

PA 18876

STAMPED NUMBER: 30489

Elektrotehnički fakultet
Univerzitet u Beogradu

Mr Marko V. Janković, dipl. inž. el.

Doktorska disertacija

**SAMOORGANIZUJUĆE NEURALNE MREŽE
ZA ANALIZU GLAVNIH KOMPONENATA**



Mr Marko V. Janković, dipl. inž. el.

Datum odbrane: 23. 04. 2006.

- Komisijski članovi:
- Dr Branimir Reljin, red. prof. Elektrotehničkog fakulteta u Beogradu, mentor
 - Dr Sedad Stančević, red. prof. Elektrotehničkog fakulteta u Beogradu
 - Dr Dragan Kandić, red. prof. Mašinskog fakulteta u Beogradu
 - Dr Mirna Reljin, docent Elektrotehničkog fakulteta u Beogradu
 - Dr Zeljka Durić, van. prof. Elektrotehničkog fakulteta u Beogradu

Beograd, 2006.

УНИВЕРЗИТЕТСКА БИБЛИОТЕКА
"СВЕТОСЛАВ МАРКОВИЋ" БЕОГРАД
И. Бр. 138462



Elektrotehnički fakultet Univerziteta u Beogradu

Mr Marko V. Janković, dipl. inž. el.

SAMOORGANIZUJUĆE NEURALNE MREŽE ZA ANALIZU GLAVNIH KOMPONENATA

Doktorska disertacija

Datum usmene odbrane: 23. 03. 2006.

Komisija:

Dr Branimir Reljin, red. prof. Elektrotehničkog fakulteta u Beogradu, mentor

Dr Srđan Stanković, red. prof. Elektrotehničkog fakulteta u Beogradu

Dr Dragan Kandić, red. prof. Mašinskog fakulteta u Beogradu

Dr Irini Reljin, docent Elektrotehničkog fakulteta u Beogradu

Dr Željko Đurović, van. prof. Elektrotehničkog fakulteta u Beogradu

Apstrakt - Ovaj rad je posvećen jednostavnim biološki verovatnim algoritmima za ekstrakciju glavnih/sporednih komponenata i/ili njihovih potprostora iz kovarijanske matrice ulaznog signala, kao i pronalaženju generalnog metoda za transformaciju metoda za analizu glavnih i sporednih potprostora u metode za analizu glavnih i sporednih komponenata. Proučavane su jednostavne, homogene neuralne mreže, bazirane na lokalnim izračunavanjima, što sve zajedno daje dobru osnovu za jednostavnu implementaciju u paralelnom hardveru.

Teorijski doprinos ovog rada se ogleda u sledećem:

- pokazano je da direktna primena Hebovog zakona ne dovodi do divergencije sinaptičkog vektora ako se taj zakon primeni na mreži odgovarajuće strukture;
- predložena je struktura neuralne mreže za izračunavanje PSA, koja je u mnogo čemu slična sa strukturom dela retine kod riba;
- prikazan je generalni metod koji transformiše PSA/MSA metode u PCA/MCA metode i tako omogućava formiranje veoma velikog broja novih PCA/MCA algoritama. Korišćenjem ove transformacije moguće je formiranje homogenih algoritama na bazi Hebovog zakona učenja, koji koriste samo lokalno dostupne podatke za modifikaciju vrednosti sinaptičke matrice, i koji bi onda mogli biti smatrani za biološki verovatne.

Praktična primena originalnih metoda koje su predložene u ovom radu se može naći u mogućem modelovanju računskih principa koji se koriste u realnim neuralnim mrežama i kod jednostavne realizacije PSA/MSA ili PCA/MCA algoritama u paralelnom hardveru.

“Unsupervised neural networks for principal component analysis”

Abstract – This thesis is devoted to the simple biologically plausible algorithms for extraction of the principal/minor components/subspace from input signal covariance matrix, as well as discovery of a general method that transforms principal/minor subspace analysis methods into principal/minor component analysis methods. Analyzed neural networks are simple, their structure is homogeneous and proposed learning rules are based on local calculations. These features make proposed neural networks suitable for implementation in parallel hardware.

Theoretical contributions of this thesis are:

- It is shown that direct implementation of the basic Hebbian scheme would not lead to unrealistic growth of the synaptic strengths, thanks to the adopted network structure;
- Proposition of the PSA algorithm which is implemented in a neural network whose structure shows high degree similarity with a part of the fish retina wiring;
- A new method which transforms PSA/MSA methods into PCA/MCA methods is proposed. By implementation of this method it is possible to create a big number of new PCA/MCA methods. Also, use of the proposed transformation facilitates creation of homogeneous algorithms based on Hebbian learning rule, which use only locally available information for modification of synaptic matrix, and which could be, consequently, considered as a biologically plausible.

Practical implementation of the proposed methods could be found in modeling of the general computational principles which are used in real neural networks, as well as in construction of simple neural networks for PSA/MSA or PCA/MCA which are suitable for realization in parallel hardware.

“Самоорганизующиеся нейронные сети для анализа главных компонентов”

Zahvaljujem se mentoru prof. dr Branimiru Reljinu na dragocenim savetima koje mi je pružio prilikom pisanja ovog rada.

Zahvaljujem se prof. dr Itsuko Teruoka na pomoći u pronalaženju istraživačke pozicije na Saitama Univerzitetu i Elektrotehničkoj laboratoriji u Tsukubi. Istraživanja koja sam obavljao tamo su direktno uticala na izbor oblasti kojom se bavi ova disertacija.

Zahvaljujem se prof. dr Hidemitsu Ogawi na savetima koji su pomogli da neki od mojih rezultata istraživanja postanu razumljiviji. Takođe, zahvaljujem mu se što mi je omogućio jednomesečni boravak na *Tokyo Institute of Technology* i tako mi olakšao postizanje rezultata koji su vezani za vremenski orijentisan hijerarhijski metod.

Zahvaljujem se svojim kolegama iz Centra za automatiku i regulaciju Instituta «Nikola Tesla», koji su mi profesionalnim i prijateljskim odnosom omogućili da, uz brojne redovne obaveze, pronađem dovoljno vremena i energije za pisanje ove teze. Posebno bih se zahvalio kolegama Predragu Ninkoviću, Zoranu Čiriću, Jasni Dragosavac i Vladimiru Vukiću. Zahvalio bih se i Marijanu Stojkoviću na pomoći koju mi je pružio prilikom realizacije grafičkih rešenja i animacija koje su pratile moja izlaganja na konferencijama i pri odbrani teze.

Zahvaljujem se svojim roditeljima na pomoći i razumevanju tokom izrade ove teze. Bez njihove pomoći ovaj proces bi bio svakako mnogo teži i dugotrajniji.

Zahvaljujem se svojoj supruzi čije su razumevanje i podrška u svakom trenutku bili od velikog značaja za završetak ove teze. Njena povremena podsećanja da bi proces pisanja trebao da se okonča su, konačno, dovela i do završetka ovog rada.

Iako je ovaj rad rezultat originalnih naučnih ideja, on je u stvari samo interpretacija nekih priča u kojima me je moja nana, moj najvažniji učitelj i najbolji prijatelj, učila o tajnama života kao što su ljubav i vreme.

Ovaj rad posvećujem svojoj deci.

SADRŽAJ

I Uvod	1
II Veštačke neuralne mreže	6
2.1 Biološki prototip	6
2.2 Veštački neuron	8
2.3 Samoorganizujuće neuralne mreže	10
2.4 Razlike u primeni samoorganizujućih neuralnih mreža u odnosu na mreže koje uče pod nadzorom	11
2.5 Osnovne kategorije samoorganizujućih neuralnih mreža	13
2.5.1 Samoorganizujuće neuralne mreže bazirane na standardnim statističkim metodama	13
2.5.1.1 Hebov zakon učenja	13
2.5.1.2 Ojin zakon učenja	15
2.5.1.3 Procesiranje signala "na slepo" (BSP)	16
2.5.2 Samoorganizujuće neuralne mreže i klasterovanje podataka	19
III Analiza glavnih i sporednih komponentata	22
3.1 Analiza glavnih komponentata (PCA)	22
3.1.1 Određivanje sopstvenih vrednosti	22
3.1.2 Procena kovarijansne matrice	25
3.2 Potprostori signala i šuma – automatski izbor dimenzije PCA	27
3.3 Osnovne osobine PCA	31
3.4 Ekstrakcija glavnih komponenta korišćenjem principa optimalne kompresije-rekonstrukcije	33
3.5 Osnovne funkcije cene i adaptivni algoritmi za PCA	36
3.5.1 Rejljev koeficijent – osnovne osobine	36
3.5.2 Osnovne funkcije cene zasekvencijalno izračunavanje glavnih i sporednih komponentata	38
3.5.3 Brzi PCA algoritam baziran na power metodu	41
3.5.4 Inverzni iterativni power metod	44
3.5.5 Robusna PCA	45
3.6 Adaptivni algoritmi za sekvencijalnu ekstrakciju sporednih komponentata	49

