Decisions and randomness

Christos Dimitrakakis

April 3, 2025

Outline

Preferences

Types of rewards

- For e.g. a student: Tickets to concerts.
- For e.g. an investor: A basket of stocks, bonds and currency.
- For everybody: Money.

Preferences among rewards

For any rewards $x, y \in R$, we either

- (a) Prefer x at least as much as y and write $x \preceq^* y$.
- (b) Prefer x not more than y and write $x \succeq^* y$.
- (c) Prefer x about the same as y and write $x \eqsim^* y$.
- (d) Similarly define $\succ^*$ and $\prec^*$

Utility and Cost

Utility and Cost

Utility function

To make it easy, assign a utility $U(x)$ to every reward through a utility function $U : R \to \mathbb{R}$.

Utility-derived preferences

We prefer items with higher utility, i.e.

- (a) $U(x) \geq U(y) \Leftrightarrow x \succeq^* y$
- (b) $U(x) \leq U(y) \Leftrightarrow y \succeq^* x$

Utility and Cost

Utility function

To make it easy, assign a utility $U(x)$ to every reward through a utility function $U : R \to \mathbb{R}$.

Utility-derived preferences

We prefer items with higher utility, i.e.

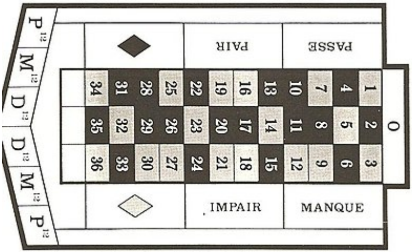
- (a) $U(x) \geq U(y) \Leftrightarrow x \succeq^* y$
- (b) $U(x) \leq U(y) \Leftrightarrow y \succeq^* x$

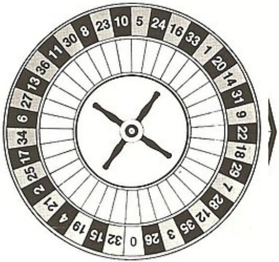
Cost

It is sometimes more convenient to define a cost function $C : R \to \mathbb{R}$ so that we prefer items with lower cost, i.e.

- $C(x) \geq C(y) \Leftrightarrow y \succeq^* x$

Random outcomes





Choosing among rewards: Roulette

- [A] Bet 10 CHF on black
- [B] Bet 10 CHF on 0
- [C] Bet nothing
- What is the reward here?
- What is the outcome?

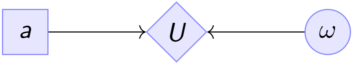
Uncertain outcomes

- [A] Taking the car to Zurich (50'-80' with delays)
- [B] Taking the train to Zurich (60' without delays)

What is the reward here?

Independent outcomes

Graphical model



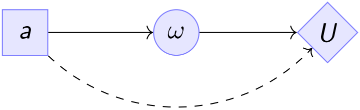
Random rewards

- We select our action.
- Outcomes are random, with $\omega \sim P$, but independent of our action
- We then obtain a random utility with distribution depending on a.

$$\mathbb{P}(U = u \mid a) = P(\{\omega : U(a, \omega) = u\})$$

General case

Graphical model



Random rewards

- We select our action.
- The action determines the outcome distribution.
- The utility may depend on both the outcome and reward.

Route selection

Example (Utility)

$U(a, \omega)$	30'	40'	50'	60'	70'	80'	90'
Train	-1	-2	-5	-10	-15	-20	-30
Car	-10	-20	-30	-40	-50	-60	-70

Example (Probability)

$P(\omega \mid a)$	30'	40'	50'	60'	70'	80'	90'
Train	0%	0%	50%	45%	4%	1%	0%
Car	0	40%	30%	15%	10%	3%	2%

Expected utility

Actions, outcomes and utility

In this setting, we obtain random outcomes that depend on our actions.

