

Blatt 5: Crossvalidation

Abgabe Montag, 4. Dezember 2000, in der Vorlesung

7. *Cross-validation*: In dieser Aufgabe soll mit *cross-validation* eine Technik implementiert werden, die es erlaubt, die Klassifikationsleistung abzuschätzen, wenn nur wenig Daten insgesamt zur Verfügung stehen. Schreiben Sie dazu eine Matlab-Funktion

```
function error = CV(SAMPLE, CLASS, K, FUNKTION)
```

die den in der Vorlesung besprochene K -fold cross validation Algorithmus implementiert. Dazu folgende Bemerkungen:

- Verwechseln Sie nicht das k des k -NN Klassifikators mit dem Parameter K der K -fold cross validation.
 - Unterteilen Sie die Trainingsdaten (`SAMPLE`, `CLASS`) in K gleich große Datensätze.
 - Benutzen Sie $K - 1$ Datensätze, um den Klassifikator zu trainieren und schätzen Sie die Klassifikationsleistung auf dem verbleibenden Datensatz.
 - Wiederholen Sie diese Prozedur, in dem Sie jeden Datensatz genau einmal hernehmen, um die Klassifikationsleistung zu evaluieren. Geben Sie als Schätzung `error` für den Klassifikationsfehler die mittlere Klassifikationsleistung zurück. Dieser wird im folgenden mit $CV(K, n)$ bezeichnet.
 - Benutzen Sie den Übergabeparameter `FUNKTION`, um den verwendeten Klassifikator auszuwählen. Maximale Flexibilität erreichen sie, wenn sie dabei die Matlab-Funktion `eval()` verwenden und in `FUNKTION` nicht nur den Funktionswert, sondern auch die Übergabeparameter des Klassifikators übergeben. Funktionspointer sind unter Matlab nicht implementiert. Alternativ können sie auch eine `if` Konstruktion verwenden und die verwendeten Klassifikatoren direkt kodieren.
- (a) Benutzen Sie zum Testen den in Aufgabe 3 (a) erzeugten Datensatz. Verwenden Sie $n = 100$ Daten. Schätzen Sie nun mittels K -fold cross validation ($CV(K, 100)$) die Klassifikationsleistung des 1-NN Klassifikators und des *Plug-in* Klassifikators, wie er in Aufgabe 5(a) implementiert wurde. (*Plug-in* Klassifikator deshalb, da der Bayes' Klassifikator mit eingesetzten (*plugged in*) empirisch geschätzten Parametern verwendet wird). Plotten Sie die Schätzwerte für die Klassifikationsleistung in Abhängigkeit von $K \in \{10, 20, 50, 100\}$. **(3P)**
- (b) Wiederholen Sie diese Prozedur mit 20 verschiedenen Datensätzen der Größe $n = 1000$ für den 1-NN Klassifikator und schätzen Sie damit den Mittelwert und die Standardabweichung von $CV(K, 100)$. Plotten sie Mittelwert und Standardabweichung in Abhängigkeit von K . **(3P)**
8. *Bootstrap*: In dieser Aufgabe soll der Vorhersagefehler des in Aufgabe 5 (a) eingeführten *Plug-In*-Klassifikators mit Hilfe der *Bootstrap*-Methode abgeschätzt werden. Die Ausgangslösung hierzu ist, zunächst von einer gegebenen Menge von n Datenpunkten (*Original-Sample*) mit Zurücklegen B sog. *Bootstrap-Samples* zu ziehen, die ebenfalls jeweils n Datenpunkte enthalten. Der zu untersuchende Klassifikator wird auf jedem der B Bootstrap-Samples einzeln trainiert und jeweils anschließend auf dem Original-Sample getestet.

Durch eine Mittelung über die dabei gemessenen B -viele Klassifikationsfehler erhält man eine Abschätzung des allgemeinen Klassifikationsfehlers.

Eine verfeinerte Variante dieses Algorithmus bestimmt außerdem jedes Mal den Klassifikationsfehler auf dem Bootstrap-Sample, mit dem der Klassifikator gerade trainiert wurde, und berechnet seine Differenz zum (in der Regel größeren) Klassifikationsfehler auf dem Original-Sample. Diese Größe mißt den sog. *Optimismus* des Bootstrap-Schätzers und wird zum auf dem Original-Sample gemessenen Klassifikationsfehler addiert. Wiederum erhält man durch eine Mittelung über alle auf diese Weise gemessenen Klassifikationsfehler eine Abschätzung des allgemeinen Klassifikationsfehlers.

Erzeugen Sie die folgenden graphischen Darstellungen:

- (a) Verwenden Sie noch einmal den in Aufgabe 3 (a) erstellten Datensatz. Ziehen Sie zufällig ein Original-Sample von $n = 200$ Datenpunkten und verwenden Sie $B = 100$ Bootstrap-Samples, um den Klassifikationsfehler des Plug-In-Klassifikators abzuschätzen. Berechnen Sie mit jedem neu bearbeiteten Bootstrap-Sample eine aktualisierte Schätzung des Klassifikationsfehlers, die sie in einer Graphik gegen B auftragen. Arbeiten Sie dabei sowohl mit der Grund- als auch mit der verfeinerten Bootstrap-Variante, und verwenden Sie für beide Verfahren eine gemeinsame Graphik. Trainieren Sie den Plug-In-Klassifikator außerdem auf dem Original-Sample und testen Sie ihn auf den bisher nicht verwendeten $(10.000 - n)$ Datenpunkten des Datensatzes. Tragen Sie den dort gefundenen Klassifikationsfehler als konstante Funktion ebenfalls in die Graphik ein. **(4P)**
- (b) Variieren Sie nun die Größe n des Original-Samples, indem sie mit $n = 50$ Datenpunkten beginnen und die Kardinalität dann schrittweise auf $n \in \{50, 100, 200, 500, 1000\}$ Datenpunkte erweitern. Vergleichen Sie den von der verfeinerten Bootstrap-Variante ($B = 100$) vorhergesagten Klassifikationsfehler mit dem durch n -fache Crossvalidierung (Aufgabe 7) ermittelten Wert, und tragen Sie beide in einer gemeinsamen Graphik gegen n auf. Fügen Sie der Graphik wiederum den Klassifikationsfehler hinzu, den man erhält, wenn man den Plug-In-Klassifikator auf dem Original-Sample trainiert und ihn anschließend auf die verbleibenden $10.000 - n$ Datenpunkte anwendet. **(4P)**