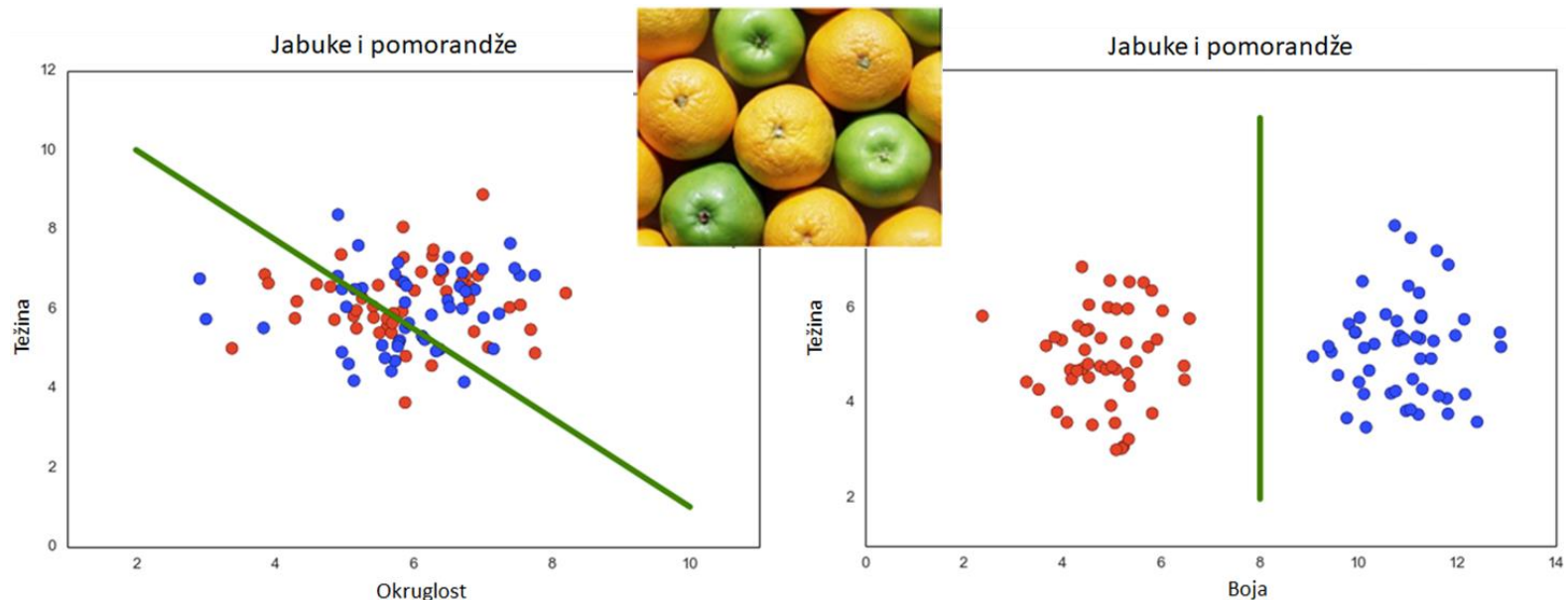


Redukcija dimenzionalnosti

PCA (*Principal Component Analysis*)

Kako da smanjimo broj dimenzija?

- Ideja: kreirati novi podskup obeležja koji dobro sumarizuje polazna obeležja
- Dobar podskup obeležja je onaj koji je *relevantan* za ciljnu funkciju f
- Na primer, onaj koji ima veliki kapacitet da napravi razliku između primera različitih klasa



Zbog čega želimo manje dimenzija?

1. Kompresija

- manje zauzeće memorije i diska
- (važnije) značajno ubrzanje obučavajućih algoritama

2. Uklanjanje šuma

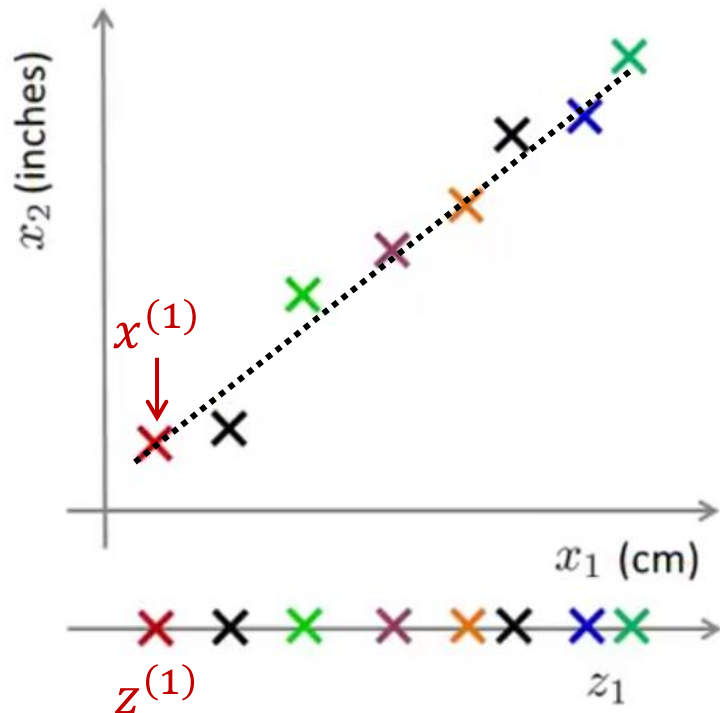
- Previše (irelevantnih) obeležja može da degradira performanse

3. Vizuelizacija

- Bolje razumevanje podataka što može da omogući izgradnju boljih modela

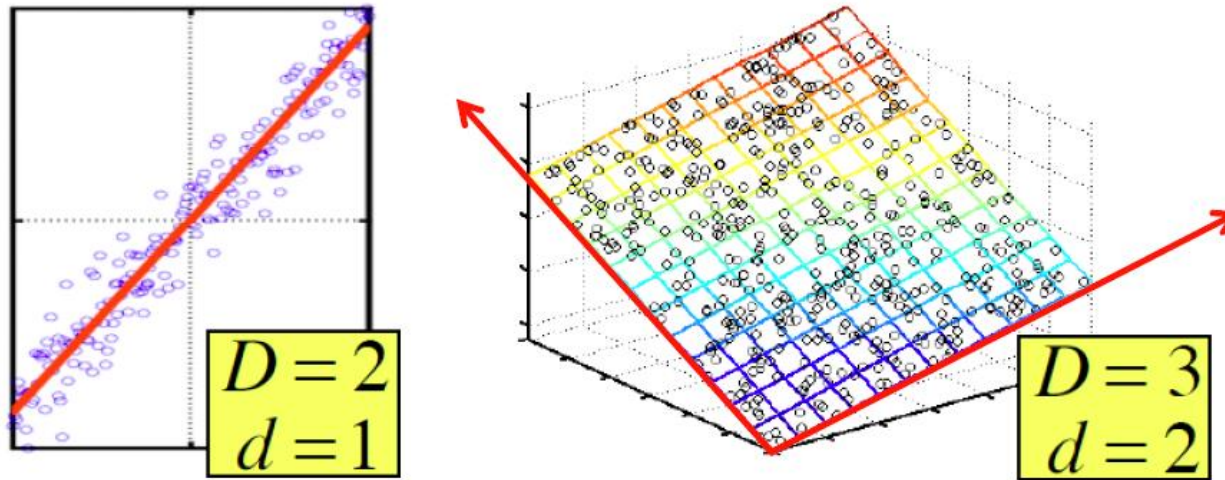
Kompresija

- Recimo da smo sakupili skup podataka sa veoma mnogo obeležja
- Ovde su grafički predstavljena samo dva obeležja:
 x_1 – dužina u cm, x_2 – ista dužinu u inčima



- Umesto da imamo dva odvojena (redundantna) obeležja, bolje bi bilo da redukujemo informaciju u jedno obeležje (jednu dimenziju)
 - $x^{(i)} \in \mathbb{R}^2 \rightarrow z^{(i)} \in \mathbb{R}$
 - Izvršili smo određenu aproksimaciju skupa podataka, ali smo prepolovili broj obeležja

Kompresija



- Pretpostavka: podaci leže tačno na ili blizu d -dimenzionog potprostora
- Ose ovog potprostora predstavljaju efektivnu reprezentaciju podataka
- U tipičnom zadatku redukcije dimenzionalnosti možemo imati više hiljada obeležja koja želimo da projektujemo u 100-dimenzioni prostor

Uklanjanje šuma

- Još jedan primer bi bilo automatsko prepoznavanje osobe koja se nalazi na slici
 - Interesuju nas sistematične varijacije koje zaista reprezentuju kako osoba izgleda
 - Ali na slikama možemo imati „šum“ poput promena u osvetljenju i drugih uslova pod kojim je snimak napravljen
- Prilikom automatskog klasifikovanja rukom pisanih cifara:
 - Pretvaranje slike u binarne
 - Skaliranje na istu dimenziju, npr. 16×16
 - Umesto 256 parametara možda možemo koristiti svega dva relevantna obeležja – prosečan intenzitet i simetrija...uklanjamo fluktuacije koje nisu relevantne za prepoznavanje o kojoj je cifri reč

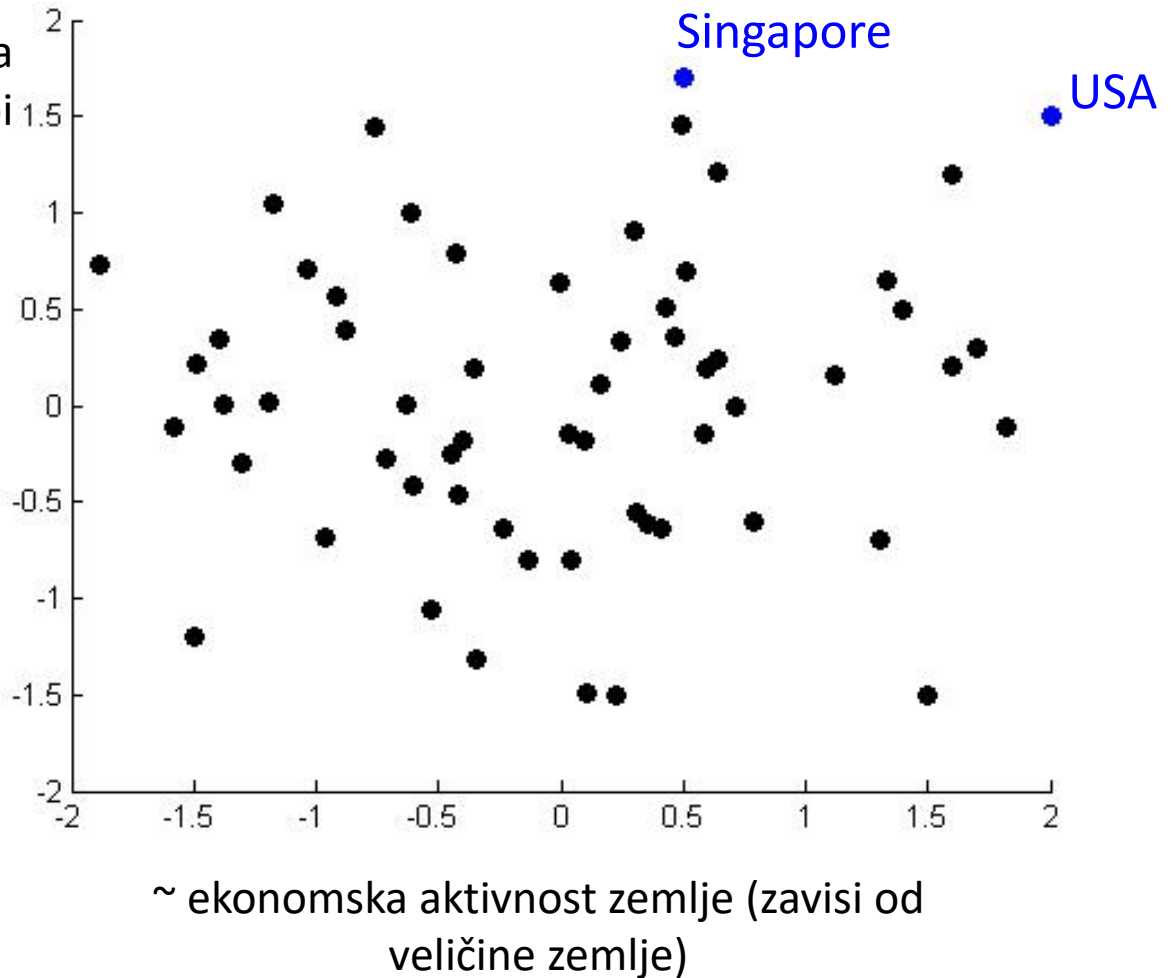
Vizuelizacija

Country	GDP (trillions of US\$)	Per capita GDP (thousands of intl. \$)	Human Develop- ment Index	Life expectancy	Poverty Index (Gini as percentage)	Mean household income (thousands of US\$)	...
Canada	1.577	39.17	0.908	80.7	32.6	67.293	...
China	5.878	7.54	0.687	73	46.9	10.22	...
India	1.632	3.41	0.547	64.7	36.8	0.735	...
Russia	1.48	19.84	0.755	65.5	39.9	0.72	...
Singapore	0.223	56.69	0.866	80	42.5	67.1	...
USA	14.527	46.86	0.91	78.3	40.8	84.3	...
...

