MUSTERERKENNUNG

Vorlesung im Sommersemester 2017

Prof. E.G. Schukat-Talamazzini

Stand: 6. März 2017

Teil XI

Klassifikatorübergreifende Verfahren

Aufgabenstellungen und Lösungsprinzipien

... die allen/vielen Klassifikationsverfahren gemeinsam sind

Offene Fragen

Alle Lernverfahren implementiert · alle Musterdaten gesammelt:

- Detektion von Ausnahmesituationen
  - *Musterklassen ohne Lerndaten*
- Kostenstruktur und Gütevergleich bei Zweiklassenproblemen
- Reduktion von Mehrklassenproblemen auf $K = 2$
- Ensembles kooperierender Klassifikatoren
  - *Expertenkomitees:* $\left\{ \begin{array}{c} \text{Zusammensetzung} \\ \text{Entscheidungsstrategien} \end{array} \right\}$
- Kombinatorische Auswahl des optimalen Merkmalsatzes
- Schritthaltendes Lernen und Klassifikatoradaption
- Lernen mit teilweise etikettierten Trainingsdaten

Unterscheidung zweier Klassen Ω_1, Ω_0

Keine Lernstichprobe ω_0 für Ω_0 verfügbar!

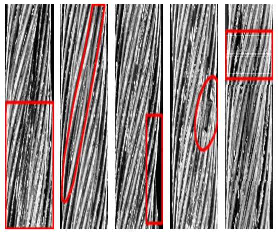
Fehlerdetektion

Ω_1: intakte Muster
Ω_0: defekte Muster

Fehler sind ziemlich **selten**.

Die Fehlerklasse ist **inhomogen**.

Beisp. „*Seilbahn*“



Statistischer Klassifikator

Prüfgröße ⇦ Laplace-Prinzip (vom unzureichenden Grunde)

$$\begin{aligned} u_1(\boldsymbol{x}) &= p_1 \cdot \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_1, \boldsymbol{S}_1) \\ u_0(\boldsymbol{x}) &= p_0 \cdot \text{const} \end{aligned}$$

Entscheidungsregel

$$\delta(\boldsymbol{x}) = \begin{cases} \Omega_1 & \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_1, \boldsymbol{S}_1) \geq \theta \\ \Omega_0 & \text{sonst} \end{cases}$$

Phantomdaten

(diskriminativ/nichtparametrisch)

Auswürfeln einer Surrogatprobe $\tilde{\omega}_0 \subset \Omega_0$

Zweiklassenszenarium

Verifikationsaufgabe — $\Omega_1/\Omega_0 = \left\{ \begin{matrix} \text{positive} \\ \text{negative} \end{matrix} \right\}$ Rückmeldung

Kostenmatrix

$$\boldsymbol{R} = \begin{pmatrix} r_{00} & r_{01} \\ r_{10} & r_{11} \end{pmatrix} = \begin{pmatrix} 0 & \rho_{FA} \\ \rho_{FR} & 0 \end{pmatrix}, \qquad \rho_{TA} = 0 = \rho_{TR}$$

Prüfgrößen

$$u_\lambda(\boldsymbol{x}) = \begin{cases} \rho_{FR} \cdot \mathrm{P}(\Omega_1, \boldsymbol{x}) & \lambda = 1 \\ \rho_{FA} \cdot \mathrm{P}(\Omega_0, \boldsymbol{x}) & \lambda = 0 \end{cases}$$

$$\Delta(\boldsymbol{x}) \overset{\text{def}}{=} \log \frac{u_1(\boldsymbol{x})}{u_0(\boldsymbol{x})} = \log \frac{\rho_{FR}}{\rho_{FA}} + \log \text{odds}(\boldsymbol{x})$$

Entscheidungsregel

$$\lambda^*(\boldsymbol{x}) = \begin{cases} 1 & \log \text{odds}(\boldsymbol{x}) \geq \theta \\ 0 & \log \text{odds}(\boldsymbol{x}) \leq \theta \end{cases}, \qquad \theta \overset{\text{def}}{=} \log \frac{\rho_{FA}}{\rho_{FR}}$$

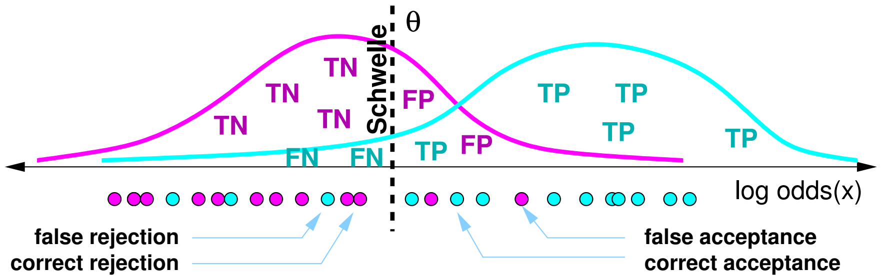
Kalibrierung der Verifikationsregel

Arbeitspunkt

Die Einstellung der Prüfgrößenschwelle θ in der Verifikationsregel

$$\log \text{odds}(\boldsymbol{x}) \overset{?}{\geq} \theta$$

heißt Arbeitspunkt. Er bestimmt, wieviele Muster akzeptiert werden und wieviele zurückgewiesen.



Ein Zoo von Fehlermaßen ...

... bei festem Arbeitspunkt θ

Verifikationsergebnis

θ	accept	reject	
good guys	N_{tt}	N_{tf}	$N_{t\cdot}$
bad guys	N_{ft}	N_{ff}	$N_{f\cdot}$
	$N_{\cdot t}$	$N_{\cdot f}$	

Referenzbezogene Raten

$^{N_{tt}}/_{N_{t\cdot}}$ TPR, sensitivity, recall

$^{N_{tf}}/_{N_{t\cdot}}$ FNR, loss $= 1-\text{recall}$

$^{N_{ft}}/_{N_{f\cdot}}$ FPR, fallout

$^{N_{ff}}/_{N_{f\cdot}}$ TNR, vigilance $= 1-\text{fallout}$

Populationsbezogene Raten

$^{(N_{tt}+N_{ff})}/_{N_{\cdot\cdot}}$ accuracy

$^{(N_{tf}+N_{ft})}/_{N_{\cdot\cdot}}$ error

Gewichteter F-Value

$$\left(\frac{\beta}{\text{PRE}} + \frac{1-\beta}{\text{REC}}\right)^{-1} = \frac{N_{tt}}{2N_{tt}+N_{tf}+N_{ft}}$$

Hypothesenbezogene Raten

$^{N_{tt}}/_{N_{\cdot t}}$ precision (Ausbeute)

