Praktikum Entscheidungsbaum

In diesem Praktikum lernen Sie, wie man klassische Machine-Learning-Algorithmen auf reale Unternehmensdaten anwendet, ohne selbst programmieren zu müssen. Nach Abschluss dieses Praktikums haben Sie eine Analyse-Infrastruktur aufgebaut, die folgende Schritte umfasst:

1. *Exploration & Segmentierung*: Daten werden ohne Zielvorgabe strukturiert, um verborgene Muster und Kundensegmente zu identifizieren.
2. *Modellierung*: Ein Algorithmus lernt aus historischen Daten die Regeln für Kündigungen.
3. *Inferenz & Anwendung*: Das trainierte Modell wird auf neue, unbekannte Kunden angewendet, um deren Kündigungswahrscheinlichkeit individuell zu prognostizieren.

Um diesen Prozess abzubilden, nutzen Sie einen modernen Software-Stack, wie er in Fachabteilungen für Business Analytics und Data Science zum Einsatz kommt:

- *Orange Data Mining*: Ein visuelles Open-Source-Werkzeug für Data Science und Machine Learning. Es ermöglicht das intuitive Zusammenklicken von Analyse-Workflows mittels Daten-Pipelines.
- *n8n*: Unsere bekannte Prozess-Engine. Wir nutzen sie in diesem Praktikum am Ende als Validierungswerkzeug, um aus Ihren individuellen Prognoseergebnissen einen verifizierbaren Abgabecode zu generieren.

Der Prozess im Detail

Sie schlüpfen in die Rolle eines Business Analysten bei einem Telekommunikationsunternehmen. Sie arbeiten mit dem bekannten *Telco Customer Churn* Datensatz. Ihr Ziel ist es herauszufinden, welche Kundengruppen existieren, warum bestimmte Kunden Verträge kündigen und wie hoch die Wahrscheinlichkeit ist, dass eine Liste von Neukunden das Unternehmen verlässt.

Vorbereitung

Bevor Sie mit der Analyse starten, bereiten Sie die notwendige Software und die Daten vor:

1. Laden Sie *Orange Data Mining* für Ihr Betriebssystem herunter und installieren Sie es: https://orangedatamining.com/download/
2. Laden Sie die Datei `lab-02-telco-neue-kunden.csv` aus von der Webseite des Moduls herunter. Diese Datei enthält die Daten von 10 neuen Kunden (Kunden-ID 0 bis 9), deren Verhalten wir vorhersagen wollen. Speichern Sie diese in Ihrem Projektordner `praktikum-is`.
3. Starten Sie Orange und wählen Sie **New**, um einen leeren Arbeitsbereich zu öffnen.

Schritt 1: Historische Daten laden

In Orange arbeiten wir mit *Widgets* (den runden Symbolen am linken Rand), die über Linien miteinander verknüpft werden.

1. Klicken Sie links in der Kategorie *Data* auf das Widget **Datasets** und ziehen Sie es auf die Arbeitsfläche.
2. Öffnen Sie das Widget mit einem Doppelklick.
3. Geben Sie in das Suchfeld `churn` ein und wählen Sie aus der Liste den Datensatz **Telco Customer Churn** aus. Orange lädt die Daten nun automatisch im Hintergrund. Schließen Sie das Fenster.

Schritt 2: Unüberwachtes Lernen (Kundensegmente finden)

Wir wollen zunächst herausfinden, welche Kundengruppen sich automatisch bilden, wenn man die Vertragslaufzeit (*tenure*) und die monatlichen Kosten (*MonthlyCharges*) betrachtet.

1. Ziehen Sie aus der Kategorie *Unsupervised* das Widget **k-Means** auf die Arbeitsfläche.
2. Verbinden Sie das *Datasets*-Widget per Drag & Drop (Kabel ziehen) mit dem *k-Means*-Widget.
3. Öffnen Sie **k-Means** mit einem Doppelklick und stellen Sie die Anzahl der Cluster (*Number of clusters*) auf **3**. Schließen Sie das Fenster.
4. Ziehen Sie aus der Kategorie *Visualize* ein **Scatter Plot**-Widget auf die Arbeitsfläche und verbinden Sie das *k-Means*-Widget damit.
5. Öffnen Sie den **Scatter Plot**:
   - Stellen Sie die X-Achse auf `tenure`.
   - Stellen Sie die Y-Achse auf `MonthlyCharges`.
   - Wählen Sie bei *Color* den Wert `Cluster` aus.

