Article | Open access | Published: 23 October 2024

Machine-guided design of cell-type-targeting *cis*-regulatory elements

Sager J. Gosai, Rodrigo I. Castro, Natalia Fuentes, John C. Butts, Kousuke Mouri, Michael Alasoadura, Susan Kales, Thanh Thanh L. Nguyen, Ramil R. Noche, Arya S. Rao, Mary T. Joy, Pardis C. Sabeti, Steven K. Reilly & Ryan Tewhey

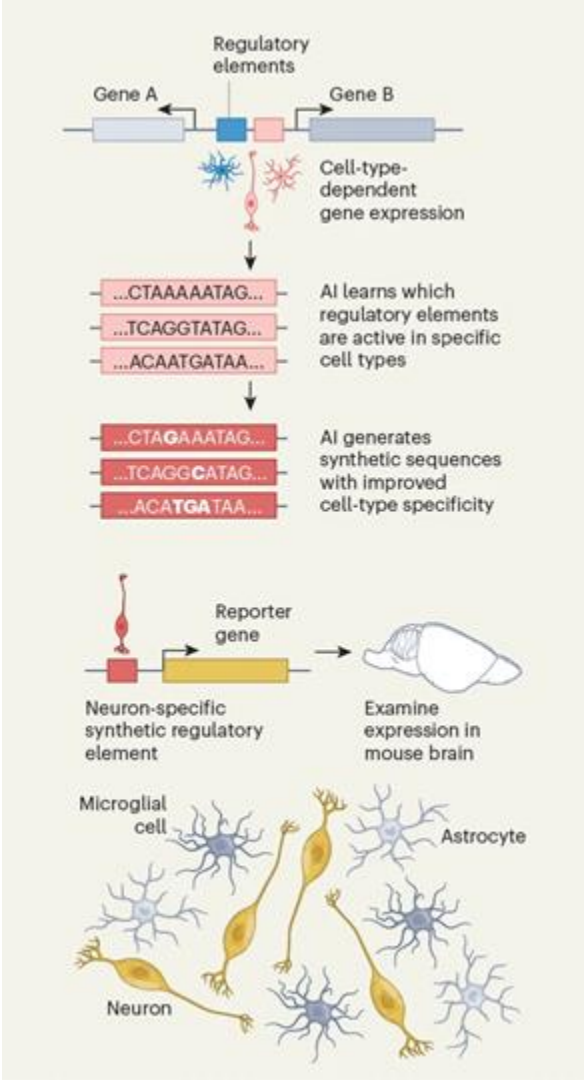
Nature **634**, 1211–1220 (2024) | Cite this article

Cristina Martin-Linares

November 19, 2024

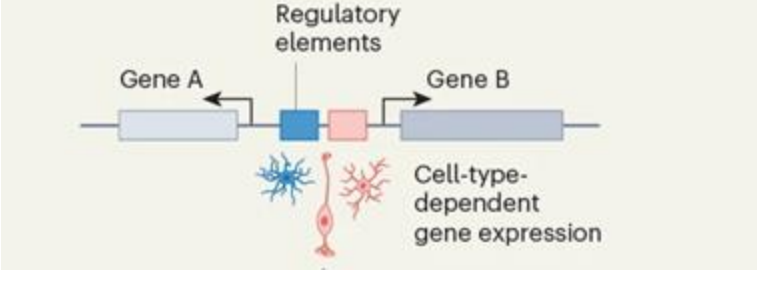
Why I chose this paper

- Practical AI-assisted design in biology
- Generation of synthetic sequences (CREs) that are relevant for many biological applications
- Method blends wet lab and dry lab, not purely generative AI

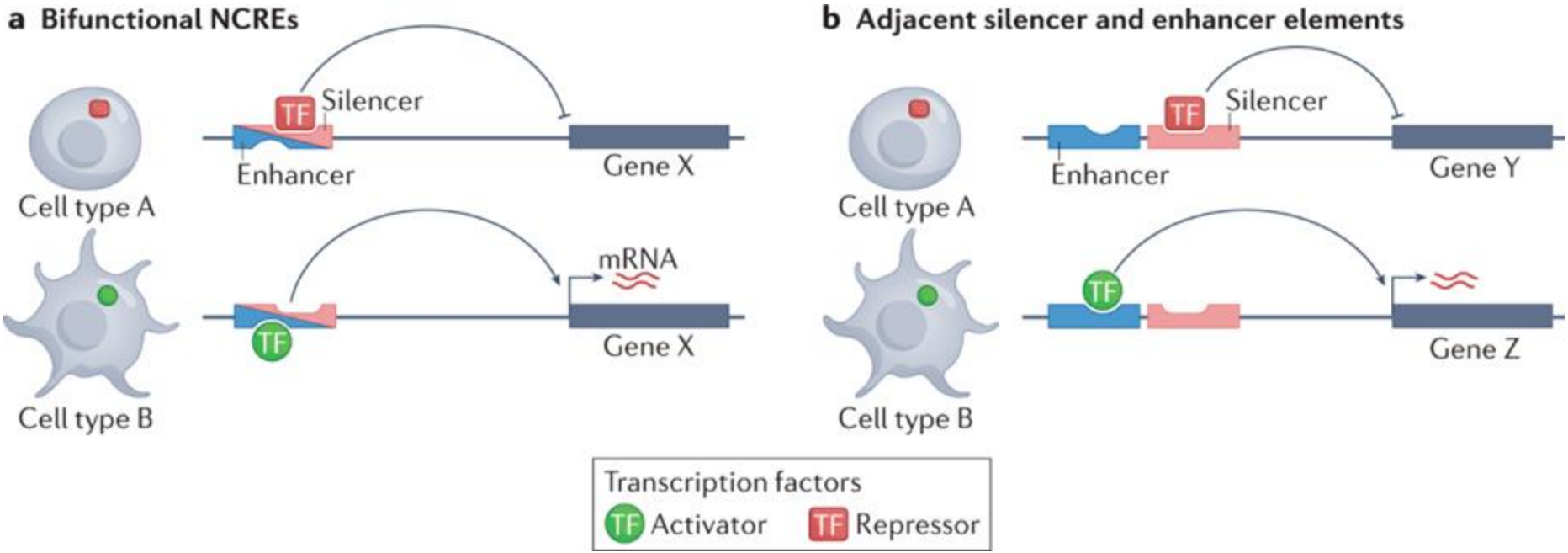


Cis-Regulatory Elements (CREs)

- What are CREs?
  - DNA sequences that regulate gene expression
- Importance
  - Control cell-type-specific gene expression
  - Important for tissue identity, development, function, etc
- Study Goal
  - Develop an AI-based method to **engineer synthetic CREs** with improved cell-type specificity



Cis-Regulatory Elements (CREs)

