

# 音楽における深層学習 ~導入~

## Deep-people #1

- **一緒に深層学習つよくなるうぜ！な会**
  - LSPCはDeepにつよくない...
  - だからみんなで勉強して知見を貯めよう
- **ゴール：院試の計画のモデルを実装・カスタマイズ可能な状態になる**
  - 深層学習/対象とする音・音楽の処理について一通りは知っている
  - PyTorchで実験を行うプログラムを書ける

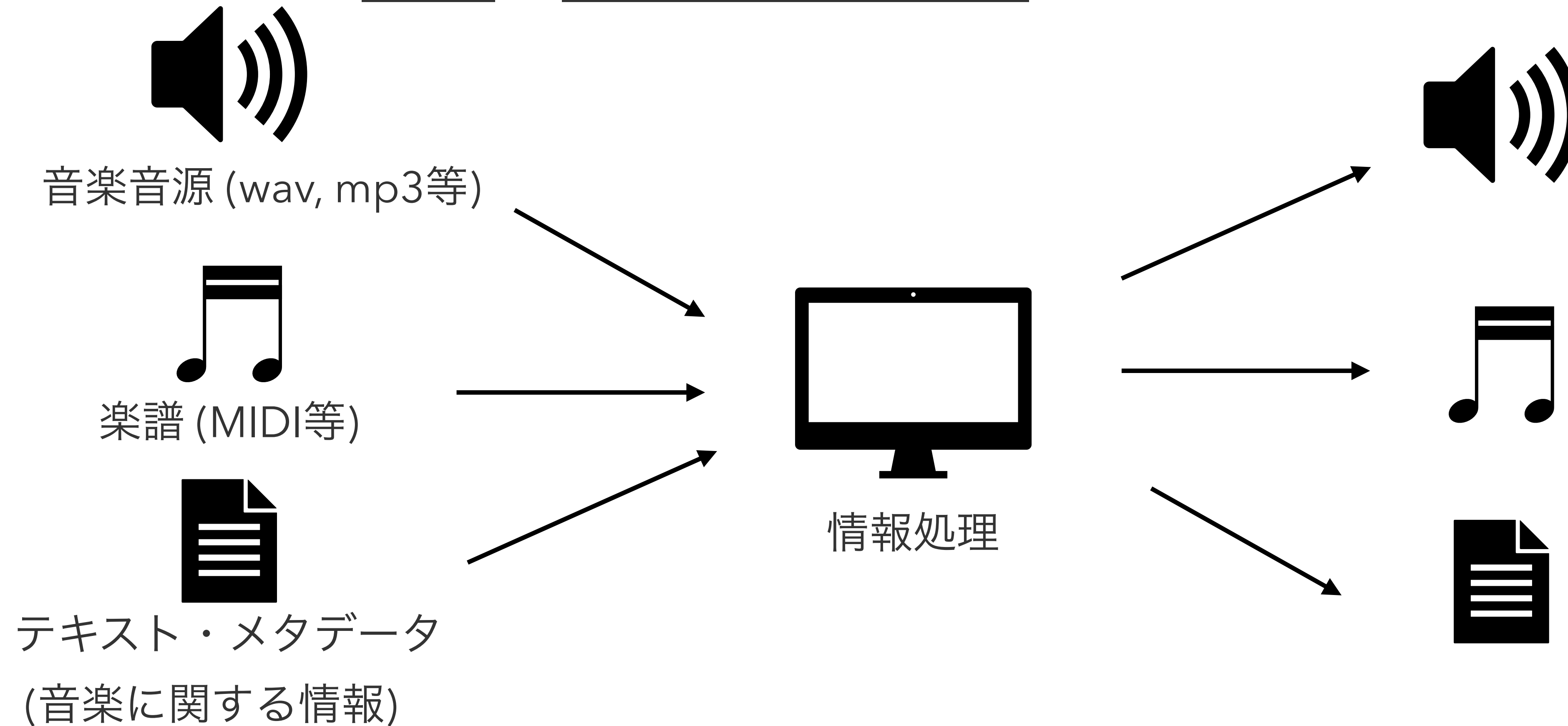
- （暫定）春ABC: 毎週水曜56限
  - おそらくどの講義ともかぶってないはず
  - ゼミの日程と被ったら変えます

