Decision Making

Kajal Dodhia, Harry Newton, Joe Rutherford, Lanya Yang

Group 1

STOR-i | Lancaster University

Introduction

We will be investigating three main problems:

1. Section A: Stochastic & Deterministic Utility Functions
2. Section B: Regret Based Decision Making
3. Section C: Bayesian Optimisation

Utility Function

- We want to select an action A. Which do you choose?
- What do we mean by best action?

We consider the existence of a utility function $u : A \to \mathbb{R}$. So decision-making becomes:

$$a \in \underset{a \in A}{\operatorname{argmax}}\ u(a).$$

Example

- A utility function can be seen as a representation to define individual preferences for goods or services beyond the explicit financial value of those goods or services.
- In other words, it is a calculation for how much someone desires something, and it is relative.
- For example, it could be used to encode objectives and preferences in investor portfolios.
- The functions allow one to place a score on outcomes and then identify optimal portfolios by maximizing utility.

Restaurant Problem

We have been provided with some data that rates the quality of restaurants in Lancaster with 0 being the worst and 10 being the best. The data is in the table below:

MacDonalds	Sultan	Blue Moon	QSF
4	2	5	3
5	6		5
			6

Our aim is to find which is the best restaurant to go to by finding:

$$\hat{a} = \underset{a \in A}{\operatorname{argmax}} \mathbb{E}_{X \sim p_X(\cdot|D,a)}[u(x,a)].$$

Model

For each restaurant a, we assume that $u(a) \sim \mathrm{N}\left(\mu_a, \sigma_a^2\right)$ independent across restaurants.

We then assume standard priors:

$$\begin{aligned} \sigma_a^2 &\sim \mathrm{Inv-Gamma}(\alpha, \beta), \\ \mu_a \mid \sigma_a^2 &\sim \mathrm{N}\left(m, \sigma_a^2 \kappa\right), \end{aligned}$$

where α, β, m, κ are hyper parameters.

Calculating the Posterior

We calculate the posterior distribution for any restaurant a.

$$
\begin{aligned}
p\left(\mu_a, \sigma_a^2 \mid x_a\right) &\propto f\left(x_a \mid \mu_a, \sigma_a^2\right) p\left(\mu_a, \sigma_a^2\right) \\
&= f\left(x_a \mid \mu_a, \sigma_a^2\right) p\left(\sigma_a^2\right) p\left(\mu_a \mid \sigma_a^2\right) \\
&\vdots \\
&\propto \left(\sigma_a^2\right)^{\alpha+3/2+n_a/2} \times \\
&\exp\left(-\frac{1}{\sigma_a^2}\left(\beta + \frac{1}{2}\kappa\left(\mu_a - m\right)^2 + \frac{1}{2}\sum_{i=1}^{n_a}\left(x_{a,i} - \mu_a\right)^2\right)\right)
\end{aligned}
$$

Full Conditionals 1

$$p\left(\sigma_a^2 \mid \mu_a, x_a\right) \sim \text{Inv-Gamma}(A, B)$$

where

$$A = \alpha + \frac{1}{2} + \frac{n_a}{2},$$

$$B = \beta + \frac{1}{2\kappa}\left(\mu_a - m\right)^2 + \frac{1}{2}\sum_{i=1}^{n_a}\left(x_{a,i} - \mu_a\right)^2$$

Full Conditionals 2

and we have,

$$p\left(\mu_a \mid \sigma_a^2, x_a\right) \sim \mathrm{N}\left(\frac{m + \kappa \sum_{i=1}^{n_a} x_{a,i}}{1 + \kappa n_a}, \frac{\kappa \sigma_a^2}{1 + \kappa n_a}\right)$$

Gibbs Sampling

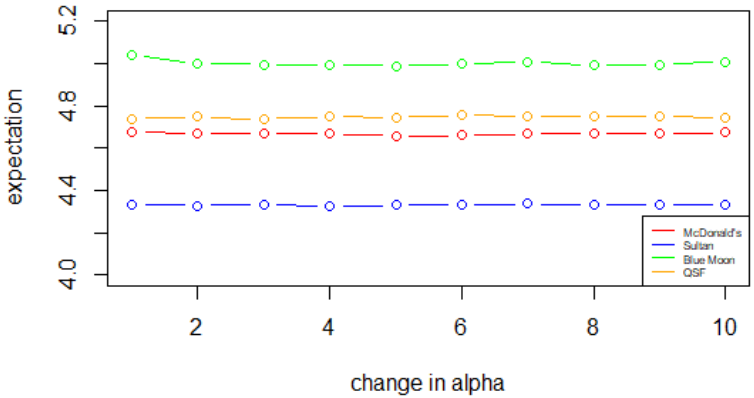
- Initialise starting hyperparameters: $m = 5$, $\kappa = 1$, $\alpha = 1$, $\beta = 1$.
- **Step 1:** For restaurant a, starting from $(\mu_a, \sigma_a^2) = (5, 5)$, generate a Markov Chain with length 1000 using Gibbs sampler and get an approximation $(\hat{\mu}_a, \hat{\sigma_a}^2)$.
- **Step 2:** Repeat step 1 1000 times to get 1000 pairs of $(\hat{\mu}_a, \hat{\sigma}_a)$.
- **Step 3:** For each pair we generate 1000 samples from $\mathcal{N}(\hat{\mu}_a, \hat{\sigma_a}^2)$ and find the sample mean $\hat{u}_a$ for each pair.
- **Step 4:** Then we take the average to get an estimate for the expectation for each restaurant a

