

Aufgaben Wahrscheinlichkeitstheorie und Statistik

1 Würfelspiel mit Münzen

Wir betrachten folgendes Spiel: Zuerst werfen wir einen sechseitigen Würfel, nun werfen wir für jedes erwürfelte Auge eine Münze (z.B. wenn wir vier Augen werfen, dürfen wir vier Münzen werfen). Sei nun Y die Anzahl *Köpfe* im zweiten Wurf. Geben Sie die Dichte-Funktion $f_Y(y)$ an, gerne dürfen Sie eine Python Funktion schreiben welche die Wahrscheinlichkeiten für Sie berechnet (Tipp: Klasse `binom` im Paket `scipy.stats`).

2 Bayes Theorem

Wir betrachten einen medizinischen Test um auf eine Krankheit zu testen (z.B. ein Corona-Test). Das Ergebnis bezeichnen wir mit Y , wobei $Y = 1$ ein **positives Testergebnis** (Person ist laut Test krank) und $Y = 0$ ein **negatives Testergebnis** bezeichnet. Mit X bezeichnen wir den Gesundheitszustand der Person, wobei $X = 0$: gesund, $X = 1$ krank. Solche Tests sind immer fehlerbehaftet, in unserem Fall sei $P(Y = 1|X = 1) = 0,9$ (das nennt man die **Sensitivität**) und $P(Y = 0|X = 0) = 0,95$ (**Spezifität**).

1. Nehmen wir an, die a-priori-Wahrscheinlichkeit krank zu sein, sei 3%, berechnen Sie nun die Wahrscheinlichkeit, dass eine positiv getestete Person gesund ist, sowie die Wahrscheinlichkeit das eine negativ getestete Person krank ist.
2. Nehmen wir nun an, die a-priori-Wahrscheinlichkeit krank zu sein, sei 30%, berechnen Sie nun die oben genannten Wahrscheinlichkeiten.

3 Normalverteilung

Im folgende Code-Stück sehen wir ein Beispiel für einen `LinePlot`:

```
1 # array with 100 points equally spaced in [-1,1]
2 x = np.linspace(-1, 1, 100)
3 y = x*x
4 plt.plot(x, y);
```

Das Statistik-Modul von `scipy` enthält die Klasse `norm`. Der folgende Befehl gibt eine standardnormalverteilte Zufallsvariable zurück.

```
1 import scipy.stats as stats
2 rv = stats.norm(1, 2) # Mittelwert und Standardabweichung
3 # Wahrscheinlichkeit am Punkt x=0
4 print('Erwartungswert', rv.mean() )
5 print('Varianz', rv.var() )
6 print('Wahrscheinlichkeiten', rv.pdf(np.array([1,2,3])))
```

Erwartungswert 1.0

Varianz 4.0

Wahrscheinlichkeiten [0.19947114 0.17603266 0.12098536]

Beachten Sie: die Zufallsvariable hat eine Methode `pdf` (probability density function). Diskrete ZV haben stattdessen die Methode `pmf` (probability mass function)

1. Stellen Sie die Dichtefunktion einer Normalverteilung für $\mu = -1$ und $\sigma = 2$ dar. Geben sie auch die Varianz aus.
2. Die Zufallsvariable `rv` besitzt die Methode `rvs`:

```
1 print (rv.rvs(size=5))
```

Erzeugen Sie 1000 Samples aus einer standardnormalverteilten Zufallsvariable und stellen Sie ein **normiertes** Histogramm mit der Funktion `plt.hist()` graphisch dar (lesen Sie die Dokumentation der Methode für die notwendige Option). Fügen Sie die Dichte der Funktion in dasselbe Schaubild hinzu (Sie erreichen dies, indem Sie die `plt.plot()` und die `plt.hist()` Funktion in der selben Zelle ausführen).

4 Multivariate Normalverteilung

Im Folgenden untersuchen wir die multivariate Normalverteilung (genauer 2 D). Implementiert sind sie im Module `scipy.stats` in der Klasse `multivariate_normal`. Machen Sie sich zuerst mit der Klasse vertraut (DocString!).

1. Erzeugen sie dann 1000 Samples aus einer 2 dimensionalen Normalverteilung mit $\mu = (0, 0)$ und der Kovarianzmatrix: $\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$.
2. Projizieren Sie die 2-D Daten (x_1, x_2) jeweils auf x_1 und x_2 und plotten Sie das Histogramm.
3. Welche Vermutung haben Sie bzgl. der Randverteilungen? Sind diese Normalverteilungen?

5 Multinomialverteilung

Zeigen Sie, dass die Binomialverteilung ein Spezialfall der Multinomialverteilung ist.