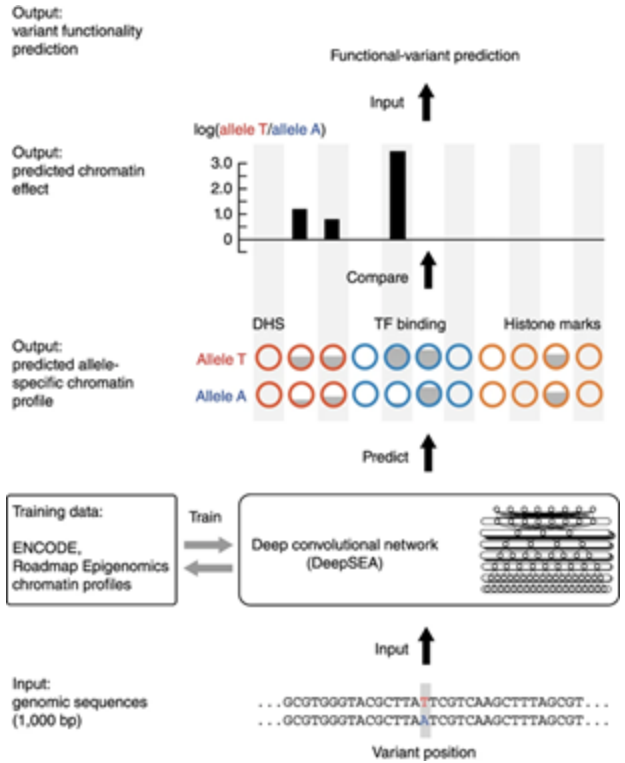


Problem Statement

- Biologists have created CREs by testing sequences taken from the genome
- Designing CREs is useful for therapeutic and research directions

- Strengths of using naturally occurring sequences
  - Iterating on genome-derived sequences can successfully create CREs
- Limitations of using naturally occurring sequences
  - Limited candidate sequences from genome
  - Difficult to design sequences that function better than naturally occuring
  - Cannot explore full space of potential sequences (10^{120} sequences if 200bp long)

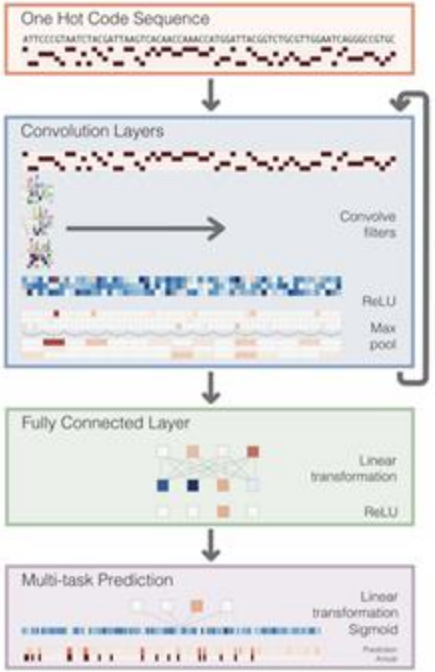
Deep learning to model chromatin dynamics



Predicting effects of noncoding variants with deep learning–based sequence model

Jian Zhou[1,2] & Olga G Troyanskaya[1,3,4]

DeepSEA



Method

Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks

David R. Kelley,[1] Jasper Snoek,[2] and John L. Rinn[1]

Recent Related Work

Article

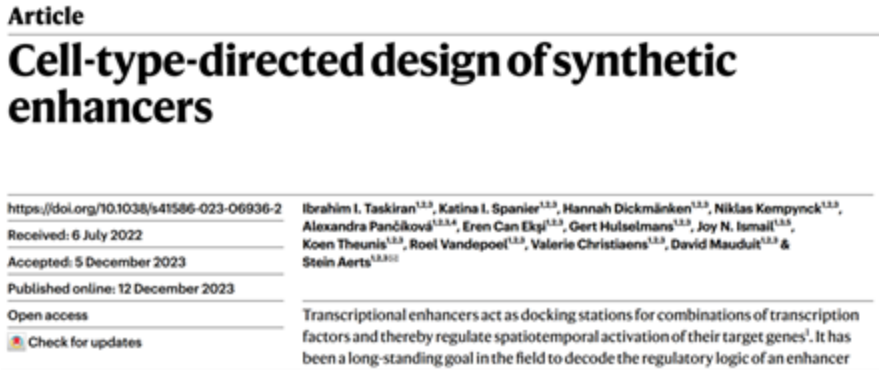
Targeted design of synthetic enhancers for selected tissues in the *Drosophila* embryo

https://doi.org/10.1038/s41586-023-06905-9
Received: 19 June 2023
Accepted: 28 November 2023
Published online: 12 December 2023
Open access
Check for updates

Bernardo P. de Almeida[1,2,5], Christoph Schaub[3], Michaela Pagani[3], Stefano Secchia[3], Eileen E. M. Furlong[3] & Alexander Stark[1,4]

Enhancers control gene expression and have crucial roles in development and homeostasis[1–3]. However, the targeted de novo design of enhancers with tissue-specific activities has remained challenging. Here we combine deep learning and transfer learning to design tissue-specific enhancers for five tissues in the *Drosophila*

Transfer learning using DNA accessibility (chromatin)



Article

Cell-type-directed design of synthetic enhancers

https://doi.org/10.1038/s41586-023-06936-2
Received: 6 July 2022
Accepted: 5 December 2023
Published online: 12 December 2023
Open access
Check for updates

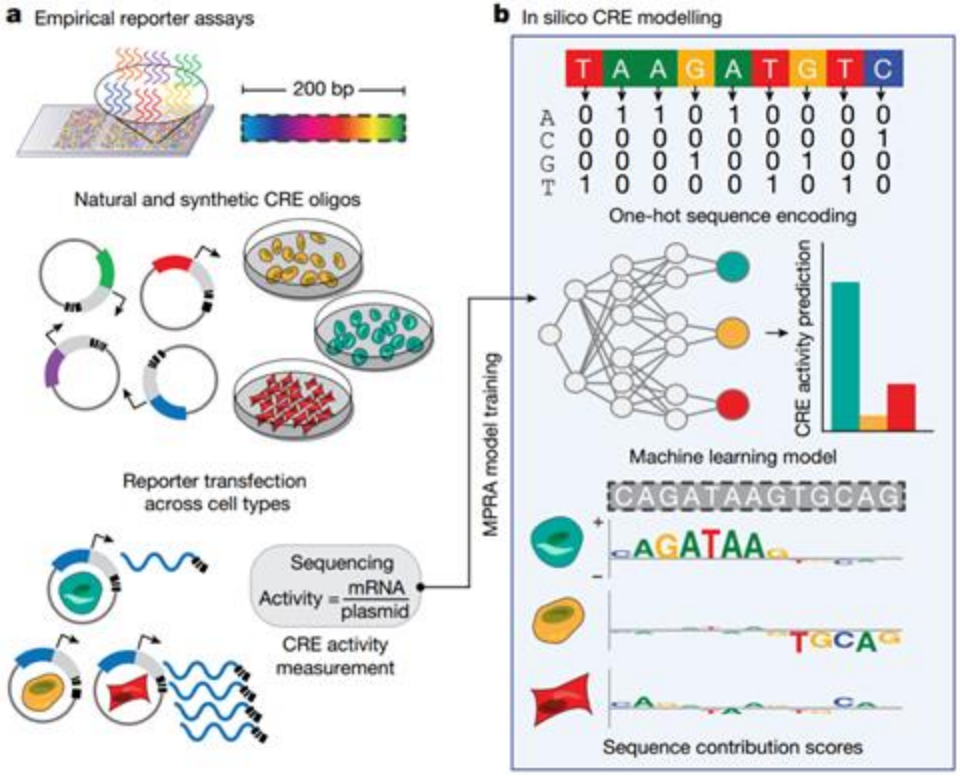
Ibrahim I. Taskiran[1,2,3], Katina I. Spanier[1,2,3], Hannah Dickmänken[1,2,3], Niklas Kempynck[1,2,3], Alexandra Pančíková[1,2,3,4], Eren Can Ekşi[1,2,3], Gert Hulselmans[1,2,3], Joy N. Ismail[1,2,3], Koen Theunis[1,2,3], Roel Vandepoel[1,2,3], Valerie Christiaens[1,2,3], David Mauduit[1,2,3] & Stein Aerts[1,2,3]

