Instanzenbasiertes Lernen: Übersicht

- k-Nearest Neighbor
- Lokal gewichtete Regression
- Fallbasiertes Schließen
- Lernen: Lazy oder Eager

Instanzenbasiertes Lernen

Idee: speichere einfach alle Trainingsbeispiele $\langle x_i, f(x_i) \rangle$

Nearest Neighbor:

- Gegeben eine Instanz x_q, suche Trainingsbeispiel x_n, das am nächsten an x_q liegt und setze $\hat{f}(x_q) \leftarrow f(x_n)$

k-Nearest Neighbor: Gegeben x_q,

Diskreter Fall wähle Mehrheit der Werte der k nächsten Nachbarn

Reellwertiger Fall wähle Mittelwerte der Werte der k nächsten Nachbarn

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} f(x_i)}{k}$$

Wann ist Nearest Neighbor geeignet?

- Instanzen bilden Punkte im $\Re^n$
- Weniger als 20 Attribute pro Instanz
- jede Menge Traingsdaten

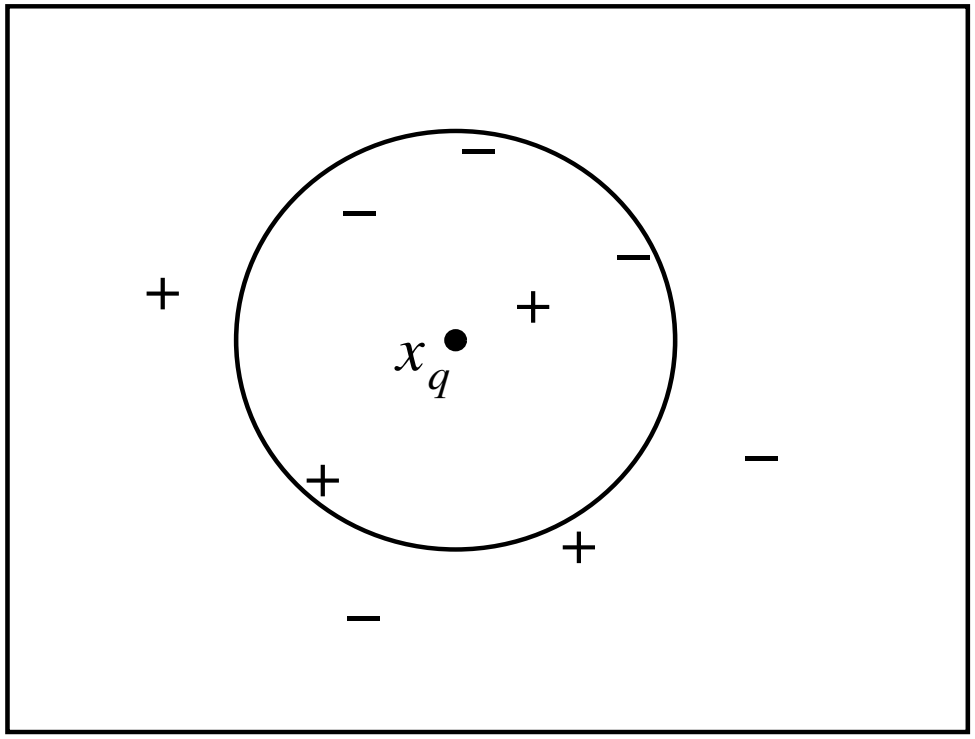
Vorteile:

