Mathematatik für Informatiker
Kombinatorik, Stochastik und Statistik

Übungsblatt 8

Tom Paßberg , Iain Dorsch

Aufgabe 1

Die Gewinnwarscheinlichkeit läßt sich durch $(\frac{1}{2})^n$ beschreiben. Der erwartete Gewinn kann durch die Summe der gewichteten Gewinne beschrieben werden. Die Summe ist aufgeteilt in den Teil in dem der Gewinn variabel ist (hat noch nicht 2^{47} erreicht) und den Teil in dem der Gewinn aufgrund der Obergrenze konstant ist.

$$\begin{aligned}
&\sum_{n=1}^{46} \frac{2^n}{2^n} + \sum_{n=47}^{\infty} \frac{2^{47}}{2^n} && (1)\\
=&\sum_{n=1}^{46} 1 + \sum_{n=0}^{\infty} \frac{2^{47}}{2^{n+47}} && (2)\\
=&46 + \sum_{n=0}^{\infty} \frac{1}{2^n} && (3)\\
=&46 + \frac{1}{1-\frac{1}{2}} && (4)\\
=&48 && (5)
\end{aligned}$$

(3) geometrische Reihe $\sum_{k=0}^{\infty} q^n = \dfrac{1}{1-q}$

Der erwartete Gewinn beträgt also 48.

Aufgabe 2

a)

```
D1: 6, D2: 2, Sum: 8
D1: 5, D2: 1, Sum: 6
D1: 5, D2: 3, Sum: 8
Gewonnen: true, Gewinn: 1
D1: 2, D2: 1, Sum: 3
Gewonnen: false, Gewinn: 0
D1: 4, D2: 2, Sum: 6
D1: 3, D2: 4, Sum: 7
Gewonnen: false, Gewinn: -1
D1: 2, D2: 2, Sum: 4
D1: 1, D2: 2, Sum: 3
D1: 5, D2: 1, Sum: 6
D1: 5, D2: 5, Sum: 10
D1: 1, D2: 6, Sum: 7
```

```
Gewonnen: false, Gewinn: -2
D1: 2, D2: 3, Sum: 5
D1: 1, D2: 6, Sum: 7
Gewonnen: false, Gewinn: -3
D1: 2, D2: 2, Sum: 4
D1: 6, D2: 4, Sum: 10
D1: 1, D2: 2, Sum: 3
D1: 2, D2: 5, Sum: 7
Gewonnen: false, Gewinn: -4
D1: 3, D2: 3, Sum: 6
D1: 2, D2: 6, Sum: 8
D1: 3, D2: 2, Sum: 5
D1: 3, D2: 4, Sum: 7
Gewonnen: false, Gewinn: -5
D1: 6, D2: 3, Sum: 9
D1: 3, D2: 4, Sum: 7
Gewonnen: false, Gewinn: -6
D1: 2, D2: 1, Sum: 3
Gewonnen: false, Gewinn: -7
D1: 2, D2: 5, Sum: 7
Gewonnen: true, Gewinn: -6
```

Mittlerer Gewinn: $\frac{-6}{10} = -0.6$

b)

Programm in Rust:

```
fn main() {
    let sample_size = 1000;

    let money: i64 = (0..sample_size)
        .map(|_| match roll_dice(2) {
            7 | 11 => 1,
            2 | 3 | 12 => -1,
            s => loop {
                match roll_dice(2) {
                    7 => return -1,
                    n => if n == s { return 1; }
                }
            }
        })
        .sum();

    println!("Mittlerer Gewinn: {:.2}",money as f64/sample_size as f64)
}

fn roll_dice(number: usize) -> u8 {
    (0..number).map(|_| rand::random::<u8>() % 6 + 1).sum()
}
```

Output:

```
Mittlerer Gewinn: -0.02
```

Aufgabe 3

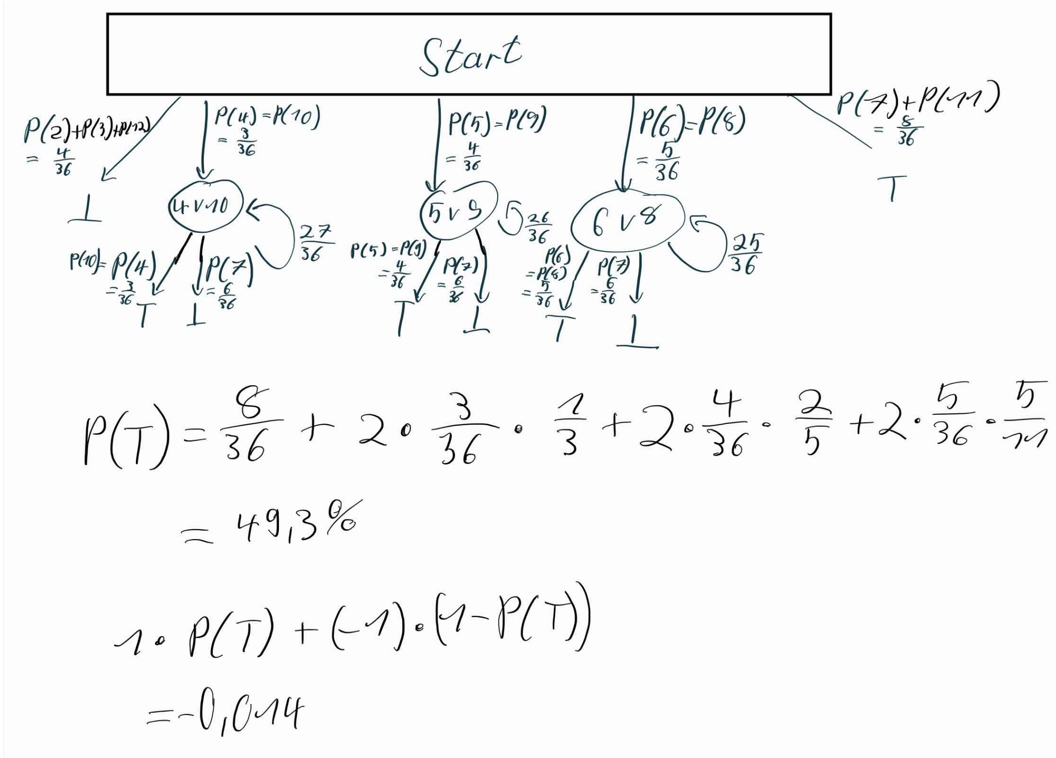

Abbildung 1: Wahrscheinlichkeitsbaum

Aufgabe 4

Sei $x \in M_1 \cup \cdots \cup M_n$ und $m(x) \in P(M_1 \cup \cdots \cup M_n)$.
$m(x)$ soll nur einmal zu $P(M_1 \cup \cdots \cup M_n)$ beitragen
$\implies$ x soll nur einmal aus $M_1 \cup \cdots \cup M_n$ ausgewählt werden.
Liegt x in genau r der Mengen M_i, gilt $x \in M_1 \cap \cdots \cap M_r$
$\implies$ $m(x) \in P(M_1 \cap \cdots \cap M_r)$. Dann wird $m(x)$ in $\sum_{|T|=k} P(M_T)$, mit $M_T = \bigcap_{i \in T} M_i$, genau $\binom{r}{k}$ - mal gezählt/ausgewählt, da man k aus r Mengen mit x wählen kann.
$\implies$ $m(x)$ trägt zu $\sum_{|T|=k} P(M_T)$ genau mit

$$a = \sum_{k=1}^{r} (-1)^{k-1} \binom{r}{k} \tag{6}$$

bei. Mit Corollar 1.2.26 gilt a=1.