Erfolgskontrolle & Analyse des "Warum"

Um zu verstehen, warum die KI diese Gruppen gebildet hat, ziehen Sie aus *Visualize* ein **Box Plot**-Widget auf die Arbeitsfläche und verbinden Sie es ebenfalls mit **k-Means**.

- Öffnen Sie den **Box Plot** und stellen Sie unten das Feld *Subgroups* auf **Cluster**.
- Klicken Sie links in der Liste nacheinander auf `MonthlyCharges` und `tenure`.
- *Erkenntnis*: Sie sehen nun visuell die Charakteristika der Gruppen (z.B. Cluster 1 = "Treue Kunden mit hohen Kosten", Cluster 2 = "Sparfüchse" etc.).

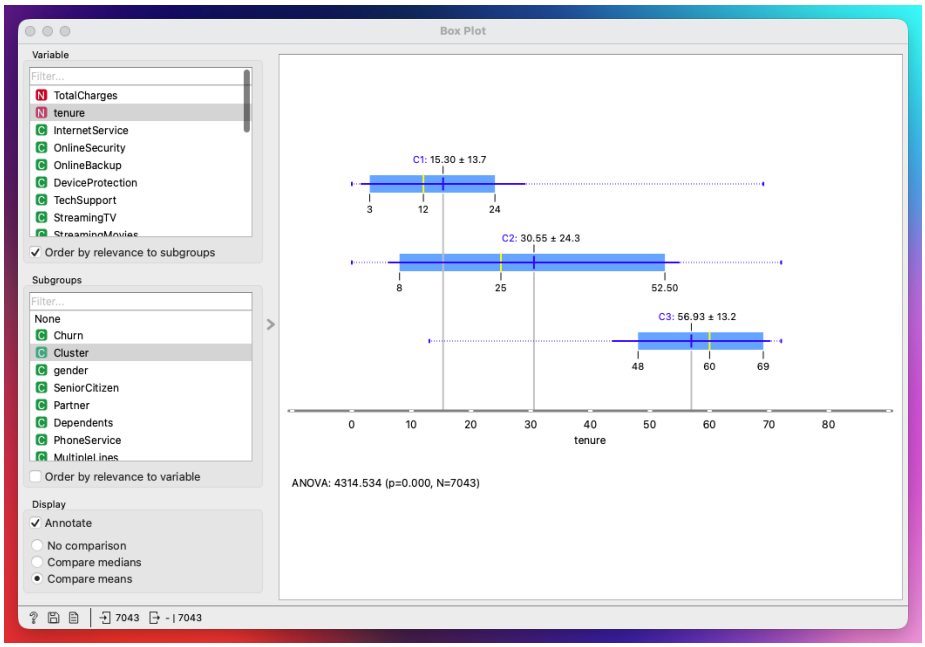

Figure 1: Clusteranalyse im Box Plot

Schritt 3: Überwachtes Lernen, Visualisierung & Modell-Optimierung

Nun trainieren wir einen Entscheidungsbaum, um konkrete Kündigungsregeln zu lernen. Standardmäßig baut die KI den Baum so tief, bis alle Daten perfekt aufgeteilt sind. Dies führt jedoch häufig zu *Overfitting*. Wir analysieren den Aufbau des Baumes visuell und experimentieren mit der Baumtiefe, um das optimal konfigurierte Modell zu finden.

1. Ziehen Sie das Widget **Tree** (Kategorie *Model*) auf die Arbeitsfläche und verbinden Sie es direkt mit dem **Datasets**-Widget.
2. Ziehen Sie das Widget **Test and Score** (Kategorie *Evaluate*) auf die Arbeitsfläche. Verbinden Sie das **Datasets**-Widget mit **Test and Score** (*Data*) und das **Tree**-Widget ebenfalls mit **Test and Score** (*Learner*).
3. Ziehen Sie das Widget **Confusion Matrix** (Kategorie *Evaluate*) auf die Arbeitsfläche und verbinden Sie **Test and Score** damit.
4. Ziehen Sie das Widget **Tree Viewer** (Kategorie *Visualize*) auf die Arbeitsfläche und verbinden Sie das **Tree**-Widget direkt mit dem **Tree Viewer**.

