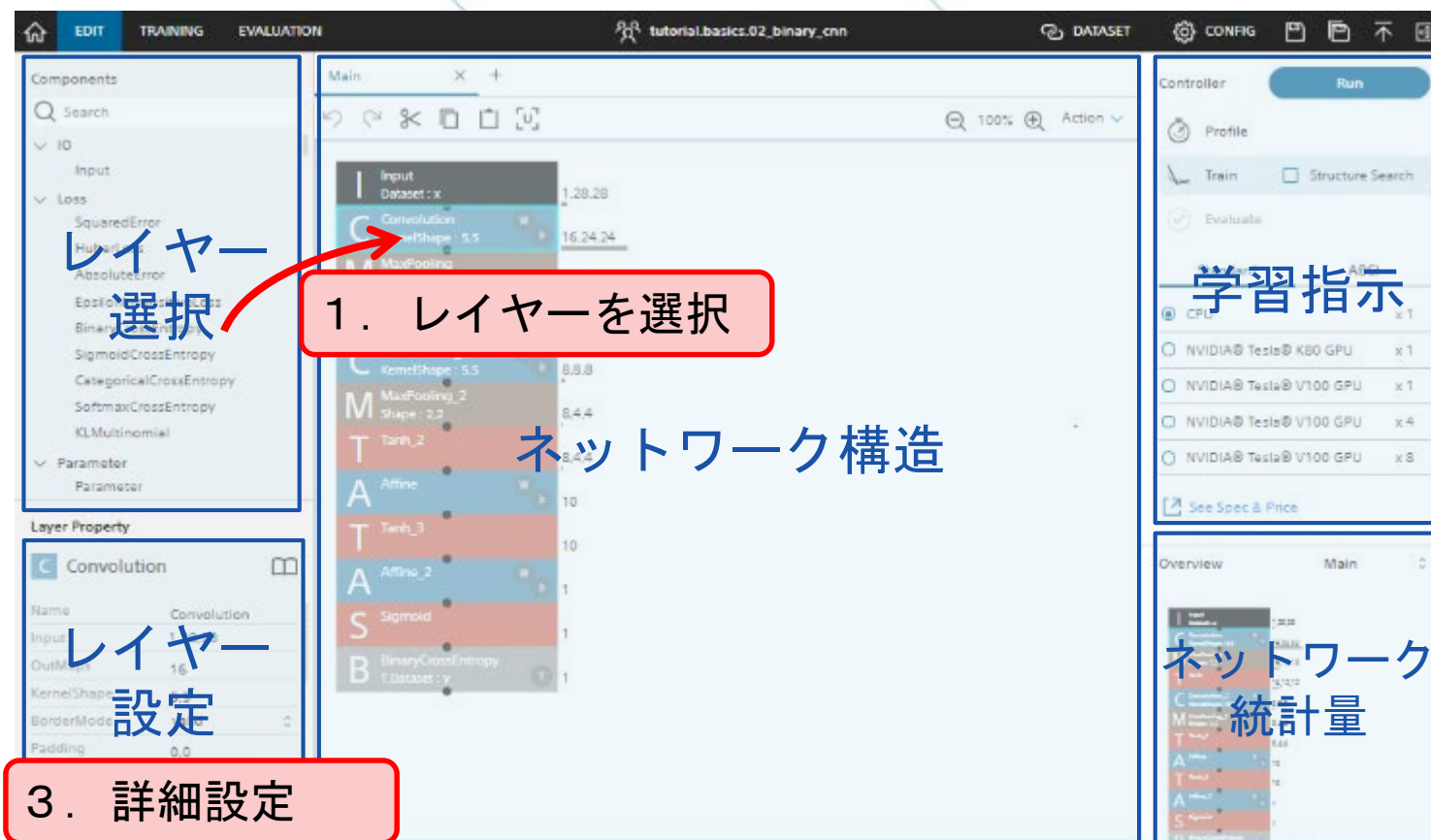
※2 ブラウザの拡大率によって表示されないことがあります。  
表示されない場合は表示の縮小をお試しください。



# ネットワークの作成方法



レイヤー選択パネルからネットワーク構造パネルにレイヤーをドラック＆ドロップします。  
すでにあるレイヤーと連結し、ネットワークを構成します。  
レイヤーパラメータパネルで各レイヤーの詳細を設定します。



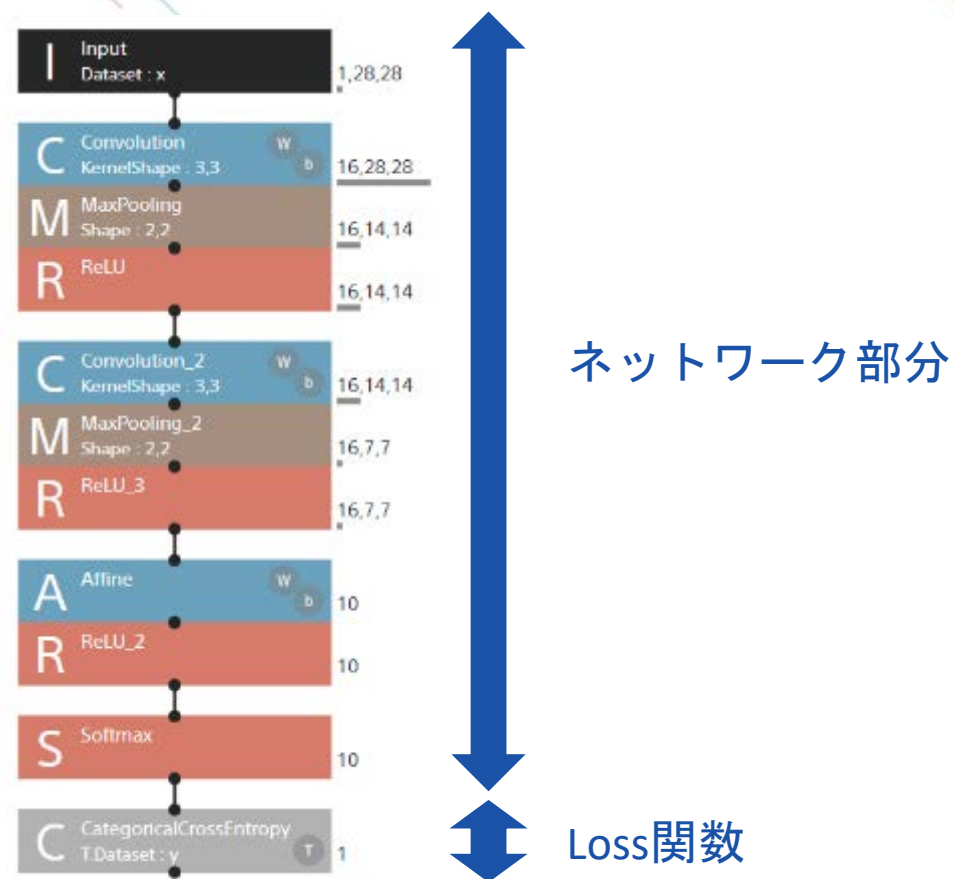


# ネットワークレイヤーの並べ方

ブロックで表現されたレイヤーを並べることでネットワークを構築します。

Inputから順にネットワークを作成し、最後にLoss関数を配置します。

推論時にはLoss関数の直前のレイヤーの値を出力します。Loss関数を利用せずLoss値を算出する場合には、ネットワークの出力部分を判断できないため、推論用のネットワークを明示的に作成する必要があります。

