Viikko 3: Lineaarista regressiota ja luokittelua

Matti Kääriäinen

matti.kaariainen@cs.helsinki.fi

Exactum C222, 12.-14.11.2008.

Tällä viikolla

Sisältösuunnitelma:

- Lineaarinen regressio
  - Pienimmän neliösumman menetelmä
- Lineaarinen luokittelu
  - Perceptron-algoritmi

Lineaarinen regressio

- Tarkastellaan tilannetta, jossa $\mathcal{X} = \mathbb{R}^d$, $\mathcal{Y} = \mathbb{R}$, ja tavoitteena on ennustaa hyvin tappiofunktion L_2 (neliövirhe) suhteen.
- Lineaarisessa regressiossa ennustajat ovat muotoa

$$f_w(x) = x \cdot w,$$

  missä $w \in \mathbb{R}^d$ on ennustajan f_w parametrivektori ja $\cdot$ tarkoittaa pistetuloa.
- On siis etukäteen päätetty, että koneoppijan tuottaman ennustajan täytyy olla f_w jollakin $w \in \mathbb{R}^d$. Koneoppijan tehtäväksi jää löytää hyvä w.

Lineaarisista ennustajista ja piirteistä

- Lineaarista regressiota kutsutaan lineaariseksi, koska kuvaus $w \mapsto f_w(x)$ on lineaarinen eli ennusteet $f_w(x)$ riippuvat parametrivektorista w lineaarisesti.
- Syötteen piirteet voivat kuitenkin olla epälineaarisia, joten funktiot $x \mapsto f_w(x)$ eivät välttämättä ole taustalla olevien "oikeiden" syötteiden (eli syötteiden josta piirteet on laskettu) suhteen lineaarisia.
- Piirteisiin lisätään usein biakseksi kutsuttu komponentti, joka saa kaikilla syötteillä arvon 1. Näin lineaarisessa mallissa ei tarvitse huolehtia erikseen vakiotermistä.

Lineaarinen regressio koneoppimistehtävänä

- Lineaarisessa regressiossa koneoppijan tehtävänä on löytää opetusdatan perusteella lineaarinen regressiofunktio f_w, jonka odotusarvoinen neliövirhe on pieni.
- Seuraavassa esitetään klassinen “pienimmän neliösumman menetelmä” tulkittuna empiirisen riskin minimointi -periaatteen sovellukseksi lineaariregressioon.
- “pienimmän neliösumman menetelmä” voidaan tulkita (ja usein tulkitaankin) toisin (esim. suurimman uskottavuuden päättelynä), mutta tästä ei enempää kalvoilla. . .

Empiirisen riskin minimointi lineaarisessa regressiossa

- Määritellään empiirinen tappiofunktio $\hat{L}\colon \mathbb{R}^d \to \mathbb{R}$ seuraavasti:

$$\hat{L}(w) = \frac{1}{n}\sum_{i=1}^{n} L(y_i, f_w(x_i)) = \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i \cdot w)^2.$$

- Siis $\hat{L}(w)$ on ennustajan f_w opetusvirhe, kun tappiofunktiona on neliövirhe.
- Sovelletaan parametrivektorin w oppimiseen empiirisen riskin minimointia: Valitaan parametrivektoriksi $\hat{w} = \arg\min_w \hat{L}(w)$.
- Jäljellä siis enää yo optimointiongelman ratkaiseminen.

Funktion $\hat{L}(w)$ minimointi

- Miten löytää w, jolle

$$\hat{L}(w) = \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i \cdot w)^2$$

  on minimaalinen?

- Kun $d = 1$, $\hat{L}(w)$ on tavallinen lukiosta tuttu ylöspäin aukeava paraabeli, jonka globaalin minimin etsiminen onnistuu derivoimalla ja ratkaisemalla yhtälö $\hat{L}'(w) = 0$.

- Seuraavassa “osoitetaan”, että samankaltainen strategia puree myös moniulotteisessa tapauksessa $d > 1$. Tavoitteena täsmällisyyden sijaan välittää päättelyn pääideat yksityiskohtiin takertumatta. . .