- Actions $a \in A$
- Outcomes $\omega \in \Omega$.
- Probability of outcomes $P(\omega \mid a)$
- Utility $U : \Omega \times A \to \mathbb{R}$

Expected utility

The expected utility of an action is:

$$\mathbb{E}_P[U \mid a] = \sum_{\omega \in \Omega} U(\omega, a) P(\omega \mid a).$$

The expected utility hypothesis

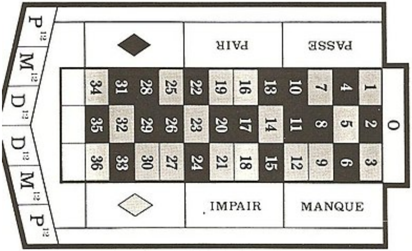
We prefer a to a' if and only if

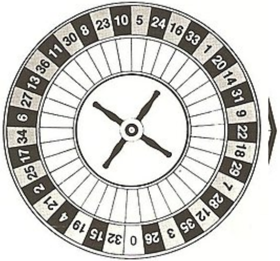
$$\mathbb{E}_P[U \mid a] \geq \mathbb{E}_P[U \mid a']$$

Example: Betting

In this example, probabilities reflect actual randomness

Choice	Win Probability p	Payout w	Expected gain
Don't play	0	0	0
Black	18/37	2	
Red	18/37	2	
0	1/37	36	
1	1/37	36	





What are the expected gains for these bets?

The St-Petersburg Paradox

The game

If you give me x CHF, then I promise to:

- (a) Throw a fair coin until it comes heads.
- (b) If it does so after T throws, then I will give you 2^T CHF.

The question

- How much x are you willing to pay to play?
- Given that the expected amount of money is infinite, why are you only willing to pay a small x?

Example: Route selection

- In this example, probabilities reflect subjective beliefs

Choice	Best time	Chance of delay	Delay amount	Expected time
Train	80	5%	5	
Car, route A	60	50%	30	
Car, route B	70	10%	10	

Example: Noisy optimisation

Simple maximisation

For a function $f : \mathbb{R} \to \mathbb{R}$, find a maximum x^* i.e. $f(x^*) \geq f(x) \forall x$.

Example: Noisy optimisation

Simple maximisation

For a function $f : \mathbb{R} \to \mathbb{R}$, find a maximum x^* i.e. $f(x^*) \geq f(x) \forall x$.

Theorem (Necessary conditions)

If $f : \mathbb{R} \to \mathbb{R}$ is a continuous function, a maximum point x^* satisfies:

$$\frac{d}{dx} f(x^*) = 0, \qquad \frac{d}{dx^2} f(x^*) < 0.$$

Example: Noisy optimisation

Simple maximisation

For a function $f : \mathbb{R} \to \mathbb{R}$, find a maximum x^* i.e. $f(x^*) \geq f(x) \forall x$.

Theorem (Necessary conditions)

If $f : \mathbb{R} \to \mathbb{R}$ is a continuous function, a maximum point x^* satisfies:

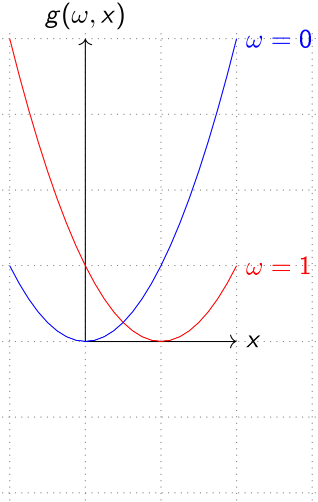
$$\frac{d}{dx} f(x^*) = 0, \qquad \frac{d}{dx^2} f(x^*) < 0.$$

Noisy optimisation

- We select x but do not observe $f(x)$.
- We observe a random g with $\mathbb{E}[g|x] = f(x)$.

$$f(x) \triangleq \mathbb{E}[g|x], \qquad \mathbb{E}[g|x] = \int_{-\infty}^{\infty} g(\omega, x) p(\omega) d\omega \tag{1}$$

Mean-squared error cost function



This example is for a quadratic loss:

$g(\omega, x) = (\omega - x)^2.$

Example: Estimation

- θ: parameter (random)
- $\hat{\theta}$: estimate (our action)
- $(\theta - \hat{\theta})^2$: cost function

Example: Estimation

- θ: parameter (random)
- $\hat{\theta}$: estimate (our action)
- $(\theta - \hat{\theta})^2$: cost function

Mean-squared error minimiser

If we want to guess $\hat{\theta}$, and we knew that $\theta \sim P$, then the guess

$$\hat{\theta} = \mathbb{E}_P(\theta) = \arg\min_{\hat{\theta}} \mathbb{E}_P[(\theta - \hat{\theta})^2]$$

minimises the squared error.

