OPTIMIZATION OF ADVANCED MARKO CHAIN MONTE CARLO METHODS WITH APPLICATIONS TO BIOLOGICAL SYSTEMS

Agenda

- Overview
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Agenda

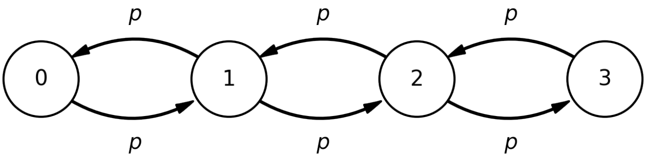
- Overview
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Markov Chains ...

- A Markov chain is a stochastic process $X_0, X_1, \ldots$ which takes values from some state space Θ and where the following properties hold:
  - $\mathbb{P}(X_0 = \theta_0) = \lambda(\theta_0)$
  - $\mathbb{P}(X_{t+1} = \theta_{t+1} | X_t = \theta_t, \ldots, X_0 = \theta_0) = \mathbb{P}(X_{t+1} = \theta_{t+1} | X_t = \theta_t) =: P(\theta_t, \theta_{t+1})$

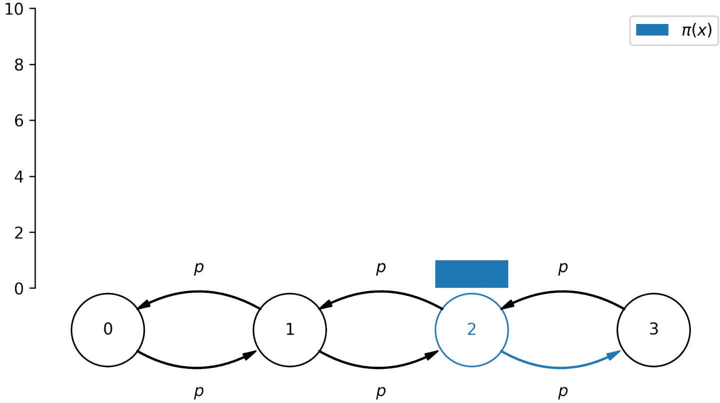
  where λ is a distribution over Θ

- An example: $\Theta = [0, 3] \cap \mathbb{N}$ and $P = \begin{pmatrix} p & p & & \\ p & & p & \\ & p & & p \\ & & p & p \end{pmatrix}$ with $p = 0.5$



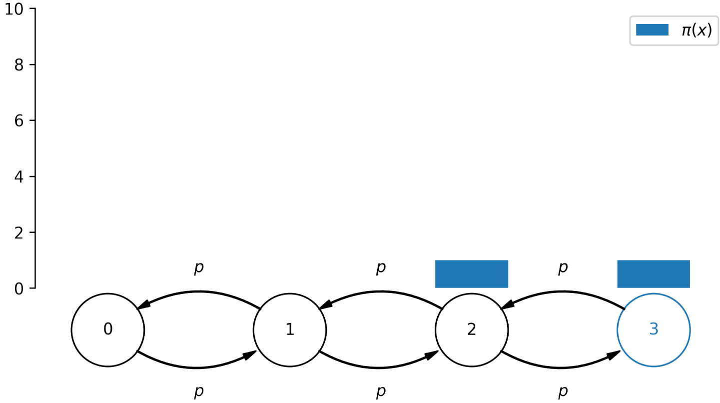
Markov Chains ...

- We simulate and count ...



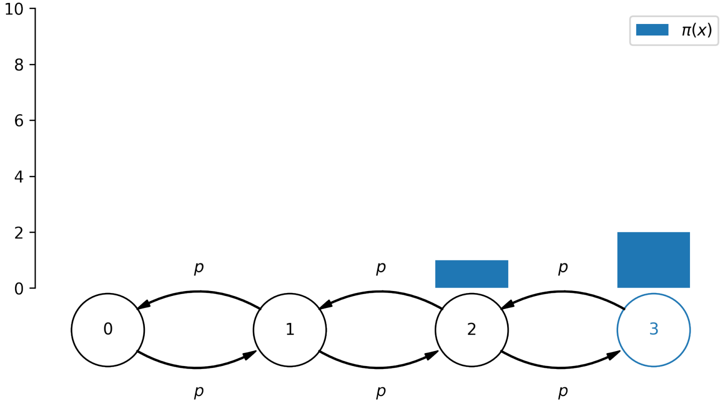
Markov Chains ...