Vizualizacija

~ GDP per
capita/ekonomska
aktivnost po osobi

Country		
Canada	1.6	1.2
China	1.7	0.3
India	1.6	0.2
Russia	1.4	0.5
Singapore	0.5	1.7
USA	2	1.5
...



Oprez!

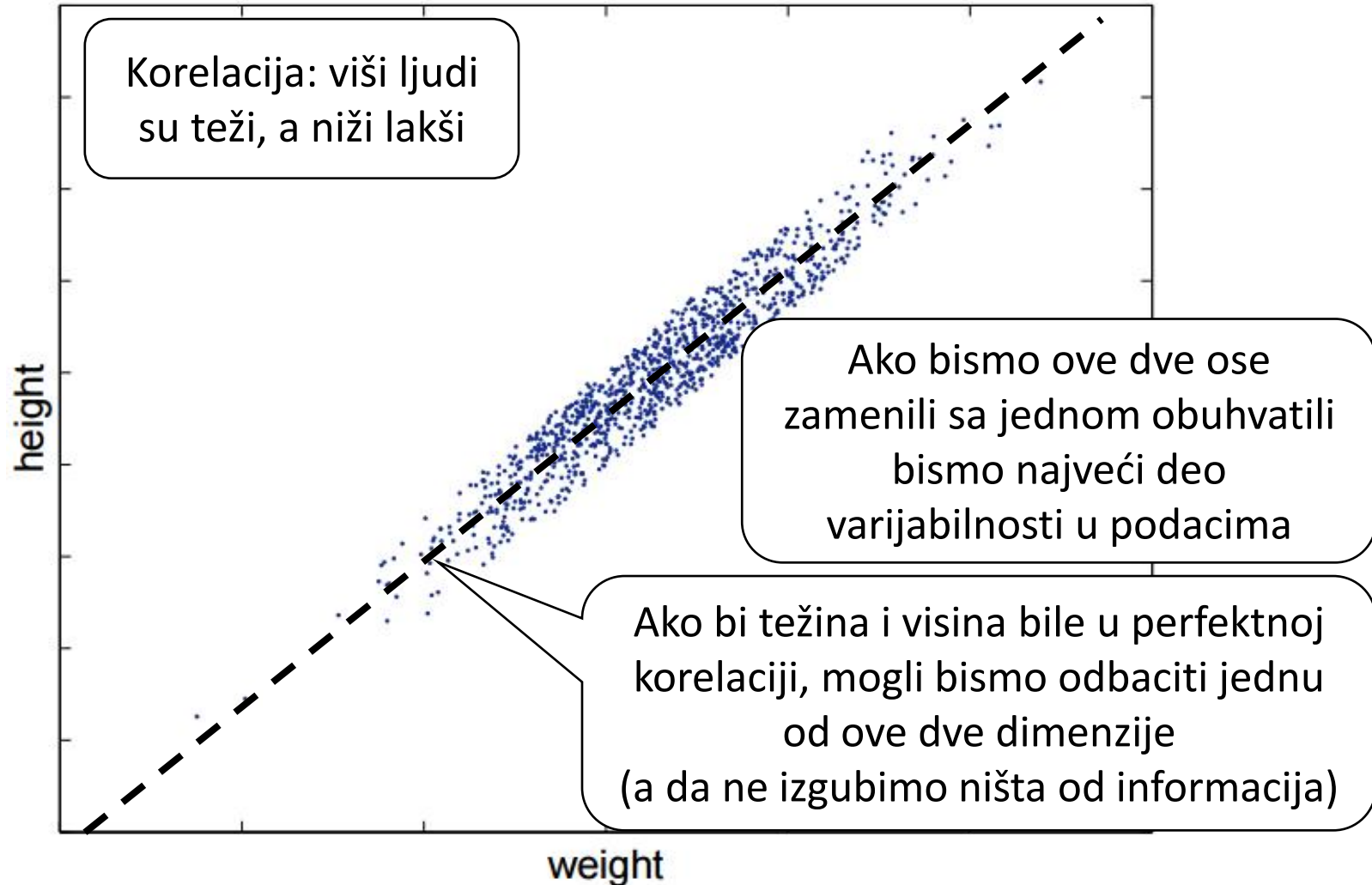
- Važno je redukciju dimenzionalnosti sprovesti na principijelan način
- Odbacujemo informacije – možemo da izgubimo one koje su ključne za obučavanje
- Važno je da algoritam sačuva koristan deo informacija, a odbaci šum

Kako da smanjimo broj dimenzija?

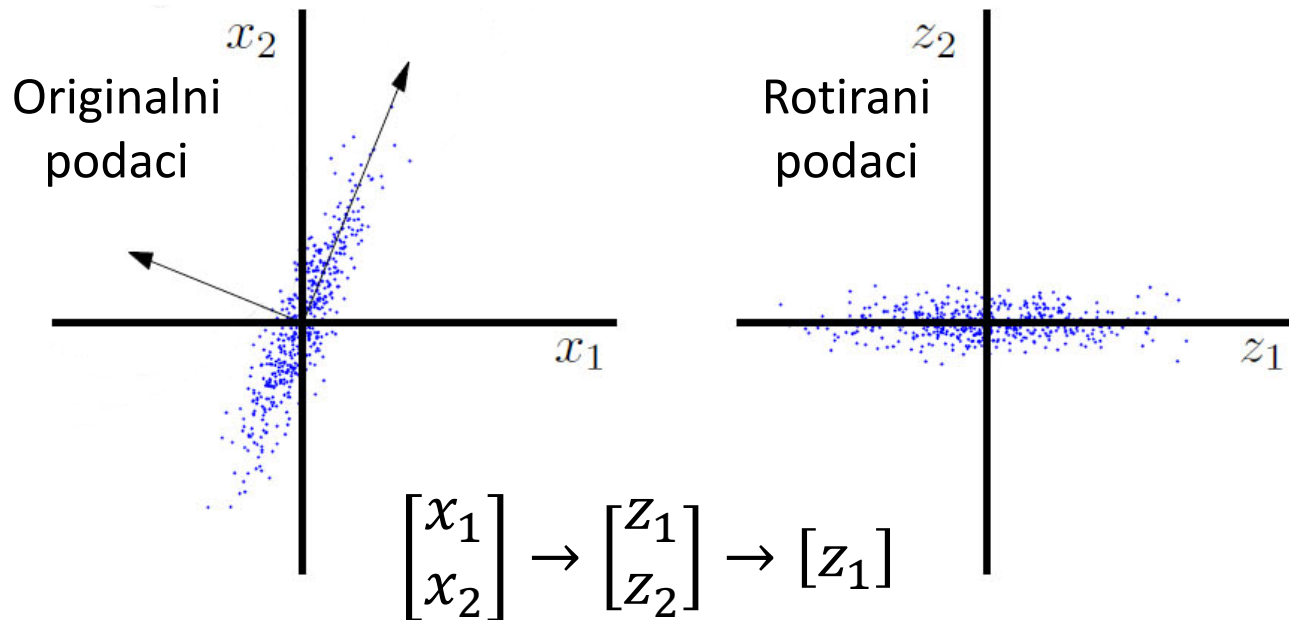
- Selekcija obeležja
 - Pronaći minimalan podskup obeležja koji nam može pomoći da razlikujemo klase
- Redukcija dimenzionalnosti
 - Kreirati nova obeležja koja će predstavljati neku kombinaciju starih obeležja

Principal Component Analysis (PCA)

- Recimo da smo sproveli anketu i zabeležili visinu i težinu grupe ljudi



Principal Component Analysis (PCA)



- PCA konstruiše mali broj *linearnih* obeležja koja sumarizuju ulazne podatke
- Ideja je da se rotiraju ose (linearna transformacija koja definiše novi koordinatni sistem), tako da u ovom sistemu
 - Identifikujemo dominantne dimenzije (informacije)
 - Odbacimo manje dimenzije (šum)

Koordinatni sistem

- Koordinatni sistem je definisan skupom ortonormalnih vektora (međusobno ortogonalni jedinični vektori)
- Dužina projekcije tačke x na *jedinični vektor* v je $x^T v$

Primer: Euklidski koordinatni sistem

- Definisan je vektorima v_1, \dots, v_D , gde vektor v_i ima i -tu koordinatu 1, a sve ostale koordinate 0
- Ulazni vektor x ima komponente $x_i = x^T v_i$ i možemo pisati

$$x = \sum_{i=1}^D x_i v_i = \sum_{i=1}^D (x^T v_i) v_i$$

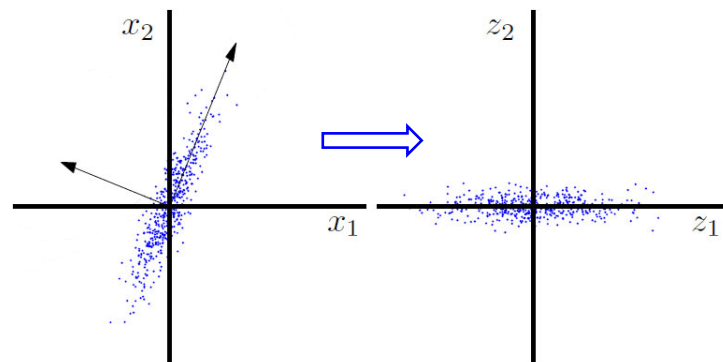
Koordinatni sistem

- Isto se može uraditi sa bilo kojom ortonormalnom bazom v_1, \dots, v_D :

$$x = \sum_{i=1}^D z_i v_i = \sum_{i=1}^D (x^T v_i) v_i$$

gde su *koordinate* u bazi v_1, \dots, v_D date sa $z_i = (x^T v_i)$

- Cilj PCA je da konstruiše intuitivniju bazu gde je većina koordinati mala



- Male koordinate tretiramo kao slučajne fluktuacije i postavljamo ih na 0
- Nadamo se da smo ovim smanjili dimenzionalnost, a sačuvali većinu važnih informacija

