Lösung der Aufgaben zur Klausurvorbereitung

1. Setzen Sie den Startwert des Zufallszahlengenerators auf Ihre Matrikelnummer!

   ```
   > set.seed(123456)
   ```

2. Welches ist die maximal mögliche Zykluslänge für einen linearen Kongruenzgenerator der Form

   $$X_{n+1} = aX_n + c \mod m?$$

   Wieso?
   Die maximale Zykluslänge ist m. Da es nur m verschiedene Eingabewerte für den Generator gibt, gibt es auch höchstens m verschiedene Ausgaben.

3. Gegeben ist eine Zufallsvariable mit Realisierungen aus $[0, 1]$, und ihre Dichte

$$f_X(x) = 3x^2 \text{ für } 0 \leq x \leq 1.$$

Erzeugen Sie jeweils ein R Programm, welches eine Zufallszahl gemäß dieser Verteilung

- mittels ARM erzeugt: Der maximale Funktionswert von f auf [0,1] beträgt 3. Damit ist

```
repeat{
kandidat <- runif(1)
if (runif(1, 0, 3) <= 3*kandidat^2) break
}
kandidat
```

ein Programmfragment, das eine Zufallszahl gemäß ARM aus f erzeugt.

- bzw. mittels der Inversionsmethode erzeugt.
  Zunächst berechnen der benötigten Verteilungsfunktion

$$\int f_X(x) = x^3 = F(x) \text{ für } 0 \leq x \leq 1.$$

  Also ist

$$F^{-1}(y) = y^{\frac{1}{3}} \text{ für } 0 \leq y \leq 1.$$

  Damit ist
  `runif(1)^(1/3)`
  ein `R` Befehl, welcher eine Zufallszahl aus F generiert.

4. Berechnen Sie mittels einer geeigneten Monte-Carlo-Methode das Integral der Funktion

$$f(x,y) = |\sin(x+y)| \cdot x^2 e^{-y-x} \text{ für } 0 \leq x \leq 10, 0 \leq y \leq 5.$$

Die Funktion in R:

```
funktionf <- function (x, y){abs(sin(x+y))*x^2*exp(-y-x)}
```

Erlauben Sie als Parameter die Anzahl der Durchläufe. Geben Sie den Schätzer für das Integral und die Konfidenzintervallbreite für $\alpha = 0.05$ und $n = 10000$ Wiederholungen an. Begründen Sie die Wahl der Simulationsmethode!
Die bessere Variante ist die Simulation durch den Erwartungswert von f über $Q := [0,10] \times [0,5]$. ARM verbietet sich etwas, da das Maximun von f über Q unbekannt ist.

Zu beachten ist, dass die Gleichverteilung diesmal über dem Quader $Q = [0, 10] \times [0, 5]$ betrachtet wird. Die Dichte hat somit im gesamten Quader den Wert

$$f(x) = \frac{1}{50} \text{ für} x \in Q.$$

Also ist

$$\int_Q f(x) \approx 50 * \sum_i f(X_i) \text{ mit } X_i \sim U_Q.$$

Die gewünschte Funktion ließe sich also beispielsweise schreiben als:

```
intf <- function(n=10000){ result <- rep(NA, n)
for (i in 1:n){ result[i] <- 50* funktionf(10*runif(1),
                    5*runif(1)) }
xquer <- mean(result); sddach <- sqrt(var(result))
cat("Mittel:", xquer, "\n")
cat("KI Breite:", 2*sddach/sqrt(n)*qnorm(0.975),"\n") }
```

Mit dem Ergebnis:

```
> intf()
Mittel: 1.315866
KI Breite: 0.1116774
```

5. Schreiben Sie ein Programm, um die Verteilung des Abstands zweier zufällig gewählter Punkte im Intervall [0,1] zu simulieren.

```
abstand1 <- function(){runif(1)-runif(1)}
n=1000
result<-replicate(n,abstand1())
```

   - Der Bereich $[0,1]^n$ heißt n-dimensionaler Einheitswürfel. $[0,1]$ ist also der 1-dimensionale Einheitswürfel.
   - Erweitern Sie dieses Programm, so dass als Parameter die Dimension des Einheitswürfels angegeben werden kann, in dem die beiden Punkte

gewählt werden.
Abstand zweier Punkte $x, y \in R^n$:

$$|x - y| := \sqrt{\sum_{1}^{n} (x_i - y_i)^2}$$

```
abstandn <- function(n){sqrt(sum(replicate(n, abstand1())^2))}
n=1000
result<-replicate(n,abstandn(2))
```