- We simulate and count ...



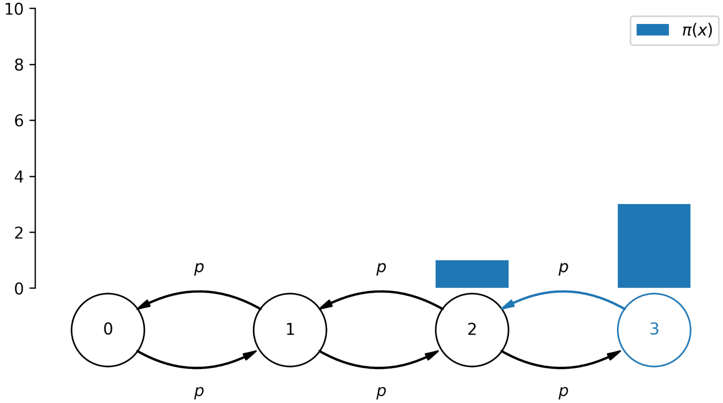
Markov Chains ...

- We simulate and count ...



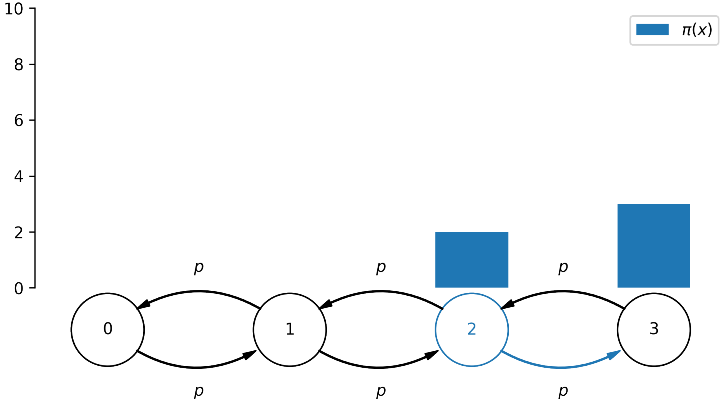
Markov Chains ...

- We simulate and count ...



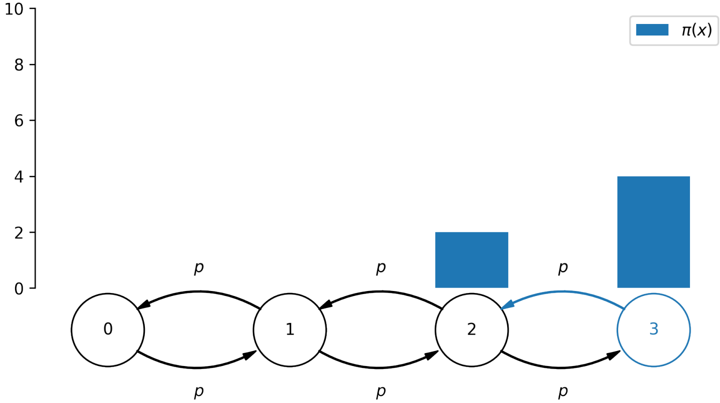
Markov Chains ...

- We simulate and count ...



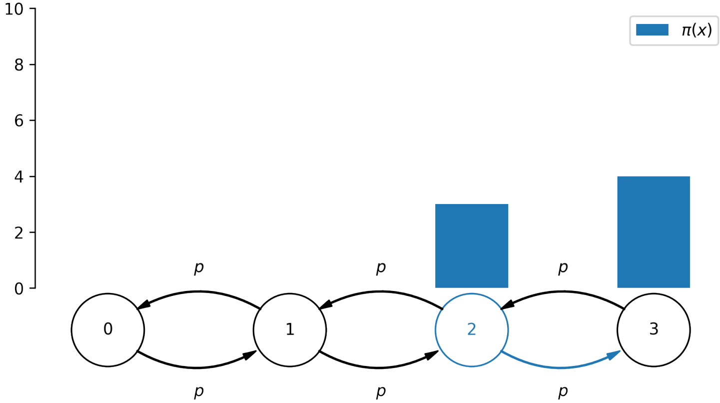
Markov Chains ...