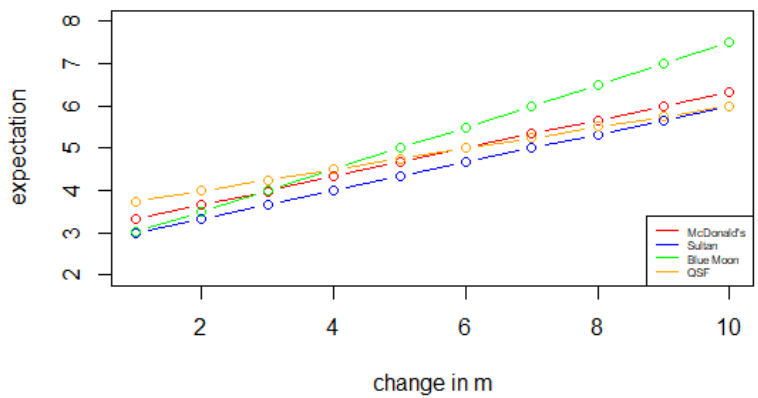
Results

	McDonalds	Sultan	Blue Moon	QSF
$\mathbb{E}(u(a) \mid \mathcal{D})$	4.6748	4.3371	5.004	4.7349

The best restaurant to visit is Blue Moon.

Result 2





- Changing α, β and κ, showed no changes in which restaurant was the best.
- When m is less than 4, QSF is the best, while Blue Moon is the best when m is larger or equal to 4.
- Sultan always performs the worst.

Further research

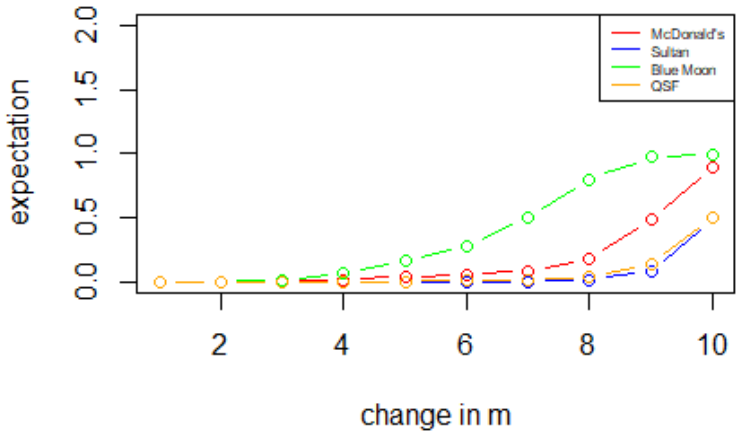


Figure 1: Change in m for $\mathbb{E}[\mathbb{P}(u(a) > 6|\mathcal{D})]$

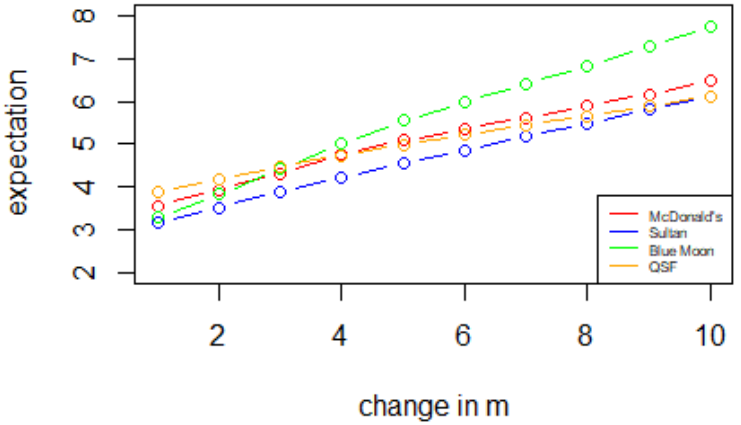


Figure 2: Change in m for 0.75-quantile

Conclusions and Further Research

- Blue Moon is the best restaurant to visit
- Our results show that the expectation did not depend significantly on the hyperparameters α, β and κ.
- Considered changing more of the fixed hyperparameters.
- Changing the data to see what happens

Defining Risk Behaviour Type

Gamble: I have just given you $\$50$ up front, you now have a choice.

1. Keep it
2. I flip a coin. Heads: earn an extra $\$25$. Tails: I take $\$25$.

Defining Risk Behaviour Type

Gamble: I have just given you $\$50$ up front, you now have a choice.

1. Keep it
2. I flip a coin. Heads: earn an extra $\$25$. Tails: I take $\$25$.

Expected Utility: $\$50$, Choice 1 = Choice 2

Defining Risk Behaviour Type

Gamble: I have just given you $\$50$ up front, you now have a choice.

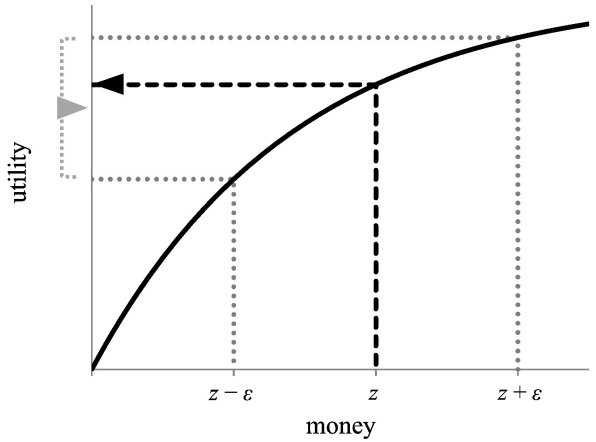
1. Keep it
2. I flip a coin. Heads: earn an extra $\$25$. Tails: I take $\$25$.