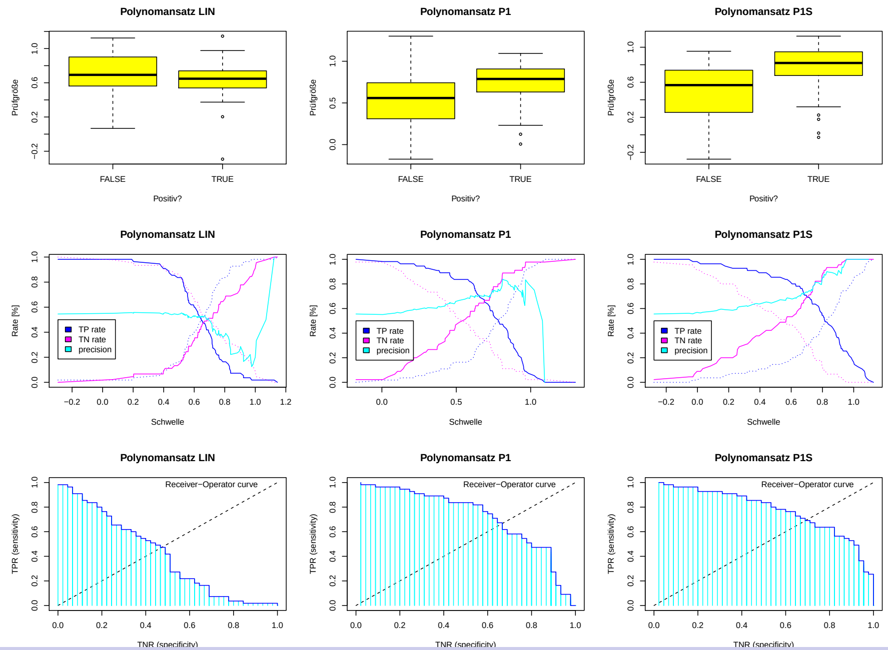
$^{N_{tf}}/_{N_{\cdot f}}$ waste

$^{N_{ft}}/_{N_{\cdot t}}$ garbage $= 1-\text{precision}$

$^{N_{ff}}/_{N_{\cdot f}}$ care $= 1-\text{waste}$

ROC — Receiver-Operator-Charakteristik

Diabetesdaten der Pima-Indianer mit verschiedenen Polynomklassifikatoren



R Programmcode

code:receiver-operator

```
load ("data/twoclass.rda")

plot.scorebox <- function (u, ...)
        plot (u~factor(names(u)), xlab="Positiv?", ylab="Prüfgröße", ...)

plot.recall <- function (u, copa=c("blue","magenta","cyan"), ...) {
        u <- sort(u)
        hit <- names(u) == "TRUE"
        fnr <- cumsum(hit) / sum(hit)
        tpr <- 1-fnr
        tnr <- cumsum(!hit) / sum(!hit)
        fpr <- 1-tnr
        pre <- (sum(hit)-cumsum(hit)) / (length(hit):1-1)
        plot (u, tpr, type="n", xlab="Schwelle", ylab="Rate [%]", ...)
        lines (u,tpr,col=copa[1],type="l")
        lines (u,fnr,col=copa[1],type="l",lty="dotted")
        lines (u,tnr,col=copa[2],type="l")
        lines (u,fpr,col=copa[2],type="l",lty="dotted")
        lines (u,pre,col=copa[3],type="l")
        legend (min(u), 0.5,
                legend=c("TP rate","TN rate","precision"), fill=copa)
        }

plot.ROC <- function (u, ...) {
        u <- sort(u)
        hit <- names(u) == "TRUE"
        fnr <- cumsum(hit) / sum(hit)
        tpr <- 1-fnr
        tnr <- cumsum(!hit) / sum(!hit)
        fpr <- 1-tnr
        plot (0:1, 0:1, type="l", lty="dashed",
                xlab="TNR (specificity)", ylab="TPR (sensitivity)", ...)
```


Vergleich von Klassifikatoren

Problem (Kostenmatrix unbekannt)

Ohne Kenntnis der Kostenmatrix läßt sich der ***tatsächliche Arbeitspunkt*** θ *der Entscheidungsregel nicht fixieren.*

Problem (Pareto-Phänomen)

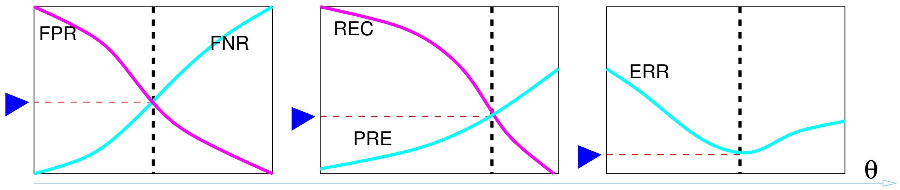
Die individuellen arbeitspunktabhängigen Fehlerkurven zweier wettbewerbender Klassifikatorprüfgrößen sind ***nicht notwendig vergleichbar****.*

Lösung

- *Bestimmung eines* ***optimalen Arbeitspunktes*** θ^* *und Vergleich der korrespondierenden Gütewerte (BEP, EER, 45°-Tangente ...)*
- *Definition einer geeigneten* ***kumulativen Kenngröße*** *für die Gütekurve (AUC, mittlere Ausbeute)*

Fehlermaße mit gleitendem Arbeitspunkt

Minimale und balancierte Fehlerraten



Equal Error Rate

Balancepunkt von Fehl**rückweisungs**rate und Fehl**akzeptanz**rate

$$\text{EER} \stackrel{\text{def}}{=} \text{FNR}_{\theta^*} \;, \qquad \theta^* = \operatorname*{argmin}_{\theta} |\text{FNR}_\theta - \text{FPR}_\theta|$$

Break-Even Point

Schnittpunktwert der **recall**-Kurve mit der **precision**-Kurve; impliziert $N^\theta_{t.} = N^\theta_{.t}$

$$\text{BEP} \stackrel{\text{def}}{=} \text{REC}_{\theta^*} \;, \qquad \theta^* = \operatorname*{argmin}_{\theta} |\text{REC}_\theta - \text{PRE}_\theta|$$

AUC — Area Under (ROC) Curve

Wilcoxon-Mann-Whitney Statistik

Definition

Den Flächeninhalt unter der Receiver-Operator-Kurve

$$\text{ROC} : \begin{cases} [0,1] & \to \quad [0,1] \\ \text{TNR}(\theta) & \mapsto \quad \text{TPR}(\theta) \end{cases}$$

bezeichnen wir als **AUC-Wert**.

Bemerkungen

1. Empirisch werden einfach alle TPR (Recallwerte) gemittelt, die sich an den Schwellen θ der Negativmuster ergeben.
2. Ähnlich definiert ist die AVP (*'average precision'*) des Information Retrieval.
3. Der AUC-Wert ist äquivalent zur **Wilcoxon-Rangsummenstatistik**

$$\frac{1}{N_{t.} N_{f.}} \cdot \sum_{x \in \omega_1} \sum_{y \in \omega_0} \mathbf{I}_{u(x) > u(y)} \;,$$

welche die Wahrscheinlichkeit $\mathrm{P}(\mathbb{U}_1 > \mathbb{U}_0)$ dafür schätzt, daß die Prüfgröße eines Positivmusters größer ist als die Prüfgröße eines Negativmusters.

mehr Information

Zwei-Klassen-Diskriminanten

Mächtige Formalismen zur Zwei-Klassen-Unterscheidung

- Supportvektormaschinen
- Logistische Regression
- Mehrschichtenperzeptron
- Polynomklassifikator (Importvektormaschine)

Nutzung von Zwei-Klassen-Diskriminanten

- Kontrastive Dichteschätzung
- Mehr-Klassen-Entscheidung ⇦ *„Jeder gegen alle“*
- Mehr-Klassen-Entscheidung ⇦ *„Jeder gegen jeden“*
- Mehr-Klassen-Entscheidung ⇦ Fehlerkorrigierende Codes

Logarithmierte Gewinnquote

Zwei Klassen $\{\Omega_1, \Omega_0\}$ — eine Prüfgröße $\log \text{odds}(\boldsymbol{x})$

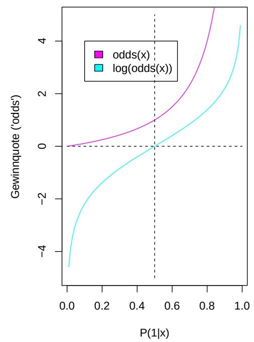
$$\kappa^*(x) \;=\; \underset{\lambda\in\{1,0\}}{\text{argmax}} \frac{p_\lambda \cdot f_\lambda(x)}{p_1 f_1(x) + p_0 f_0(x)} \;=\; \begin{cases} 1 & p_1 \cdot f_1(x) \;\geq\; p_0 \cdot f_0(x) \\ 0 & p_1 \cdot f_1(x) \;\not\geq\; p_0 \cdot f_0(x) \end{cases}$$

Definition

Die a posteriori Klassenverteilung kann in der Zweiklassenwelt in einer einzigen Größe, der **Gewinnquote** oder den **Odds**

$$\text{odds}(x) \;\stackrel{\text{def}}{=}\; \frac{\mathrm{P}(1|x)}{\mathrm{P}(0|x)} \;=\; \frac{p_1 \cdot f_1(x)}{p_0 \cdot f_0(x)}$$

subsumiert werden.