IV Paralelni algoritmi za estimaciju glavnih komponenta, sporednih komponenta i njihovih potprostora	54
4.1 Funkcija cene za paralelnu ekstrakciju glavnih komponenta i njihovih potprostora	54
4.2 Potprostorni algoritam	58
4.3 Gradijent funkcije cene $J(W)$	62
4.4 Analiza stabilnostipredloženog algoritma	67
V Biološki verovatan zakon učenja za izračunavanje dominantne glavne komponente	69
5.1 Jednostavan biološki verovatan ∞ OH model neurona	70
5.2 Matematička analiza ∞ OH algoritma	74
5.3 Poređenje sa nekim poznatim algoritmima za ekstrakciju dominantne glavne komponente	83
5.4 Generalizacija predloženog algoritma za slučaj mreža sa više izlaza	86
5.5 Analiza u slučaju kada ulazna sekvenca nema srednju vrednost nula	87
VI Modulisani Hebov (MH) metod - biološki verovatan metod za izračunavanje glavnog potprostora (PSA)	89
6.1 Matematička analiza MH metoda	92
6.2 Poređenje usvojene strukture sa delom retine kod riba	96
6.3 Modulisani Heb-Oja zakon učenja	98
6.4 Matematička analiza MHO metoda	99
6.5 Rezultati simulacija	110
6.6 Poređenje sa poznatim PSA algoritmima	112
VII Vremenski orijentisan hijerarhijski metod (TOHM) za izračunavanje PCA i MCA	119
7.1 Matematička analiza predloženog principa	124
7.2 Jedan primer primene TOHM metoda za PCA	128
7.3 Analiza asimptotske stabilnosti rešenja	134

7.4 Primer primene TOHM metoda za MCA 139

7.5 Generalizovani TOHM 148

Lista korišćenih skraćenica

VIII Zaključak 149

Literatura 154

- AKT Adaptive Resonant Theory - teorija adaptivne rezonancije
- BCM Bienenstock Cooper Munro
- BIC Bayesian Information Criterion - Bajesov informacijski kriterijum
- BMU Best Matching Unit - najbolja jedinica
- BSB Blind Signal Extraction - izdvajanje signala «na slepo»
- BSP Blind Signal Processing - procesiranje signala «na slepo»
- BSS Blind Signal Separation - razdvajanje signala «na slepo»
- EOF Empirical Orthogonal Function - empirijska ortogonalna funkcija
- FA Factor Analysis - faktor analiza
- FLA Family part of the Learning Algorithm - porodični deo algoritma za učenje
- FLIN Foldiak Lateral Inhibition Network - Foldiakova mreža sa bočnom inhibicijom
- FP Family Part - porodični deo
- GHA General Hebbian Algorithm - opšti Hebov algoritam
- GTOHM Generalized Time Oriented Hierarchical Learning - generalizovani vremenski orijentisan hijerarhijski metod
- ICA Independent Component Analysis - analiza nezavisnih komponenti
- ILA Individual part of the Learning Algorithm - individualni deo algoritma za učenje
- LA Learning Algorithm - algoritam za učenje
- MBD Multichannel Blind Deconvolution - višekanalna dekonvolucija «na slepo»
- MCA Minor Component Analysis - analiza sporodnih komponenti
- MCD Minimum Covariance Determinant - minimalna kovarijansa determinanta

Lista korišćenih skraćenica

AIC	Akaike Information Criterion – Akaikeov informacioni kriterijum
ART	Adaptive Resonant Theory – teorija adaptivne rezonancije
BCM	Bienestock Cooper Munro
BIC	Bayesian Information Criterion – Bajesov informacioni kriterijum
BMU	Best Matching Unit – najpribližnija jedinica
BSE	Blind Signal Extraction – izdvajanje signala «na slepo»
BSP	Blind Signal Processing – procesiranje signala «na slepo»
BSS	Blind Signal Separation – razdvajanje signala «na slepo»
EOF	Empirical Orthogonal Function – empirijska ortogonalna funkcija
FA	Factor Analysis – faktor analiza
FLA	Family part of the Learning Algorithm – «porodični» deo algoritma za učenje
FLIN	Foldiak Lateral Inhibition Network – Foldiakova mreža sa bočnom inhibicijom
FP	Family Part – porodični deo
GHA	General Hebbian Algorithm – opšti Hebov algoritam
GTOHM	Generalized Time Oriented Hierarchical Learning – generalizovani vremenski orijentisan hijerarhijski metod
ICA	Independent Component Analysis – analiza nezavisnih komponentata
ILA	Individual part of the Learning Algorithm – individualni deo algoritma za učenje
LA	Learning Algorithm – algoritam za učenje
MBD	Multichannel Blind Deconvolution – višekanalna dekonvolucija naslepo
MCA	Minor Component Analysis – analiza sporednih komponentata
MCD	Minimum Covariance Determinant – minimalna kovarijansna determinanta

MDL	Minimum Description Length – minimalna dužina zapisa
MH	Modulated Hebbian – modulisani Hebov metod
MHO	Modulated Hebb Oja – modulisani Heb-Oja
MIBS	Minka Bayesian model Selection – Minka-Bajesov metod za selekciju Modela
MIMO	Multiple-Input Multiple-Output – više ulaza – više izlaza
MSA	Minor Subspace Analysis – analiza sporednog podprostora
PCA	Principal Component Analysis – analiza glavnih komponentata
PSA	Principal Subspace Analysis – analiza glavnog podprostora
REC	Recurrent Error Correction – rekurentna korekcija greške
RK	Rayleigh Quotient – Rejljev koeficijent
SEC	Symmetrical Error Correction – simetrična korekcija greške
SLA	Subspace Learning Algorithm – podprostorni algoritam
SMCA	Single Minor Component Analysis – analiza najsporednije glavne komponente
SNR	Signal-Noise Ratio – odnos signal šum
SOM	Self-Organized Maps – samoorganizujuće mape
SPCA	Single Principal Component Analysis – analiza dominantne glavne komponente
TOHM	Time-Oriented Hierarchical Method – vremenski orijentisan hijerarhijski metod
WTA	Winner Takes All – pobedniku pripada sve
∞ OH	∞ Oja Heb

I Poglavlje

Uvod

Razumevanje principa izračunavanja koje koristi mozak za svoje funkcionisanje je jedan od najinteresantnijih i najtežih naučnih problema. Iako u nekoliko poslednjih decenija postoji veliki broj otkrića koja rasvetljavaju strukturu mozga na nekoliko različitih nivoa, još uvek nije moguće razumeti na koji način nervni sistem omogućava čoveku da vidi, čuje, uči, pamti događaje, planira aktivnosti ili donosi odluke.

Istraživanja na polju obrade slike, robotike i ostalih grana veštačke inteligencije su pokazala da i najjednostavniji zadaci koje mozak ljudi ili životinja obavlja bez skoro ikakvog napora, kao što su prepoznavanje objekata, koordinacija pokreta, ili snalaženje i kretanje u nepoznatom okruženju, predstavljaju veoma složene računске probleme. Smatra se da je glavni razlog za dosadašnji neuspeh svih mašina koje je stvorio čovek da emuliraju ili se bar malo približe performansama životinja ili čoveka, leži u činjenici da je za rešavanje i naizgled veoma jednostavnih problema potrebno veoma veliko znanje o statističkim i drugim karakteristikama signala koje iz okruženja prikupljaju senzori (naša čula). Sposobnost mozga da na osnovu tih znanja kodira signale, uči i pristupa znanju (memoriji) na efikasan način predstavlja osnovu za njegovu sposobnost da efikasno rešava mnogobrojne probleme.

Računarska nauka o neurologiji je naučna oblast koja proučava kako se realni signali (električni i hemijski) koriste u mozgu u cilju ispravne reprezentacije i obrade prikupljenih informacija. U cilju pronalaženja rešenja, obično se koriste pojednostavljeni računarski modeli koji bi trebalo da na nekom nivou apstrakcije eventualno omogućе shvatanje principa rada neuralnih mreža u nervnom sistemu. Glavna prednost takvih modela leži u njihovoj internoj konzistentnosti kao i činjenici da mogu biti analizirani, bilo analitički ili korišćenjem računarskih simulacija.

Pojednostavljeni modeli se veoma često koriste u raznim granama nauke. Isto tako, korišćenje uprošćenih modela nervnog sistema omogućava analizu osnovnih problema (principa) izračunavanja čije bi razumevanje vodilo ka razumevanju principa nekih od procesa izračunavanja koji se odvijaju u mozgu.

Jedan od mogućih pristupa je da se pri modeliranju mozga počne od naših saznanja iz anatomije i psihologije i da se postepeno razvijaju modeli koji bi uključivali sve više i više detalja, a koji bi se ubacivali na osnovu pretpostavki, u slučaju da ne postoje činjenice koje bi mogle da potvrde njihovo prisustvo u mozgu. To je takozvani inverzni inženjerski pristup, čija je primena veoma teška čak i slučaju kada se radi o sistemima srednje složenosti, i to čak i u slučaju kada znamo osnovne principe na kojima se sistem bazira. To je, na primer, slučaj u pokušaju otkrivanja osobina nepoznatog integrisanog kola. Naravno, situacija je mnogo složenija kada se radi o mozgu. Zato se mnogo privlačnijom čini ideja da se pođe od pokušaja da se identifikuju ciljevi proračuna, razmotre mogući mehanizmi za realizaciju tih ciljeva i da se tek onda testira stepen sličnosti predloženog modela i stvarnog sistema. Takav model se od realnog sistema može razlikovati u mnogim aspektima, ali je svakako nemoguće razumeti kompleksan sistem bez razumevanja nekih osnovnih principa na kojima je on baziran. U tom slučaju se modeliranje i testiranje sistema obavljaju istovremeno.

