

Übungsblatt 10 – Approximation Algorithms

Randomisierte Algorithmik

Aufgabe 1 – Die Jensensche Ungleichung

Sei $D \subseteq \mathbb{R}$ ein zusammenhängender Definitionsbereich und $f : D \rightarrow \mathbb{R}$ eine Funktion. Die Funktion f heißt konvex, wenn sie „linksgekrümmt“ ist und konkav wenn sie „rechtsgekrümmt“ ist.¹ Eine Funktion ist konvex genau dann wenn ihre Negation konkav ist. Für eine formale Definition siehe: Wikipedia

- (a) Entscheide (ohne Beweis) für folgende Funktionen, ob diese auf ihrem jeweiligen Definitionsbereich konvex ist, konkav ist, beides ist oder weder noch.

$$f_1(x) = x, \quad f_2(x) = x^2, \quad f_3(x) = x^3, \quad f_4(x) = \log(x), \quad f_5(x) = \log^2(x).$$

- (b) Sei f eine konvexe Funktion mit Definitionsbereich D . Argumentiere geometrisch, dass es für jedes $x_0 \in D$ eine lineare Funktion g gibt, sodass gilt:

- (i) $f(x) \geq g(x)$ für alle $x \in D$
- (ii) $f(x_0) = g(x_0)$.

- (c) Schlussfolgere, dass für jede konvexe Funktion f und für jede Zufallsvariable X mit Werten im Definitionsbereich D von f gilt:

$$\mathbb{E}[f(X)] \geq f(\mathbb{E}[X]).$$

Hinweis: Betrachte $x_0 = \mathbb{E}[X]$ und das zugehörige g aus der vorherigen Teilaufgabe.

- (d) Zeige, dass analog für jede konkave Funktion f mit Definitionsbereich D und für jede Zufallsvariable X mit Werten in D gilt:

$$\mathbb{E}[f(X)] \leq f(\mathbb{E}[X]).$$

Die Ungleichung aus (c) sowie Varianten wie in (d) nennt man Jensensche Ungleichung.

¹Das „linksgekrümmt“ in Anführungszeichen lässt neben Linkskrümmungen (einer zweimal stetig differenzierbaren Funktion) auch Linksknicke und geradlinige Verläufe zu.

Aufgabe 2 – Analyse von Lossy Counting

Erinnerung: Lossy Counting ist ein einfacher Streaming Algorithmus, der die Länge m eines Streams approximativ zählt. Darin kommt ein Parameter $p \in (0, 1]$ vor. Der Algorithmus selbst sowie die Art und Weise, wie er verwendet wird, ist rechts nochmal zu sehen. Beweise:

- (a) $\mathbb{E}[\text{result}] = m$
- (b) $\Pr[|\text{result} - m| \leq \varepsilon m] \geq 1 - 2 \exp(-\varepsilon^2 pm/3)$.
- (c) $\mathbb{E}[\text{space}] \leq \log(1 + mp) + 1$.

Hinweis: Mit `space` bezeichnen wir den maximalen Speicherverbrauch, der für den Zustand Z von `LossyCounting` benötigt wird. Eine Zahl $i \in \mathbb{N}$ lässt sich mit $\lceil \log_2(i + 1) \rceil$ Bits kodieren. Verwende die Jensensche Ungleichung aus Aufgabe 1.

Algorithm `init`:

```
Z ← 0
return Z
```

Algorithm `update(Z, a)`:

```
with probability p do
  Z ← Z + 1
return Z
```

Algorithm `result(Z)`:

```
return Z/p
```

Verwendung:

```
Z ← init()
for i = 1 to m do
  Z ← update(Z, ai)
return result(Z)
```