Zadatak

- Projektovati D -dimenzioni prostor u K -dimenzioni prostor $x^{(i)} \in \mathbb{R}^D \rightarrow z^{(i)} \in \mathbb{R}^K$ ($K \leq D$)
1. Pronaći ose novog koordinatnog sistema: v_1, v_2, \dots, v_D
 2. Transformisati x u novi prostor (koordinate transformisanog vektora su z_1, z_2, \dots, z_D)
 3. Recimo da su prvih $K \leq D$ koordinati informativne. Odbacićemo preostale koordinate da bismo dobili ulazni vektor redukovane dimenzionalnosti:

$$z = \begin{bmatrix} z_1 \\ z_2 \\ \dots \\ z_K \end{bmatrix} = \begin{bmatrix} x^T v_1 \\ x^T v_2 \\ \dots \\ x^T v_K \end{bmatrix} = \Phi(x)$$

Odbačene informacije

- Da smo sačuvali sve komponente z_1, \dots, z_D , mogli bismo rekonstruisati originalni vektor x :

$$x = \sum_{i=1}^D z_i v_i$$

- Sa prvih K komponenti, rekonstrukcija je

$$\hat{x} = \sum_{i=1}^K z_i v_i$$

- Magnituda odbačenih informacija (greška rekonstrukcije) je:

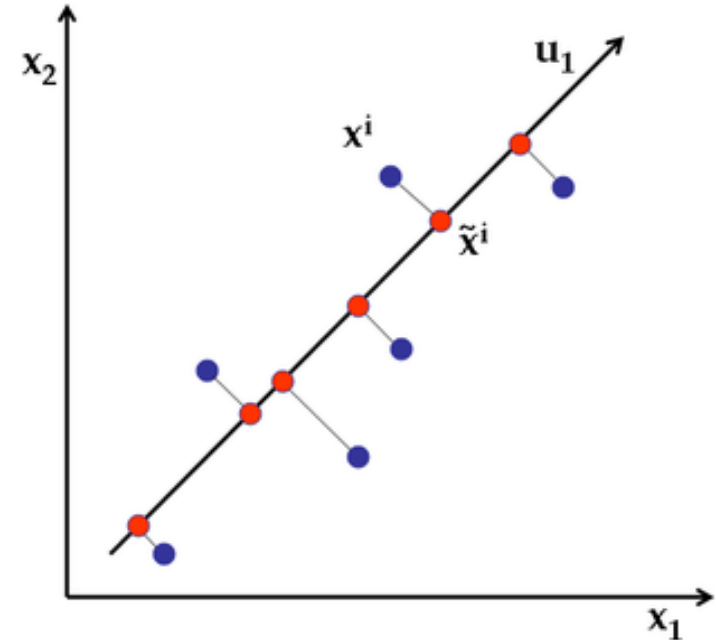
$$\|x - \hat{x}\|^2 = \left\| \sum_{i=K+1}^D z_i v_i \right\|^2 = \sum_{i=K+1}^D z_i^2$$

Odbačene informacije

- Novi koordinatni sistem je dobar ako je suma grešaka rekonstrukcije izračunata za sve primere skupa podataka mala ($\hat{x}_n \approx x_n$ za $n \in \{1, \dots, N\}$), tj., ako je malo:

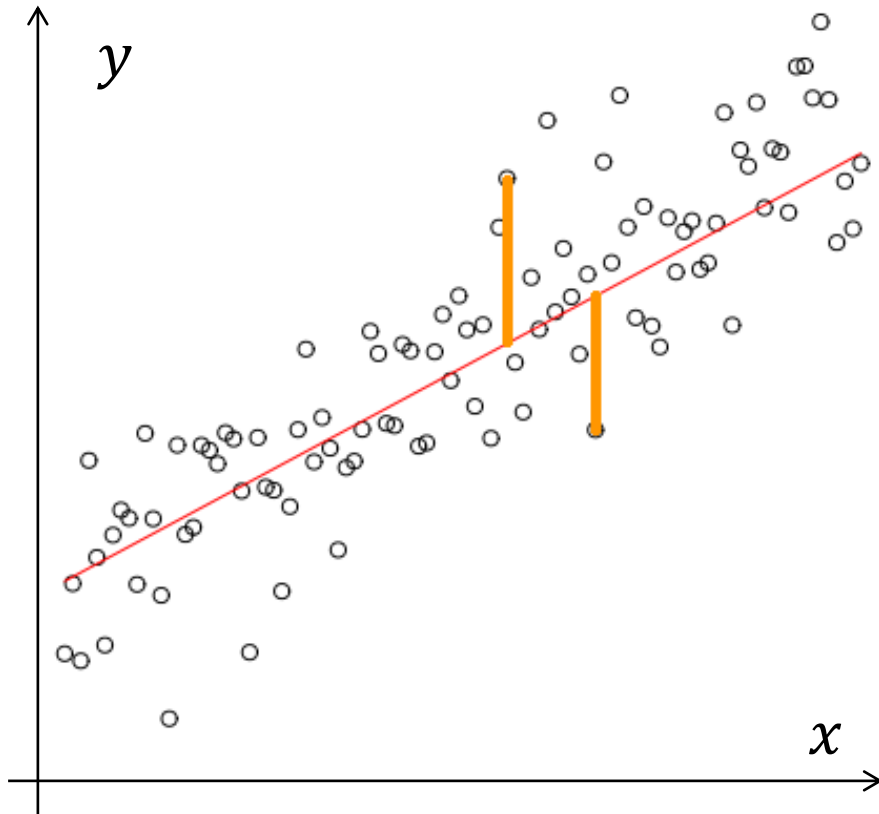
$$\sum_{n=1}^N \|x_n - \hat{x}_n\|^2$$

- PCA zato pronalazi koordinatni sistem koji **mimimizuje ukupnu grešku rekonstrukcije**
- Minimizujemo rastojanja tačaka od njihovih projekcija