Pri kreiranju modela potrebno je napraviti veliki broj uprošćavanja iz dva razloga – jedan često leži u nedostatku adekvatnog znanja o detaljima sistema, a drugi u ograničenim računarskim resursima sa aspekta potrebnog vremena za testiranje pretpostavke. Međutim, čak i u slučajevima kada dva gore pomenuta problema ne predstavljaju ograničenje, jednostavniji model se može smatrati poželjnijim ukoliko dodatni detalji ne doprinose suštinskom poboljšanju modela. Model sa manje parametara je neprecizniji u modelovanju dostupnih podataka ali ima veće izgleda za korektnu generalizaciju novih podataka.

Istraživači u oblasti neuralnih mreža se u suštini bave sledećim problemom: kako mreža jednostavnih procesnih jedinica koje imaju veliki broj međusobnih veza može da se upotrebi za izvođenje kompleksnih (ili barem netrivialnih) izračunavanja. Tipično, veštačka neuralna mreža se sastoji od velikog broja jedinica koje imaju određene sličnosti sa karakteristikama realnih neurona, kao što su međusobne veze koje podsećaju na aksone i dendrite kao i promenljive težine tih veza koje podsećaju na sinapse. Same jedinice najčešće obavljaju veoma jednostavna izračunavanja (obično izračunavaju težinsku sumu ulaza) i daju skalarnu vrednost na izlazu koja se dalje prosleđuje ka drugim jedinicama. I pored jednostavnosti osnovnih jedinica, mreže koje te jedinice čine mogu izvršavati veoma različita izračunavanja, usled velikog broja specifičnih međusobnih veza. Za razliku od većine konvencionalnih

algoritama, neuralne mreže poseduju inherentnu sposobnost učenja. Učenje se obavlja modifikacijom veza koje spajaju pojedine jedinice, a modifikacija se obavlja na osnovu različitih algoritama za učenje. Primenom takvih algoritama mreža se, u stvari, obučava za izvođenje određenih kompleksnih algoritama, umesto da bude programirana za njihovo izvođenje. Interesantno je da takvi algoritmi veoma često prevazilaze performanse konvencionalnih kompjuterskih programa, posebno u problemima kao što su obrada slike i govora ili prepoznavanje oblika, pošto je za tu vrstu problema veoma teško formalizovati rešenje problema, a prema tome i napisati odgovarajući program. Druga osobina neuralnih mreža je brzina odziva - u situaciji kada je mreža dobro obučena njen odziv je veoma brz, skoro trenutani. Za razliku od konvencionalnih računarskih algoritama koji izračunavaju rešenje, neuralne mreže su suštinski bazirane na pamćenju podataka i interpolaciji između njih.

Ukoliko se želi korišćenje neuralnih mreža za modelovanje procesa izračunavanja u mozgu, moraju se napraviti određene pretpostavke i onda izvršiti proveru da li one mogu biti podržane ili odbačene na osnovu poznatih anatomskih i fizioloških otkrića. Kako je trenutno poznavanje funkcionisanja nervnog sistema još uvek na relativno niskom nivou, to otvara prostor za postavljanje velikog broja hipoteza koje se ne mogu definitivno ni potvrditi ni odbaciti.

Postoji veliki broj hipoteza o potencijalnim mogućnostima izračunavanja pojedinačnog neurona. Jedna od najčešće korišćenih hipoteza je da neuron vrši linearno sumiranje svojih ulaza. Iako su postojale i drugačije pretpostavke (o mnogo većoj sofisticiranosti izračunavanja koje obavlja neuron) obično su anatomske i fiziološke analize pokazivale da je računarska sposobnost pojedinih neurona strogo ograničena i da je izvestan stepen sofisticiranosti od sekundarnog značaja. Takođe, pretpostavka o linearnom sumiranju izgleda prihvatljiva imajući u vidu rezultate koje su objavili vizuelni neurofiziolozi za odziv većeg broja ćelija vizuelnog korteksa [Enroth, 1966], [Movshon, 1978].

Druga pretpostavka koja se često koristi je da neuron komunicira sa ostalim neuronima, sa kojima je povezan, putem signala koji odgovara frekvenciji impulsa na njegovom izlazu (aksonu). Iako postoje pretpostavke da fina vremenska struktura impulsa predstavlja specifičan kod [Cattaneo, 1981], [Dayhoff, 1983], [Sherry, 1982] detaljnijom analizom je dokazano [Douglas, 1988], [Douglas, 1991], [Young, 1991] da, čak i slučaju da međusobni položaj impulsa nosi nekakvu količinu informacija, te informacije su od sekundarnog značaja.

Možda najvažnija hipoteza oko modela veštačkih neuralnih mreža, kao modela nervnog sistema, odnosi se na zakone učenja koji su odgovorni za modifikaciju sinaptičkih vrednosti.

Jedna od glavnih hipoteza je da zakon učenja poseduje osobinu lokalnosti, što znači da modifikacija pojedinačne sinapse treba što manje da se oslanja na informacije o vrednostima drugih sinapsi pogotovo ako su one vezane za druge neurone. Za ovu pretpostavku postoji sve veći broj dokaza koji potiču iz istraživanja biofizičkih mehanizama odgovornih za proces učenja u nervnom sistemu (na primer [Brown, 1990]).

Deo nervnog sistema o kome se danas najviše zna je onaj deo koji prikuplja i obrađuje informacije sa senzora (čula). Iako još uvek nije jasno kako mozak vrši interpretaciju i koristi podatke sa miliona senzora, interesantno je da mi, kao živa bića, pri korišćenju čula ne doživljavamo informacije koje primamo sa tih senzora kao promene u pojedinim komponentama matrica signala, već detektujemo promene na nivou objekata, situacija ili događaja u okruženju. Poznato je da signali iz našeg okruženja nisu potpuno slučajni, već da (kao na primeru čula vida) signali koji potiču od istog objekta imaju visok stepen korelacije i da su razdvojeni objekti međusobno relativno nezavisni. Na primer, skupovi signala koji su jako korelisani, kao što su leva i desna strana lica, ne smatraju se razdvojenim objektima. Odatle je jasno da bi jedan način za razlikovanje objekata bio traženje skupova signala koji pokazuju visok stepen međusobne korelacije i istovremeno su relativno nekorelisani sa drugim skupovima. Smatra se da delovi mozga koji služe za prijem signala sa senzora i njihovu inicijalnu obradu, kao što je na primer retina, imaju zadatak da smanje redundantnost komponenta ulaznog signala, kako bi se mogla napraviti njegova efikasnija reprezentacija. Osnovna ideja je da retina ili drugi slični organi, imaju zadatak da izvrše inicijalnu dekorelaciju između reprezentacionih jedinica (neurona), koje bi se kasnije, u sledećem stepenu procesiranja, svele na statistički nezavisne jedinice, a za koje se veruje da bi omogućile najefikasnije procesiranje u višim delovima korteksa. U ovom radu će biti analizirani potencijalni mehanizmi za uklanjanje korelacije među ulaznim signalima. Jedan od načina da se to učini je primena analize glavnih komponenta (PCA – engleski Principal Component Analysis). Biće predloženo više algoritama koji bi se mogli smatrati biološki mogućim PCA algoritmima. Kao ekstenzije PCA algoritama biće razmatrani i analiza sporednih komponenta (MCA – engleski Minor Component Analysis) kao i analiza glavnih i sporednih potprostora (PSA/MSA – engleski Principal/Minor Subspace Analysis).

U drugom poglavlju je dat prikaz veštačkih neuralnih mreža. Poglavlje se uglavnom bavi samoorganizujućim neuralnim mrežama. Tu su objašnjeni glavni pravci razvoja i ukratko prikazani najpoznatiji modeli. Takođe, tu će biti objašnjena osnovna ideja na kojoj se bazira učenje u samoorganizujućim neuralnim mrežama, a koja se ogleda u Hebovom zakonu učenja.

Definicija, područje primene i načini realizacije PCA su analizirani u trećem poglavlju. U ovom poglavlju su prikazane najvažnije osobine PCA, objašnjeni osnovni principi za izračunavanje PCA i prikazani sekvencijalni algoritmi koji se koriste za izračunavanje PCA u neuralnim mrežama. Takođe, ovo poglavlje sadrži i definiciju MCA kao i prikaz osnovnih algoritama koji se koriste za sekvencijano izračunavanje MCA u neuralnim mrežama.

Najznačajniji poznati algoritmi za paralelnu ekstrakciju PCA (kao i analizu sporednih komponenata - MCA) u samoorganizujućim neuralnim mrežama će biti prikazani u četvrtom poglavlju.

Poglavlja pet, šest i sedam sadrže originalni doprinos ovog rada.

Peto i šesto poglavlje sadrže rezultate istraživanja u vezi sa biološki verovatnim strukturama i algoritmima za implementaciju PSA/PCA u neuralnim mrežama. U petom poglavlju je prikazan algoritam ∞ OH za ekstrakciju dominantne glavne komponente koji zahvaljujući usvojenoj strukturi neuralne mreže omogućava direktnu primenu Hebovog zakona učenja, bez uvođenja bilo kakvih dodatnih članova za stabilizaciju. Imajući u vidu da su sva izračunavanja bazirana na lokalno dostupnim vrednostima, ovaj algoritam se može smatrati biološki verovatnim. Šesto poglavlje sadrži algoritme za izračunavanje PSA koji su nazvani MH (modulisani Hebov) i MHO (modulisani Heb-Ojin), i koji su inspirisani delom strukture retine kod riba. Ovi algoritmi koriste osnovni Hebov zakon sa dodatnim modulišućim članom. Algoritam MH predstavlja ekstenziju ∞ OH za slučaj kada je broj neurona na izlazu veći od jedan, pri čemu su lokalne povratne veze zamenjene globalnom povratnom vezom. Algoritmi MH i MHO predstavljaju kompromis između broja globalnih izračunavanja i brzine konvergencije neuralne mreže za izračunavanje PSA.