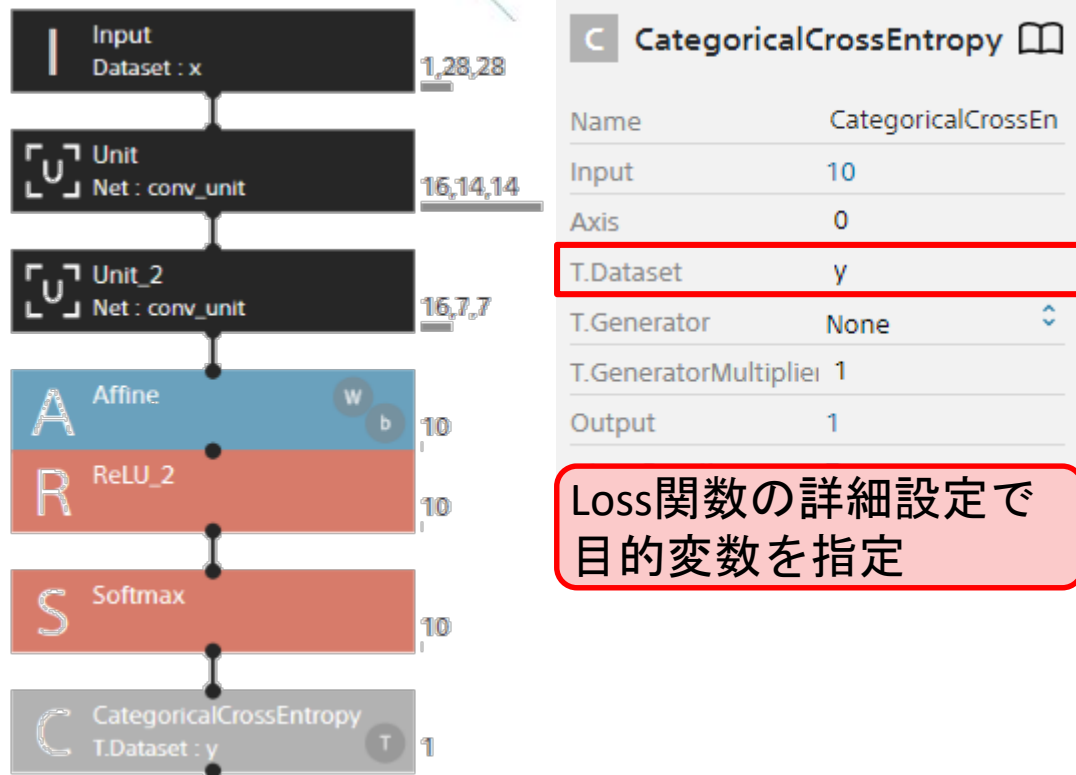


# 目的変数の設定

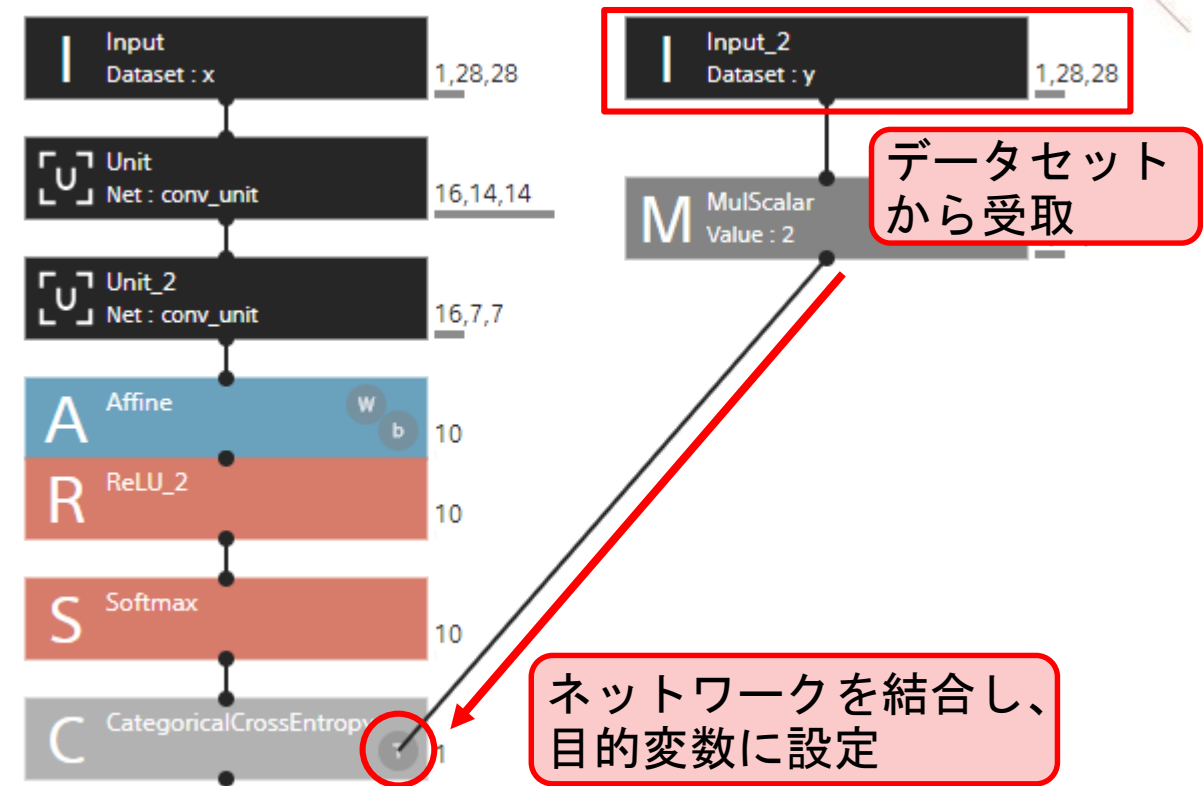
Loss関数に目的変数を設定する方法には、Loss関数の詳細設定から設定する方法と、ネットワーク上で設定する方法の2種類があります。

## 目的変数の設定方法

### 詳細設定でデータセット名を設定



### ネットワーク内で加工し、目的変数を設定



# ネットワークタブ

EDITページのネットワークタブは複数設置することができます。

CNNなど繰り返し構造が多いネットワークの共通部分をコンポーネントとして作成する場合や、学習用、検証用、推論用でネットワーク構造を変える場合※に複数のネットワークタブを利用します。

1. 「+」ボタンをクリックし、新規タブを作成

複数のネットワークタブが存在

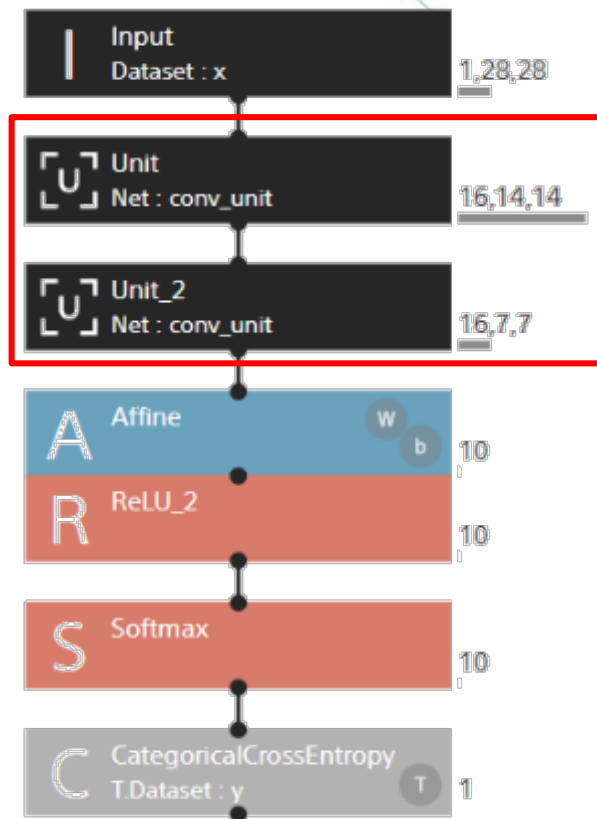
※ ネットワークタブがデフォルトのMainのみの場合、バックエンドで自動的に検証用のMainValidation、推論用のMainRuntimeが生成されます。

# ネットワークタブの引用(Unitレイヤー)

Unitレイヤーを利用することで、他のネットワークタブを引用利用することができます。

## Unitレイヤーの使い方の例

Mainタブ



Unitレイヤーを用いて  
Conv\_unitタブを引用

### Layer Property

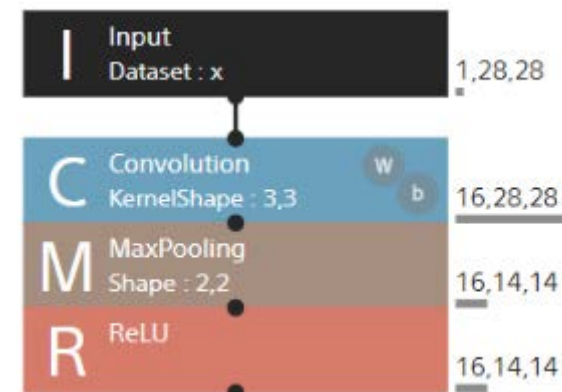
Unit

Name	Unit
Input	1,28,28
Network	conv_unit

引用するネットワー  
クタブの名称を指定

Convolution.W.LRate 1  
Convolution.b.File

Conv\_unitタブ



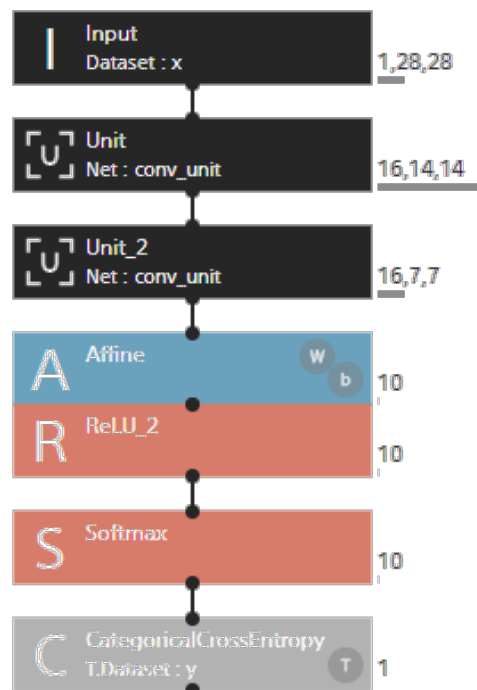
引用元のネットワークタブはInputレイヤーからはじまるネットワークを作成します。  
右側に表示されるデータの次元は引用先に応じて自動で変換されます。



# 変数の利用(Argumentレイヤー)

Argumentレイヤーを利用することで、ネットワーク内の数値などを変数として取り扱うことができます。Argumentレイヤーで変数を設定し、その変数名をネットワークの設定値に利用することができます。また、Unitレイヤーの引用ネットワークタブ内で利用することで、ネットワークの引数として利用することもできます。

