

レビュー論文：

- ・ **Toward a Science of Computational Ethology**
<https://www.sciencedirect.com/science/article/pii/S0896627314007934>
- ・ **Computational Analysis of Behavior**
<https://www.annualreviews.org/doi/abs/10.1146/annurev-neuro-070815-013845>
- ・ **Big behavioral data: psychology, ethology and the foundations of neuroscience**
<https://www.nature.com/articles/nn.3812>
- ・ **Decision-making behaviors: weighing ethology, complexity, and sensorimotor compatibility**
<https://www.sciencedirect.com/science/article/abs/pii/S0959438817301630>
- ・ **From classic ethology to modern neuroethology: overcoming the three biases in social behavior research**
<https://www.sciencedirect.com/science/article/pii/S0959438816300496>
- ・ **Ethology as a physical science**
<https://www.nature.com/articles/s41567-018-0093-0>
- ・ **Supervised and Unsupervised Learning Technology in the Study of Rodent Behavior**
<https://www.frontiersin.org/articles/10.3389/fnbeh.2017.00141/full>
- ・ **Computational Neuroethology: A Call to Action**
- ・ **Harnessing behavioral diversity to understand neural computations for cognition**

以上は大体言っていることは似たり寄ったりで

- (1) 計測技術の発達により行動に関する大規模データの取得が可能になってきている
 - (2) 解析技術も高度化していて、自動的に動物の動作や姿勢を推定する技術が増えてきている
 - (3) その結果、従来では考えられなかったほどの詳細かつ長期の行動データが得られるようになってきた
- 例えば、よりclosed loop (economy) な実験が実現可能になる
- (4) 一方で、それらを捉えるための概念的なフレームワークや理論はまだ十分とは言えない

Computational ethologyが求められる背景としては

- ・ **Neuroscience Needs Behavior: Correcting a Reductionist Bias**
<https://www.sciencedirect.com/science/article/pii/S0896627316310406>
- 神経科学の計測技術は凄まじい勢いで発展している
- 一方で、それと比較すると行動解析はそのペースに追いつけていない

- 結果、神経活動と行動の間で一对一の全単射関係を求めるのは (改めて) 困難であることがわかる
- よって、より詳細な行動解析が必要である
- 特に、「行動それ自体」で概念的な結論が導き出せるくらいの、fine-grainedな行動解析が神経科学には必要
- ただし、この論文はただ闇雲にデータを取ることを推奨しているわけではなく、大規模になった分、行動に関する概念的なフレームワークはますます重要になるということも強調している点に注意

(4) については

- ・ **Measuring behavior across scales**でも語られていて

<https://bmcbiol.biomedcentral.com/articles/10.1186/s12915-018-0494-7>

- どういう単位や次元にデータを圧縮するかは、結局事前に持っている仮説や理論に依存する
- 従って、どういふ分析単位を持つべきかは解析者側が自覚的である必要がある

Computational ethologyの具体的な研究例としては

- ・ **Statistical structure of locomotion and its modulation by odors**

<https://elifesciences.org/articles/41235>

- ショウジョウバエの動きを階層隠れマルコフでクラスタリングかけて、動作を分類
- 匂い刺激のある/なしでの行動生起率を比較
- 行動レパートリーは変わらず、その配分を変えてる

- ・ **Mapping Sub-Second Structure in Mouse Behavior**

[https://www.cell.com/neuron/fulltext/S0896-6273\(15\)01037-5](https://www.cell.com/neuron/fulltext/S0896-6273(15)01037-5)

- マウスのアリーナ上での行動を深度カメラで撮影
- ピクセルごとの深度をrandom projection -> 主成分分析で10個の特徴に削減
- 自己回帰隠れマルコフにより分類
- 動物の一連の動作 (1秒以内) をうまく分類するという手法を提案

- ・ **The Striatum Organizes 3D Behavior via Moment-to-Moment Action Selection**

<https://www.sciencedirect.com/science/article/pii/S0092867418305129>

- 上の論文を実際の研究に応用した例
- カルシウムイメージングで線条体のニューロンを大規模に記録しながら動作を分類
- ニューロン活動の時間的なパターンが3次元のごくごく短い運動のシーケンスを符号化していることを見出した

