

Testverfahren Holdout und Kreuzvalidierung	Grundidee: Qualität Vorhersagen auf unbekannten Daten evaluieren Holdout: Die Daten werden in zwei Teile aufgeteilt: <ul style="list-style-type: none">- Trainingsmenge (80% Daten)- Testmenge (20% Daten) Kreuzvalidierung (z.B. 10-fach) <ul style="list-style-type: none">- Daten werden in 10 Teile (Folds) aufgeteilt- Modell wird 10 mal trainiert & getestet- Jedes Mal wird ein anderer Teil als Testmenge verwendet- Ergebnis = Durchschnitt Modellleistung aller Durchläufe														
Testverfahren Lernkurve	Eine Lernkurve zeigt, wie gut ein Modell wird, wenn es mehr Trainingsdaten bekommt. Hilft zu verstehen, ob ein Modell genug Trainingsdaten bekommt. <ul style="list-style-type: none">- Steigende Kurve: mehr Daten können helfen- Flache Kurve: Mehr Daten bringen kaum Verbesserung														
Testmetriken Genauigkeit / Fehlerrate / Accuracy (CA)	Genauigkeit: Zeigt wie viele Vorhersagen richtig waren <div>Genauigkeit = $\frac{\text{Anzahl korrekt klassifizierte Instanzen}}{N}$</div> Fehlerrate: Zeigt wie viele Vorhersagen falsch waren <div>Fehlerrate = $\frac{\text{Anzahl inkorrekt klassifizierte Instanzen}}{N} = 1 - \text{Genauigkeit}$</div> <p>Wenn die Genauigkeit hoch ist, ist die Fehlerrate niedrig. N ist die Grösse der Testmenge</p>														
Genauigkeit: immer eine gute Metrik?	Genauigkeit nicht immer die beste Metrik weil: <ul style="list-style-type: none">- Wenn eine Klasse viel häufiger vorkommt als die andere, kann ein Modell immer «männlich» vorhersagen und hat trotzdem hohe Genauigkeit- Das Modell ist schlecht, wenn es die seltenere Klasse «weiblich» fast nie richtig vorhersagt. Genauigkeit alleine sagt also nicht aus, ob das Modell gut zwischen Klassen unterscheidet.														
Confusion Matrix	Zeigt, wie gut ein Modell zwischen richtigen und falschen Vorhersagen unterscheidet <ul style="list-style-type: none">- TP (True Positive): Richtig positiv, Das Modell sagt «JA» und es ist auch wirklich «JA»- FN(False Negative): Falsch negativ: Das Modell sagt «Nein», aber es ist eigentlich «Ja»- FP (False Positive): Falsch positiv: Das Modell sagt «JA» aber es ist eigentlich «Nein»- TN (True Negative): Richtig negativ: Das Modell sagt «Nein» und es ist auch wirklich «Nein» <table><tr><td></td><td colspan="3">Vorhergesagte Klasse</td></tr><tr><td rowspan="3">Echte Klasse</td><td></td><td>Ja</td><td>Nein</td></tr><tr><td>Ja</td><td>A (TP)</td><td>B (FN)</td></tr><tr><td>Nein</td><td>C (FP)</td><td>D (TN)</td></tr></table>		Vorhergesagte Klasse			Echte Klasse		Ja	Nein	Ja	A (TP)	B (FN)	Nein	C (FP)	D (TN)
	Vorhergesagte Klasse														
Echte Klasse		Ja	Nein												
	Ja	A (TP)	B (FN)												
	Nein	C (FP)	D (TN)												

<p>Wichtige Metriken</p> <p>Precision</p> <p>Recall</p> <p>F-Measure (F1-Score)</p>	<p>Precision (Genauigkeit): zeigt wie zuverlässig die positiven Vorhersagen sind. («Von allen Ja-Vorhersagen, wie viele sind wirklich Ja?»)</p> <p>Beispiel: Von 10 vorhergesagten Betrugsfällen (Ja) waren 8 wirklich Betrug. Precision = $8/10 = 0.8$ (80%).</p> $\text{Precision} = \frac{a}{a + c}$ <p>Recall (Empfindlichkeit): zeigt wie viele der tatsächlichen positiven Fälle auch als positiv erkannt wurden. («Von allen echten Ja-Fällen, wie viele wurden richtig erkannt?»)</p> <p>Beispiel: Es gibt 20 Betrugsfälle insgesamt, und das Modell hat 15 richtig erkannt. Recall = $15/20 = 0.75$ (75%).</p> $\text{Recall} = \frac{a}{a + b}$ <p>F-Measure (F1-Score): kombiniert Precision & Recall zu einer Zahl. Misst das Gleichgewicht zwischen Recall und Precision. Gut wenn beide Werte ähnlich sind. 😊</p> $\text{F-Measure} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$ <p>*Confusion Matrix:</p> <ul style="list-style-type: none"> Precision ist wichtig, wenn falsche Alarme (FP) ein Problem sind (z.B. bei Betrugserkennung). Recall ist wichtig, wenn es darum geht, keine positiven Fälle zu übersehen (z.B. bei Krankheitsdiagnosen). F-Measure gibt dir eine ausgeglichene Bewertung zwischen beiden.
<p>Area Under Curve (Roc)</p>	<p>AUC: Wie gut Modell zwischen positiven und negativen Klassen unterscheidet. Sie basiert auf der ROC Kurve. (Zahl: Fläche)</p> <p>ROC: zeigt Verhältnis von True Positive Rate und False Positive Rate (Bild: Kurve)</p> <p>Grundidee:</p> <ol style="list-style-type: none"> Modelle geben oft Score aus (Wahrscheinlichkeit) Wir sortieren Vorhersagen nach diesem Score von hoch nach niedrig AUC misst wie gut positive Beispiele oben und die negativen Beispiele unten in dieser Liste stehen <p>Wertbereich:</p> <ul style="list-style-type: none"> AUC = 0.5 (zufällige Entscheidung) «schlecht» (0.5 unterscheidet nicht besser als ein Münzwurf) AUC = 1 (Perfekte Entscheidung)
<p>Kostenbasierte Evaluation</p> <p>*Einfachsten zum erklären für Stakeholder</p>	<p>Die Vorhersage des Modells können Gewinne oder Verluste verursachen. Deshalb ist es nicht nur Wichtig die Genauigkeit zu betrachten, sondern auch die Kosten, die durch richtige oder falsche Vorhersagen entstehen.</p> <p>Kostenmatrix: zeigt finanzielle Auswirkungen je nach Vorhersage und Realität</p> <ul style="list-style-type: none"> Es reicht nicht, nur die Genauigkeit des Modells anzuschauen. Man muss auch die finanziellen Auswirkungen (Kosten/Gewinn) berücksichtigen. Die Kostenmatrix hilft dabei zu sehen, ob die Modellvorhersagen wirtschaftlich sinnvoll sind.

Kostenmatrizen formulieren	<ul style="list-style-type: none">• Maschinenlernmodelle treffen Vorhersagen (z.B. Kredit genehmigen oder nicht).• Diese Vorhersagen können entweder richtig oder falsch sein.• Je nach Szenario (Bank oder Versicherung) entstehen dabei unterschiedliche Kosten.• Die Kostenmatrix hilft zu bewerten, wie teuer Fehler sind. <p>Kostenmatrizen helfen, die wirtschaftlichen Konsequenzen von richtigen und falschen Vorhersagen zu verstehen.</p>
-----------------------------------	---