Gewinnquote als Dichteverhältnis

Lemma

Aus der Chancenfunktion odds$(\cdot)$ sind die a posteriori Klassenwahrscheinlichkeiten via

$$\mathrm{P}(1|x) \;=\; \frac{odds(x)}{1 + odds(x)} \quad \textit{bzw.} \quad \mathrm{P}(0|x) \;=\; \frac{1}{1 + odds(x)}$$

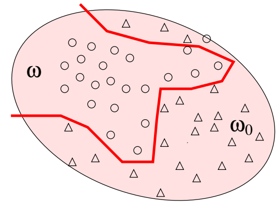
exakt reproduzierbar.

Zwischen den beiden Klassenverteilungen f_1 und f_0 für „positive" ($\kappa = 1$) und „negative" ($\kappa = 0$) Muster besteht der wichtige Zusammenhang

$$f_1(x) \;=\; odds(x) \cdot \frac{p_0}{p_1} \cdot f_0(x) \qquad (\forall x \in \boldsymbol{\Omega}) \;.$$

Bemerkung

Ist also die Negativverteilung f_0 bekannt sowie die Klassendiskriminante odds : $\boldsymbol{\Omega} \to \mathbb{R}^+$, so ist auch die Positivverteilung f_1 schon festgelegt.



Kontrastive Verteilungsdichteschätzung

(Algorithmus)

1. KONTRASTMODELL
   Wähle eine Verteilungsfunktion $f_0 : \boldsymbol{\Omega} \to \mathbb{R}$.
2. AUSWÜRFELN DER KONTRASTPROBE
   Erzeuge eine Beispieldatensammlung $\omega_0 = \{y_1, \ldots, y_S\}$ für $f_0(\cdot)$.
3. DISKRIMINANTE LERNEN
   Lerne unter Verwendung der Positiv- und Negativmuster in ω bzw. ω_0 einen binären Klassifikator mit der Diskriminanzfunktion $\widehat{\text{odds}}(\cdot)$ an.
4. DICHTESCHÄTZUNG
   Reproduziere die gesuchte Verteilung näherungsweise durch

$$\hat{f}(x) \;\stackrel{\text{def}}{=}\; \widehat{\text{odds}}(x) \cdot \frac{|\omega_0|}{|\omega|} \cdot f_0(x) \;.$$

(Algorithmus)

Bemerkung

Statt Setzen der Kontrastdichte und Auswürfeln der Kontrastdaten können auch vorliegende Daten als Kontrast verwendet werden; die Parameter der Kontrastdichte werden daraus geschätzt.

Klassifikatoren für $K > 2$ Klassen

Standardtechniken: $\tilde{\omega}_\kappa = \bigcup_\lambda \omega_\lambda$ oder $\tilde{\omega}_\kappa = \bigcup_{\lambda\neq\kappa} \omega_\lambda$

Klassenweise Kontrastdiskriminaten

Hypothetische Kontrastverteilungen $h_1, \ldots, h_K$:

$$\text{odds}_\lambda(x) \;=\; \frac{q_\lambda \cdot f_\lambda(x)}{(1 - q_\lambda) \cdot h_\lambda(x)} \;, \qquad \lambda \in \{1, \ldots, K\}$$

(Kontrastdaten im Anzahlverhältnis $(1 - q_\kappa)$ zu q_κ)

Klassenweise Kontrastschätzung

$$\hat{f}_\lambda(x) \;\stackrel{\text{def}}{=}\; \widehat{\text{odds}}_\lambda(x) \cdot \frac{1 - q_\lambda}{q_\lambda} \cdot h_\lambda(x) \;, \qquad \lambda \in \{1, \ldots, K\}$$

Reproduzierte Bayesregel

$$\hat{\mathrm{P}}(\lambda|x) \;\propto\; T_\lambda \cdot \widehat{\text{odds}}_\lambda(x) \cdot \frac{S_\lambda}{T_\lambda} \cdot h_\lambda(x) \;\propto\; S_\lambda \cdot \widehat{\text{odds}}_\lambda(x)$$

(Die letzte Proportionalität gilt nur, wenn alle Kontrastverteilungen identisch sind!)

Kreuzklassifizierende Diskriminanten

Jeder gegen jeden — aber welches ist die wahrscheinlichste Klasse?

Ideale Kreuzdiskriminanten

$$\text{odds}_{\kappa\lambda}(x) = \frac{q_\kappa \cdot f_\kappa(x)}{q_\lambda \cdot f_\lambda(x)} \qquad (\forall \lambda, \kappa, x)$$

Für ein festes $x \in \boldsymbol{\Omega}$ sei $\boldsymbol{R}(\boldsymbol{x})$ die quadratische Matrix mit Einträgen

$$r_{\kappa\lambda} \stackrel{\text{def}}{=} \log \text{odds}_{\kappa\lambda}(x) \ .$$

Fakt

Die Matrix $\boldsymbol{R}$ hat einen Rang ≤ 2 und besitzt wegen

$$r_{\kappa\lambda} = \underbrace{\log(q_\kappa \cdot f_\kappa(x))}_{=:u_\kappa} - \underbrace{\log(q_\lambda \cdot f_\lambda(x))}_{=:u_\lambda} ,$$

die Darstellung $(\boldsymbol{u}\boldsymbol{1}^\top - \boldsymbol{1}\boldsymbol{u}^\top)$ mit den K-Klassen-Diskriminanten $u_\lambda(\cdot)$, $\lambda = 1, \ldots, K$.

Kreuzklassifizierende Diskriminanten

Näherungsweise Rekonstruktion der originalen Bayesprüfgrößen

Minimierung des mittleren quadratischen Fehlers

$$\varepsilon(\boldsymbol{u}) \stackrel{\text{def}}{=} \left\| \hat{\boldsymbol{R}} - \boldsymbol{u}\boldsymbol{1}^\top + \boldsymbol{1}\boldsymbol{u}^\top \right\|_{\mathcal{F}}^2$$

Eindeutige Lösung

$$\hat{u}_\kappa = \frac{1}{K}\sum_{\lambda=1}^{K} \hat{r}_{\kappa\lambda} = \frac{1}{K}\sum_{\lambda=1}^{K} \log \widehat{\text{odds}}_{\kappa\lambda}(x)$$

für alle Klassen $\kappa \in \{1, \ldots, K\}$

Reproduzierte Bayesregel

$$\kappa^*(x) = \underset{\kappa}{\text{argmax}} \prod_{\lambda=1}^{K} \widehat{\text{odds}}_{\kappa\lambda}(x)$$

Ensemblemethoden

Klassifikationsentscheidungen — im Expertenteam gefällt

Welche Wissensquellen werden kombiniert?

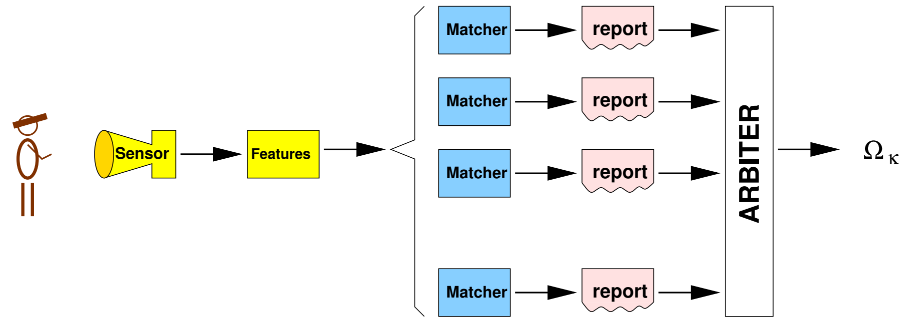
- **Orthogonale Sensoren**
- **Wiederholte Messungen**
- **Konkurrierende Klassifikatortypen**
- **Streuende Parameterschätzwerte**
- **Spezialisierte Entscheider**

Wie werden die Quellen kombiniert?