Generalni metod za transformaciju metoda za analizu glavnih i sporednih potprostora u metode za analizu glavnih i sporednih komponenata je prikazan u sedmom poglavlju. Prema saznanjima autora i na osnovu raspoložive literature, ovaj metod je prvi metod takve vrste. Pokazano je da se PSA/MSA algoritmi mogu transformisati u PCA/MCA algoritme bez uvođenja nelinearnosti ili asimetrije u neuralnu mrežu. Princip koji omogućava ovu transformaciju se bazira na vremenski orijentisanoj hijerarhijskoj strukturi – različiti delovi algoritma se obavljaju na različitim vremenskim skalama.

U zaključku je data rekapitulacija rada i ukazano je na moguće pravce daljeg razvoja istraživanja.

Korišćena literatura (oko 130 referenci) je prikazana na kraju ove disertacije.

II Poglavlje

Veštačke neuralne mreže

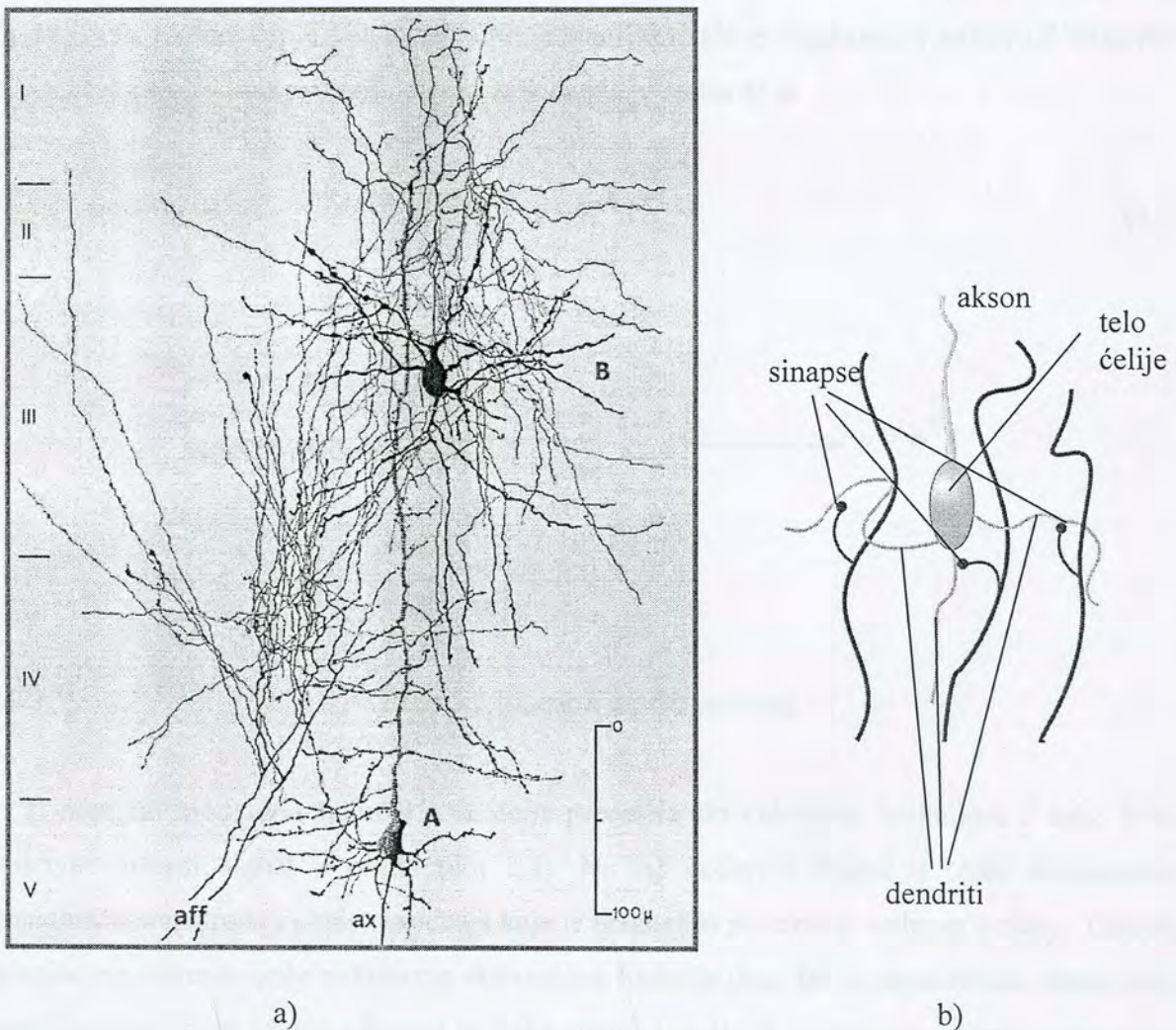
Veštačke neuralne mreže su razvijene u veoma različitim konfiguracijama. Uprkos velikom broju različitih konfiguracija sva rešenja u mnogo čemu imaju mnogo sličnosti. U ovom poglavlju će biti opisano šta se u najopštijem smislu podrazumeva pod veštačkim neuralnim mrežama, pri čemu će akcenat biti stavljen na samoorganizujuće neuralne mreže.

2.1 Biološki prototip

Veštačke neuralne mreže su biološki inspirisane – istraživači obično misle na organizaciju mozga kada razmatraju nove strukture mreža i algoritme koji se u njima primenjuju. I tu negde se može i završiti priča o sličnosti između bioloških uzora i njihovih veštačkih imitacija. Razlog za to je činjenica da se i pored velikog napretka u proučavanju mozga i njegovih principa rada, još uvek prilično malo zna o tome. Stoga je teško naći principe koji bi sa sigurnošću bili pokazatelj o uspešnosti veštačke neuralne mreže u emulaciji pojedinih ili ukupnih funkcija mozga. Iz tog razloga, istraživači na polju neuralnih mreža moraju da pokušavaju da pronađu strukture koje mogu realizovati korisne funkcije, a koje bi na osnovu nekih dodatnih principa mogle biti smatrane za manje ili više verovatne modele delova ljudskog mozga

Ljudski nervni sistem je sagrađen od nervnih ćelija i njegova složenost je veoma velika. Procenjeno je da oko 10^{11} - 10^{12} nervnih ćelija učestvuje u oko 10^{15} međuveza preko prenosnih puteva koji mogu biti i do jednog metra dužine. Sve nervne ćelije imaju mnogo zajedničkih osobina sa ostalim ćelijama u telu, ali poseduju jedinstvenu karakteristiku da su u stanju da prime, procesiraju i prosleđuju elektrohemijske signale preko neuralnih veza koje čine komunikacioni sistem mozga.

Slika 2.1 prikazuje strukturu para tipičnih bioloških neurona. Dendriti (nervni završeci) se prostiru od tela ćelije ka drugim neuronima gde primaju signale na konekcionim tačkama koje se nazivaju sinapse. Sa prijemnih strana sinapsi ti ulazi se odvođe do tela ćelije. Tu se sumiraju, pri čemu neki signali imaju tendenciju da pobude ćeliju dok drugi imaju tendenciju da je inhibiraju. Kada kumulativna eksitacija pređe određeni prag, ćelija počinje da generiše izlaz u vidu povorke impulsa i tako putem aksona prosleđuje signale ka drugim neuronima. Ovo predstavlja osnovni princip funkcionisanja nervne ćelije u odnosu na koji postoje brojni izuzeci. Bez obzira na to, većina veštačkih neuralnih mreža modeluje ovakav način rada.



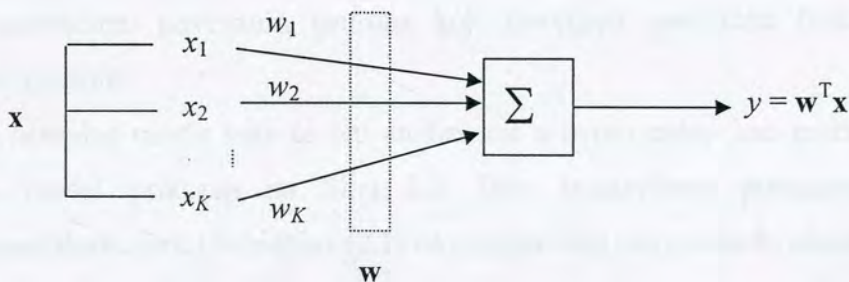
Slika 2.1 a) Prikaz para realnih neurona; b) uprošćeni prikaz neurona