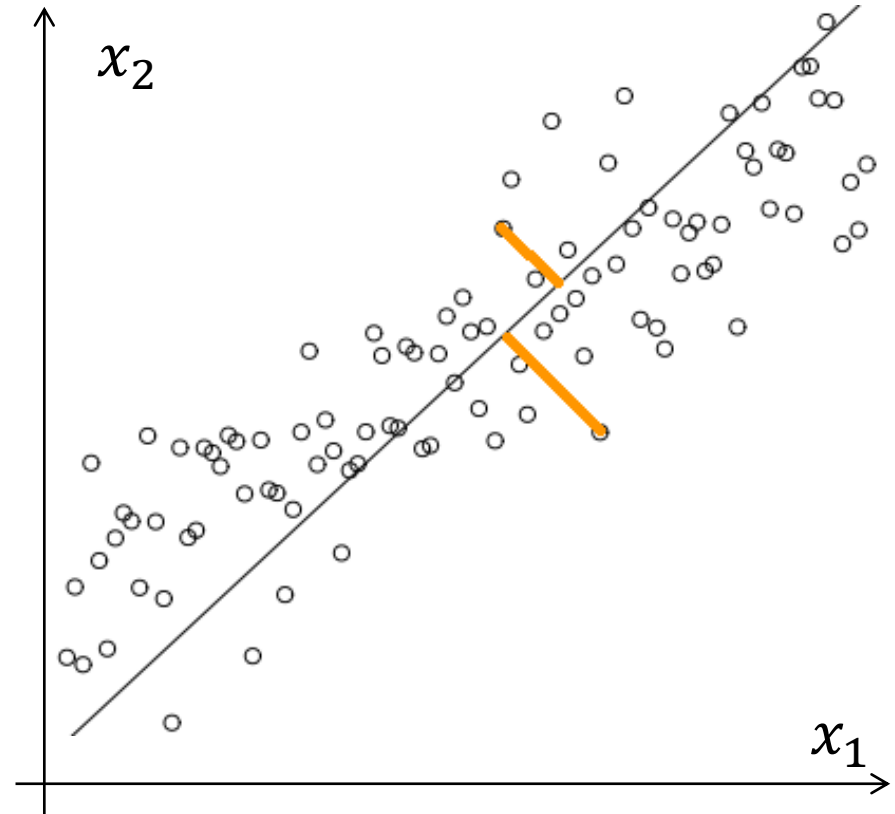
PCA nije linearna regresija

Linearna regresija



Specijalno obeležje y (ciljna varijabla) $y = f(x)$

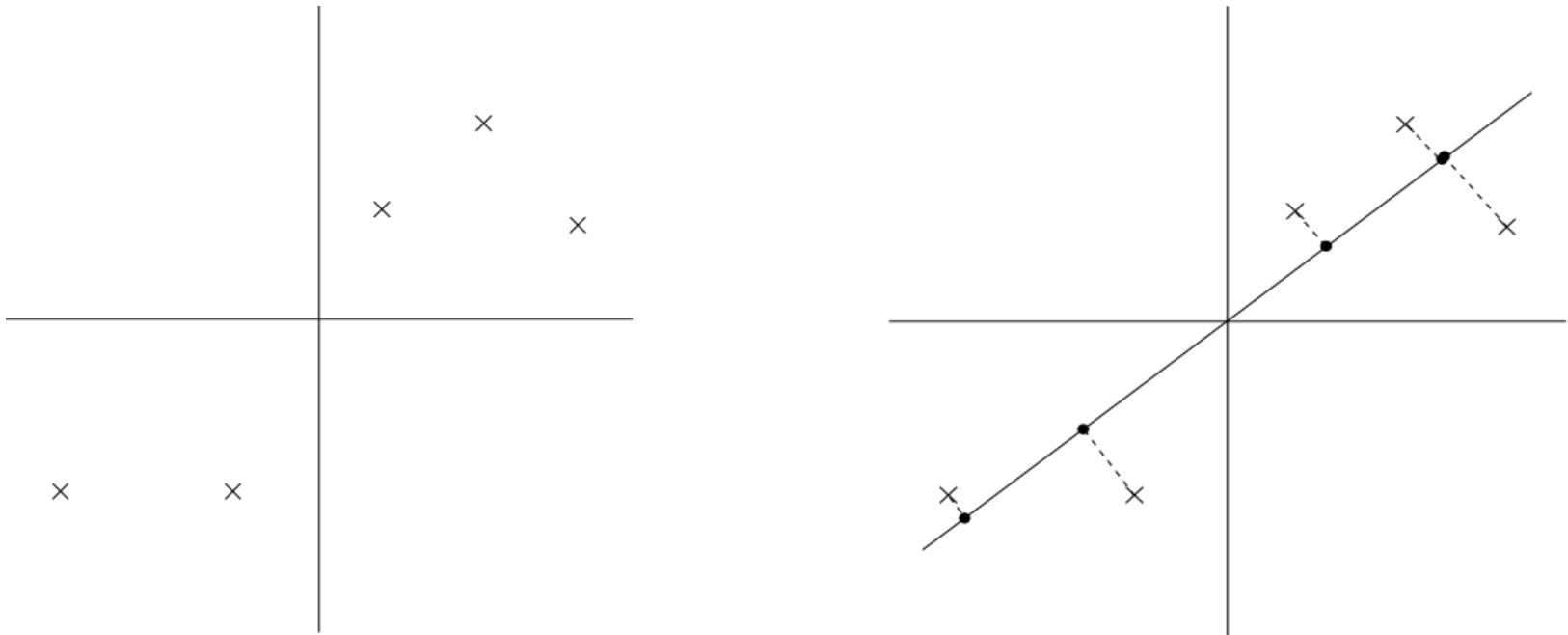
PCA



Sva obeležja su tretirana identično

PCA – druga (ekvivalentna) formulacija

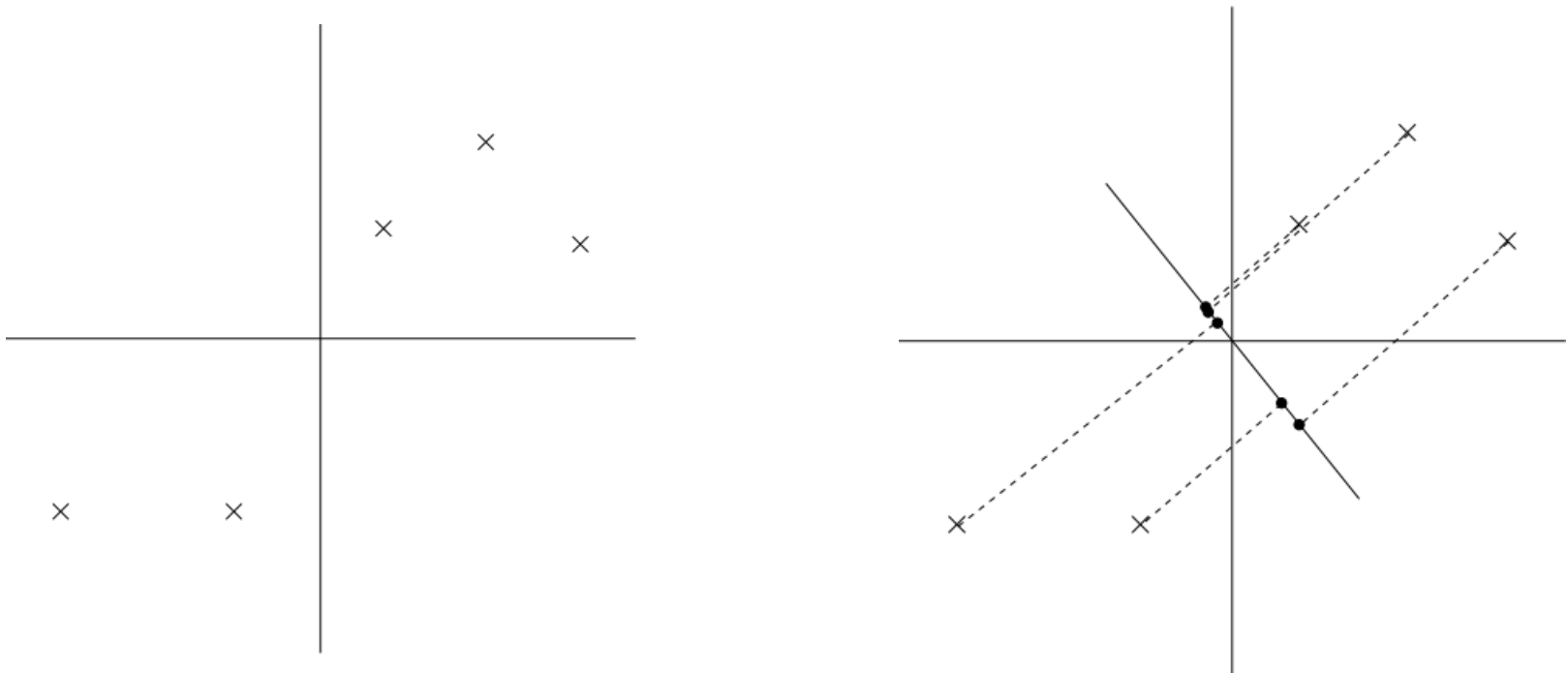
- Cilj: zadržati što više informacija u podacima
⇒ novi koordinatni sistem određujemo tako da zadržimo što je moguće više *varijabilnosti* u dobijenoj projekciji



Projektovani podaci i dalje imaju dosta veliku varijansu. Podaci imaju tendenciju da budu daleko od nule

PCA – druga (ekvivalentna) formulacija

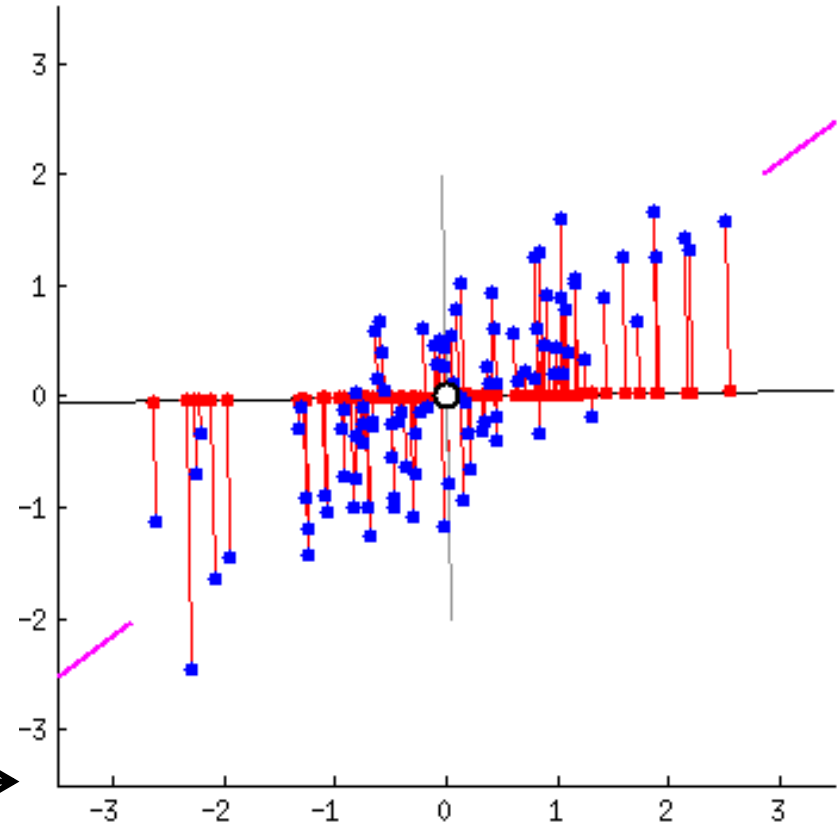
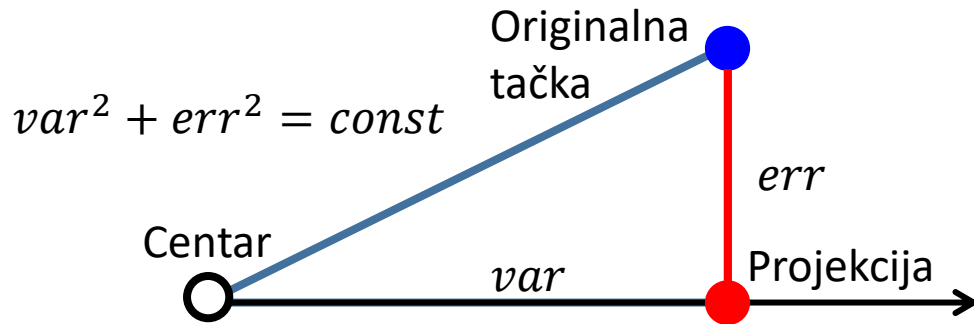
- Cilj: zadržati što više informacija u podacima
⇒ novi koordinatni sistem određujemo tako da zadržimo što je moguće više *varijabilnosti* u dobijenoj projekciji



Projektovani podaci imaju znatno manju varijansu i mnogo su bliži nuli

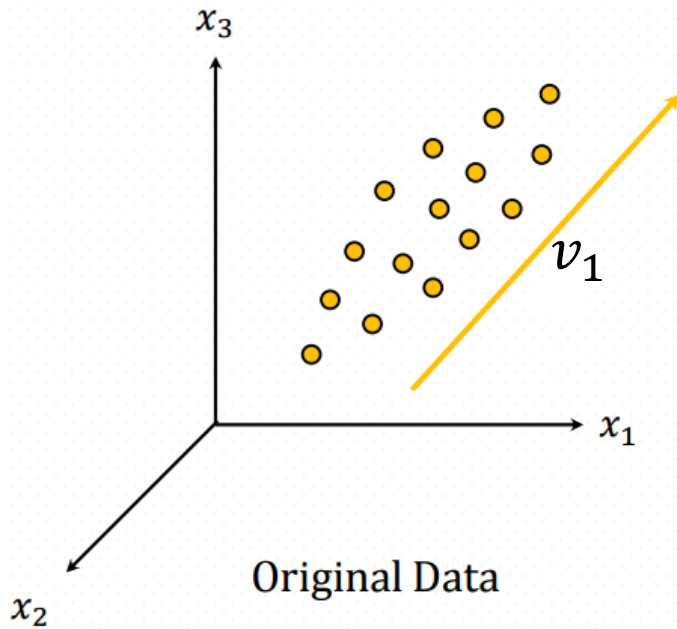
PCA – dve ekvivalentne formulacije

- Ove dve formulacije PCA su ekvivalentne
 - Na slici možete primetiti da je zadržana varijansa projekcija („širina“ crvenih tačaka na novoj osi) najveća istovremeno kada je greška rekonstrukcije (suma crvenih linija) najmanja

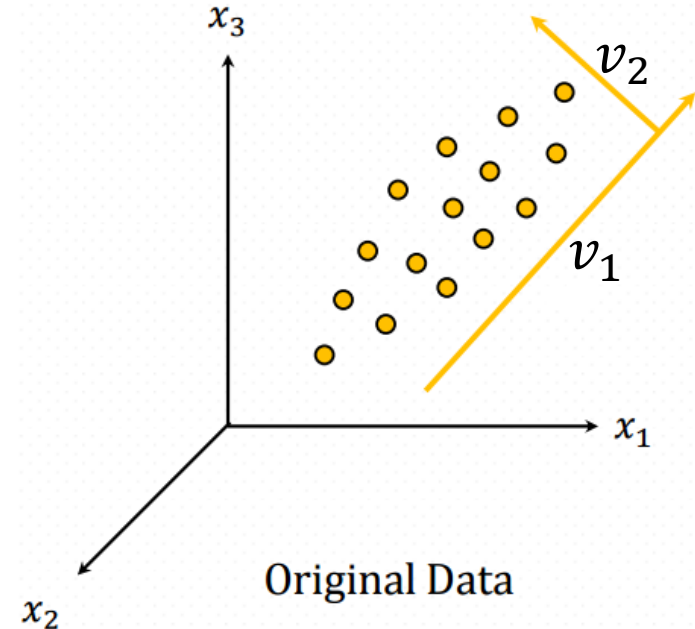


- PCA: Principal Component Analysis (analiza glavnih komponenti)
 - Nova osa je **prva glavna komponenta**

PCA: dekompozicija matrice Σ



Prvi sopstveni vektor v_1 (sa najvećim λ_1) je u pravcu najveće varijabilnosti podataka