- ・ **Automated measurement of mouse social behaviors using depth sensing, video tracking, and machine learning**

<https://www.pnas.org/content/112/38/E5351>

- 深度カメラと普通のカメラを組み合わせた行動計測
- 教師あり学習で社会行動を自動分類している (SVM、決定木、ブースティング)
- BTBRマウスの社会行動頻度の低さを応用例として報告している。
- **Operant avoidance learning in crayfish, *Orconectes rusticus*: Computational ethology and the development of an automated learning paradigm**
 - これは比較的とっつきやすい例で
 - 回避学習をカニでやった実験
 - 長期間の位置移動をトラッキングかけて、回避の移動があることを示している
- **Discovery of Brainwide Neural-Behavioral Maps via Multiscale Unsupervised Structure Learning**
 - 教師なし学習でショウジョウバエの幼虫の行動のパターンを分類
 - 光刺激によって各パターン生成に固有のニューロン活動パターンがあることを同定した
- **Efficient cortical coding of 3D posture in freely behaving rats**
<https://science.sciencemag.org/content/362/6414/584>
 - ラットの姿勢を3Dで記録 (赤外線マーカー)
 - PPC-M2のニューロン活動記録
 - 姿勢と活動の相関性を検出した結果
 - よく起きる姿勢の方が少ないニューロンで符号化されている (経済的)
- **Machine Learning Reveals Modules of Economic Behavior from Foraging Mice**
<https://www.biorxiv.org/content/biorxiv/early/2018/06/29/357434.full.pdf>
 - トラッキングかけたXY座標の特徴を階層クラスタリングをかけてmoduleに分割する
 - 各モジュールが生起ようになる発達過程を調べた
 - 遺伝子改変マウス使って遺伝基盤を調べた
- **Mapping the stereotyped behaviour of freely moving fruit flies**
 - ショウジョウバエの動きをトラッキング
 - 体軸でアラインした画像に対し、PCA -> Wavelet変換 -> PCA
- **Structure of the Zebrafish Locomotor Repertoire Revealed with Unsupervised Behavioral Clustering**
- **A dictionary of behavioral motifs reveals clusters of genes affecting *Caenorhabditis elegans* locomotion**
 - *C.elegans*の体軸を49分割した点を使って行動分類した
 - 次元削減にはmRMRという方法を使っている
 - Peng H, Long F, Ding C (2005) Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal Mach Intell 27(8):1226–1238.
 - クラスタリングにはaffinity propagationなる方法を使っている

- TJOはk-meansでええやんって言ってる方法 <https://tjo.hatenablog.com/entry/2014/07/31/190218>

• The Behavioral Space of Zebrafish Locomotion and Its Neural Network Analog

- zebrafishの研究で、動きをlow dimensionに写す試み

Computational ethologyが盛り上がってる一因としては、行動解析に必要なツールキットが、オープンソースのソフトウェアとして配布され始めていることが挙げられる

• DeepLabCut: markerless pose estimation of user-defined body parts with deep learning

<https://www.nature.com/articles/s41593-018-0209-y>

- Deep Learningベースの複数部位を同時に追跡可能なトラッキングソフト
- Pythonから使用可能

• UMATracker: an intuitive image-based tracking platform

<http://jeb.biologists.org/content/221/16/jeb182469.abstract>

- GUIで使用可能なトラッキングソフト
- 無料の中では一番使いやすい

• GroupTracker: Video tracking system for multiple animals under severe occlusion

<https://www.sciencedirect.com/science/article/pii/S1476927115000237>

- 複数個体を同時追跡するためのトラッカー (UMATrackerにも入っている)

• DeepPoseKit: pose estimation using deep learning

- Konstanz謹製のトラッキングソフト
- deeplabcutと同じくらいの精度で速度が2倍
- ただし、GUIがクッソ使いにくい

• Deep learning tools for the measurement of animal behavior in neuroscience

- 近年の発展の状況を概観した論文

行動の自動分類の方法論

Computational ethologyの肝になるのは行動の分類である。座標やピクセルの情報はそのままでは解釈できないが、人間が手動でラベルづけをできる量ではないためだ。

