Conditioning by adaptive sampling for robust design

David Brookes
Biophysics Graduate Group
University California, Berkeley

Jennifer Listgarten
EECS and Center for Computational Biology
University California, Berkeley
Motivating problem: design protein sequences

- Proteins are made up of sequences of amino acids (20 possibilities)
- Huge variety of proteins whose function we would like to improve
Motivating problem: design protein sequences

- Proteins are made up of sequences of amino acids (20 possibilities)
- Huge variety of proteins whose function we would like to improve
Motivating problem: design protein sequences

• Proteins are made up of sequences of amino acids (20 possibilities)
• Huge variety of proteins whose function we would like to improve

Proteins that fluoresce

... that act as drugs
Motivating problem: design protein sequences

- Proteins are made up of sequences of amino acids (20 possibilities)
- Huge variety of proteins whose function we would like to improve

... that fixate carbon in the atmosphere

... that act as drugs

Proteins that fluoresce
Motivating problem: design protein sequences

- Proteins are made up of sequences of amino acids (20 possibilities)
- Huge variety of proteins whose function we would like to improve

Proteins that fluoresce

... that act as drugs

... that fixate carbon in the atmosphere

... that deliver gene-editing tools to tissues
How to map sequence to function?

A law of molecular biology:

Sequence → Structure → Function

ex: fluorescence

http://www.rcsb.org/structure/6FWW
Bypassing the structure relationships

A law of molecular biology:

Sequence \[\text{VTDLQNSTEKFGFRLSALDV}\] Structure \[\text{Function}\]

High throughput experiments (& ML)

http://www.rcsb.org/structure/6FWW
Can we solve the inverse problem?

A law of molecular biology:

**Sequence** → **Structure** → **Function**

Design problem: Given a model, find sequences with desired function


http://www.rcsb.org/structure/6FWW
Why is protein design difficult?

- Huge, rugged search space
  \[ \Rightarrow \text{size scales as } 20^L \]
Why is protein design difficult?

• Huge, rugged search space
  \[\Rightarrow \text{size scales as } 20^L\]
• Discrete search space (no gradients)
Why is protein design difficult?

• Huge, rugged search space
  \[ \Rightarrow \text{size scales as } 20^L \]
• Discrete search space (no gradients)
• Uncertainty in predictor

https://livingthing.danmackinlay.name/gaussian_processes.html69
Possible solution: model-based optimization (MBO)

Idea: replace the standard (hard) objective

$$\max_{x \in \mathcal{X}} f(x)$$

e.g. the space of sequences
Possible solution: model-based optimization (MBO)

Idea: replace the standard (hard) objective with a potentially easier one

\[
\max_{x \in \mathcal{X}} f(x) \quad \rightarrow \quad \max_{\theta \in \mathbb{R}^d} \mathbb{E}_{p(x|\theta)}[f(x)]
\]

the space of sequences

model over sequence space
Possible solution: model-based optimization (MBO)

Idea: replace the standard (hard) objective with a potentially easier one

\[
\max_{x \in \mathcal{X}} f(x) \quad \rightarrow \quad \max_{\theta \in \mathbb{R}^d} \mathbb{E}_{p(x|\theta)}[f(x)]
\]

Solution approach is to iterate:
1. Sample from “search model” \( p(x|\theta) \)
2. Evaluate samples on \( f(x) \)
3. Adjust \( \theta \) so the model favors samples with large function evals
Possible solution: model-based optimization (MBO)

Idea: replace the standard (hard) objective with a potentially easier one

\[
\max_{x \in \mathcal{X}} f(x) \quad \rightarrow \quad \max_{\theta \in \mathbb{R}^d} \mathbb{E}_{p(x|\theta)}[f(x)]
\]

Solution approach is to iterate:

1. Sample from “search model” \( p(x|\theta) \)
2. Evaluate samples on \( f(x) \)
3. Adjust \( \theta \) so the model favors sequences with large function evals

* ✓ Model can sample broad areas of sequence space
* ✓ Does not require gradients of \( f \)
* ✓ Can incorporate uncertainty
First attempt at MBO for protein design: Design by Adaptive Sampling (DbAS)