Transcriptional enhancers act as docking stations for combinations of transcription factors and thereby regulate spatiotemporal activation of their target genes[1]. It has been a long-standing goal in the field to decode the regulatory logic of an enhancer

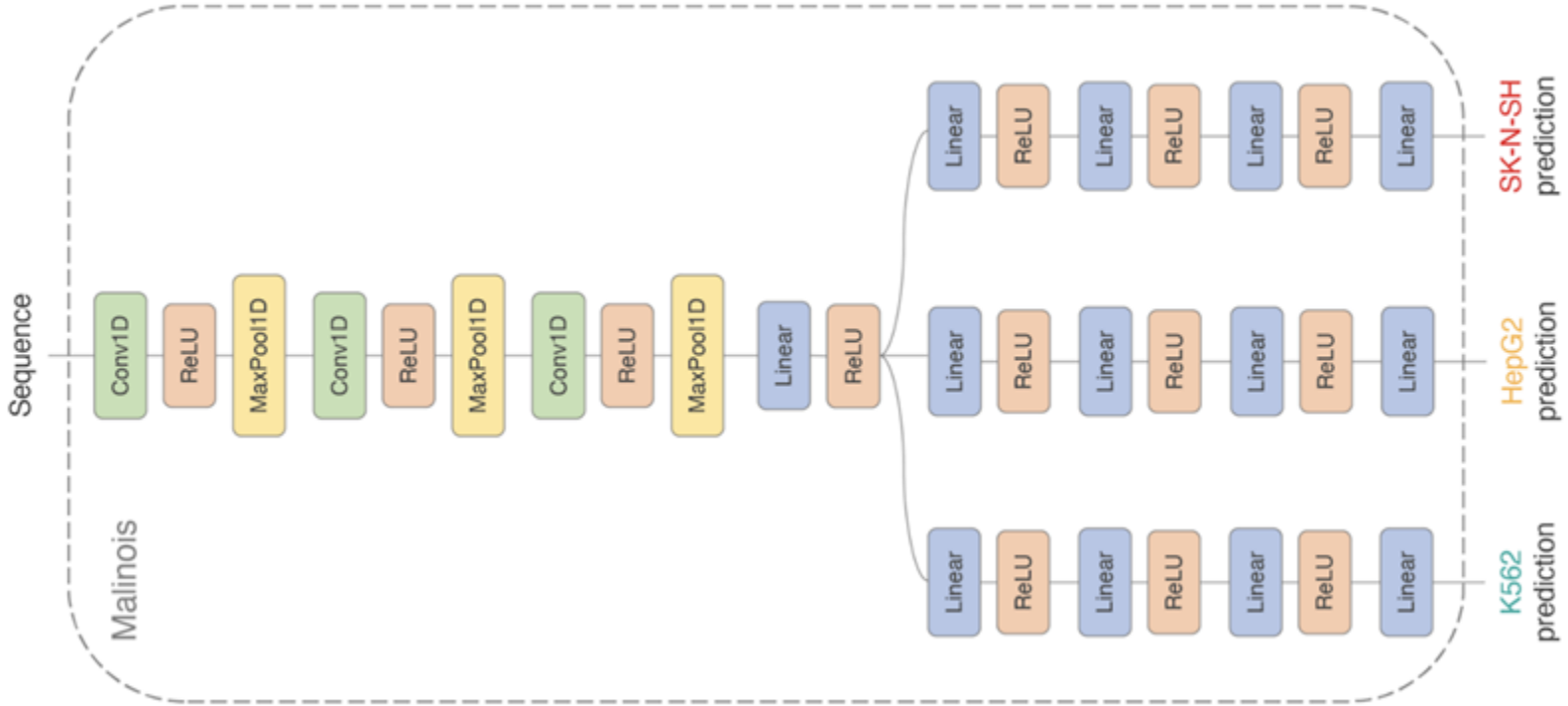
Iterations of **saturation mutagenesis** to explore the space.
Not automated or scalable due to experimental limitations

Overview of Approach

- Design synthetic CREs (colors) that are barcoded (black lines)
- Massively Parallel Reporter Assay (**MPRA**)
- Test in three cells
  - K562 (immune cell)
  - HepG2 (liver cell)
  - SK-N-SH (neuron cell)
- Sequence RNA to measure **activity**
- Use activity measurements to train **Malinois**
- Identify sequence contribution scores



Malinois


Sequence
Malinois
Conv1D
ReLU
MaxPool1D
Conv1D
ReLU
MaxPool1D
Conv1D
ReLU
MaxPool1D
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
ReLU
Linear
SK-N-SH prediction
HepG2 prediction
K562 prediction

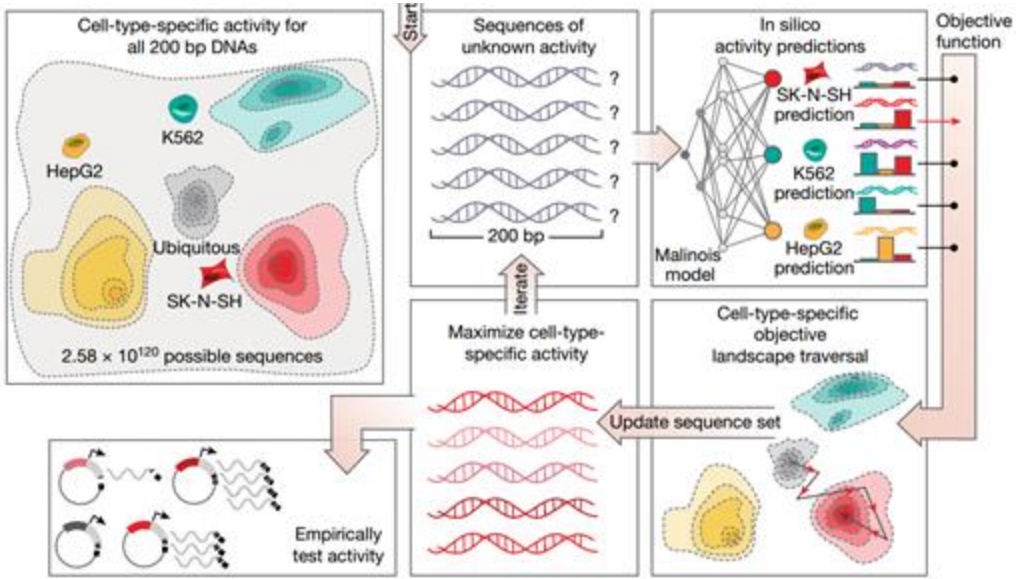
Training and Hyperparameter Tuning

- Training with backpropagation; using log2 transformed fold change (FC) as target values.
- Training/Validation/Test Split: Unique chromosomes for each set.
- Hyperparameter tuning with Bayesian optimization using a validation set.
- Performance metrics: Pearson and Spearman correlation with empirical log2(FC) values.