• k-means

- 一番簡単
- お手軽にできる

• 隠れマルコフ

- 第一選択肢に考えられる
- 時系列データを離散的な「状態」から生成されていると仮定
- 状態は潜在変数なので、推定する
- pythonのhmmlearnが使いやすい
- ・ **自己回帰隠れマルコフ**
 - 通常の隠れマルコフモデルはある状態内ではデータが定常であることを仮定する
 - つまり、ある状態の中ではデータは切片のみ
 - 状態の中でデータが自己回帰過程に従って変動すると仮定するのが自己回帰隠れマルコフモデル
 - matijさんが公開しているpyshmmに実装されている
 - <https://github.com/mattij/pyhsmm>
- ・ **セミ隠れマルコフ**
 - 通常の隠れマルコフは各時間での状態を推定するが、
 - セミ隠れマルコフなら状態の「持続時間」もパラメータとして推定できる
 - ある程度の時間分解能でとったデータであれば、現実的な時系列データは、瞬間瞬間でコロコロ状態が変わるわけがないので、セミマルコフは有用
 - これもmatijさんが公開してくれている
- ・ **locally linear segmentation**
 - 局所的 (ごく短い時間) には時系列は一次の自己回帰に従うと仮定する
 - ただし、所属クラスが変わると自己回帰のパラメータも変わる
 - そこで、変化点検出を使って、時系列がそれまでの自己回帰からは残差が大きくなったところで「所属クラスが変わった」とみなす
 - 所属クラスが変わったら別のパラメータを持つ自己回帰が始まる
 - 分割した時系列に階層クラスタリングをかけて、時系列をいい感じのクラスに分節化する
 - 2019年のPNASで提案された方法だが、AntonioさんがGithubに公開している <https://github.com/AntonioCCosta/local-linear-segmentation>
- ・ **混合ガウスモデル**
 - データを混合ガウスモデルと考える方法
 - 「どのガウス分布に従うか」がクラスタリングとなる
 - Rのmclustが使いやすい
- ・ **階層クラスタリング**
 - データを「距離」に基づいて分類する方法
 - これもお手軽にできる
 - 階層的に推定されるので、どの階層までで「クラス」として分類するのが肝
 - 一応方法があるらしいけどよくわかってない

次元削減

データが多次元になると、モデルの推定そのものが困難であったり、時間的に現実的な範囲で終わなくなったりする。そこで、データから冗長な特徴を除くことが行われる。

「うまくいったやつを使ってる」人が多分多いけど、最近、Plos Computational Biology にTIPSが出た

Ten quick tips for effective dimensionality reduction

<https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1006907>

- ・ **主成分分析**
 - 第一選択肢はこれ
- ・ **独立成分分析**
 - セカンドチョイスはこれ
- ・ **非負値行列因子分解**
 - 0以上のデータに使うもの
 - 時系列データとかをいい感じに処理してくれるイメージがある
- ・ **t-sne**
 - 比較的新しいやつ。ただし重い
- ・ **ランダムプロジェクション**
 - 乱数で生成した行列をデータ行列にかけて次元を落とす方法
 - なんでこれでうまくいくのよくわからん謎の方法
- ・ **カーネル主成分分析**
 - 非線形の特徴量を抽出してくれる主成分分析
- ・ **GPLVM (Gaussian Process Latent Variable Model)**
 - 主成分分析の最終形態的な方法
- ・ **UMAP**
 - 2018年に出た新しめの方法。t-SNEと類似した結果を返すが、こっちの方が明らかに速い
 - pythonで使いやすい関数あり
 - <https://qiita.com/cheerfularge/items/27a55ebde4a671880666>

その他有用なリンク

- ・ ASAB-2019 ワークショプ資料
- <https://github.com/ietheredge/ASAB-2019-MLWS>
- ・ 色々な隠れマルコフモデルを実装したやばい人 (そのうち試す)
- <https://github.com/slinderman/ssm>