- We simulate and count ...



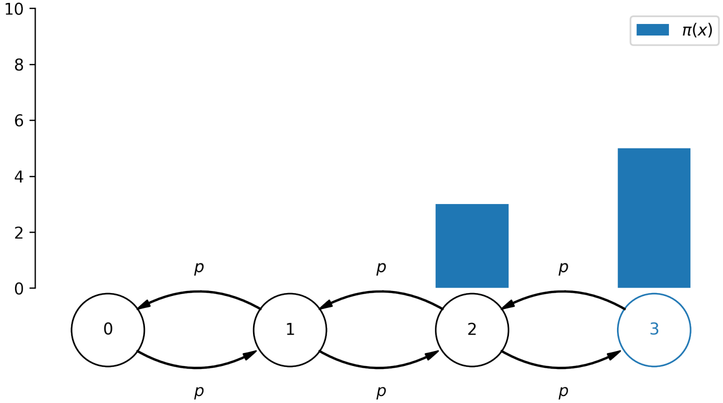
Markov Chains ...

- We simulate and count ...



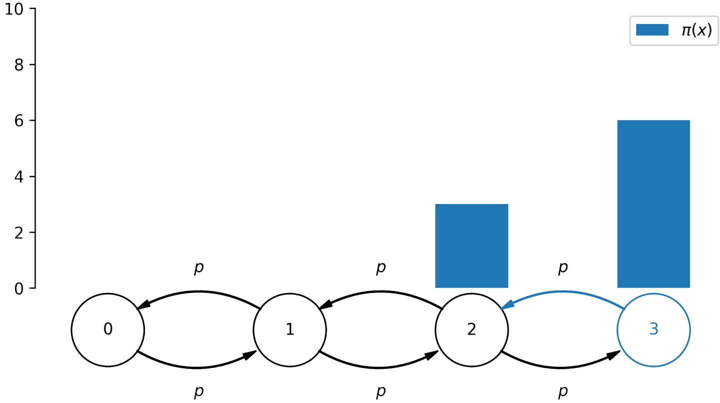
Markov Chains ...

- We simulate and count ...



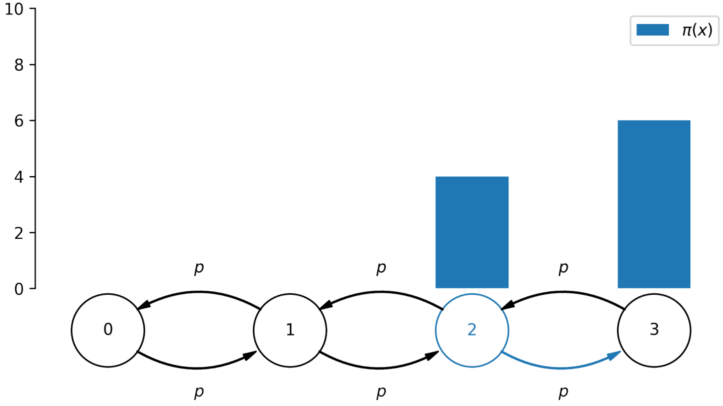
Markov Chains ...

- We simulate and count ...



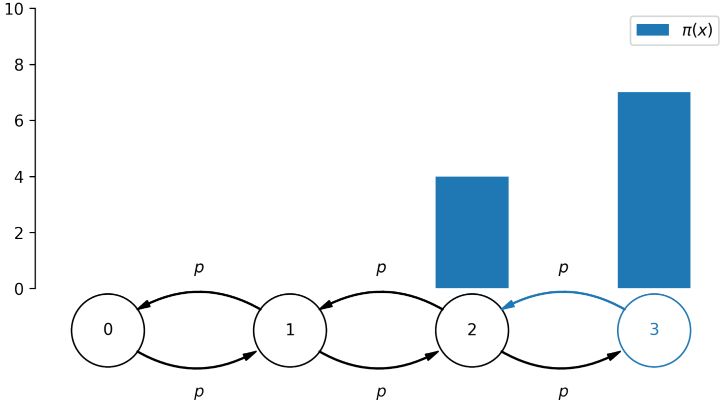
Markov Chains ...

