WMuEMaL

Friedrich-Schiller-Universität Jena

Fakultät für Mathematik und Informatik
INSTITUT FÜR INFORMATIK
Prof. E.G. Schukat-Talamazzini

WERKZEUGE MUSTERERKENNUNG & MASCHINELLES LERNEN

Aufgabenblatt 9

(Ausgabe am Fr 15.6.2018 — Abgabe bis So 24.6.2018)

Aufgabe 1 — 8 P

Wir implementieren eine Klasse `parzen` *für eine* ***univariate*** *Parzenschätzung (ME-Skriptum VI.6, Blatt 12–13) mit je einer Gaußglocke* $\mathcal{N}(x \mid z_i, s^2)$ *als Potentialfunktion (Skalenfaktor* s*) für Lernprobenwerte* $z_1, \ldots, z_n \in \mathbb{R}$.

(a) *Schreiben Sie einen Konstruktor* `parzen(x,sigma)`*, der ein Objekt der Klasse* `parzen` *mit Komponenten* `o$support` *und* `o$sigma` *für Lernprobe und Skalenfaktor abliefert.*

(b) *Schreiben Sie eine Abrufmethode* `predict.parzen(o,newdata=NULL)`*, die den Vektor der Dichtewerte des Parzenobjekts* `o` *für die Eingabedaten des Vektors* `newdata` *zurückgibt. Verwenden Sie dafür die 'R'-Implementierung* `dnorm()` *der Gaußdichte!*

(c) *Schreiben Sie eine Funktion* `plot.parzen(o,xlim=?,...)` *zur Grafikdarstellung der Parzendichte* `o` *im Intervall* `xlim`*. Verwenden Sie* `curve()` *und zur Fransendarstellung der Lernprobewerte* $z_1, \ldots, z_n$ *die Funktion* `rug()`*. Die* `xlim`*-Voreinstellung wähle einen sinnvollen Bereich um alle Stützstellen. Den Skalenfaktor* s *platzieren Sie bitte an der Grafiknordseite.*

(d) *Laden Sie jetzt* `parzen.rda` *und zeichnen Sie den Parzendichteverlauf der Datenprobe* `samples` *für alle* `sigma`*-Werte* s^m *mit* $m \in \mathbb{Z}$ *zwischen* 7 *und* -4 *und der Basis* $s = 0.7$ *(2 Grafikseiten im Format* 3×2*).*

(e) *Ergänzen Sie* `predict.parzen`*, so dass im Fall* `newdata=NULL` *der Vektor aller Leave-One-Out-Dichtewerte für die Stützstellen in* `o$support` *berechnet und zurückgegeben werden. (Der Dichtewert für* z_j *wird auf der Basis der Parzendichte mit den Stützstellen* $\{z_1, \ldots, z_n\} \setminus \{z_j\}$ *ermittelt.)*

(f) *Ergänzen Sie* `plot.parzen`*, so dass auch die oben implementierten* L^1O*-Dichtewerte mit* `points()` *in die Grafik einbezogen werden. Wiederholen Sie die Grafikaufrufe aus (d).*

(g) *Ergänzen Sie den Konstruktor* `parzen`*, so dass im Fall* `sigma=NULL` *der Skalenfaktor mit maximaler (logarithmierter!)* L^1O*-Zielgröße (Produkt der* L^1O*-Dichtewerte aller Stützstellen) berechnet und verwendet wird. Realisieren Sie die Maximierung durch einen geeigneten Aufruf der 'R'-Funktion* `optimize()`*. (Die mitgelieferte Variante* `Optimize()` *erzeugt bei Bedarf eine Grafikausgabe des Suchprozesses.)*

(h) Testen Sie Ihre Implementierung mit dem Grafikaufruf `plot(parzen(samples))`.

Abzuliefern ist bitte Ihr Programmcode in `parzen.R`.

Aufgabe 2 — *12 P*

Es ist ein Quadratmittelklassifikator (QMK, ME-Skript VI.5) mit — hinsichtlich einer Polynomtermexpansion — ***linearen*** *Prüfgrößen zu implementieren. Dazu ist im Wesentlichen nur ein passendes* `formula`*-Objekt für den* `lm()`*-Aufruf zu konstruieren.*

(a) Realisieren Sie zunächst eine Hilfsfunktion `cumex(n,dmax)`*, die alle kombinatorisch möglichen Exponentenvektoren mit Gesamtgrad* $d \leq$ `dmax` *zu* `n` *Variablen* $V_1, \ldots, V_n$ *als Zeilen einer Matrix abliefert. Dringende Empfehlung: Rekursion über* `n`*. Testhilfe:* `cumex(8,5)` *ergibt eine* (1287×8)*-Matrix.*