- **Sensordatenfusion** (Mittel- und Produktbildung)
- **Interaktionsmodelle** (Zeitreihen; Konsortiale Klassifikation)
- **Prüfgrößenkopplung** (Metaklassifikation)
- **Votierungsverfahren** (Mehrheit und Präferenz)

Komitee-Maschinen treffen Mehrheitsentscheidung

Ein Sensor · eine Messung · M unabhängige Experten



Society of Experts

Konkurrierende Klassifikationsverfahren oder konkurrierende Modellkapazitäten

Bagging

Bootstrap Aggregation
Konkurrierende ausgewürfelte Lerndatensammlungen

Randomisierung

Konkurrierende Parametersätze (zufällige Startwerte iterativer Lernverfahren)

Bootstrap-Stichproben

T-mal zufälliges Ziehen aus ω mit Zurücklegen

Lemma

Es sei $\omega = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\} \subset \boldsymbol{\Omega}$ ein Datensatz und

$$\tilde{\omega} \;=\; \{\boldsymbol{x}_{\tau(1)}, \boldsymbol{x}_{\tau(2)}, \boldsymbol{x}_{\tau(3)}, \ldots, \boldsymbol{x}_{\tau(T)}, \}\ , \qquad \tau : [1..T] \to [1..T]$$

*eine **Bootstrapprobe** der Größe T daraus.*

1. *Die Wahrscheinlichkeit dafür, beim Bootstrap übergangen zu werden, beträgt asymptotisch*

$$\lim_{T\to\infty} \mathrm{P}(\boldsymbol{x}_t \notin \tilde{\omega}) \;=\; \lim_{T\to\infty} \left(1 - {}^{1}\!/_{T}\right)^T \;=\; \frac{1}{e} \;\approx\; 0.368\ .$$

2. *Ist $s : \boldsymbol{\Omega}^T \to \mathbb{R}$ eine beliebige Stichprobenstatistik, so ist ihr Erwartungwert für Originalprobe und Bootstrapprobe gleich:*

$$\mathcal{E}_\omega[s(\mathbb{X}_1, \ldots, \mathbb{X}_T)] \;=\; \mathcal{E}_{\tilde{\omega}}[s(\mathbb{X}_{\mathbb{T}(1)}, \ldots, \mathbb{X}_{\mathbb{T}(T)})]$$

Bootstrappingverfahren zum Lernen und Testen

Simuliertes Auswürfeln von lerndatenbezogenen a posteriori-Verteilungen

Kreuzvalidierung

· Anlernen des Klassifikators mit $\tilde{\omega}$
· Testen des Klassifikators mit ω

$$\hat{p}_\varepsilon(\omega, \tilde{\omega}) \;\stackrel{\text{def}}{=}\; (1 - {}^{1}\!/_{e}) \cdot p_\varepsilon(\tilde{\omega} \mid \delta_{\tilde{\omega}}) \;+\; {}^{1}\!/_{e} \cdot p_\varepsilon(\omega \setminus \tilde{\omega} \mid \delta_{\tilde{\omega}})$$

Fehlerratenstreuung

Bestimme Mittelwert und Varianz der Schätzungen $\hat{p}_\varepsilon(\omega, \tilde{\omega}^{(m)})$, $m = 1..M$.

Parameterschätzung

Mittele Parameterschätzwerte $\boldsymbol{\theta}^{(m)}$ einer Reihe von Bootstrapproben $\tilde{\omega}^{(m)}$.

Bootstrap Aggregation

Mittele Entscheidungen $\delta^{(m)}(\boldsymbol{x})$ einer Reihe von Bootstrapproben $\tilde{\omega}^{(m)}$.

Boosting — Konsultation komplementärer Klassifikatoren

Ein Klassifikatortyp · Parameter nach sequentieller Probenselektion

(Algorithmus)

1. INITIALISIERUNG
   Setze konstante Gewichte $\beta_t^{(1)} = 1/T$ und für alle $m = 1, 2, \ldots$:
2. ENTSCHEIDUNGSREGEL LERNEN
   Lerne neuen Klassifikator $\delta^{(m)} = \delta(\omega, \boldsymbol{\beta}^{(m)})$.
3. REKLASSIFIKATIONSTEST
   Berechne eine neue (gewichtete) Fehlerrate
   $\varepsilon_m := p_\varepsilon(\omega, \boldsymbol{\beta}^{(m)} \mid \delta^{(m)})$.
4. TERMINIERUNG (falls $\varepsilon_m \notin (0, {}^{1}\!/_{2})$)
5. ERFOLGSGEWICHTE AKTUALISIEREN

$$\beta_t^{(m+1)} \;\propto\; \begin{cases} \beta_t^{(m)} & \boldsymbol{x}_t \text{ falsch klassifiziert} \\ \beta_t^{(m)} \cdot \varepsilon_m/(1 - \varepsilon_m) & \boldsymbol{x}_t \text{ korrekt klassifiziert} \end{cases}$$

Nächster AdaBoost.M1-Iterationsschritt $\rightsquigarrow$ 2

(Algorithmus)

Boosting — aufmerksamkeitsgesteuertes Lernen

Sequentielles Lernen schwer klassifizierbarer Muster durch „Nachsitzen“

Klassifikationsphase

Expertenentscheidungen gewichtet summieren:

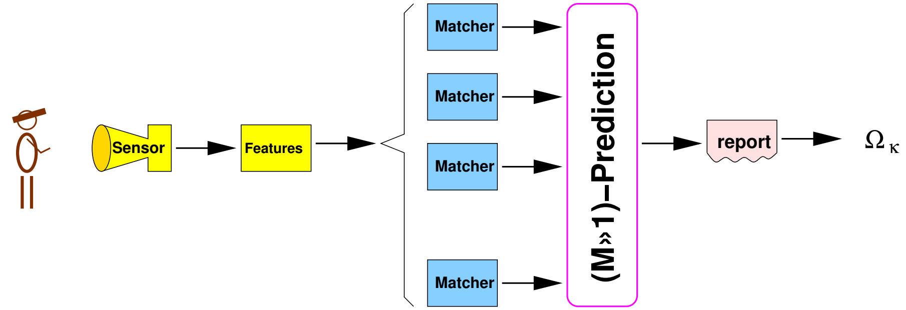
$$u_\kappa(\boldsymbol{x}) \; = \; \sum_{m=1}^{M} -\log \frac{\varepsilon_m}{1-\varepsilon_m} \cdot \delta_\kappa^{(m)}(\boldsymbol{x})$$

Bemerkungen

1. AdaBoost endet wenn $p_\varepsilon^{(m)} \geq 1/2$ — kein Problem für $K = 2$.
2. **Gewichtetes Lernen** von Klassifikatoren:
   · Integration in das Parameterlernverfahren (ML, QM)
   · Simulation durch gewichtetes Auswürfeln einer Bootstrap-Lernprobe.
   · Vergröberte Simulation durch Probe mit $C \cdot T \cdot \beta_t^{(m)}$ vielen $\boldsymbol{x}_t$-Kopien
3. Je geringer die Rate $\varepsilon_m = p_\varepsilon^{(m)}$, desto größer das Abstimmungsgewicht.

Prüfgrößeninterpolation

Frühe Integration quantitativer Expertenurteile



Lineare Interpolation

Die **Mischungskoeffizienten** $\{\pi_1, \ldots, \pi_M\}$ des Vorhersagemodells

$$\boldsymbol{u}(\boldsymbol{x}) \; = \; \sum_{m=1}^{M} \pi_m \cdot \boldsymbol{u}^{(m)}(\boldsymbol{x}) \quad \stackrel{!}{\approx} \quad \boldsymbol{\delta}^*(\boldsymbol{x}) \; , \qquad \sum_{m=1}^{M} \pi_m = 1$$

werden mit einer Variante des EM-Algorithmus bestimmt (*held-out*-Daten!)

Merkmalgewinnung

Komplexer Arbeitsprozeß beim Entwurf eines ME-Systems

1. **Domänenwissen** ad hoc Merkmalsatz entwerfen
2. **Vergleichbarkeit** Merkmale normieren
3. **Interaktionen** Produkte/Konjunktionen dazu
4. **Hilfsvariablen** Linearkombinationen/Disjunktionen dazu
5. **Variablenreihung** Nützlichkeit der Merkmale quantifizieren
6. **Ausreißerdetektion** Kriterium sind die höchstrangigen Merkmale
7. **Teilmengenauswahl** schnelles, robustes Kriterium
8. **Wettbewerb austragen** Klassifikations- und Selektionstechniken
9. **Endauswertung** stabile Lösung mit Bootstrap/Kreuzvalidierung

Merkmalauswahl

Welche Teilmenge des Merkmalsatzes $\{x_1, \ldots, x_D\}$ wird verwendet?