Experimentier- & Optimierungsphase

Öffnen Sie parallel die Fenster des **Tree**-Widgets, des **Tree Viewers** und der **Confusion Matrix**, sodass Sie alle drei Bereiche nebeneinander auf Ihrem

Bildschirm sehen können.

1. Schauen Sie in den **Tree Viewer**. Ohne Einschränkung baut die KI einen riesigen, unübersichtlichen Baum auf.
2. Öffnen Sie das Widget **Tree** mit einem Doppelklick. Hier können Sie die maximale Tiefe des Baumes begrenzen:

- Aktivieren Sie das Kontrollkästchen **Limit maximum depth to:**.
- Stellen Sie den Wert nacheinander auf **2**, **3**, **5** und **10**.

1. Beobachten Sie die Auswirkungen:

- Sehen Sie im **Tree Viewer**, wie sich die Verästelungen und Regeln des Baumes dynamisch vereinfachen oder verkomplizieren.
- Beobachten Sie im Fenster **Test and Score**, wie sich der Wert **CA** (Classification Accuracy / Gesamtgenauigkeit) verändert.
- Analysieren Sie in der **Confusion Matrix**, wie viele Abwanderer korrekt als Abwanderer erkannt wurden (*True Positives*) und wie viele treue Kunden fälschlicherweise als Abwanderer eingestuft wurden (*False Positives*).

Ihre Optimierungs-Aufgabe

Finden Sie durch systematisches Ausprobieren heraus, bei welcher maximalen Baumtiefe das Modell die **höchste Gesamtgenauigkeit (CA)** in *Test and Score* erzielt.

Wichtig für die Abgabe: Stellen Sie Ihr **Tree**-Widget fest auf diese optimale Baumtiefe ein (z.B. *5*). Lassen Sie diese Einstellung für den nächsten Schritt aktiv. Notieren Sie sich diese gefundene optimale Tiefe (als reine Ganzzahl, z.B. `5`).

Schritt 4: Inferenz & Anwendung (Predictions)

Nachdem Sie das beste Modell konfiguriert haben, wenden wir diese KI-Logik nun auf unsere Neukunden an, um deren Kündigungswahrscheinlichkeit vorherzusagen.

1. Ziehen Sie das Widget **File** (Kategorie *Data*) auf die Arbeitsfläche. Doppelklicken Sie darauf und laden Sie Ihre aus ILIAS heruntergeladene Datei `telco_neue_kunden.csv`.
2. Ziehen Sie das Widget **Predictions** (Kategorie *Evaluate*) auf die Arbeitsfläche.
3. Verbinden Sie nun:
   - Das **Tree**-Widget mit **Predictions** (Das optimierte Modell liefert die Logik).
   - Das **File**-Widget mit **Predictions** (Die neuen Kundendaten, für die wir eine Vorhersage wollen).
4. Öffnen Sie das Widget **Predictions** mit einem Doppelklick. Sie sehen eine Tabelle mit den 10 neuen Kunden (IDs 0 bis 9).