Our aim is solve the MBO objective:

$$\underset{\theta}{\text{argmax}} \log \mathbb{E}_p(x|\theta) \left[ P(S|x) \right]$$
First attempt at MBO for protein design: Design by Adaptive Sampling (DbAS)

Our aim is solve the MBO objective:

$$\arg\max_{\theta} \log \mathbb{E}_{p(x|\theta)} \left[ P(S|x) \right]$$

where

- $p(x|\theta)$ is the search model (VAE, HMM...)
First attempt at MBO for protein design: Design by Adaptive Sampling (DbAS)

Our aim is to solve the MBO objective:

\[
\arg\max_{\theta} \log \mathbb{E}_{p(x|\theta)} \left[ P(S|x) \right]
\]

where

- \( p(x|\theta) \) is the search model (VAE, HMM...)
- \( S \) is the desired set of property values

\[ \rightarrow \text{e.g. fluorescence} > \alpha \]
First attempt at MBO for protein design: Design by Adaptive Sampling (DbAS)

Our aim is to solve the MBO objective:

$$\arg\max_{\theta} \log \mathbb{E}_{p(x|\theta)} [P(S|x)]$$

where

- $p(x|\theta)$ is the search model (VAE, HMM...)
- $S$ is the desired set of property values
  \(\rightarrow\) e.g. fluorescence > $\alpha$
- $P(S|x)$ is a stochastic predictive model (“oracle”) that maps sequences to property
Design by Adaptive Sampling (cont.)

Two issues:

1. $\theta$ is in the expectation distribution.

$$\arg\max_{\theta} \log \mathbb{E}_{p(x|\theta)} [P(S|x)]$$
Design by Adaptive Sampling (cont.)

Two issues:

1. $\theta$ is in the expectation distribution.

$\maximize a lower bound$

$$\arg\max_{\theta} \log \mathbb{E}_{p(x|\theta)} [P(S|x)] ,$$

$$\geq$$

$$\arg\max_{\theta} \mathbb{E}_{p(x|\theta^{(t)})} [P(S|x) \log p(x|\theta)]$$
Design by Adaptive Sampling (cont.)

Two issues:

1. $\theta$ is in the expectation distribution.
2. MC estimates for rare events.

\[
\begin{align*}
&\text{maximize a lower bound} \\
&\quad \arg\max_{\theta} \log \mathbb{E}_{p(x|\theta)} \left[ P(S|x) \right], \\
&\quad \downarrow \quad \geq \\
&\quad \arg\max_{\theta} \mathbb{E}_{p(x|\theta^{(t)})} \left[ P(S|x) \log p(x|\theta) \right]
\end{align*}
\]
Design by Adaptive Sampling (cont.)

Two issues:

1. $\theta$ is in the expectation distribution.
2. MC estimates for rare events.

maximize a lower bound

$$\arg\max_{\theta} \log \mathbb{E}_{p(x|\theta)} [P(S|x)],$$

$$\geq$$

$$\arg\max_{\theta} \mathbb{E}_{p(x|\theta^{(t)})} [P(S|x) \log p(x|\theta)]$$

anneal a sequence of relaxations:

$S^t \rightarrow S$, where $S^t \supset S^{t+1}$
Design by Adaptive Sampling (cont.)

Two issues:

1. $\theta$ is in the expectation distribution.

2. MC estimates for rare events.

To maximize a lower bound:

$$\arg \max_{\theta} \log \mathbb{E}_{p(x|\theta)} [P(S|x)] ,$$

$$\geq$$

$$\arg \max_{\theta} \mathbb{E}_{p(x|\theta^{(t)})} [P(S|x) \log p(x|\theta)]$$

Anneal and MC

$$\theta^{(t+1)} = \arg \max_{\theta} \sum_{i=1}^{M} P(S^{(t)}|x_i^{(t)}) \log p(x_i^{(t)}|\theta)$$
Design by Adaptive Sampling (cont.)

Two issues:

1. $\theta$ is in the distribution.
2. MC estimates for rare events.