- **Musical Applications of Machine Learning (2021)**
  - <https://mac.kaist.ac.kr/~juhan/gct634/index.html>
  - KAISTの大学院の講義資料, 音楽情報処理研究者のJuhan Nam氏によって作成
  - 音楽情報処理 × 機械学習・深層学習のトピックをカバー
  - ちょっと音響寄り, 楽譜系のトピックをカバーしている資料は探します
- 宿題でサンプルコードが付いている
  - <https://github.com/juhannam/gct634-ai613-2021>

- レクチャー
  - 内容に関して輪講形式で発表
  - スライドにまとめる or コードがあればgoogle colab等を駆使
  - 資料をそのまま解説でもおk (however 他人のプレゼンで解説するのは難しい...)
- 輪講方式
  - 特に勉強したいトピックは個人に割り当て, 誰も希望者がいないトピックは山本が解説
  - 担当はこの時間の終わりに決めましょう

導入①：なぜ深層学習を使うのか？

## 音楽をコンピュータで扱う研究分野

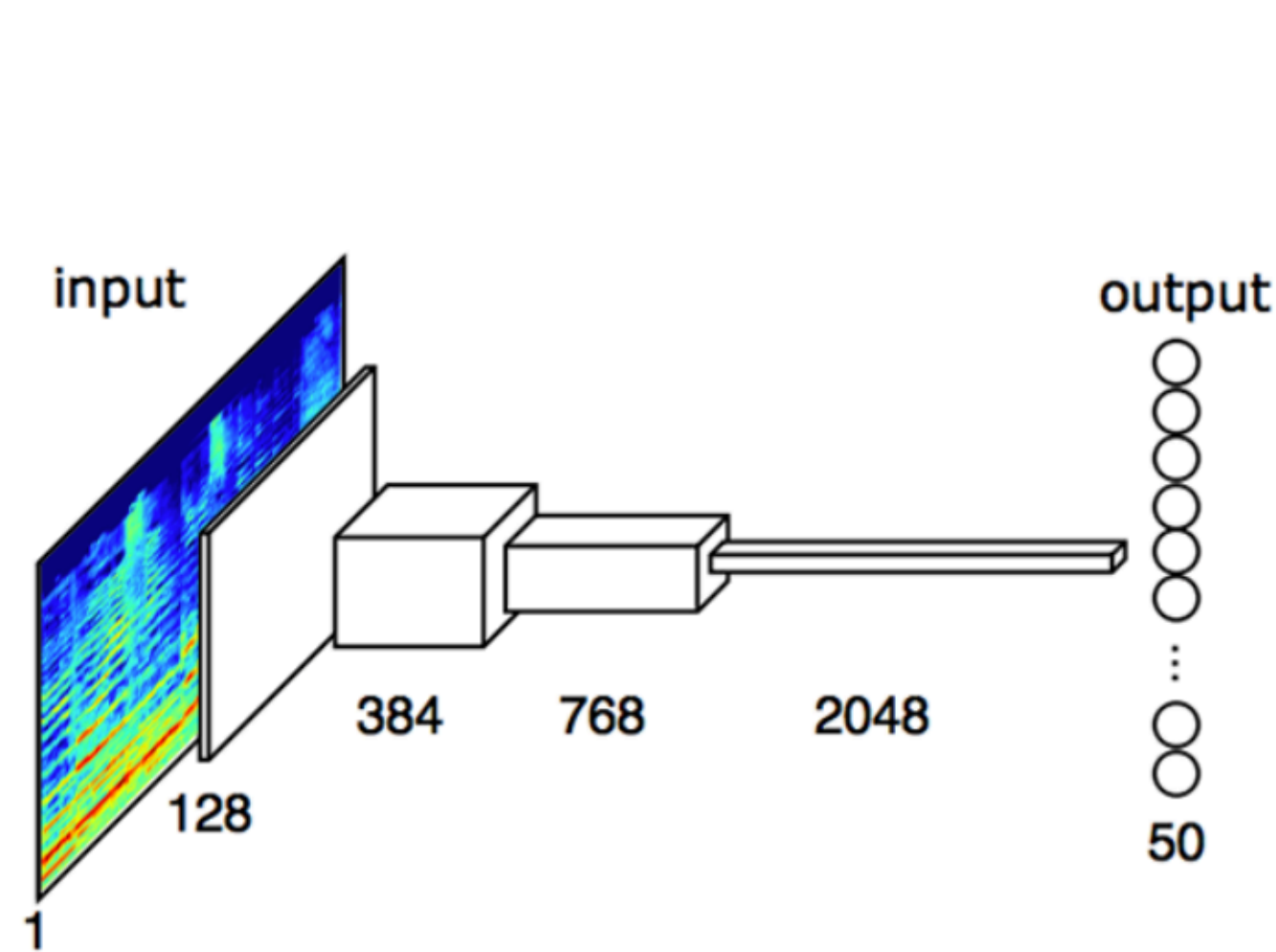


音楽を採譜して楽譜にする, 音符から演奏を生成する,  
作曲する, その音楽についての情報を獲得する etc...

