Primjena tehnika smanjenja dimenzije podataka u klasifikaciji peludi

Palinkaš, Julija

Undergraduate thesis / Završni rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Department of Mathematics / Sveučilište Josipa Jurja Strossmayera u Osijeku, Odjel za matematiku**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:126:189018

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**



Repository / Repozitorij:

Repository of School of Applied Mathematics and Computer Science







Sveučilište J. J. Strossmayera u Osijeku
Odjel za matematiku
Sveučilišni preddiplomski studij matematike i računarstva

Julija Palinkaš

Primjena tehnika smanjenja dimenzije podataka u klasifikaciji peludi

Završni rad

Osijek, 2021.

Sveučilište J. J. Strossmayera u Osijeku
Odjel za matematiku
Sveučilišni preddiplomski studij matematike i računarstva

Julija Palinkaš

Primjena tehnika smanjenja dimenzije podataka u klasifikaciji peludi

Završni rad

Mentor: doc. dr. sc. Slobodan Jelić

Osijek, 2021.

Application of dimension reduction techniques in airborne pollen classification

Sažetak

Često se podaci, s kojima radimo, nalaze u visokodimenzionalnim prostorima. Zbog toga, je rad s njima otežan i dugotrajan. U ovom projektu obrađene su četiri tehnike za smanjenje dimenzije podataka. To su:

- Analiza glavnih komponenti
- Analiza nezavisnih komponenti
- Metoda slučajne projekcije
- Klasično višedimenzionalno skaliranje

Tehnike su primjenjene na podatke u klasifikaciji peludi, a nakon toga, prikazani su rezultati koji su dobiveni primjenom tehnika na podatke.

Ključne riječi

Višedimenzionalno skaliranje, matrica udaljenosti, smanjivanje dimenzije, slučajna projekcija.

Abstract

Often, the data we work with, is located in high-dimensional spaces. Because of that, working with them is difficult and time consuming. In this project, four techniques for reducing the dimension of data are discussed. These are:

- Principal components analysis
- Independent components analysis
- Random projection method
- Classic multidimensional scaling

The techniques were applied to the data in the pollen classification, and after that, the results obtained by applying the techniques to the data are presented.

Key words

Multidimensional scaling, distance matrix, dimension reduction, random projection.

Sadržaj

Uvod **1**

1 Tehnike za smanjenje dimenzije **2**
1.1 Analiza glavnih komponenti 2
1.2 Analiza nezavisnih komponenti 3
1.3 Metoda slučajne projekcije 3
1.4 Višedimenzionalno skaliranje 4
1.4.1 Metrično višedimenzionalno skaliranje 4

2 Primjena tehnika na podatke u klasifikaciji peludi **7**
2.1 Primjena PCA matode na podatke 7
2.2 Primjena ICA matode na podatke 7
2.3 Primjena RP matode na podatke 8
2.4 Primjena cMDS matode na podatke 9

Literatura **10**

Uvod

Laserskim uređajem dobiveni su podaci za klasifikaciju peludi. Podaci sadrže vrlo velik broj varijabli (ili, značajki) te ih je vrlo teško vizualizirati i raditi s njima. Da bi se lakše i brže radilo s ovakvim podacima potrebno je smanjiti visokodimenzionalni prostor, u kojem se nalaze, u niždimenzionalni prostor. To ćemo nazvati smanjenje ili redukcija dimenzije. Postoji mnogo tehnika s različitim pristupima ovom problemu.
Neke od poznatijih tehnika su:

- analiza glavnih komponenti (eng. principal component analysis),
- analiza nezavisnih komponenti (eng. independent component analysis),
- metoda slučajne projekcije (eng. random projection),
- klasično višedimenzionalno skaliranje (eng classical multidimensional scaling)

Kada smanjujemo dimenziju, vrlo je bitno da se može što više smanjiti dimenzija, ali na način, da najbitnije informacije o podacima ostanu sačuvane. Kada ćemo htjeti klasificirati naše podatke, odnosno odrediti kojoj vrsti peludi pripada pojedina pelud, onda će nam sačuvane informacije biti od velike pomoći.
Na kraju će se vizualno prikazati rezultati koji se dobiju primjenom pojedine tehnike na podatke.

1 Tehnike za smanjenje dimenzije

1.1 Analiza glavnih komponenti

Analiza glavnih komponenti (ili. kraće, PCA), vrlo je poznata i često korištena tehnika u smanjivanju dimenzije, ali i u vizualizaciji podataka. Ova tehnika je također poznata i kao Karhunen-Loéveova transformacija.