## Argumentレイヤーの使い方(出力サイズを変数とする例)



A output\_size  
Value : 10

1. Argumentレイヤーを追加

Layer Property	
A Argument	
Name	output_size
Value	10
Type	Int

2. Argumentレイヤーの詳細設定に変数名、値、変数型を指定

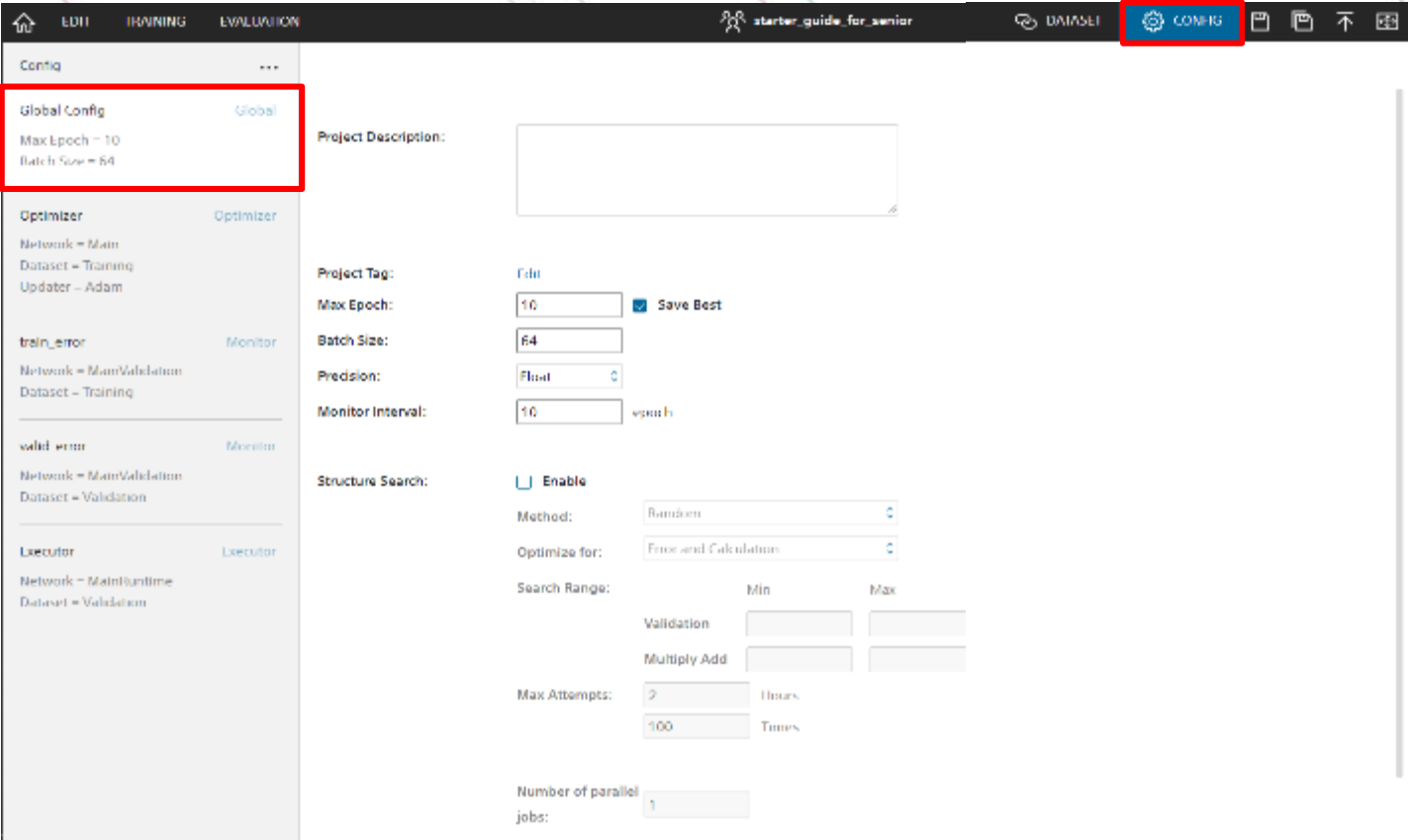
Layer Property	
A Affine	
Name	Affine
Input	16, 7, 7
OutShape	output_size
BaseAxis	0
WithBias	<input checked="" type="checkbox"/>
ParameterScope	Affine
W.Initializer	NormalAffineGlor

3. 変数名をそのまま入力



# 学習パラメータ (Global Config)の設定

学習パラメータを右上のCONFIGタブから設定します。  
Global Configでは、学習時の基本的な設定を行います。



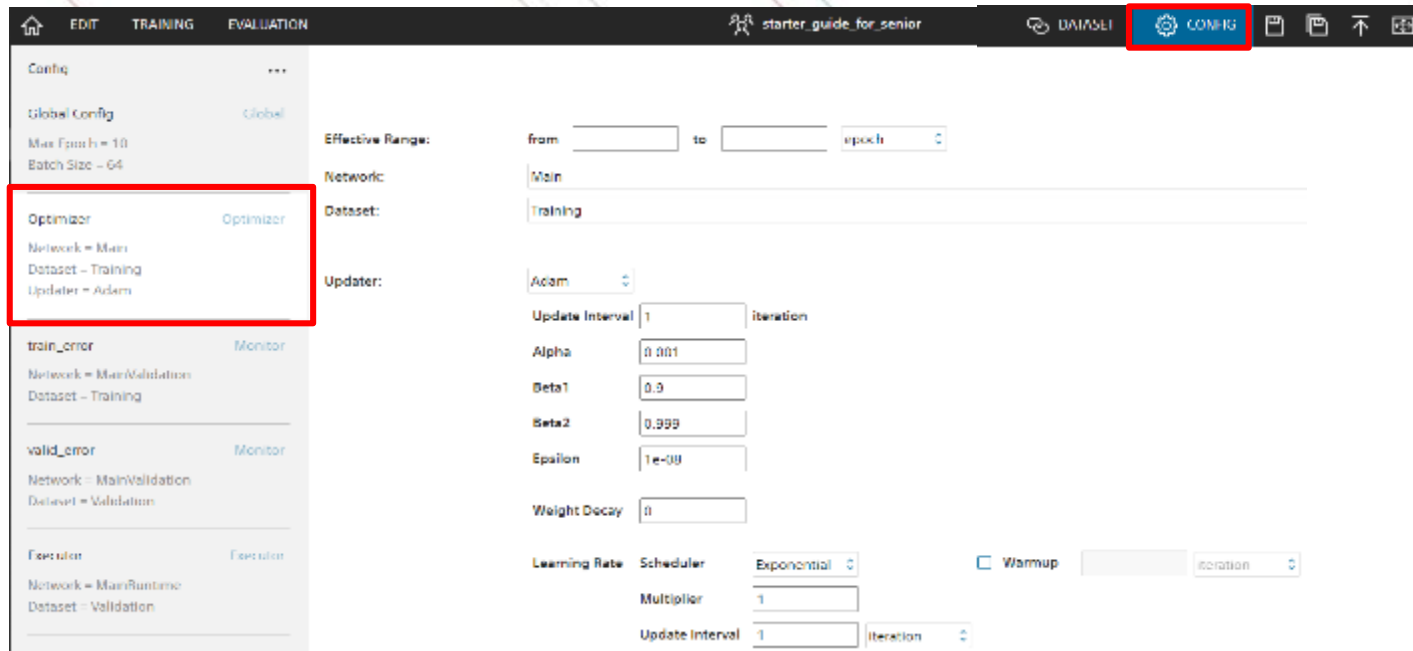
※詳細な解説は[ウェブページ](#)をご確認ください。



項目	説明
Project Description	プロジェクト概要など任意のコメントを入力。 半角英数字のみ使用可能。
Project Tag	Public Projectの検索に利用するタグを設定。 タグ設定により、プロジェクト公開時に他のユーザがを見つけやすくなる。
Max Epoch	学習の回数を指定。 学習曲線の横軸のため、収束していない際には値を大きくし、すでに収束している際には値を小さくする。
Batch Size	学習時のバッチサイズを指定。 値が大きいくほど、GPUの並列計算による学習時間短縮の効果が大きい。 上限はデータやネットワークのサイズ、GPUの種類によって変化。
Precision	演算精度を指定。 Halfを選択することで、メモリ使用量の削減やGPU演算の高速化が可能。
Monitor Interval	精度評価を行う間隔を指定。
Structure Search	構造探索機能の実施可否を指定。

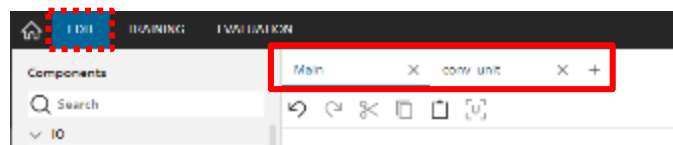
# 学習パラメータ (Optimizer)の設定

学習パラメータを右上のCONFIGタブから設定します。  
Optimizerでは、学習最適化の基本的な設定を行います。



項目	説明
Effective Range	最適化の実施範囲を指定。 空欄の場合には全区間で実施。 学習途中で最適化の手法やパラメータを変更する際に、区間を指定した複数のOptimizerを作成。
Network	最適化に用いるネットワークタブの名称を指定
Dataset	最適化に用いるデータセットタブの名称を指定
Update	最適化の手法と選択した手法のパラメータ値を設定