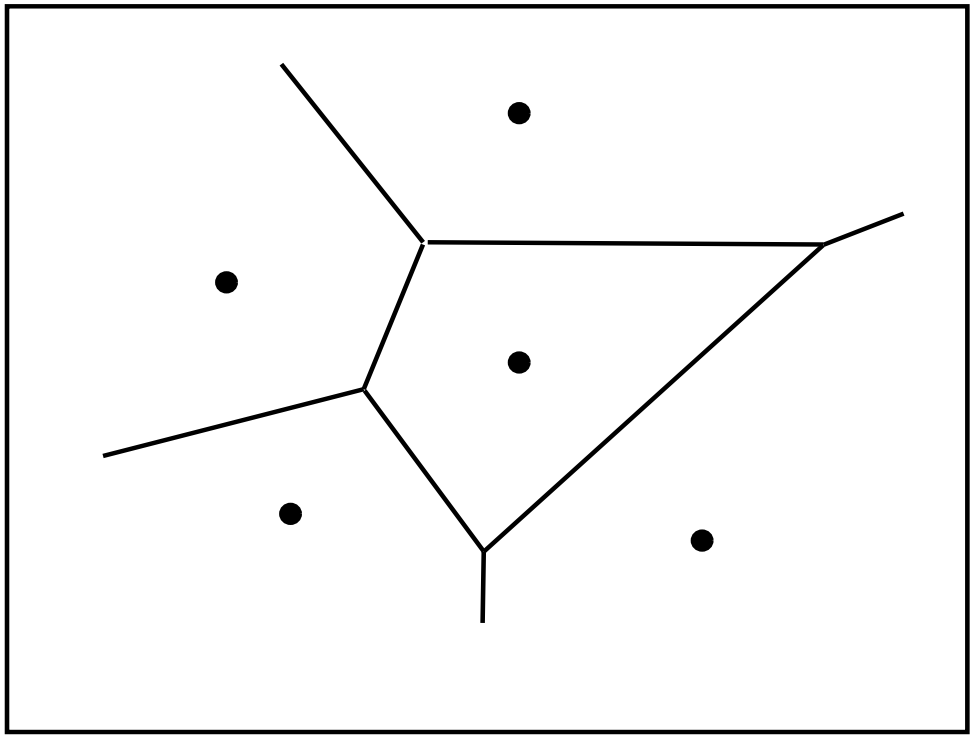
- Training ist sehr schnell
- auch komplexeste Zielfunktionen lernbar
- kein Generalisierungsmechanismus nötig
- Kein Informationsverlust

Nachteile:

- Abstandsmaß muß angegeben werden
- Langsam zur Anwendungszeit
- sehr sensitiv gegenüber irrelevanten Attributen
- nicht vom Menschen interpretierbar/kommunizierbar

Voronoi-Diagramm





Verhalten im Limes

$p(x)$: WK, daß Instanz x mit 1 (gegenüber 0) bewertet wird

Nearest Neighbor:

- Wenn Zahl der Traingsbeispiele $\longrightarrow \infty$ ergibt sich Gibbs Algorithmus

  Gibbs: mit WK $p(x)$ sage 1 voraus, sonst 0

k-Nearest neighbor:

- Wenn Zahl der Traingsbeispiele $\longrightarrow \infty$ und k groß genug ist, dann wird Bayes'sche Optimalklassifikation angenähert

  Bayes'sche Optimalklassifikation: wenn $p(x) > .5$ dann sage 1 voraus, sonst 0

Bem.: Gibbs hat höchstens doppelten erwarteten Fehler wie Bayes'sche Optimalklassifikation

Abstandsgewichtes k-NN

Möchten möglicherweise nähere Nachbarn stärker gewichten

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} w_i f(x_i)}{\sum_{i=1}^{k} w_i}$$

wobei

$$w_i \equiv \frac{1}{d(x_q, x_i)^2}$$

und $d(x_q, x_i)$ ist Abstand von x_q und x_i

Jetzt können *alle* Trainingsbeispiele (statt bisher k) einbezogen werden

$\longrightarrow$ Shepard's Methode

Anmerkungen zur Dimensionalität

Beispiel: Instanzen durch 20 Attribute beschrieben, aber nur 2 davon sind relevant

Fluch der Dimensionalität: Nearest Neighbor wird durch hochdimensionale Räume in die Irre geführt

Ansatz:

- Dehne $j$te Achse durch Gewicht z_j, wobei $z_1, \ldots, z_n$ so gewählt werden, daß erwarteter Fehler minimiert wird
- Benutze Cross-Validation zur automatischen Bestimmung von $z_1, \ldots, z_n$
- ($z_j = 0$ eliminiert diese Dimension vollständig)

Lokal gewichtete Regression

k-NN bildet lokale Approximation für f zu jedem Punkt x_q

Warum Approximation $\hat{f}(x)$ für Region um x_q nicht explizit angeben?