Prema C.M. Bishopu (vidi [1], str.561) postoje sljedeće dvije definicije koje rezultiraju istim algoritmom: "PCA se može definirati kao ortogonalna projekcija podataka na niže dimenzionalni linearni prostor, poznatiji kao principijalni podprostor, tako da je varijanca projiciranih podataka maksimizirana. Isto tako, PCA se može definirati kao linearna projekcija koja minimizira prosječni trošak projekcije, definirana kao srednja kavdratna udaljenost između točaka podataka i njihovih projekcija."

Za smanjivanje dimenzije koristi se dekompozicija singularnom vrijednošću (eng. singular value decomposition, kraće SVD). Sljedeći teorem stoji iza točnosti PCA tehnike.

Teorem 1.1 (Eckart-Young-Mirsky teorem)
Neka je $X \epsilon R^{m \times n}$ *centrirana matrica čiji je rang* $r(X)$ *te je pripadna SVD dekompozicija (vidi [9]) matrice*

$$X = U \Sigma V^T,$$

pri čemu je Σ *dijagonalna matrica*

$$\Sigma = diag(\sigma_1, ..., \sigma_r, 0, ..., 0) = \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_r & \\ & & & \ddots \end{bmatrix}$$

sa singularnim vrijednostima na dijagonali $\sigma_1 \geq \cdots \geq \sigma_r \geq 0$, *a matrice* U *i* V *ortogonalne matrice. Tada je*

$$\tilde{X} = U S_k V^T$$

rješenje optimizacijskog problema

$$\min_{r(Y)=k} = \|X - Y\|_F,$$

gdje je matrica S_k *dijagonalna matrica sa prvih* k *najvećih singularnih vrijednosti matrice* Σ *na dijagonali, odnosno*

$$\Sigma = diag(\sigma_1, ..., \sigma_k, 0, ..., 0) = \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_k & \\ & & & \ddots \end{bmatrix}.$$

Dokaz teorema može se pronaći u [5].

Mana PCA tehnike je u tome što je vrijeme izvođenja ove tehnike kvadratno po broju izvorne dimenzije pa ukoliko proslijedimo podatke nešto veće dimenzije to postaje predugo za izvršavanje.

1.2 Analiza nezavisnih komponenti

Analiza nezavisnih komponenti (ili, kraće, ICA) je tehnika za linearno smanjivanje dimenzije. Kroz prošlost, razvili su se mnogi ICA algoritmi, ali najveća primjena započela je kada se razvio FastICA algoritam. Razvili su ga Hyvarinen i Oja (vidi [6]).

ICA transformira skup varijabli u novi skup komponenti na način da je statistička nezavisnost između novih komponenti maksimizirana. Prema J. Wang i C. Changu (vidi [12]) ključna ideja ICA metode je da su: "podaci linearno pomiješani skupom zasebnih neovisnih izvora i demiksiraju te izvore signala prema njihovoj statističkoj neovisnosti izmjerenoj međusobnim informacijama."

Ako su podaci prikazani sa slučajnim vektorom $X = (x_1, \ldots, x_m)^T$ i sa skrivenim komponentama slučajnog vektora $S = (s_1, \ldots, s_n)^T$, zadatka je transformirati promatrane podatke X, koristeći linearni statički transformator B tako da je $S = BX$, u vektor sa maksimalno nezavisnim komponentama S uz mjeru neovisnosti koju daje funkcija $F(s_1, \ldots, s_n)$.

ICA se osim u redukciji dimenzije koristi i za prepoznavanje lica, izjednačavanje kanala za prepoznavanje govora i slično.

1.3 Metoda slučajne projekcije

Metoda slučajne projekcije (ili, kraće RP) jednostavnija je metoda za smanjivanje dimenzije podataka. Ideja metode je koristiti matricu, s random generiranim vrijednostima, čiji stupci imaju jediničnu duljinu. Cilj je spustiti podatke X iz višedimenzionalnog prostora u nižedimenzionalni prostor koristeći matricu R s random popunjenim vektorima.

$$S_{[m*n]} = R_{[m*d]} \times X_{[d*n]}$$

Gdje je n broj točaka, d originalna dimenzija, a m dimenzija na koju se smanjuje.
Iza točnosti rezultata ove metode stoji Johnson–Lindenstraussova lema i lema slučajne projekcije koje su navedene ispod, bez dokaza.

