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 So 23.6.2019 — Abgabe bis So 30.6.2019)

Aufgabe 1 — 8 P

In dieser Aufgabe geht es um die Anpassung von Ausgleichspolynomen (in einer Veränderlichen) mit der 'R'-Funktion `lm()` *durch lineare Regression (ME-Skript VI.5).*

(a) Laden Sie den Datensatz `mydata` *(#1=Quellvariable und #2=Zielvariable) aus der Datei* `unipoly.rda` *und lesen Sie die Dokumentation zu* `lm()` *und* `formula`*-Objekten.*

(b) Schreiben Sie eine Funktion `polyfun(x,a)`*, die für alle Einträge des Vektors* `x` *den Funktionswert des Polynoms mit Koeffizienten* `a` *(aufsteigend als* $a_0, \ldots, a_n$ *gespeichert) berechnet und als Vektor zurückgibt.*

(c) Schreiben Sie eine Funktion `polyfit(xy,n)`*, die für den Datensatz* `xy` *ein Ausgleichspolynom* `n`*-ten Grades zur Vorhersage des zweiten aus dem ersten Attribut berechnet und als Regressionsobjekt (Klasse* `lm`*) zurückliefert. Rufen Sie dazu* `lm()` *mit einer geeigneten Modellformel auf.*

(d) Schreiben Sie eine Funktion `polyfits(xy,deg,plot=FALSE)`*, die eine Liste der Ausgleichspolynomobjekte für* `xy` *zu allen Polynomgraden in* `deg` *abliefert. Im Fall* `plot=TRUE` *extrahiert sie das Akaike- und das Bayes-Informationskriterium aller Modelle (Funktionen* `AIC()` *und* `BIC()`*) und trägt die Werte in einer gemeinsamen Grafik über den Polynomgraden auf.*

(e) Schreiben Sie eine Funktion `polyplot(o,xy)` *zur Erzeugung einer Grafik mit (1) dem Datensatz* `xy` *als Punktwolke, (2) dem Funktionsverlauf des Ausgleichspolynoms* `o` *und (3) seinem BIC-Wert und (4) seinen Polynomkoeffizienten (gerundet) als Texteintrag.*

(f) Schreiben Sie jetzt noch eine Funktion `polyplots(xy,deg=0:11)`*, die zum Datensatz* `xy` *mittels* `polyfits`*-Aufruf Grafik und Polynomobjektliste aus (d) erzeugt und anschließend alle gelisteten Polynome mittels* `polyplot` *auf* (2×2)*-Leinwände zeichnet.*

(g) Nun wenden Sie bitte `polyplots` *auf die Datensätze (1)* `mydata`*, (2)* `cars` *und (3) den fünfspaltigen* `LifeCycleSavings` *an, im letzten Fall für alle sechs Kombinationen der drei Attribute* `pop15`*,* `pop75` *und* `dpi` *(insgesamt* $(1+1+6) \cdot (1+3) = 32$ *Grafikseiten).*

Abzugeben ist die Datei `unipoly.R` *mit Ihrem Programmcode.*

Aufgabe 2 — *12 P*

Wir schreiben 'R'-Funktionen zur Klassifikation mit einem ***Mehrschichtenperzeptron*** *(MLP; ME-Skript VIII.5).*

(a) Laden Sie per Kommando `library(nnet)` *das 'R'-Paket zum Lernen und Testen von MLPs mit* ***einer*** *verborgenen Schicht künstlicher Neuronen. Studieren Sie die Beschreibungstexte zur Lernmethode* `nnet` *und zur Vorhersagemethode* `predict.nnet`*. Beachten Sie die Aufrufbeispiele, die Hinweise zur Gestaltung von* `formula`*-Objekten und die Informationen zur Verwendung des MLP zu Klassifikationszwecken (Zielvariable=*`factor`*).*

(b) Schreiben Sie eine Methode `SHLP(x,hidden,rescale=FALSE)` *zum Lernen eines MLP mit H=*`hidden` *verborgenen Neuronen aus den etikettierten Daten* `x`*. Nutzen Sie dazu die* `nnet`*-Methode unter Beibehaltung aller Defaulteinstellungen.*

(c) Schreiben Sie eine Methode `predict.SHLP(o,newdata)` *zur Vorhersage der Klassennamen (*`factor`*!) für die nicht etikettierten Daten* `newdata`*. Der 'R'-Code passt quasi in eine Zeile!*

(d) Wiederbeleben Sie Ihre Funktion `heldout()` *(Aufgabe 7/2c) und tätigen Sie nun einige Testaufrufe. Die erbrachten Resultate wären leider nicht reproduzierbar, weil die Startgewichte vom Lernverfahren* `nnet` *in Werkseinstellung mit Zufallszahlen vorbesetzt werden. Korrigieren Sie diesen Missstand, indem Sie* `nnet` *mittels Aufrufargument* `Wts = cos(1:m*883)` *von den Vorteilen deterministisch vorgegebener Startwerte überzeugen. Damit das auch reibungslos funktioniert, müssen Sie allerdings die Anzahl* `m` *zu lernender MLP-Gewichte kennen . . .*

(e) Entwickeln Sie daher eine Formel $m = \rho(D, H, K)$ zur Berechnung der Gewichteanzahl m aus der Merkmaldimension D, der Anzahl H verborgener Neuronen und der Anzahl K der Musterklassen. (Vergessen Sie nicht die konstanten Schwellwertneuronen und die außerplanmäßige Modellierung von Zweiklassenproblemen!)

(f) Sobald die reproduzierbare Version funktioniert, berechnen Sie bitte die Test- und die Reklassifikationsfehlerraten für die Datensätze `diabetes`*,* `heart`*,* `vehicle` *und* `segment` *für MLPs mit $H \in \{1, 2, 3, 5, 8, 13, 21\}$ verborgenen Neuronen.*

(g) Künstliche Neuronale Netze sind dafür berüchtigt, dass sie sensibel auf die Skalierung ihrer Eingabedaten reagieren. Nutzen Sie den Schalter `rescale`*, um eine Verfahrensvariante zu realisieren, die Lern- und Testdaten durch eine* ***gemeinsame*** *Lineartransformation standardisiert, welche die Lerndaten merkmalweise auf das $[-1, +1]$-Intervall abbildet.*
TIPP: Verankern Sie die Normierungoperation als 'R'-Funktionsobjekt in der `SHLP`*-Klasse!*

(h) Wiederholen und tabellieren Sie nun die obige Testreihe. Erzeugen Sie aus Ihren Resultaten je Datensatz eine Grafik mit den vier Fehlerkurven (Lernfehler/Testfehler × rohe/normierte Daten).

Abzugeben sind bitte der 'R'-Programmcode `SHLP.R` *sowie die Gewichtanzahlformel (e) und die 2 Tabellen aus (f,h) als schriftliche Lösungskomponenten.*

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