Statističko učenje 21./22.
Prvi praktični zadatak (seminar)

Rok za predaju: **28. siječnja, 2022.**

Cilj praktičnog zadatka je izrada (1) `R` *koda koji rješava donje zadatke, te (2) kratke prezentacije ili dokumenta u kojem su prezentirani glavni rezultati (slike, tablice itd.) te zaključci i komentari. Maksimalan broj bodova koji se može dobiti je 25. Diskusija među studentima je naravno dopuštena (i preporučena), ali očekuje se da svaki student potpuno samostalno napiše svoje rješenje. Pripazite da se vaši rezultati mogu reproducirati (koristite funkciju* `set.seed`*). Ukoliko vam se učini zanimljivo, možete (dapače, poželjno je) napraviti i više nego što se od vas traži u zadacima. U slučaju bilo kakvih nejasnoća, pitajte!*

Zadatak 1

Prisjetimo se, LDA procedura u slučaju $K \geq 2$ klasa može se implementirati na sljedeći način:

1. Procijeni $\hat{\pi}_k, \hat{\mu}_k, \hat{\Sigma}$, za sve $k \in S = \{0, \ldots, K-1\}$;
2. Za svaki $x \in \mathbb{R}^p$, transformiraj x u $\tilde{x} := Vx \in \mathbb{R}^{K-1}$, te stavi

$$\hat{f}(x) = \arg\min_{k \in S} \underbrace{\left\{ \|\tilde{x} - \tilde{\mu}_k\|_2^2 - \log(\hat{\pi}_k) \right\}}_{=:\hat{\delta}_k(\tilde{x})},$$

   pri čemu je $\tilde{\mu}_k := V\hat{\mu}_k$, a V matrica dimenzija $(K-1) \times p$ čiji retci $v_1, \ldots, v_{K-1} \in \mathbb{R}^p$ predstavljaju koeficijente tzv. *diskriminacijskih koordinata*.

Riječima, $\hat{f}(x) = k$ ako je nakon transformacije centroid $\tilde{\mu}_k$ najbliži točki $\tilde{x}$ (uz korekciju za vjerojatnosti $\hat{\pi}_k$). Transformacija $x \mapsto \tilde{x}$ je rezultat dvaju operacija: prvo rotacije i skaliranja tako da je za nove podatke procjena zajedničke kovarijacijske matrice $\hat{\Sigma}$ upravo identiteta (engl. *sphering the data*), a zatim projekcije na potprostor H_{K-1} razapet s centroidima $\hat{\mu}_1, \ldots, \hat{\mu}_{K-1}$ (nakon prve transformacije); vidi ESL, Poglavlje 4.3.3. Dakle, LDA prirodno smanjuje dimenziju problema – originalnih p kovarijata dovoljno je gledati u prostoru dimenzije $K-1$.

Ideja linearne diskriminacijske analize *smanjenog ranga* je dodatno smanjiti dimenziju tako što ćemo podatke gledati samo u prvih L diskriminacijskih koordinata. Preciznije, za fiksni $L \in \{1, \ldots, K-1\}$ matricu V u gornjem algoritmu zamijenimo s matricom V_L dimenzija $L \times p$ čijih su L redaka koeficijenti prvih L diskriminacijskih koordinata (za $L = K-1$ to je standardna LDA metoda).

U ovom zadatku, primijenit ćete LDA metodu smanjenog ranga na problem prepoznavanja govora sa $K = 11$ klasa i $p = 10$ kovarijata. Klase odgovaraju različitim zvukovima samoglasnika u engleskom jeziku, a svaki zvuk je sadržan u jednoj od 11 različitih riječi (vidi tablicu u ESL, strana 443). Podaci trening `vowel_train.csv` su dobiveni tako da je 8 osoba izgovorilo svaku riječ 6 puta (dakle $n = 8 \cdot 11 \cdot 6 = 528$); prvih $6 \cdot 11 = 66$ redaka odgovara prvoj osobi, sljedećih 66 redaka drugoj osobi itd. Testni skup `vowel_test.csv` dobiven je na isti način uz 7 (novih) osoba. Deset kovarijata su izvedene iz izgovorenih riječi na određeni (kompliciran) način.