Monen muuttujan analyysistä

- Kun $d > 1$, joudutaan turvautumaan monen muuttujan analyysiin.
- Derivaatan $\hat{L}'$ moniulotteinen "vastine" on gradientti $\nabla \hat{L} \colon \mathbb{R}^d \to \mathbb{R}^d$, joka voidaan esittää funktion $\hat{L}$ osittaisderivaattojen

$$\frac{\partial \hat{L}}{\partial w^k} : \mathbb{R}^d \to \mathbb{R}, \quad k = 1, \ldots, d$$

avulla seuraavasti:

$$\nabla \hat{L} = \left(\frac{\partial \hat{L}}{\partial w^1}, \ldots, \frac{\partial \hat{L}}{\partial w^d}\right).$$

- Geometrisesti funktion gradienttia voi ajatella vektorina, joka osoittaa suuntaan, johon siirryttäessä funktio kasvaa nopeimmin.

Gradientti ja ääriarvot

- Jos funktion gradientti pisteessä $w \in \mathbb{R}^d$ ei ole nollavektori, funktio kasvaa suunnassa johon gradientti osoittaa. Siispä w ei voi olla funktion ääriarvokohta.
- Erityisesti funktion $\hat{L}$ minimi voi löytyä vain pisteestä, jossa $\hat{L}$:n gradientti on nollavektori.
- Funktion $\hat{L}$ erityispiirteistä (konveksius) johtuen jokainen gradientin nollakohta on $\hat{L}$:n globaali minimi.
- Minimointiongelman ratkaisemiseksi riittää siis etsiä w, jolle $\nabla \hat{L}(w) = 0$.

Gradientin laskemisesta

- Gradientin laskemiseksi pisteessa $w \in \mathbb{R}^d$ riittää laskea osittaisderivaatat w:ssä.
- Osittaisderivaatta $\frac{\partial \hat{L}}{\partial w^k}$ pisteessä $w = (w^1, \ldots, w^d) \in \mathbb{R}^d$ on yhden muuttujan funktion

$$w^k \mapsto \hat{L}(w^1, \ldots, w^k, \ldots, w^d)$$

derivaatta. Tämä funktio saadaan kiinnittämällä parametrit w^i, $i \neq k$, pisteen w koordinaatteihin, jolloin jäljelle jää w^k:sta riippuva yhden muuttujan funktio, joka voidaan derivoida lukiosta tutuin keinoin.

Funktion $\hat{L}$ osittaisderivaatat ja gradientti

- Lasketaan nyt funktion $\hat{L}(w) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{d} x_i^j w^j)^2$ osittaisderivaatat.
- Saadaan (taululla):

$$\frac{\partial \hat{L}}{\partial w^k} = -\frac{2}{n}\sum_{i=1}^{n} y_i x_i^k + \frac{2}{n}\sum_{j=1}^{d}\Big(\sum_{i=1}^{n} x_i^j x_i^k\Big) w^j$$

- Yhdistämällä nämä ja siirtymällä matriisinotaatioon nähdään, että

$$\nabla \hat{L}(w) = -\frac{2}{n} X^T Y + \frac{2}{n} X^T X w^T.$$

Tässä X on $n \times d$ matriisi, jonka rivillä i on x_i, ja Y on $n \times 1$ sarakevektori, jonka rivillä i on y_i.

Gradientin $\nabla \hat{L}$ nollakohdan etsiminen

- Pisteet w, joissa $\nabla \hat{L}(w) = 0$, löytyvät siis ratkaisemalla lineaarinen yhtälöryhmä $X^T X w^T = X^T Y$.
- Yhtälöryhmä voidaan ratkaista esim. Gaussin eliminaatiomenetelmällä (katso `Matlab`issa komento $\backslash$ eli `mldivide`).
- Voidaan osoittaa, että ratkaisu w on aina olemassa, mutta ei välttämättä yksikäsitteinen – monella eri ennustajalla f_w voi siis olla minimaalinen opetusvirhe.
- Jos kuitenkin x_i:t virittävät koko syöteavaruuden $\mathcal{X} = \mathbb{R}^d$, ratkaisu on aina yksikäsitteinen. Tällöin $X^T X$ on kääntyvä, ja yhtälöryhmän ratkaisu löytyy kertomalla yhtälö puolittain vasemmalta $X^T X$:n käänteismatriisilla $(X^T X)^{-1}$.