$$log2(FC) = log2\frac{RNA}{DNA}$$

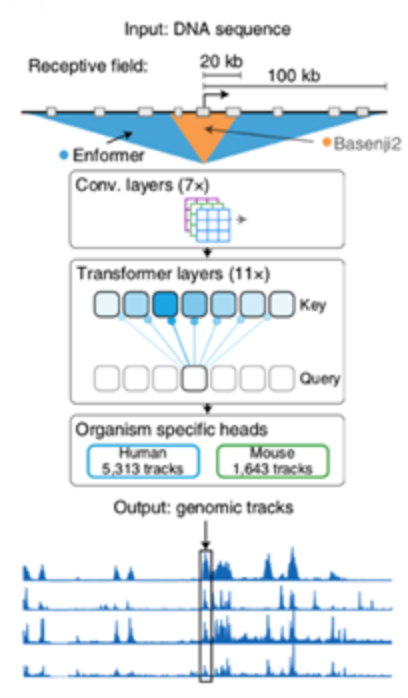
Optimization framework

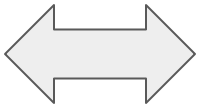
- Create and test massively parallel reporter assay **(MPRA)**
- Train **deep learning model** to predict CRE activity (Malinois)
- **Optimization framework** for designing synthetic, cell-type-targeting CREs
- Goal is to maximize cell-type specificity by **optimizing the difference between on-target and off-target** expression (MinGap)

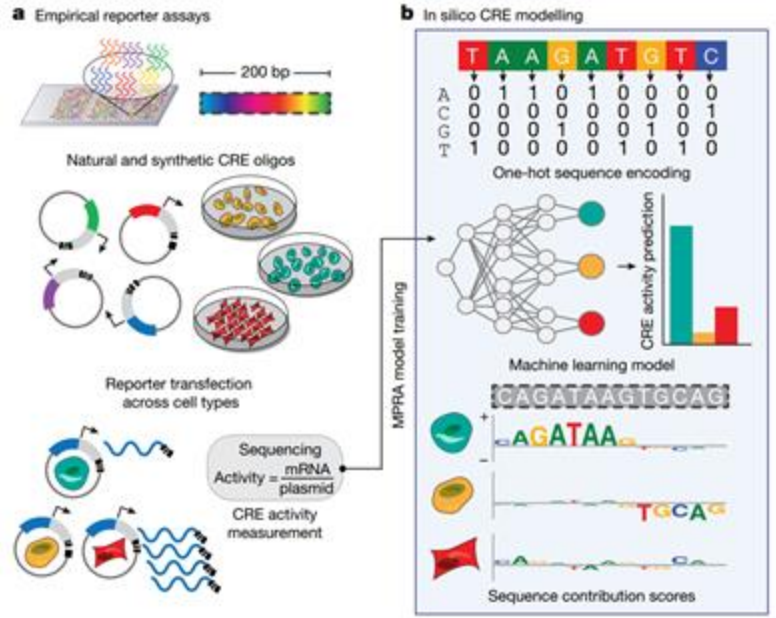


Cell type Optimizing Design Algorithm (**CODA**)

Predicting Genetic Code vs. Creating Genetic Code







Enformer

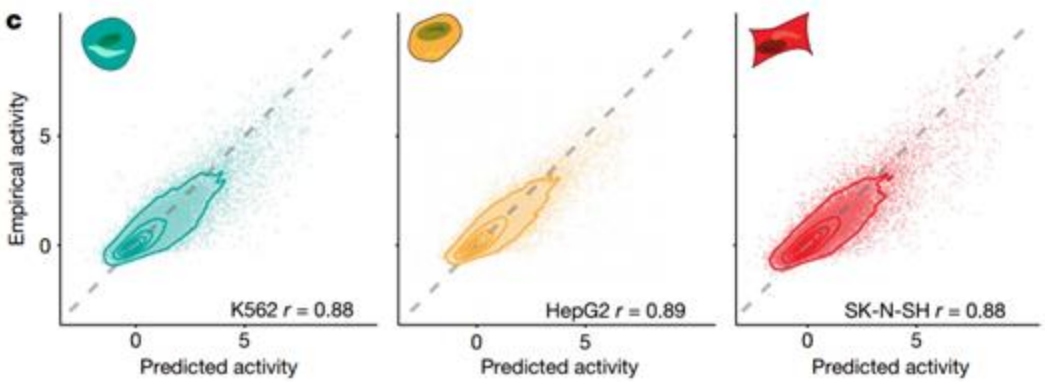
(observation of different modalities)

CODA

(guided iterations of synthetic data)

Figure 1

- Malinois **predicted activity is well correlated with empirical activity** from the MPRAs
- Malinois predictions match other “ground truth” biological signals associated with promoter activity in chr13



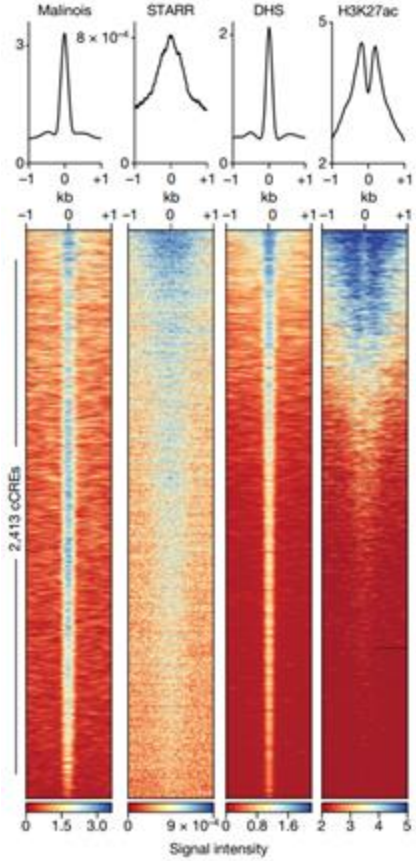


Figure 1

- Malinois predictions match empirical predictions from sequences taken from the human genome, surrounding the GATA1 gene.
- Pink-only indicates few Malinois false positives