Lema 1.1 (Johnson–Lindenstrauss lema)
Za dani $0 < \varepsilon < 1, X \epsilon R^{m \times n}$ *i realan broj* $k > C\varepsilon^{-2} logn$, *gdje je* $C > 0$ *dovoljno velika konstanta, postoji linearno mapiranje* $f : R^n \to R^k$ *takvo da vrijedi*

$$(1-\varepsilon)\|x_i - x_j\|_2 \leq \|f(x_i) - f(x_j)\|_2 \leq (1+\varepsilon)\|x_i - x_j\|_2, \forall i, j = 1 \ldots m.$$

Lema 1.2 (Lema slučajne projekcije) *Neka je* $\varepsilon \epsilon (0,1)$ *i slučajno normalizirano linearno mapiranje* $T : R^n \to R^k$. *Tada za svaki* $x \epsilon R^n$ *vrijedi*

$$P((1-\varepsilon)\|x\|_2 \leq \|Tx\|_2 \leq (1+\varepsilon)\|x\|_2) \geq 1 - 2e^{-c\varepsilon^2 k},$$

gdje je $c > 0$ *neka konstanta neovisna o* n, k *i* ε.

Po Johnson–Lindenstraussovoj lemi, čiji dokaz možemo pronaći na [3], ako napravimo ortogonalnu projekciju n točaka, sa vektorskog prostora (R^d) na nižedimenzionalni prostor rezultirat će sačuvanom udaljenošću između točaka. To je dobra prednost ove metode kao i brzina kada se radi s visokodimenzionalnim podacima.

1.4 Višedimenzionalno skaliranje

Višedimenzionalno skaliranje (ili, kraće, MDS) koristi se pri vizualnom prikazu udaljenosti između podataka i kod prikaza različitosti skupova objekata. Ponekad je zadatak takav da nam nisu dane točke već udaljenosti između točaka. Algoritam koji radi s takvim skupom podataka je MDS.

Početna točka MDS-a su jednosmjerni i dvosmjerni podaci te posebno, mjerenje različitosti. Ako pretpostavima da skup od n točaka ima različitosti δ_{rs} izmjerenih između svih parova objekata, konfiguracija od n točaka će predstavljati objekt u p dimenzionalnom prostoru. Svaka točka predstavlja objekt. Dakle sa rth-om točkom predstavljen je r-ti objekt. S d_{rs} označimo udaljenosti, koji ne moraju biti nužno Euklidske udaljenosti, između točaka. Cilj MDS-a je naći takve konfiguracije da se udaljenosti d_{rs} "podudaraju", što je više moguće, s različitostima δ_{rs}.

Postoji više podvrsta višedimenzionalnog sklairanja ovisno o podacima s kojima rade i ovisno o tome kako je dana notacija "podudaranja". Neke od podvrsta su:

- Klasično Višedimenzionalno Skaliranje
- Metrično Višedimenzionalno Skaliranje
- Nemetrično Višedimenzionalno Skaliranje
- Generalizirano Višedimenzionalno Skaliranje

1.4.1 Metrično višedimenzionalno skaliranje

Matematički gledano, metrično višedimenzionalno skaliranje može se opisati na sljedeći način:

Neka objekt obuhvaća skup O. Neka je različitost definirana na $O \times O$ prostoru, između objekata r i s koji su iz O. Neka je ϕ proizvoljno mapiranje s O u E, gdje je E najčešće Euklidski prostor, u kojem skup točaka predstavlja objekt. Neka $\phi(r) = x_r$ pri čemu $r \,\epsilon\, O, x_r \,\epsilon\, E$ i neka je X skup koji sadrži $x_r : r \,\epsilon\, O$ skup slika. Neka je udaljenost između točaka x_r, x_s u X dana s d_{rs}. Cilj je naći mapiranje ϕ, za koji je d_{rs} približno jednak $f(\delta_{rs})$ za svaki $r, s \,\epsilon\, O$.

Jedno od glavnih metoda za metrično višedimenzionalno skaliranje je klasično višedimenzionalno skaliranje (ili, kraće, cMDS).