Example: Estimation

- θ: parameter (random)
- $\hat{\theta}$: estimate (our action)
- $(\theta - \hat{\theta})^2$: cost function

Mean-squared error minimiser

If we want to guess $\hat{\theta}$, and we knew that $\theta \sim P$, then the guess

$$\hat{\theta} = \mathbb{E}_P(\theta) = \arg\min_{\hat{\theta}} \mathbb{E}_P[(\theta - \hat{\theta})^2]$$

minimises the squared error. This is because

$$\frac{d}{d\hat{\theta}} \mathbb{E}_P[(\theta - \hat{\theta})^2] = \frac{d}{d\hat{\theta}} \sum_\omega [\theta(\omega) - \hat{\theta}]^2 P(\omega) \tag{2}$$

$$= \sum_\omega \frac{d}{d\hat{\theta}} [\theta(\omega) - \hat{\theta}]^2 P(\omega) \tag{3}$$

$$= \sum_\omega 2[\theta(\omega) - \hat{\theta}](-1) P(\omega) \qquad = 2(\hat{\theta} - \mathbb{E}_P[\theta]). \tag{4}$$

Setting this to 0 gives $\hat{\theta} = \mathbb{E}_P[\theta]$

The gradient descent method: one dimension

- Function to minimise $f : \mathbb{R} \to \mathbb{R}$.
- Derivative $\frac{d}{d\theta} f(\theta)$

The gradient descent method: one dimension

- Function to minimise $f : \mathbb{R} \to \mathbb{R}$.
- Derivative $\frac{d}{d\theta} f(\theta)$

Gradient descent algorithm

- Input: initial value θ^0, learning rate schedule α_t
- For $t = 1, \ldots, T$
  - $\theta^{t+1} = \theta^t - \alpha_t \frac{d}{d\theta} f(\theta^t)$
- Return θ^T

The gradient descent method: one dimension

- Function to minimise $f : \mathbb{R} \to \mathbb{R}$.
- Derivative $\frac{d}{d\theta} f(\theta)$

Gradient descent algorithm

- Input: initial value θ^0, learning rate schedule α_t
- For $t = 1, \ldots, T$
  - $\theta^{t+1} = \theta^t - \alpha_t \frac{d}{d\theta} f(\theta^t)$
- Return θ^T

Properties

- If $\sum_t \alpha_t = \infty$ and $\sum_t \alpha_t^2 < \infty$, it finds a local minimum θ^T, i.e. there is $\epsilon > 0$ so that

$$f(\theta^T) < f(\theta), \forall \theta : \|\theta^T - \theta\| < \epsilon.$$

Gradient methods for expected value

Estimate the expected value

$x_t \sim P$ with $\mathbb{E}_P[x_t] = \mu$.

Gradient methods for expected value

Estimate the expected value

$x_t \sim P$ with $\mathbb{E}_P[x_t] = \mu$.

Objective: mean squared error

Here $\ell(x, \theta) = (x - \theta)^2$.

$$\min_\theta \mathbb{E}_P[(x_t - \theta)^2].$$

Gradient methods for expected value

Estimate the expected value

$x_t \sim P$ with $\mathbb{E}_P[x_t] = \mu$.