Expected Utility: $\$50$, Choice 1 = Choice 2

Assumes Linear utility function!

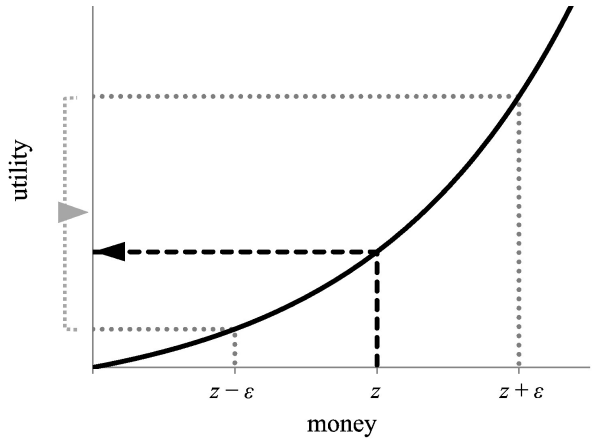
Defining Risk Behaviour Type (2)[1]

Risk-Sensitive: Choice 1



$$\mathbb{E}_z u(z) > \mathbb{E}_{z\pm\epsilon} u(z)$$

Risk-Seeking: Choice 2



$$\mathbb{E}_z u(z) < \mathbb{E}_{z\pm\epsilon} u(z)$$

$$\mathbb{E}_{z\pm\epsilon} u(z) = \frac{1}{2} u(z-\epsilon) + \frac{1}{2} u(z+\epsilon)$$

Figure Credit: *Houston et al.[4]*

Utility Function (1)

For **Gains** we are **Risk-Sensitive**:

- We choose guaranteed/highest probable gains, even if we can win more through gambling.
- We are put off by prizes with a larger variance between them.
- **Why?** If we have nothing we will take anything that guarantees a gain.

Utility Function (2)

For losses we are **Risk-Seeking**. Our utility function is convex

- We choose the option which incurs the lowest loss, even if there exists a guaranteed loss of a lower quantity.
- E.g. If we inverse the first bet; pay $\$50$ or, 50/50 chance of losing $\$25$ or $\$50$. We pick the option with greater uncertainty, but smaller losses.

Combining these facts provides us with an S-curve.

Utility Function (3)

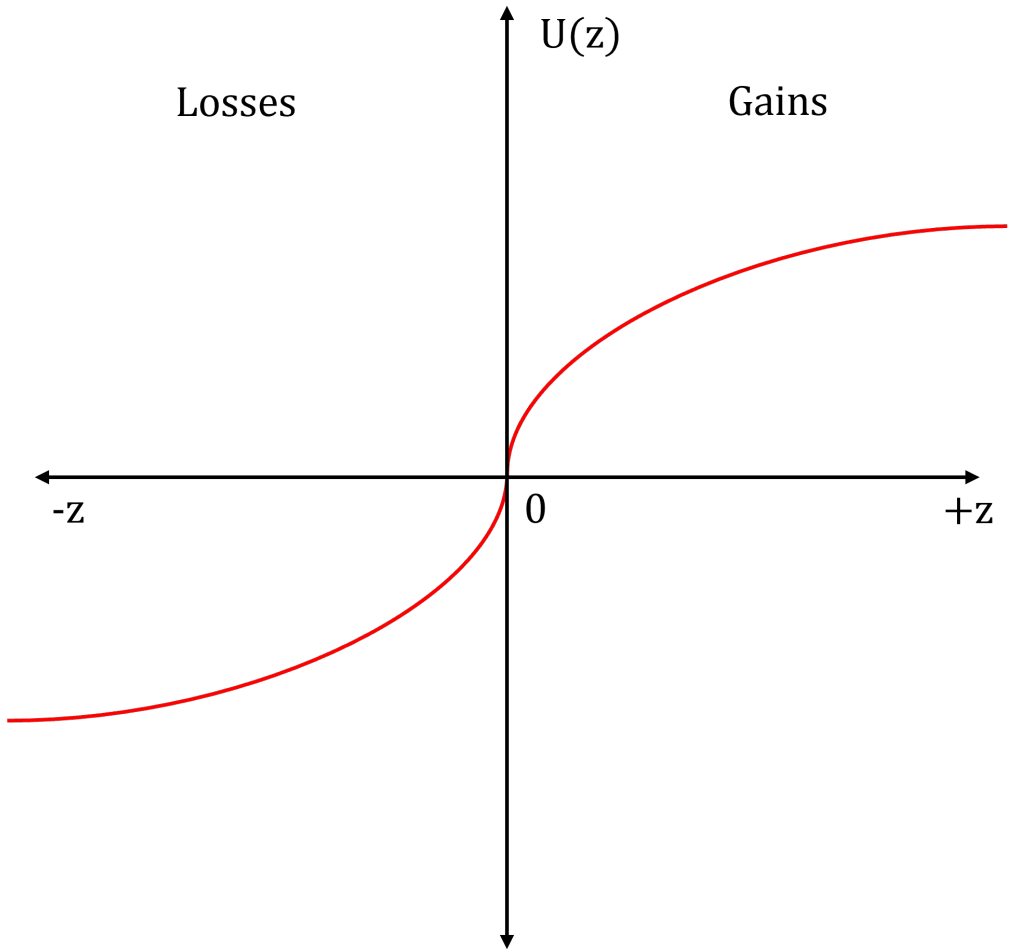


Figure 3: Combined S-curve utility function, encapsulating both attitudes to risk

Modifications

- The curves gradient is proportional to an individual's risk e.g. steep: minute gains and losses impact utility greatly.
- S-curve may be asymmetrical about 0. Difference in gradient between loss and gain axis.
- Utility curve will change based upon the current assets of the individual.

