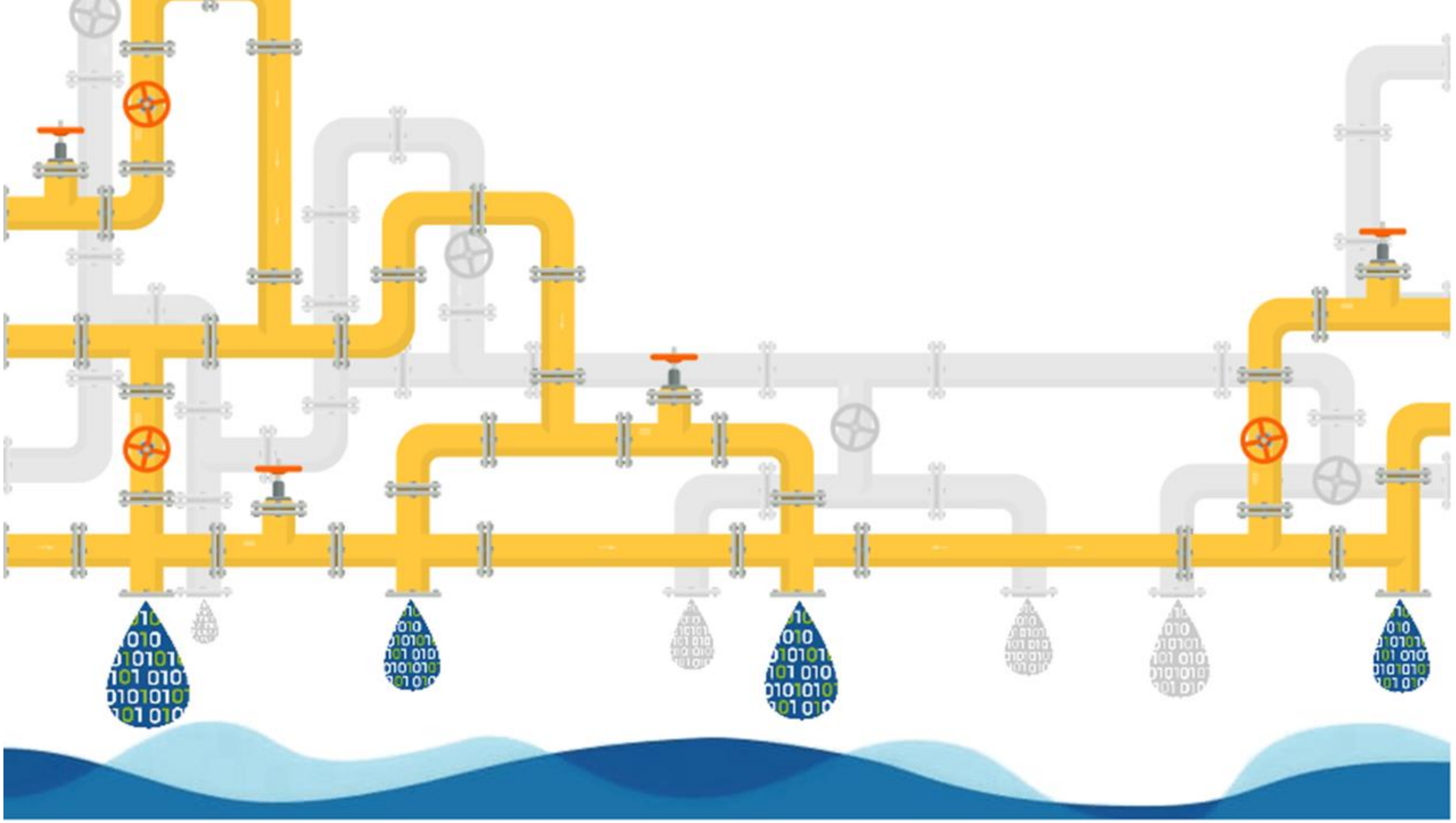




Feature Selection in der Data Science Pipeline am Beispiel von medizinischen Diagnosen