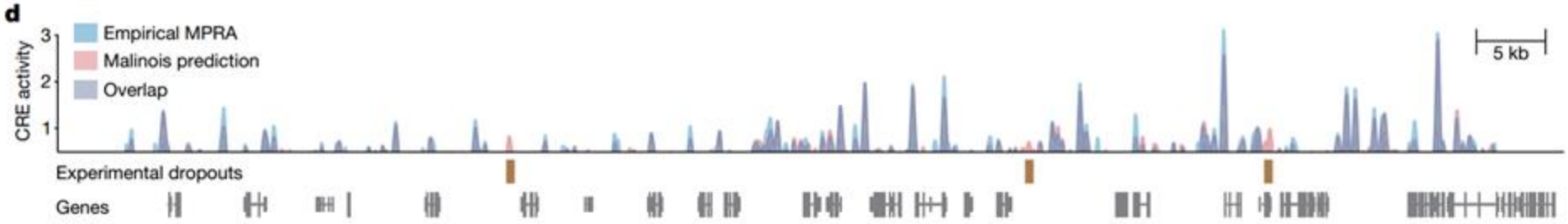
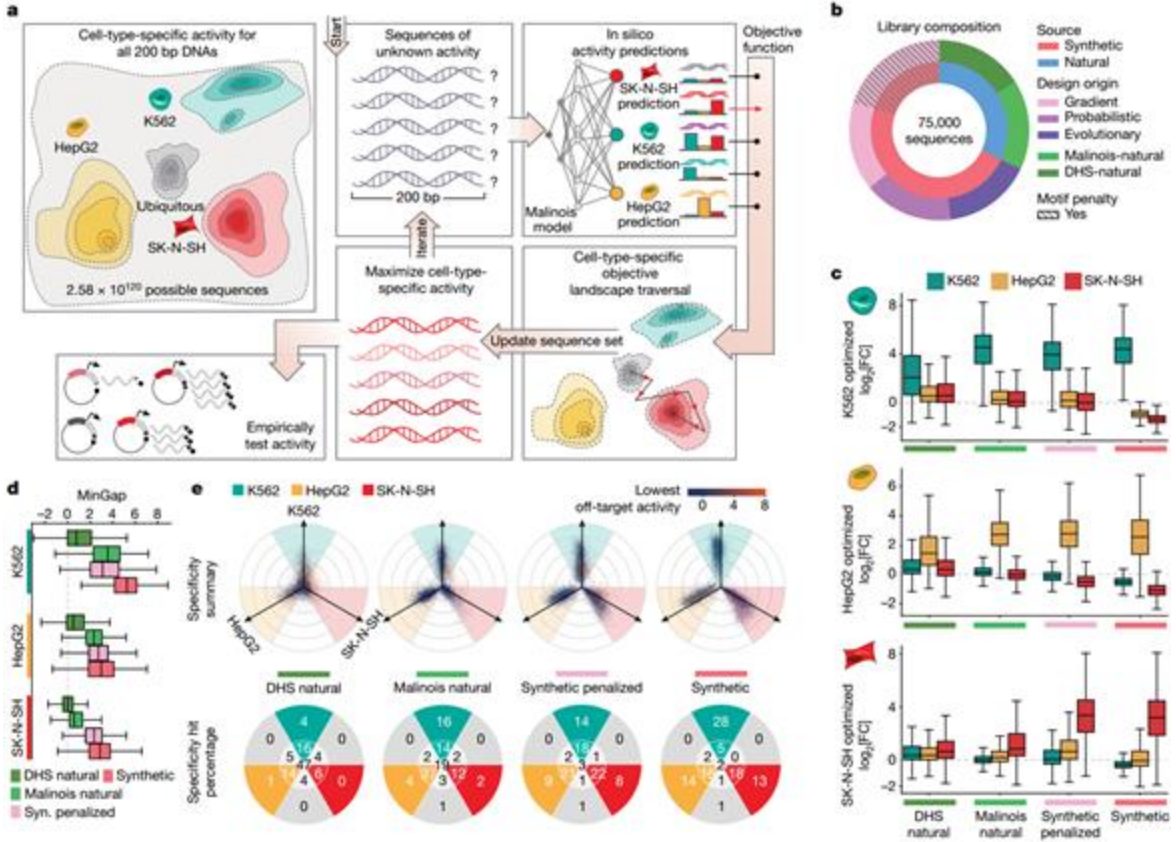
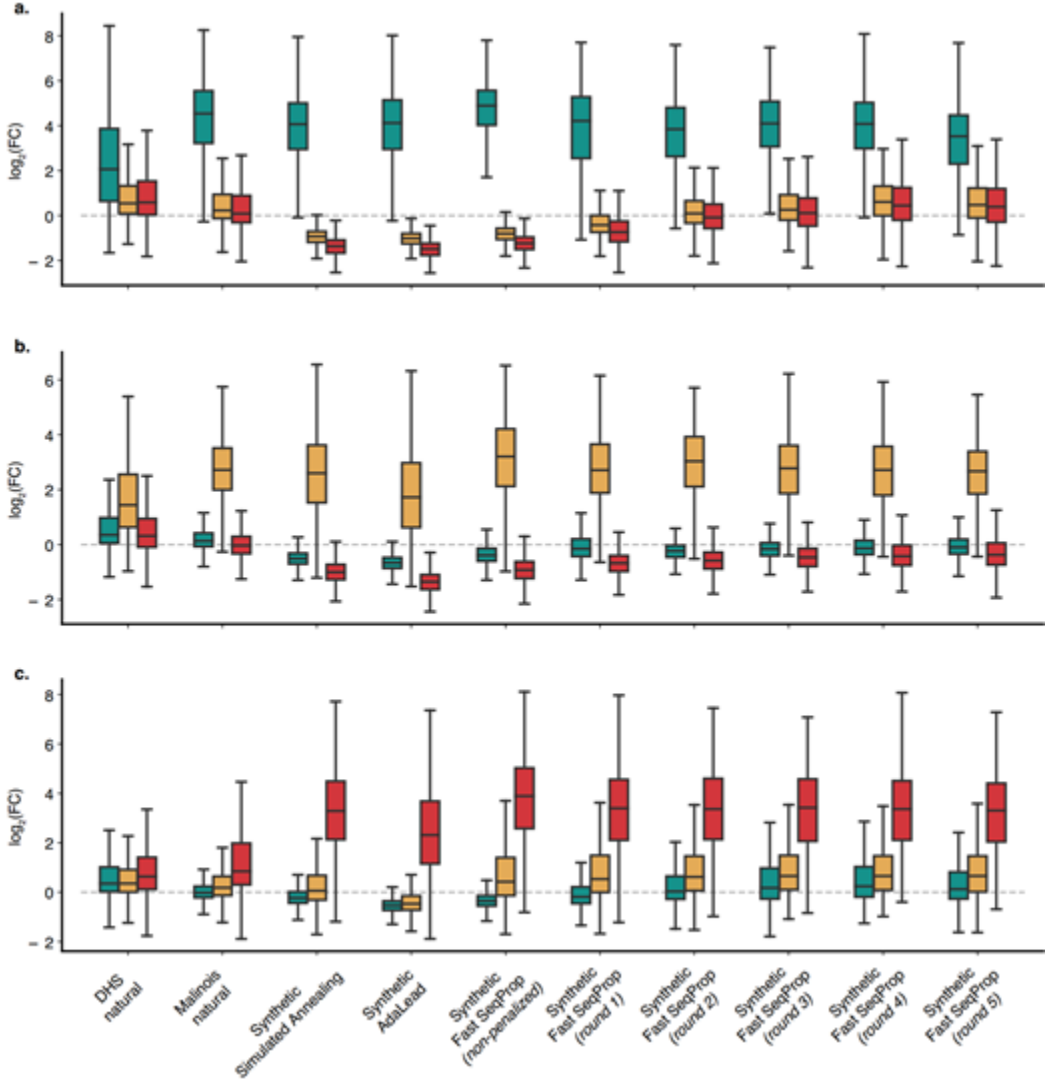