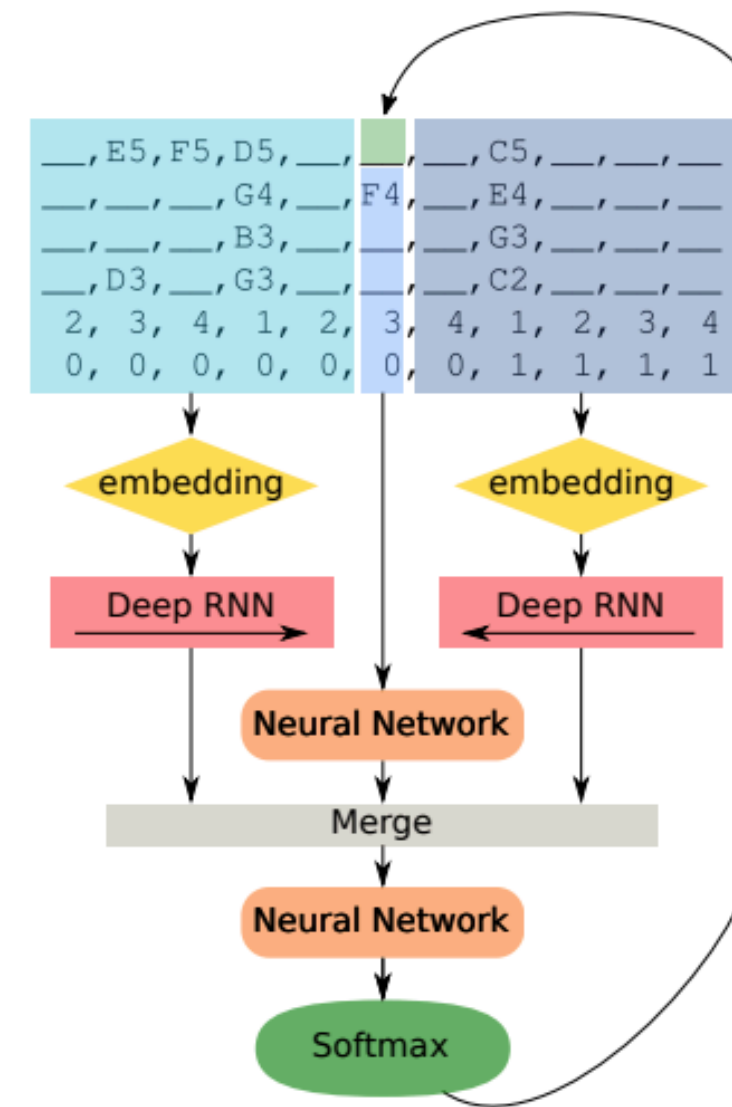
- 処理の方向
  - 作る：音や楽譜を生成 or 加工
  - 聴く：音や音楽から情報を獲得
- 対象データ
  - 音響データ：音そのもの. wav等
  - 記号表現：音楽を符号化したもの = 楽譜. MIDI等
  - テキストデータ：ジャンル・ムード等のメタデータ, 歌詞等
  - レコードデータ：音楽配信の聴いた履歴等
  - etc...