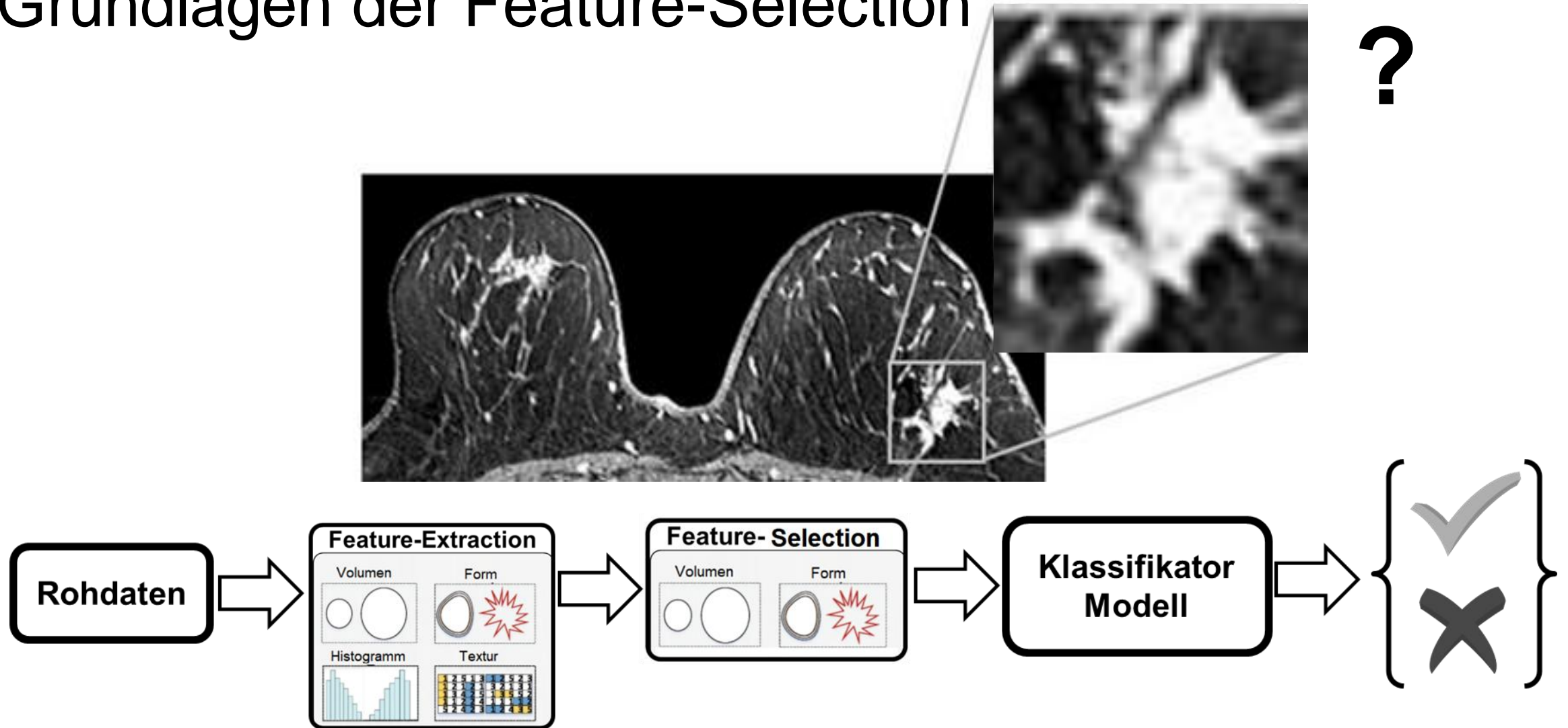
Inhalt

- **Data Science Pipelines**
- **Grundlagen** der Feature Selection
- **Methodik** der Feature Selection
- **Umsetzung in der medizinischen Diagnostik:** die Krebsklassifizierung
 - Anwendung der Methoden zur Feature-Selektion
 - Data-Engineering-Pipelines
 - Analyse und Auswertung
- **Fazit**