- We simulate and count ...



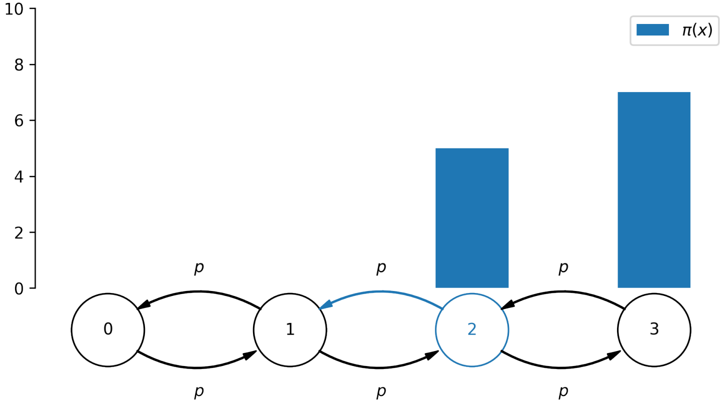
Markov Chains ...

- We simulate and count ...



Markov Chains ...

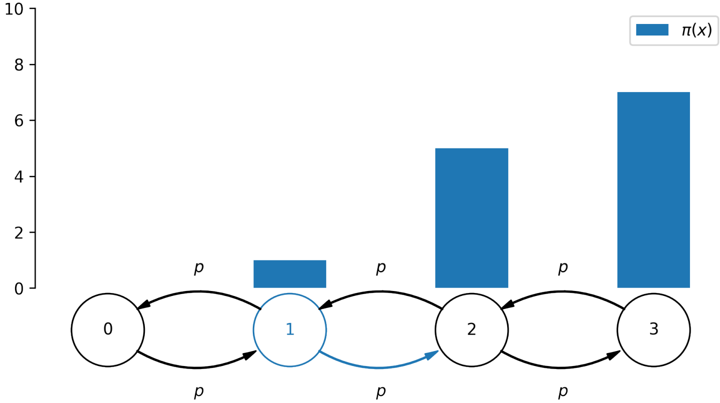
- We simulate and count ...



JÜLICH Forschungszentrum

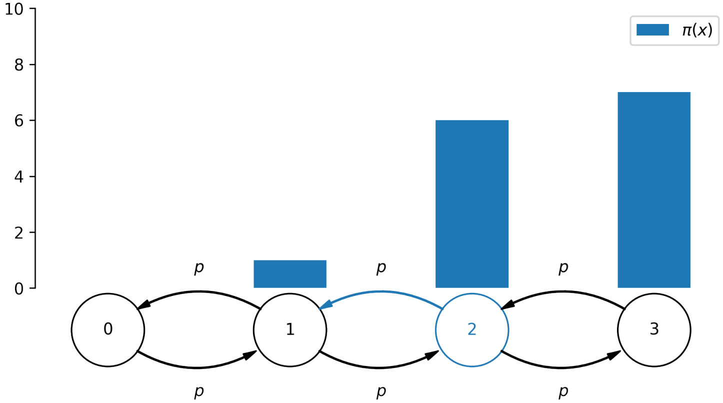
Markov Chains ...

- We simulate and count ...



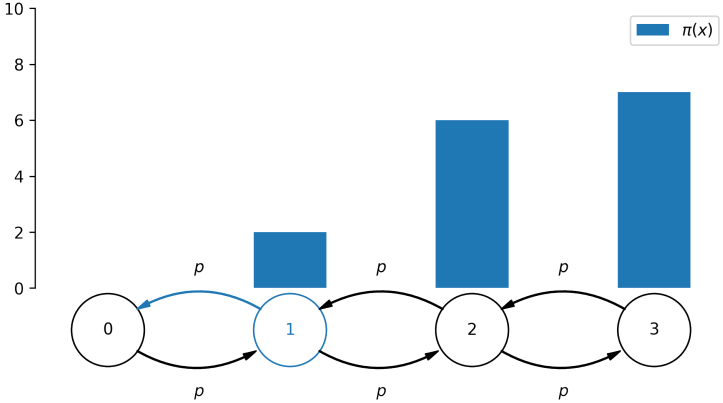
Markov Chains ...

- We simulate and count ...



Markov Chains ...