Entwurfsziele

geringere Fehlerrate
schnellere Verarbeitung
Strukturaufdeckung

Kombinatorische Explosion

Es gibt insgesamt 2^D viele Teilmengen $\mathcal{S} \subset \{x_1, \ldots, x_D\}$.

Standalone-Gütekriterien

Relevanz eines Merkmals
Redundanz eines Merkmals

Ranker

1. Bewerte alle Merkmale x_d nach Relevanz
2. Verwende nur die M Bestbewerteten

Filter

1. Bewerte Merkmale nach Relevanz
2. Analysiere Abhängigkeiten untereinander
3. Treffe eine kompatible Auswahl

Wrapper

1. Wähle Basisklassifikationsverfahren
2. Bewerte Teilmengen $\mathcal{S}$ durch Evaluation
3. Suche die höchstperformante Teilmenge

Ranker — Variablenreihung nach Relevanz

Wie groß ist der individuelle Beitrag von x_d zur Klassenentscheidung?

Pearsons Korrelationskoeffizient

Abhängigkeitsmaß für stetige Zufallsvariable (schwierig für $K \neq 2$)

$$r(d) \;=\; \frac{\mathrm{Cov}[\mathbb{X}_d, \mathbb{Y}]}{\sqrt{\mathrm{Var}[\mathbb{X}_d] \cdot \mathrm{Var}[\mathbb{Y}]}}$$

Transinformation

Divergenz zwischen gemeinsamer Verteilung und Produktverteilung

$$r(d) \;=\; \mathcal{D}(f_{\mathbb{X}_d\mathbb{Y}} \| f_{\mathbb{X}_d} f_{\mathbb{Y}}) \;=\; \int\!\!\int f_{\mathbb{X}_d\mathbb{Y}}(x_d, y) \cdot \log \frac{f_{\mathbb{X}_d\mathbb{Y}}(x_i, y)}{f_{\mathbb{X}_d}(x_i) \cdot f_{\mathbb{Y}}(y)}\, dx_i\, dy$$

Ein-Merkmal-Klassifikatoren

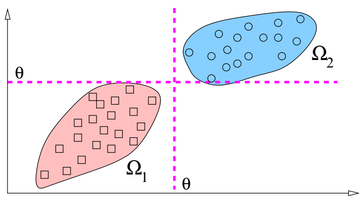
Betreibe ein Standardklassifikationssystem mit Mustern $\boldsymbol{x} = (x_d) \in \mathbb{R}^1$ und ermittle z.B. die *Held-out*-Erkennungsrate $r(d) = 1 - p_\varepsilon(\delta(\omega^{(d)}))$.

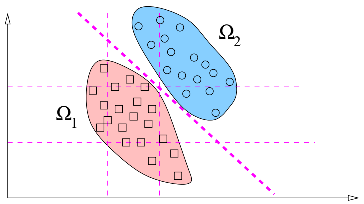
Ranker — Variablenreihung nach Relevanz

$$\mathcal{S}_\theta \stackrel{\text{def}}{=} \{x_d \mid r(d) \geq \theta\} \;, \qquad \theta \in \mathbb{R}_+$$

PRO

Extrem schnelles und einfaches Verfahren





CONTRA

Die Welt ist komplizierter, als der Ranker denkt:

- Ein **hoch relevantes** Merkmal ist u.U. **völlig nutzlos**, wenn es den bereits selektierten keine Musterinformation hinzufügt.
- Zwei oder mehrere **nicht relevante** Merkmale besitzen u.U. **hohen Nutzwert**, wenn sie der Auswahl **gemeinsam** hinzugefügt werden.

FCBF — Fast Correlation-Based Filtering

Relevanz eines Merkmals · Redundanz eines Merkmals

Relevanz von $\mathbb{X}_i$

Symmetrische Unsicherheit $\tilde{\Im}(\mathbb{X}_i, \mathbb{Y})$ zwischen Merkmal und Klassenvariable, wobei

$$\tilde{\Im}(\mathbb{X}, \mathbb{Y}) \;=\; \frac{2 \cdot \Im(\mathbb{X}, \mathbb{Y})}{\mathcal{H}(\mathbb{X}) + \mathcal{H}(\mathbb{Y})}$$

mit der **Transinformation**

$$\Im(\mathbb{X}, \mathbb{Y}) \stackrel{\text{def}}{=} \mathcal{H}(\mathbb{X}) + \mathcal{H}(\mathbb{Y}) - \mathcal{H}(\mathbb{X}, \mathbb{Y})$$

Redundanz von $\mathbb{X}_i$

Für zwei relevante (SU $\geq \theta$) Merkmale $\mathbb{X}_i$, $\mathbb{X}_j$ heiße $\mathbb{X}_j$ **ebenbürtig** zu $\mathbb{X}_i$, falls

$$\tilde{\Im}(\mathbb{X}_i, \mathbb{X}_j) \;\geq\; \tilde{\Im}(\mathbb{X}_i, \mathbb{Y})$$

gilt. Merkmal $\mathbb{X}_i$ ist **redundant**, wenn es ein ebenbürtigen Wettbewerber $\mathbb{X}_j$ gibt mit:

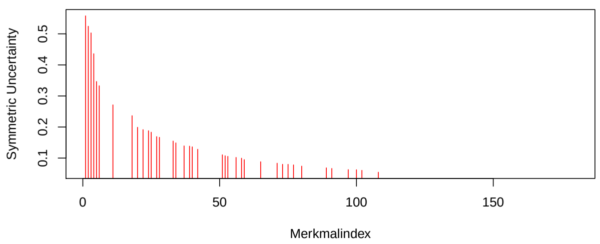
$$\tilde{\Im}(\mathbb{X}_j, \mathbb{Y}) \;\geq\; \tilde{\Im}(\mathbb{X}_i, \mathbb{Y})$$

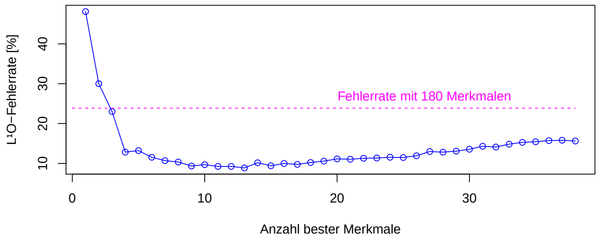
Problem

*Die Redundanzeigenschaft ist nur **relativ** zu einem willkürlichen Abschnitt* $\mathcal{S}_\theta \;=\; \{x_i \mid \tilde{\Im}(\mathbb{X}_i, \mathbb{Y}) \geq \theta\}$ *relevanter Merkmale definiert!*

FCBF — Anwendungsbeispiel

DNA-Datensatz · 3 Klassen · 180 $^0\!/_1$-Merkmale · 3186 Muster





FCBF wählt maximal ($\delta = 0$) 40 von 180 Merkmalen aus.

Fehlerraten

23.8%	180
09.3%	9
12.8%	4
23.0%	3

Erfolgsbilanz

Faktor 20	Zeit
Faktor 2.5	Fehler

Bemerkung

$180^3 = 5.8 \cdot 10^6$ Kombinationen, um (x_{90}, x_{85}, x_{93}) zu finden!

Wrapper-Methoden

Gütebewertung durch Klassifikationstest

Mit Hilfe eines *robusten* und *CV-fähigen* Klassifikationsverfahrens wird

$$\mathrm{J}(\mathcal{S}) \stackrel{\text{def}}{=} 1 - \hat{p}_\varepsilon^{\mathrm{CV}}(\delta(\omega^{\mathcal{S}})) , \qquad \omega^{\mathcal{S}} = \{\boldsymbol{x}^{\mathcal{S}} \mid \boldsymbol{x} \in \omega\}$$

definiert.