Regularisoinnista

- Edellä on selvitetty, kuinka lineaarista regressiota voidaan ajatella empiirisen riskin minimointina, ja kuinka näin syntyvä pienimmän neliösumman ongelma voidaan ratkaista.
- Jäljellä on kuitenkin kaksi potentiaalista ongelmaa: Ratkaisun yksikäsitteisyys ja epästabiilius, sekä ylisovittaminen.
- Samoihin ongelmiin törmätään toistuvasti sovellettaessa empiirisen riskin minimointia, ei siis vain lineaariregressiossa.
- Molempia ongelmia voidaan yrittää ratkaista regularisoidun empiirisen riskin minimoinnin avulla:
  - Minimoidaan pelkän opetusvirheen sijaan opetusvirheen ja “hyviksi oletettuja” parametrivektoreita/ennustajia suosivan regularisointitermin summaa.

Harjanneregressio

- Harjanneregressio (engl. ridge regression) on regularisoitu versio “pienimmän neliösumman menetelmästä”, jossa regularisoinnilla suositaan normiltaan lyhyitä painovektoreita w.
- Tarkemmin ilmaistuna harjanneregressiossa f_w valitaan minimoimalla funktion $\hat{L}(w)$ sijaan funktiota

$$\hat{L}_{\text{ridge}}(w) = \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i \cdot w)^2 + \frac{1}{n}\delta^2\|w\|_2^2,$$

  missä $\delta > 0$ on erikseen säädettävä regularisoinnin määrää kontrolloiva parametri.
- Harjanneregressio siis puoltaa ennustajia, joiden painovektorin L_2-normin neliö on pieni – sitä enemmän, mitä suurempi δ on.

Funktion $\hat{L}_{\text{ridge}}$ minimoinnista

- Laskemalla funktion $\hat{L}_{\text{ridge}}$ gradientti ja asettamalla se nollaksi samaan tapaan kuin edellä nähdään, että optimiratkaisun w pitää nyt toteuttaa yhtälöryhmä

$$(X^T X + \delta I)w = X^T Y.$$

- Matriisi $(X^T X + \delta I)$ on aina kääntyvä, joten optimaalinen w on aina yksikäsitteinen ja saadaan kaavalla

$$w = (X^T X + \delta I)^{-1} X^T Y.$$

- Pienikin $\delta > 0$ riittää siis optimointiongelman ratkaisun yksikäsitteistämiseen, sopiva suurempi δ lisää ratkaisun stabiiliutta ja saattaa auttaa ylisovittamiseen.

Lineaarinen luokittelu

- Siirrytään nyt lineaarisesta regressiosta lineaariseen luokitteluun: edelleen $\mathcal{X} = \mathbb{R}^d$, mutta $\mathcal{Y} = \{-1, +1\}$, ja tappiofunktiona 0/1-tappio.
- Lineaarisessa luokittelussa ennustajat ovat muotoa

$$f_w(x) = \text{sign}(x \cdot w) = \begin{cases} +1 & \text{jos } x \cdot w \geq 0 \\ -1 & \text{jos } x \cdot w < 0. \end{cases}$$

- Ennustajat f_w eivät siis ole lineaarisia funktioita, vaan ne saadaan lineaarisista funktioista ottamalla niiden merkki.

Lineaarinen luokittelija geometrisesti

- Ennustajaa f_w vastaava päätöspinta on avaruuden $\mathcal{X}$ origon kautta kulkeva $d-1$-ulotteinen hypertaso, jonka normaali on w:n suuntainen.
- Hypertaso jakaa $\mathcal{X}$:n positiiviseen ja negatiiviseen puoliavaruuteen. Positiivinen puoli on sillä puolella, johon w osoittaa.
- Vaikka päätöspinta on piirreavaruudessa $\mathcal{X}$ hypertaso, piirteet voivat olla taustalla olevien “oikeiden syötteiden” suhteen epälineaarisia. Tällä tavalla lineaariset luokittelijat voivat esittää “oikeiden syötteiden” avaruudessa monimutkaisia päätöspintoja.

Moniluokkainen luokittelu lineaaristen luokittelijoiden avulla

- Lineaariset luokittelijat soveltuvat suoraan vain kaksiluokkaiseen tapaukseen.
- Moniluokkainen ennustusongelma voidaan kuitenkin ratkaista lineaaristen luokittimien avulla esim. seuraavasti:
  1. Muodostetaan jokaiselle luokalle $y \in \mathcal{Y}$ kaksiluokkainen ennustusongelma, jossa tehtävänä on ennustaa, onko oikea luokka y $(+1)$ vai ei (-1). Opetusdata saadaan alkuperäisestä opetusdatasta korvaamalla oikeat luokat luokilla $+1/-1$.
  2. Opitaan kutakin luokkaa y vastaava lineaarinen ennustaja f_{w_y}
  3. Ennustusvaiheessa syötteellä $x \in \mathcal{X}$ ennustetaan sitä $y \in \mathcal{Y}$, jolle $x \cdot w_y$ on suurin.