Prospect Theory (Kahneman et al. 1979)[5]: Proposes alternative 'value' function is defined by gains relative to current held assets.

Introduction

Question: How can we best allocate a finite amount of resources, with eye to maximising some reward which is also unknown? How is this done none wastefully?

Applications

- **Clinical Trials** Allocating finite samples to promising drug treatments[2]
- **Portfolio Maximisation** Finding then investing in the most promising stocks.

2-Armed Bandit Problem

Problem: There are 2 slot machines with fixed winning probabilities $p_1 = 0.5$ and $p_2 = 0.55$.

The win/loss for each machine, after one lever-pull k:

$$X_k \backsim \text{Bernoulli}(p_k)$$

Where $p_k \in \{p_1, p_2\}$

2-Armed Bandit Problem

Problem: There are 2 slot machines with fixed winning probabilities $p_1 = 0.5$ and $p_2 = 0.55$.

The win/loss for each machine, after one lever-pull k:

$$X_k \backsim \text{Bernoulli}(p_k)$$

Where $p_k \in \{p_1, p_2\}$

We have T tries on the machines and a single lever-pull per try $t = 0, 1, ..., T$.

But, we don't know p_1 or p_2!

Explore-Exploit Algorithms

Explore all levers and identify the most prosperous bandit, then, **Exploit** for the highest gains.

Explore-Exploit Algorithms

Explore all levers and identify the most prosperous bandit, then, **Exploit** for the highest gains.

Explore-Exploit Methods:

- Upper Confidence Bound-1 (UCB-1)
- Thompson Sampling (TS)

UCB-1

- We have 2 arms: a_1 & a_2
- For each round $t \in 1, ..., T$, we assign it an arm to pull
- For $t = 1$ we allocate a_1 and for $t = 2$ we allocate a_2
- For values of $t \in \{3, ..., T\}$, for each arm k we calculate :

$$\overline{\mu}_{k,t} = \frac{\sum_{s=1}^{t-1} X_{k,s} \mathbb{I}\{a_s = k\}}{\sum_{s=1}^{t-1} \mathbb{I}\{a_s = k\}} + \sqrt{\frac{2log(t)}{\sum_{s=1}^{t-1} \mathbb{I}\{a_s = k\}}}$$

- We assign each $t \in \{3, ..., T\}$ to arm k^*, where

$$k^* = \text{argmax}_k \ \overline{\mu}_{k,t}$$

TS

Thompson Sampling allows us to elucidate bandit probabilities; sampling from assumed posteriors:

$$\hat{p}_{k,t} \backsim \pi(p_k|\mathbf{X}_{k,1:t-1})$$

TS

Thompson Sampling allows us to elucidate bandit probabilities; sampling from assumed posteriors:

$$\hat{p}_{k,t} \backsim \pi(p_k|\mathbf{X}_{k,1:t-1})$$

On each trial t $\hat{p}_{k,t}$ is calculated for both levers k.

$$k^* = \text{argmax}_k \ \hat{\mathbf{p}}_{k,t}$$

k^*'s lever is pulled and its posterior is updated from machine outcome.

TS: Posterior

Conjugate prior for Bernoulli is Beta so

$$\pi(\hat{p}_k) \backsim \text{Beta}(\alpha_k, \beta_k)$$

Therefore $\pi(p_k|\mathbf{X}_{k,1:t-1}) \backsim \text{Beta}(a_0 k + w_{k,1:t-1}, B_0 k + l_{k,1:t-1})$.

TS: Posterior

Conjugate prior for Bernoulli is Beta so

$$\pi(\hat{p}_k) \backsim \text{Beta}(\alpha_k, \beta_k)$$

Therefore $\pi(p_k|\mathbf{X}_{k,1:t-1}) \backsim \text{Beta}(a_0 k + w_{k,1:t-1}, B_0 k + l_{k,1:t-1})$.

Hyperparameters α_k and β_k are defined by initial quantities $a_0 k$ and $B_0 k$.

$w_{k,1:t-1}$ and $l_{k,1:t-1}$ are the cumulative wins and losses for each lever k at a particular trial t.

Experiment

Goal: Compare the effectiveness of each method based upon how they handle regret.

$$\text{Regret} = \sum_{k:\Delta_k<0} \Delta_k \mathbb{E}\left(\sum_{t=1}^{T} \mathbb{I}\{k_t^* = k\}\right)$$

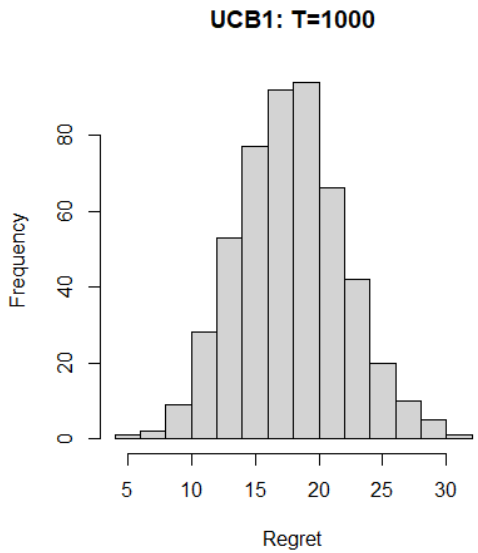
Where $\Delta_k = p^* - p_k$, is the probability difference between optimal and given lever k.

