Kurs: KI T-InfT-008 und 010 Datenmengen und Embedded Systems

Cândido Vieira

30.09.2024

Balthasar-Neumann-Technikum (BNT)

Inhaltsverzeichnis

1. Datenbereinigung

- Datenbereinigung ist ein kritischer Schritt in der Datenvorbereitung.
- Outlier können das Verhalten von Modellen erheblich beeinflussen.
- Ziel: Outlier identifizieren und ihre Auswirkungen analysieren.

1. Datenbereinigung - Warum ist Datenbereinigung wichtig?

- Fehlerhaften Daten führen zu fehlerhaften Modellen.
- Unbehandelte Outlier können Modelle verzerren.
- Datenqualität ist entscheidend für robuste Modelle.

1. Datenbereinigung - Methoden zur Identifikation von Outliern

- Visuelle Inspektion: Boxplots, Streudiagramme.
- Statistische Methoden: Standardabweichung, IQR (Interquartilsabstand).
- Automatisierte Methoden: Z-Score, Local Outlier Factor (LOF).

2. Was sind Outlier?

- Werte, die signifikant von anderen Datenpunkten abweichen.

- Können auf Fehler, extreme Ereignisse oder ungewöhnliche Muster hinweisen.

- Beeinflussen statistische Maße wie Mittelwert und Standardabweichung stark.

2. Warum sind Outlier problematisch?

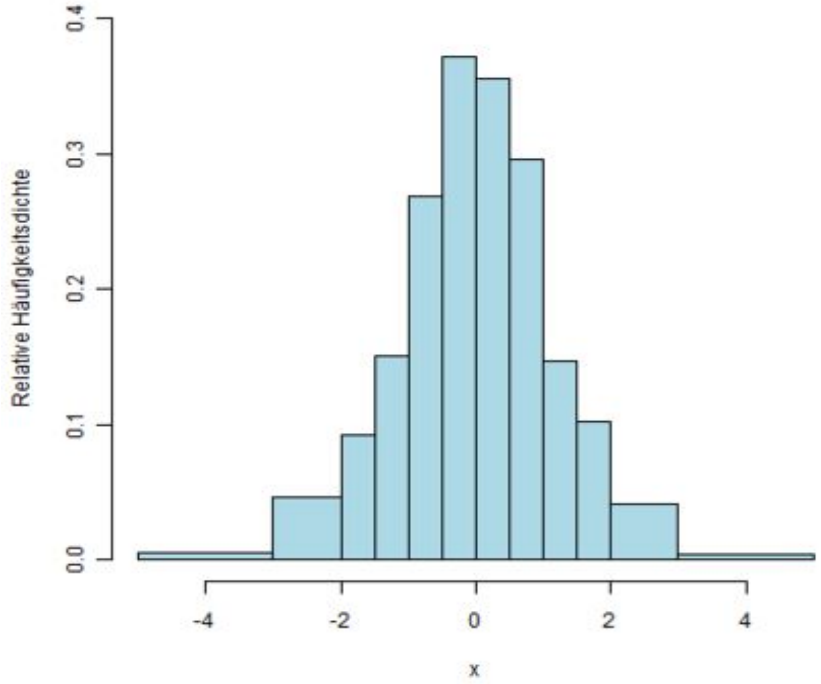
- Verzerrung des Mittelwerts und der Varianz.
- Mögliche Fehlinterpretationen der Daten.
- Starke Beeinflussung von Regressions-, Klassifizierungs- und Clustering-Modellen.

2. Auswirkungen von Outliern auf statistische Maße

- Mittelwert: Starke Verzerrung durch extreme Werte.
- Varianz: Größere Streuung der Daten.
- Regression: Sensibel auf Outlier, besonders bei linearen Modellen.

3. Methoden zur Erkennung von Outliern

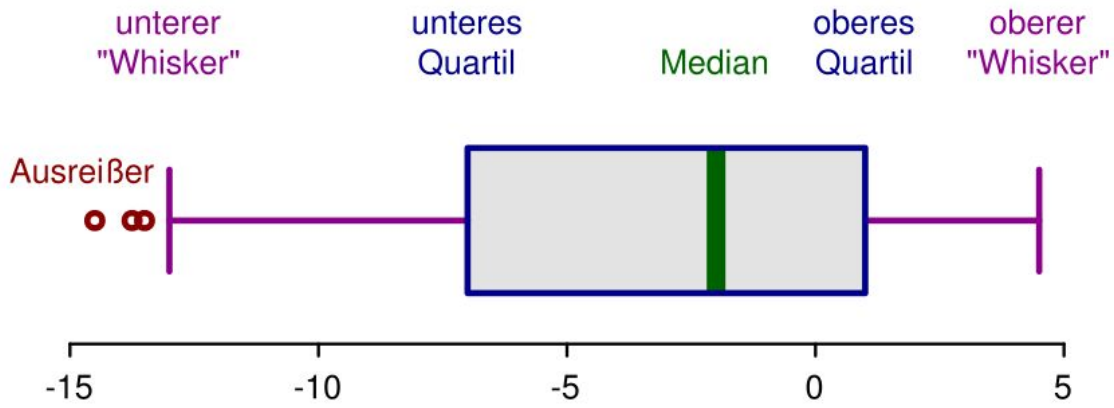
- Visuelle Methoden:
  - Histogramm:



Quelle: Thist German by Gundolf69, CC0 1.0, Wikimedia

3. Methoden zur Erkennung von Outliern

- Visuelle Methoden
  - Boxplot:



Quelle: Elements of a boxplot, Libellule, CC BY-SA 4.0, via Wikimedia

3. Methoden zur Erkennung von Outliern

- Statistische Methoden (Z-Score, IQR)

4. Outlier-Identifikation mit der Standardabweichung

- Annahme: Daten sind normalverteilt.

- Werte, die mehr als n Standardabweichungen vom Mittelwert entfernt sind, gelten als Outlier.

- Typische Schwellenwerte: 2 oder 3 Standardabweichungen.

4. Outlier-Identifikation mit der Standardabweichung

- Standardabweichung: Effektiv bei normalverteilten Daten.

4. Formel für die Standardabweichung

- Daten:

$$x = [10, 23, 44, 35, 26]$$

- Mittelwert:

$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$$

- Standardabweichung:

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2}$$

5. Outlier-Identifikation mit dem IQR (Interquartilsabstand)

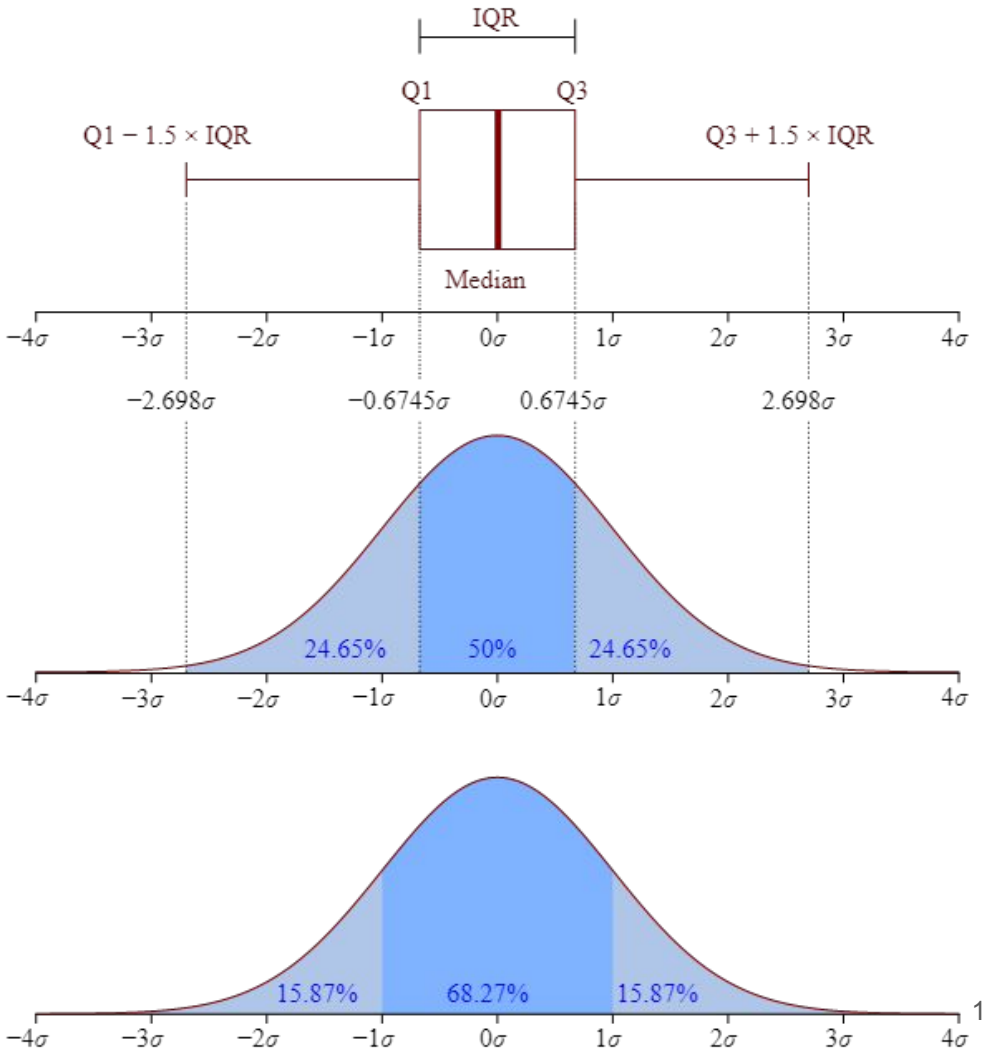
- Berechnet die Differenz zwischen dem 75. und 25. Perzentil.