2.2 Veštački neuron

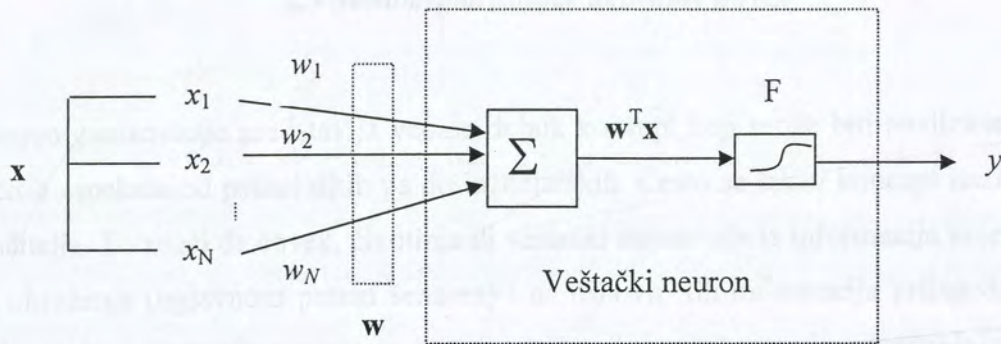
Veštački neuroni imaju zadatak da imitiraju osnovne karakteristike bioloških neurona. U principu ulaze za pojedinačni neuron, predstavljaju izlazi drugih neurona. Svaki ulaz se množi sa odgovarajućom težinom, koja predstavlja sinaptičku «snagu», i onda se svi ti ponderisani ulazi sabiraju kako bi se dobio ukupni aktivacioni nivo neurona. Slika 2.2 prikazuje model za implementaciju ove ideje. Ulazi, koji odgovaraju signalima na sinapsama, su označeni vektorom \mathbf{x} (čije su komponente x_1, x_2, \dots, x_K). Signal je pomnožen sinaptičkim vektorom \mathbf{w} (koji čine težine w_1, w_2, \dots, w_K) i uveden u sumator Σ . Svaka težina sinaptičkog vektora odgovara «snazi» pojedinačne biološke sinaptičke veze. Sumacioni blok približno odgovara funkciji tela nervne ćelije. Sumator sabira ponderisane ulaze algebarski i proizvodi izlaz koji je označen sa y . U kompaktnoj notaciji to se može predstaviti sa

$$y = \mathbf{w}^T \mathbf{x}. \quad (2.1)$$



Slika 2.2 Linearni model neurona

U mnogim modelima suma $\mathbf{w}^T \mathbf{x}$ se dalje procesira aktivacionom funkcijom F kako bi se proizveo izlazni signal y (vidi sliku 2.3). Na taj način se uzima u obzir nelinearnost pojedinačnog neurona i oblast zasićenja koja je neizbežno prisutna u realnom slučaju. Takođe, usvajanjem odgovarajuće nelinearne aktivacione funkcije (kao što su sigmoidalna funkcija ili tanh funkcija) može se dati odgovor na neka pitanja kao što je, na primer, pitanje postavljeno od strane Grosberga [Grossberg, 1973], a koje se tiče na primer dileme šum/zasićenje – drugim rečima, kako ista mreža može da obrađuje i male i velike signale.



Slika 2.3 Nelinearni model neurona

Naravno gore predloženi model ne uzima u obzir brojne karakteristike realnog neurona kao što su vremenska kašnjenja, koja utiču na dinamiku sistema, ili problemi sinhronizma, ili frekvencijske modulacije signala. I pored toga što ti nedostaci predstavljaju ograničenja u modelovanju realnog neurona, ovaj model se smatra prihvatljivim jer mreže čija je osnovna jedinica mogu procesirati signale i proizvoditi rezultate koji podsećaju na rezultate procesiranja realnih neurona.

Više međusobno povezanih neurona koji obavljaju specifičnu funkciju se nazivaju neuralnom mrežom.

Za sve neuralne mreže koje će biti analizirane u ovom radu, kao model neurona će biti prihvaćen model prikazan na Slici 2.2. Biće izostavljeno procesiranje nelinearnom aktivacionom funkcijom i jednačina (2.1) će predstavljati vezu između ulaznih signala i izlaza neurona. Takav model daje mogućnost formiranja neuralnih mreža čije prenosne funkcije podsećaju na funkcije realnih neuralnih sklopova i to pre svega onih koje se bave prvostepenom obradom signala sa senzora (čula).

U principu, veštačke neuralne mreže se dele po načinu učenja na mreže koje uče pod nadzorom i samoorganizujuće neuralne mreže, koje će dalje biti analizirane u ovom radu.

2.3 Samoorganizujuće neuralne mreže

Samoorganizovanje predstavlja veoma dubok koncept koji može biti analiziran sa veoma različitim aspektata od psiholoških pa do inženjerskih. Često se takav koncept naziva i učenje bez učitelja. To znači da čovek, životinja ili veštački sistem uče iz informacija koje dobijaju iz svog okruženja (uglavnom putem senzora) i na osnovu tih informacija prilagođavaju svoje ponašanje bez dodatnih uputstava o tome kako da povežu svoja zapažanja sa željenim ponašanjem (što je slučaj kod učenja pod nadzorom) i bez ikakvih ideja da li je rezultujuće ponašanje ispravno ili ne (što je slučaj kod *reinforcement* učenja). Obično, rezultat učenja bez nadzora je nova interpretacija ili reprezentacija ulaznih signala koja bi trebalo da rezultuje poboljšanim odzivom na slične situacije ili na donošenje boljih odluka u budućnosti.

U oblasti mašinskog učenja i veštačke inteligencije reprezentacija ulaznih podataka predstavlja skup koncepata i pravila koja povezuju te koncepte, što omogućava simboličko objašnjenje podataka. U oblasti veštačkih neuralnih mreža reprezentacija može predstavljati grupisanje podataka, diskretno mapiranje ili kontinualno nižedimenziono projektovanje na mnogostrukost u vektorskom prostoru definisanom ulaznim podacima, čiji je mogući zadatak objašnjenje strukture podataka ili otkrivanje međusobnih zavisnosti.

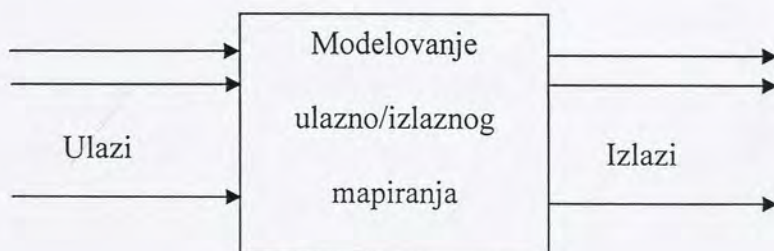
Prema dosadašnjim istraživanjima, izgleda da je samoorganizovanje osnovni mehanizam koji se koristi u takozvanom vizuelnom putu u mozgu [Foldiak, 1991], a koji omogućava adaptaciju na senzorne signale (u smislu reprezentacije i posledičnog izvođenja odgovarajućih aktivnosti). Sa inženjerske tačke gledišta, predstavlja jedan od obećavajućih i veoma moćnih pristupa za neke praktične probleme u vezi sa procesiranjem podataka, kao što su *data mining* i otkrivanje znanja u velikim bazama podataka, ili otkrivanje novih načina interakcije između čoveka i računara, a koji omogućavaju da se softver prilagodi potrebama i navikama individualnog korisnika na osnovu praćenja njegovog ponašanja.

Ovde ćemo se isključivo baviti samoorganizujućim mrežama koje se primenjuju u oblasti analize signala i podataka. Naravno, biće reči samo o glavnim principima i kategorijama samoorganizujućih mreža.

2.4 Razlike u primeni samoorganizujućih neuralnih mreža u odnosu na neuralne mreže koje uče pod nadzorom

Startnu tačku za učenje u veštačkim neuralnim mrežama predstavljaju skupovi numeričkih vektora podataka za učenja, koji su tipično velikih dimenzija. Mreže koje uče na bazi skupa podataka za obučavanje kao i skupa željenih ostvarenih vrednosti se nazivaju mreže koje uče pod nadzorom. Tipični predstavnici te vrste mreža su višeslojni perceptron i *radial basis* mreže. Svrha učenja je formiranje nelinearnog mapiranja između ulaza i izlaza korišćenjem baze podataka za učenje (i za ulaz i za izlaz). Ovakve mreže se primenjuju u oblastima kao što su prepoznavanje slike, teksta ili govora, industrijska dijagnostika, monitorisanje stanja, modelovanje kompleksnih *black box* kontrolnih sistema kao i analiza signala. Ulazi su obično vektori koji su dobijeni odabiranjem karakterističnih signala, dok izlazi predstavljaju ili klase signala koje su kodirane kao binarni vektori, ili neka vrsta željenog odziva koji se može predstaviti numeričkim vektorima. Tipično, učenje je kompletno bazirano na skupovima podataka za učenje, bez ikakve dodatne informacije o tome koja je poželjna nelinearna funkcija koja služi za klasifikaciju ili aproksimaciju. *Black box* modelovanje je prikazano na Slici 2.4.

U mnogim realnim problemima kao što su: analiza slike, *data mining*, istraživačka analiza podataka ili modelovanje velikih kompleksnih sistema, dimenzije ulaza sistema su reda stotina i hiljada, i funkcija koja treba da se aproksimira je veoma nelinearna i kompleksna. U tom slučaju se od mreže zahteva da ima veliki broj parametara kako bi se omogućila adekvatna aproksimacija ili generalizacija ulaznog domena. To dalje znači da veličina skupa



Slika 2.4 *Black box* modelovanje

za obučavanje mora rasti proporcionalno sa brojem slobodnih parametara. To znači, takođe, da je, pored velike količine podataka za obučavanje mreže, neophodno i veoma veliko vreme za realizaciju tog obučavanja. U tom slučaju obučavanje postaje veoma skupo, a možda i

nemoguće. To ujedno i predstavlja glavni problem u primeni neuralnih mreža koje uče pod nadzorom za praktične probleme analize podataka.