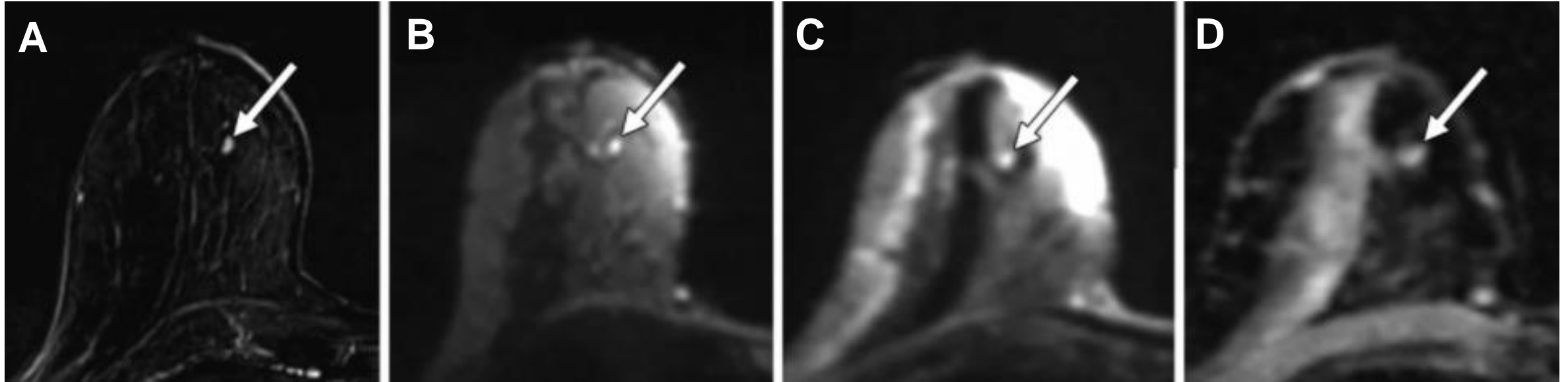
Data Science Pipelines

Grundlagen der Feature-Selection



Feature-Selection: Auswahl der wichtigsten / relevanten / informativen / erklärbaren / beschreibenden Features zur Verbesserung der Vorhersagequalität!