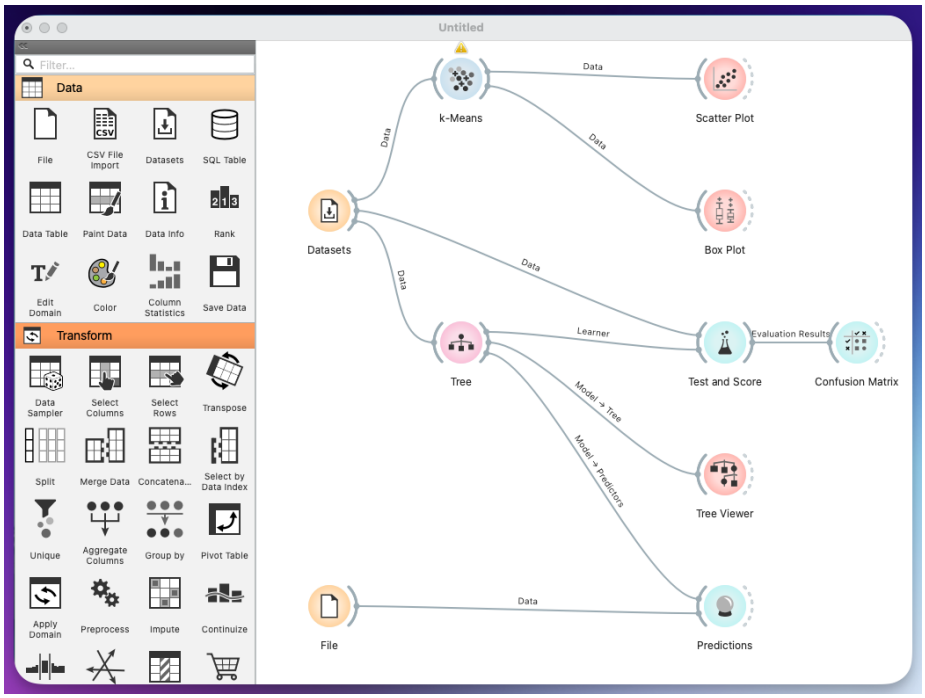

Figure 2: Vollständiger Data-Mining-Workflow inklusive Optimierung und Inferenz

5. Aktivieren Sie auf der linken Seite des Fensters das Kontrollkästchen **Show probabilities for** und wählen Sie im dazugehörigen Dropdown-Menü den Wert *Classes knwon to the model* aus.
6. In der Tabelle erscheint nun in der Spalte Tree die Wahrscheinlichkeit, dass ein Kunde kündigt oder nicht.

   Hintergrundwissen: Woher kommt diese Wahrscheinlichkeit? Wenn ein neuer Kunde die Weichen des Entscheidungsbaums durchläuft, landet er an einem bestimmten Endknoten (Blattknoten). Die Wahrscheinlichkeit entspricht exakt der historischen Klassenverteilung in diesem spezifischen Endknoten der Trainingsdaten. Wenn in diesem Knoten historisch gesehen beispielsweise 85 % der Kunden gekündigt haben, erhält der neue Kunde dort eine Kündigungswahrscheinlichkeit von genau `0.85`.

Wichtig für die individuelle Abgabe: Schauen Sie auf die **letzte Ziffer Ihrer Benutzernamens**. Diese Ziffer bestimmt, welchen Kunden Sie auswerten müssen.

- Wenn die letzte Ziffer Ihres Benutzernamens die *3* ist, lesen Sie den ersten Wert für die Zeile mit der **Kunden_ID 3** in der Spalte **Tree** ab.
- Notieren Sie sich diesen Wert exakt (z.B. `0.85` oder `0.12`).

Schritt 5: Abgabe via n8n

Um Ihr Ergebnis zu verifizieren, nutzen wir n8n zur Erstellung des finalen Abgabecodes.

1. Stellen Sie sicher, dass Ihre Docker-Umgebung läuft und öffnen Sie n8n unter http://localhost:5678.
2. Erstellen Sie einen neuen Workflow und fügen Sie einen **Edit Fields**-Knoten hinzu.
3. Fügen Sie über *Add Field* (Typ: *String*) ein neues Feld mit dem exakten Namen `ILIAS_ABGABE_CODE` hinzu.
4. Klicken Sie auf das Zahnrad-Symbol neben dem Feld und wählen Sie **Add Expression**. Kopieren Sie folgende Formel exakt in das Feld:

   ```
   {{ ( "IhrAnmeldename" + "IS-PRAKTIKUM-2026" +
         "OptimaleTiefe" + "IhrAbgelesenerWert" )
         .hash("sha256").substring(0, 15).toUpperCase() }}
   ```

5. Anpassung der Expression:
   - Ersetzen Sie `IhrAnmeldename` durch Ihren persönlichen ILIAS-Loginnamen (z.B. `ab1234x`).
   - Ersetzen Sie `OptimaleTiefe` durch die Zahl Ihrer ermittelten besten Baumtiefe aus Schritt 3 (z.B. `5`).

- Ersetzen Sie `IhrAbgelesenerWert` durch die exakte Zahl aus dem Predictions-Widget (z.B. `0.45`). Lassen Sie die Anführungszeichen bestehen.

6. Führen Sie den Knoten aus.
7. Kopieren Sie den resultierenden 15-stelligen Code aus der Spalte `ILIAS_ABGABE_CODE` und tragen Sie diesen gemeinsam mit Ihrer Matrikelnummer im ILIAS-Test ein.