Lineaaristen luokittelijoiden oppiminen

- Tuntuisi luontevalta yrittää soveltaa empiirisen riskin minimointi -periaatetta myös lineaaristen luokittelijoiden oppimiseen, eli valita w siten, että opetusvirhe

$$\hat{L}(w) = \frac{1}{n}\sum_{i=1}^{n} L_{0/1}(y_i, f_w(x_i)) = \frac{1}{n}\left|\{(x_i, y_i) \mid y_i \neq \text{sign}(x_i \cdot w)\}\right|$$

  minimoituu.

- Funktion $\hat{L}$ minimoiminen on kuitenkin yleisessä tapauksessa laskennallisesti vaikeaa — NP-täydellistä ja hankalasti approksimoitavaa!

- Jos kuitenkin tiedetään, että opetusdata on lineaarisesti separoituvaa eli $\hat{L}(w) = 0$ jollakin w, minimointiongelma voidaan ratkaista tehokkaasti.

Perceptron-algoritmi

- Perceptron-algoritmi on yksinkertainen iteratiivinen menetelmä, jota voidaan käyttää lineaaristen luokittelijoiden oppimiseen.
- Perceptron-algoritmi konvergoituu vain, jos opetusdata on lineaarisesti separoituvaa. Sitä (tai sen variaatioita) voi toki kuitenkin yrittää käyttää myös ei-separoituvan datan tapauksessa. . .
- Seuraavassa esitettävä Perceptron-algoritmin versio käy toistuvasti läpi opetusdatan joukkoa kunnes nollavirheinen luokittelija on löytynyt, tai ennalta asetettu kierrosmäärä T tulee täyteen.

Perceptron-algoritmi pseudokoodina

$w =$ zeros$(1, d)$
for round=1:T
    update=false
    for i=1:n
        $\hat{y}_i =$ sign$(x_i \cdot w)$
        if $\hat{y}_i \neq y_i$
            $w = w + y_i x_i$
            update=true
    if update==false
        break
return w

Perceptron-algoritmin idea

- Algoritmi ylläpitää ja päivittää kandidaattipainovektoria w
- Aina kun $f_w(x_i) \neq y_i$, algoritmi päivittää kandidaattia w lisäämällä siihen vektorin $y_i x_i$.
- Ideana on, että päivitys kääntää w:tä kohti opetusdatan oikein (tai paremmin) separoivan hypertason painovektoria.
- Päivityksiä jatketaan, kunnes $f_w(x_i) = y_i$ kaikilla $i = 1, \ldots, n$, tai kierrosmäärä T tulee täyteen.

Perceptron-algoritmin konvergenssi

- Oletetaan, että
  - Opetusdata on lineaarisesti separoituvaa marginaalilla $\gamma > 0$: On olemassa painovektori $w^* \in \mathbb{R}^d$, jolle $\|w^*\|_2 = 1$ ja $y_i x_i \cdot w^* \geq \gamma$ kaikilla $i = 1, \ldots, n$
  - Opetusdata mahtuu origokeskeiseen R-säteiseen palloon: $\|x_i\|_2 \leq R$ kaikilla $i = 1, \ldots, n$.
- Tällöin Perceptron-algoritmi tekee korkeintaan R^2/γ^2 päivitystä ennen kaiken opetusdatan oikein luokittelevan ennustajan f_w painovektorin w löytämistä.

Todistus

Kaksi havaintoa (taululla):

1. Jokainen päivitys kasvattaa pistetuloa $w \cdot w^*$ vähintään γ:lla
2. Jokainen päivitys kasvattaa w:n pituuden neliötä $\|w\|_2^2$ korkeintaan R^2 verran.

Jos siis p on päivitysten kokonaismäärä, pätee

$$1 \geq \frac{w \cdot w^*}{\|w\|_2 \|w^*\|_2} \geq \frac{p\gamma}{\sqrt{pR^2}},$$

missä ensimmäinen epäyhtälö on tunnetun epäyhtälön $\frac{x \cdot y}{\|x\|_2 \|y\|_2} \leq 1$ sovellus. Näin ollen

$$p \leq \frac{R^2}{\gamma^2}.$$