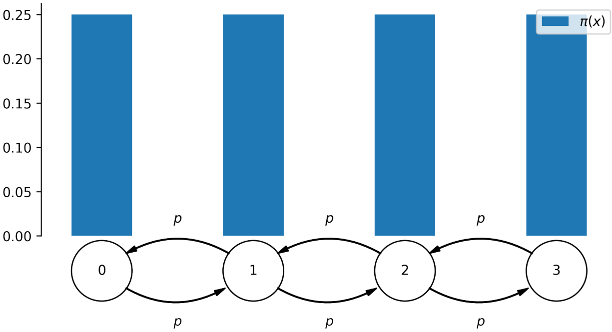
- We simulate and count ...



Markov Chains ...

- At some point, we would reach the invariant or equilibrium distribution π

$$\pi = P\pi$$



- Theory also tells us, that under suitable conditions (e.g finite Θ), we have that

$$\lim_{n\to\infty} (P^n)_i = \pi \quad \forall i \in \Theta$$

Markov Chains ...

- We went from given transition matrix P to the **unique** invariant distribution π
- We can also go from given invariant distribution π (or measure!) to **non-unique** transition probabilities P!
- Given π, we have

$$\pi_i = \sum_j \pi_j P_{ji}$$

$$\underbrace{\sum_j P_{ij}}_{=1} \pi_i = \sum_j \pi_j P_{ji}$$

- Now,

$$\underbrace{\pi_i P_{ij} = \pi_j P_{ji}}_{\text{detailed balance condition}} \implies \sum_j P_{ij}\pi_i = \sum_j \pi_j P_{ji}$$

Markov Chains ...

- First, rearrange ...

$$\pi_i P_{ij} = \pi_j P_{ji} \implies \frac{P_{ij}}{P_{ji}} = \frac{\pi_j}{\pi_i}$$

- ... and finally choose

$$P_{ij} = \min\left\{1, \frac{\pi_j}{\pi_i}\right\}$$

the famous **Metropolis Filter**

… & Stepsizes

- In continuous space, we split the transition probability $P(\theta, \theta')$ into proposal $q(\theta, \theta^*)$ and acceptance probability $\alpha(\theta, \theta^*)$ and use

$$\alpha(\theta, \theta^*) = \min\left\{1, \frac{\pi(\theta^*)q(\theta^*, \theta)}{\pi(\theta)q(\theta, \theta^*)}\right\}$$

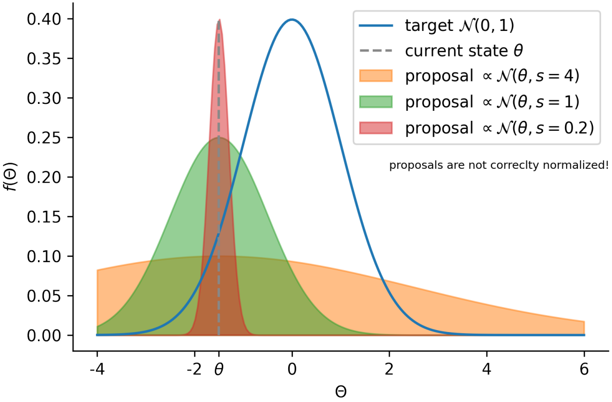
- q usually depends on some parameters, e.g. using

$$q(\theta, \theta^*) = q_s(\theta^*|\theta) \sim \mathcal{N}(\theta, s),$$

what stepsize s should we use?

… & Stepsizes

- What s should we use?



- Again: Transition kernels to a given invariant distribution are **not unique**

Agenda

- Overview
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Effectiveness ...

- Given samples $\theta_0, \ldots, \theta_n$, the sample average is $\bar{\theta} = \frac{1}{n}\sum_{i=0}^{n}\theta_i$ and the squared statistical error in $\bar{x}$ is the variance

$$
\begin{aligned}
\mathrm{Var}[\bar{\theta}] &= \mathbb{E}[\bar{\theta}^2] - \mathbb{E}[\bar{\theta}]^2 \\
&= \frac{\sigma^2}{n^2}\left(n + \sum_{t=1}^{n-1}(n-t)\rho_t\right) \quad \text{with} \quad \rho_t = \frac{\mathbb{E}[\theta_0\theta_t] - \mu^2}{\sigma^2} \\
&= \frac{\sigma^2}{n}\left(1 + \sum_{t=1}^{n-1}\left(1 - \frac{t}{n}\right)\rho_t\right) < \frac{\sigma^2}{n}\underbrace{\left(1 + \sum_{t=1}^{n-1}\rho_t\right)}_{=:\tau}
\end{aligned}
$$