ネットワークタブの名称確認



データセットタブの名称確認



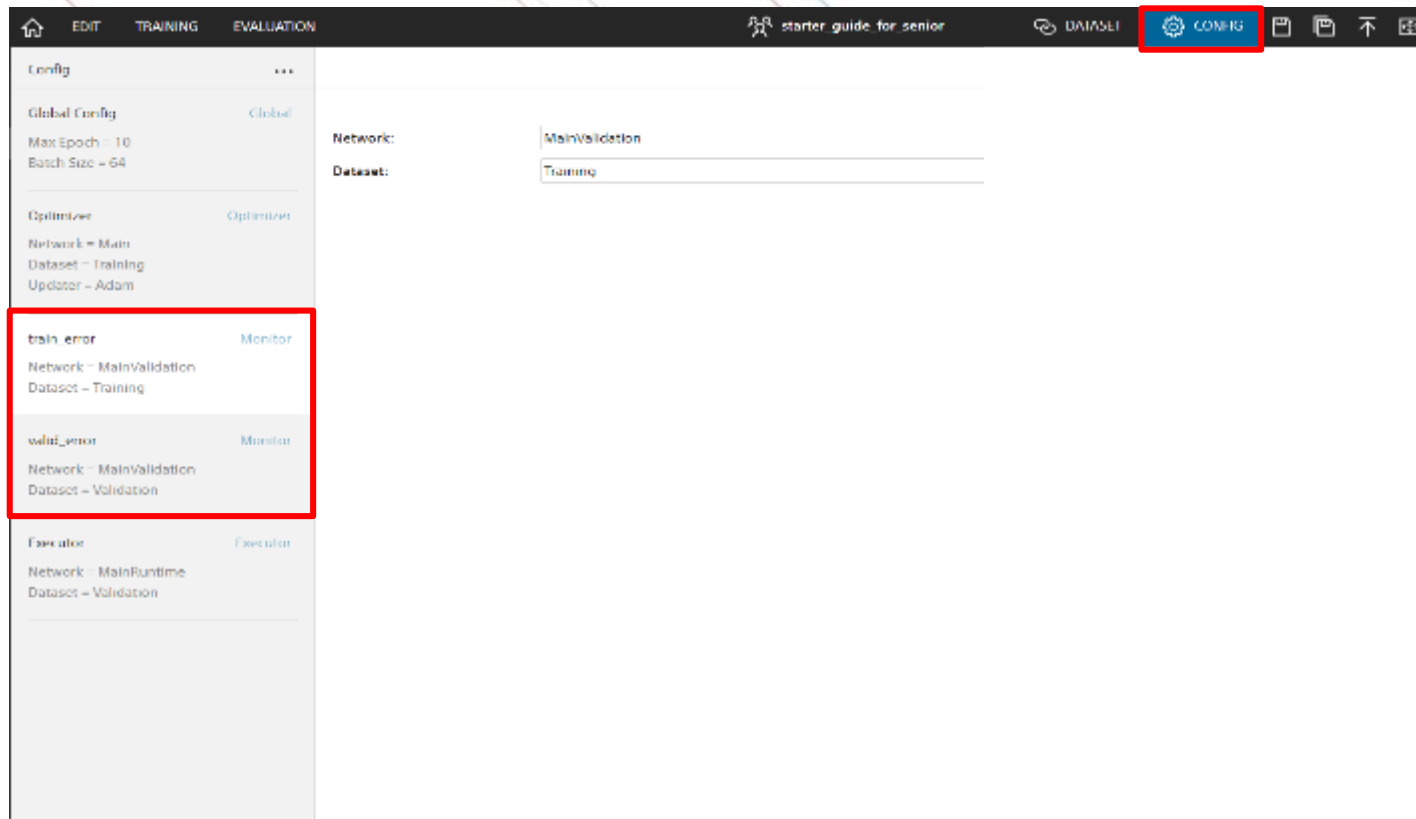
※詳細な解説は[ウェブページ](#)をご確認ください。

# 学習パラメータ(training\_error, valid\_error)の設定



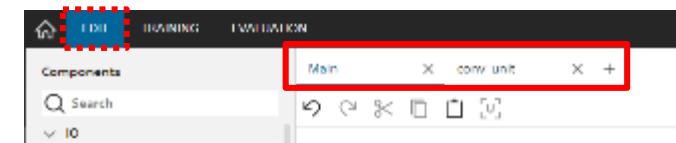
学習パラメータを右上のCONFIGタブから設定します。

training\_error, valid\_errorでは、学習曲線に表示するError値の計算方法の設定を行います。

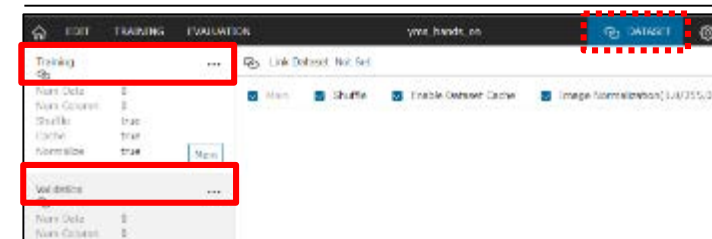


項目	説明
Network	Error値の計算のためのネットワークタブの名称を指定 ネットワークタブがデフォルトのMainだけの場合、MainValidationと指定※
Dataset	Error値の計算のためのデータセットタブの名称を指定

## ネットワークタブの名称確認



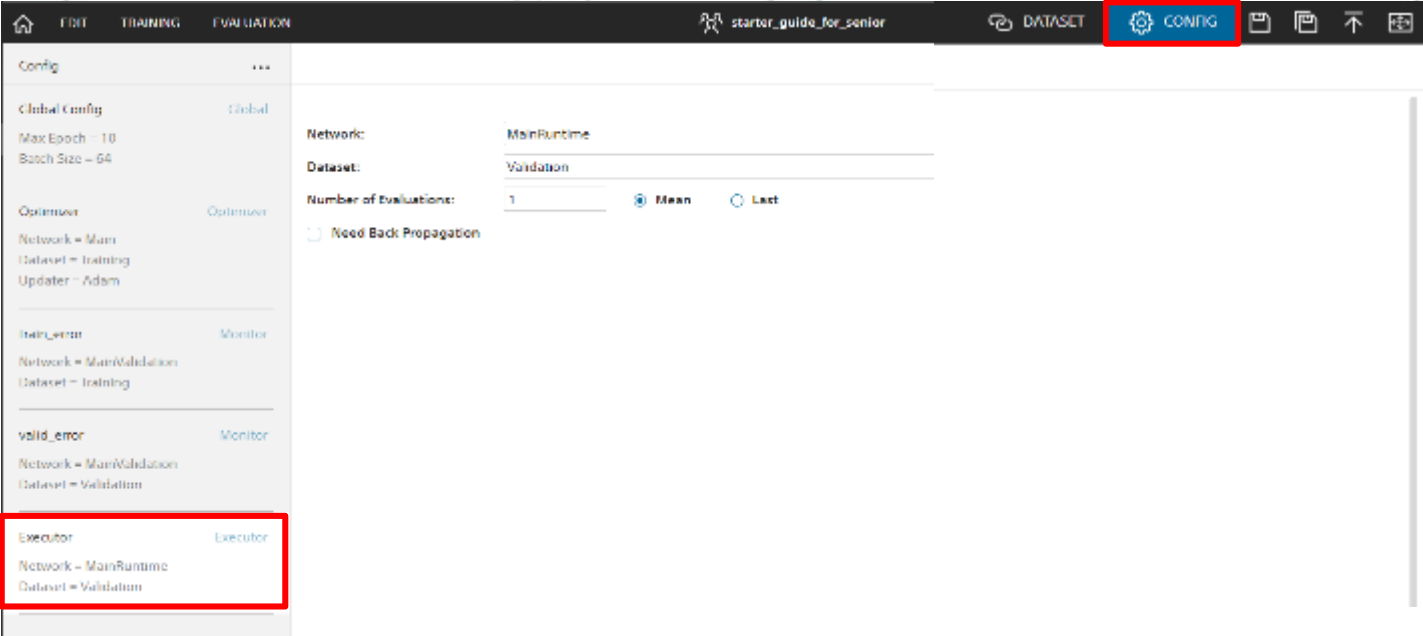
## データセットタブの名称確認



※ ネットワークタブがデフォルトのMainのみの場合、バックエンドで自動的に検証用のMainValidation、推論用のMainRuntimeが生成されます。

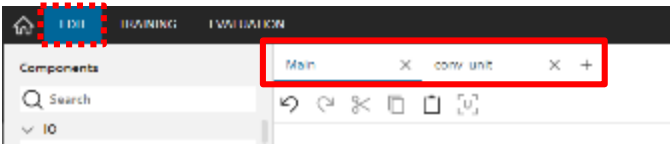
# 学習パラメータ (Executor)の設定

学習パラメータを右上のCONFIGタブから設定します。  
Executorでは、評価推論用の設定を行います。



項目	説明
Network	評価のためのネットワークタブの名称を指定 ネットワークタブがデフォルトのMainだけの場合、MainRuntimeと指定※
Dataset	評価のためのデータセットタブの名称を指定
Number of Evaluation	評価実行の回数を指定 乱数処理などを含む場合に、複数回の推論の平均で評価を実施する際に利用

ネットワークタブの名称確認



データセットタブの名称確認



※ ネットワークタブがデフォルトのMainのみの場合、バックエンドで自動的に検証用のMainValidation、推論用のMainRuntimeが生成されます。

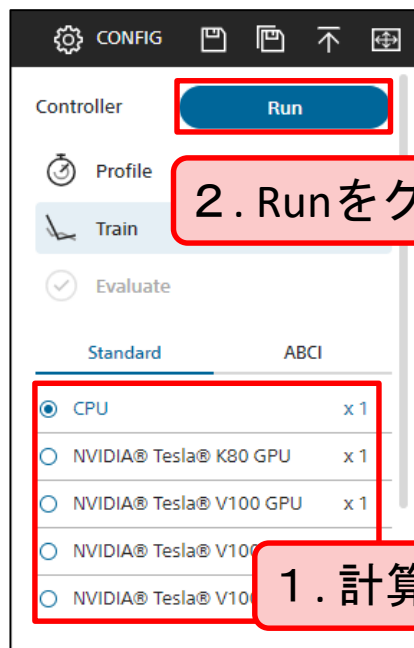


# 学習の実行



EDITページのRunボタンをクリックすることで学習が実行されます。  
GPU等有料のメニューを利用する場合は事前にクレジットカード登録もしくは法人契約が必要になります。  
(法人契約: <https://dl.sony.com/ja/business/>)