Kombinatorische Optimierung

Wähle die bestbewertete Merkmalteilmenge

$$\mathcal{S}^* = \underset{\mathcal{S}\subseteq\mathcal{F}}{\operatorname{argmax}} \, \mathrm{J}(\mathcal{S}) , \qquad \mathcal{F} \stackrel{\text{def}}{=} \{x_1, \ldots, x_D\}$$

Suchverfahren

Vollständige Suche	Aufwand $\mathrm{O}(2^D)$
Gierige Suche	Vorwärts-/Rückwärtsauswahl
Zulässige Suche	A*-Algorithmus, Branch&Bound
Approximative Suche	Prophet statt Orakel

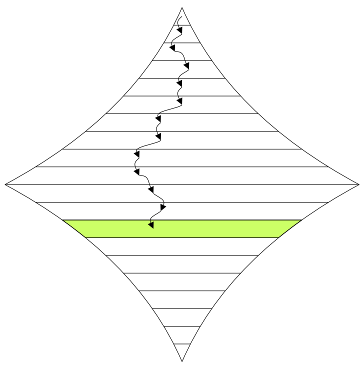
SFS — Sequential Forward Selection

Gierige Bottom-up Auswahlstrategie

1. INITIALISIERUNG
   Setze $\mathcal{S} \leftarrow \emptyset$.
2. AUSWAHL AUFWÄRTS
   Bestimme das beste zusätzliche Merkmal

   $$x_d = \operatorname{argmax} \{\mathrm{J}(\mathcal{S}, x_d) \mid x_d \notin \mathcal{S}\}$$

3. TERMINIERUNG
   Wenn $\mathrm{J}(\mathcal{S}, x_d) \leq \mathrm{J}(\mathcal{S}) \rightsquigarrow$ Ende.
   Sonst $\mathcal{S} \leftarrow \mathcal{S} \cup \{x_d\}$ und $\rightsquigarrow$ 2.



Bemerkung

SFS trifft voreilige Entscheidungen (Horizont=1) und verfehlt i.a. die Optimallösung.

$$\mathcal{S}^{(1)} \subset \mathcal{S}^{(2)} \subset \ldots \subset \mathcal{S}^{(i)} \subset \ldots$$

⇩ Redundanz · ⇩ Kombination

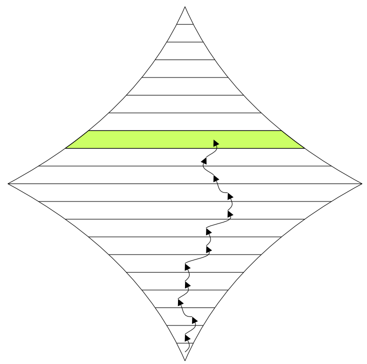
SBE — Sequential Backward Elimination

Gierige Top-down Auswahlstrategie

1. INITIALISIERUNG
   Setze $\mathcal{S} \leftarrow \mathcal{F} = \{x_1, \ldots, x_D\}$.
2. AUSWAHL ABWÄRTS
   Bestimme das verzichtbarste Merkmal

   $$x_d = \operatorname{argmax} \{\mathrm{J}(\mathcal{S} \backslash x_d) \mid x_d \in \mathcal{S}\}$$

3. TERMINIERUNG
   Wenn $\mathrm{J}(\mathcal{S} \backslash x_d) \leq \mathrm{J}(\mathcal{S}) \rightsquigarrow$ Ende.
   Sonst $\mathcal{S} \leftarrow \mathcal{S} \setminus \{x_d\}$ und $\rightsquigarrow$ 2.



Bemerkung

SBE aufwändiger als SFS:
· Start mit umfangreicheren $\mathcal{S}$
· Längerer Weg zum Ziel

$$\mathrm{O}(n \cdot (D-n)^2) \gg \mathrm{O}(n^2 \cdot (D-n))$$

⇧ Redundanz · ⇧ Kombination

PTA(k,ℓ) — Plus k Take Away ℓ

Gierige bidirektionale Auswahlstrategie

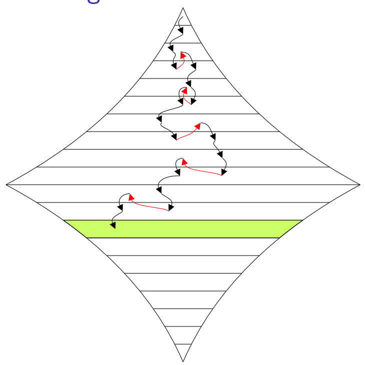
1. INITIALISIERUNG · $\mathcal{S}' \leftarrow \mathcal{S} \leftarrow \emptyset$
2. MEHRFACHAUSWAHL AUFWÄRTS
   Bestimme k-mal nacheinander das beste zusätzliche Merkmal

   $$x_d \;=\; \operatorname{argmax}\,\{\mathrm{J}(\mathcal{S},x_d) \mid x_d \notin \mathcal{S}\}$$

   und setze $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{x_d\}$.
3. MEHRFACHAUSWAHL ABWÄRTS
   Bestimme ℓ-mal nacheinander das verzichtbarste Merkmal

   $$x_d \;=\; \operatorname{argmax}\,\{\mathrm{J}(\mathcal{S}\backslash x_d) \mid x_d \in \mathcal{S}\}$$

   und setze $\mathcal{S}' \leftarrow \mathcal{S}' \setminus \{x_d\}$.
4. TERMINIERUNG
   Wenn $\mathrm{J}(\mathcal{S}') \le \mathrm{J}(\mathcal{S}) \rightsquigarrow$ Ende.
   Sonst $\mathcal{S} \leftarrow \mathcal{S}'$ und $\rightsquigarrow$ 2.



Bemerkung

Redundant gewordene Merkmale können jetzt wieder eliminiert werden.

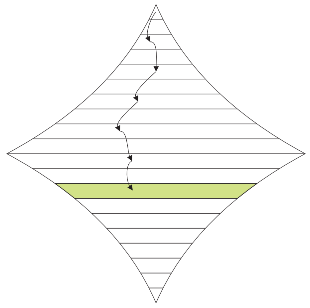
Mehraufwand gegenüber SFS:

$$(k+\ell)/(k-\ell)$$

⇧ Redundanz · ⇩ Kombination

Verallgemeinerte sequentielle Verfahren

Gierige blockweise Auswahlstrategien · Polynomialer Aufwand $\mathrm{O}(n^{m+1} \cdot (D-n))$



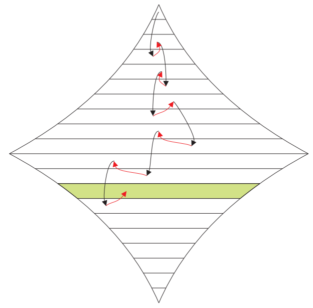
GSFS(m)

Je Aufwärtsschritt wird die performanteste Erweiterungsmenge

$$\mathcal{S}^*_\oplus \;=\; \{x_{d_1}, x_{d_2}, \ldots, x_{d_m}\}$$

ermittelt und ggf. hinzugefügt.

⇩ Redundanz · ⇧ Kombination



GPTA(k,ℓ)

In den Akkumulations- und Eliminationsschritten werden optimale Auswahlen $\mathcal{S}^*_\oplus$, $\mathcal{S}^*_\ominus$ mit

$$|\mathcal{S}^*_\oplus| = k \quad \text{bzw.} \quad |\mathcal{S}^*_\ominus| = \ell$$

getroffen.

