Projekt 1: Överanpassning, LASSO och korsvalidering

2 april 2019

Flerdimensionell linjär regression.

I vanlig endimensionell linjär regression utan konstantterm jobbar man med modellen

$$y_i = b_1 x_i + \epsilon_i,\ i = 1, \ldots, n,$$

där ϵ_i:na är oberoende och $N(0, \sigma^2)$-fördelade, för en okänd varians σ^2, (x_i, y_i) är parvisa datapunkter och b_1 är en okänd parameter. I flerdimensionell linjär regression är x_i en p-dimensionell vektor, $\mathbf{x}_i = (x_{1i}, x_{2i}, \ldots, x_{pi})^T$. Man har datapunkter $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, n$ och tror att y:na kan vara linjärt beroende av samtliga kovariater, dvs samtliga x-koordinater. Man ansätter då modellen

$$y_i = b_1 x_{1i} + b_2 x_{2i} + \ldots + b_p x_{pi} + \epsilon_i,\ i = 1, \ldots, n.$$

Detta skrivas mer kortfattat som

$$y_i = \mathbf{b}^T \mathbf{x}_i + \epsilon_i,$$

där $\mathbf{b} = (b_1, \ldots, b_p)^T$, och ännu kortare som

$$\mathbf{y} = \mathbf{X}\mathbf{b} + \epsilon$$

där $\mathbf{y} = (y_1, \ldots, y_n)^T$, $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^T$ och $\mathbf{X}$ är $n \times p$-matrisen $\mathbf{X} = [\mathbf{x}_1\, \mathbf{x}_2\, \ldots\, \mathbf{x}_p]$. Vi har då p parametrar som ska skattas. Om man skattar enligt ML-principen får man, helt analogt med det endimensionella fallet, att skattningarna $\hat{\mathbf{b}}$ fås genom att minimera

$$\sum_{i=1}^{n} (y_i - \mathbf{b}^T \mathbf{x}_i)^2$$

som funktion av $\mathbf{b}$. I traditionella tillämpningar är antalet datapunkter n betydligt större än p och det är inga problem med denna skattningsmetod. I moderna tillämpningar är det dock vanligt att antalet parametrar som man vill skatta är många fler än antalet datapunkter, dvs p är betydligt större än n. Men datan innehåller inte tillräckligt mycket information för att skatta så många parametrar och risken är stor att den skattade modellen skulle prediktera nya data dåligt. Vi måste alltså på något sätt begränsa parametrarna, samtidigt som vi inte vet vilka av dem som är mest intressanta. Man skulle kunna tänka sig många olika metoder för att göra det och vid val av sådan metod måste man ta hänsyn både till hur väl resultatet predikterar nya data och hur beräkningsmässigt komplex metoden är. En metod som blivit mycket populär är så kallad LASSO-regression. Istället för att använda minsta kvadratanpassningen rakt av, minimerar man i stället

$$l(\mathbf{b}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \mathbf{b}^T \mathbf{x}_i)^2 + \lambda \sum_{r=1}^{p} |b_r|,$$

där $\lambda > 0$ är ett förspecificerat tal, en s.k. hyperparameter. Ju mindre värde på λ, desto mer "straffas" modellen för att välja b_r med alltför stora belopp. En finess med LASSO är att när $l(\mathbf{b})$ minimeras blir väldigt många b_r noll, dvs ett slags rensning bland alla parametrar sker.

Standardiserade kovariater. Det kan uppstå ett problem i och med det att olika kovariater mäts i olika skalor. Därför kan somliga b_r vara mer intressanta än andra trots att deras belopp är mindre. De riskerar då att bli skattade till 0 på grund av skalan. Därför brukar man standardisera kovariaterna genom att subtrahera medelvärdet och sedan skala så att stickprovsvariansen för kovariatat r blir 1. Resultatet blir alltså att vi jobbar med data för vilka

$$\sum_{i=1}^{n} x_{ri} = 0, \sum_{i=1}^{n} x_{ri}^2 = 1$$

för alla $j = r, \ldots, p$.[1] Var uppmärksam på det inte är $\mathbf{x}_i$-vektorerna som standardiseras, utan koordinaterna, dvs "tvärs över" $\mathbf{x}_i$-vektorerna.

Resultatet av LASSO-regressionen beror förstås på λ, så hur ska λ väljas? Meningen med att använda LASSO var ju att få en modell som predikterar nya data bra. Men denna tanke som ledstjärna brukar man då använda s.k. *korsvalidering*. Dela in data på måfå in i G stycken lika stora mängder $D_1, \ldots, D_G$ (Standard är att ta $G = 10$, men andra värden går förstås också bra.) Fixera ett λ. För varje $g = 1, \ldots, G$, plocka bort all data i D_g och gör en LASSO-regression baserad på all övrig data. Detta ger en skattning $\hat{\mathbf{b}}_g$. Beräkna sedan $S_g = \sum_{i \in D_g} (y_i - \hat{\mathbf{b}}_g \mathbf{x}_i)^2$. Beräkna $S(\lambda) = \sum_{g=1}^{G} S_g$. Välj till sist det λ som minimerar $S(\lambda)$. Eftersom man naturligtvis inte får ett specifikt funktionsuttryck för $S(\lambda)$, så sker det sista steget genom lämplig prövning.

Er uppgift är nu att genomföra detta med $p = 10^4$, $n = 10^2$ och $G = 10$. Ni ska simulera er egen datamängd. Välj b_r:en oberoende och Laplacefördelade med parameter 1. Välj sedan ut lämpliga standardiserade $\mathbf{x}_i$:n och simulera datapunkter y_i med σ^2 satt till 1. Gör nu LASSO-regression på denna datamängd, med korsvalidering enligt ovan. Ni kan göra detta antingen genom att skriva kod själva eller använda färdiga funktioner i Matlab.

Bayesiansk tolkning. Att hänga på termen $\lambda \sum_{r=1}^{p} |b_r|$ är en aning godtyckligt, men det finns ett Bayesianskt synsätt som resulterar i det resultatet. Betrakta $\mathbf{b}$ som en (n-dimensionell) stokastisk variabel $\mathbf{B} = (B_1, \ldots, B_p)$ med en àpriorifördelning sådan att B_r:en är oberoende och Laplacefördelade med parameter $n\lambda/2$. Laplacefördelningen, som också kallas dubbelt exponential är en exponentialfördelning med tecken. Tätheten för Laplacefördelningen med parameter τ ges av $(\tau/2)e^{-\tau|t|}$, $t \in \mathbb{R}$. Visa att àposterioritätheten $f_{\mathbf{B}|\mathbf{Y}}(\mathbf{b}|\mathbf{y})$ är maximal för de b_r som minimerar $l(b_1, \ldots, b_p)$, dvs denna Bayesianska modell ger på detta sätt upphov till LASSO.

Betrakta slutligen vad som händer om man som àprioritäthet för B_r använder $N(0, 1/(n\lambda))$ istället för Laplacefördelning och skattar $\mathbf{b}$ analogt med ovanstående. Vilken blir den målfunktion som ska minimeras i detta fall? Vad kallas denna typ av regression?

[1] Om ni undrar varför konstanttermen utelämnats, så är det pga att den "försvinner" när man arbetar med standardiserade kovariater.