- From this we derive the effective sample size as

$$
\mathrm{Var}[\bar{x}] = \frac{\sigma^2}{n_{\mathrm{eff}}} \quad \text{with} \quad n_{\mathrm{eff}} = \frac{n}{\tau}
$$

Effectiveness ...

- The effective sample size is commonly used as a measure of quality for Markov chains
  - Note that $\tau = \tau\big((\theta_t)_{t=0}^n\big)$ and $(\theta_t)_{t=0}^n = \big((\theta_t)_{t=0}^n\big)(s)$
- We now increase effectiveness by reducing τ, since $n_{\mathrm{eff}} = \frac{n}{\tau}$
- Autocorrelations are usually assumed to be decreasing in time: $\rho_t \geq \rho_{t+k}$, for $k \geq 1$
- Thus, **decrease autocorrelation for small lags**!

... & Expected Squared Jump Distance

- We define

$$
\begin{aligned}
\mathrm{ESJD}(\theta) := \mathbb{E}[(\theta_{t+1} - \theta_t)^2] &= \underbrace{\mathbb{E}[\theta_{t+1}^2]}_{=\mathbb{E}[\theta^2]} - 2\mathbb{E}[\theta_{t+1}\theta_t] + \underbrace{\mathbb{E}[\theta_t^2]}_{=\mathbb{E}[\theta^2]} \\
&= 2\big(\mathbb{E}[\theta^2] - \mathbb{E}[\theta_{t+1}\theta_t] + \mu^2 - \mu^2\big) \\
&= 2\big(\sigma^2 - \big(\mathbb{E}[\theta_{t+1}\theta_t] - \mu^2\big)\big) \\
&= 2\sigma^2\big(1 - \frac{\mathbb{E}[\theta_{t+1}\theta_t] - \mu^2}{\sigma^2}\big) \\
&= 2\sigma^2\big(1 - \rho_1\big)
\end{aligned}
$$

- Since σ^2 is constant, maximizing the ESJD reduces the lag-1 autocorrelation, **increasing efficiency**.

... & Expected Squared Jump Distance

- However, the ESJD is a function of the **unknown** true target distribution, so we have to **estimate** it.
- We have a guess, how the function $\mathrm{ESJD}(s)$ looks like, but in general, it remains a black box function.

$\implies$ **Thompson Sampling**

Agenda

- Overwiew
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Thompson Sampling

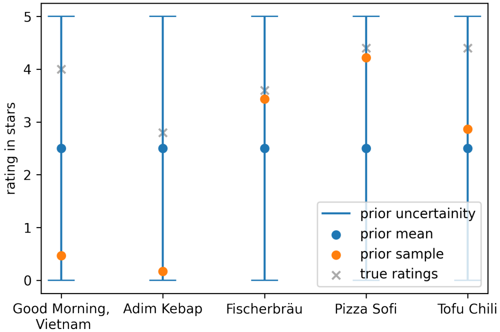
- A Bayesian optimization approach to optimize an unknown **black box** objective function:
  - We can evaluate it pointwise, but that's it!
- Thus, we have to explore the function and search for the optimum simultaneously

- Thompson Sampling:
  - Assume a distribution over the possible shapes of the black box function
  - Sample that distribution and evaluate the black box function, where the sampled function is maximized
  - Update the distribution over the possible function shapes based on the observed evaluation

Going to the restaurant!

- You're coming to Vienna and there exist a total of 5 different restaurants, the internet has not yet been invented, I never went out in 8 years here
- Any restaurant might be good or bad, so we choose the first one uniformly

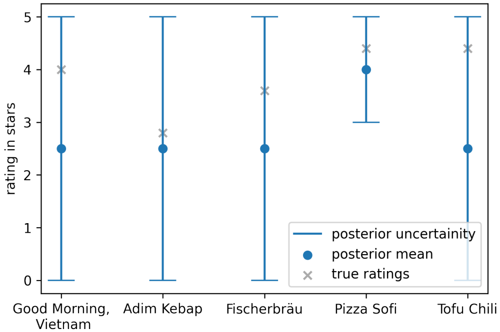
$\Longrightarrow$ We already assumed a (uniform) distribution over the restaurants quality!