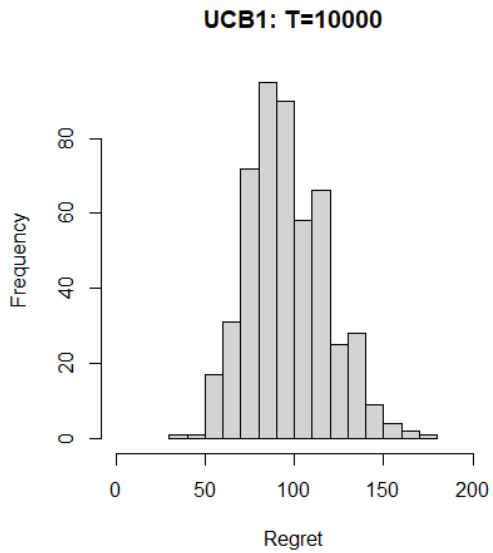
Experiment 2

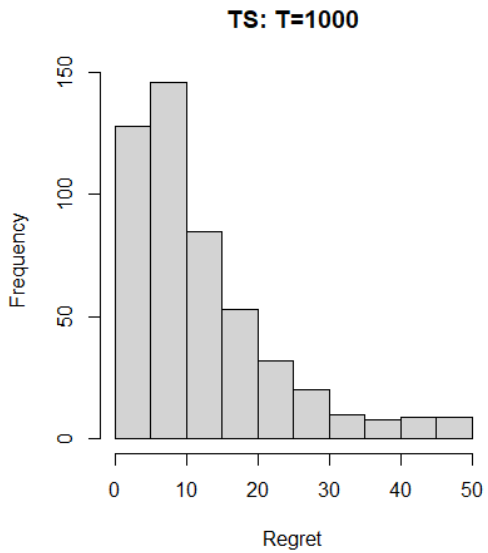
In all experiments the number of repeat samples $N = 500$. TS initial hyperparameters $a_0 1 = a_0 2 = B_0 1 = B_0 2 = 2$ [3]:

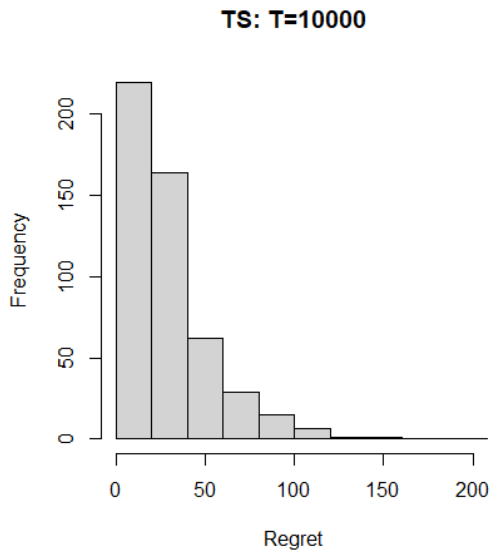
1. Varying $T \in \{10^3, 10^4\}$
2. Setting $p_1 = 0.05$ and $p_2 = 0.95$ (keeping $T = 10^3$)
3. (TS Only) varying posterior hyperparameters