Grundlagen der Feature-Selection



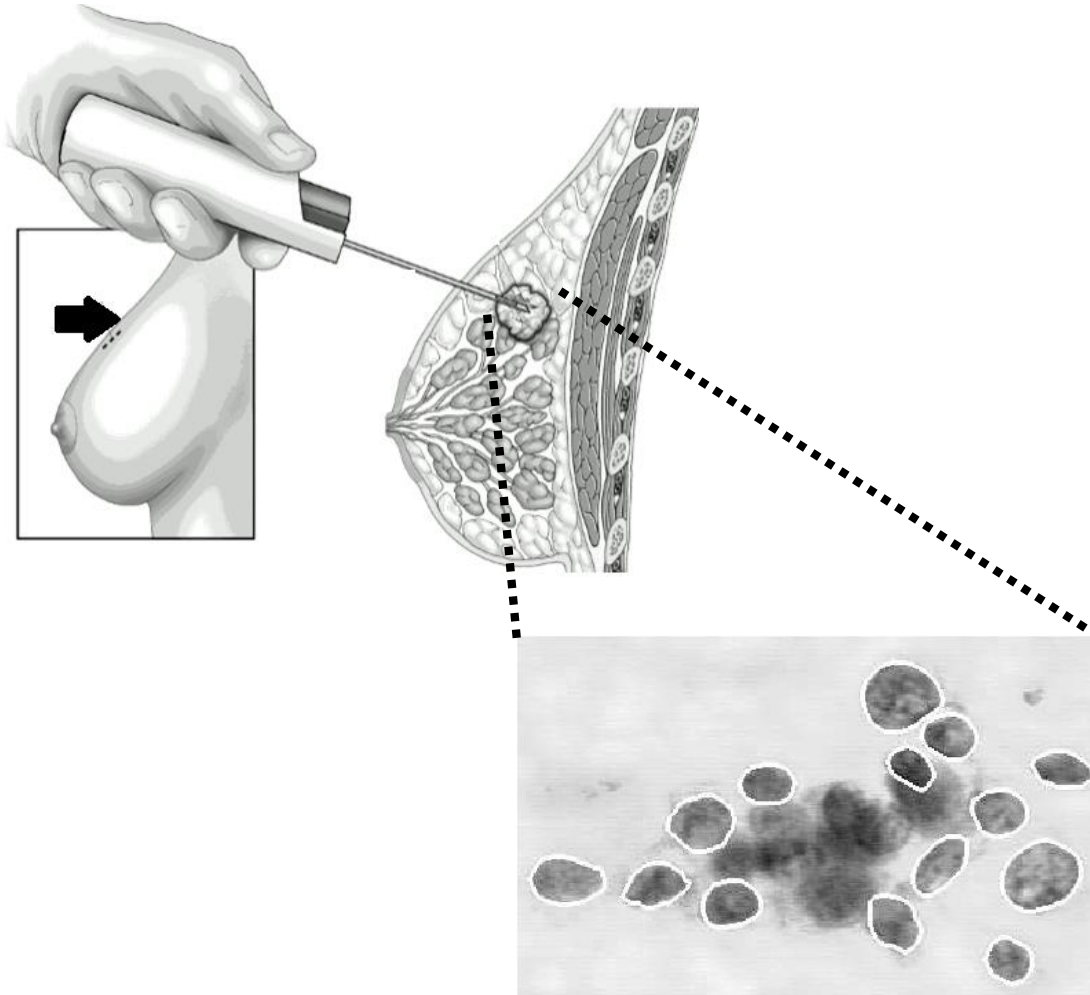
Methodik der Feature Selection

Methodik der Feature Selection

Methodik der Feature Selection

Umsetzung in der medizinischen Diagnostik

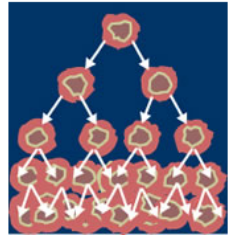
Brustkrebs-Diagnose



Breast Cancer Wisconsin (Diagnostic) Data Set

Download: [Data Folder](#) [Data Set Description](#)

Abstract: Diagnostic Wisconsin Breast Cancer Database



569 Patientinnen, 30 Features

10 Charakteristika des Brustmassenzellkerns wurden gemessen:

- Radius (Mittelwert aller Abstände vom Zentrum zu Punkten auf dem Perimeter)
- Textur (Standardabweichung der Grauskala-Werte)
- Umfang
- Fläche
- Glattheit (lokale Variation der Radiuslängen)
- Kompaktheit ($\text{Umfang}^2 / \text{Fläche} - 1,0$)
- Konkavität (Stärke der konkaven Teile der Kontur)
- Konkavitätspunkte (Anzahl der konkaven Teile der Kontur)
- Symmetrie
- Fraktale Dimension ("Küstenlinienapproximation" - 1)

Für jedes Feature werden 3 Maße angegeben:

- Kleinste
- Standardfehler
- Größte/"schlechteste"

Aufgabe: Die Brustmasse als **gut-** oder **bösartig** zu klassifizieren

[https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+\(diagnostic\)](https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic))

Umsetzung in der medizinischen Diagnostik

 χ^2

Demo-Code
auf
Github



```
▼ # Download des Krebs-Datensatzes
import seaborn as sns
from sklearn import preprocessing
(X, y) = load_breast_cancer(return_X_y=True, as_frame=True)

▼ # Chi-Quadrat Feature Selection
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2
# Ermitteln der besten k = 20 Features
chi_best = SelectKBest(chi2, k=20).fit(X, y)
mask_features = chi_best.get_support()
new_features = [] # die Liste mit die K bested Features
▼ for bool, feature in zip(mask_features, X.columns):
▼     if bool:
        new_features.append(feature)
X_new = chi_best.fit_transform(X, y)
# Überblick über die filtrierte Daten
▼ df = pd.DataFrame(data=np.array(list(zip(new_features, chi_best.scores_, chi_best.get_support()))),
                    index=pd.RangeIndex(start=0, stop=20, step=1),
                    columns=["Name", "Wert", "Entscheidung"])
```

Umsetzung in der medizinischen Diagnostik

minimum-Redundancy-Maximum-Relevance (mRMR)

Demo-Code
auf
Github



```
import pymrmr
rel_feat = pymrmr.mRMR(X, 'MID', 20)

X_new = X[X.columns.intersection(rel_feat)]

# Überblick über die filtrierte Daten↔
```