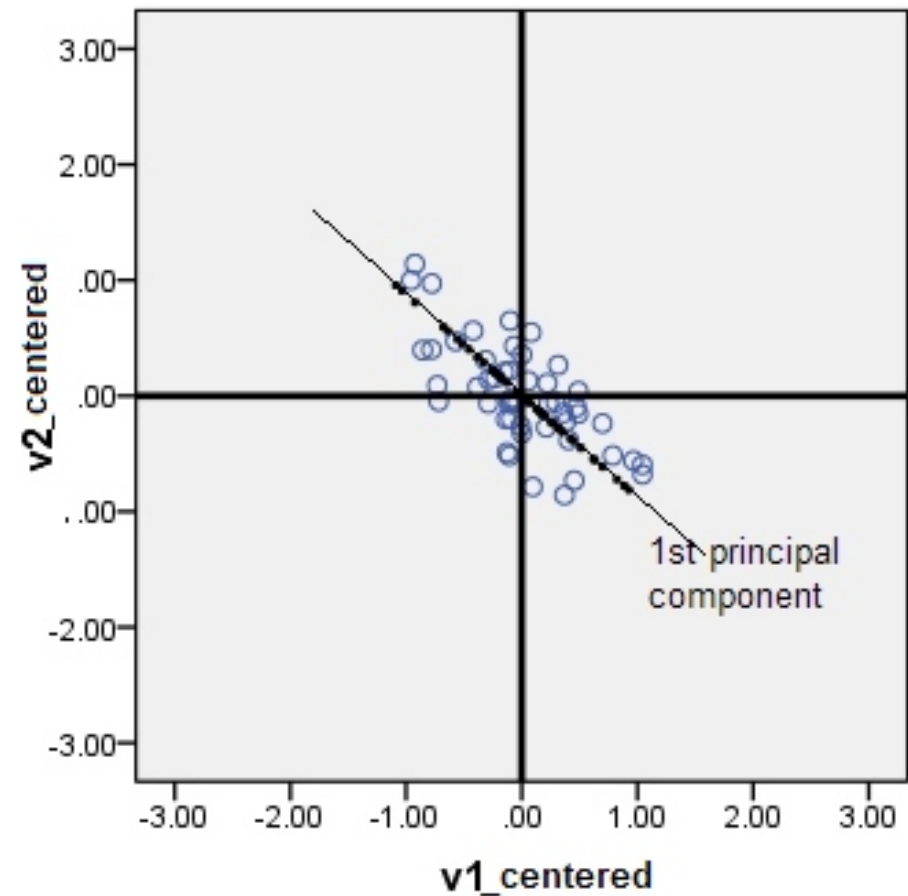
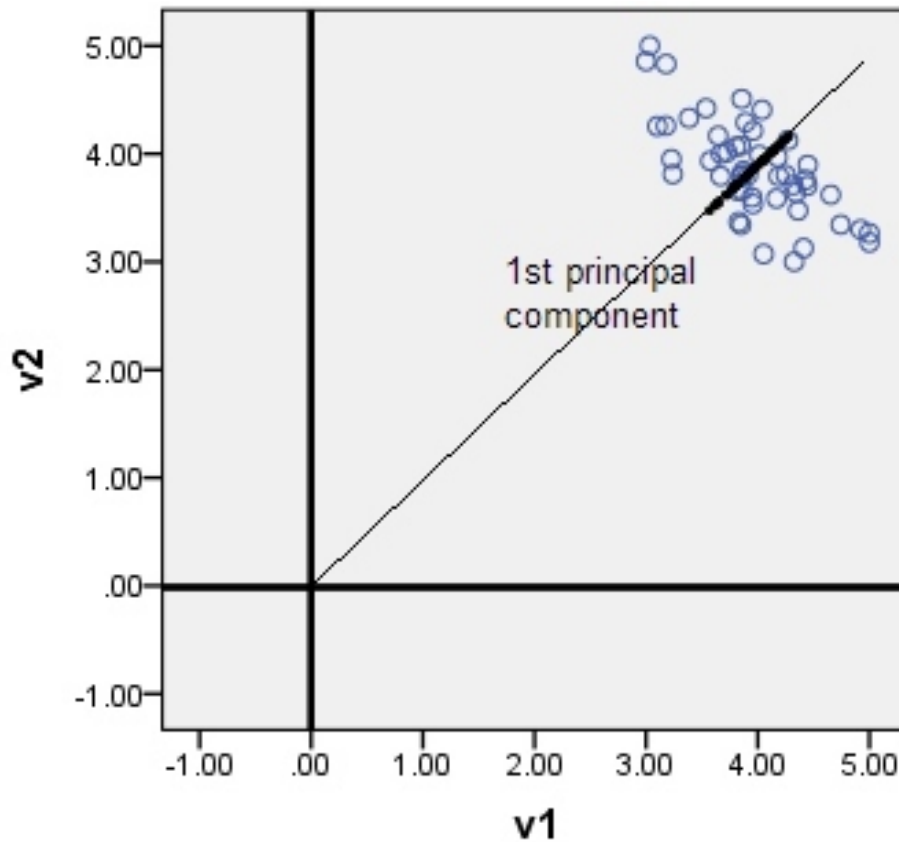


Naredna projekcija v_2 (drugo po veličini λ_2) je u pravcu sledeće najveće varijabilnosti, pri čemu je ortogonalna na sve prethodne projekcije

(da nije ortogonalna, obuhvatala bi varijabilnost već obuhvaćenu prethodnim projekcijama)

Pretprocesiranje podataka za PCA

1. Podaci moraju biti centrirani



Pretprocesiranje podataka za PCA

- Postupak centriranja podataka (*mean normalization*):
 - a. Dat je trening skup $T = \{(x^{(i)}, y^{(i)}), i \in \{1, \dots, N\}, x^{(i)} \in \mathbb{R}^D\}$
 - b. Za svako obeležje $d \in \{1, \dots, D\}$ izračunati srednju vrednost:
$$\mu_d = \frac{1}{N} \sum_{i=1}^N x_d^{(i)}$$
 - c. Za svako obeležje d : $x_d^{(i)} \leftarrow x_d^{(i)} - \mu_d$

Pretprocesiranje podataka za PCA

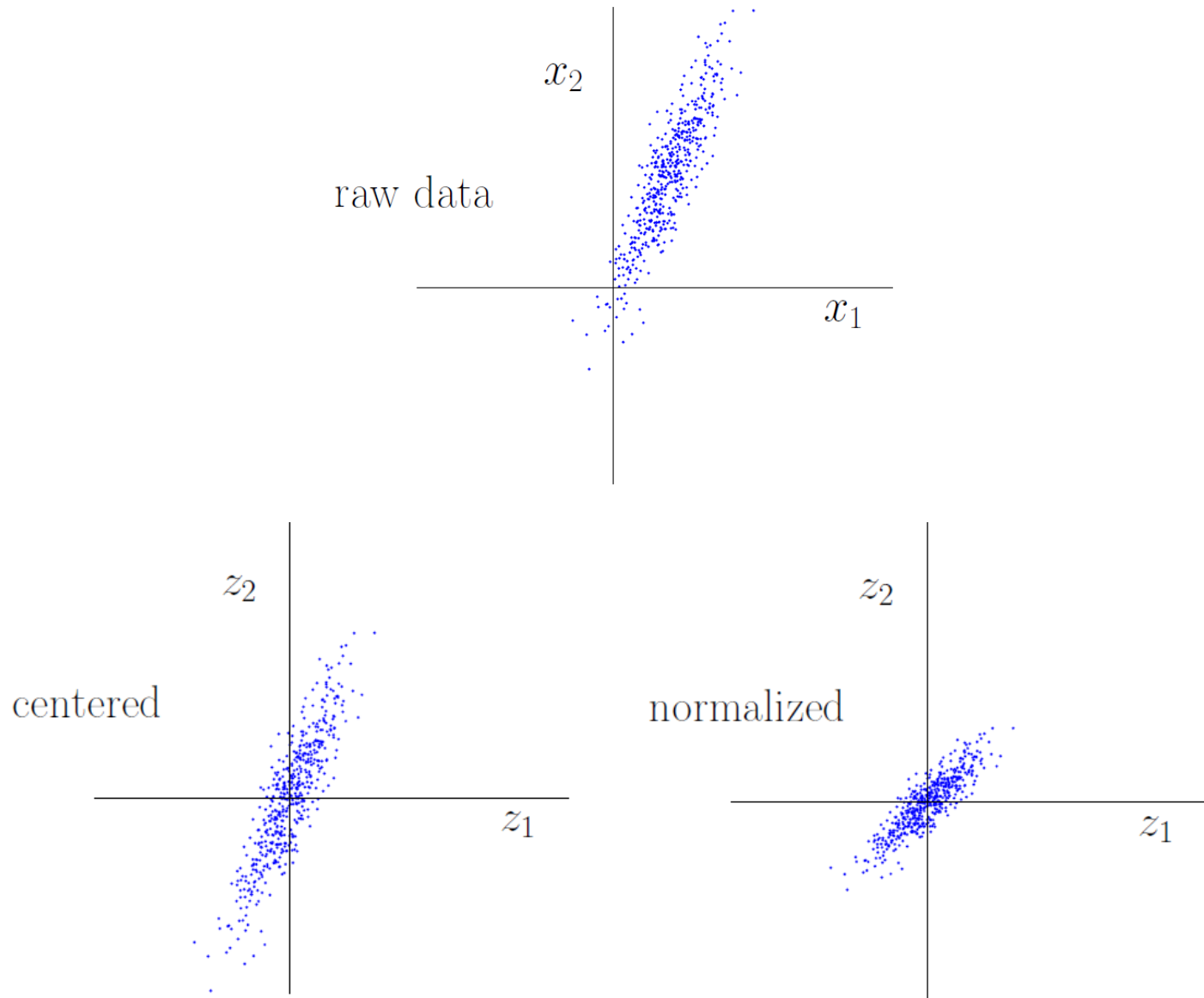
2. Normalizovati podatke (opciono)

- Ako se opsezi vrednosti različitih obeležja veoma razlikuju, obeležja treba skalirati tako da se kreću u približno istom opsegu, npr.

$$x_d^{(i)} \leftarrow \frac{x_d^{(i)}}{\sigma_d}$$

- Velike razlike u opsezima varijabli koje potiču iz (proizvoljnog) odabira jedinice u kojima ih izražavamo su problem za PCA

Pretprocesiranje podataka za PCA



Važne napomene

- Obeležja dobijena pomoću PCA metode su **linearne kombinacije** originalnih obeležja
- Svi koraci primenjeni u PCA, uključujući korake pretprocesiranja (centriranje i normalizaciju) su **nenadgledani** (ne zavise od izlaza y)

Primena PCA: ubrzavanje treniranja modela

Imamo trening skup $T =$

$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(N)}, y^{(N)})\}$$

1. Neanotirani skup: $x^{(1)}, x^{(2)}, \dots, x^{(N)}, x \in \mathbb{R}^D$
2. Primenimo PCA: $z^{(1)}, z^{(2)}, \dots, z^{(N)}, z \in \mathbb{R}^K, K < D$
3. Treniramo model na novom trening skupu: $T' = \{(z^{(1)}, y^{(1)}), (z^{(2)}, y^{(2)}), \dots, (z^{(N)}, y^{(N)})\}$

Kako odabrati broj novih obeležja K ?

- Ako nam je cilj vizuelizacija, odabraćemo $K = 2$ ili $K = 3$
- Ako nam je cilj da ubrzamo obučavajući algoritam, obično ćemo K odabrati tako da zadržimo određeni procenat varijanse (tipično 99%, 95% ili 90%)

Odabir K – zadržavanje % varijanse

- Zadržani procenat varijanse u podacima (želimo da bude što bliže 1):

$$\frac{\lambda_1 + \lambda_2 + \dots + \lambda_K}{\lambda_1 + \lambda_2 + \dots + \lambda_D}$$

- Procedura:

1. $[U, S, V] = \text{svd}(X)$

2. S je dijagonalna matrica:

$$\begin{bmatrix} \sqrt{\lambda_1} & & & \\ & \sqrt{\lambda_2} & & \\ & & \ddots & \\ & & & \sqrt{\lambda_D} \end{bmatrix}$$

3. Ako želimo da sačuvamo 99% varijanse u podacima, odabrati K tako da:

$$\frac{\lambda_1 + \lambda_2 + \dots + \lambda_K}{\lambda_1 + \lambda_2 + \dots + \lambda_D} \geq 0.99$$