Figure 2


a
Cell-type-specific activity for all 200 bp DNAs
K562
HepG2
Ubiquitous
SK-N-SH
2.58 × 10^120 possible sequences
Start
Sequences of unknown activity
200 bp
In silico activity predictions
SK-N-SH prediction
K562 prediction
HepG2 prediction
Malinois model
Objective function
Iterate
Maximize cell-type-specific activity
Cell-type-specific objective landscape traversal
Update sequence set
Empirically test activity
b
Library composition
75,000 sequences
Source
Synthetic
Natural
Design origin
Gradient
Probabilistic
Evolutionary
Malinois-natural
DHS-natural
Motif penalty
Yes
c
K562
HepG2
SK-N-SH
K562 optimized log2[FC]
HepG2 optimized log2[FC]
SK-N-SH optimized log2[FC]
DHS natural
Malinois natural
Synthetic penalized
Synthetic
d
MinGap
DHS natural
Synthetic
Malinois natural
Syn. penalized
e
Lowest off-target activity
Specificity summary
Specificity hit percentage
DHS natural
Malinois natural
Synthetic penalized
Synthetic


a.
b.
c.
log2(FC)
8
6
4
2
0
−2
DHS natural
Malinois natural
Synthetic Simulated Annealing
Synthetic AdaLead
Synthetic Fast SeqProp (non-penalized)
Synthetic Fast SeqProp (round 1)
Synthetic Fast SeqProp (round 2)
Synthetic Fast SeqProp (round 3)
Synthetic Fast SeqProp (round 4)
Synthetic Fast SeqProp (round 5)

Optimization Objective

- The objective function is MinGap (simulated annealing) or Bent-MinGap (Fast SeqProp and AdaLead)
- MinGap is defined as the difference between activity in target and highest off-target cell type.
- Bent-MinGap uses **Bent transformation** to smooth out extreme values in non-target cells

$$\text{MinGap} = y_+ - y_- \qquad y = log2(FC)$$

$$\text{Bent-minGap} = g(y_+) - g(y_-) \qquad g(y) = y - e^{-y} + 1$$

Optimization Strategies

- FastSeqProp (Gradient-Based)
  - Fine-tuning
- AdaLead (Evolutionary Search)
  - Balanced between exploration and exploitation
- Simulated Annealing (Probabilistic)
  - More exploration
- Motif penalty to reduce overuse of strong motifs

Optimization Strategies

Simulated Annealing (Probabilistic)

- **Initialization**: A single MPRA sequence is selected as the starting point
- **Perturbation**: A random mutation is made to the sequence
- **Evaluation**: The new sequence is scored for MinGap
- **Rule**:
  - If the new sequence improves MinGap, accept it
  - Otherwise, its acceptance depends on probability $P=e^{-\Delta/T}$, where Δ is the decrease in MinGap and T is a temperature parameter
  - High temperature = all mutated sequences accepted
  - Low temperature = list of sequences is frozen
- **Cooling**: T is decreased over multiple iterations to reduce the probability of accepting seq
- **Repeat**: Repeat until satisfied with sequences

Optimization Strategies

FastSeqProp (Gradient-Based)

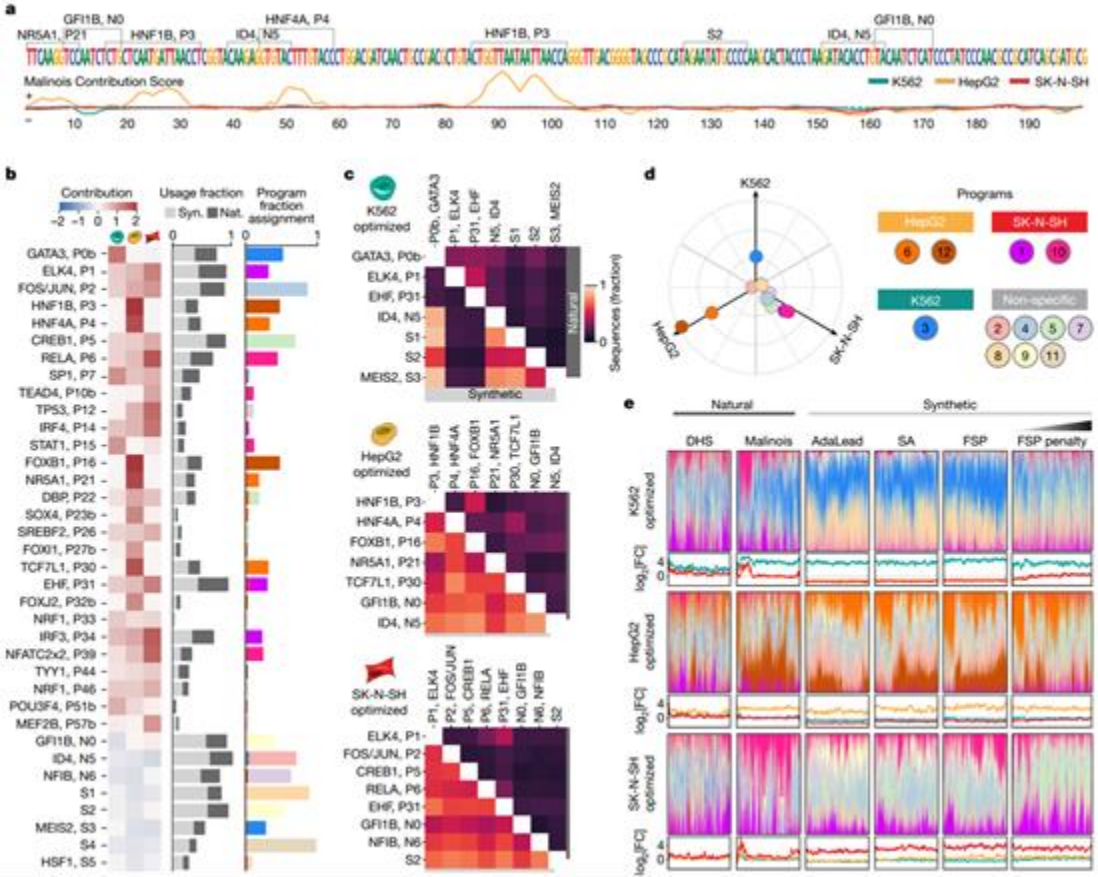
- **Gradient Calculation**: compute the gradient of the MinGap loss function with respect to the input sequence using backprop through Malinois
- **Update Sequence**: use gradients to iteratively adjust basepairs in directions that improve MinGap to finetune the sequences
- **Stop**: Iterations stop when MinGap improvements reach a pre-defined limit

