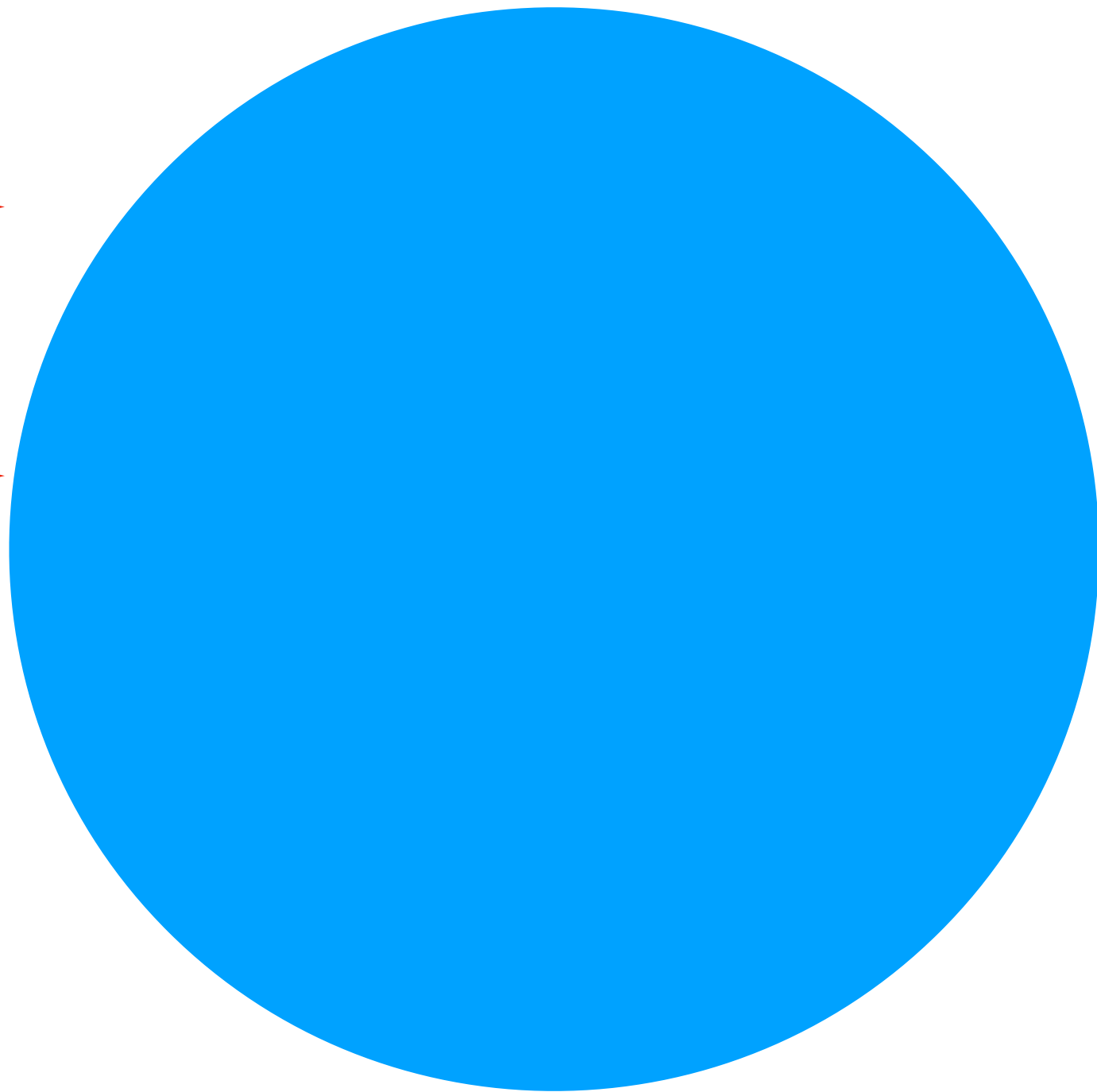
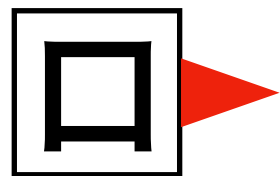
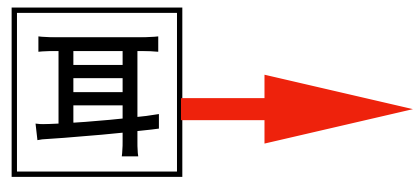
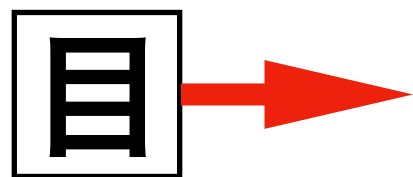
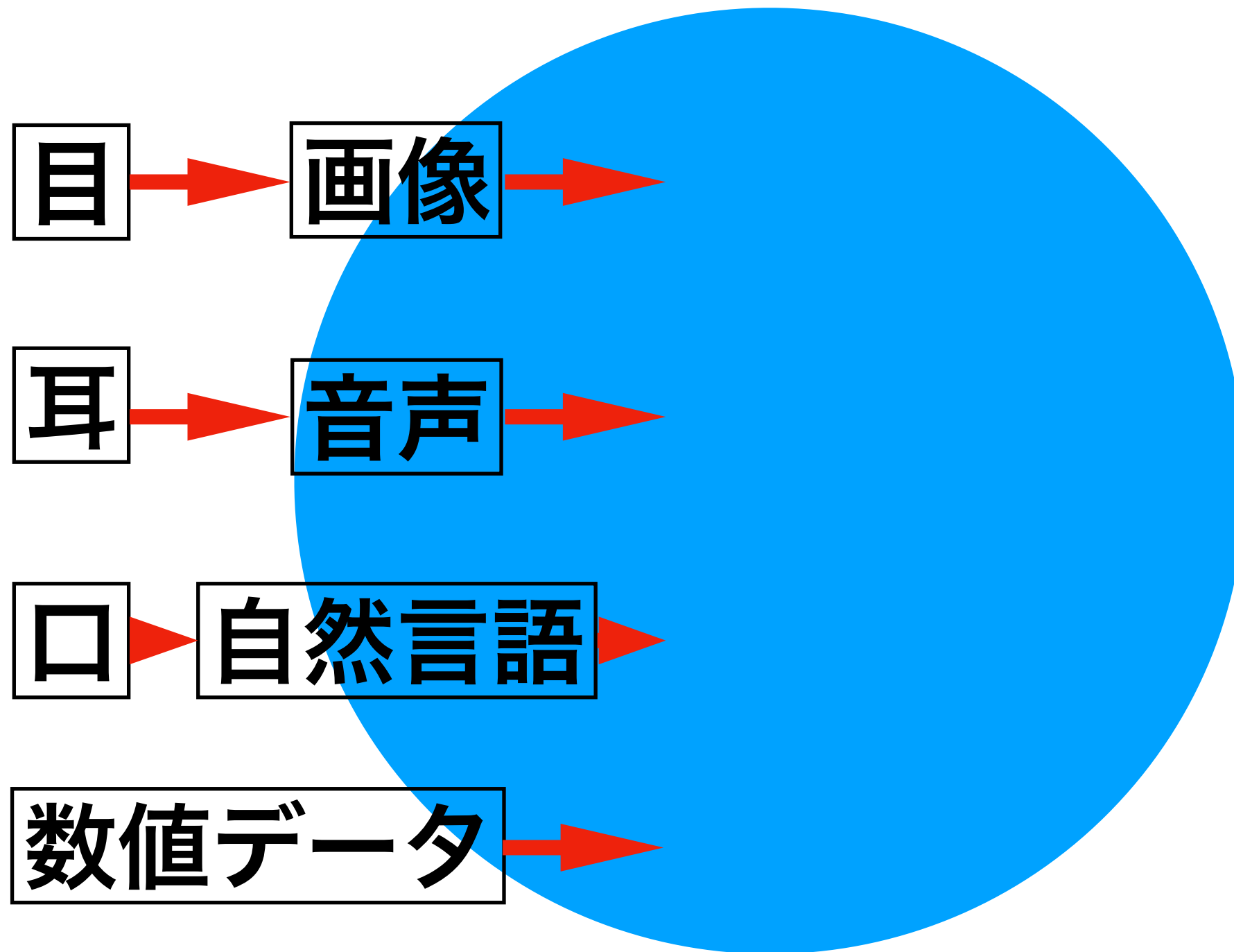