## 学習実行の方法



2. Runをクリック

1. 計算資源を選択

## TRAININGページの概要



※データ件数が少ない場合には、バッチサイズのエラーがポップアップで表示される場合があります。この場合には、[Global Config](#)でバッチサイズをデータ件数よりも小さい値に変更してください。



# パラメータの可視化



TRAININGタブに表示されるレイヤーをダブルクリックすることで、そのレイヤーのパラメータを可視化することができます。

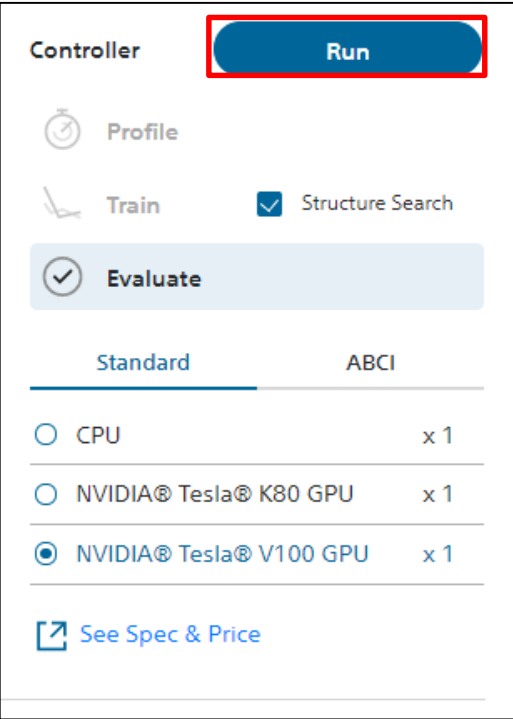
2. パラメータが可視化

1. レイヤー※をダブルクリック

※ReLUなど表示するパラメータがない場合には、パラメータは表示されません。

# 評価の実行

TRAININGページのRunをクリックするとEVALUTIONページに遷移し、詳細な判定結果を確認できます。  
各データに対するモデルの判定結果や統計的な精度や指数、混同行列などを確認できます。

評価実行の方法	表示可能なグラフの概要		
	評価グラフ	内容	問題
	Output Result	各データの1つ1つの判定結果	分類/回帰
	Confusion Matrix	データセット全体の統計的な指標と混同行列(分類ラベルごとに結果を集計した表)	分類
	Classification Result	各データの判定確率上位3カテゴリーの確率	分類
	Classification Matrix	カテゴリーごとのモデルの判定傾向	分類
	Likelihood Graph	判定確率と正答率の傾向	分類

# 評価の見方：Output Result

検証用データ (DatasetタブのValidationデータ)の右側にモデルの予測結果が追記されます。  
1つ1つの検証用データに対し、モデルがどのように判断したかを確認できます。

## Output Resultページのスナップショットとその見方

クリックして選択

Output Result

Confusion Matrix

y - y'

<< < 1 / 50 > >>

Classification Result



y - y'

Classification Matrix

y - y': Recall

Likelihood Graph

y - y'

Index	x:image	y:9	y'_0	y'_1
1		0	0.9930715	0.0069284593
2		0	0.99998736	1.2614053e-05
3		1	0.00012720657	0.9998728
4		1	0.0025280912	0.99747187

検証用データから転記

モデルの予測結果

ラベル"0"の  
予想確率

ラベル"1"の  
予想確率

ページを変更し全ての結果を確認可能

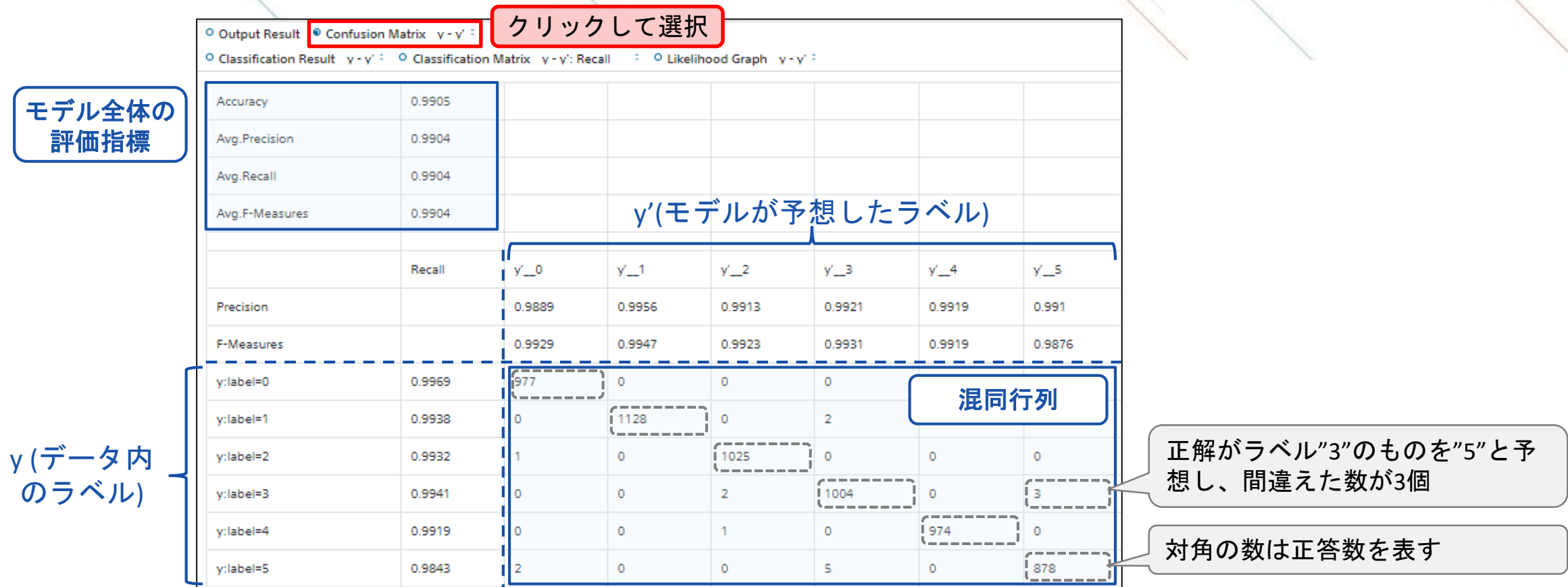
追記カラム名について

- 出力をアポストロフィー付きで表記  
例: 学習時に $x \rightarrow y$ であれば、 $y'$ を出力
- 分類問題の場合にはさらにindexを用い、各クラスの予想確率を出力  
例: 2値分類の場合、 $y'_0$ と $y'_1$ を出力

# 評価の見方：Confusion Matrix

検証用データに対する統計的な評価指標と混同行列を表示します。  
混同行列を用いて、全体の正答数や間違いやすいラベルの傾向などを確認できます。

## Confusion Matrixページのスナップショットとその見方





# 評価の見方：Classification Result

各データごとにモデルが確率が高いと予測した上位3つのラベルと確率を表示します。  
第1候補の確率を昇順でソートすれば、モデルが判定困難なデータを確認できます。

## Classification Resultページのスナップショットとその見方

クリックして選択

ソート結果のリセット

列の↑をクリックでソートが可能

Index	x:image	y:label ↑	y__1st ↑	y__1st_value ↑	y__2nd ↑	y__2nd_value ↑	y__3rd ↑	y__3rd_value ↑
1		9	7	0.24524365	3	0.20676546	9	0.1785611
2		7	7	0.40404041	9	0.34343435	4	0.25252526
3		8	7	0.3802535	9	0.34199467	8	0.19017594
4		9	8	0.3802535	9	0.34199467	4	0.19017594
5		3	3	0.43550995	5	0.28446183	8	0.18889685



# 評価の見方：Classification Matrix (Recall)

ラベルごとにモデルが間違えた上位3つの誤判定のラベルと件数を表示します。  
Recallでは、データセット内の正解ラベルを軸に誤判定結果の確認できます。

## Classification Result (Recall) ページのスナップショットとその見方



○ Output Result ○ Confusion Matrix y - y' Recallを選択

クリックして選択

☒ Classification Matrix y - y': Recall ○ Likelihood Graph y - y'

	Recall	true-positive	false-positive 1st	false-positive 2nd	false-positive 3rd	other
y:label=0	0.9969	y'_0: 977	y'_6: 1	y'_7: 1	y'_8: 1	0
y:label=1	0.9938	y'_1: 977	y'_0: 1	y'_2: 1	y'_3: 1	0
y:label=2	0.9932	y'_2: 977	y'_1: 1	y'_0: 1	y'_3: 1	0
y:label=3	0.9941	y'_3: 1004	y'_5: 3	y'_2: 2	y'_8: 1	0
y:label=4	0.9919	y'_4: 974	y'_9: 4	y'_6: 3	y'_2: 1	0
y:label=5	0.9843	y'_5: 878	y'_3: 5	y'_6: 5	y'_0: 2	2
y:label=6	0.9906	y'_6: 949	y'_0: 2	y'_1: 2	y'_4: 2	3
y:label=7	0.9932	y'_7: 1021	y'_2: 4	y'_0: 1	y'_1: 1	1
y:label=8	0.9877	y'_8: 962	y'_0: 3	y'_2: 2	y'_7: 2	5
y:label=9	0.9782	y'_9: 987	y'_7: 7	y'_4: 5	y'_5: 3	7