Objective: mean squared error

Here $\ell(x, \theta) = (x - \theta)^2$.

$$\min_\theta \mathbb{E}_P[(x_t - \theta)^2].$$

Exact derivative update

If we know P, then we can calculate

$$\theta^{t+1} = \theta^t - \alpha_t \frac{d}{d\theta} \mathbb{E}_P[(x - \theta^t)^2] \tag{5}$$

$$\frac{d}{d\theta} \mathbb{E}_P[(x - \theta^t)^2] = 2(\mathbb{E}_P[x] - \theta^t) \tag{6}$$

Stochastic derivative

- Function to minimise $f : \mathbb{R} \to \mathbb{R}$.
- Derivative $\frac{d}{d\theta} f(\theta)$
- $f(\theta) = \mathbb{E}[g|\theta]$
- $\frac{d}{d\theta} f = \mathbb{E}[\frac{d}{d\theta} g|\theta]$

Stochastic derivative

- Function to minimise $f : \mathbb{R} \to \mathbb{R}$.
- Derivative $\frac{d}{d\theta} f(\theta)$
- $f(\theta) = \mathbb{E}[g|\theta]$
- $\frac{d}{d\theta} f = \mathbb{E}[\frac{d}{d\theta} g|\theta]$

Stochastic derivative algorithm

- Input: initial value θ^0, learning rate schedule α_t
- For $t = 1, \ldots, T$
  - Observe $g(\omega_t, \theta^t)$, where $\omega_t \sim P$.
  - $\theta^{t+1} = \theta^t - \alpha_t \frac{d}{d\theta} g(\omega_t, \theta^t)$
- Return θ^T

Stochastic gradient for mean estimation

Theorem (Sampling)

For any bounded random variable f,

$$\mathbb{E}_P[f] = \int_X dP(x)f(x) = \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} f(x_t) = \mathbb{E}_P\left[\frac{1}{T}\sum_{t=1}^{T} f(x_t)\right], \qquad x_t \sim P$$

Example (Derivative ampling)

We can also approximate the gradient through sampling:

$$\begin{aligned}\frac{d}{d\theta}\mathbb{E}_P[(x-\theta)^2] &= \int_{-\infty}^{\infty} dP(x)\frac{d}{d\theta}(x-\theta)^2 \\ &\approx \frac{1}{T}\sum_{t=1}^{T}\frac{d}{d\theta}(x_t-\theta)^2 = \frac{1}{T}\sum_{t=1}^{T} 2(x_t-\theta)\end{aligned}$$

Stochastic gradient for mean estimation

Theorem (Sampling)

For any bounded random variable f,

$$\mathbb{E}_P[f] = \int_X dP(x) f(x) = \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} f(x_t) = \mathbb{E}_P\left[\frac{1}{T}\sum_{t=1}^{T} f(x_t)\right], \qquad x_t \sim P$$

Example (Derivative ampling)

We can also approximate the gradient through sampling:

$$\frac{d}{d\theta}\mathbb{E}_P[(x-\theta)^2] = \int_{-\infty}^{\infty} dP(x)\frac{d}{d\theta}(x-\theta)^2$$
$$\approx \frac{1}{T}\sum_{t=1}^{T}\frac{d}{d\theta}(x_t-\theta)^2 = \frac{1}{T}\sum_{t=1}^{T} 2(x_t-\theta)$$

- Wen can even update θ after each sample x_t:
  $$\theta^{t+1} = \theta^t + 2\alpha_t(x_t - \theta^t)$$

The gradient method

- Function to minimise $f : \mathbb{R}^n \to \mathbb{R}$.
- Gradient $\nabla_\theta f(\theta) = \left(\frac{\partial f(\theta)}{\partial \theta_1}, \dots, \frac{\partial f(\theta)}{\partial \theta_n}\right)$,
- Partial derivative $\frac{\partial f}{\partial \theta_n}$

The gradient method

- Function to minimise $f : \mathbb{R}^n \to \mathbb{R}$.
- Gradient $\nabla_\theta f(\theta) = \left(\frac{\partial f(\theta)}{\partial \theta_1}, \ldots, \frac{\partial f(\theta)}{\partial \theta_n} \right)$,
- Partial derivative $\frac{\partial f}{\partial \theta_n}$