- Geben Sie für n= 100 und 10000 Wiederholungen und für die Dimensionen 1 bis 10 den Erwartungswert und die Varianz der Schätzer an.

```
for (runs in c(100, 1000)){  for (dimension in 1:10){
    result <- replicate(runs,abstandn(dimension))
    cat("Wiederholungen: ", runs, "Dim: ", dimension,
        "Mittel: ", mean(result), "Var: ", var(result), "\n")
  }}
```

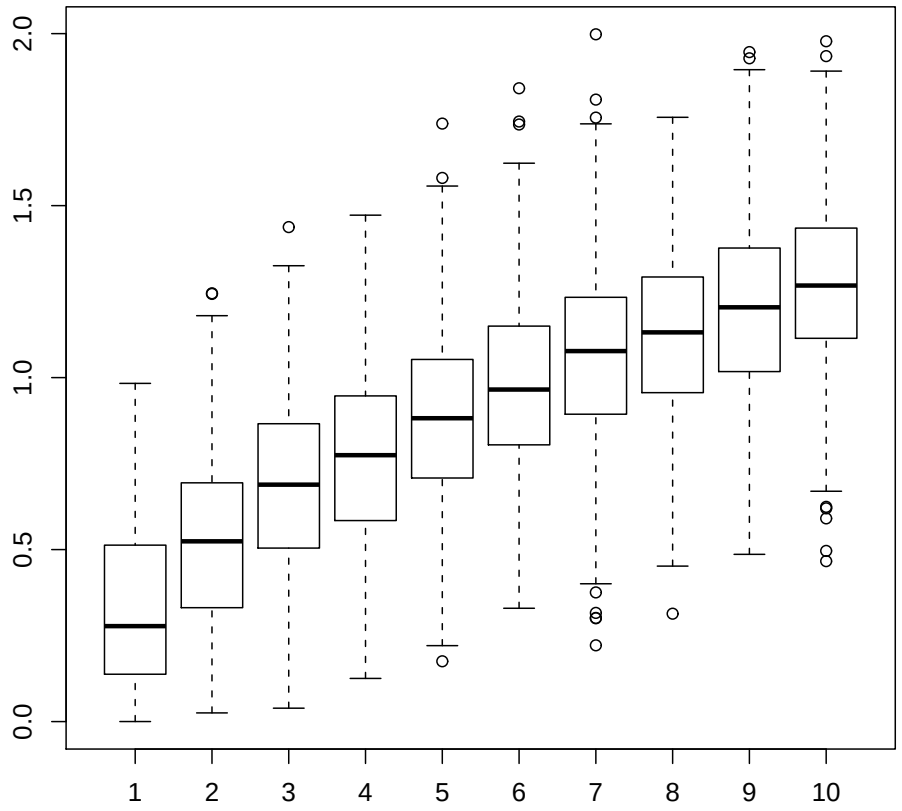
Mit dem Ergebnis:

```
Wiederholungen:  100 Dim:  1 Mittel:  0.3529291 Var:  0.07268224
Wiederholungen:  100 Dim:  2 Mittel:  0.5124869 Var:  0.05084794
Wiederholungen:  100 Dim:  3 Mittel:  0.6567936 Var:  0.05607701
Wiederholungen:  100 Dim:  4 Mittel:  0.7936078 Var:  0.0698053
Wiederholungen:  100 Dim:  5 Mittel:  0.8767297 Var:  0.05746693
Wiederholungen:  100 Dim:  6 Mittel:  0.9530936 Var:  0.07585346
Wiederholungen:  100 Dim:  7 Mittel:  1.059059 Var:  0.05522959
Wiederholungen:  100 Dim:  8 Mittel:  1.148202 Var:  0.05305126
Wiederholungen:  100 Dim:  9 Mittel:  1.208734 Var:  0.05694839
Wiederholungen:  100 Dim:  10 Mittel:  1.263896 Var:  0.04907321
Wiederholungen:  1000 Dim:  1 Mittel:  0.3426647 Var:  0.05708579
Wiederholungen:  1000 Dim:  2 Mittel:  0.5128693 Var:  0.05666529
Wiederholungen:  1000 Dim:  3 Mittel:  0.6623771 Var:  0.06290974
Wiederholungen:  1000 Dim:  4 Mittel:  0.778397 Var:  0.06303114
Wiederholungen:  1000 Dim:  5 Mittel:  0.892686 Var:  0.05847993
Wiederholungen:  1000 Dim:  6 Mittel:  0.9639408 Var:  0.06722068
Wiederholungen:  1000 Dim:  7 Mittel:  1.048060 Var:  0.05860652
```

```
Wiederholungen:  1000 Dim:  8 Mittel:  1.13597 Var:  0.06477617
Wiederholungen:  1000 Dim:  9 Mittel:  1.207557 Var:  0.06086894
Wiederholungen:  1000 Dim:  10 Mittel:  1.256934 Var:  0.060757
```