セルをクリックすると  
誤判定データの詳細にジャンプ

Index	x:image	y:label ↑	y'_1st ↑	y'_1st_value ↑	y'_2nd ↑	y'_2nd_value ↑	y'_3rd ↑	y'_3rd_value ↑
1		6	4	0.9877814	9	0.006788728	6	0.005254835
2		6	4	0.5159155	6	0.4840762	5	7.991703e-06

# 評価の見方：Classification Matrix (Precision)

ラベルごとにモデルが間違えた上位3つの誤判定のラベルと件数を表示します。  
Precisionでは、モデルが判定したラベルを軸に誤判定結果の確認できます。

## Classification Result（Precision）ページのスナップショットとその見方

Output Result

Confusion Matrix

y - y'

Precisionを選択

Classification Result

y - y'

Classification Matrix

y - y': Precision

Likelihood Graph

y - y'

	Precision	F-Measures	true-positive	false-negative 1st	false-negative 2nd	false-negative 3rd	other
y'_0	0.9889	0.9929	y:label=0: 977	y:label=8: 3	y:label=5: 2	y:label=6: 2	4
		0.9947					
		0.9923	1025				
y'_3	0.9921	0.9931	y:label=3: 1004	y:label=5: 5	y:label=1: 2	y:label=6: 1	0
y'_4	0.9919	0.9919	y:label=4: 974	y:label=9: 5	y:label=6: 2	y:label=8: 1	0
y'_5	0.991	0.9876	y:label=5: 878	y:label=3: 3	y:label=9: 3	y:label=6: 1	1

Deep Learningが判定したラベル

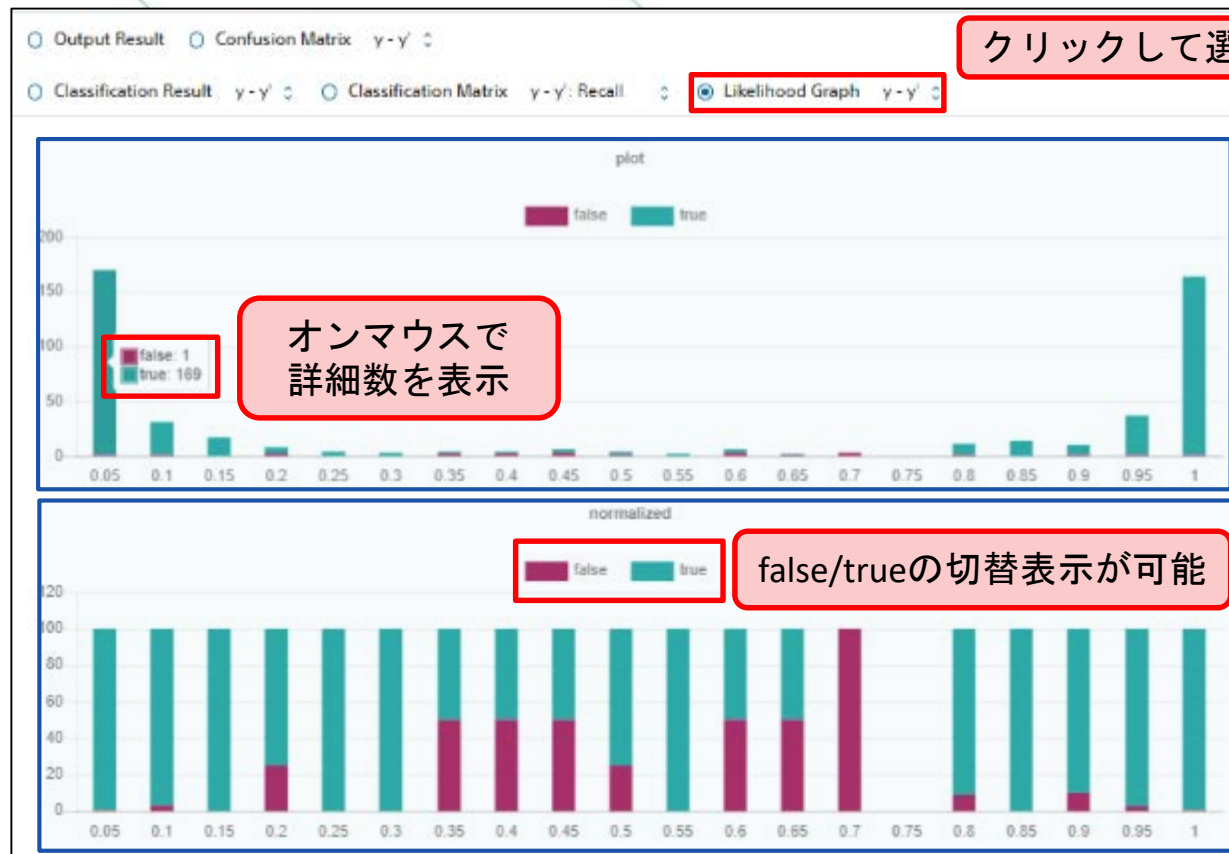
正解数

間違いやすい上位3件のラベルと件数

# 評価の見方：Likelihood Graph (Sigmoid)

スコアや確率ごとの判定結果により、各ラベルをどの程度分離できているかを確認できます。  
Sigmoidを用いた分類問題の場合には、判定スコアとその正答の対応を表示します。

## Classification Result (Sigmoid) ページのスナップショットとその見方



### plot

- 横軸：判定スコア（0～1）
- 縦軸：判定数（件）

### normalized

- 横軸：判定スコア（0～1）
- 縦軸：正誤割合（%）

# 評価の見方：Likelihood Graph (Softmax)

スコアや確率ごとの判定結果により、各ラベルをどの程度分離できているかを確認できます。  
Softmaxを用いた分類問題の場合には、判定ラベルの予測確率とその正答の対応を表示します。

## Classification Result (Softmax) ページのスナップショットとその見方



判定ラベルの予測確率が  
1/(ラベルの数)以下になること  
がないので、左側は表示がない

### plot

- 横軸：判定ラベルの予測確率 (0~1)
- 縦軸：判定数 (件)

### normalized

- 横軸：判定ラベルの予測確率 (0~1)
- 縦軸：正誤割合 (%)



# 判断根拠の可視化

問題設定が画像分類の場合、作成したモデルが画像のどの箇所に着目し判断したかを、以下に示す2つの方法で可視化することができます。

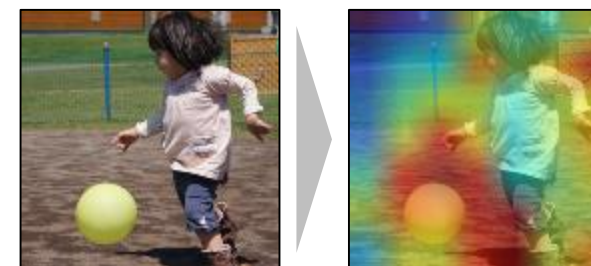
Grad-CAMはConvolutionレイヤーを含んだCNNモデルでしか動作しないため、ご注意ください。