- Formel: IQR = Q3 - Q1

- Werte außerhalb von 1,5-facher IQR über Q3 oder unter Q1 gelten als Outlier.

5. Visuelle Darstellung des IQR

- Q1: 25. Perzentil
- Q3: 75. Perzentil
- Werte außerhalb von Q1 - 1,5 * IQR und Q3 + 1,5 * IQR sind Outlier.



Quelle: **Boxplot vs PDF**, Mariordo, CC BY-SA 3.0, via Wikimedia

7. Umgang mit Outliern

- Entfernen: Falls der Outlier auf einen Fehler zurückzuführen ist.
- Transformieren: Log-Transformation, Box-Cox.
- Robuste Methoden verwenden: Median statt Mittelwert.

7. Techniken zur Entfernung von Outliern

- Manuelles Entfernen auf Basis visueller Analyse.

- Automatisches Entfernen mit Standardabweichung/IQR.

Übung

Standardabweichungsmethode

```python
import numpy as np
import matplotlib.pyplot as plt

# Generierung eines synthetischen Datensatzes
data = np.random.normal(0, 1, 1000)
data = np.append(data, [5, 6, 7])  # Fügen Sie einige Outlier hinzu

# Berechnung des Mittelwerts und der Standardabweichung
mean = np.mean(data)
std_dev = np.std(data)

# Identifikation von Outliern (mehr als 3 Standardabweichungen entfernt)
outliers = data[np.abs(data - mean) > 3 * std_dev]
print(f"Gefundene Outlier: {outliers}")

# Visualisierung
plt.figure(figsize=(10, 6))
plt.hist(data, bins=30, alpha=0.7)
plt.axvline(mean + 3*std_dev, color='r', linestyle='dashed', linewidth=2)
plt.axvline(mean - 3*std_dev, color='r', linestyle='dashed', linewidth=2)
plt.title('Outlier Detektion mit Standardabweichung')
plt.show()
```

IQR Methode

```
# IQR Methode
Q1 = np.percentile(data, 25)
Q3 = np.percentile(data, 75)
IQR = Q3 - Q1

# Definieren der Outlier-Grenzen
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

# Identifikation der Outlier
outliers_iqr = data[(data < lower_bound) | (data > upper_bound)]
print(f"Outlier (IQR Methode): {outliers_iqr}")
```

Vergleich der Methoden: Standardabweichung vs. IQR

- Standardabweichung: Effektiv bei normalverteilten Daten.

- IQR: Robust gegenüber extremen Ausreißern, auc

Übung: Synthetischer Datensatz mit Outliern, Duplikaten und fehlenden Werten

- Ziel: Den Datensatz bereinigen und Outlier identifizieren

```
# Erstellen eines Datensatzes mit Outliern, Duplikaten und fehlenden Werten
np.random.seed(42)
data = np.random.normal(50, 10, 100).tolist()
data += [100, 105, 110]  # Outlier hinzufügen
data += [50, 50]  # Duplikate hinzufügen
data += [None, None]  # Fehlende Werte hinzufügen

df = pd.DataFrame(data, columns=['Werte'])
print(df.head(10))
```

9. Übung: Datenbereinigung und Outlier-Analyse

- Schritt 1: Identifikation von Outliern mit der Standardabweichung und IQR.

- Schritt 2: Statistische Analyse des Datensatzes vor und nach der Entfernung von Outliern.

Statistischer Vergleich

- Berechnen Sie den Mittelwert, Median und die Standardabweichung des Datensatzes:
  - Vor der Bereinigung
  - Nach der Entfernung von Outliern

Statistischer Vergleich

```
# Statistische Analyse des ursprünglichen Datensatzes
mean_before = df['Werte'].mean()
median_before = df['Werte'].median()
std_before = df['Werte'].std()

# Bereinigung der Daten
df_cleaned = df.dropna().drop_duplicates()
outliers_removed = df_cleaned[np.abs(df_cleaned['Werte'] - df_cleaned['Werte'].mean()) <= (3 *
df_cleaned['Werte'].std())]

# Statistische Analyse nach der Bereinigung
mean_after = outliers_removed['Werte'].mean()
median_after = outliers_removed['Werte'].median()
std_after = outliers_removed['Werte'].std()

print(f"Vor der Bereinigung - Mittelwert: {mean_before}, Median: {median_before}, Standardabweichung: {std_before}")
print(f"Nach der Bereinigung - Mittelwert: {mean_after}, Median: {median_after}, Standardabweichung: {std_after}")
```

Fazit

- Outlier können das Verhalten eines Modells stark beeinflussen.

- Robuste Methoden zur Outlier-Entfernung und Datenbereinigung verbessern die Datenqualität.