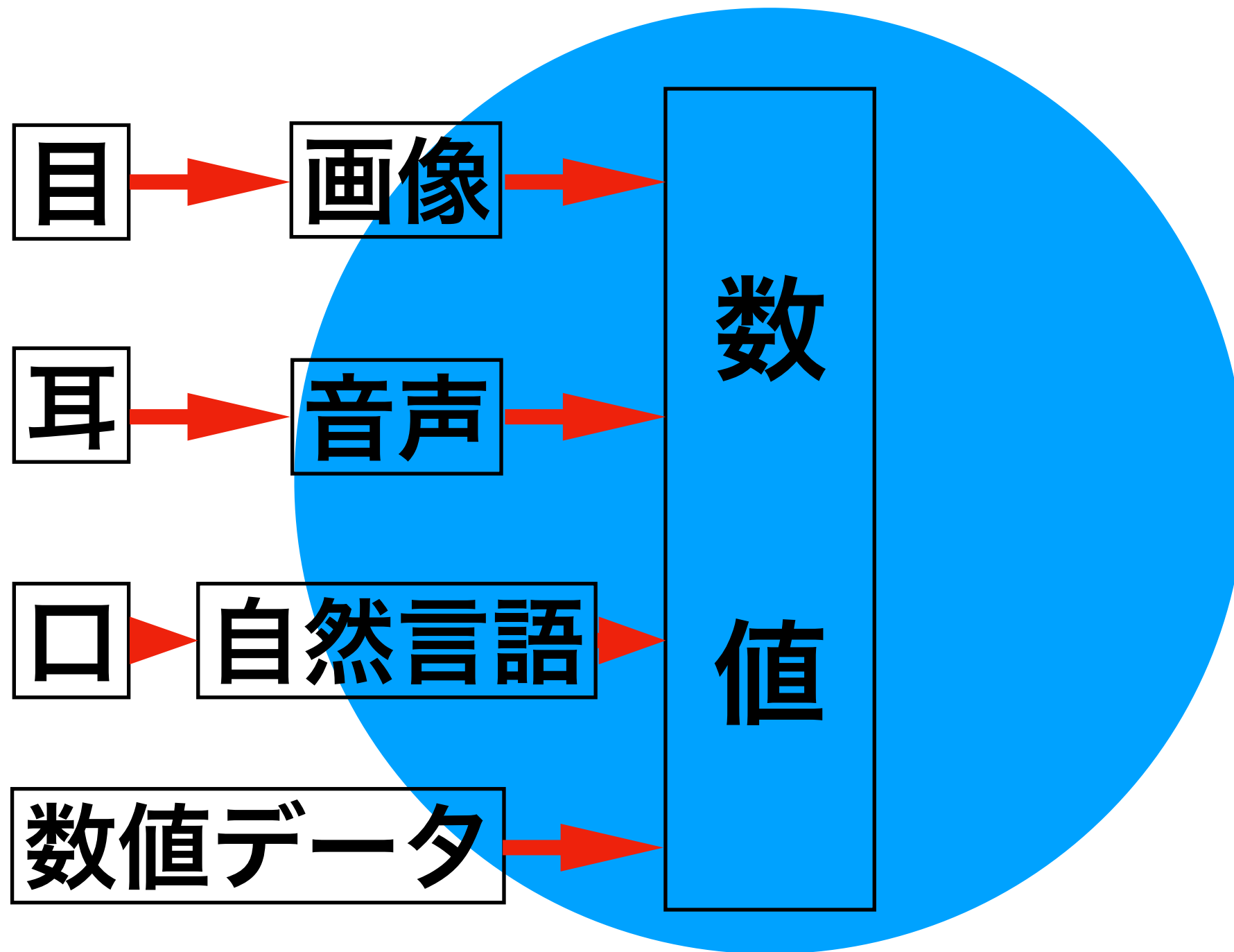
# 機械学習とは



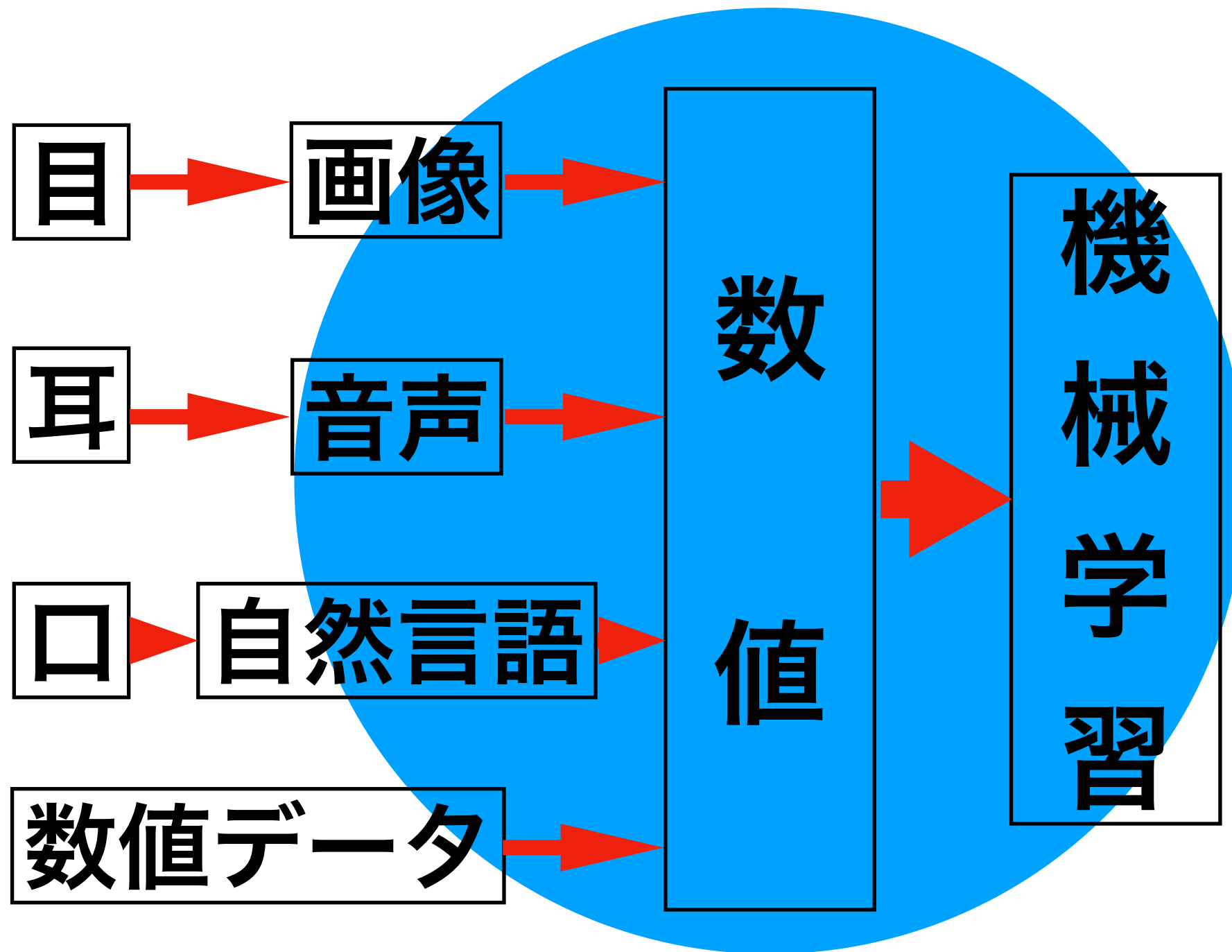
# 機械学習とは



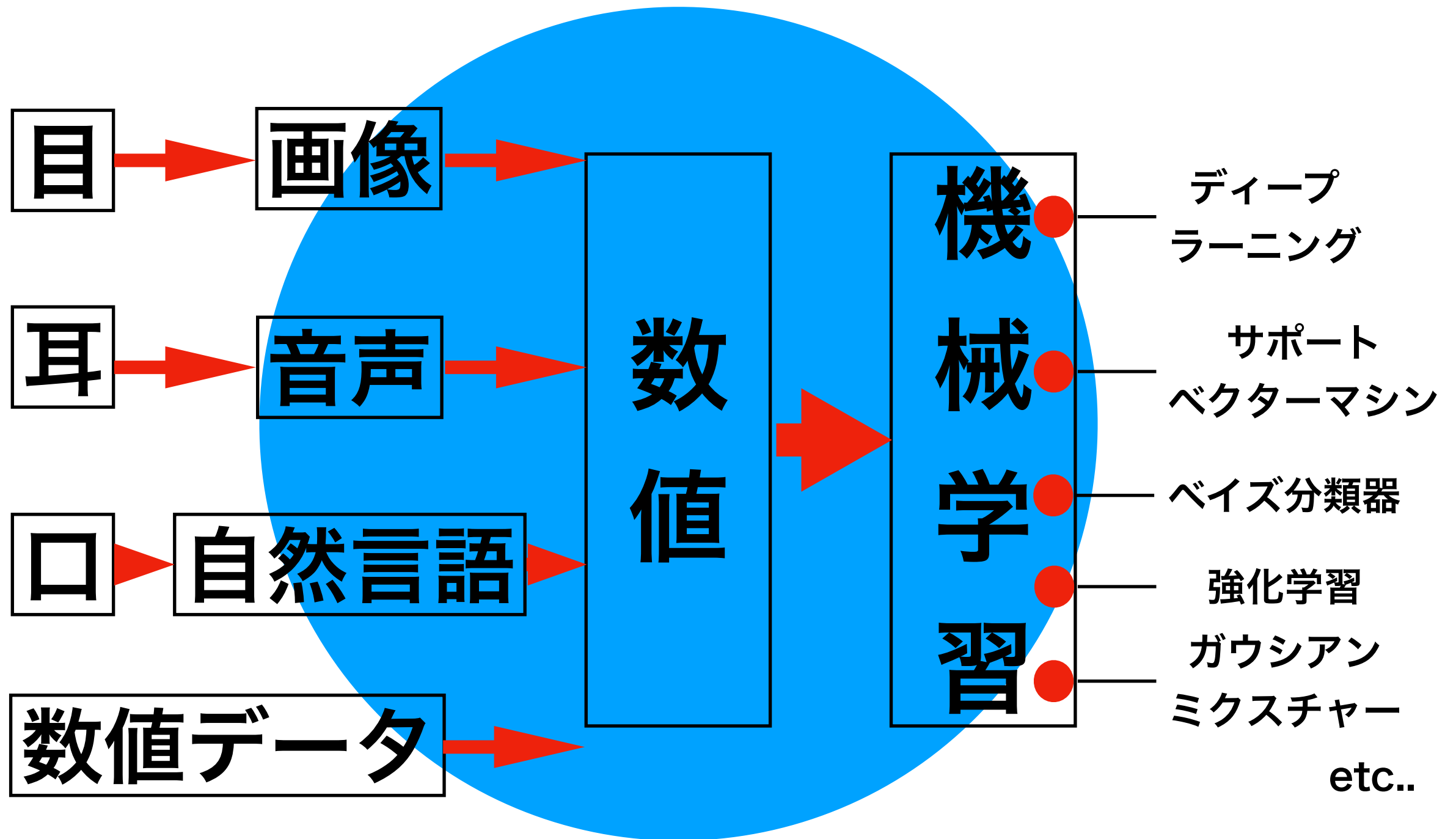
# 機械学習とは



# 機械学習とは

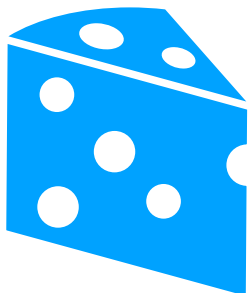
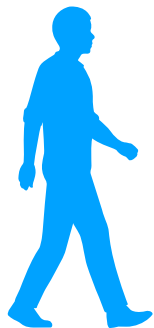
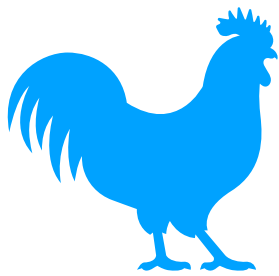