- Zusatzaufgabe: Erzeugen Sie parallele Boxplots für die Dimensionen 1 bis 10 und n=1000 Wiederholungen für die gesuchten Abstände.

```
result <- matrix(NA, ncol=10, nrow=1000)
for (dimension in 1:10)
  result[,dimension] <- replicate(1000, abstandn(dimension))
boxplot(result)
```


0.0
0.5
1.0
1.5
2.0
1
2
3
4
5
6
7
8
9
10

Grundlagen der Resampling Methoden

- Angelehnt an eine Vorlesung von Rozenn Dahyot, Trinity College, Dublin.
- Literatur:
  - An Introduction to the Bootstrap, B. Efron und R.J. Tibshirani
  - Computer Intensive Statistical Methods, J.S. Urban Hjorth

Überblick

- Die Resampling Methoden zerfallen grob in drei oder vier Klassen.
- Der Bootstrap, auch Münchhausenmethode, sowohl in der parametrischen, als auch der nicht-parametrischen Variante.
- Der Jackknife.
- (Permutationstests, werden hier nicht behandelt).
- Die Kreuzvalidierung (*cross-validation) CV.*

Einige Notationen

- Gegeben sei eine Stichprobe $\mathbf{x} = (x_1, \ldots, x_n)$ als Realisierung von n i.i.d. Zufallsvariablen $X_i \sim F$, einer unbekannten Verteilungsfunktion.

- Die empirische Verteilungsfunktion wird mit $\hat{F}$ bezeichnet, $\hat{f}$ ist die empirische Dichte.

- Evtl. neu ist die empirische Dichte:

$$\hat{f}(x) := \frac{1}{n} \sum_{1}^{n} \delta(x - x_i),$$

wobei die δ-Funktion (Kroneckers-δ) definiert ist als:

$$\delta(0) = 1,\ \delta(x) = 0 \text{ sonst.}$$

- Eine Statistik oder auch Schätzfunktion $\hat{\theta}(\mathbf{x})$ ist eine Abbildung von einer Stichprobe in den Parameterraum (z.B. $\mathcal{R}$), z.B. für den Mittelwert

$$\hat{\theta}(\mathbf{x}) = \frac{1}{n}\sum x_i\ (= \hat{\mu}).$$

  Um die Abhängigkeit von der Stichprobe zu betonen schreibt man auch $s(\mathbf{x})$ für $\hat{\theta}$.

- Eine andere Sicht auf die Schätzfunktion fasst einen Parameter einer Dichte auf als eine Abbildung t aus der Menge der Dichten in den Parameterraum: $\theta = t(f)$.

- Z.B. ist für den Mittelwert einer bekannten Dichte f

$$\theta = E_f(X) = t(f) = \int_{-\infty}^{\infty} x f(x) dx.$$

- Daraus ergibt sich eine naheliegende Definition für einen Schätzer eines Parameters θ über das sogenannte *plug-in* Prinzip. Es wird statt der unbekannten Dichte f einfach das empirische Gegenstück $\hat{f}$ eingesetzt:

$$\hat{\theta} = t(\hat{f})!$$

- $\hat{\theta}$ heißt der Plug-in Schätzer für θ.