Klasično višedimenzionalno skaliranje

Prema J. Imperial (vidi [7]): "Klasična, ne-prerađena verzija MDS-a nastala je oko 1930-e godine kada su Young i Householder pokazali kako se, počevši od matrice udaljenosti između točaka u euklidskom prostoru, mogu pronaći koordinate točaka tako da se udaljenosti sačuvaju. Postoje, međutim, brojne neeuklidske mjere udaljenosti koje su ograničene na vrlo specifična istraživačka pitanja. Na kraju je Torgerson doveo temu do popularnosti koristeći tehniku skaliranja. To je dovelo do višedimenzionalnog skaliranja kao tehnike smanjenja dimenzija i vizualizacije za visokodimenzionalne podatke."

Metoda pronalaska originalnih euklidskih koordinata je sljedeća:
Neka su koordinate od n točaka u p dimenzionalnom Euklidskom prostoru dane s $x_r\,(r = 1,\ldots,n)$, gdje je $x_r = (x_{r1},\ldots,x_{rp})^T$. Euklidska udaljenost između rth i sth točaka dana je s

$$d_{rs}^2 = (x_r - x_s)^T(x_r - x_s).$$

Neka je matrica B dana s $[B]_{rs} = b_{rs} = x_r^T x_s$.
Pomoću kvadratnih udaljenosti d_{rs}, dobije se matrica B, a nakon toga pomoću matrice B dobiju se nepoznate koordinate.

Za pronalazak matrice B
Prvo, se središte konfiguracijskih točaka postavlja u ishodište. Stoga

$$\sum_{r=1}^{n} x_{ri} = 0 (i = 1,\ldots,p).$$

Za pronalaz B-a, iz d_{rs}^2

$$d_{rs}^2 = x_r^T x_r + x_s^T x_s - 2x_r^T x_s,$$

i

$$\frac{1}{n}\sum_{r=1}^{n} d_{rs}^2 = \frac{1}{n}\sum_{r=1}^{n} x_r^T x_r + x_s^T x_s,$$

$$\frac{1}{n}\sum_{s=1}^{n} d_{rs}^2 = x_r^T x_r + \frac{1}{n}\sum_{s=1}^{n} x_s^T x_s,$$

$$\frac{1}{n^2}\sum_{r=1}^{n}\sum_{s=1}^{n} d_{rs}^2 = \frac{2}{n}\sum_{r=1}^{n} x_r^T x_r.$$

Zamijenimo li, dobijemo

$$b_{rs} = x_r^t x_s = -\frac{1}{2}\left(d_{rs}^2 - \frac{1}{n}\sum_{r=1}^{n} d_{rs}^2 - -\frac{1}{n}\sum_{s=1}^{n} d_{rs}^2 + \frac{1}{n^2}\sum_{r=1}^{n}\sum_{s=1}^{n} d_{rs}^2\right) = a_{rs} - a_{r.} - a_{.s} + a_{..},$$

gdje je $a_{rs} = -\frac{1}{2}d_{rs}^2$, i

$$a_{r.} = n^{-1}\sum_{s} a_{rs}, \qquad a_{.s} = n^{-1}\sum_{r} a_{rs} \qquad a_{..} = n^{-2}\sum_{r}\sum_{s} a_{rs}.$$

Definiramo matricu A kao $[A]_{rs} = a_{rs}$, i tada je matrica B

$$B = CAC$$

pri čemu je C matrica centriranja dana s $C = I - n^{-1}JJ^T$, gdje je $J = (1,1,\ldots,1)^T$, vektor sa n jedinica.

Za oporavak koordinata iz matrice B
Matrica B se može izraziti kao $B = XX^T$, gdje je $X = [x_1,\ldots,x_n]^T$, $n \times p$ matrica s koordinatama. Rang matrice B, $r(B)$, je

$$r(B) = r(XX^T) = r(X) = p.$$

B je simetrična, semi-definitna matrica s rangom p i ima p nenegativnih svojstvenih vrijednosti i $n-p$ nul svojstvenih vrijednosti.
B se može sada zapisati u okviru spektralne dekompozicije,

$$B = V\Lambda V^T,$$

gdje je $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$ diagonalna matrica sa svojstvenim vrijednostima λ_i i $V = [v_1, \ldots, v_n]$ matrica s odgovarajućim svostvenim vektorima, koji su normalizirani, odnsono $v_i^T v_i = 1$. Da bi bilo lakše raditi s njima, svojstvene vrijednosti matrice B su označeni tako da je $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$.
Zbog $n-p$ nul svojstvenih vrijednosti B se sad može zapisati kao