Problem bi se mogao ublažiti ako bi na neki način bilo moguće smanjiti kompleksnost sistema. U svakom realnom procesu analize podataka, podaci nisu potpuno slučajni već su generisani pomoću nekakvih fizičkih procesa ili su posledice sistema koji imaju ograničenu složenost [Hinton, 1987], [Sejnowski, 1987]. U tom slučaju je potrebno naći manji broj latentnih promenljivih koje dobro karakterišu proces i taj zadatak se može ostvariti upotrebom samoorganizujućih neuralnih mreža.

Još jedna velika razlika između samoorganizujućih mreža i onih koje uče pod nadzorom je mogućnost njihove primene u modelovanju realnog nervnog sistema. Veoma je teško zamisliti odakle dolaze signali za nadzor, na primer, pri obradi podataka signala koji dolaze sa senzora (čula). Takođe, ako prihvatimo hipotezu da je učenje bazirano na modifikaciji sinapsi, problem je kako lokalno implementirati zakone učenja koji su karakteristični za mreže koje uče pod nadzorom, kao što je metod «prostiranja unazad» (backpropagation). Prema trenutnim fiziološkim i anatomskim saznanjima realne neuralne mreže se mogu sa mnogo većom verovatnoćom modelovati samoorganizujućim neuralnim mrežama.

2.5 Osnovne kategorije samoorganizujućih neuralnih mreža

Samoorganizujuće neuralne mreže se, na osnovu algoritama za učenje koji se u njima primenjuju, mogu podeliti u dve osnovne grupe: jednu grupu čine zakoni učenja koji predstavljaju ekstenzije linearnih statističkih metoda i koje u suštini predstavljaju kodiranje u potprostor niže dimenzije, dok drugu grupu čine metode koje se bave vektorskim kodiranjem ili grupisanjem i bazirane su na kompetitivnom učenju.

2.5.1 Samoorganizujuće neuralne mreže bazirane na standardnim statističkim metodama

U ovu klasu algoritama spadaju metode koje se baziraju na standardnim statističkim metodama kao što su analiza glavnih komponenta (PCA) i faktor analiza (FA), koje rezultuju redukovanim podskupom linearnih kombinacija originalnih ulaznih varijabli. Mnoge metode za PCA su bazirane na Ojinom zakonu učenja [Oja, 1982], koji u suštini predstavlja modifikaciju Hebovog zakona učenja [Hebb, 1949]. U ovu klasu spadaju i tehnike koje se bave otkrivanjem nezavisnih komponentata, koje maksimalno redukuju redundansu među latentnim varijablama. Generalno taj način obrade signala je poznat kao obrada signala «na slepo» (blind signal processing - BSP). U tu vrstu spadaju analiza nezavisnih komponentata (independent component analysis - ICA) i/ili razdvajanje signala «na slepo» (blind signal separation - BSS). BSS i ICA su tehnike koje omogućavaju izdvajanje nezavisnih latentnih varijabli iz paralelnih signala kao što su govorni signal, elektromagnetska merenja na mozgu ili nekakve sekvence iz oblasti finansija, pri čemu se smatra da su ulazni signali linearne kombinacije nezavisnih latentnih varijabli.

Ovde će biti predstavljen Hebov zakon učenja, Ojin zakon učenja kao i osnovne ideje vezane za ICA i BSS. PCA neće biti analizirana u ovom poglavlju jer će biti predmet detaljnije analize u narednom poglavlju.

2.5.1.1 Hebov zakon učenja

Ovde ćemo analizirati jednoslojnu mrežu sa jednim izlaznim neuronom, kao primer. Linearni model neurona sa K ulaznih signala se može opisati jednačinom (2.1). Kao što je već rečeno neuron uči promenom vrednosti komponentata sinaptičkog vektora \mathbf{w} . U samoorganizujućim mrežama taj zakon učenja je baziran na sledećem principu koji je definisao Heb [Hebb, 1949]:

Kada jedan akson ćelije A često ili stalno učestvuje u eksitaciji ćelije B, tada se dešava proces rasta ili metabolička promena u jednoj ili obe ćelije tako da se efikasnost ćelije A u eksitaciji ćelije B uvećava.

Ova ideja je atraktivna iz razloga što je veoma jednostavna (bazirana je na korelaciji ulaznog i izlaznog signala) i što je bazirana na lokalnom principu (informacije koje su potrebne za modifikaciju sinaptičke vrednosti su prisutne na krajevima sinapse i nije neophodno postojanje dodatnih mehanizama koji bi ih obezbeđivali). Iako je Hebov zakon učenja 1949. godine bio samo hipoteza, do današnjeg dana postoji veliki broj neurofizioloških rezultata koji idu u prilog toj tezi [Brown, 1990]. U najjednostavnijoj matematičkoj interpretaciji ovaj zakon učenja se može pisati kao

$$\Delta w_k(i) = \gamma x_k(i)y(i) \quad (2.2)$$

gde Δw_k predstavlja modifikaciju vrednosti sinapse koja spaja k -ti ulazni signal i izlazni neuron, i gde je γ faktor učenja, a i predstavlja indeks koji označava diskretne trenutke u vremenu. Zakon učenja opisan jednačinom (2.2) je poznat kao Hebov zakon učenja. Za jednu izlaznu jedinicu on se može interpretirati kao rezultat želje za maksimizacijom varijanse izlaznog signala. Ako usvojimo da je ulazni signal sa srednjom vrednošću nula imamo

$$E(y^2) = E\left[\left(\mathbf{w}^T \mathbf{x}\right)^2\right] = \mathbf{w}^T \mathbf{C} \mathbf{w} \quad (2.3)$$

gde je \mathbf{C} kovarijansna matrica ulaznog signala, a E označava matematičko očekivanje. Jednostavno je uočiti da formiranje uzlaznog gradijentnog metoda za maksimizaciju varijanse izlaznog signala (uz usvajanje dovoljno malih vremenskih koraka) vodi direktno do Hebovog zakona učenja. Ako se ovakav zakon primeni direktno na linearnoj mreži koja je prikazana na Slici 2.2, jasno je da će vrednosti sinaptičkog vektora težiti beskonačnosti i imati pravac dominantnog sopstvenog vektora matrice \mathbf{C} [Krasulina, 1970].

Veliki broj radova se do danas bavio stabilizacijom (ograničavanjem vrednosti) osnovnog Hebovog zakona učenja dodavanjem određenih stabilizacionih članova. Jedan od najpoznatijih zakona je Ojin zakon učenja [Oja, 1982], koji ćemo predstaviti u sledećem paragrafu.

2.5.1.2 Ojin zakon učenja

I ovde ćemo se baviti jednoslojnom neuralnom mrežom koja na izlazu ima samo jedan neuron, a koja primenjuje jedan od najpoznatijih zakona učenja koji je predložen 1982. godine i naziva se Ojin zakon učenja, prema autoru. Postavlja se sledeće pitanje: Zašto je interesantno analizirati takvu mrežu? Prvo ćemo pokušati da damo odgovor na to pitanje.

Iako se korteks ljudskog mozga deli na veliki broj funkcionalno specijalizovanih oblasti, njegova anatomna struktura je iznenađujuće uniformna [Hubbel, 1974], [Rockel, 1980]. Takođe, činjenica da se neokorteks razvija veoma brzo za vreme filogene faze razvoja ukazuje na to da se on relativno lako replicira i da dalja ekspanzija zahteva mali broj dodatnih genetičkih instrukcija. Tipovi ćelija i osnovni tipovi lokalnih veza među njima su veoma slični u različitim regionima korteksa. Razlike su mnogo češće kvantitativne nego kvalitativne i povezane su sa tipom funkcije koju region obavlja. Na primer, slojevi koji obrađuju senzorne signale imaju deblje ulazne slojeve, dok najveći broj izlaza potiče od motornih oblasti. Korteks pokazuje uniformnost ne samo u anatomiji već i u mnogim aspektima svog razvoja. Veštačkim preusmeravanjem projekcije signala retine kod fereta (vrsta glodara), auditorni korteks može razviti tipove ćelija koje su karakteristične za video korteks, kao što su orijentaciono i direkciono selektivne ćelije [Sur, 1988]. Rezultati koji su dobijeni kod hrčaka kod kojih su signali retine projektovani u somatosenzorni korteks pokazali su da ćelije somatosenzornog korteksa daju odziv koji je karakterističan za oblast 17 primarnog vizuelnog korteksa normalne životinje, sa procentom ćelija koje razvijaju orijentacionu i direkcionu selektivnost, veoma sličnim kao u oblasti 17 [Metin, 1989].

Čini se da glavna razlika između pojedinih oblasti korteksa leži u tome što primaju ulaze iz, i prosleđuju izlaze u različite oblasti mozga. Postoji mnogo dokaza koji bi upućivali na to da sve ćelije korteksa ili koriste isti algoritam pri radu [Gilbert, 1988] ili u najgorem slučaju koriste iste radne principe. Na primer, potpuno isti mehanizmi mogu biti korišćeni da izazovu direkcionu selektivnost u vizuelnom i somatosenzornom korteksu.

Ako sve ćelije korteksa rade na istom principu, koji je to princip? Pošto taj princip još uvek sa sigurnošću nije poznat postoji prostor za pretpostavke (špekulacije) koje će biti potvrđene ili odbačene kada za to bude postojalo dovoljno saznanja. Znači, proučavanje osnovnog mehanizma rada jednog neurona može potencijalno biti od velikog značaja.