Feature Selection Ergebnisvergleich

Chi-Quadrat-Test

mRMR

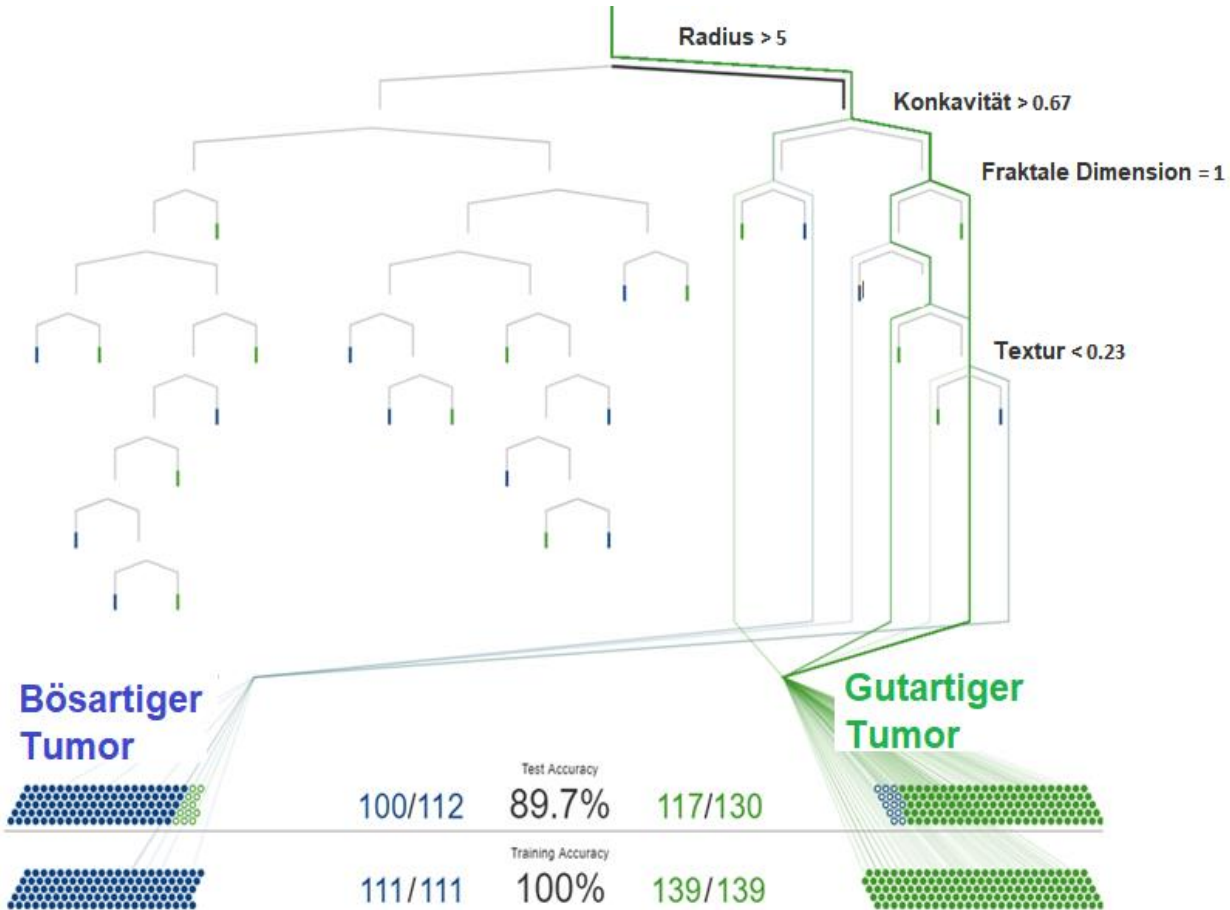
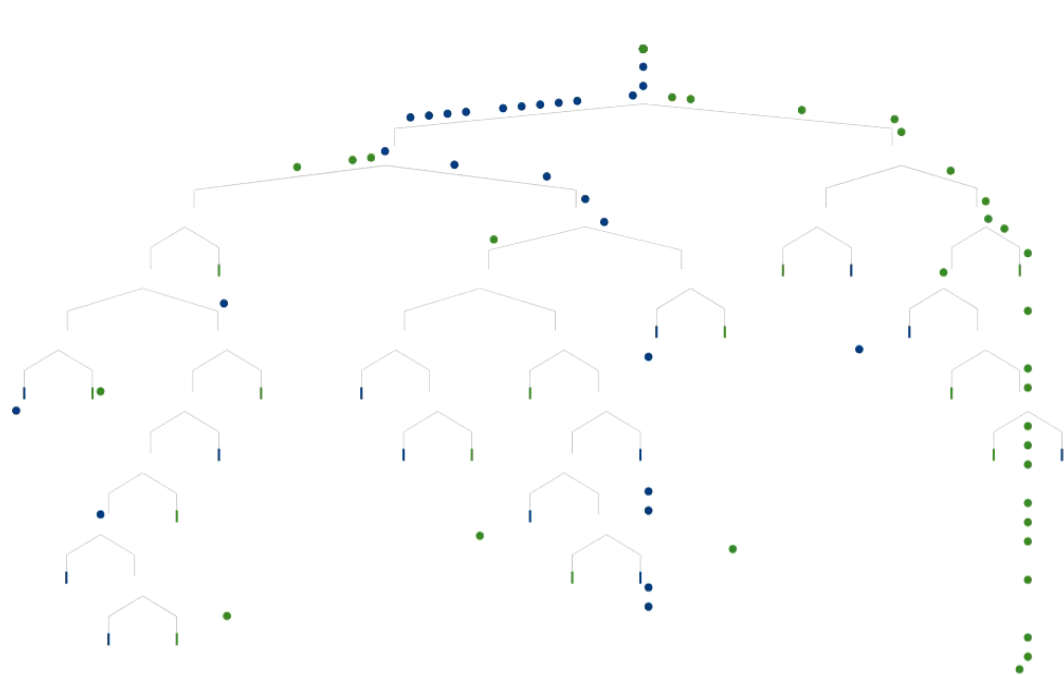
| | Name | Wert | Entscheidung |
|---|------------------|---------------------|--------------|
| 0 | mean radius | 266.1049171951787 | True |
| 1 | mean texture | 93.8975080986333 | True |
| 2 | mean perimeter | 2011.102863767906 | True |
| 3 | mean area | 53991.65592375089 | True |
| 4 | mean compactness | 0.14989926383938243 | False |
| 5 | mean concavity | 5.403075490732707 | True |