# 機械学習とは



# 学習と推論

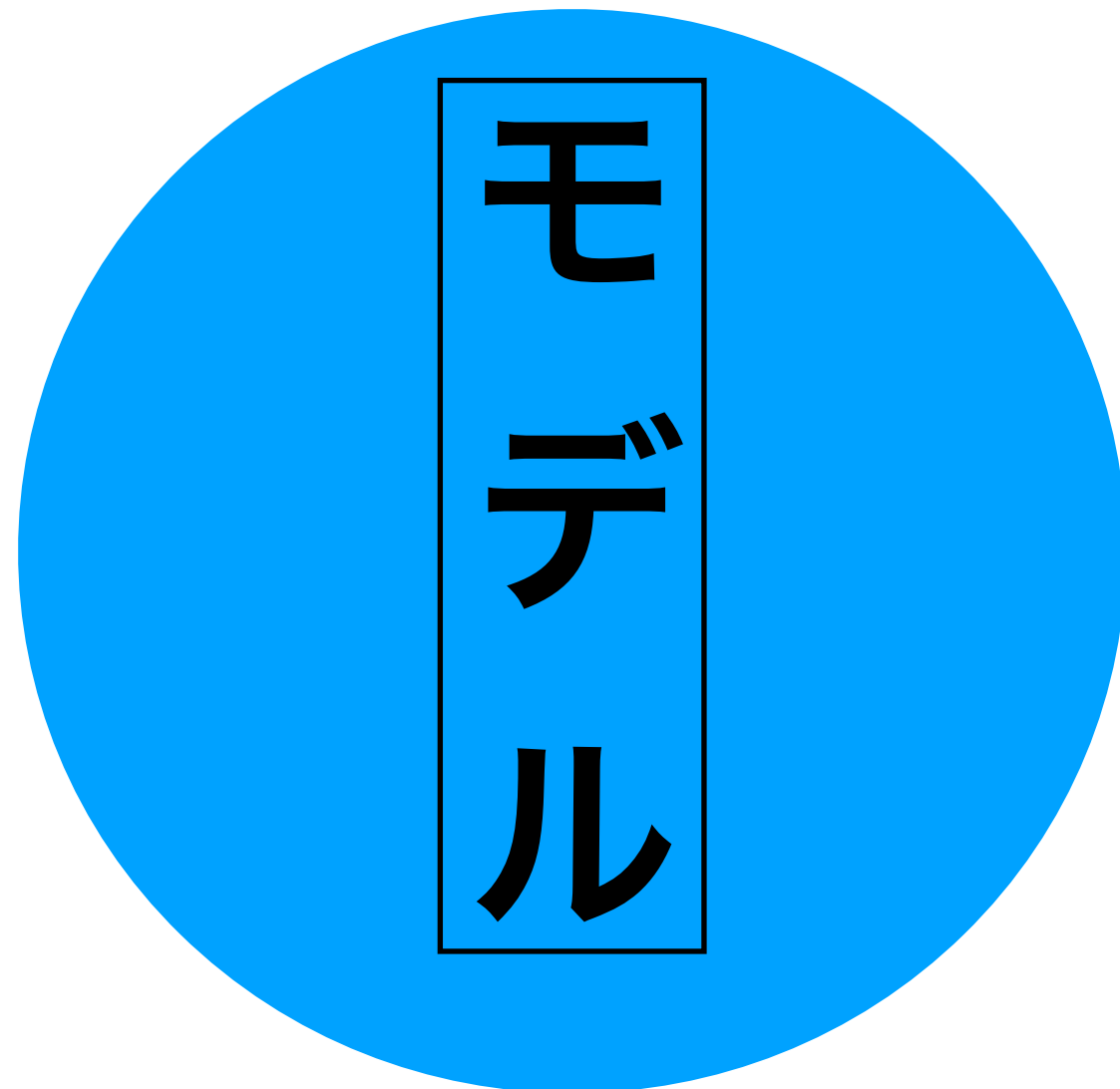
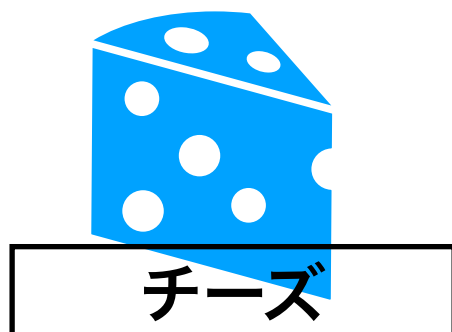
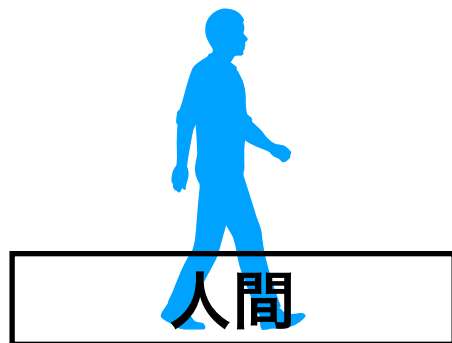
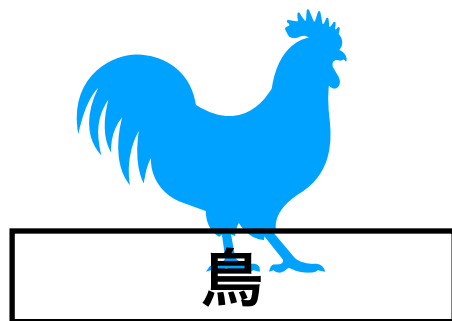
機械学習では教師ありデータをもとにトレーニングを行いモデルを作る。  
できたモデルを学習済みモデルという