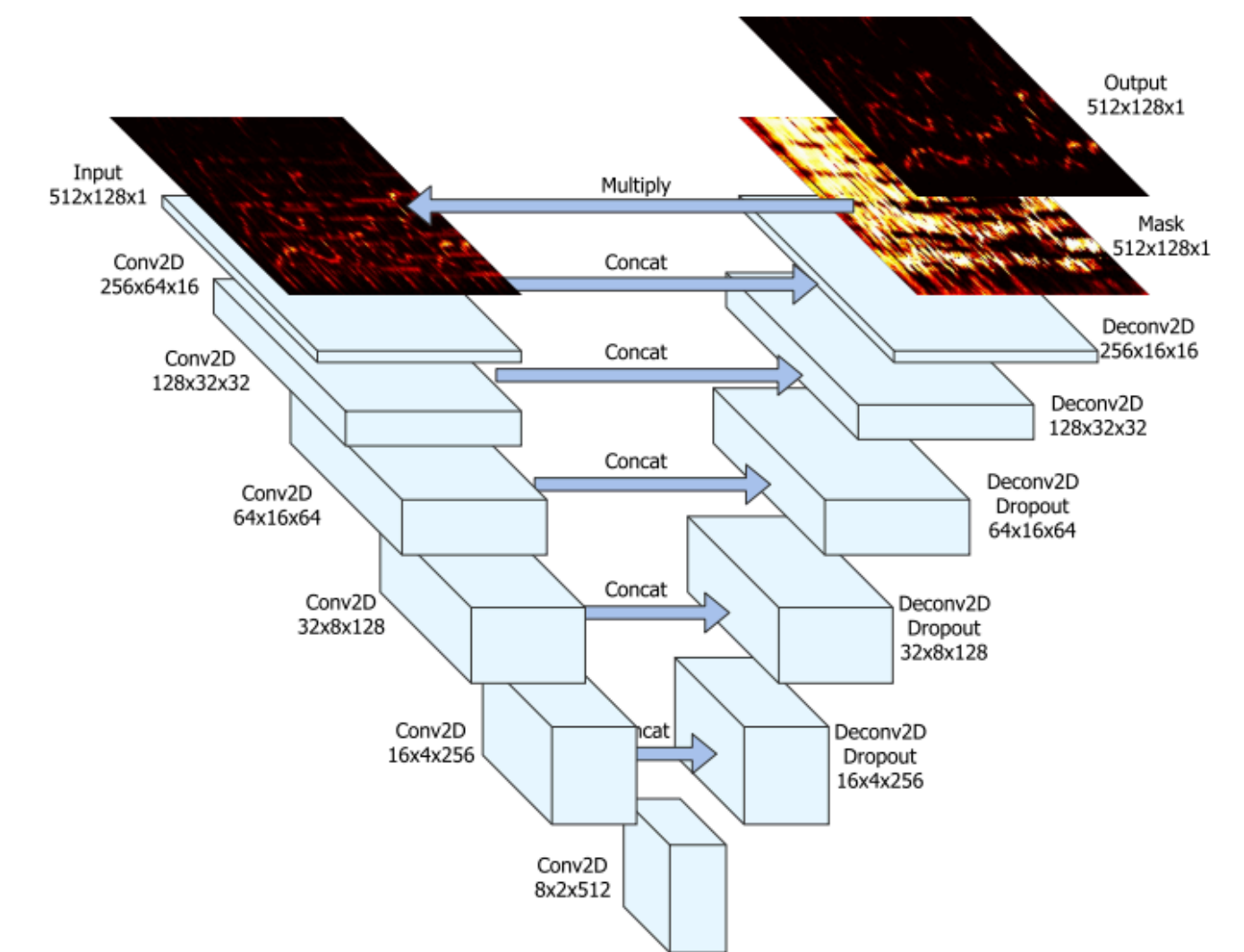
- 深層学習が音楽情報処理で急速に広まったのは2015~2018年くらい



Choi et al. AUTOMATIC TAGGING USING DEEP CONVOLUTIONAL NEURAL NETWORKS. ISMIR2016



Hadjeres et al. DeepBach: a Steerable Model for Bach Chorales Generation  
. ICML2017 <https://www.flow-machines.com/history/projects/deepbach-polyphonic-music-generation-bach-chorales/>