| | Name | Wert |
|---|-----------------|-------|
| 0 | mean area | 3.655 |
| 1 | worst area | 3.483 |
| 2 | mean perimeter | 3.314 |
| 3 | worst perimeter | 2.623 |
| 4 | worst radius | 2.228 |
| 5 | area error | 1.729 |

Umsetzung in der medizinischen Diagnostik

Klassifizierung Modell

Anwendung von Entscheidungsbäumen

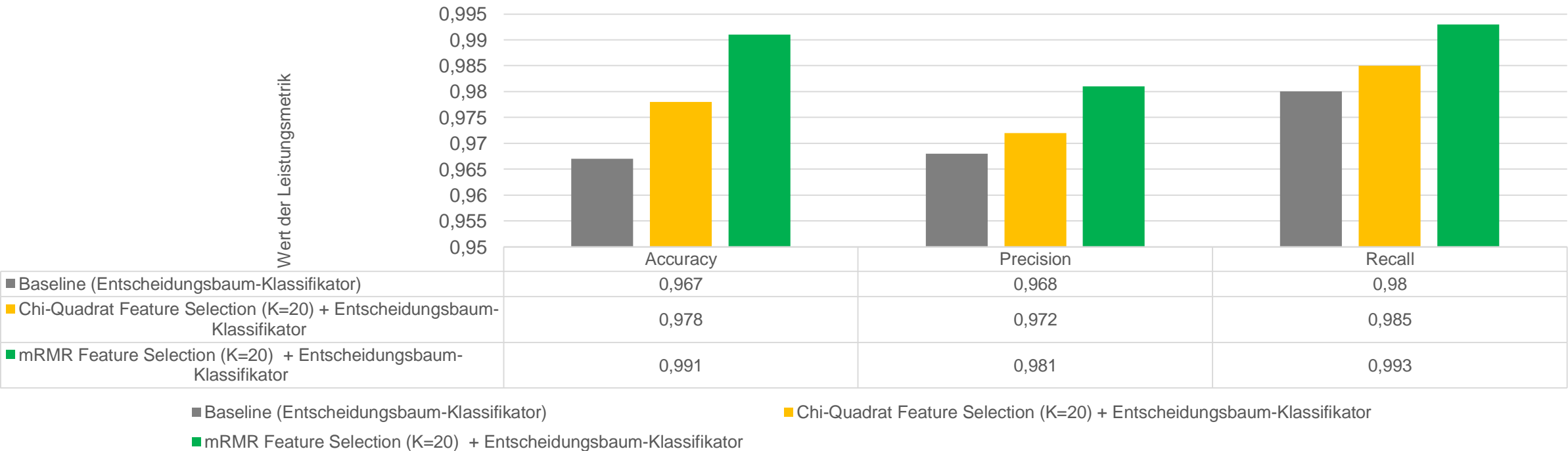


Umsetzung in der medizinischen Diagnostik

Data-Engineering-Pipelines Auswertung

- Daten-Skalierung + Baseline-Klassifikator (Entscheidungsbäumen)
- Daten-Skalierung + Chi-Quadrat-Test + Baseline-Klassifikator
- Daten-Skalierung + mRMR + Baseline-Klassifikator

Demo-Code
auf
Github



Fazit

Die Feature-Selection ist:

- nützlich, wenn wir die **Anzahl** der für die Verarbeitung benötigten **Ressourcen reduzieren** müssen, ohne wichtige oder **relevante Informationen** zu verlieren
- ein **wichtiger Schritt in der Data-Engineering-Pipeline**, bevor das Prädiktivmodell erstellt wird
- **domänen-** und **datenspezifisch**
- **unterstützt** eine bessere **Analyse** und **Interpretation** der Vorhersagen