Results: T Length



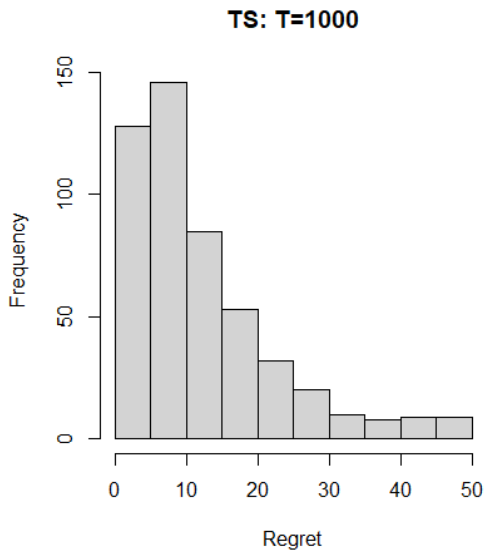


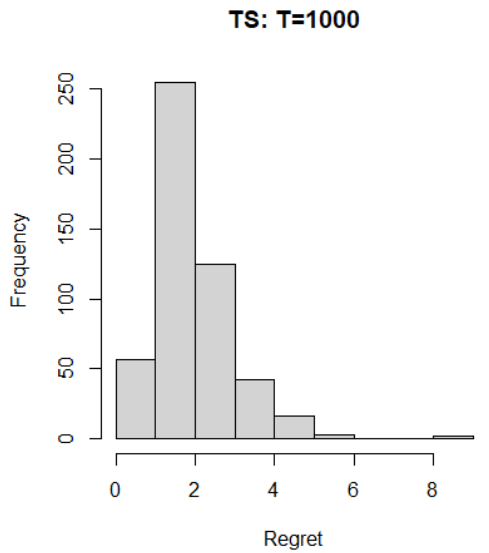


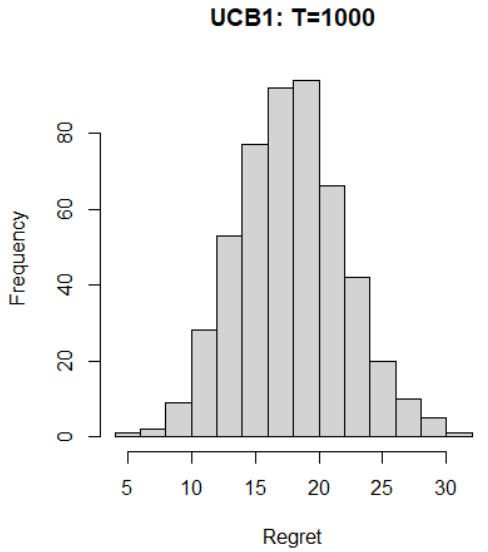


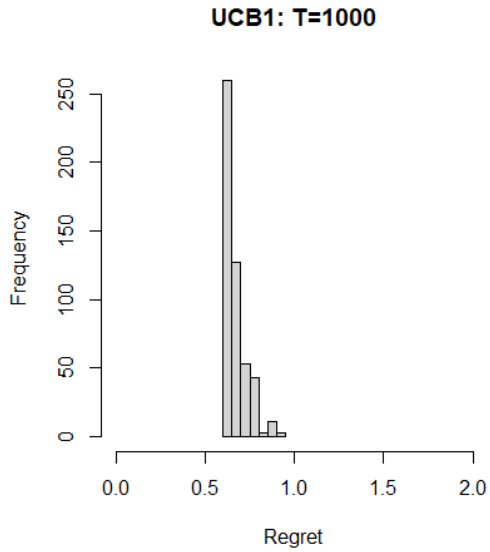
Histogram of the regrets of UCB1 and TS for varied T lengths.

Results: Varying p_1 & p_2









Histogram of the regrets of UCB-1 and TS.

With p_1=0.5 & $p_2 = 0.55$ (left) and p_1=0.05 & $p_2 = 0.95$ (right)

Results: Changing hyperparameters in TS

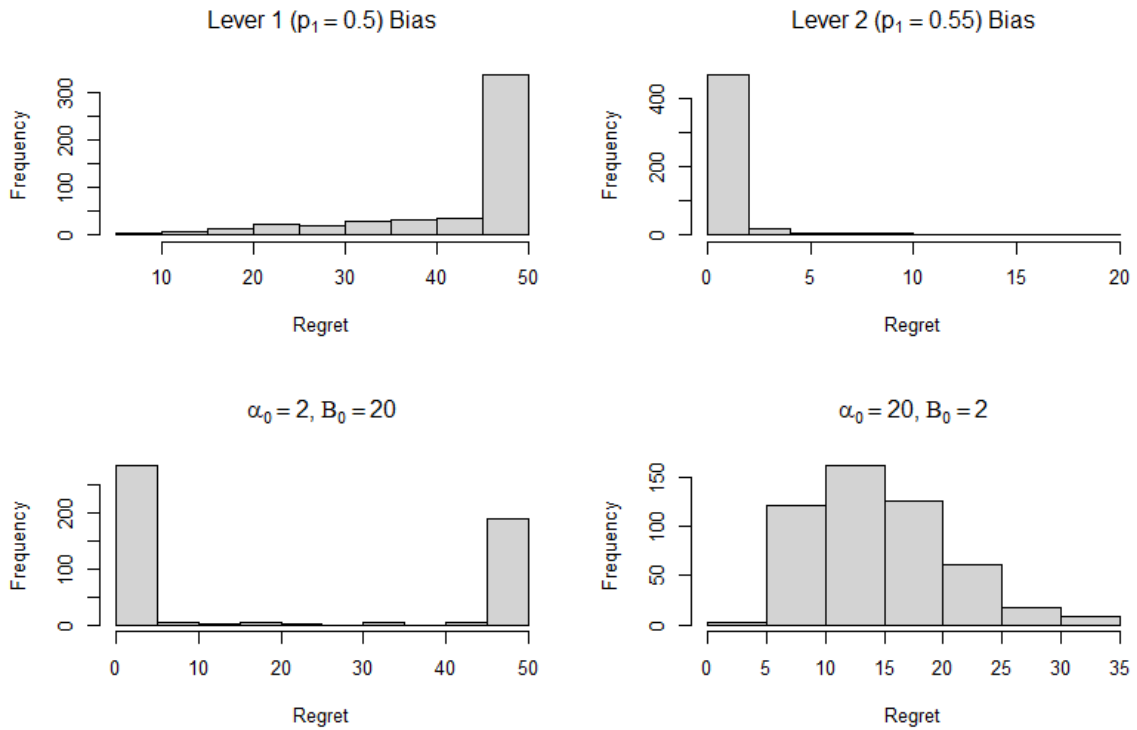


Figure 4: Histograms of TS for different initial hyperparameters

Further Research

- Investigate method responses to $K >> 2$ algorithms
- Bandits which produce different reward amounts as
- Adversarial bandits; win/loss probabilities change to counter algorithm.
- Measure **Regret per turn** to measure the rate at which the methods converge on the best levers.

Reference

[1] K. G. Binmore.
Rational decisions.
Gorman lectures in economics. Princeton University Press, Princeton, N.J., course book. edition, 2009.

[2] Djallel Bouneffouf and Irina Rish.
A Survey on Practical Applications of Multi-Armed and Contextual Bandits, 2019.

[3] Neha Gupta, Ole-Christoffer Granmo, and Ashok Agrawala.
Thompson Sampling for Dynamic Multi-armed Bandits.
In *2011 10th International Conference on Machine Learning and Applications and Workshops*, volume 1, pages 484–489, 2011.

[4] Alasdair I. Houston, Tim W. Fawcett, Dave E.W. Mallpress, and John M. McNamara.
Clarifying the relationship between prospect theory and risk-sensitive foraging theory.