$$B = V_1\Lambda_1 V_1^T,$$

gdje je $\Lambda_1 = diag(\lambda_1, \ldots, \lambda_p)$, a $V_1 = [v_1, \ldots, v_p]$.
Ako je matrica $B = XX^T$, matrica koordinata X je tada dana s $X = V_1\Lambda_1^{\frac{1}{2}}$, gdje je $\Lambda_1^{\frac{1}{2}} = diag(\lambda_1^{\frac{1}{2}}, \ldots, \lambda_p^{\frac{1}{2}})$, te su tako koordinate točaka oporavljene od udaljenosti između točaka. Proizvoljna oznaka svojstvenih vektora v_i dovodi do invarijantnosti rješenja s obzirom na refleksiju u ishodištu.

Algoritam, za klasično višedimenzionalno skaliranje, sastoji se od sljedećih sažetih koraka:

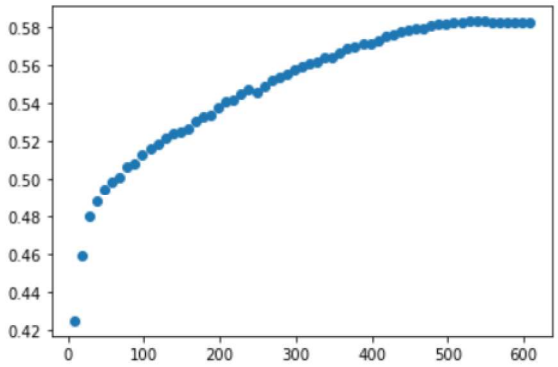
- Podaci su dani matricom M. Treba izračunati različitosti δ_{rs}.
- Pronaći matricu $A = [-\frac{1}{2}\delta_{rs}^2]$
- Pronaći matricu $B = [a_{rs} - a_{r.} - a_{.s} + a_{..}]$.
- Nakon toga, treba odrediti najveće svojstvene vrijednosti $(\lambda_1, \lambda_2, \ldots, \lambda_{n-1})$ i odgovarajuće svojstvene vektore $(v_1, v_2, \ldots, v_{n-1})$ pri čemu su svojstveni vektori normalizirani, odnosno $v_i^T v_i = \lambda_i$. Ako B nije pozitivno semi-definitna, onda, ili se ignoriraju negativne vrijednosti, ili se uvodi konstanta c te je $\delta'_{rs} = \delta_{rs} + c(1 - \delta^r s)$ i vrati se na drugi korak.
- Treba odabrati odgovarajući broj dimenzije p.
- Koordinate od n točaka u p dimenzionalnom Euklidskom prostoru dana su s $x_{ri} = v_{ri}$ za $r = 1, \ldots, n$ i za $i = 1, \ldots, p$.

2 Primjena tehnika na podatke u klasifikaciji peludi

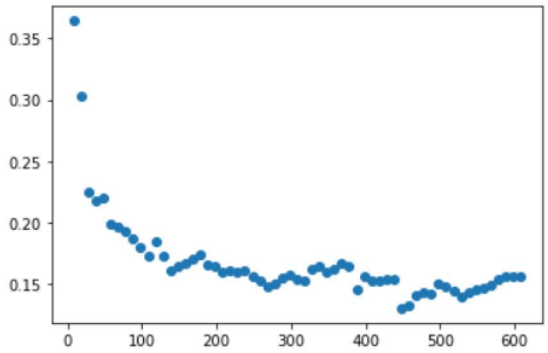
Podaci s kojima smo radili nalaze se u velikoj tablici koja je sadržavala sveukupno 16 vrsta peludi. Svaka pelud je imala 608 svojstava pa smo htjeli vidjeti kako će, primjenom pojedine tehnike na te podatke, točnost "ponašati", odnosno pri klasificiranju, koliko će se peludi dobro klasificirati ukoliko uzmemo samo dio svojstava.

2.1 Primjena PCA matode na podatke

Varijabli d zadali smo trenutnu dimenziju podataka. Nakon toga smo ponavljali sljedeći postupak sve dok je $d > 0$: primjenili PCA metodu na podatke i dobili podatke s novom dimenzijom d, učili model, za kojeg smo odabrali logističku regresiju, na skupu za učenje, a nakon toga izračunali smo postotak dobro klasificiranih peludi i na skupu za učenje i na skupu za testiranje. Na kraju smo smanjili d za 10.
Na slici 2.1 prikazano je kako točnost pada na skupu za učenje, a na slici 2.2 prikazano je kako na skupu za testiranje raste.