モデル

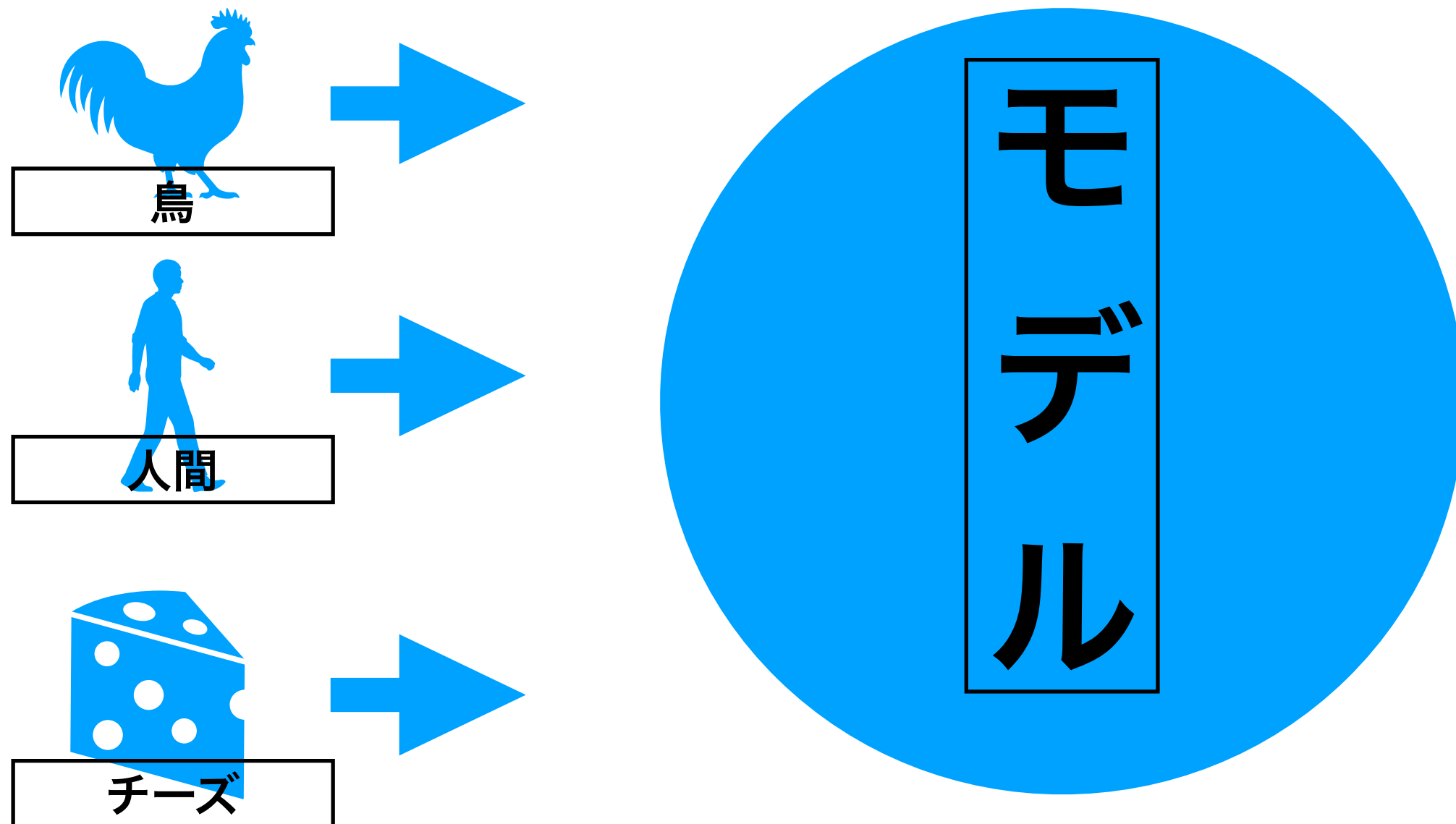
# 学習と推論

機械学習では教師ありデータをもとにトレーニングを行いモデルを作る。  
できたモデルを学習済みモデルという



# 学習と推論

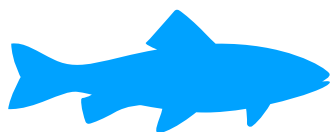
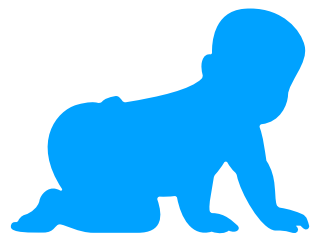
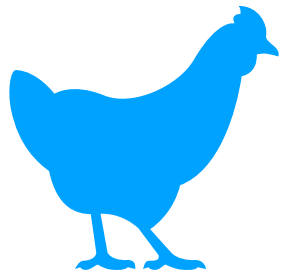
機械学習では教師ありデータをもとにトレーニングを行いモデルを作る。  
できたモデルを学習済みモデルという





# 学習と推論

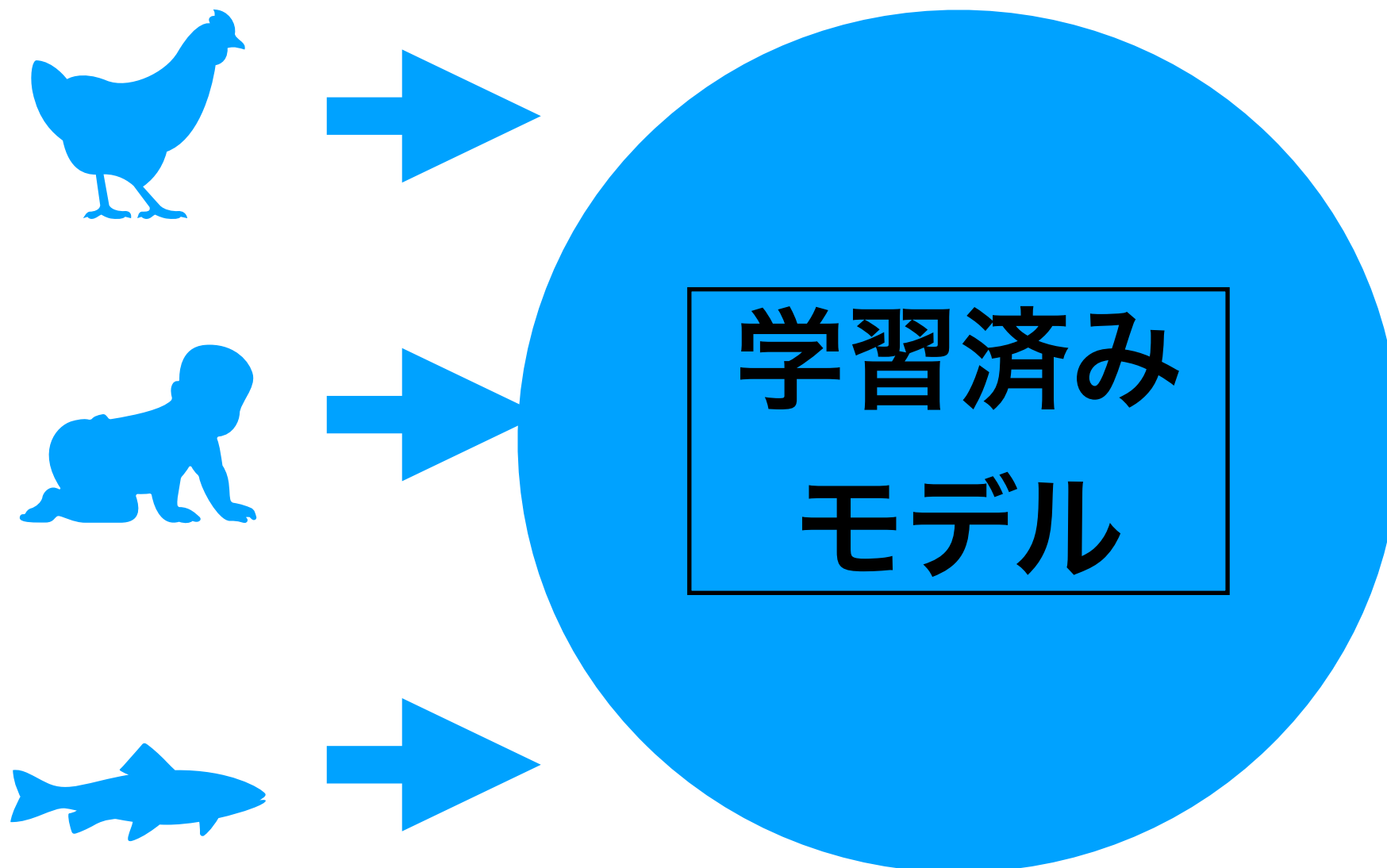
学習済みモデルから答えを取り出す。学習データに該当するデータがないとうまく識別できない。