Going to the restaurant!

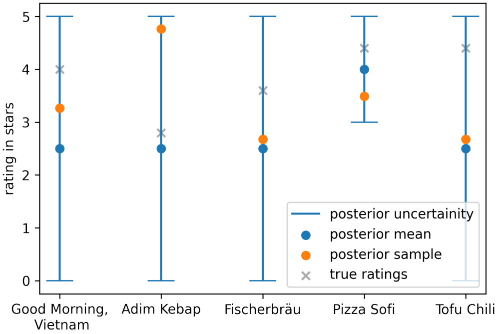
- We go to Pizza Sofi and rate it with 4 stars, but we only had a fraction of their menu
- Would we have rated the same, if we had eaten something else? Did the cook have an exceptional good day?

$\Longrightarrow$ There is function noise and uncertainity left in our rating!



Going to the restaurant!

- Once more, we want to go out eating. What restaurant to choose?
- ”Sample” from your posterior belief about the quality of the restaurants and go to the one, where your sample tells you, that you will have the best experience:
- Afterwards, update your beliefs based on your experience.



Dinner with William Thompson and Andrey Markov

- Our restaurants are the stepsizes s, that we test.

- Our friends, with whom we go to the restaurant, is the number of chains, that we use.
  $\Longrightarrow$ Each friend rates the restaurant differently, each chain measures another value for the objective function

- The amount of money we spend, is the number of samples, we use to test the stepsize
  $\Longrightarrow$ We never take everything, that the restaurant has to offer. But taking more than just a salad is likely to reveal more about the kitchen's qualities, that is reduce the error in the estimate of the objective function

More formal Thompson Sampling

- We model our distribution over the possible black box functions as a Gaussian Process (GP)
- The algorithm then is:

for $t \in 0, \ldots, m$ **do**
  sample $\hat{f}_t \sim GP(\mu_t, \Sigma_t)$
  let $x_t := \arg\max \hat{f}_t(x)$
  let $y_t := f(x_t)$
  let $D_{t+1} := D_t \cup \{(x_t, y_t)\}$
  update μ_{t+1} using D_{t+1}
  update Σ_{t+1} using D_{t+1}
end

JÜLICH Forschungszentrum

Agenda

- Overwiew
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Once more: Why not to tune the acceptance rate!

-

Agenda

- Overwiew
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Results

Agenda

- Overwiew
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Acceptance Rate Tuning with Thompson Sampling

- Formally, the acceptance rate is

$$\alpha = \int_{\Theta^2} \underbrace{\pi(\theta)}_{!!} q(\theta, \theta^*) \alpha(\theta, \theta^*) \,\mathrm{d}\theta \,\mathrm{d}\theta^*$$

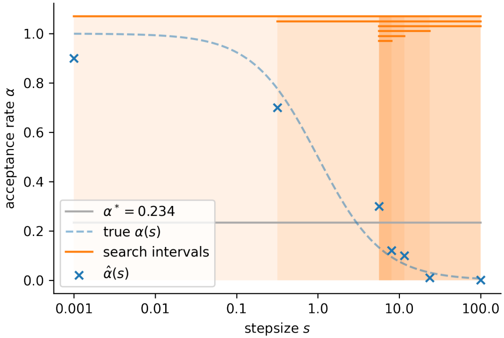
  and thus it is a function of the **unknown** true target distribution
- Once more, we can only **estimate** it:

$$\hat{\alpha}(s) = \frac{1}{n} \sum_{i=0}^{n} \mathbb{1}_{\{\theta_{t+1} \neq \theta_t\}} = \frac{\#\text{accepted moves}}{n}$$

- As a function of the stepsize the acceptance rate is **strictly monotonous**
- Use **binary search** to find the intersection of $\alpha(s)$ with some desired target value α^*?

Acceptance Rate Tuning with Thompson Sampling

- Use **binary search** to find the intersection of $\alpha(s)$ with some desired target value α^*?