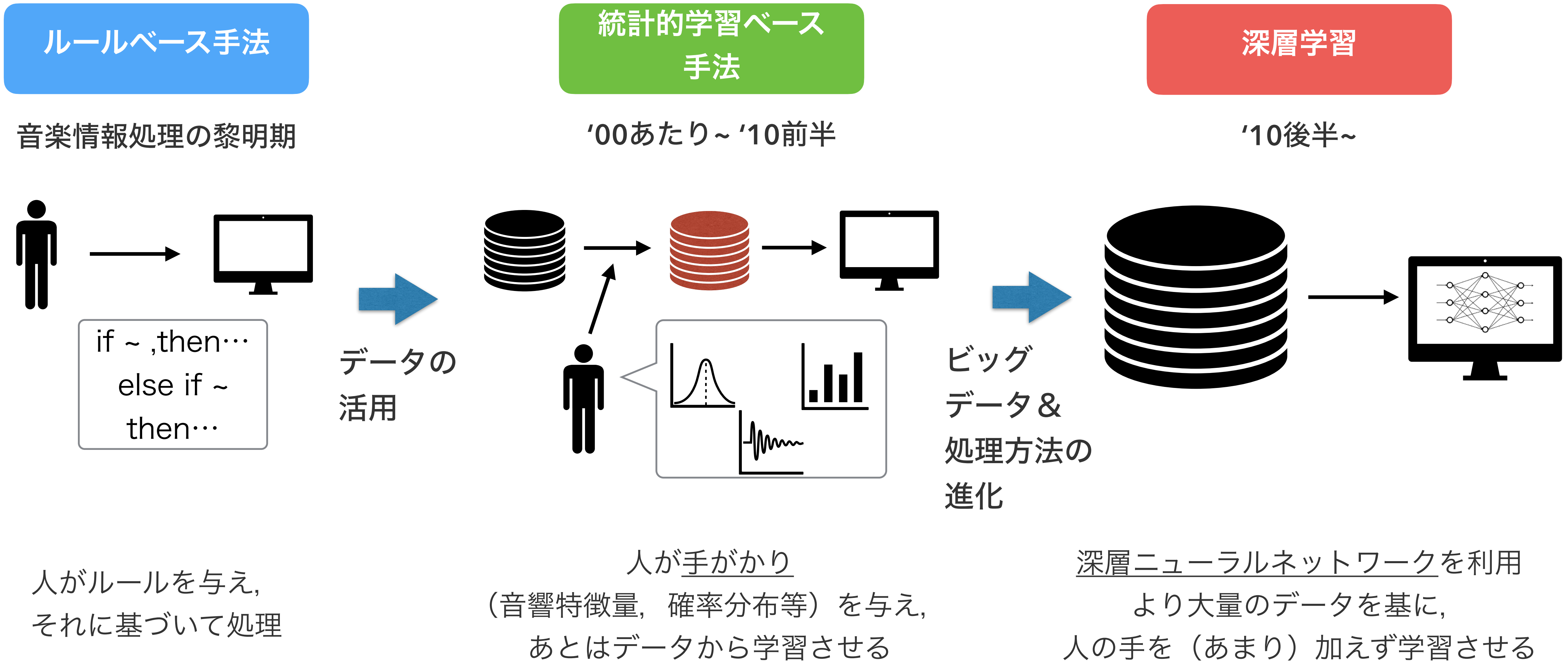
Jansson et al.  
Singing voice separation with deep u-net convolutional networks.  
ISMIR2017

- (この辺の年代をサーベイするとタスクをDeep化した始祖的な論文に出会えるかも)

## どの方面に関しても処理の精度が上がった

- 聴く側 -> 人間の能力に迫る認識能力
  - 自動採譜：ほぼ完璧に近い耳コピを実現
  - 音源分離：雑音の少ない分離音
  - 音楽ジャンル分類：人間と遜色ない程度の正解率 etc...
- 作る側 -> 違和感の少ない創作物
  - 歌声・楽音合成：本物と間違える程度の高精度な合成
  - 自動作曲：従来技術の不自然さをなくした楽曲の生成

## (大まかに) 3段階



観点 \ 手法	データ駆動型アプローチ		ルールベース
	深層学習	特徴量+古典的機械学習	
必要なラベルデータ量	大量	少量	なくともok
必要な計算機リソース	×	○	△（ルールの計算量に依る）
スケーラビリティ （他タスクへの転用などを柔軟に対応できるか）	◎	○	×
対象のモデリングの表現力	◎	○	×
ドメイン特有の知識の排除	○	△	×
結果の解釈可能性	△	○	◎

- こういう場合に選択肢に入る
  - 大量のデータが扱えるとき
  - とにかく性能が欲しいとき
  - 専門知識によるモデリングに限界を感じた時
  - 汎用的で柔軟なモデリングをしたいとき
- こういう場合は使わない方がいい
  - データが大量にない
  - 対象に対する性質が知りたい（∵ 解釈可能性に難がある）
  - 関係性が自明というところまで落とし込めるタスク

# 導入②：深層学習の流れ



1. データの用意
2. データの前処理
3. モデルの学習
4. モデルのテスト（評価）

- Meta社による深層学習ライブラリ
- 現在おそらく最もメジャーで参考になる資料も多い
- データの前処理や準備周りを,  
Dataloaderという独自の仕組みでやってくれる