Optimization Strategies

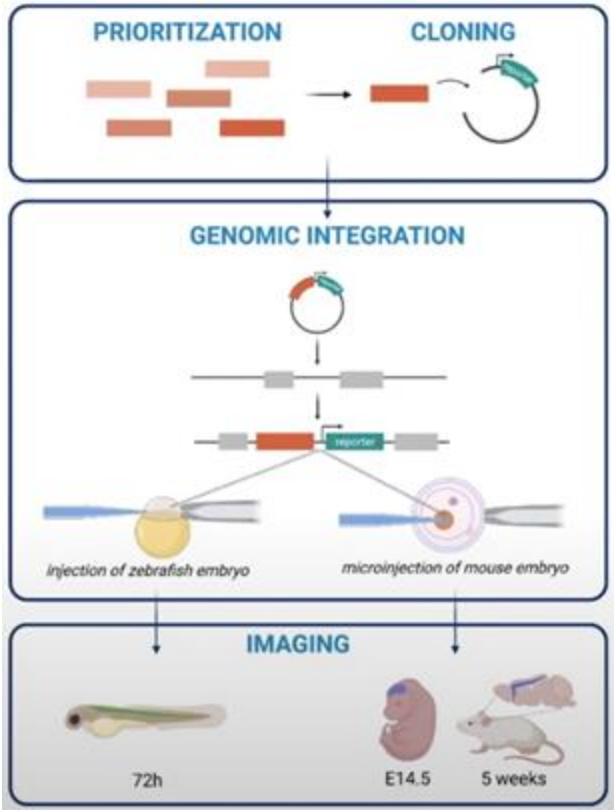
AdaLead (Evolutionary Search)

- **Initialize**: Set of sequences is randomly selected from promising MPRA candidates
- **Evaluate**: Each sequence is scored using MinGap
- **Selection**: Best sequences are chosen to form a "parent" pool
- **Mutation and Crossover**:
  - Mutations introduce random changes in individual sequences
  - Crossover combines parts of two sequences to create new sequences
- **Iteration**: The process repeats with each generation, gradually improving MinGap
- **Stop**: Iterations stop when MinGap improvements reach a pre-defined limit

Figure 3

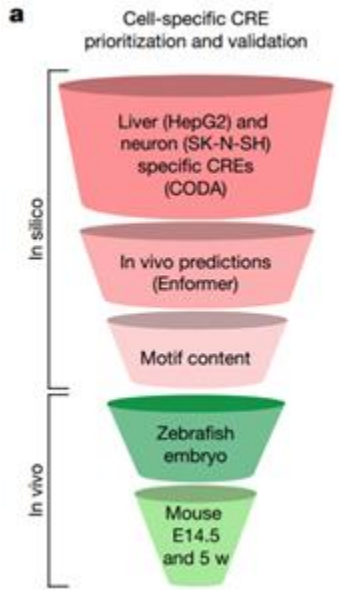

a
GFI1B, N0
NR5A1, P21
HNF1B, P3
HNF4A, P4
ID4, N5
HNF1B, P3
S2
ID4, N5
GFI1B, N0
Malinois Contribution Score
K562
HepG2
SK-N-SH
b
Contribution
Usage fraction
Syn. Nat.
Program fraction assignment
GATA3, P0b
ELK4, P1
FOS/JUN, P2
HNF1B, P3
HNF4A, P4
CREB1, P5
RELA, P6
SP1, P7
TEAD4, P10b
TPS3, P12
IRF4, P14
STAT1, P15
FOXB1, P16
NR5A1, P21
DBP, P22
SOX4, P23b
SREBF2, P26
FOXI1, P27b
TCF7L1, P30
EHF, P31
FOXJ2, P32b
NRF1, P33
IRF3, P34
NFATC2x2, P39
TYY1, P44
NRF1, P46
POU3F4, P51b
MEF2B, P57b
GFI1B, N0
ID4, N5
NFIB, N6
S1
S2
MEIS2, S3
S4
HSF1, S5
c
K562 optimized
HepG2 optimized
SK-N-SH optimized
Natural
Synthetic
Sequences (fraction)
d
K562
HepG2
SK-N-SH
Programs
e
Natural
Synthetic
DHS
Malinois
AdaLead
SA
FSP
FSP penalty
K562 optimized
HepG2 optimized
SK-N-SH optimized
log2[FC]

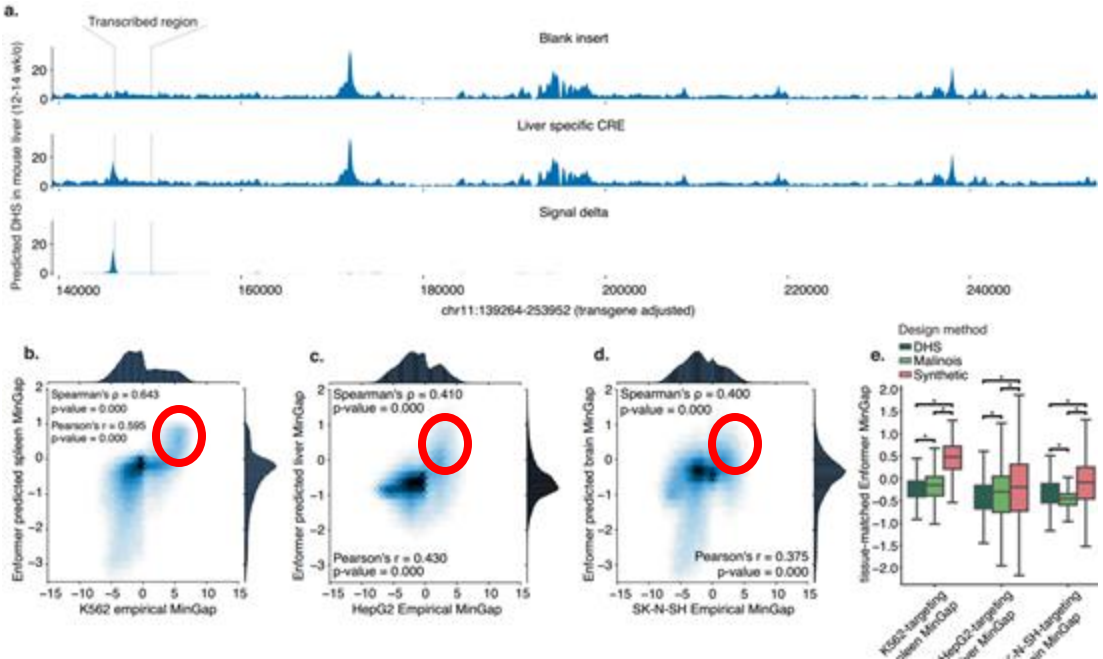
Framework for in vivo assessment of synthetic CREs