Bayesian Optimisation Introduction

- We want to find the maximizer or the minimizer of an unknown function $f(x)$.
- It is usually used to optimize functions that are expensive to evaluate.
- Applications of Bayesian Optimization include optimizing control parameters in robotics and evaluating the performance of wind turbines

Bayesian Optimisation Method

Algorithm 1 Bayesian Optimisation Algorithm

Require: Domain $\mathcal{X}$, Objective function f, prior distribution π_0,
  Empty data set $\mathcal{D}_0$
  for t=1,2,... **do**
    Select $\mathbf{x}_t = \arg\max_{\mathbf{x}\in\mathcal{X}} \alpha_n(\mathbf{x}|\mathcal{D}_{t-1})$
    Observe $\mathbf{y}_t = f(\mathbf{x}) + \epsilon_t$
    $\mathcal{D}_t = \mathcal{D}_{t-1} \cup \{(\mathbf{x}_t, \mathbf{y}_t)\}$
    Perform Bayesian update to obtain π_t
  end for
  return x^*

Acquisition Function

Definition: An **acquisition function** is a function that helps you determine which data areas you should exploit, and which areas of data you should explore to help you search for the global optimum solution.
The acquisition function estimates the benefit that is offered by an evaluation with respect to solving $x^* = \arg\max_{x \in \mathcal{X}} f(x)$

Set up

$f(x) \sim N(\mu_n(x), \sigma_n^2(x))$

$f_n^* = \min\{f(x_1), f(x_2), ..., f(x_n)\}$

$\mathcal{D}_n = \{(x_1, f(x_1)), (x_2, f(x_2)), ..., (x_n, f(x_n))\}$

Utility function: $u(x) = \max\{0, f_n^* - f(x)\}$

$$u(x) = \begin{cases} 0 & \text{if } f(x) > f_n^* \\ f_n^* - f(x) & \text{if } f(x) \leq f_n^* \end{cases}$$

Closed Form Expression i

Expected Improvement (EI) acquisition function $\alpha_n(x)$:
The expected size of improvement over the current optimum f_n^*

$$
\begin{aligned}
\alpha_n(x) &= \mathbb{E}[u(x)|\mathcal{D}_n] \\
&= \mathbb{P}[f(x) > f_n^*] \cdot \mathbb{E}[0|\mathcal{D}_n, f(x) > f_n^*] \\
&+ \mathbb{P}[f(x) \leq f_n^*] \cdot \mathbb{E}[f_n^* - f(x)|\mathcal{D}_n, f(x) \leq f_n^*] \\
&= \mathbb{P}[f(x) \leq f_n^*] \cdot \mathbb{E}[f_n^* - f(x)|\mathcal{D}_n, f(x) \leq f_n^*] \\
&= \Phi\left(\frac{f_n^* - \mu_n(x)}{\sigma_n(x)}\right) \cdot \mathbb{E}[f_n^* - f(x)|\mathcal{D}_n, f(x) \leq f_n^*] \\
&= \Phi(g_n^*) \cdot \mathbb{E}[f_n^* - f(x)|\mathcal{D}_n, f(x) \leq f_n^*]
\end{aligned}
$$

Closed Form Expression ii

$$\mathbb{E}[f_n^* - f(x) | \mathcal{D}_n, f(x) \leq f_n^*]$$
$$= \mathbb{E}[f_n^* | \mathcal{D}_n, f(x) \leq f_n^*] + \mathbb{E}[-f(x) | \mathcal{D}_n, f(x) \leq f_n^*]$$
$$= f_n^* - \mathbb{E}[f(x) | D_n, f(x) \leq f_n^*]$$

$$Z \sim N(0,1)$$
$$\mathrm{f(x)} \sim N(\mu_n(x), \sigma_n^2(x))$$
$$\mathrm{f(x)} = \mu_n(x) + \sigma_n(x) Z$$

$$f(x) \leq f_n^*$$
$$\implies \frac{f(x) - \mu_n(x)}{\sigma_n(x)} \leq \frac{f_n^* - \mu_n(x)}{\sigma_n(x)}$$
$$\implies Z \leq g_n^*$$

Closed Form Expression iii

$$
\begin{aligned}
&\mathbb{E}[f(x)|\mathcal{D}_n, f(x) \leq f_n^*] \\
&\quad = \mathbb{E}[\mu_n(x) + \sigma_n(x)Z|\mathcal{D}_n, Z \leq g_n^*] \\
&\quad = \mathbb{E}[\mu_n(x)|\mathcal{D}_n, Z \leq g_n^*] + \mathbb{E}[\sigma_n(x)Z|\mathcal{D}_n, Z \leq g_n^*] \\
&\quad = \frac{\int_{-\infty}^{g_n^*} \mu_n(x)\phi(z) \cdot dz + \int_{-\infty}^{g_n^*} \sigma_n(x) z\phi(z) \cdot dz}{\mathbb{P}(Z \leq g_n^*)} \\
&\quad = \frac{\mu_n(x) \int_{-\infty}^{g_n^*} \phi(z) \cdot dz + \sigma_n(x) \int_{-\infty}^{g_n^*} z\phi(z) \cdot dz}{\mathbb{P}(Z \leq g_n^*)}
\end{aligned}
$$

Closed Form Expression iv

$$= \frac{\mu_n(x)\int_{-\infty}^{g_n^*}\phi(z)\cdot dz + \sigma_n(x)\int_{-\infty}^{g_n^*} z\phi(z)\cdot dz}{\mathbb{P}(Z \leq g_n^*)}$$

$$= \mu_n(x)\frac{\Phi(g_n^*)}{\Phi(g_n^*)} - \sigma_n(x)\frac{\phi(g_n^*)}{\Phi(g_n^*)}$$

$$= \mu_n(x) - \sigma_n(x)\frac{\phi(g_n^*)}{\Phi(g_n^*)}$$

Closed Form Expression v

Combining these equations:

$$\alpha_n(x) = \Phi(g_n^*) \cdot \mathbb{E}[f_n^* - f(x) | \mathcal{D}_n, f(x) \leq f_n^*]$$
$$\mathbb{E}[f_n^* - f(x) | \mathcal{D}_n, f(x) \leq f_n^*] = f_n^* - \mathbb{E}[f(x) | \mathcal{D}_n, f(x) \leq f_n^*]$$
$$\mathbb{E}[f(x) | \mathcal{D}_n, f(x) \leq f_n^*] = \mu_n(x) - \sigma_n(x) \frac{\phi(g_n^*)}{\Phi(g_n^*)}$$