## 判断根拠の可視化手法

	手法	出力画像
1	Grad-CAM	入力画像の上に、判断時に注視した箇所をヒートマップで描画
2	LIME	入力画像の中で判断時の着目箇所だけを描画

## 判断根拠の可視化の実行例

Grad-CAM



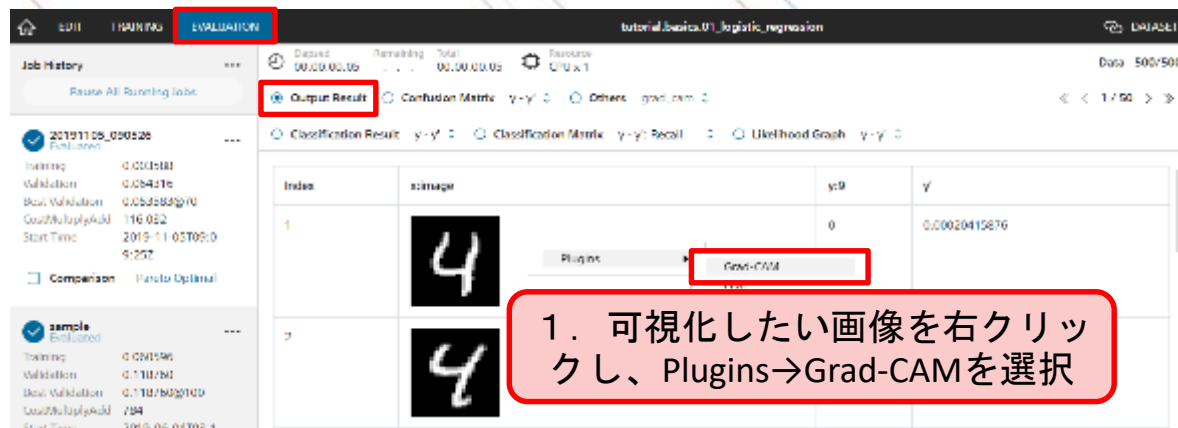
LIME



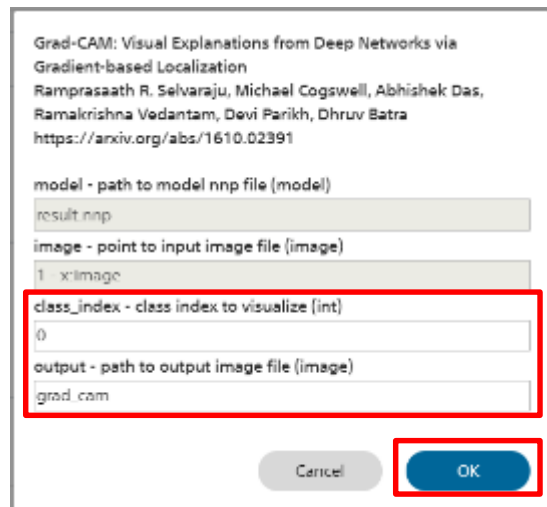
# 判断根拠の可視化: Grad-CAM



EvaluationタブのOutput Resultのページから、可視化したい画像を選択し、Pluginsから実行します。



1. 可視化したい画像を右クリックし、Plugins→Grad-CAMを選択



2. ポップアップでパラメータを設定し、“OK”を押下

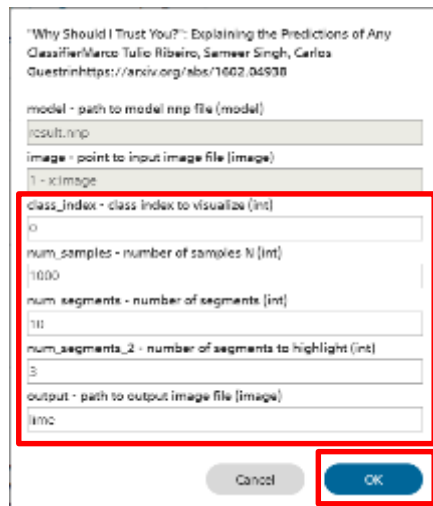
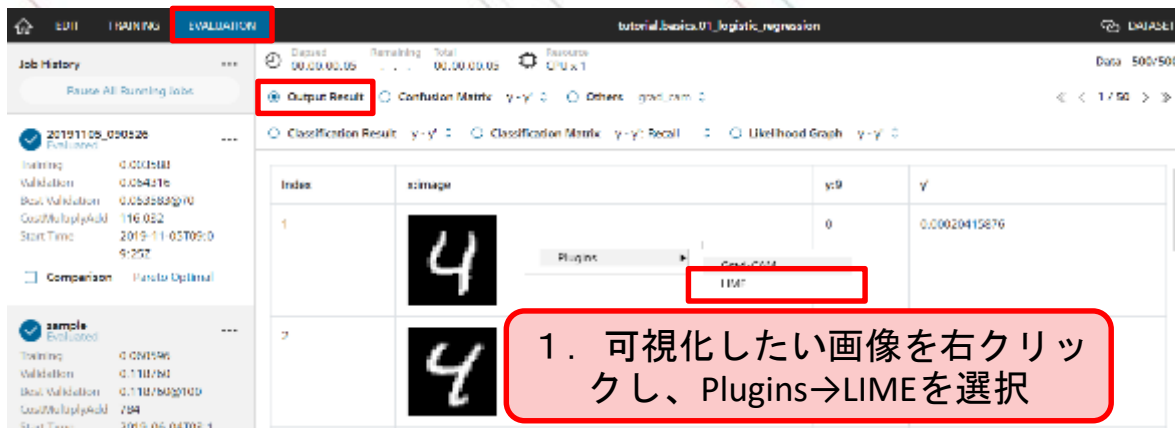
## ポップアップの設定項目

項目	説明
class_index	判断箇所を描画するクラスを指定。基本的には正解クラスを入力。この例では4と入力するとモデルがどの箇所を見て4と判断したかが画像からわかる。
output	出力ファイル名を指定。画像ファイルをダウンロードしたい場合は作成された画像をクリックすることで、outputで入力した名前で保存される。

# 判断根拠の可視化: LIME



EvaluationタブのOutput Resultのページから、可視化したい画像を選択し、Pluginsから実行します。



## ポップアップの設定項目

項目	説明
class_index	判断箇所を描画するクラスを指定。基本的には正解クラスを入力。この例では4と入力するとモデルがどの箇所を見て4と判断したかが画像からわかる。
num_samples	選択画像と比較するデータの件数を指定。数値が大きすぎると処理に時間がかかるため注意が必要。
num_segments	画像を分割する数を指定。
num_segments_2	分割領域を注視した順に並べた際に上位いくつまで表示するかを指定。
output	出力ファイル名を指定。画像ファイルをダウンロードしたい場合は作成された画像をクリックすることで、outputで入力した名前で作成される。

# 構造自動探索機能



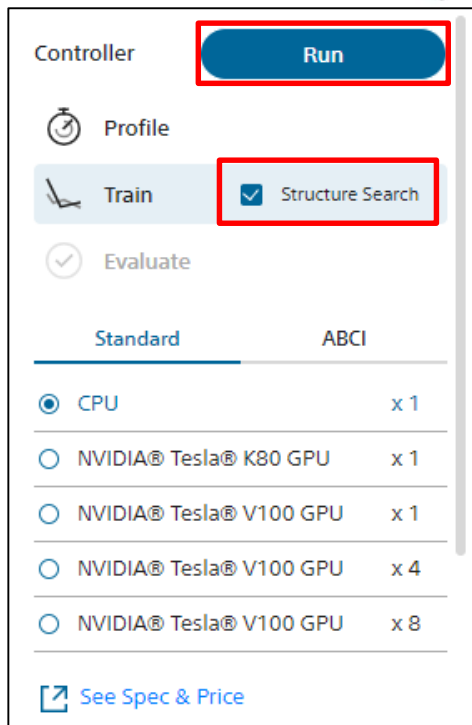
構造自動探索機能は、AIによるネットワークの編集と学習が自動で繰り返され、ネットワーク作成の試行錯誤を自動的に行います。

EDITページのStructure Searchのチェックボックスを選択し、Runボタンで評価が実行されます※。

各モデルの結果はTRAININGページのTrade-off Graphで比較することが可能です。

## 構造探索機能の実行の方法

## 構造探索機能の結果



ネットワークを自動生成

※終了条件など細かな設定は[Global Config](#)で設定をします。



# 目次

1

アカウントサインイン

2

データセットのアップロード

3

モデル作成

4

推論実行

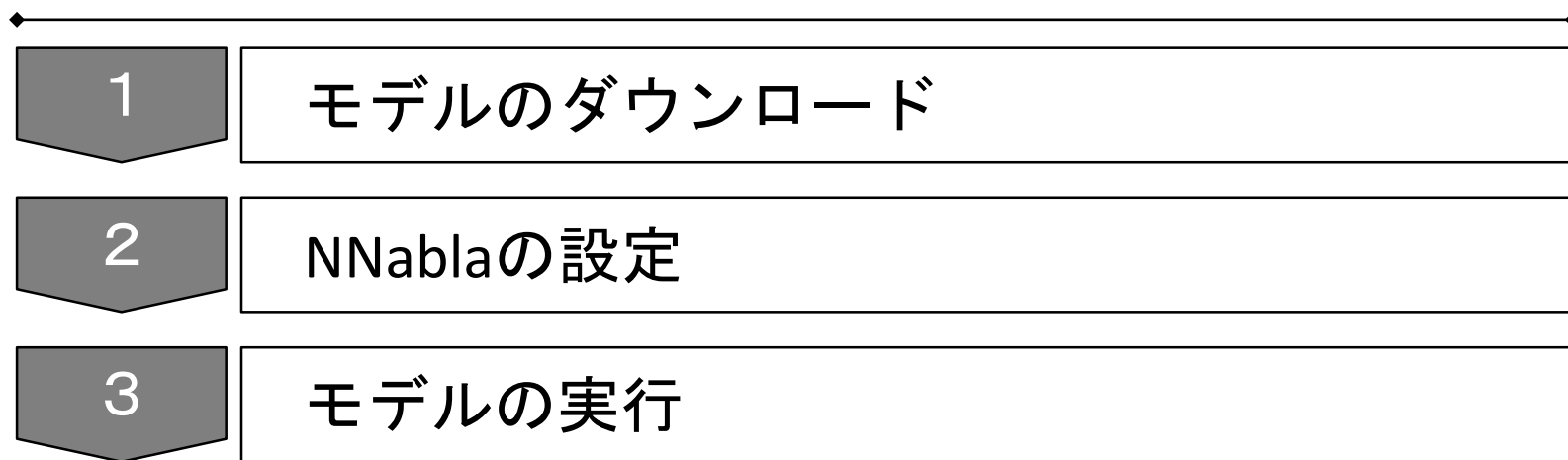
# 推論実行のステップ

NNCから作成したモデルをダウンロードすることでお客様の環境で自由にモデル利用ができます。

モデルを実行するためには、NNablaが必要になります。

NNablaを用いることでコマンドラインやpythonなど様々な方法で画像分類モデルが実行可能となります。

## 推論実行のステップ



# 画像分類モデルのダウンロード

モデルの  
ダウンロード

NNablaの設定

モデルの実行

学習が完了し最適なモデルを作成した後は、「Job History」の中から該当のモデルを右クリックし、選択肢の中の「Download」をクリックすることでモデルをダウンロードできます。

NNP、NNB、ONNXは作成したネットワークと学習済みパラメータの値が含まれたファイルで、実行方法に応じて使い分けをします(詳細は[モデルの実行方法](#))。また、html betaは学習結果などの内容をhtml形式で出力したものです。