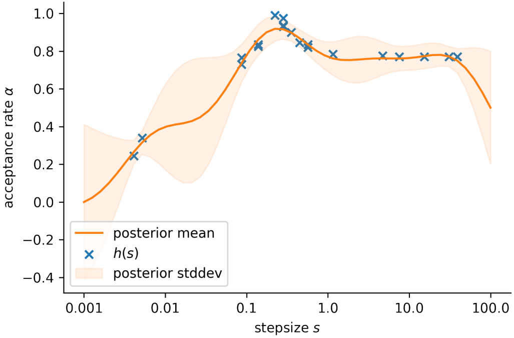
- **No!** This fails with noisy function evaluations which the estimator $\hat{\alpha}(s)$ gives us!
  $\rightarrow$ too deterministic! Instead, use Thompson Sampling!

Acceptance Rate Tuning with Thompson Sampling

- Given α^* and $\hat{\alpha}(s)$, define the score function:

$$h(s) := 1 - \Delta_\alpha := 1 - |\alpha^* - \hat{\alpha}(s)|$$

which is maximized when $\hat{\alpha}(s) = \alpha^*$ and takes values in $[0, 1]$



Agenda

- Overwiew
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Tune higher-order Autocorrelation Lags

- Recap the definition $\mathrm{ESJD}(\theta) = \mathbb{E}[(\theta_{t+1} - \theta_t)^2]$ and observe, that

$$
\begin{aligned}
\mathbb{E}[(\theta_{t+k} - \theta_t)^2] = &= \underbrace{\mathbb{E}[\theta_{t+k}^2]}_{=\mathbb{E}[\theta^2]} - 2\mathbb{E}[\theta_{t+k}\theta_t] + \underbrace{\mathbb{E}[\theta_t^2]}_{=\mathbb{E}[\theta^2]} \\
&= 2\left(\mathbb{E}[\theta^2] - \mathbb{E}[\theta_{t+k}\theta_t] + \mu^2 - \mu^2\right) \\
&= 2\left(\sigma^2 - \left(\mathbb{E}[\theta_{t+k}\theta_t] - \mu^2\right)\right) \\
&= 2\sigma^2\left(1 - \frac{\mathbb{E}[\theta_{t+k}\theta_t] - \mu^2}{\sigma^2}\right) \\
&= 2\sigma^2\left(1 - \rho_k\right)
\end{aligned}
$$

Tune higher-order Autocorrelation Lags

- So we can minimize the lag-1 and lag-2 autocorrelation by maximizing the 1 and 2-jump ESJD:

$$
\begin{aligned}
\mathbb{E}[(\theta_{t+1} - \theta_t)^2] + \mathbb{E}[(\theta_{t+2} - \theta_t)^2] &= 2\sigma^2\big(1 - \rho_1\big) + 2\sigma^2\big(1 - \rho_2\big) \\
&= 2\sigma^2\big(2 - \rho_1 - \rho_2\big)
\end{aligned}
$$

Agenda

- Overwiew
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Gaussian Process Time Costs

- Recap Thompson Sampling:

  for $t \in 0, \ldots, m$ **do**

  sample $\hat{f}_t \sim GP(\mu_t, \Sigma_t)$; // $O(k(tc)^2)$

  let $x_t := \arg\max \hat{f}_t(x)$; // $O(k)$

  let $y_t := f(x_t)$; // $O(cn)$

  let $D_{t+1} := D_t \cup \{(x_t, y_t)\}$

  update μ_{t+1} using D_{t+1} ; // $O((tc)^3)$

  update Σ_{t+1} using D_{t+1} ; // $O((tc)^3)$

  end

  with $k = |X|$ the search grid, n test samples, c parallel chains and m iterations rounds.
- Overall, we have $O((mc)^4)^*$

* oof

Agenda

- Overwiew
  - Markov Chains & Stepsizes
  - Effectiveness & Expected Squared Jump Distance
  - Thompson Sampling
  - Once more: Why not to tune the acceptance rate!
- Results
- Improvements & More
  - Acceptance Rate Tuning with Thompson Sampling
  - Tune higher-order Autocorrelation Lags
  - Gaussian Process Time Costs
  - Monotonicity of Thinning

Monotonicity of Thinning

-

Thanks!