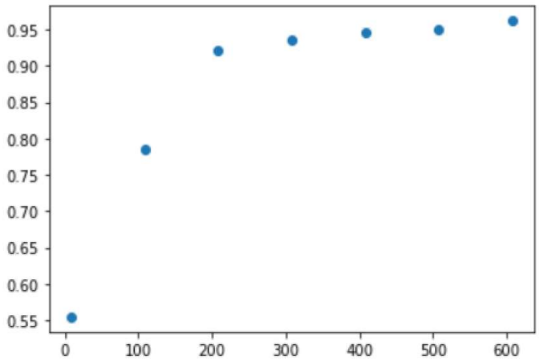
Slika 2.1 Točnost smanjivanjem dimenzije opada na skupu za učenje



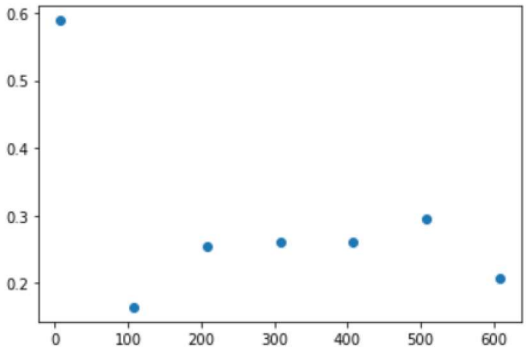
Slika 2.2 Točnost smanjivanjem dimenzije raste na skupu za testiranje

2.2 Primjena ICA matode na podatke

Varijabli d zadali smo trenutnu dimenziju podataka. Nakon toga smo ponavljali sljedeći postupak sve dok je $d > 0$: primjenili ICA metodu na podatke i dobili podatke s novom dimenzijom d, učili model, za kojeg smo odabrali logističku regresiju, na skupu za učenje, a nakon toga izračunali smo postotak dobro klasificiranih peludi i na skupu za učenje i na skupu za testiranje. Na kraju smo smanjili d za 100.
Na slici 2.3 prikazano je kako točnost pada na skupu za učenje, a na slici 2.4 prikazano je kako na skupu za testiranje prvo pada, a nakon toga raste.



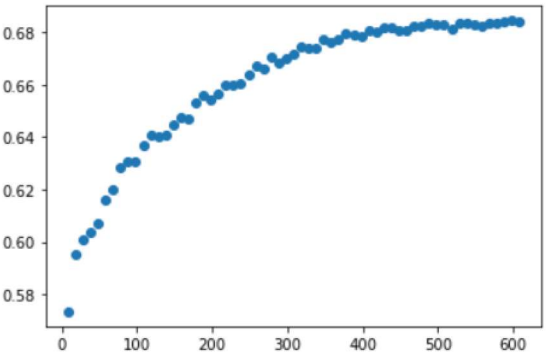
Slika 2.3 Točnost smanjivanjem dimenzije opada na skupu za učenje



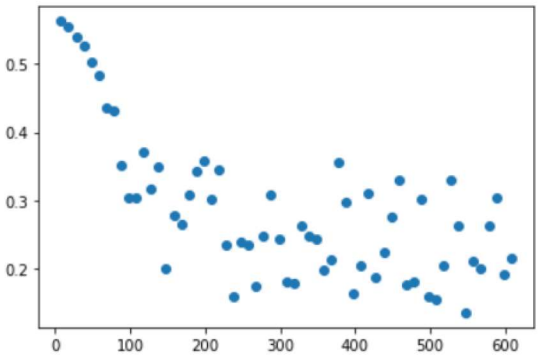
Slika 2.4 Točnost smanjivanjem dimenzije raste na skupu za testiranje

2.3 Primjena RP matode na podatke

Varijabli d zadali smo trenutnu dimenziju podataka. Nakon toga smo ponavljali sljedeći postupak sve dok je $d > 0$: primjenili RP metodu na podatke i dobili podatke s novom dimenzijom d, učili model, za kojeg smo odabrali logističku regresiju, na skupu za učenje, a nakon toga izračunali smo postotak dobro klasificiranih peludi i na skupu za učenje i na skupu za testiranje. Na kraju smo smanjili d za 10.
Na slici 2.5 prikazano je kako točnost pada na skupu za učenje, a na slici 2.6 prikazano je kako na skupu za testiranje do 88 dimenzije ne možemo utvrditi, a ispod 80 raste točnost.