Assumes oracle is unbiased and has good uncertainty estimates

maximize a lower bound

\[
\max \mathbb{E}_{\theta(t)} \left[ P(S|x) \right] - \mathbb{E}_{\theta(t)} \left[ P(S|x) \log p(x|\theta) \right] 
\]

Anneal and MC

\[
\theta^{(t+1)} = \arg\max_\theta \sum_{i=1}^{M} P(S^{(t)}|x^{(t)}_i) \log p(x^{(t)}_i|\theta) 
\]
How pathological oracles lead you astray
How pathological oracles lead you astray

Acceptable

Many training examples
How pathological oracles lead you astray

Acceptable

Many training examples

Pathological

 Fewer training examples
How pathological oracles lead you astray

Idea: estimate training distribution of $x$ conditioned on high values of oracle
Fixing pathological oracles w/ conditioning

Idea: estimate training distribution of $x$ *conditioned* on high values of oracle
Fixing pathological oracles w/ conditioning

Idea: estimate training distribution of $x$ conditioned on high values of oracle

Don’t have access to training distribution, but can build a model $p(x|\theta^{(0)})$ to approximate it
Conditioning by Adaptive Sampling (CbAS)

Previous formulation:

\[
\text{argmax}_\theta \log \mathbb{E}_{p(x|\theta)} \left[ P(S|x) \right] \geq \text{argmax}_\theta \mathbb{E}_{p(x|\theta(t))} \left[ P(S|x) \log p(x|\theta) \right] \]

\[
\theta^{(t+1)} = \text{argmax}_\theta \sum_{i=1}^{M} P(S^{(t)}|x_i^{(t)}) \log p(x_i^{(t)}|\theta)
\]

Anneal and MC

New formulation:

\[
\text{argmin}_\theta D_{KL} \left( p(x|S, \theta^{(0)}) || p(x|\theta) \right)
\]

\[p(x|\theta^{(0)})\] models the training distribution
Conditioning by Adaptive Sampling (CbAS)

Previous formulation:

$$\underset{\theta}{\arg\max} \log \mathbb{E}_{p(x|\theta)}[P(S|x)] \geq \underset{\theta}{\arg\max} \mathbb{E}_{p(x|\theta^{(t)})}[P(S|x) \log p(x|\theta)]$$

Anneal and MC

$$\theta^{(t+1)} = \underset{\theta}{\arg\max} \sum_{i=1}^{M} P(S^{(t)}|x_i^{(t)}) \log p(x_i^{(t)}|\theta)$$

New formulation:

$$\underset{\theta}{\arg\min} D_{KL} \left( p(x|S, \theta^{(0)}) \| p(x|\theta) \right) = \underset{\theta}{\arg\max} \mathbb{E}_{p(x|\theta^{(0)})}[P(S|x) \log p(x|\theta)]$$
Conditioning by Adaptive Sampling (CbAS)

Previous formulation:

\[
\begin{align*}
\argmax_{\theta} \log \mathbb{E}_{p(x|\theta)} [P(S|x)] \\
\geq \\
\argmax_{\theta} \mathbb{E}_{p(x|\theta(t))} [P(S|x) \log p(x|\theta)]
\end{align*}
\]

Anneal and MC

\[\theta^{(t+1)} = \argmax_{\theta} \sum_{i=1}^{M} P(S^{(t)}|x^{(t)}_i) \log p(x^{(t)}_i|\theta)\]

New formulation:

\[
\begin{align*}
\argmin_{\theta} D_{KL} \left(p(x|S, \theta^{(0)}) || p(x|\theta)\right) \\
= \\
\argmax_{\theta} \mathbb{E}_{p(x|\theta^{(0)})} [P(S|x) \log p(x|\theta)]
\end{align*}
\]

Can’t anneal when sampling dist. doesn’t change!
Conditioning by Adaptive Sampling (CbAS)

Previous formulation:

\[
\begin{align*}
\text{argmax} & \mathbb{E}_{p(x|\theta^{(t)})} [P(S|x) \log p(x|\theta)] \\
& \Downarrow \geq \\
& \text{Anneal and MC} \\
\theta^{(t+1)} & = \text{argmax} \sum_{i=1}^{M} P(S^{(t)}|x_{i}^{(t)}) \log p(x_{i}^{(t)}|\theta)
\end{align*}
\]

