Computational Learning Theory

Sommersemester 2017

Mario Holldack, M. Sc.
Prof. Dr. Georg Schnitger
Hannes Seiwert, M. Sc.

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Institut für Informatik
AG Theoretische Informatik

Übungsblatt 6

Ausgabe: 29.05.2017
Abgabe: 06.06.2017, 10 Uhr

Wegen des Feiertags (Pfingstmontag) wird die Abgabefrist um einen Tag verlängert.

Aufgabe 6.1 *Weighted Majority und Online-Expertenauswahl* (3+3+3+3+3* Punkte)

Wir betrachten das Online-Problem der Auswahl von Experten: Gegeben sind $n \geq 2$ Experten $h_1, \ldots, h_n$, die in jedem Schritt $i = 1, 2, \ldots, t$ eine Ja/Nein-Empfehlung abgeben. Nach jeder Runde wird von einem Orakel mitgeteilt, welche Empfehlung (Ja oder Nein) richtig war. Ein Online-Algorithmus A hat die Aufgabe, anhand vergangener sowie aktueller Empfehlungen eine Ja/Nein-Entscheidung zu treffen.
Als Qualitätsmaß betrachten wir den sog. *(externen) Regret* von A.

$$\mathsf{Regret}_t(A) := \max_x \left(\text{Fehler}_{A,t}(x) - \min_{1 \leq j \leq n} \text{Fehler}_{h_j,t}(x) \right)$$

wobei $\text{Fehler}_{A,t}(x)$ die Anzahl der falschen Entscheidungen von A zum Zeitpunkt t auf Eingabe x (bestehend aus allen Empfehlungen und allen Orakelsprüchen) bezeichne. Wir vergleichen also A nach t Schritten mit dem besten Experten h_{opt_t} für eine schlimmstmögliche Eingabe.

a) Zeigen Sie, dass für jeden *deterministischen* Algorithmus A gilt: $\mathsf{Regret}_t(A) = \Omega(t)$

Deterministische Strategien haben also „keine Chance“. Wir betrachten nun *randomisierte* Strategien. Hier ist $\text{Fehler}_{A,t}(x)$ die *erwartete* Fehlerzahl, wobei der Erwartungswert über die internen Münzwürfe von A zu bilden ist.

b) Zeigen Sie, dass für die randomisierte Weighted-Majority-Strategie WM gilt:

$$\mathsf{Regret}_t(\text{WM}) = \mathcal{O}\left(\sqrt{t \ln(n)} + \ln(n) \right)$$

c) Zeigen Sie $\mathsf{Regret}_t(A) = \Omega(\log(n))$ für jedes $t \geq \log_2(n)$ und alle Algorithmen A.

Hinweis: Konstruieren Sie eine zufällige Eingabe, sodass jeder Algorithmus in Erwartung mindestens $\log_2(n)/2$ Fehler macht, sich der beste Experte aber nie irrt. In Runde i (für $i \leq \log_2(n)$) sollten sich genau $n/2^i$ Experten irren.

Fazit: Logarithmische „Aufwärmzeit“ ist für jeden Algorithmus notwendig.

d) Zeigen Sie $\mathsf{Regret}_t(A) = \Omega(\sqrt{t})$ für alle Algorithmen A.

Hinweis: Konstruieren Sie für $n = 2$ eine zufällige Eingabe, sodass jeder Algorithmus A in Erwartung $t/2$ Fehler macht.
Sie dürfen in d) und e) ohne Beweis benutzen: Für unabhängige und identisch binomialverteilte Zufallsvariablen $X_1, \ldots, X_n$ mit Varianz σ^2 und $\mathbb{E}[X_1] > \log_2(n)$ gilt: $\mathbb{E}\left[\min_{1 \leq i \leq n} X_i \right] = \mathbb{E}[X_1] - \Theta(\sigma \sqrt{\log(n)})$

e) **Bonusaufgabe:** Verbessern Sie die Schranke aus d) auf $\mathsf{Regret}_t(A) = \Omega(\sqrt{t \log n})$ für $t > \log_2 n$.

Fazit: Weighted Majority ist asymptotisch optimal im Hinblick auf den Regret.

Aufgabe 6.2 *Winnow für dünnbesetzte Formeln* (4 + 4 Punkte)

Betrachten Sie die Konzeptklasse k-TERM-ℓ-DNF$_n$ aller DNFs mit höchstens k Termen und höchstens ℓ Literalen pro Klausel auf den Variablen $V_1, \ldots, V_n$.

a) Es gelte $k = \mathcal{O}(n^{1/2})$. Zeigen Sie $\mathrm{VC}(k\text{-TERM-1-DNF}_n) = \Omega(k \cdot \log(n))$.

   Hinweis: Wir betrachten hier also Disjunktionen mit höchstens k Literalen. Nutzen Sie das Ergebnis aus Aufgabe 5.2 a).

b) Es gelte $k, \ell = \mathcal{O}(n^{1/3})$. Zeigen Sie $\mathrm{VC}(k\text{-TERM-}\ell\text{-DNF}_n) = \Omega(k\ell \cdot \log(n))$.

Fazit: Für „dünnbesetzte“ Formeln ist Winnow asymptotisch optimal.

Aufgabe 6.3 *Perzeptron vs. Winnow* (8 + 4 Punkte)

Sei im Folgenden $k = o(n)$ und der Beispielraum sei $X = \{0,1\}^n \times \{1\}$.

a) Bestimmen Sie

$$\mathrm{Gegenbeispiel}_{\mathrm{Perzeptron}}\Big(\big\{ \{b \in \{0,1\}^n : b \text{ erfüllt } x_1 \vee \cdots \vee x_k\} \big\} \Big)$$

   asymptotisch, d. h. bestimmen Sie sowohl eine asymptotische untere als auch eine asymptotische obere Schranke.

   Hinweis: Konstruieren Sie zum Nachweis der unteren Schranke eine Folge von Gegenbeispielen, die Perzeptron zu möglichst vielen Aktualisierungen zwingt. Betrachten Sie zunächst den Fall $k = 1$. Nehmen Sie an, dass Perzeptron mit dem Nullvektor beginnt.

   Wie viele Gegenbeispiele reichen für Winnow aus?

b) Betrachten Sie die Funktion $f_k : \{0,1\}^n \to \{0,1\}$ mit

$$f_k(x_1, \ldots, x_n) = 1 \iff \sum_{i=1}^{n} x_i \geq k$$

   Bestimmen Sie die hinreichenden Gegenbeispielzahlen von Winnow und Perzeptron für f_k asymptotisch. Bestimmen Sie dazu zunächst den jeweiligen Margin.

   Wie ändern sich jeweils die Gegenbeispielzahlen, wenn wir nur Beispiele mit höchstens m Einsen (für ein $k \leq m < n$) erhalten?