Einführung in die Empirische Wirtschaftsforschung

Übungsaufgaben 6
(Softwareübung 1)

Vorhersage - Kennzahlen eines Modelles - Regression und KNN

1. Betrachten Sie das Datenset "luft.csv" in R.

   a) Regressieren Sie "Ozon (max)" (Y) auf "Lufttemperatur" (X). Interpretieren Sie den Output (Koeffizienten und Standardfehler) und diskutieren Sie die Plausibilität der Annahmen des linearen Modelles.
   Tipp: Nach dem Sie die Daten geplottet haben, können Sie ganz einfach mittels der R Funktion `abline(object)` Ihre Regressionsgerade darstellen - wobei `object` Ihr geschätztes Modell darstellt.

   b) Schätzen Sie ein RegressionsModell welches die in a) identifizierten Probleme behebt und zeigen Sie das neue Modell in einem Plot. Verwenden Sie dieses Modell für den Rest der Aufgaben.

   c) Berechnen Sie in R ein Konfidenzintervall für μ_{neu}, wenn $x_{neu} = \bar{x}$.

   d) Berechnen Sie in R ein Vorhersageintervall für y_{neu}, wenn $x_{neu} = \bar{x}$. Überprüfen Sie die Plausibilität der Annahme der normalverteilten Fehlerterme auf geeignete Art und Weise.
   Tipp: Um mit einem geschätzten RegressionsModell eine Vorhersage zu machen verwenden Sie am besten die generische R Funktion `predict(object, newdata)` - wobei `object` Ihr geschätztes Modell darstellt.

   e) Vergleichen Sie die Intervalle aus c) und d). Was stellen Sie fest?

   f) Berechnen Sie in R ein 90% Vorhersageintervall für y_{neu}, wenn $x_{neu} = 150$. Würden Sie dieser Vorhersage vertrauen?

2. In dieser Aufgabe verwenden wir den Datensatz "nlsy.csv". Die Variablen von Interesse sind "lnearn" und "iq". Wir gehen von folgender linearen Einfachregression aus

$$lnearn_i = \beta_0 + \beta_1 iq_i + u_i, \quad i = 1, \ldots, N.$$

   a) Unterteilen Sie die Daten zufällig in ein Training Datenset und ein Testing Datenset. Verwenden Sie (1/3) der Daten fürs Testing und (2/3) der Daten um das Modell zu schätzen. Um Ihre Ergebnisse replizierbar zu machen, setzen Sie in R den Seed 420 (`set.seed(420)`) vor dem zufälligen Aufteilen der Daten in Training und Testing.
   Tipp: Um zufällig eine gewisse Anzahl an Beobachtungen aus einem Datensatz zu ziehen (mit oder ohne Zurücklegen), können Sie die R Funktion
   `sample(x, size, replace = FALSE)` verwenden.

b) Regressieren Sie "lnearn" auf "iq". Verwenden Sie dazu lediglich die Beobachtungen im Training Datenset. Ist die Verwendung einer linearen Einfachregression angebracht?

c) Berechnen und vergleichen Sie jeweils die Kennzahlen Train MSE und Test MSE, sowie Train R^2 und Test R^2. Wieso sollte man mit der Analyse des Train MSE und des Train R^2 vorsichtig umgehen?

d) Was passiert mit den 4 Kennzahlen, wenn Sie nur 50 Beobachtungen im Training verwenden (die restlichen 885 fürs Testing). Setzen Sie wiederum in R den Seed 420 (`set.seed(420)`) vor dem zufälligen Aufteilen der Daten in Training und Testing.

3. Wie in der obigen Aufgabe betrachten wir den Datensatz "nlsy.csv". Wir lockern etwas die Annahmen von Aufgabe 2 und gehen nun von der folgenden, nicht zwingend linearen Beziehung zwischen "lnearn" und "iq", aus

$$lnearn_i = f(iq_i) + \epsilon_i, \quad i = 1, \ldots, N.$$

a) Unterteilen Sie wie in Aufgabe 2 die Daten zufällig in ein Training Datenset und ein Testing Datenset. Verwenden Sie (1/3) der Daten fürs Testing und (2/3) der Daten um das Modell zu schätzen. Um Ihre Ergebnisse replizierbar zu machen, setzen Sie in R den Seed 420 (`set.seed(420)`) vor dem zufälligen Aufteilen der Daten in Training und Testing.

b) Schätzen Sie die Beziehung zwischen "lnearn" und "iq" auf dem Training Datensatz mittels K-nearest-neighbors (KNN). Verwenden Sie dazu die R Funktion
`knnreg(y ~ x, data, subset, K)` vom `caret` Package. Probieren Sie verschiedene Werte für K aus.
Tipp: Um die Funktionen eines Packages zu benutzen, installieren Sie zuerst das Package mit der R Funktion `install.packages('caret')` und rufen Sie anschliessend mit `library(caret)` das Package auf. Alternativ können Sie auch die R Funktion `require(.)` benutzen.

c) Erstellen Sie ein Streuungsdiagramm und plotten Sie die geschätzte Funktion $\hat{f}$.
Tipp: Im Gegensatz zu der `lm` Funktion können Sie bei `knn` nicht `abline` benutzen. Verwenden Sie `predict` um die Vorhersagen zu erhalten und anschliessend `lines` um die vorhergesagten Werte im Streuungsdiagramm darzustellen. Mittels der Funktion `order` stellen Sie sicher, dass die Trainings Beobachtungen des Prediktors ("iq") und die vorhergesagten Werte die gleiche Reihenfolge haben.

d) Berechnen und vergleichen Sie jeweils Training MSE und Test MSE, sowie Training R^2 und Test R^2.

e) Finden Sie nun heraus für welchen k-Wert der Test MSE minimiert ist.
Tipp: Schreiben Sie eine Funktion die für gegeben Daten und k eine KNN Regression auf den Trainingsdaten berechnet und den Training und Test MSE ausgibt. Verwenden Sie diese Funktion in einem "loop", um den MSE für unterschiedliche k-Werte zu erhalten. Plotten Sie nun den Trainings und Test MSE und finden Sie k-Wert mit dem geringsten Test MSE.

f) Wie würden Sie in der Praxis K wählen? Gehen Sie bei Ihrer Argumentation auf den MSE und das R^2 ein.

4. Vergleichen Sie die beiden Modelle aus den Aufgaben 2 und 3. Welches Modell würden Sie in der Praxis bevorzugen? Begründen Sie Ihre Wahl anhand der Ergebnisse und der theoretischen Eigenschaften der beiden Modellen.