New formulation:

\[
\begin{align*}
& \text{argmin} \quad D_{KL} \left( p(x|S, \theta^{(0)}) || p(x|\theta) \right) \\
& \Downarrow = \\
& \text{argmax} \quad \mathbb{E}_{p(x|\theta^{(0)})} [P(S|x) \log p(x|\theta)] \\
& \Downarrow = \\
& \text{argmax} \quad \mathbb{E}_{p(x|\theta^{(t)})} \left[ \frac{p(x|\theta^{(0)})}{p(x|\theta^{(t)})} P(S|x) \log p(x|\theta) \right]
\end{align*}
\]

Importance sampling proposal dist.
Conditioning by Adaptive Sampling (CbAS)

Previous formulation:

$$\arg\max_{\theta} \log \mathbb{E}_{p(x|\theta)} [P(S|x)] \geq \arg\max_{\theta} \mathbb{E}_{p(x|\theta^{(t)})} [P(S|x) \log p(x|\theta)]$$

$$\theta^{(t+1)} = \arg\max_{\theta} \sum_{i=1}^{M} P(S^{(t)}|x_{i}^{(t)}) \log p(x_{i}^{(t)}|\theta)$$

New formulation:

$$\arg\min_{\theta} D_{KL} \left( p(x|S, \theta^{(0)}) || p(x|\theta) \right) = \arg\max_{\theta} \mathbb{E}_{p(x|\theta^{(0)})} [P(S|x) \log p(x|\theta)]$$

$$\arg\max_{\theta} \mathbb{E}_{p(x|\theta^{(t)})} \left[ \frac{p(x|\theta^{(0)})}{p(x|\theta^{(t)})} P(S|x) \log p(x|\theta) \right]$$

$$\theta^{(t+1)} = \arg\max_{\theta} \sum_{i=1}^{M} \frac{p(x_{i}^{(t)}|\theta^{(0)})}{p(x_{i}^{(t)}|\theta^{(t)})} P(S^{(t)}|x_{i}^{(t)}) \log p(x_{i}^{(t)}|\theta)$$
Testing is fundamentally different

• We don’t trust our oracle and generally can’t query the ground truth
Testing is fundamentally different

• We don’t trust our oracle and generally can’t query the ground truth
• We can’t hold-out a test set of good sequences
  • Near-zero chance of any of these sequences being found by the method
Testing is fundamentally different

• We don’t trust our oracle and generally can’t query the ground truth

• We can’t hold-out a test set of good sequences
  • Near-zero chance of any of these sequences being found by the method

• We can’t use some canonical test function as the oracle
  • In our problem it is untrustworthy
Testing strategy

- Simulate a ground truth based on real data
  → “Ground truth” is a GP mean function
Testing strategy

• Simulate a ground truth based on real data
  → “Ground truth” is a GP mean function
• Ground truth values values are sampled from the GP for given sequences
• Use these input-output pairs to train oracles.
Testing strategy

• Simulate a ground truth based on real data
  → “Ground truth” is a GP mean function
• Ground truth values values are sampled from the GP for given sequences
• Use these input-output pairs to train oracles
• Coerce training set so these oracles exhibit pathologies
Results
Results

Model-based optimizations

Use weighted ML updates with weights:

- **CbAS:** \( \frac{p(x|\theta^{(0)})}{p(x|\theta^{(t)})} P(S^{(t)}|x) \)
- **DbAS:** \( P(S^{(t)}|x) \)
- **RWR:** \( e^{\alpha f(x)} \)
- **CEM-PI:** \( \mathbb{1}_{\{PI(x)>\gamma^{(t)}\}}(x) \)
- **FB-VAE:** \( \mathbb{1}_{\{f(x)>\gamma^{(t)}\}}(x) \) w/ additional considerations
Results

Model-based optimizations

Gradient descent on latent spaces
Results

What does each bar represent?

<table>
<thead>
<tr>
<th></th>
<th>Oracle</th>
<th>Ground truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq9</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Results

What does each bar represent?

<table>
<thead>
<tr>
<th></th>
<th>Oracle</th>
<th>Ground truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seq9</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Results
Wrap-up

• Introduced a new model-based optimization method that is robust to pathological oracles
• Specifically targeted for discrete design problems
• Ongoing work to move beyond proof-of-principle:
  • Collaboration with wet-lab to perform end-to-end validation
Thanks!

Funding: