Robuste Statistik

Musterlösung zu Blatt 4

Aufgabe 4.1:
Zu zeigen: $\rho(z) = \ln((1 + z^2))$ definiert eine Maximum-Likelihood-Schätzung für μ.

Um eine Maximum-Likelihood-Schätzung zu definieren muss $\rho(z - \mu) = -\ln(f_\mu(z))$ gelten und $f_\mu(z)$ eine Dichtefunktion darstellen.

Berechne zunächst das potentielle $f_\mu(z)$:

$$\begin{aligned} \rho(z-\mu) = &\ln((1+(z-\mu)^2)) && = -\ln(f_\mu(z)) \\ \Leftrightarrow &-\ln((1+(z-\mu)^2)) && = \ln(f_\mu(z)) \\ \Leftrightarrow &\frac{1}{(1+(z-\mu)^2)} && = f_\mu(z) \end{aligned}$$

Zeige nun, dass $f_\mu(z)$ Dichte ist bzw. sich streng monoton in eine Dichte transformieren lässt, indem gezeigt wird, dass $f_\mu(z)$ nicht-negativ ist und das Integral über $f_\mu(z)$ Eins ergibt bzw. die entsprechende Konstante bestimmt wird.

Nicht-Negativität:
Alle Konstanten in der Funktion sind größer als Null und die Variable befindet sich zusammen mit dem potentiell negativen Parameter der Funktion in einem quadratischen Term, wodurch alle Teile der Funktion größer als Null sind. Da in dem Term keine Subtraktionen vorkommen, ist dieser Term insgesamt immer positiv.

Integral:

$$\begin{aligned} \int_{-\infty}^{\infty} f_\mu(z)dz &= \int_{-\infty}^{\infty} \frac{1}{(1+(z-\mu)^2)}dz \\ &= \int_{-\infty}^{\infty} \frac{1}{1+(z-\mu)^2}dz \\ &= [\arctan(z-\mu)]_{-\infty}^{\infty} \\ &= (\arctan(\infty-\mu) - \arctan(-\infty-\mu)) \\ &= (\arctan(\infty) - \arctan(-\infty)) \\ &= \left(\frac{\pi}{2} - \left(-\frac{\pi}{2}\right)\right) \\ &= \pi \end{aligned}$$

Somit ist $\frac{1}{\pi} f_\mu(z)$ eine Dichtefunktion und $\rho(z)$ definiert eine Maximum-Likelihood-Schätzung für μ. (Die restliche Rechnung dazu ist analog zu Aufgabe 3.2.)

Aufgabe 4.2:
Gegeben ist der Datensatz 2, 3, 5, 6, 9.

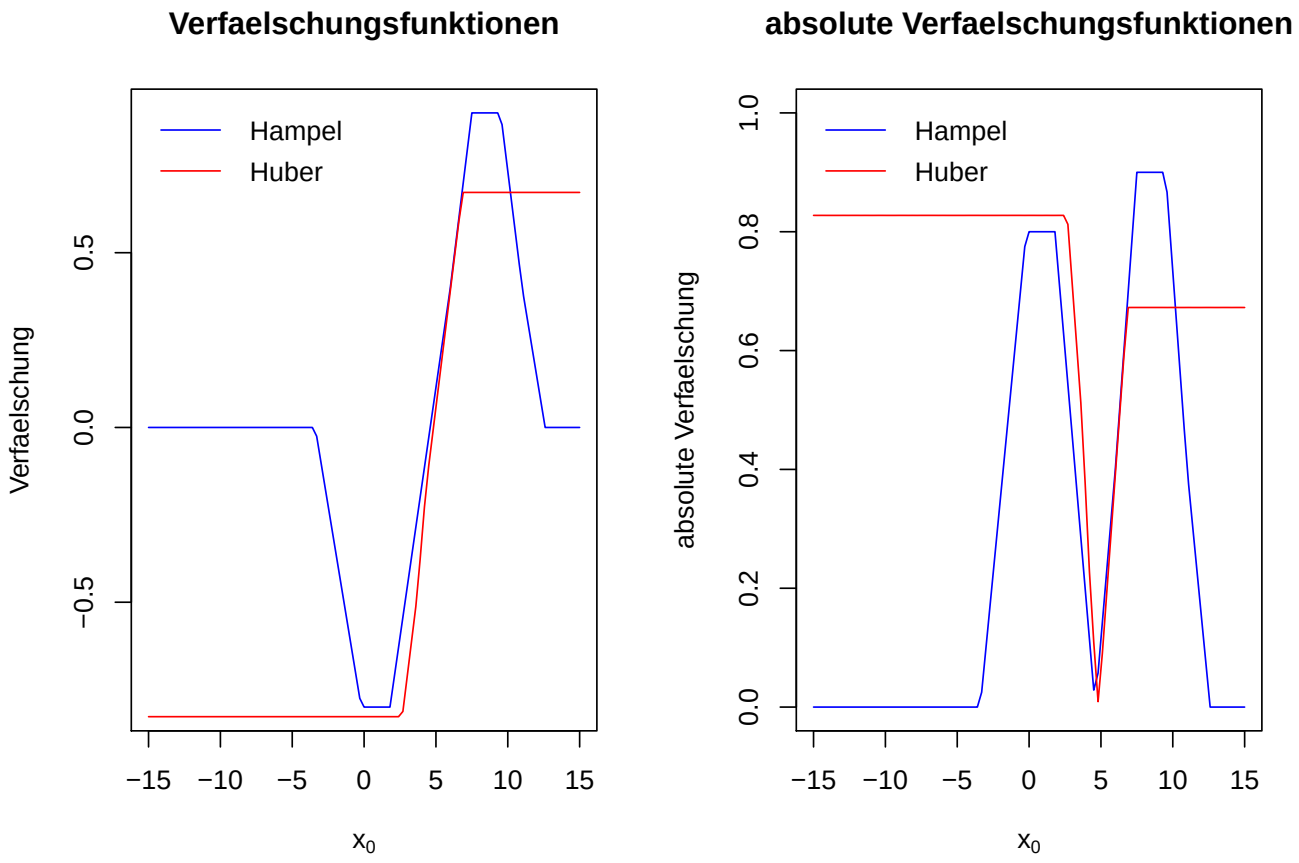
R-Code zu dieser Aufgabe (siehe auch R-Datei):

```r
# Datensatz:
data <- c(2, 3, 5, 6, 9)

# rho_hampel: Score-Funktion des Hampel-M-Schaetzers
# Eingabe: z (numeric): Vektor, an dessen Stellen die Score-Funktion ausgewertet
#                       werden soll
#          a, b, c, d (numeric(1)): Parameter des Schaetzers, hier voreingestellt
#                                   auf a = 2, b = 4 und c = 8
# Ausgabe: (numeric): Vektor mit Werten der Score-Funktion
rho_hampel <- function(z, a = 2, b = 4, c = 8){
  ifelse(abs(z) <= a, 1/2 * z^2,
    ifelse(abs(z) <= b, a * abs(z) - a^2/2,
      ifelse(abs(z) < c, a * b  - a^2/2 + a/2 * (c - b) * (1 - ((c - abs(z))/(c - b))^2),
        a * b  - a^2/2 + a/2 * (c - b))))
}

# rho_huber: Score-Funktion des Huber-M-Schaetzers
# Eingabe: z (numeric): Vektor, an dessen Stellen die Score-Funktion ausgewertet
#                       werden soll
#          d (numeric(1)): Parameter des Schaetzers, hier voreingestellt auf d = 1.345
# Ausgabe: (numeric): Vektor mit Werten der Score-Funktion
rho_huber <- function(z, d = 1.345){
  ifelse(abs(z) <= d, 1/2 * z^2, d * abs(z) - d^2/2)
}



# estimator: Funktion zur Berechnung der Lageschaetzung aus den Score-Funktionen
# Eingabe: rho (function): Score-Funktion, welche vom R^N in den R^N abbildet
# Ausgabe: (function): Funktion zur Berechnung der Lageschaetzung unter
#                      Beruecksichtigung der Score-Funktion. Die zurueckgegebene
#                      Funktion erhaelt einen Datenvektor und eventuell weitere
#                      Hyperparameter als Eingabe
estimator <- function(rho){
  function(dat, ...) optimize(function(x) sum(rho(dat - x, ...)),
    interval = range(dat))$minimum
}

# verfaelschung: Funktion zur Berechnung der Verfaelschung
# Eingabe: mu (function): Funktion zur Berechnung der Lageschaetzung
#          data (numeric): Datensatz zu dem die Verfaelschung berechnet werden soll
#          x0 (numeric): Vektor mit Ausreissern
#          ... : weitere Argumente an die mu-Funktion (Hyperparameter)
# Ausgabe: (numeric): Vektor mit Verfaelschungswerten an den Stellen x0
verfaelschung <- function(mu, data, x0, ...){
  true <- mu(data, ...)
  sapply(x0, function(x) mu(c(x, data), ...) - true)
}

# abs_verfaelschung: Funktion zur Berechnung der absoluten Verfaelschung
# Eingabe: mu (function): Funktion zur Berechnung der Lageschaetzung
#          data (numeric): Datensatz zu dem die Verfaelschung berechnet werden soll
#          x0 (numeric): Vektor mit Ausreissern
#          ... : weitere Argumente an die mu-Funktion (Hyperparameter)
# Ausgabe: (numeric): Vektor mit absoluten Verfaelschungswerten an den Stellen x0
abs_verfaelschung <- function(mu, data, x0,...){
  true <- mu(data, ...)
  sapply(x0, function(x) abs(mu(c(x, data), ...) - true))
}
```

```
# Plotten der Verfaelschungen in einem Bereich von -15 bis 15
curve(verfaelschung(mu = estimator(rho_hampel), data = data, x0 = x),
  from = -15, to = 15, xlab = expression(x[0]), ylab = "Verfaelschung",
  col = "blue", main = "Verfaelschungsfunktionen")
curve(verfaelschung(mu = estimator(rho_huber), data = data, x0 = x),
  from = -15, to = 15, add = TRUE, col = "red")
legend("topleft", legend = c("Hampel", "Huber"), col = c("blue", "red"),
  lty = 1, bty = "n")

# Plotten der absoluten Verfaelschungen in einem Bereich von -15 bis 15
curve(abs_verfaelschung(mu = estimator(rho_hampel), data = data, x0 = x),
  from = -15, to = 15, xlab = expression(x[0]), ylab = "absolute Verfaelschung",
  col = "blue", main = "absolute Verfaelschungsfunktionen", ylim = c(0, 1))
curve(abs_verfaelschung(mu = estimator(rho_huber), data = data, x0 = x),
  from = -15, to = 15, add = TRUE, col = "red")
legend("topleft", legend = c("Hampel", "Huber"), col = c("blue", "red"),
  lty = 1, bty = "n")
```

Man beachte bei der Optimierung des Hampel-M-Schätzers, dass für zu große Definitionsbereiche `optimize()` eventuell nur ein lokales Minimum findet, da in den Grenzen außerhalb des Intervalls $(-c, c)$ konstante Plateaus vorliegen, welche als Minimum erkannt werden und mehrere Hügel vorliegen können. Hier müsste man im Allgemeinen mit effizienteren Optimierungsalgorithmen arbeiten, z.B. mit verschiedenen Initialpunkten mit der Funktion `optim()` verwenden, um das globale Minimum zu finden. Sinnvolle Startpunkte sind verschiedene p-Quantile für $p \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$ z.B., da die Hampel-M-Schätzung in der Regel nahe eines Bulks der Daten liegt. Die Huber-M-Schätzung ist unproblematisch, da ein konvexes Optimierungsproblem vorliegt.

Die daraus entstandenen Grafiken:



Interpretation:
Die Charakteristik der Hampel- und der Huber-M-Verfälschungsfunktion ist verschieden. Während die Huber-M-Schätzung durch große Ausreißer beeinflusst wird, wird der Hampel-M-Schätzer nur von Ausreißern im ungefähren Bereich von -3 bis 13 beeinflusst. Ausreißer kleiner als -3 oder größer als 13 führen zu keiner Verfälschung. Dies beruht darauf, dass die Score-Funktion des Hampel-M-Schätzers für betragsmäßig hinreichend große Werte konstant ist und die Einflussfunktion (die Ableitung ψ_{Hampel}) dann den Wert 0 annimmt.

Hingegen wird die Huber-M-Schätzung von betragsmäßig großen Ausreißern beeinflusst. Allerdings ist sowohl beim Hampel- als auch beim Huber-M-Schätzer die maximale Verfälschung vergleichsweise gering. Für den gleichen Datensatz haben wir in Aufgabe 1.2 (durch Ersetzen) maximale Verfälschungen von ∞ für das arithmetische Mittel bzw. 2 für den Median erhalten. Die maximale Verfälschung des Hampel-M-Schätzers liegt hingegen bei 0.9 und die des Huber-M-Schätzers liegt bei 0.8275. Diese M-Schätzer liefern somit deutlich geringere maximale Verfälschungen als die einfachen Schätzer vom ersten Übungsblatt. Man beachte bei dem Vergleich, dass hier die Stichprobengröße sechs und beim ersten Übungsblatt fünf beträgt und der Vergleich evtl. hinken könnte. Des Weiteren ist zu sehen, dass die maximale Verfälschung des Hampel-M-Schätzers hier größer ist als die maximale Verfälschung des Huber-M-Schätzers. Betrachtet man allerdings andere Maßzahlen der Verfälschung (z.B. das Integral), so gewinnt der Hampel-M-Schätzer. (Anreize sollten in der Übung für verschiedene Maßzahlen neben dem Bruchpunkt, z.B. Einflussfunktionen, gesetzt werden.)

Aufgabe 4.3:
Diese Aufgabe kann sowohl mit `R` gelöst werden als auch per Hand.

Lösung mit `R`:

```
# Datensatz
daten <- c(-1, 0, 1, 5, 1000)

# lts: Funktion zur Berechnung des LTS-Schaetzers
# Eingabe: data (numeric): Datenvektor
#          k, h (numeric(1)): Hyperparameter
# Ausgabe: (numeric(1)): LTS-Schaetzung fuer den Datensatz
lts <- function(data, k, h){
  tmp <- function(mu){
    r <- sort(abs(data - mu))
    summanden <- r[k:h]
    sum(summanden^2)
  }
  optimize(tmp, interval = range(data))$minimum
}

# Bestimme die LTS-Schaetzung fuer den gegebenen Datensatz mit k = 1, und h = 3
lts(daten, 1, 3)
# [1] -1.796586e-10

# Bestimme das getrimmte arithmetische Mittel mit beta = 0.2
mean(daten, trim = 0.2)
# [1] 2
```

Lösung per Hand:
LTS-Schätzung:
Benötigte Mengen: $M_1 = \{-1, 0, 1\},\ M_2 = \{0, 1, 5\},\ M_3 = \{1, 5, 1\,000\}$

Für M_1: $\mu = \frac{1}{3}(-1+0+1) = 0 \Rightarrow Q(\overline{y(M_1)}, y(M_1)) = (-1-0)^2 + (0-0)^2 + (1-0)^2 = 1+0+1 = 2$
Für M_2: $\mu = \frac{1}{3}(0+1+5) = 2 \Rightarrow Q(\overline{y(M_2)}, y(M_2)) = (0-2)^2 + (1-2)^2 + (5-2)^2 = 4+1+9 = 14$
Für M_3: $\mu = \frac{1}{3}(1+5+1\,000) = \frac{1006}{3} \Rightarrow Q(\overline{y(M_3)}, y(M_3)) = \left(1 - \frac{1006}{3}\right)^2 + \left(5 - \frac{1006}{3}\right)^2 + \left(1\,000 - \frac{1006}{3}\right)^2 = \frac{5988090}{9}$
$\Rightarrow \hat{\mu} = 0$, da $2 < 14 < \frac{5988090}{9}$

Wesentlich ist hier, dass die Werte nicht ausgerechnet werden brauchen, sondern dass nur erkannt wird, welcher Wert minimal ist.

Getrimmtes Mittel:
$\frac{1}{5-2\cdot 0.2\cdot 0.5} \sum_{n=0.2\cdot 5+1}^{5+0.2\cdot 5} y_{(n)} = \frac{1}{3} \sum_{n=2}^{4} y_{(n)} = \frac{1}{3}(0+1+5) = 2$

Beachte: Beim getrimmten Mittel muss gegebenenfalls gerundet werden.

Interpretation:
Es ist zu sehen, dass die LTS-Schätzung einen kleineren Schätzwert liefert als das getrimmte Mittel. Dies ist damit zu erklären, dass das getrimmte Mittel sowohl große als auch kleine Datenpunkte wegtrimmt, wohingegen bei der LTS-Schätzung die Beobachtungen mit den größten absoluten Residuen weggelassen werden. Bei beiden Schätzern gehen hier drei der fünf Beobachtungen in die Schätzung ein. Bei dem getrimmten Mittel sind dies in diesem Fall jedoch die Beobachtungen $y_{(2)}$, $y_{(3)}$, $y_{(4)}$, wohingegen für die LTS-Schätzung die Beobachtungen $y_{(1)}$, $y_{(1)}$, $y_{(3)}$ verwendet werden.
Im Allgemeinen neigt das getrimmte Mittel dazu potentiell wichtige Informationen weg zu trimmen, wohingegen die LTS-Schätzung mehr Struktur berücksichtigt.