Gradient descent algorithm

- Input: initial value θ^0, learning rate schedule α_t
- For $t = 1, \ldots, T$
  - $\theta^{t+1} = \theta^t - \alpha_t \nabla_\theta f(\theta^t)$
- Return θ^T

The gradient method

- Function to minimise $f : \mathbb{R}^n \to \mathbb{R}$.
- Gradient $\nabla_\theta f(\theta) = \left(\frac{\partial f(\theta)}{\partial \theta_1}, \ldots, \frac{\partial f(\theta)}{\partial \theta_n}\right)$,
- Partial derivative $\frac{\partial f}{\partial \theta_n}$

Gradient descent algorithm

- Input: initial value θ^0, learning rate schedule α_t
- For $t = 1, \ldots, T$
  - $\theta^{t+1} = \theta^t - \alpha_t \nabla_\theta f(\theta^t)$
- Return θ^T

Properties

- If $\sum_t \alpha_t = \infty$ and $\sum_t \alpha_t^2 < \infty$, it finds a local minimum θ^T, i.e. there is $\epsilon > 0$ so that

$$f(\theta^T) < f(\theta), \forall \theta : \|\theta^T - \theta\| < \epsilon.$$

When the cost is an expectation

In machine learning, we sometimes want to minimise the expectation of a cost ℓ,

$$f(\theta) \triangleq \mathbb{E}[\ell|\theta] = \int_{\Omega} dP(\omega)\ell(\omega, \theta)$$

This can be approwith a sample

$$f(\theta) \approx \frac{1}{T}\sum_{t}\ell(\omega_t, \theta)$$

The same holds for the gradient:

$$\nabla_\theta f(\theta) = \int_d P(\omega)\nabla_\theta \ell(\omega, \theta) \approx \frac{1}{T}\sum_{t}\nabla_\theta \ell(\omega_t, \theta)$$

Stochastic gradient method

- Function to minimise $f : \mathbb{R}^n \to \mathbb{R}$.
- Gradient $\nabla f(\theta)$
- $f(\theta) = \mathbb{E}[\ell|\theta]$
- $\nabla_\theta f = \mathbb{E}[\nabla_\theta \ell|\theta]$

Stochastic gradient method

- Function to minimise $f : \mathbb{R}^n \to \mathbb{R}$.
- Gradient $\nabla f(\theta)$
- $f(\theta) = \mathbb{E}[\ell|\theta]$
- $\nabla_\theta f = \mathbb{E}[\nabla_\theta \ell|\theta]$

Algorithm

- Input: initial value θ^0, learning rate schedule α_t
- For $t = 1, \ldots, T$
  - Observe $\ell(\omega_t, \theta^t)$, where $\omega_t \sim P$.
  - $\theta^{t+1} = \theta^t - \alpha_t \nabla_\theta g(\omega_t, \theta^t)$
- Return θ^T

Alternative view: Noisy gradients

- $\theta^{t+1} = \theta^t - \alpha_t[\nabla_\theta f(\theta^t) + \epsilon_t]$
- $\mathbb{E}[\epsilon_t] = 0$ is sufficient for convergence.

Stochastic gradient method

- Function to minimise $f : \mathbb{R}^n \to \mathbb{R}$.
- Gradient $\nabla f(\theta)$
- $f(\theta) = \mathbb{E}[\ell|\theta]$
- $\nabla_\theta f = \mathbb{E}[\nabla_\theta \ell|\theta]$

Algorithm

- Input: initial value θ^0, learning rate schedule α_t
- For $t = 1, \ldots, T$
  - Observe $\ell(\omega_t, \theta^t)$, where $\omega_t \sim P$.
  - $\theta^{t+1} = \theta^t - \alpha_t \nabla_\theta g(\omega_t, \theta^t)$
- Return θ^T

Alternative view: Noisy gradients

- $\theta^{t+1} = \theta^t - \alpha_t[\nabla_\theta f(\theta^t) + \epsilon_t]$
- $\mathbb{E}[\epsilon_t] = 0$ is sufficient for convergence.