- Passe lineare (quadratische, . . .) Fkt. den k nächsten Nachbarn an
- Resultiert in "stückweiser Annäherung" an f

Möglichkeiten der zu minimierenden Zielfehlern:

- Quadratischer Fehler über k nächsten Nachbarn

$$E_1(x_q) \equiv \frac{1}{2} \sum_{x \in \text{ den } k \text{ nächsten Nachbarn von } x_q} (f(x) - \hat{f}(x))^2$$

- Abstandsgewichteter Quadratischer Fehler über allen Instanzen

$$E_2(x_q) \equiv \frac{1}{2} \sum_{x \in D} (f(x) - \hat{f}(x))^2 \, d(x_q, x)$$

Strategien zur Beispielauswahl

Fallstudie: Aha, Kibler and Albert: Instance-based learning. MLJ 6 (1991).

IB1: Speichere alle Beispiele

- Gute Noisetoleranz, hoher Speicherbedarf

IB2: Speichere nur solche Beispiele, die von bisheriger Fallbasis falsch klassifiziert werden

- Geringe Noisetoleranz, geringer Speicherbedarf

IB3: Wie IB2, aber halte Zähler zu jedem Beispiel, wie oft es an richtiger bzw. falscher Vorhersage beteiligt war. Benutze Signifikanztest um herauszufinden, welche Beispiele vermutlich verrauscht sind (diese werden gelöscht)

- erhöhte Noisetoleranz bei geringem Speicherbedarf

Fallbasiertes Schließen

Können Instanz-basiertes Lernen auch anwenden, wenn $X \neq \Re^n$

$\longrightarrow$ brauchen anderes “Abstands”-Maß: Ähnlichkeit

Verschiedene Möglichkeiten:

- Farben: Abstand im Farbkreis
- Attributvektoren
  - Hammingabstand
- Strings:
  - Anzahl der unterschiedlichen Buchstaben
  - Differenz der Längen
  - Anzahl der Editoperationen

Im allgemeinen: Abstand/Ähnlichkeit domainabhängig, frei wählbar

Fallbasiertes Schließen

Warum eigentlich Ähnlichkeit über Zahlen definieren?

- Terme: Abstand mittels Antiunifikator
  - Abstand von $f(g(a, f(b, b)), c, d$ und $f(c, c, h(a, a))$ ist $f(X, c, Y)$
- Formeln: lgg (least general generalization)
- Graphen: Größter gemeinsamer Teilgraph
- Bilder: Menge gemeinsamer Bildteile

Allgemein: Ähnlichkeit definiert *Halbordnung* über den Instanzen
$\longrightarrow$ bestimmte Instanzen sind mglw. unvergleichbar

Was ist mit Symmetrie?

- sim(ICE,Zug) = sim(Bummelzug,Zug), aber
  sim(Zug,ICE) $>$ sim(Zug,Bummelzug)

Lernen: Lazy vs. Eager

Lazy: warte auf Anfrage, bevor generalisiert wird

- k-NEAREST NEIGHBOR, Fallbasiertes Schließen

Eager: Generalisiere, bevor Anfrage kommt

- ID3, NaiveBayes, . . .

Was ist besser?

- Eager Learning muß globale Approximation finden
- Lazy Learner kann viele lokale Approximationen kombinieren
- für den gleichen Hypothesenraum können Lazy Learner komplexere Funktionen repräsentieren (Beispiel: lineare Funktionen)