- Im Beispiel für den Mittelwert

$$\hat{\theta} = E_{\hat{f}}(X) = t(\hat{f}) = \int_{-\infty}^{\infty} x\hat{f}(x)dx = \bar{x}$$

- oder für die Varianz

$$\hat{\theta} = t^*(\hat{f}) = \int_{-\infty}^{\infty} (x - \bar{x})^2 \hat{f}(x)dx = \frac{1}{n}\sum_{1}^{n}(x_i - \bar{x})^2 = \hat{\sigma}^2.$$

Genauigkeit von Schätzern

- Mittels der Funktion t (bzw. s) kann man nun bei gegebener Stichprobe $\mathbf{x}$ einen Schätzer $\hat{\theta}$ berechnen.

- Bleibt die Frage, wie man die Eignung der Schätzfunktion als Schätzer für den wahren Parameter beurteilt.

- Aus Statistik II sind Erwartungstreue und Konsistenz als Kriterien bekannt, ebenso wie die Standardfehler, die bei der Konstruktion von Konfidenzintervallen genutzt werden.

- Zur Erinnerung: Der Standardfehler ist die Standardabweichung eines Schätzers $\hat{\theta}$ aufgefasst als Zufallsvariable.

$$\mathsf{se}(\hat{\theta}) := \sqrt{\mathsf{var}_f(\hat{\theta})}.$$

- Ist f unbekannt, so kann man auch hier mittels *plug-in* Prinzip zu einer Schätzung des Standardfehlers gelangen. Allgemein also

$$\hat{\mathsf{se}}(\hat{\theta}) = \mathsf{se}_{\hat{f}}(\hat{\theta}) = \sqrt{\mathsf{var}_{\hat{f}}(\hat{\theta})}$$

  und für das Beispiel des Standardfehlers bei $\bar{x}$ gilt

$$\hat{\mathsf{se}}(\bar{x}) = \frac{\hat{\sigma}}{\sqrt{n}}.$$

- Entsprechend können über dieses Prinzip auch die bekannten Konfidenzintervalle und ihre Schätzer hergeleitet werden.

Noch Probleme?

- Ja! Denn oft kann man die Funktionen t nicht (leicht) explizit aufschreiben. Wie sieht z.B. die entsprechende Abbildung für den Median aus?

- In der Praxis gibt es nur eine Stichprobe $\mathbf{x}$, also auch nur eine Realisierung der Schätzfunktion $\hat{\theta}$. Wie kann man also z.B. se$(\hat{\theta})$ angeben, wenn nur eine Stichprobe existiert?

- An dieser Stelle setzen die Resampling Methoden an.

Der Bootstrap

- Gegeben sei eine beliebige Schätzfunktion $\hat{\theta}$, sowie die einzige Stichprobe $\mathbf{x}$ mittels derer ein Schätzwert als Realisierung von $\hat{\theta}$ berechnet werde.
- Wie kann man trotzdem die Güte dieses Schätzers beurteilen? Wie kommt man an Schätzungen für Bias, Varianz etc dieser Schätzfunktion?
- Eine Methode ist die computerintensive Resampling-Technik des Bootstrapping, die im Folgenden vorgestellt wird.
- Der Name kommt übrigens aus den Abenteuern des Baron Münchhausen!
- Der Bootstrap kommt nicht nur mit einer einzigen Stichprobe aus, er verzichtet auch auf jede Modellannahme!

- Die Struktur des Schätzers $\hat{\theta}$ darf beliebig kompliziert sein, da die theoretischen Eigenschaften von θ uns nicht interessieren. Es sind nur die Realisierungen, die aus der Stichprobe gewonnen werden interessant!

- Das hier vorgestellte Prinzip wurde 1979 von Efron entwickelt, um die Standardfehler beliebiger Schätzer zu bestimmen. In den Anwendungen wurde das Vorgehen verfeinert, das Prinzip ist aber unverändert bestehen geblieben.

Die Bootstrap-Stichprobe

- Definition: Eine Bootstrap-Stichprobe $x^* = (x_1^*, \ldots, x_n^*)$ erhält man, indem aus $\mathbf{x} = (x_1, \ldots, x_n)$ n-mal mit Zurücklegen gezogen wird.

- Beispiel: Sei $\mathbf{x} = (1, 2, 3)$, dann sind z.B. (1,1,3) oder (2, 1, 3) mögliche Bootstrap-Stichproben.