Loše upotrebe PCA – prevencija overfittinga

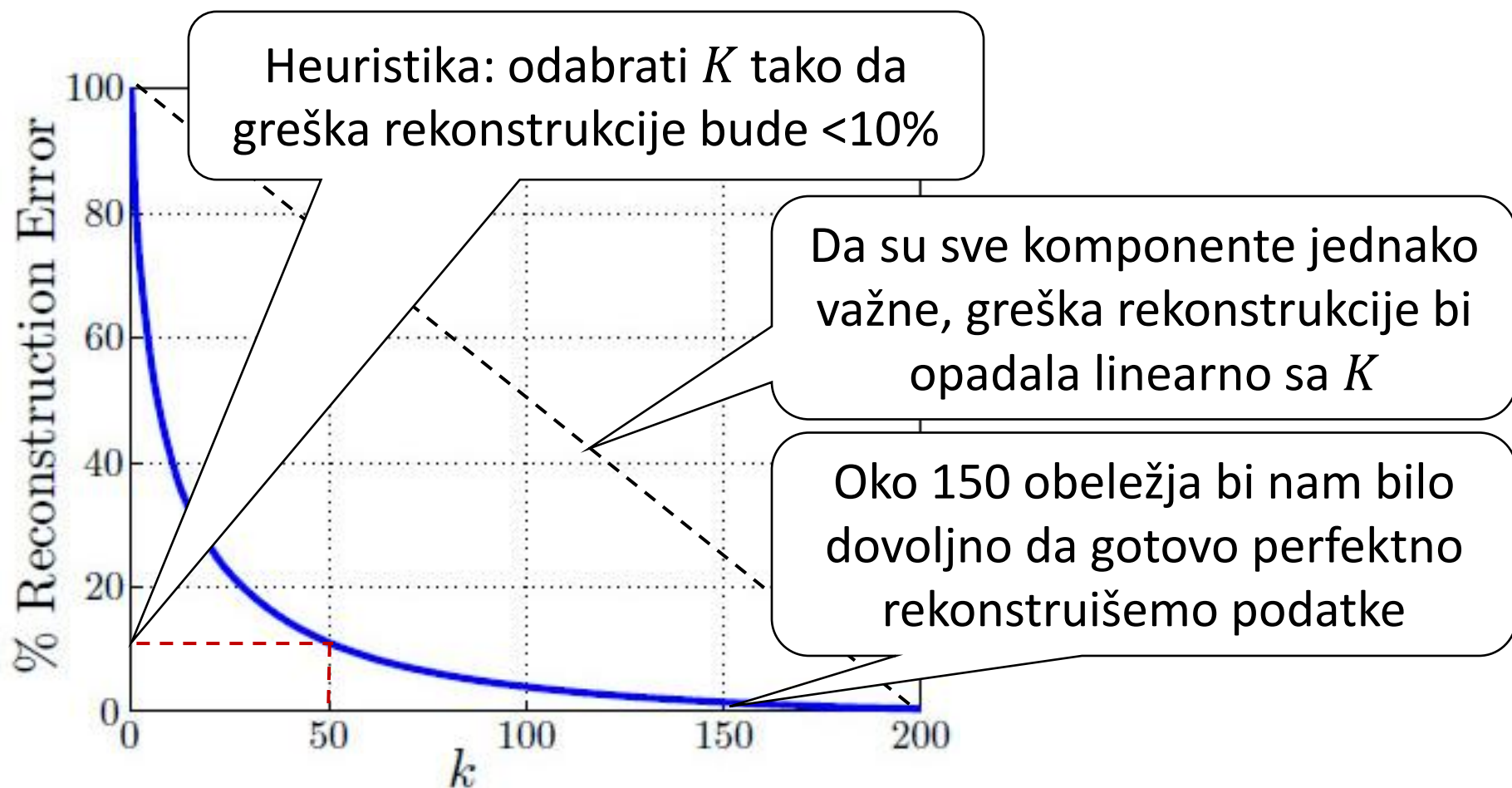
- Ideja: manje obeležja – manje šanse za overfitting
- Ova ideja nije suluda, i može se desiti da dobijemo dobre performanse, ali ovo nije dobar način da adresiramo problem overfitting-a
 - PCA ne uzima u obzir labele y , a odbacuje deo informacija
 - Može se desiti da je deo koji odbacimo veoma važan
- Mnogo bolji način da se adresira overfitting jeste regularizacija
 - Radiće isto ili bolje nego PCA za ovu primenu

Loše upotrebe PCA

- Hipotetički dizajn ML sistema:
 1. Prikupiti trening skup
 2. Primeniti PCA za redukciju dimenzionalnosti
 3. Trenirati model
 4. Testirati na test skupu
- Pre ovog „komplikovanog“ plana – probati primenu algoritma bez PCA na originalnim podacima
- Tek ako to ne uspe (algoritam je previše spor, zahteva previše memorije,...), probati PCA

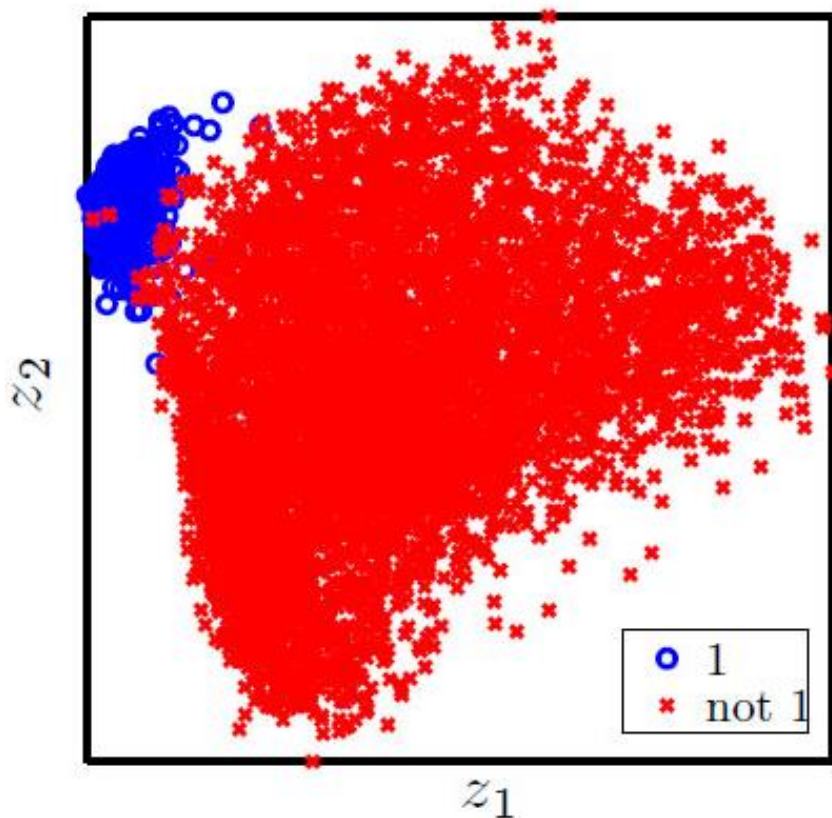
Primer: OCR

- prepoznavanje rukom pisanih cifara sa slika 16×16
- $X \in \mathbb{R}^{7291 \times 256}$ ćemo transformisati u prostor $\mathbb{R}^{7291 \times 2}$



Primer: OCR

- prepoznavanje rukom pisanih cifara sa slika 16×16
- $X \in \mathbb{R}^{7291 \times 256}$ ćemo transformisati u prostor $\mathbb{R}^{7291 \times 2}$



(b) Top-2 PCA-features

- Prikazane su dva najvažnija obeležja dobijena uz pomoć PCA
- Skup podataka je prikazan u novom prostoru i obeležene su instance anotirane kao 1 (plavo) i ostale (crveno)
- Vidimo da nova obeležja omogućavaju prilično jasno razlikovanje klasa

PCA obeležja i ručna konstrukcija obeležja

- **Ručno** konstruisana obeležja: simetriju i intenzitet
- Slično ovim obeležjima, dva obeležja dobijena pomoću PCA omogućavaju prilično jasno razlikovanje klasa
- Glavna prednost PCA obeležja nad ručno konstruisanim obeležjima jeste što su ova automatski generisana – ne moramo znati ništa o problemu prepoznavanja cifara kako bismo ih dobili
- Ovo je takođe njihov najveći nedostatak – dobijena obeležja su dokazano dobra da rekonstruišu originalne ulazne podatke, ali, **nema garancije** da će biti korisna prilikom rešavanja problema koji pokušavamo da rešimo
- U praksi, korišćenje domenskog znanja za konstrukciju obeležja, zajedno sa automatskim metodama za redukciju dimenzionalnosti obično ima najbolje performanse

Primer PCA: ocene opština

- Places Rated Almanac (Boyer and Savageau)
- 329 opština ocenjeno na osnovu sledećih kriterijuma:
 1. Klima i zemljište
 2. Cena stambenog prostora
 3. Zdravstvo i životna sredina
 4. Kriminal
 5. Transport
 6. Obrazovanje
 7. Umetnost
 8. Rekreacija
 9. Ekonomija
- Problem: puno dimenzija – teško za interpretaciju podataka

Primena PCA na podatke

Component	Eigenvalue	Proportion	Cumulative
1	0.3775	0.7227	0.7227
2	0.0511	0.0977	0.8204
3	0.0279	0.0535	0.8739
4	0.0230	0.0440	0.9178
5	0.0168	0.0321	0.9500
6	0.0120	0.0229	0.9728
7	0.0085	0.0162	0.9890
8	0.0039	0.0075	0.9966
9	0.0018	0.0034	1.0000
Total	0.5225		

Ukupna varijansa ($\lambda_1 + \dots + \lambda_9$)

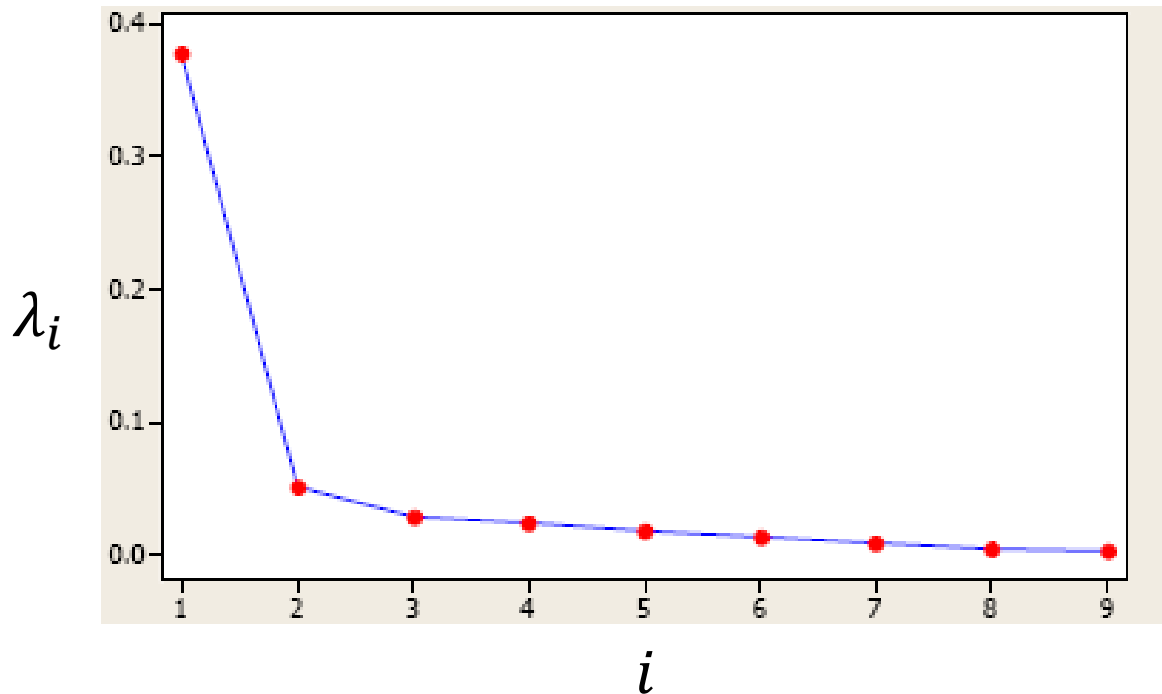
Proportion:

- deo varijanse objašnjen od strane glavne komponente
- Npr., za prvu komponentu $\frac{0.3775}{0.5223} = 0.7227$, odnosno oko 72% varijanse u podacima je objašnjeno prvom glavnim komponentom

Cumulative:

- Kumulativna varijansa dobijena dodavanjem uzastopnih udela u varijansi prvih K glavnih komponenti
- Npr. prve dve glavne komponente zajedno objašnjavaju $0.7227 + 0.0977 = 0.8204$

Odabir K



- Jedan način: odabrati prvih 5 komponenti jer je na taj način zadržano 95% varijanse u podacima. Ovo je razuman procenat ako je naš cilj prediktivno modelovanje
- Drugi način: pogledati grafik na slici – nakon 3. komponente, preostale sopstvene vrednosti su male i približno iste veličine. Prve 3 komponente objašnjavaju 87% varijanse. Ovo je razumno visok procenat ako je naš cilj interpretacija podataka

Prva glavna komponenta

$$Z = XV$$

$$\begin{aligned} Z_1 &= 0.0351 \cdot \text{climate} + 0.0993 \cdot \text{housing} + 0.4078 \cdot \text{health} \\ &+ 0.1004 \cdot \text{crime} + 0.1501 \cdot \text{transportation} + 0.0321 \\ &\cdot \text{education} + 0.8743 \cdot \text{arts} + 0.1590 \cdot \text{recreation} \\ &+ 0.0195 \cdot \text{economy} \end{aligned}$$

- Magnitude koeficijenata predstavljaju udeo originalnih varijabli u datoj glavnoj komponenti
- Ali, imajte na umu da ove magnitude zavise i od varijanse datih varijabli

Interpretacija glavnih komponenti

	Principal Component		
Variable	1	2	3
Climate	0.190	0.017	0.207
Housing	0.544	0.020	0.204
Health	0.782	-0.605	0.144
Crime	0.365	0.294	0.585
Transportation	0.585	0.085	0.234
Education	0.394	-0.273	0.027
Arts	0.985	0.126	-0.111
Recreation	0.520	0.402	0.519
Economy	0.142	0.150	0.239

- U cilju interpretacije, izračunaćemo korelaciju originalnih varijabli sa glavnim komponentama
- Posmatraćemo najsnažnije korelacije po apsolutnoj vrednosti
- Šta se smatra „snažnom“ korelacijom je subjektivno. Ovde su uzete u obzir korelacije preko 0.5

Interpretacija glavnih komponenti

	Principal Component		
Variable	1	2	3
Climate	0.190	0.017	0.207
Housing	0.544	0.020	0.204
Health	0.782	-0.605	0.144
Crime	0.365	0.294	0.585
Transportation	0.585	0.085	0.234
Education	0.394	-0.273	0.027
Arts	0.985	0.126	-0.111
Recreation	0.520	0.402	0.519
Economy	0.142	0.150	0.239

PCA1:

- Uvećava se sa uvećanjem *housing, health, transportation, arts i recreation* – ovih 5 kriterijuma variraju zajedno, ako se jedna poveća, i ostale će
- Komponentu možemo videti kao meru kvaliteta umetnosti, zdravlja, transporta i rekreacije i viših cena nekretnina
- Najjača korelacija je sa umetnošću
- Ima smisla da su opštine sa puno umetnosti i najskuplje

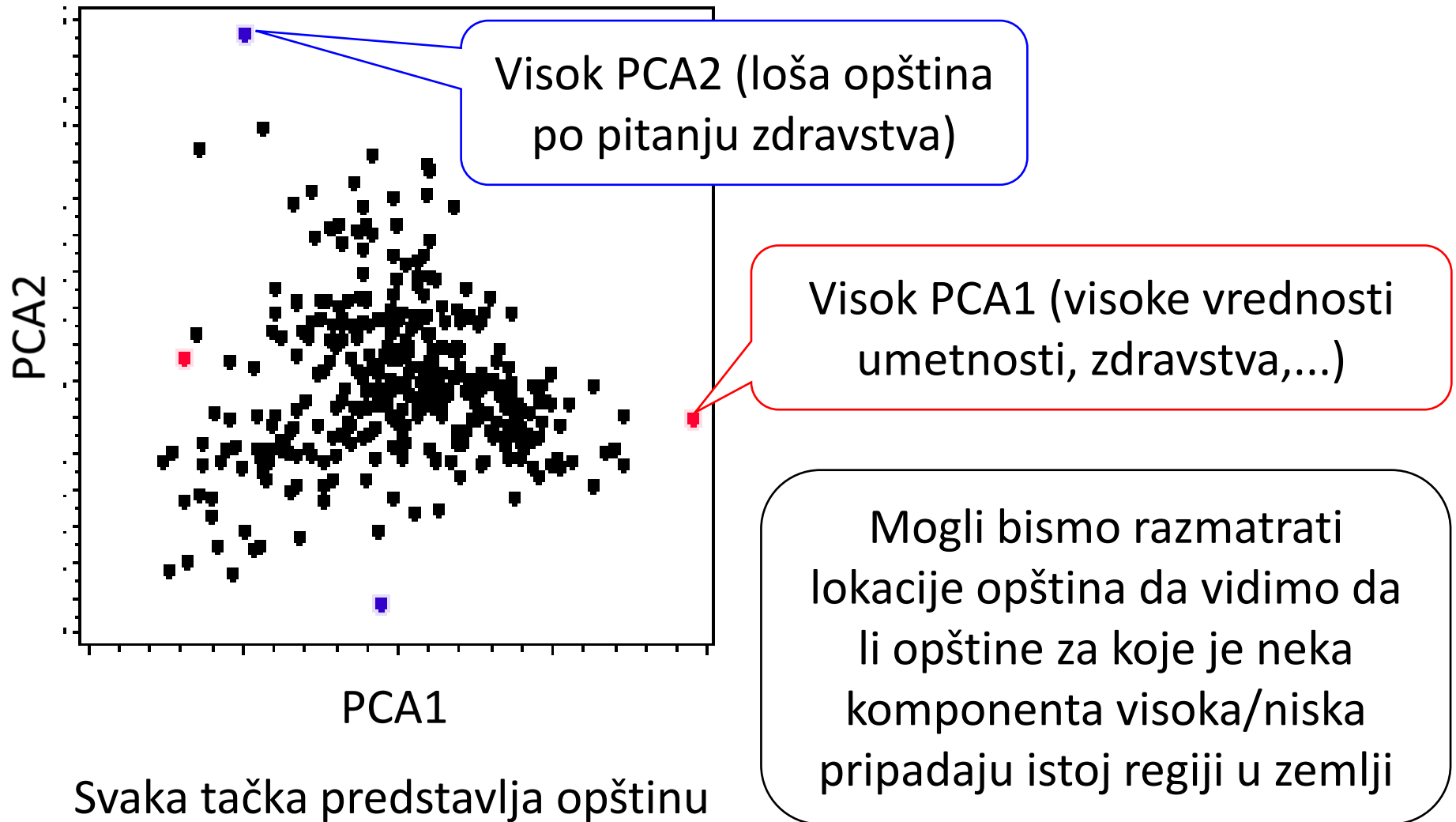
PCA2:

- Uvećava se sa opadanjem kvaliteta zdravlja

PCA3:

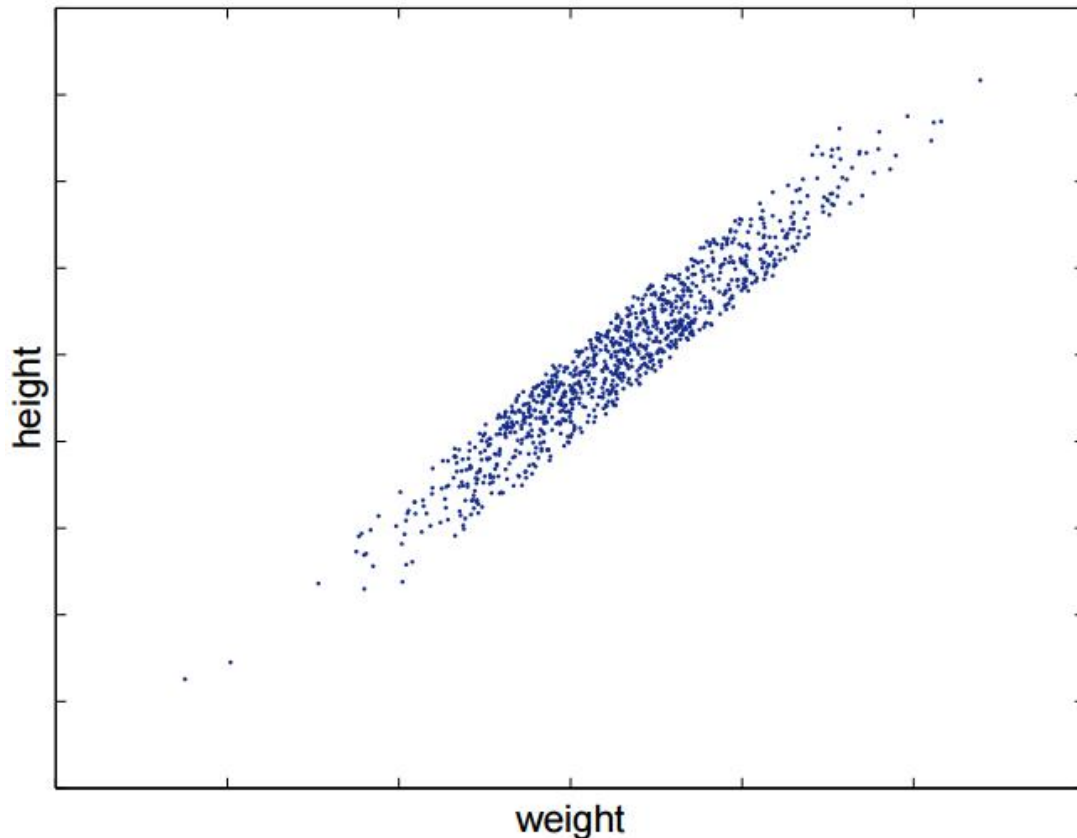
- Uvećava se sa porastom kriminala i rekreacije
- Ovo ukazuje da opštine sa većim kriminalom imaju i više rekreacionih ustanova

Interpretacija glavnih komponenti



Nedostaci PCA: Interpretacija

- Obeležja dobijena pomoću PCA metode su **linearne kombinacije** originalnih obeležja. Ovo je često teško interpretirati

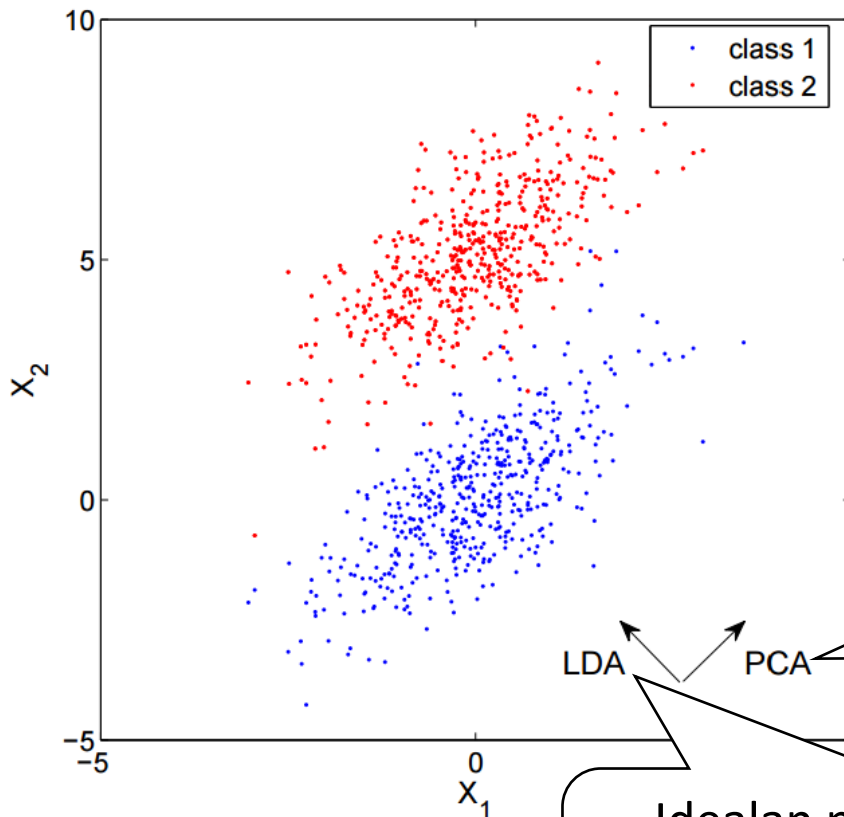


Težinu i visinu menjamo samo jednom koordinatom.