学習済み  
モデル

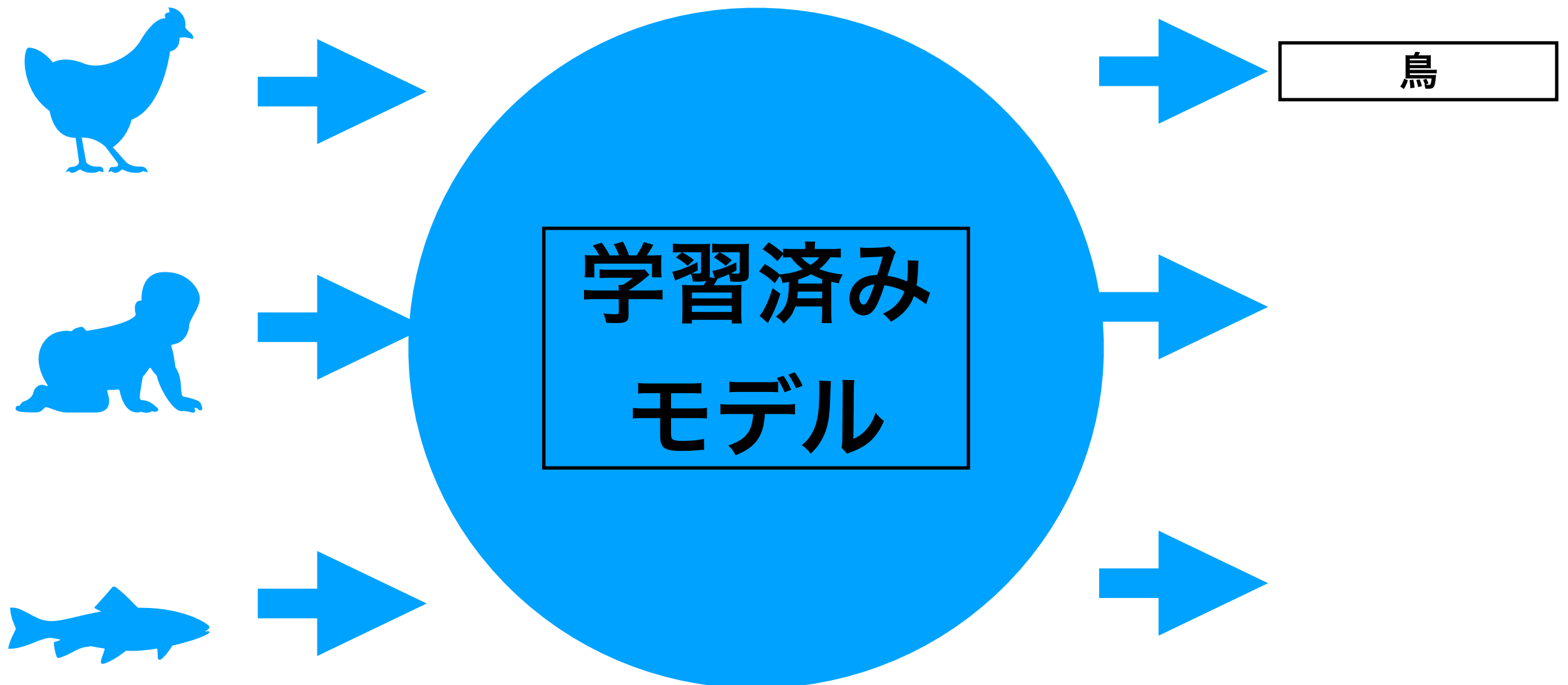
# 学習と推論

学習済みモデルから答えを取り出す。学習データに該当するデータがないとうまく識別できない。



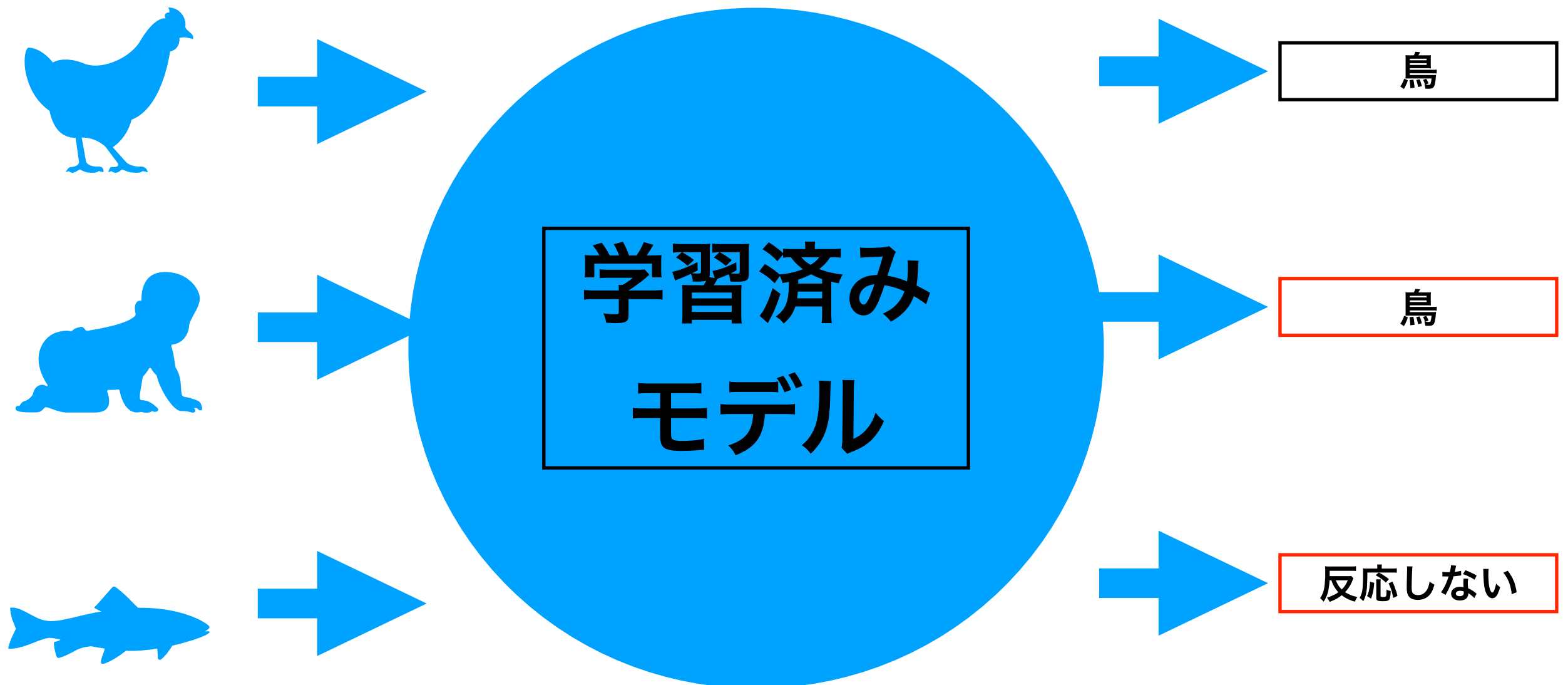
# 学習と推論

学習済みモデルから答えを取り出す。学習データに該当するデータがないとうまく識別できない。



# 学習と推論

学習済みモデルから答えを取り出す。学習データに該当するデータがないとうまく識別できない。

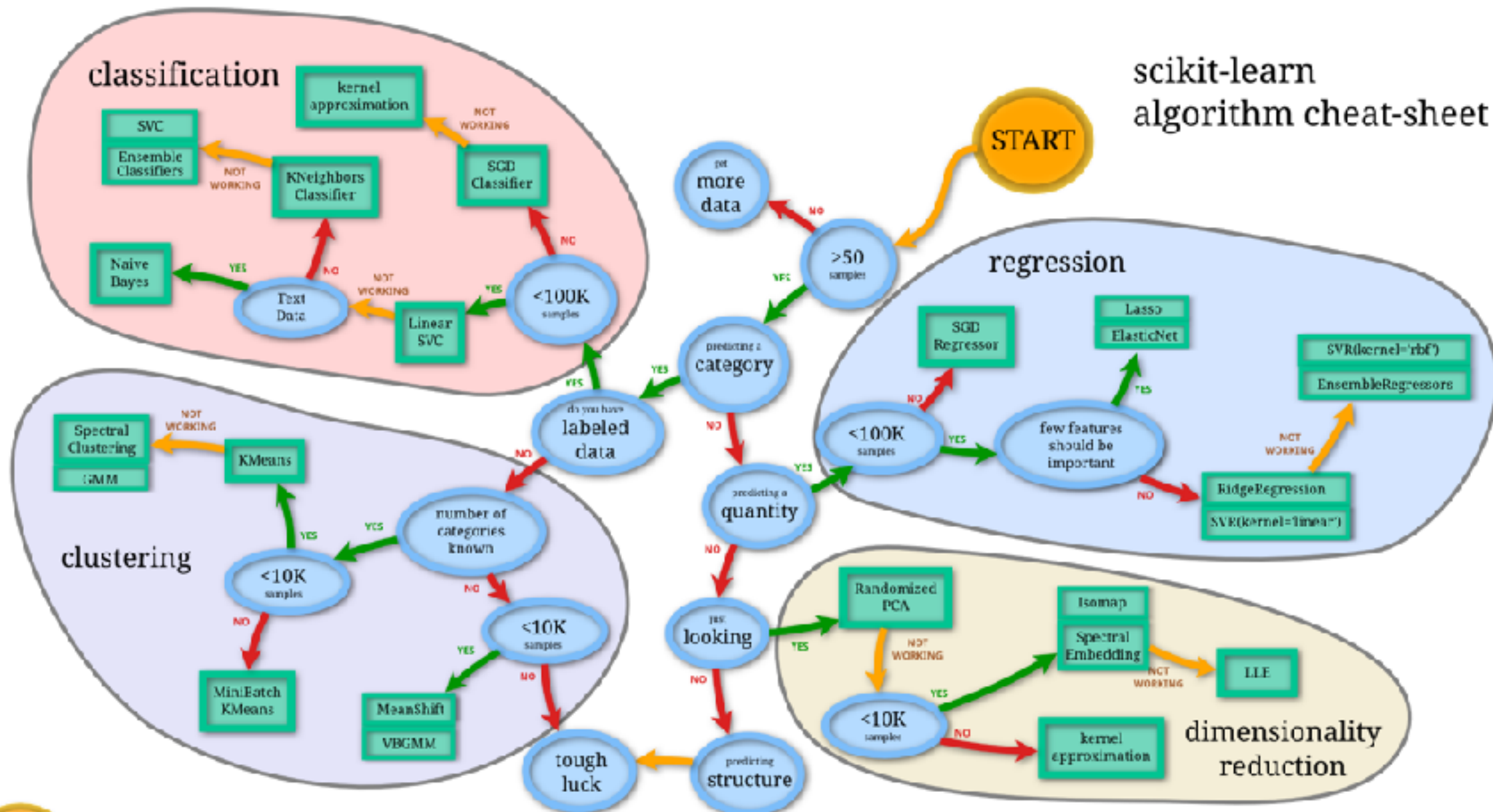