(b) Packen Sie `cumex()` *in eine weitere Hilfsfunktion* `polyterms.RHS(n,dmax,vname)`*, die zu jeder* `cumex(n,dmax)`*-Zeile einen entsprechenden Inhibit-Ausdruck der Art* `I(x^2*y^5*z^0)` *mit jeweiligen Exponenten und Variablennamen aus* `vname` *erzeugt, mit dem* `formula`*-Vereinigungsoperator* `+` *verknüpft und als Zeichenkette (*`character[1]`*-Objekt) zurückliefert.*

(c) Implementieren Sie nun den Konstruktor `QMK(x,dmax)` *eines QM-Klassifikators mit Polynomansatz. Er berechnet ein* `QMK`*-Objekt mit termlinearen Diskriminantenfunktionen für die etikettierten Lerndaten* `x` *auf Basis von Polynomtermen mit Maximalgrad* `dmax`*. Ihr Job ist es lediglich, die* `polyterms.RHS()`*-Ausgabe mit einer geeigneten linken Seite (für die ideale Trennfunktion: 1/0 für die richtige/falsche Klasse) zu versehen und das resultierende* `formula`*-Objekt (Details dazu im WMM-Skript IV.7) an* `lm()` *zu übermitteln.*

(d) Die nächste Funktion `predict.QMK(o,newdata)` *liefert zu gegebenem* `QMK`*-Objekt* `o` *und (nicht etikettierten) Testdaten* `newdata` *einen Faktorvektor mit den vorhergesagten Klassennamen.*

(e) Die Variante `heldout.dim(x,newdata,method=QMK,...)` *der Fehlerauswertungsmethode lerne mit* `x` *und berechne Fehlerraten zu allen (etikettierten) Datensätzen der Liste* `newdata`*. Als erstes Element des abzuliefernden Fehlerratenvektors werde zudem die Anzahl der Polynomterme des Experiments eingetragen; realisieren Sie dazu am besten ein Funktiönchen* `dim.QMK(o)`*.*

(f) Die Funktion `error.table(x,newdata,modelset,...)` *ruft* `heldout.dim` *für alle Polynomgrade aus* `modelset` *auf und gibt die Lern- und Testfehlerraten in Abhängigkeit von der Anzahl der Merkmalterme als Grafik (zwei Kurven) aus und als Tabelle (drei Zeilen) zurück.*

(g) Berechnen Sie nun abschließend die Fehlertabellen für die Datensatzpaare `heart`*,* `diabetes`*,* `australia`*,* `segment` *und* `vehicle` *mit der Einstellung* `modelset=0:3`*.*

(h) Ab welchem Polynomgrad wird die Lerndatenfehlerrate bei `diabetes` *gleich Null? Ausprobieren!*

Abzugeben sind Ihr 'R'-Programmcode in `QMK.R` *und (schriftlich) die Tabellen mit den* $5 \times 1 \times 4$ *Termanzahlen und* $5 \times 2 \times 4$ *Fehlerraten.*

Hinweise zum Übungsablauf

- Die wöchentliche WMM-Vorlesung findet am Mittwoch um 12:15 Uhr statt.
  Das Aufgabenblatt gibt es immer am Freitag (PDF im Netz).
  Der späteste Abgabetermin ist Sonntag 23:59 Uhr.
- Die Übungsaufgaben dürfen natürlich (und sollten sogar) in Gruppenarbeit (2 Mitglieder) gelöst werden.
- Schriftliche Lösungen (*„Textantworten“*) sind als PDF beizufügen oder direkt im e-Mail-Textkörper unterzubringen.
- Alle anderen Lösungen (Programmieraufgaben, Daten und Grafiken) sind als elektronischer Anhang der Lösungs-e-Mail abzuliefern.
- Programmcode (Dateien `*.R`) muss auch wirklich in 'R' ausführbar sein.
  (Kommando `Rscript «name.R»` auf einem der Rechner des FRZ-Pools)
- Ganz wichtig:
  Schriftliche Antworten werden von mir gedruckt, gelesen, kommentiert und korrigiert. Deshalb diese Textteile bitte **niemals** im abgegebenen Programmcode verstecken!
- Je Gruppe und je Aufgabenblatt ist **genau eine** e-Mail zu senden:
  - Vermerk »**WMM**/n« und Gruppenname im `subject`-Feld
    ($n \in \mathbb{N}$ ist die laufende Nummer des Übungsblattes)
  - die Namen der beteiligten Gruppenmitglieder im Textrumpf
  - Tabellen, Bilder, Programmcode, Sensordaten als Attachments
    (elektronische Anlagen)
  - etwaige schriftliche Antworten im Textrumpf der Post oder als Attachment
    (Text/PDF)
- Einige Aufgabentexte verweisen Sie zum Nachschlagen von Details auf das Folienskript zur Vorlesung Mustererkennung; Sie finden es unter der URL
  `http://www.minet.uni-jena.de/fakultaet/schukat/ME/Scriptum/`.
  Die Angabe *ME-Skript II.6* bedeutet: Kapitel II, Abschnitt 6