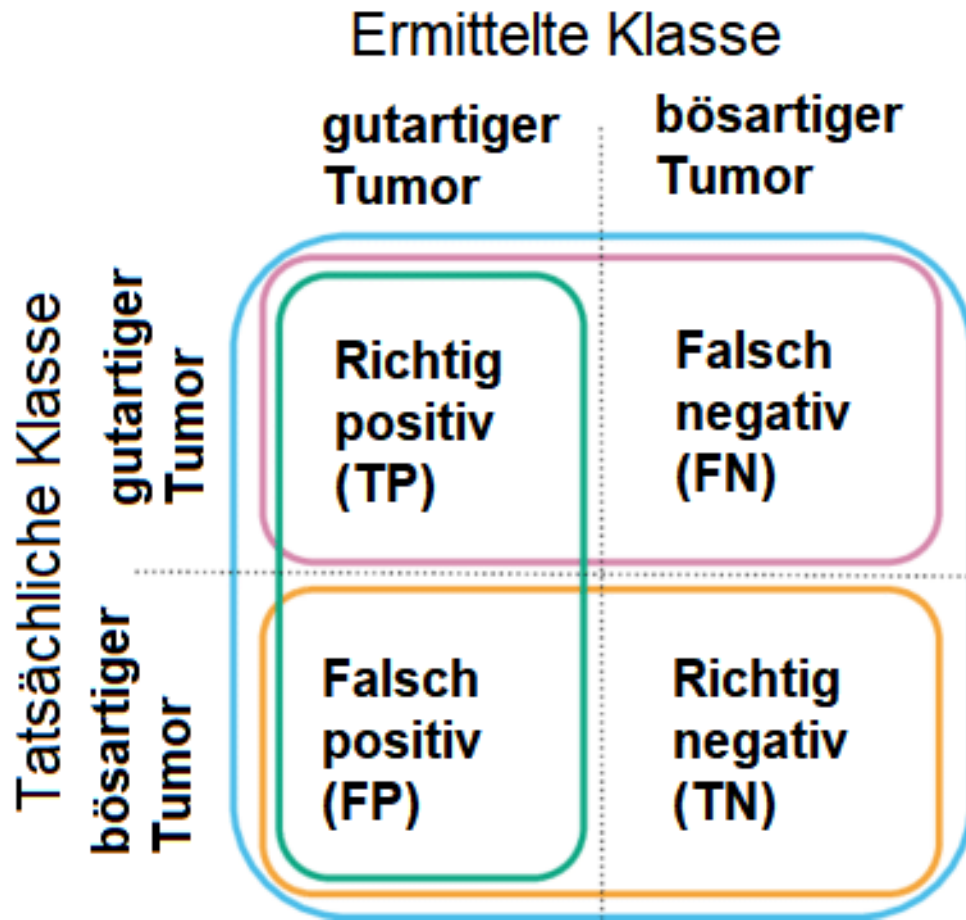



Feature Selection in der Data Science Pipeline


am Beispiel von medizinischen Diagnosen


Umsetzung in der medizinischen Diagnostik


Auswertung



 Accuracy = $\frac{TP + TN}{TP + TN + FP + FN}$

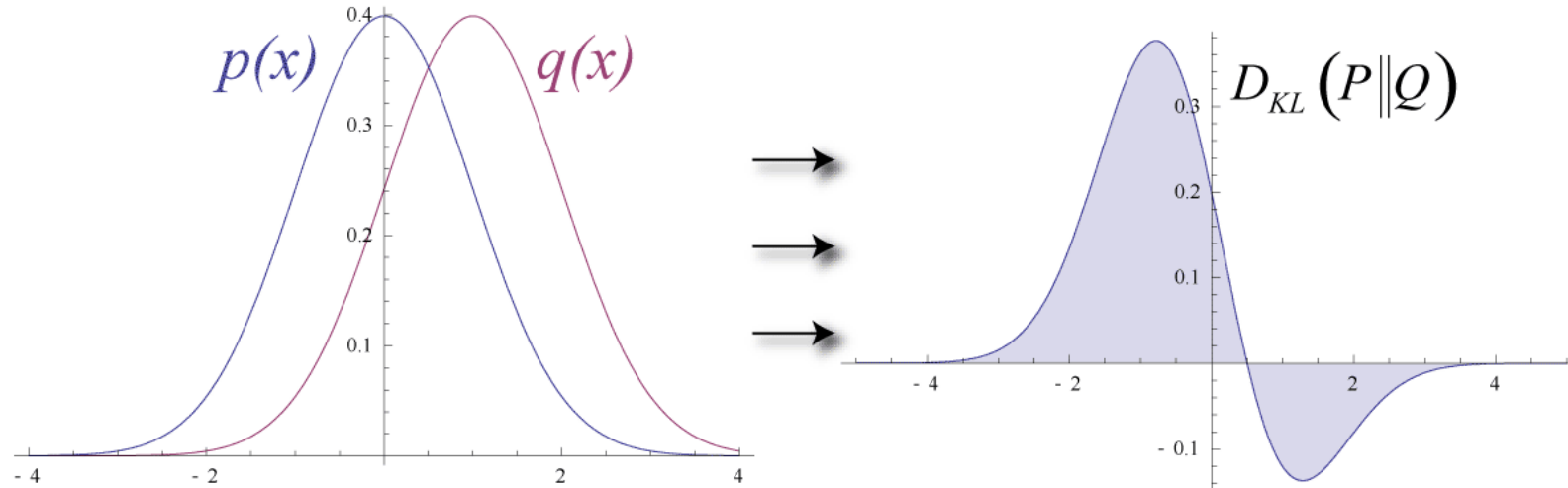
 Specificity = $\frac{TN}{TN + FP}$

 Precision = $\frac{TP}{TP + FP}$

 Recall = $\frac{TP}{TP + FN}$

Umsetzung in der medizinischen Diagnostik

Kullback-Leibler-Divergenz (*KL-Divergenz*) bezeichnen ein Maß für die Unterschiedlichkeit zweier Wahrscheinlichkeitsverteilungen.



Transinformation Definition über die Kullback-Leibler-Divergenz:

$$I(X; Y) = D(p(x, y) || p(x)p(y))$$