# 機械学習のフレームワーク有名どころ (python)

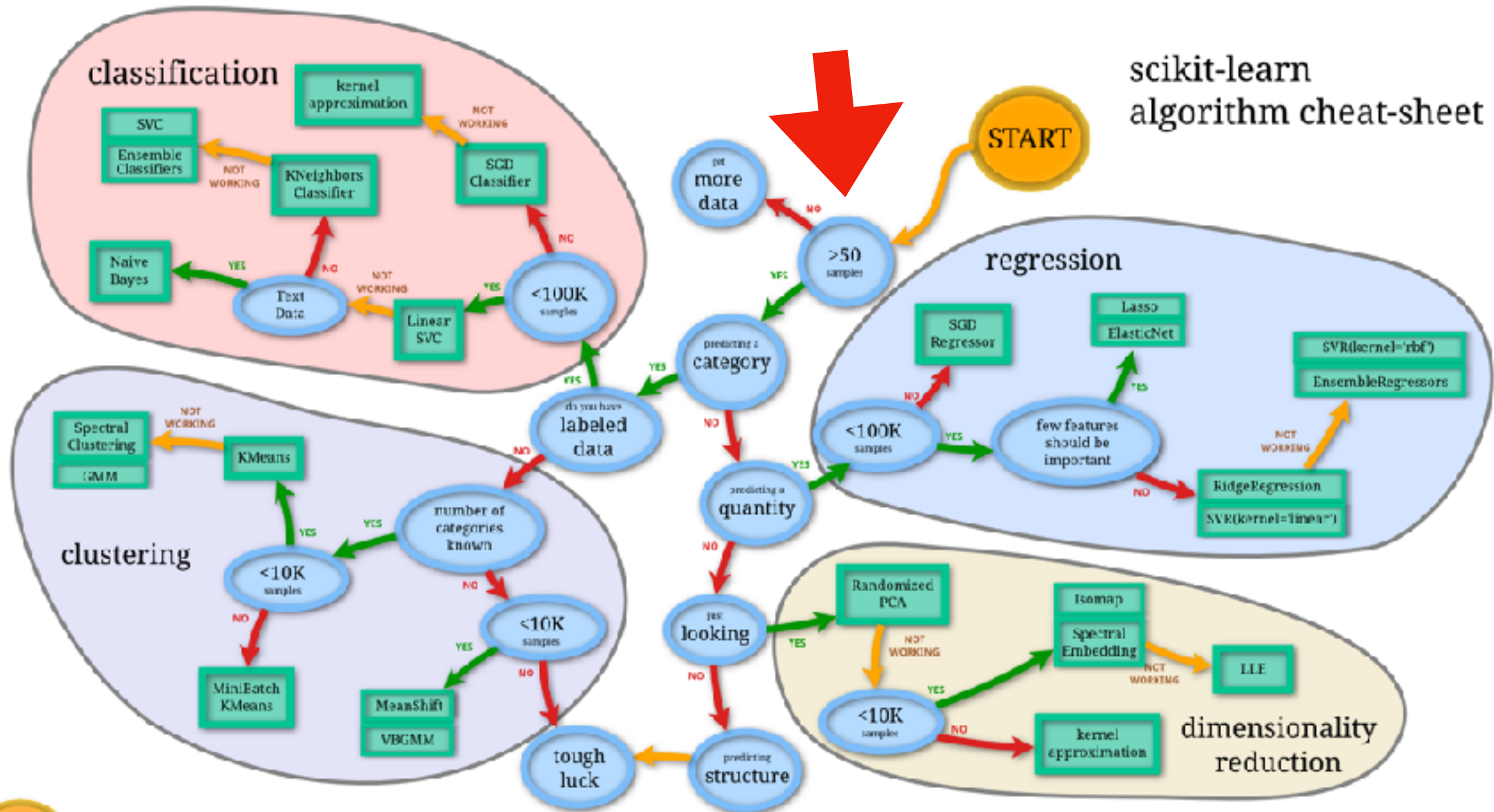
- scikit-learn (サイキットラーン)  
データマイニング (データ分析) で広く使われている。機械学習の標準。
- TensorFlow (テンソールフロー)  
世界的に一番普及している?。ソースが多い。Keras (TensorFlowなど簡単にかけるライブラリー)。クラウドサーバーも対応している。
- Chainer (ちえいなー)  
日本製。日本語ソースが豊富。azureとawsは使えるっぽい。