- Nun wird endlich klar, warum die Methoden Resampling genannt werden!

- Zu jeder Bootstrap-Stichprobe x^* kann mittels des *plug-in*-Prinzips eine Bootstrap-Schätzung $\hat{\theta}^* = s(\mathbf{x}^*)$ durchgeführt werden. Über viele sogenannte Replikationen kann dann deren Standardfehler empirisch bestimmt werden!

- In `R` wird mit dem Befehl `sample(x, length(x), replace=TRUE)` eine Bootstrap-Stichprobe aus dem Vektor `x` erzeugt.

Welchen Anteil der Beobachtungen verliert man?

- Zieht man mit Zurücklegen aus einem endlichen Vektor, werden manche Einträge doppelt, manche gar nicht im Bootstrap-Sample auftreten.
- Die Wahrscheinlichkeit für eine spezielle Beobachtung x_i in einer Bootstrap-Stichprobe zu fehlen, lässt sich berechen.
- Angenommen alle x_i sind verschieden, dann ist die Wkeit, dass ein Wert x' nicht gezogen wird

$$P(x_i \neq x',\ 1 \leq i \leq n) = (1 - \frac{1}{n})^n.$$

Für $n \to \infty$ konvergiert diese Wkeit gegen $\frac{1}{e} = 0.37$.

Efrons Bootstrap

- Nach diesen Vorarbeiten kann nur für eine beliebige Schätzfunktion $s(\mathbf{x})$ mittels des Bootstrap ein Schätzer für ihren Standardfehler $\hat{\text{se}}(\hat{s(\mathbf{x})})$angegeben werden.

- Gegeben seien N Bootstrap-Stichproben $x_1^*, \ldots, x_N^*$ aus $\mathbf{x}$.

- Berechne nun zu jeder Stichprobe $1 \leq k \leq N$

$$\hat{\theta}_k^* = s(x_k^*)$$

- Damit ergibt sich der Bootstrap-Schätzer $\hat{\text{se}}_B$ für den Standardfehler

se$(\hat{\theta})$ durch die N Wiederholungen zu

$$\hat{\text{se}}_B = \left[\sum_1^N \left[\hat{\theta}^* - \bar{\theta^*} \right]^2 \right]^{\frac{1}{2}}$$

mit $\bar{\theta^*} = \frac{1}{N} \sum_1^N \hat{\theta}^*$.

- Bei genügend Replikationen lassen sich die vollständigen Verteilungen beliebiger Teststatistiken approximieren!

Liveaufgabe zum Bootstrap

- Erzeugen Sie sich eine Stichprobe vom Umfang 100 aus der N(1,2) Verteilung.
- Bestimmen Sie den Standardfehler für den Mittelwertschätzer mittels Bootstrapping. Nutzen Sie 100, 200, ..., 1000 Replikationen!
- Vergleichen Sie das Ergebnis mit dem theoretischen Wert. Welche Zahl von Replikationen scheint angemessen?
- Schätzen Sie die Verteilung des Schätzers mittels eines Histogramms!

Lösung

```
x <- rnorm(100, mean=1, sd=sqrt(2))

for (runs in seq(100, 1000, 100)){
  bootschaetzer <- replicate(runs,
                    mean(sample(x,length(x), replace=TRUE)))
  cat ("Mittel der Bootstrapschätzer ",mean(bootschaetzer),"\n")
  cat ("Standardfehler für ", runs, " Replikationen: ",
            sd(bootschaetzer), "\n")
}


sqrt(2)/10 ### theoretisch
hist(bootschaetzer)
```

Aufgabe zum Bootstrap

- Die vorhergehende Aufgabe ließe sich auch noch theoretisch lösen. Diese nur sehr schwer ...

- Kombinieren Sie eine Stichprobe vom Umfang 35 aus der Gleichverteilung auf $[0, 6]$ und eine Stichprobe vom Umfang 65 aus $N(-1, 1)$ zu einer Stichprobe vom Umfang 100.

- Schätzen Sie den Median der zugrundeliegenden Verteilung und leiten Sie mittels Bootstrap eine Schätzung des Standardfehlers für diesen Medianschätzer her. (N=200 Replikationen)

- Ensprechend für den Interquartilsabstand. (N=200 Replikationen)