SFFS — Sequential Forward Floating Search

Pulsierende bidirektionale Suche (Pudil, 1994)

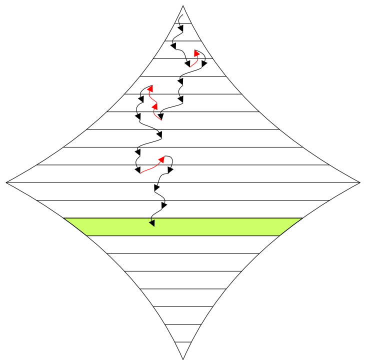
1. INITIALISIERUNG
   Setze $\mathcal{S} \leftarrow \emptyset$, $n=0$, $\iota_0 = \mathrm{J}(\emptyset)$.
2. AUSWAHL AUFWÄRTS
   Bestimme Bestmerkmal

   $$x_d \;=\; \operatorname{argmax}\,\{\mathrm{J}(\mathcal{S},x_d) \mid x_d \notin \mathcal{S}\}$$

   und setze $\mathcal{S} \leftarrow \mathcal{S} \cup \{x_d\}$, n++, $\iota_n \leftarrow \mathrm{J}(\mathcal{S})$.
3. AUSWAHL ABWÄRTS
   Bestimme Schlechtmerkmal

   $$x_d \;=\; \operatorname{argmax}\,\{\mathrm{J}(\mathcal{S}\backslash x_d) \mid x_d \in \mathcal{S}\}$$

   Wenn $\mathrm{J}(\mathcal{S}\backslash x_d) \le \iota_{n-1}$, dann $\rightsquigarrow$ 2.
   Sonst $\mathcal{S} \leftarrow \mathcal{S} \backslash \{x_d\}$, n--, $\iota_n \leftarrow \mathrm{J}(\mathcal{S})$, 3.
4. TERMINIERUNG
   Zielkardinalität n^* oder ι_n-Schranke.



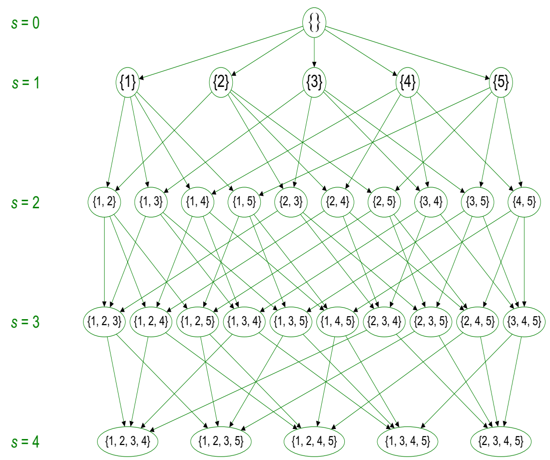
Bemerkung

Effizient und effektiv.

Liefert näherungsweise Bestlösungen $|\mathcal{S}_n| = n$ im Kern des traversierten Bereichs $0 \le n \le n_{\max}$.

A*-Algorithmus

Suche im Nachbarschaftsgraphen aller $\mathcal{S} \subset \mathcal{F}$



Merkmalmengen und Nachbarschaftsrelation

Pro

A* findet die optimale Lösung.
A* findet die n-besten Lösungen.

Contra

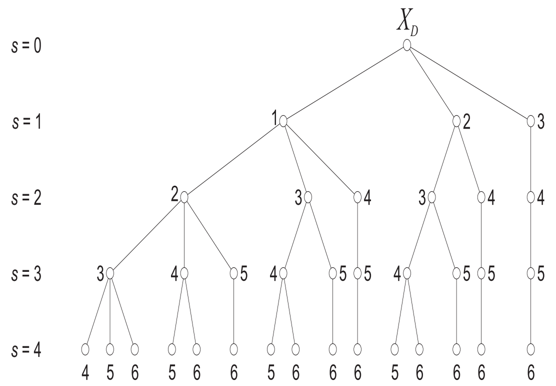
Je nach Schärfe der heuristischen Restschätzung eine aufwändige Suche.

Ausweg

Adaptives Lernen eines **Propheten** für unausgewertete $\mathrm{J}(\mathcal{S})$.

Branch&Bound-Algorithmus

Trivialvariante des A*-Algorithmus · Restschätzung ist $\hat{h}(\cdot) \equiv 0$



B&B-Suchbaum in Top-down-Richtung

Zulässigkeit

Nur garantiert, wenn die Gütefunktion **monoton** ist:

$\mathcal{S}_1 \subseteq \mathcal{S}_2 \Rightarrow \mathrm{J}(\mathcal{S}_1) \leq \mathrm{J}(\mathcal{S}_2)$

Effizienz

B&B ist nur **Top-down** anwendbar.

Hoher $\mathrm{J}(\cdot)$-Aufwand, späte Resultate.

Dualisierte Quadratmittelaufgabe $\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{a}\|^2 \stackrel{!}{\to}$ MIN

Speicheraufwand $\mathrm{O}(T^2)$ und Rechenaufwand $\mathrm{O}(T^3)$

Duale Lösungsdarstellung

als Linearkombination der Objektvektoren:

$$\boldsymbol{a} \;=\; \boldsymbol{X}^\top \boldsymbol{b} \;=\; \sum_{t=1}^{T} b_t \cdot \boldsymbol{x}_t \;, \qquad \boldsymbol{b} \in \mathbb{R}^T$$

Duale Regressionsfehlerformel

in Abhängigkeit vom Vektor $\boldsymbol{b}$ der Lösungskoeffizienten:

$$\varepsilon(\boldsymbol{b}) \;=\; \left\| \boldsymbol{y} - \boldsymbol{X} \cdot \boldsymbol{X}^\top \boldsymbol{b} \right\|^2 \quad \stackrel{!}{\to} \quad \text{MIN}$$

Duale Gauß'sche Normalengleichungen

Lineares Gleichungssystem (Dimension $T \times T$) mit Gram'scher Matrix:

$$\boldsymbol{G}^2 \cdot \boldsymbol{b} \;=\; \boldsymbol{G} \cdot \boldsymbol{y} \;, \qquad \boldsymbol{G} \;=\; \boldsymbol{X} \cdot \boldsymbol{X}^\top \;, \qquad G_{st} \;=\; \langle \boldsymbol{x}_s, \boldsymbol{x}_t \rangle$$

Kombinatorische Regression

Ausgleichsrechnung für Funktionen $f : \mathfrak{P}\mathcal{F} \to \mathbb{R}$ bzw. $f : 2^{\{1,2,\ldots,D\}} \to \mathbb{R}$

Aufgabenstellung

Güteschätzung $\tilde{\mathrm{J}}(\mathcal{S})$ für Merkmalteilmengen $\mathcal{S} \subseteq \mathcal{F} = \{x_1, \ldots, x_D\}$

Termexpansion

Binärattribute: Merkmal-k-Subsets

$$\phi : \mathfrak{P}\mathcal{F} \to \{0,1\}^{\binom{D}{k}}$$

mit

$$\phi_U(\mathcal{S}) = \begin{cases} 1 & U \subseteq \mathcal{S} \\ 0 & U \not\subseteq \mathcal{S} \end{cases}$$

Quadratmittelaufgabe

der Dimension $\binom{D}{k}$ (bzw. 2^D)

Duale Quadratmittelaufgabe

Gramsche n^2-Matrix mit Einträgen

$$K_k(\mathcal{S}_i, \mathcal{S}_j) = \langle \phi(\mathcal{S}_i), \phi(\mathcal{S}_j) \rangle$$

Kombinat. Kernoperator

$$\begin{aligned} K_k(\mathcal{S}, \mathcal{S}') &= \sum_{|U|=k} \phi_U(\mathcal{S}) \cdot \phi_U(\mathcal{S}') \\ &= \sum_{|U|=k} \phi_U(\mathcal{S} \cap \mathcal{S}') \\ &= \binom{|\mathcal{S} \cap \mathcal{S}'|}{k} \quad \text{bzw. } 0 \end{aligned}$$

Dto. für die kumulativen Kerne $K^{\leq k} := K_0 + K_1 + K_2 + \ldots + K_k$

Ein-Klassen-Szenarien

Zwei-Klassen-Szenarien

Mehr-Klassen-Szenarien

Ensemblemethoden — Bagging & Boosting

Kombinatorische Merkmalauswahl

Inkrementelles Lernen

Transduktion

Inkrementelles Lernen

$\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$ ➪ $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{T+1}\}$

Gegeben:

Ein Klassifikator mit den Parametern $\boldsymbol{\theta}^{(T)}$ auf Grundlage der Lerndaten $\omega^{(T)} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$

Gesucht:

Ein schnelles Berechnungsverfahren für die Parameter $\boldsymbol{\theta}^{(T+1)}$ zur erweiterten Probe $\omega^{(T+1)} = \omega^{(T)} \cup \{\boldsymbol{x}_{T+1}\}$