# scikit-learnの使いどころ

scikit-learn  
algorithm cheat-sheet



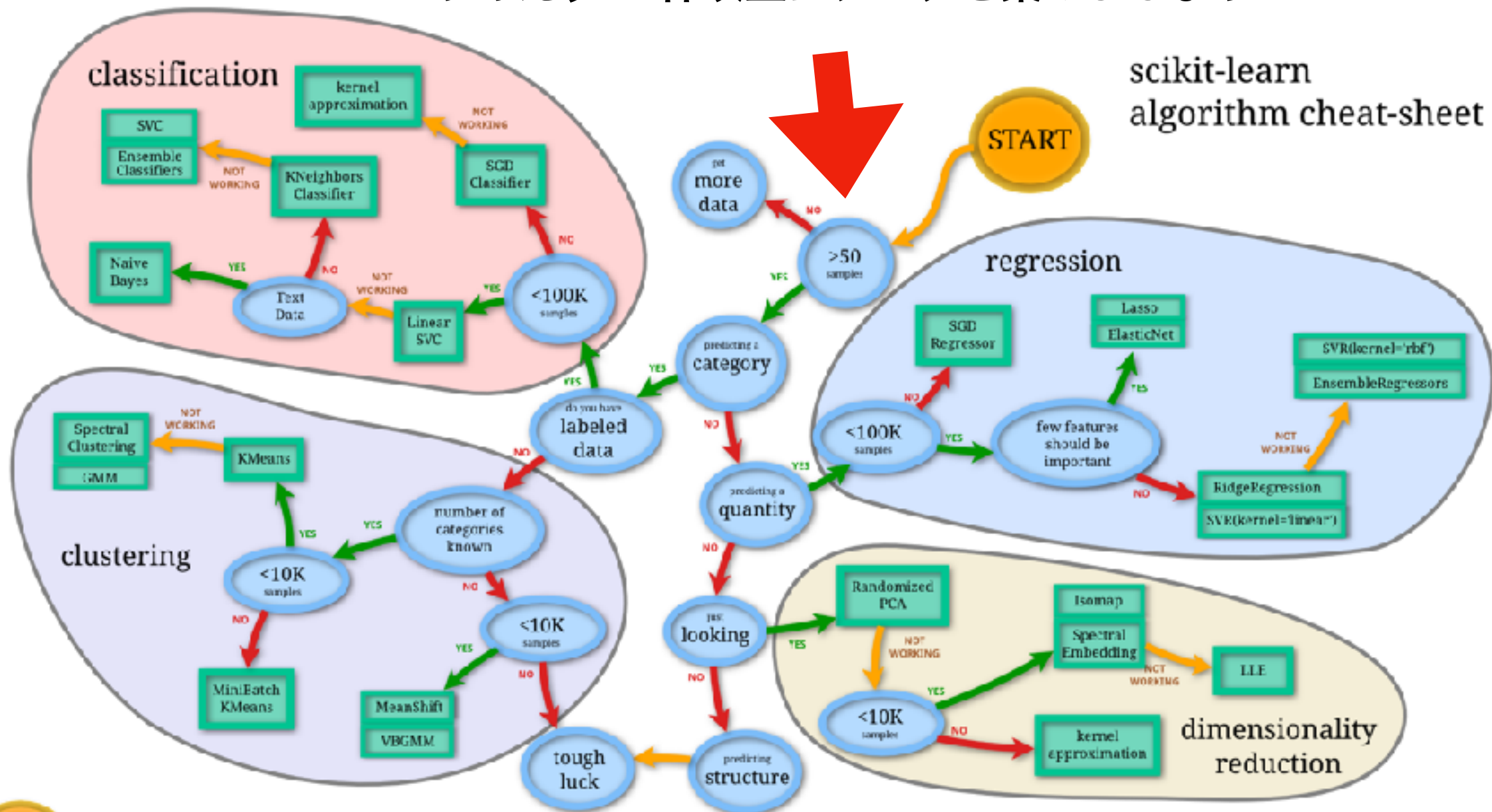
# scikit-learnの使いどころ





# scikit-learnの使いどころ

とりあえず50件以上はデータを集めましょう





# クラウドサービス（GCPで画像系）

- CLOUD VISION API

Google Cloud ML APIsの一つ。Googleの学習済みモデルを使える。

- CLOUD MACHINE LEARNING ENGINE

TensorFlow,scikit-learn使って学習モデル作り

真ん中にもう一つありますが。。。。

# 無料のデータ環境

- COLAB

GCP系の無料jupyter環境でGPUも使えるが  
共有なのでマシンパワーの奪い合いになるとか

- Azure Notebook

以前使った . . . .

お金のある人は有料環境を使ったほうが無難です。

# SSD とは

畳み込みニューラルネットワークと 並行して別の処理をしているようです。画像全体の検出位置をスライドしながら物体認識を並行して行っているとのこと。

# バージョンに合わせてインストール

- kerasでやってみます。
- [https://github.com/rykov8/ssd\\_keras](https://github.com/rykov8/ssd_keras)
- 参照:
- <https://qiita.com/PonDad/items/6f9e6d9397951cad6be>

# 合わせるバージョンは

Keras v1.2.2, Tensorflow v1.0.0, OpenCV v3.1.0-dev

## A port of **SSD: Single Shot MultiBox Detector** to Keras framework.

For more details, please refer to [arXiv paper](#). For forward pass for 300x300 model, please, follow `SSD.ipynb` for examples. For training procedure for 300x300 model, please, follow `SSD_training.ipynb` for examples. Moreover, in `testing_utils` folder there is a useful script to test `SSD` on video or on camera input.

Weights are ported from the original models and are available [here](#). You need `weights_SSD300.hdf5`, `weights_300x300_old.hdf5` is for the old version of architecture with 3x3 convolution for `pool6`.

This code was tested with Keras v1.2.2, Tensorflow v1.0.0, OpenCV v3.1.0-dev

# コマンド

Keras v1.2.2, Tensorflow v1.0.0, OpenCV v3.1.0-dev

```
$ pip install keras==1.2.2
```

```
$ pip install tensorflow==1.0.0
```

```
$ pip install opencv-python
```

conda環境で実行、opencvのバージョンがみつけれませんでした。m(\_ \_)m Ansibleみたいなやり方もあるとは思いますが誰か教えてください。

# コード修正（簡略化の波）

- 結局、このバージョン設定がないがため（だと思われる）ソースコードの修正が発生。

`cv2.cv.CV_CAP_PROP_FRAME_WIDTH`

↓ 簡略化されてる※Qiitaで書かれてる。

`cv2.CAP_PROP_FRAME_WIDTH`

↓ さらに簡略化されてる※Qiitaで書かれてる。