(a) Prikažite podatke (iz skupa za trening) zajedno s centroidima samo u prve dvije diskriminacijske koordinate. Preciznije, nacrtajte točke $V_2 x^{(i)} = (v_1^\tau x^{(i)}, v_2^\tau x^{(i)}) \in \mathbb{R}^2$ za $i = 1, \ldots, n$ te analogno za centroide, s tim da treba označiti točke s obzirom na klasu y_i kojoj pripadaju. Fiksirajte jednu klasu, označimo je s k, te za sve $j \neq k$ nacrtajte granicu odluke između klasa k i j za LDA metodu uz $L = 2$, tj. skup svih $z \in \mathbb{R}^2$ takvih da je $\hat{\delta}_k(z) = \hat{\delta}_j(z)$.

(b) Prikažite podatke i u nekoliko drugih parova diskriminacijskih koordinata (npr. prvu i treću, drugu i treću, devetu i desetu, i slično). Što zamjećujete?

(c) Koristeći unakrsnu validaciju procijenite testnu grešku (uz 0-1 funkciju gubitka) LDA metode za svaki $L \in \{1, \ldots, K-1\}$ te pronađite optimalnu dimenziju $\hat{L}$. Procijenite testnu grešku odabranog modela na testnom skupu. (*Napomena:* Razmislite kako konstruirati blokove za unakrsnu validaciju.)

(d) Za sve ostale metode obrađene u poglavlju o klasifikaciji (logistička regresija, QDA, Naivni Bayes te metoda k-najbližih susjeda) iz skupa za trening generirajte odgovarajuću $\hat{f}$, te procijenite njenu testnu grešku. Koja metoda (uključujući onu iz (a) dijela) daje najmanju grešku? (*Napomena:* Kod Naivnog Bayesa pretpostavite normalnu razdiobu za sve kovarijate, a kod metode najbližih susjeda koristite parametar $k = 1$ ili ga odaberite unakrsnom validacijom kao u (c) dijelu.)

Zadatak 2

U `bstar.Rdata` nalazi se vektor `bstar` duljine $p = 2500$ koji predstavlja jednostavnu sliku dimenzija 50×50 – možete ju nacrtati koristeći funkciju `plot.image`. Vaš cilj je na temelju n slučajnih linearnih kombinacije piksela, pri čemu je n dosta manji od p, rekonstruirati originalnu sliku; ovdje je ključno što je slika `bstar` *rijetka* (engl. *sparse*), pogledajte na webu nešto o *compressed sensing*.

Preciznije, zadan je vektor `y` koji sadrži $n = 1500$ skalarnih produkata vektora `bstar` sa vektorom $x^{(i)}$ (duljine p) sačinjenim od njd $\mathrm{N}(0, 1)$ slučajnih varijabli, s tim da je svakoj linearnoj kombinaciji dodana slučajna greška ϵ_i koja ima $\mathrm{N}(0, 5^2)$ razdiobu – vidi `zad2.R`. Cilj zadatka je dobiti što bolju procjenu za vektor koeficijenata `bstar` koristeći ridge regresiju i lasso, s tim da nećete uključivati slobodni član (engl. *intercept*).

(a) Provedite unakrsnu validaciju za obje metode te nacrtajte CV testne greške. Za obje metode, odredite λ koji minimizira CV testnu grešku te onaj dobiven tzv. pravilom jedne standardne greške (engl *one SE rule*). Za koju je metodu minimalna CV testna greška manja?

(b) Nacrtajte procjene vektora `bstar` dobivene koristeći ridge regresiju i lasso za parametre iz (a) dijela. Komentirajte rezultate.

(c) Koristeći funkciju `truncate` modificirajte procjene iz (b) dijela tako da sadrže samo vrijednosti iz $[0, 1]$ te izračunajte njihove udaljenosti od vektora `bstar` u Euklidskoj normi. Koja metoda daje najmanju grešku?

(d) Ponovite korake (a)-(c) s tim da ćete vektor `y` generirati kao gore, ali koristeći samo $n = 750$ linearnih kombinacija. Koje su razlike u odnosu na $n = 1500$?

(e) Provedite sljedeću simulacijsku studiju. Za svaki $n = 750, 1500$, $M = 1000$ puta ponovite gornje korake, tj. M puta generirajte vektor `y` koristeći n linearnih kombinacija (s novim simulacijama vektora $x^{(i)}$, $i = 1, \ldots, n$, i greške ϵ_i) te izračunajte Euklidske udaljenosti između vektora `bstar` i dobivene 4 procjene kao u (c) dijelu. Za oba n-a, nacrtajte *boxplotove* dobivenih M vrijednosti za sva 4 slučaja. Komentirajte rezultate.