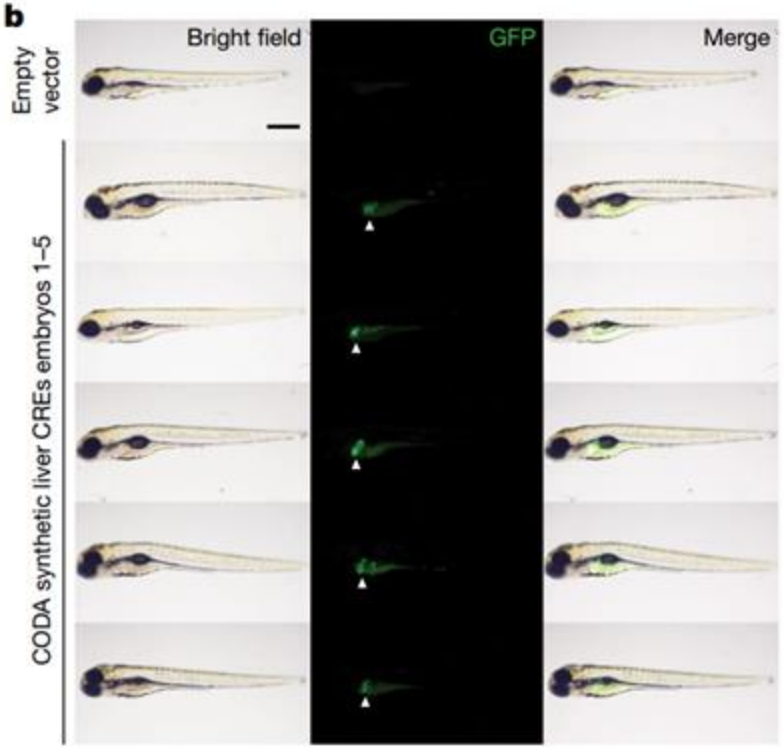
Enformer Helps Optimize for Genomic Integration (Mouse)

Optimized CODA-derived CREs using Enformer and tested in zebrafish





Enformer Helps Optimize for Genomic Integration (Mouse)



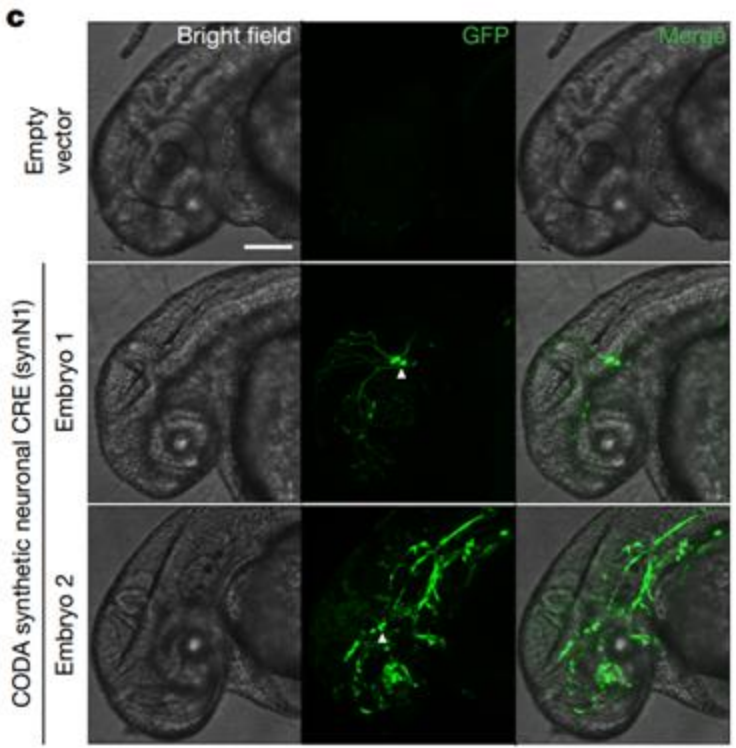
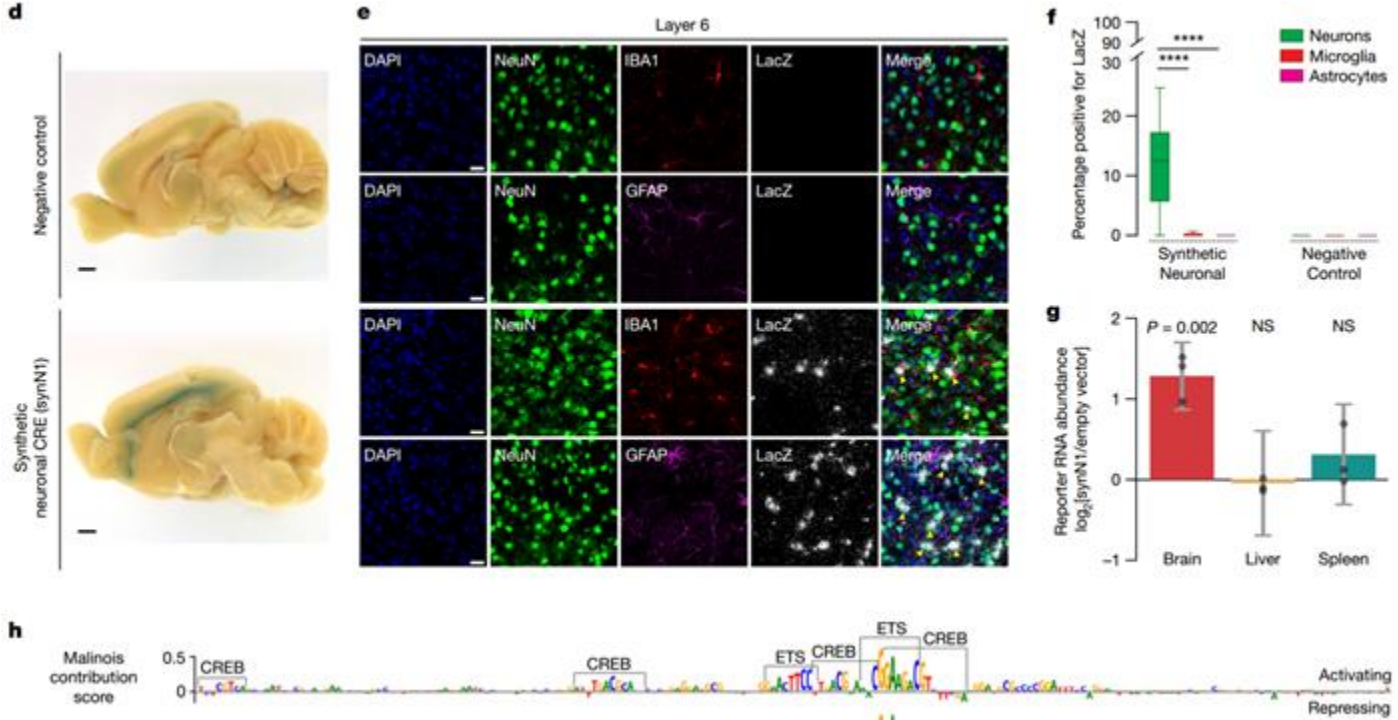


Figure 4

CODA CREs, screened by Enformer and tested in zebrafish, work in mice



Summary

- Platform for creating synthetic cell-type specific CREs from massively parallel reporter assay models
- CODA allows us to optimally explore a large decision space in finding the best performing CREs
- A convolutional neural network Malinois was trained on these CREs to predict the gene expression in each cell type
- The created synthetic CREs have good performance when validated in vivo, even when integrated directly into the mouse genome

Discussion and Limitations

- Advantages: High specificity, diversity of motifs, predictive power of Malinois.
- Challenges: Limitations in fully exploring sequence space, computational intensity, transferability across more diverse cell types.
- Future Directions: Expand to other cell types, refine CRE design for therapeutic use, incorporate other deep learning techniques.