We find the closed form expression:

$$\begin{aligned} \alpha_n(x) &= \Phi(g_n^*) \cdot \left(f_n^* - \mu_n(x) + \sigma_n(x) \frac{\phi(g_n^*)}{\Phi(g_n^*)} \right) \\ &= (f_n^* - \mu_n(x)) \Phi(g_n^*) + \sigma_n(x) \phi(g_n^*) \end{aligned}$$

Interpreting $\alpha_n(x)$

$$\alpha_n(x) = \underbrace{(f_n^* - \mu_n(x))\Phi(g_n^*)}_{\text{Exploitation}} + \underbrace{\sigma_n(x)\phi(g_n^*)}_{\text{Exploration}}$$

- The first term increases when the query points provide a low mean, thus encouraging exploitation (evaluating at points with low mean)
- The second term increases for when the query points provide a high variance, encouraging exploration. (evaluating at points with high uncertainty)

Exploitation and Exploration Graphs

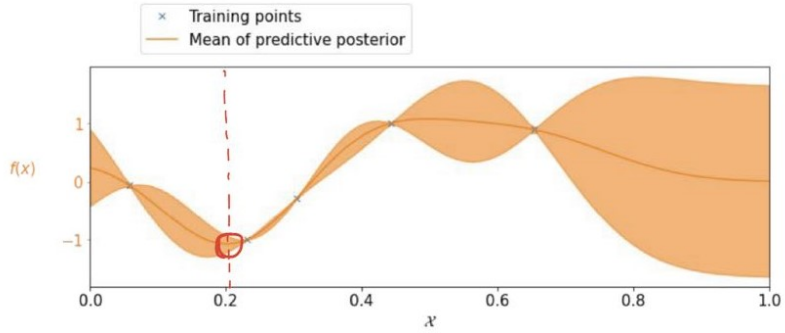


Figure 1: Exploitation (Sampling $\mathbf{x}$ with small $\mu^2_{t-1}(\mathbf{x})$)

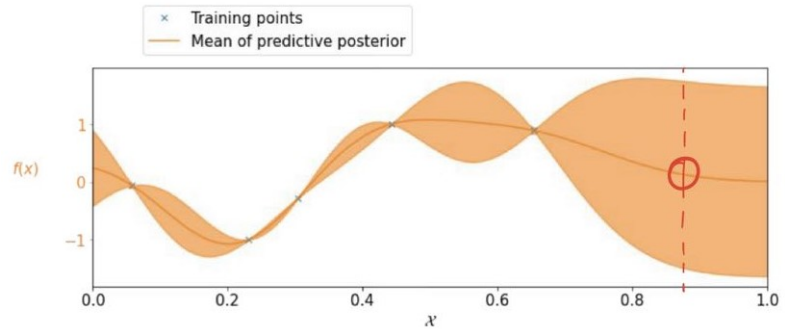


Figure 2: Exploration (Sampling $\mathbf{x}$ with large $\sigma^2_{t-1}(\mathbf{x})$)

[3]

EI vs GP-UCB

GP-UCB:

$$\alpha_n(x;\beta) = -\mu_n(x) + \beta\sigma_n(x)$$

- The difference is that with EI the acquisition is derived from the utility function.
- The advantage of using EI is that we do not have to choose the value of tuning parameter.
- Drawbacks: over-exploitation and under-exploration [2]

Discrete vs Continuous

Multi-armed Bandits problem (Discrete):

- Given a finite number of actions, maximises expected cumulative reward over T rounds.
- Strategies: UCB, Thompson Sampling, ...

Bayesian Optimization (Continuous):

- Given a black-box function, find the global maximizer or minimizer.
- Regret-based strategies: GP-UCB, Thompson Sampling, ...
- Other strategies: Expected Improvement(EI), ...

Regret-based strategies i

GP-UCB:

- Minimize the cumulative regret, $R_T = \sum_{t=1}^{T} r_t$, $r_t = f(\mathbf{x}_t) - f(\mathbf{x}^*)$
- A regret bound of $\mathcal{O}(\sqrt{dT\gamma})$ with high probability [4]

Thompson:

- $P(\| x^t - x^* \| > \epsilon) \leq C \frac{t^{d/2}}{\delta_\epsilon^d} \exp(-c\delta_\epsilon^2 t)$ where C, c are positive constants and [1]

Both are provably convergent.

Regret-based strategies ii

- Regret-based strategies focus on minimizing cumulative regret, which is not entirely equivalent to finding the minimizer or maximizer.
- GP-UCB performs similarly as to EI.[4]

Further Research and Conclusion

- Simulation study comparing Gp-UCB and Thompson Sampling to EI.
- We have explored the breadth of the literature around decision making under sincerity.
- It has opened us up to many questions, which we will be interesting research further.

Reference

[1] Kinjal Basu and Souvik Ghosh.
Adaptive Rate of Convergence of Thompson Sampling for Gaussian Process Optimization, 2020.

[2] Adam D. Bull.
Convergence rates of efficient global optimization algorithms, 2011.

[3] Henry Moss.
Bayesian Optimization.

[4] Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger.
Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting.
IEEE Transactions on Information Theory, 58(5):3250–3265, May 2012.