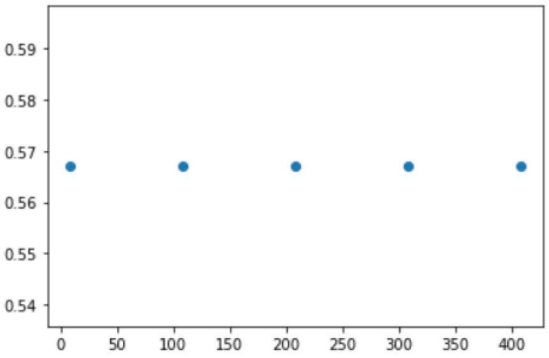
Slika 2.5 Točnost smanjivanjem dimenzije opada na skupu za učenje



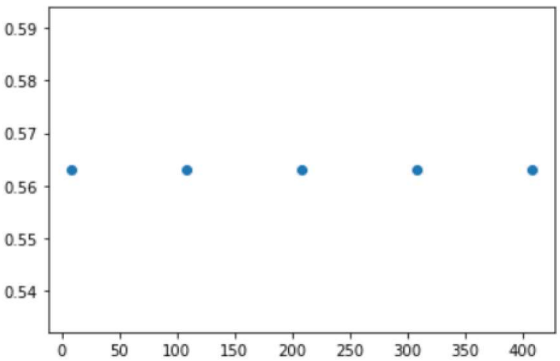
Slika 2.6 Točnost smanjivanjem dimenzije raste na skupu za testiranje kada je dimenzija manja od osamdeset

2.4 Primjena cMDS matode na podatke

Varijabli d zadali smo trenutnu dimenziju podataka smanjenu za 200. Nakon toga smo ponavljali sljedeći postupak sve dok je $d > 0$: primjenili cMDS metodu na podatke i dobili podatke s novom dimenzijom d, učili model, za kojeg smo odabrali logističku regresiju, na skupu za učenje, a nakon toga izračunali smo postotak dobro klasificiranih peludi i na skupu za učenje i na skupu za testiranje. Na kraju smo smanjili d za 100.
Na slici 2.7 prikazano je kako je točnost jednaka za svaku dimenziju na kupu za učenje, a na slici 2.8 prikazano je kako i na skupu za testiranje, točnost ima istu vrijednost za svaku dimenziju.



Slika 2.7 Točnost smanjivanjem dimenzije ostaje jednaka na skupu za učenje



Slika 2.8 Točnost smanjivanjem dimenzije ostaje jednaka na skupu za testiranje

Literatura

[1] C. M. Bishop, Pattern recognition and machine learning, Springer, New York, 2006

[2] T. C. Cox, M. A. A. Cox, Multidimensional scaling, Chapman & Hall/CRC, 2001

[3] S. Dasgupta and A. Gupta. An elementary proof of the johnson-lindenstrauss lemma. Technical report, 1999.

[4] S. Dasgupta, Experiments with random projection, AT&T Labs – Research.

[5] Eckart–Young–Mirsky theorem, Wikipedia. Dostupno na: https://en.wikipedia.org/wiki/Low-rank_approximation (4.10.2021.)

[6] Hyvarinen, J.Karhunen i E. Oja Independent Component Analysis, JOHN WILEY & SONS, INC. New York: Wiley, 2001.

[7] J. Imperial, The Multidimensional Scaling (MDS) algorithm for dimensionality reduction, 2019. Dostupno na: https://medium.datadriveninvestor.com/the-multidimensional-scaling-mds-algorithm-for-dimensionality-reduction-9211f7fa5345 (21.9.2021)

[8] Independent component analysis, Wikipedia. Dostupno na: https://en.wikipedia.org/wiki/Independent_component_analysis (21.9.2021.)

[9] Singular Value Decomposition, Wikipedia. Dostupno na: https://en.wikipedia.org/ wiki/Singular_value_decomposition (21.9.2021.)

[10] V. Sulić, J. Perš, M. Kristan i S. Kovačič, Efficient Dimensionality Reduction Using Random Projection, Slovenija, 2010.

[11] A. Treadway, ICA on images with python. Dostupno na: http://theautomatic.net/2018/06/23/ica-on-images-with-python/ (21.9.2021.)

[12] J. Wang, C. Chang, Independent Component Analysis-Based Dimensionality Reduction With Applications in Hyperspectral Image Analysis, IEEE, 2006.