Kako interpretirati ovu novu dimenziju, odnosno, pridodati joj neko fizičko značenje?

Nedostaci PCA: Bez garancije da će pomoći

- Nema garancije da će biti korisna prilikom rešavanja problema koji pokušavamo da rešimo

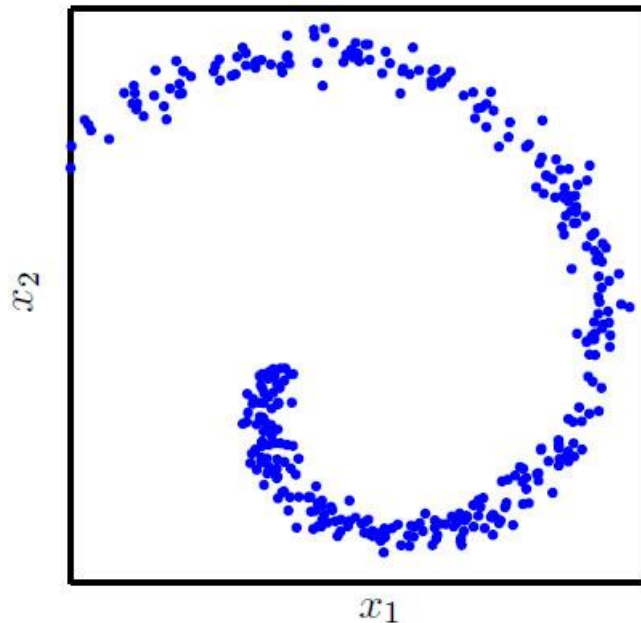


Ako je naš cilj da klasifikujemo tačke, projekcija bi činila više štete nego dobra jer bi se većina crvenih i plavih tačaka preklapala u novoj dimenziji

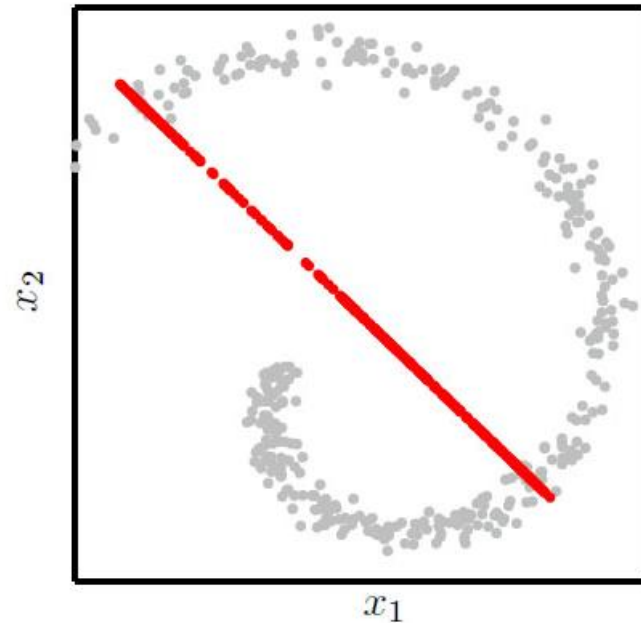
PCA bi prvu glavnu komponentu postavio u pravcu maksimalne varijanse

Idealan pravac projekcije u ovom slučaju je normalan na prvu glavnu komponentu

Nedostaci PCA: Linearnost



(a) Data in \mathcal{X} space



(b) Top-1 PCA reconstruction

- Podaci (a) približno leže na jednodimenzionoj površi (krivoj)
- Međutim, ako pokušamo da rekonstruišemo ove podatke pomoću PCA, rezultati su katastrofalni (b)
- Razlog je što kriva nije linearna, a PCA može da kreira samo linearna obeležja

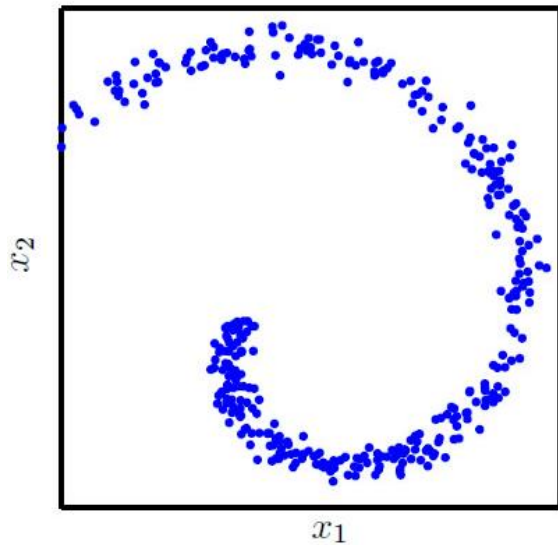
Checklist za primenu PCA

1. Da li očekujete da podaci imaju **linearnu** strukturu
 - Na primer, da li podaci leže u gotovo linearnom potprostoru?
2. Da li donje glavne komponente sadrže uglavnom male slučajne fluktuacije koje liče na šum i treba da budu odbačene?
 - Činjenicu da su fluktuacije male možemo odrediti pomoću greške rekonstrukcije
 - Činjenica da se radi o šumu je ne više od nagađanja

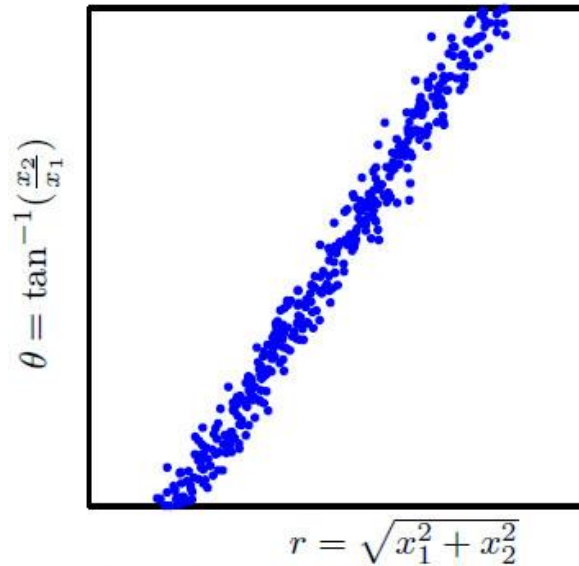
Checklist za primenu PCA

3. Da li ciljna funkcija f zavisi prvenstveno od gornjih glavih komponenti ili su male fluktuacije u donjim glavnim komponentama ključne?
 - Ako je ovo drugo slučaj, PCA neće pomoći u rešavanju datog problema
 - U praksi, teško je proveriti da li je ovo tačno
 - Validacijom možemo proveriti da li koristiti PCA ili ne
 - Obično, odbacivanje donjih komponenti ne odbacuje informacije bitne sa aspekta f
 - A ono što je odbačeno je nadoknađeno smanjenom greškom generalizacije (zbog redukovane dimenzionalnosti)

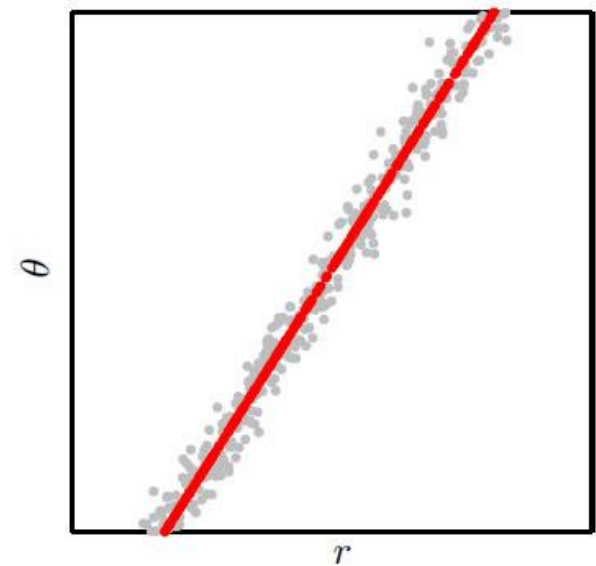
Jedno rešenje: nelinearna transformacija



(a) Data in \mathcal{X} space



(a) Transformed data in \mathcal{Z} space

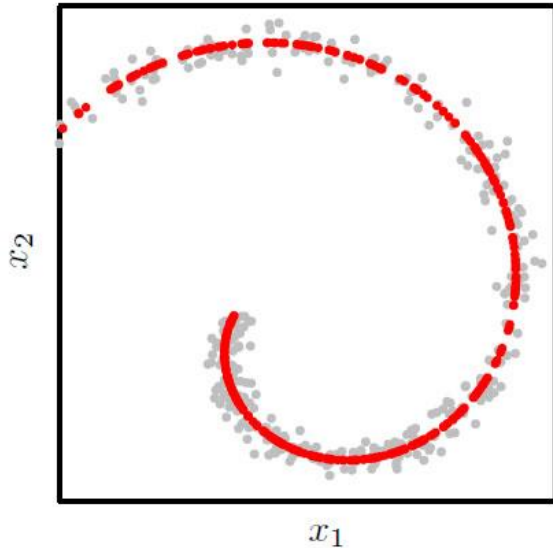


(b) Reconstruction in \mathcal{Z} space

- Možemo primeniti nelinearnu transformaciju na ulazne podatke:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \xrightarrow{\Phi} \begin{bmatrix} r \\ \theta \end{bmatrix} = \begin{bmatrix} \sqrt{x_1^2 + x_2^2} \\ \tan^{-1}\left(\frac{x_2}{x_1}\right) \end{bmatrix}$$

Jedno rešenje: nelinearna transformacija



- Na slici je prikazana rekonstrukcija podataka vraćena u originalni prostor \mathcal{X}
 - Ovo ne mora uvek biti moguće (Φ mora biti invertibilno)
 - Generalno, možemo želiti da primenjujemo Φ koje nije invertibilno, poput polinoma 2. stepena
- Ali, ovo nas ne sprečava da primenjujemo PCA sa nelinearnim transformacijama – naš cilj je predikcija
 1. Transformisaćemo u Z prostor
 2. Primenićemo PCA u Z prostoru
 3. Primenićemo obučavajući algoritam koristeći gornjih K glavnih komponenti u prostoru Z
 4. Prilikom klasifikacije novog primera x , transformisaćemo ga u Z prostor i tamo klasifikovati

PCA sa nelinearnom transformacijom

- Postoje i drugi pristupi za nelinearnu redukciju dimenzionalnosti PCA metode
- Kernel-PCA: način da se kombinuje PCA sa nelinearnim transformacijama obeležja, bez da moramo posetiti Z prostor (slično kernel triku kod SVM)
- Još neki popularni pristupi: Neural-Network auto-encoder, nonlinear principal curves and surfaces
- Neparametarski pristupi: Laplacian Eigenmap, Locally Linear Embedding (LLE)