Aktualisierungsformeln für NVK

(Aufzählung der $\boldsymbol{x}_t$ für ein ausgewähltes Ω_κ)

Mittelwertvektor:

$$\boldsymbol{\mu}^{(T+1)} = \frac{T}{T+1} \cdot \boldsymbol{\mu}^{(T)} + \frac{1}{T+1} \cdot \boldsymbol{x}_{T+1}$$

Kovarianzmatrix:

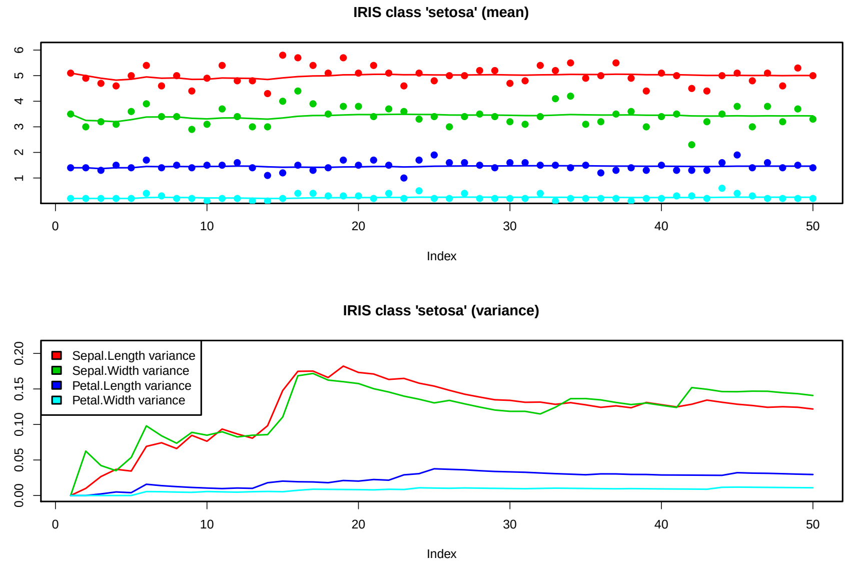
$$\boldsymbol{R}^{(T+1)} = \frac{T}{T+1} \cdot \boldsymbol{R}^{(T)} + \frac{\boldsymbol{x}_{T+1} \cdot \boldsymbol{x}_{T+1}^\top}{T+1}$$

Inverse Kovarianzmatrix: *(via Sherman-Morrison-Woodbury-Formel)*

$$\left(\boldsymbol{A} - \boldsymbol{u}\boldsymbol{v}^\top\right)^{-1} = \boldsymbol{A}^{-1} + \frac{1}{1 - \boldsymbol{v}^\top \boldsymbol{A}^{-1} \boldsymbol{u}} \cdot \boldsymbol{A}^{-1} \boldsymbol{u} \boldsymbol{v}^\top \boldsymbol{A}^{-1}$$

IRIS-Datensatz · Klasse „*setosa*" [1:50]

Inkrementelle Mittelwerte und Varianzen



Gewichtetes inkrementelles Lernen

Gegeben:

Ein Klassifikator mit den Parametern $\boldsymbol{\theta}^{(T)}$ auf Grundlage der Lerndaten

$$\omega^{(T)} = \{(\boldsymbol{x}_t, \beta_t)\}_{t=1}^{T}$$

Gesucht:

Ein schnelles Berechnungsverfahren für die Parameter $\boldsymbol{\theta}^{(T+1)}$ zur erweiterten Probe

$$\omega^{(T+1)} = \left\{(\boldsymbol{x}_t, \tilde{\beta}_t)\right\}_{t=1}^{T+1}$$

Schnelle Aktualisierungsformeln

für „*selbstähnliche*" Gewichtstrukturen

$$\tilde{\beta}_t = \begin{cases} \alpha_t \cdot \beta_t & t \leq T \\ 1 - \alpha_t & t = T+1 \end{cases}, \quad \alpha_t \in (0,1)$$

Gewichteter Mittelwertvektor

$$\boldsymbol{\mu}^{(T+1)} = \alpha_t \cdot \boldsymbol{\mu}^{(T)} + (1 - \alpha_t) \cdot \boldsymbol{x}_{T+1}$$

Beispielgewichtungen

Ungewichtetes Mittel:

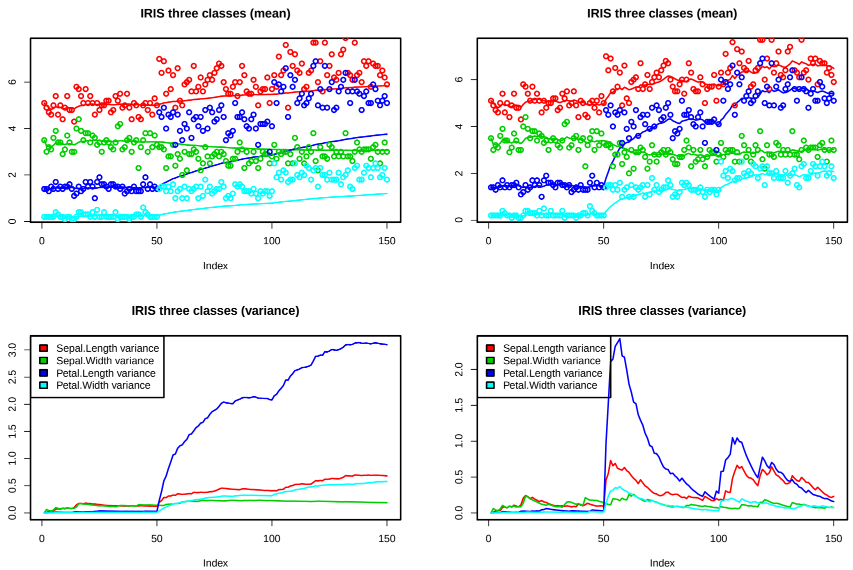
$$\beta_t = {}^1\!/_T\,, \qquad \tilde{\beta}_t = {}^1\!/_{T+1}\,, \qquad \alpha_t \equiv {}^T\!/_{(T+1)}$$

Exponentielles Vergessen:

$$\beta_t = (1-\alpha) \cdot \alpha^{T-t}, \qquad \tilde{\beta}_t = (1-\alpha) \cdot \alpha^{(T+1)-t}$$

IRIS-Datensatz · 3 Klassen [1:50:100:150]

Mittelwerte und Varianzen · Uniforme und exponentielle Gewichte



Zusammenfassung (11)

1. Im Fall eines **Zweiklassenproblems** ist die Gestaltung der **Kostenmatrix** äquivalent zur Festsetzung eines Schwellenwertes für die **logodds-Prüfgröße**.
2. Eine kostenunabhängige Leistungscharakteristik ist die **Receiver-Operator-Kurve**, das einschlägige **skalare** Gütemaß ist die **Wilcoxon-Mann-Whitney**- oder **AUC**-Statistik.
3. **Mehrklassenentscheidungsregeln** lassen sich auf Zweiklassenregeln zurückführen; wir unterscheiden die **Jeder-gegen-jeden**- und die **Jeder-gegen-den-Rest**-Strategie.
4. Das **Bagging**-Ensemble mittelt die Entscheidungen von Klassifikatorinstanzen mit unterschiedlich ausgewürfelten **Bootstrap**-Lernstichproben.
5. Das **Boosting**-Ensemble staffelt seine Entscheidungsregeln nach **erfolgsgewichteten** Lernstichproben.
6. Wettbewerbende Klassifikatorinstanzen werden auf **Prüfgrößenbasis** durch **EM-Interpolation** fusioniert.
7. **Ranker** wählen Merkmale ausschließlich auf Grundlage ihrer **Relevanz** aus.
8. **Filter** streben gleichzeitig die Unterdrückung **redundanter** Merkmalkombinationen an.
9. **Wrapper** suchen die fehlerminimale unter **allen** Merkmalkombinationen; diese NP-harte Aufgabe wird mit heuristischen **Graphsuchverfahren** angegriffen.
10. Rekursionsformeln für **inkrementelle Lernschritte** erlauben dem Klassifikator **lebenslanges Lernen** und **schritthaltende Anpassung** an gleitende Umgebungsbedingungen.