1

1. Zu Beginn müssen die Daten vorberitet werden; das Bereinigen von fehlenden Werten, das Kodieren kategorialer Variablen und ggf. das Normalisieren von Merkmalen.

2. Für jeden potenziellen Split muss der Klassifizierungsfehler berechnet werden, diese gibt an, wie gut der Split die Daten in homogene Gruppen aufteilt.

3. Für jeden Knoten muss der beste Split basierend auf dem niedrigsten Klassifizierungsfehler gefunden werden. Dies erfordert das Durchlaufen aller Merkmale und ihrer möglichen Split-Punkte.

4. Nachdem der beste Split gefunden wurde, muss der Prozess rekursiv auf die resultierenden Untergruppen angewendet werden, bis ein Abbruchkriterium erreicht ist (z.B. maximale Baumtiefe, minimale Anzahl von Datenpunkten in einem Knoten).

5. Um Überanpassung zu vermeiden, sollte der Baum nach der Konstruktion beschnitten werden. Dies kann durch das Entfernen von Knoten erfolgen, die wenig zur Vorhersagegenauigkeit beitragen.

   Die Laufzeit ist O(NMlog(N)), wobei N die Anzahl der Datenpunkte und M die Anzahl der Merkmale ist. Dies berücksichtigt, dass für jeden Knoten die Daten sortiert werden müssen (was O(N*log(N)) Zeit kostet), und dies muss für jedes Merkmal M durchgeführt werden.

2

Die hohe Precision für "abgelehnt" sagt aus, dass vom Pool aus den abgelehnten Bewerbern 95% richtig klassifiziert wurden.
Der Recall bei "gut" bedeutet, dass 85% der "guten" Bewerber richtig klassifiziert wurde.
Es neigt dazu, sicherzustellen, dass die abgelehnten Kandidaten wirklich ungeeignet sind, wobei es einige gute Kandidaten möglicherweise übersieht.
Das Frauen nicht in "abgelehnt" sortiert wurden, sagt nicht aus, dass das Modell Männer unfair behandelt. Allerdings deutet es auf Verzerrung hin. Es wäre essenziell zu wissen, wieviele Männer und Frauen anteilig unter den Bewerbern sind. Zudem könnte man den F1-Score geschlechtsspezifisch berechnen, um zu prüfen, ob das Modell für Männer und Frauen unterschiedliche Leistungen zeigt.

3

Naive Bayes Vorwissen könnte in die Wahrscheinlichkeitsverteilung integriert werden, indem man die initialen Wahrscheinlichkeiten künstlich abändert. Beispielsweise könnte man die Wahrscheinlichkeit, dass bei Glatteis eher ein

Taxi genommen wird, trotz der Seltenheit dieses Ereignisses in den Daten, erhöhen.

Decision Trees In Decision Trees könnte man Vorwissen verwenden, um die Kriterien für das Teilen von Knoten zu beeinflussen. Zum Beispiel könnte man bei der Entscheidung, ob ein Taxi oder der ÖPNV genommen wird, Glatteis als ein entscheidendes Feature priorisieren, auch wenn es in den Daten selten vorkommt.

kNN Bei kNN könnte das Vorwissen dazu verwendet werden, die Distanzfunktion anzupassen. Beispielsweise könnte man das Gewicht von seltenen, aber wichtigen Ereignissen wie Glatteis erhöhen, sodass Instanzen mit diesen Eigenschaften stärker in die Klassifikation einfließen.

Herausforderungen

1. Datenverzerrung: Zu starkes Verlassen auf Vorwissen kann die die Ergebnisse verzerren

2. Komplexität: Integration von Vorwissen kann die Komplexität der Modellierung erhöhen

3. Generalisierung: Risiko, dass Modelle, welche stark auf spezifischem Vorwissen basieren, schlecheter auf neue, abweichende Daten generalisieren.