Ojin zakon učenja predstavlja, u suštini, Hebov zakon učenja koji je stabilizovan dodatnim članom. Stabilizacija je obezbeđena na bazi lokalno dostupnih informacija. On u suštini

predstavlja lokalnu primenu ideje normalizacije sinaptičkog vektora koji je već bio predlagan u literaturi i pre Oje [Amari, 1978; Kohonen, 1982]. Polazna tačka Ojinog zakona učenja je sledeća formula [Amari, 1978; Kohonen, 1982; Oja, 1982]:

$$w_k(i+1) = \frac{w_k(i) + \gamma x_k(i)y(i)}{\sqrt{\sum_{j=1}^N (w_j(i) + \gamma x_j(i)y(i))}} \quad (2.4)$$

koja, pod pretpostavkom da je γ dovoljno malo, može biti razvijena u Tejlorov red po γ , što daje

$$w_k(i+1) = w_k(i) + \gamma y(i)(x_k(i) - y(i)w_k(i)) + O(\gamma^2), \quad (2.5)$$

gde $O(\gamma^2)$ predstavlja kvadratni i više članove reda po γ . Ako usvojimo da je γ dovoljno malo, dobijamo sledeći zakon učenja

$$w_k(i+1) = w_k(i) + \gamma y(i)(x_k(i) - y(i)w_k(i)), \quad (2.6)$$

koji je poznat kao Ojin zakon učenja. Pod određenim pretpostavkama (o ulaznim signalima, faktoru učenja γ i td.), pokazano je [Oja, 1982; Oja, 1985] da sinaptički vektor teži dominantnom sopstvenom vektoru matrice \mathbf{C} pod pretpostvkom da nije normalan na njega u početnom trenutku, što je tačno sa verovatnoćom 1.

2.5.1.3 Obrada signala «na slepo» (BSP)

Obrada signala «na slepo» (Blind signal processing - BSP) je pristup analizi signala koji se generalno može formulisati na sledeći način:

Pretpostavimo da prikupljamo signale sa senzora koje ćemo označiti sa $\mathbf{x}(t) = (x_1(t), x_2(t), \dots, x_K(t))^T$ iz MIMO nelinearnog dinamičkog sistema. Cilj je pronalaženje inverznog sistema, koji ćemo nazvati rekonstrukcioni sistem, neuralna mreža ili inverzni adaptivni sistem, ako on postoji i ako je stabilan, kako bi se odredili primarni izvori signala $\mathbf{s}(t) = (s_1(t), s_2(t), \dots, s_N(t))^T$. Ta estimacija se obavlja na osnovu izlaznih signala $\mathbf{y}(t) = (y_1(t), y_2(t), \dots, y_N(t))^T$ i senzorskih signala kao i nekog prethodnog znanja o sistemu koji vrši mešanje primarnih

signala. U principu, poželjno je da inverzni sistem bude adaptivan u tom smislu da ima mogućnost praćenja nestacionarnog okruženja. Umesto estimiranja izvora signala direktno, često je mnogo jednostavnije prvo identifikovati dinamički sistem za mešanje i filtriranje (pogotovu u slučajevima kada inverzni sistem ne postoji ili ako je broj opservacija manji od broja izvora signala) i onda estimirati izvore signala implicitno korišćenjem nekih *a priori* informacija o sistemu i primenom pogodnih optimizacionih procedura.

U mnogim slučajevima, izvorni signali su istovremeno linearno filtrirani i izmešani. Cilj BSP je procesiranje opservacija na takav način da adaptivni sistem ekstrahuje originalne izvore. Problem estimacije i razdvajanja originalnih izvora talasnih oblika može ukratko biti izražen u vidu brojnih međusobno povezanih problema kao što su analiza nezavisnih komponenata (ICA), razdvajanje signala «na slepo» (BSS), ekstrakcija signala «na slepo» (BSE) ili višekanalna dekonvolucija signala «na slepo» (MBD).

Uproščeno govoreći ovi problemi mogu biti predstavljeni kao problemi razdvajanja i estimiranja talasnih oblika originalnih izvora iz mreže senzora ili prijemnika, a bez poznavanja karakteristika predajnog kanala.

Generalno gledano BSS i ICA se mogu definisati na sledeći način:

BSS slučajnog vektora $\mathbf{x}(t) = (\mathbf{x}_1(t), \mathbf{x}_2(t), \dots, \mathbf{x}_K(t))^T$ se dobija pronalaženjem $K \times N$, maksimalnog ranga, linearne transformacije (matrice) \mathbf{W} takve da izlazni signalni vektor $\mathbf{y} = (y_1(t), y_2(t), \dots, y_N(t))^T$, definisan kao $\mathbf{y} = \mathbf{W}^T \mathbf{x}$, sadrži komponente koje su međusobno nezavisne koliko je to više moguće, pri čemu se nezavisnost utvrđuje na osnovu informatičko-teorijskog kriterijuma kao što je Kulbak-Lajblerova divergencija [Deco, 1996] ili nekog drugog kriterijuma kao što su nepopunjenost ili linearna prediktabilnost. Pri tome se smatra da izvorni signali imaju srednju vrednost nula.

U principu za ICA metod postoji nekoliko vrsta definicija koje zavise od tipa problema na koji se metod primenjuje. Ovde ćemo navesti dve osnovne definicije (postoje i druge koje ovde neće biti prikazane – vidi [Cichocki, 2002]).

Definicija 1 (Temporalna ICA)

ICA slučajnog vektora $\mathbf{x}(i) \in R^K$ se dobija pronalaženjem $K \times N$ separacione matrice \mathbf{W} ($N \leq K$) tako da izlazni signal $\mathbf{y}(i) = (y_1(i), y_2(i), \dots, y_N(i))$ definisan sa

$$\mathbf{y}(i) = \mathbf{W}^T \mathbf{x}(i) \quad (2.7)$$

sadrži estimirane izvorne komponente $\mathbf{s}(i) \in R^N$ koje su nezavisne koliko je to moguće, a što se procenjuje na osnovu informaciono-teorijske funkcije cene kao što je Kulbak-Lajblerova divergencija. Lako je uočiti da je ova definicija koja je identična definiciji BSS.

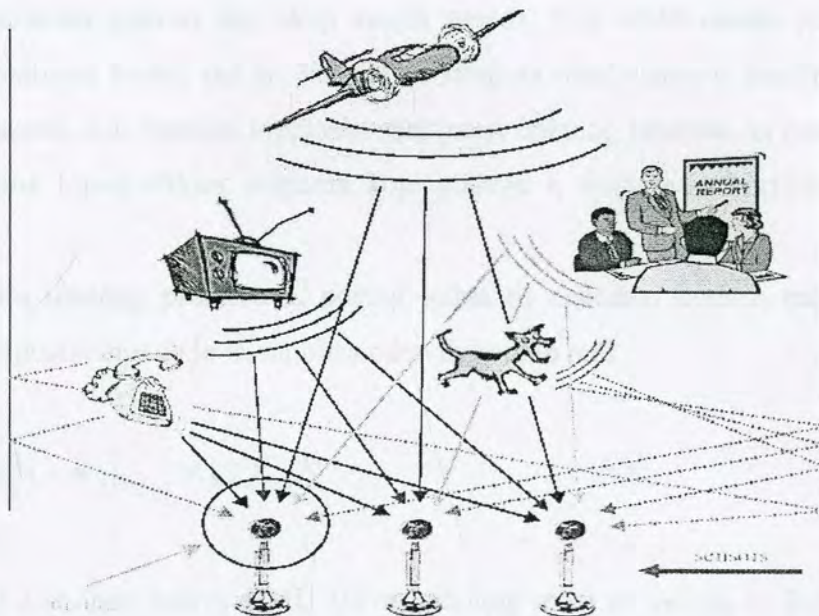
Definicija 2

Za slučajan vektor $\mathbf{x}(i)$ koji sadrži i šum, definisan sa

$$\mathbf{x}(i) = \mathbf{H}\mathbf{s}(i) + \mathbf{v}(i), \quad (2.8)$$

gde je \mathbf{H} ($N \times K$) matrica za mešanje, $\mathbf{s}(i) = (s_1(i), s_2(i), \dots, s_N(i))$ vektor izvora statistički nezavisnih signala i $\mathbf{v}(i) = (v_1(i), v_2(i), \dots, v_M(i))$ vektor nekorelisanih članova koji reprezentuju šum, ICA se dobija estimiranjem matrice za mešanje \mathbf{H} i nezavisnih komponentata $\mathbf{s}(i) = (s_1(i), s_2(i), \dots, s_N(i))$.

Pored formalnih definicija, uobičajeno je da se ICA problem predstavlja na osnovu jedne konkretne primene tog metoda na takozvani «cocktail party» problem. «Cocktail party» problem se može opisati kao sposobnost fokusiranja na govor pojedinca u prisustvu kakofonije drugih konverzacija kao i prisustvu šuma. Ovaj problem je veoma interesantan i tehnički veoma složen za rešavanje. Može se opisati i kao problem izdvajanja originalnih signala iz mešavine signala koji se detektuju matricom mikrofona (vidi Sliku 2.5). Poznato je da ljudski mozak rešava ovaj problem veoma uspešno.