作成したモデルの権利は作成者に帰属し、自由にDeep Learningモデルを利用することができます。

The screenshot shows the 'Job History' tab in the Neural Network Console. A context menu is open for the job '20190409\_051421'. The 'Download' option is highlighted, and a sub-menu is displayed showing the following options:

- NNP(Neural Network Libraries file format)
- NNB(NNabla C Runtime file format)
- ONNX
- html beta

# NNablaの設定

モデルの  
ダウンロード

NNablaの設定

モデルの実行

任意のPCにNNablaをインストールします。

NNablaのインストールについては、以下のドキュメントをご参照ください。

<http://nnabla.readthedocs.io/en/latest/python/installation.html>



# モデルの実行方法

モデルの  
ダウンロード

NNablaの設定

モデルの実行

NNablaを用いてモデルを実行する方法は、使用する言語に応じて様々な方法があります。  
また、ONNXを利用することで、他のDeep Learningのフレームワークを利用することも可能です。  
次頁以降では、コマンドラインとPythonで実行する方法を解説いたします。

	実行方法	GPU 利用	特徴	ダウンロード ファイル	参考URL
1	コマンドライン	可能	最も簡単に利用可能	NNPファイル	<a href="https://support.dl.sony.com/docs-ja/チュートリアル：neural-network-consoleによる学習済みニューラ/">https://support.dl.sony.com/docs-ja/チュートリアル：neural-network-consoleによる学習済みニューラ/</a>
2	Python	可能	比較的容易に利用可能	NNPファイル	<a href="https://support.dl.sony.com/docs-ja/チュートリアル：neural-network-consoleによる学習済みニューラ/">https://support.dl.sony.com/docs-ja/チュートリアル：neural-network-consoleによる学習済みニューラ/</a>
3	C++	可能	推論環境にPythonのインストールが不要	NNPファイル	<a href="https://github.com/sony/nnabla/tree/master/examples/cpp/mnist_runtime">https://github.com/sony/nnabla/tree/master/examples/cpp/mnist_runtime</a>
4	C	不可	非常にコンパクトであり、組み込み利用向き	NNBファイル	<a href="https://github.com/sony/nnabla-c-runtime">https://github.com/sony/nnabla-c-runtime</a>
5	他Deep Learning フレームワーク	環境 依存	環境依存	ONNXファイル	<a href="https://nnabla.readthedocs.io/en/latest/python/file_format_converter/file_format_converter.html">https://nnabla.readthedocs.io/en/latest/python/file_format_converter/file_format_converter.html</a>
6	TensorFlow	環境 依存	環境依存	TensorFlow frozen graphファイル	TensorFlowのウェブページをご覧ください

次頁に  
解説あり

# コマンドラインでの推論実行

モデルの  
ダウンロード

NNablaの設定

モデルの実行

NNablaのインストールされたPython環境で、コマンドラインから以下を実行します。

```
nnabla_cli forward ¥  
-c [ダウンロードしたネットワークモデルファイル(*.nnp)] ¥  
-d [推論をするデータセットを取りまとめたCSVファイル] ¥  
-o [推論結果の出力ディレクトリ]
```

※ Neural Network ConsoleのEVALUATIONタブでの推論実行時に同様のコマンドを使用しているため、ログの出力ウインドウに同様のものが出力されています。

```
2017-10-24 05:54:28,942 [worker]: [INFO]: nnabla_cli forward -c  
/home/nnabla/results/results_current_100.nnp -d ccbf15a0-bcb6-4ba6-b10e-  
27fc877c4348/1002/index.csv -o /home/nnabla/results
```

※NNPファイルには推論用のネットワークのほかに、学習用のネットワークや学習用パラメータなども保存されています。  
nnabla\_cliの他のオプションを利用することで、コマンドラインから再学習などを実施することもできます。  
詳細はnnabla\_cli -hからご確認ください。

# Pythonでの実行方法

モデルの  
ダウンロード

NNablaの設定

モデルの実行

ダウンロードしたネットワークファイルをNNablaで実行可能な形に変換し、  
変換後のネットワークファイルをPythonで読み込んで利用します。

コマンドラインから以下を実行し、RepeatなどNNC専用のレイヤーをNNLに対応するようネットワークファイルを変換します

```
nnabla_cli convert result.nnp result_expanded.nnp
```

変換したネットワークファイルをPythonで読み込んで利用します

```
# NNablaのインポート
import nnabla as nn
from nnabla.utils import nnp_graph
# nnpファイルの読み込み、ネットワークモデルの取り出し
nnpFile = nnp_graph.NnpLoader('./result_expanded.nnp')
networkModel = nnpFile.get_network('MainRuntime ', batch_size=1)
# 入出力レイヤーの名前を取得
inputName = list(networkModel.inputs.keys())[0]
outputName = list(networkModel.outputs.keys())[0]
# 入出力レイヤーの数値変数を取得
x = networkModel.inputs[inputName]
y = networkModel.outputs[outputName]
# 推論の実行
x.d = np.array(img) * (1.0 / 255.0)
y.forward(clear_buffer=True)
# 推論結果の表示
print(y.d[0])
```

※NNablaを用いたNNPファイルの利用方法の詳細は、[ウェブページ](#)をご確認ください。



# Appendix

---



# SONYアカウントの取得方法

アカウント作成ページに移動し、メールアドレスやパスワードなどを設定します。

## 1. 作成ページへの移動 1

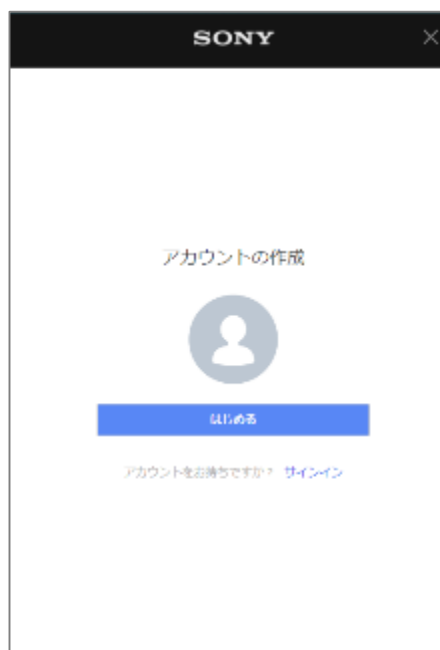
✓ 「新しいアカウントの作成」を押下



The screenshot shows the Sony login page. At the top, it says "サインイン" (Sign In). Below that, there's a link "このアカウントで、Sonyグループの様々なサービスへアクセスをっと詳しく" (Access various Sony Group services with this account, learn more). The main form has two input fields: "サインインID" (Sign In ID) with a sub-label "Eメールアドレス" (Email address), and "パスワード" (Password). There are toggle icons for "パスワードの表示/非表示" (Show/Hide password). A blue "サインイン" (Sign In) button is at the bottom. Below the button, it says "サインインがお困りですか?" (Having trouble signing in?) and a link "新しいアカウントの作成" (Create new account).

## 2. 作成ページへの移動 2

✓ 「はじめる」を押下



The screenshot shows the Sony account creation page. At the top, it says "アカウントの作成" (Account creation). In the center, there's a large blue circle with a white person icon. Below it is a blue button labeled "はじめる" (Get started). At the bottom, it says "アカウントをお持ちですか? サインイン" (Do you have an account? Sign in).

## 3. メールアドレス等の入力

✓ 登録するメールアドレスとパスワードを入力



The screenshot shows the Sony account creation page with input fields. At the top, it says "アカウントの作成" (Account creation). Below that, there's a progress indicator "1/3". The main form has two input fields: "サインインID" (Sign In ID) with a sub-label "youraddress@example.com", and "パスワード" (Password). There's a toggle icon for "パスワードの表示/非表示" (Show/Hide password). Below the password field, it says "パスワードの再入力" (Re-enter password). At the bottom, there's a blue "次へ" (Next) button and a link "戻る" (Back).

# SONYアカウントの取得方法

生年月日などを入力し、利用規約などの確認を行います。

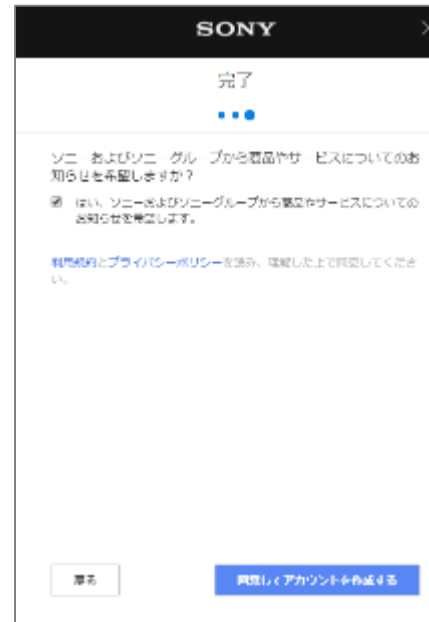
## 4. 生年月日の入力

✓国/地域、言語、生年月日を入力



## 5. 利用規約への同意

✓メール配信の有無を選択  
✓利用規約・アカウントポリシーの確認



## 6. セキュリティ認証

✓「私はロボットではありません」を押下  
※画像選択が表示された場合には指示に従う



# SONYアカウントの取得方法

確認メールを受信し、アカウントの有効化を行います。

## 7. 確認メールの送付

- ✓登録したメールアドレス宛に確認メールが送付される



## 8. 確認メールの確認

- ✓確認メールを開き、「確認する」を押下