Slika 2.5 «Cocktail party» problem - ilustracija

2.5.2 Samoorganizujuće neuralne mreže i klasterovanje podataka

Ova klasa algoritama se najčešće povezuje sa grupisanjem podataka. Tipične aplikacije u kojima se koriste ovakve metode su *data mininig* i profilisanje podataka iz veoma velikih baza. Najpoznatije metode u ovoj klasi su Kohonenov [Kohonen, 1995] metod, tzv. metod samoorganizujućih mapa (SOM), a koji je rezultat njegovog istraživanja u oblasti prepoznavanja oblika, Grosbergovi INSTAR i njegovi derivati, OUTSTAR i njegovi derivati i ART1, ART2 i ART3 topologije. Kao što je već ranije rečeno ovu klasu učenja nazivamo još i algoritmi za vektorsko kodiranje. Kod vektorskog kodiranja zadatak je da se prostor ulaznog signala koji je obično visoke ili veoma visoke dimenzije predstavi fiksnim brojem vektora koji se nazivaju kodne reči. Ulazni podaci su obično dati kao skup za obučavanje $\mathbf{x}(1)$, $\mathbf{x}(2)$, ..., $\mathbf{x}(T)$. Na primer ulazi mogu biti digitalne slike u nijansama sivog, merenja sa mašine ili industrijskog procesa, finansijski podaci koji opisuju kompaniju ili korisnika ili deo teksta koji je reprezentovan histogramom reči koje sadrži. SOM algoritam se primenjuje veoma često na probleme čija je dimenzija i do 50000 a pri tome skup za obučavanje sadrži i do 7000000 primera [Oja, 1992a][Kohonen, 1995]. Ovde ćemo detaljnije opisati SOM algoritam.

Pored grupisanja podataka koje je osnovni zadatak ovog tipa algoritama SOM mreža sadrži još jednu dodatnu osobinu: neuroni su organizovani u 1-, 2- i multidimenzione *lattice* (rešetke) tako da svaki neuron ima skup svojih suseda. Cilj SOM mreže je ne samo da pronade reprezentativnu kodnu reč za dati ulazni skup za obučavanje u smislu minimalnog rastojanja, već, takođe, i da formira topološko mapiranje ulaznog prostora na rešetku neurona. Ideja je inspirisana topografskim mapama koje postoje u senzornim kortičkim oblastima mozga.

Za svaku tačku ulaznog prostora \mathbf{x} , postoji jedna ili nekoliko kodnih reči koje su joj najbliže. Ako pretpostavimo da je \mathbf{w}_i najbliža od svih kodnih reči

$$\|\mathbf{x} - \mathbf{w}_i\| = \min_j \|\mathbf{x} - \mathbf{w}_j\|, \quad j = 1, 2, K, N \quad (2.9)$$

jedinica (neuron) i se tada naziva BMU (*best matching unit*) za vektor \mathbf{x} . Zapazimo da za fiksni vektor \mathbf{x} jednačina definiše indeks $i = i(\mathbf{x})$ BMU, dok za fiksno i , jednačina definiše

skup tačaka \mathbf{x} koje su mapirane tim indeksom. To znači da je gornjom relacijom vektor \mathbf{x} mapiran na diskretan skup indeksa i .

Pod topološkim mapiranjem se podrazumeva sledeća osobina: ako se dati vektor \mathbf{x} mapira jedinicom i , tada se sve tačke iz okruženja \mathbf{x} mapiraju ili istim indeksom i ili nekim od suseda i u rešeci. Pošto, u striktno matematičkom smislu, ne postoji topološko mapiranje između dva prostora različitih dimenzija, dvodimenziona neuralna mreža može samo lokalno pratiti dve dimenzije višedimenzionog prostora ulaznih signala.

SOM mreža je prikazana slici 2.6. Uloga WTA (pobednik uzima sve – winner takes all) sloja je da poredi sve izlaze iz sloja za mapiranje (odnosno rastojanje $\mathbf{x} - \mathbf{w}_i$) i da generiše indeks BMU.

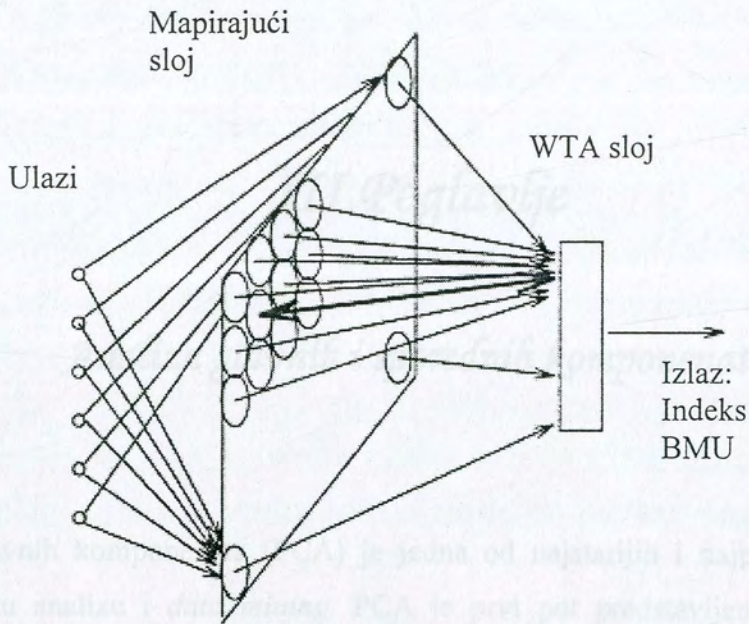
Svaki neuron u mapirajućem sloju prima iste ulaze. BMU se pronalazi pomoću WTA sloja. Nakon toga u fazi učenja, samo BMU i njegovi najbliži susedi menjaju svoje sinaptičke veze. BMU je prikazan debelom strelicom.

Kohonenov algoritam za samoorganizovanje kodnih vektora je dat sa

1. Izaberi inicijalnu vrednost težinskih vektora \mathbf{w}_i .
2. Ponovi korake 3 i 4 sve dok algoritam ne konvergira.
3. Izaberi iz skupa za obučavanje vektor \mathbf{x} i nađi BMU $i=i(\mathbf{x})$ u skladu sa relacijom (2.9)
4. Modifikuj težinske vektore svih jedinica sledećom formulom

$$\mathbf{w}_j(k+1) = \mathbf{w}_j(k) + \gamma h_r(\mathbf{x} - \mathbf{w}_j) \quad (2.10)$$

gde je γ faktor učenja a h_r funkcija koja zavisi od rastojanja $r = |i-j|$ jedinice i i jedinica j duž rešetke.



Slika 2.6 SOM mreža.

Postoji nekoliko varijanti algoritma u zavisnosti od toga kako se biraju inicijalne vrednosti i kako se bira h_r . Te metode i njihova matematička analiza su prikazani u [Kohonen, 1995; Ritter, 1992; Oja, 1999; VanHulle, 2000].

III Poglavlje

Analiza glavnih i sporednih komponentata

Analiza glavnih komponentata (PCA) je jedna od najstarijih i najpoznatijih metoda za multivarijabilnu analizu i *data mining*. PCA je prvi put predstavljena od strane Pirsona [Pearson, 1901] koji ju je koristio u biološkom kontekstu, i kasnije razvijana od strane Hotellinga [Hotelling, 1933] koji je obavljao istraživanja na području psihometrije. PCA je nezavisno razvijena od strane Karhunenena u kontekstu teorije verovatnoće i kasnije generalizovana od strane Leva [Loève, 1948]. Poslednjih godina, razvijeni su mnogobrojni efikasni adaptivni algoritmi za implementaciju PCA, analize sporednih komponentata (MCA), kao i njihovih ekstenzija [Amari, 1978; Oja, 1982; Oja, 1992a; Oja, 1997; Bannour, 1989; Cichocki, 2001]. Glavni cilj ovog poglavlja je definicija PCA i analize sporednih komponentata (MCA) kao i prikaz osnovnih algoritama za njihovu realizaciju u neuralnim mrežama. Ovde će isključivo biti reči o sekvencijalnim algoritmima za ekstrakciju glavnih i sporednih komponentata.

3.1 Analiza glavnih komponentata (PCA)

3.1.1 Određivanje sopstvenih vrednosti

Generalno govoreći, namena PCA je određivanje relativno malog broja glavnih komponentata (međusobno nekorelisanih) koje reprezentuju skup slučajnih signala sa srednjom vrednošću nula na ulazu sistema i koje pri tome zadržavaju maksimalnu moguću količinu informacija koja se sadrži u originalnim signalima.

Među ciljevima PCA su:

1. smanjenje dimenzije signala,
2. otkrivanje linearne zavisnosti među komponentama signala,
3. određivanje karakterističnih osobina signala,
4. vizuelizacija višedimenzionih podataka,
5. identifikacija grupa objekata i tako dalje.

Takođe, PCA je korisna kao pretprocesirajući korak za analizu nezavisnih komponentata (engleski: *Independent Component Analysis* – ICA).

PCA je široko korišćena i proučavana u oblasti prepoznavanja oblika i procesiranja signala. To je veoma važan alat u mnogim inženjerskim i naučnim disciplinama kao što su, na primer, kompresija podataka, otkrivanje karakterističnih osobina signala, filtriranje šuma, restauriranje signala, klasifikacija [Diamantaras, 1996]. PCA se koristi u *data mining* kao tehnika za redukciju podataka. U oblasti obrade slike i kompjuterskog vida (*computer vision*) PCA reprezentacije su korišćene za rešavanje problema kao što su: prepoznavanje lica i objekata, praćenje pokretnog objekta, detekcija, parametrizacija oblika, pojava objekta ili kretanje objekta [Turk, 1991; Kung, 1991]. Takođe, pod nazivom EOF ova tehnika se koristi i u meteorološkim istraživanjima, kod prognoze vremena [Lorenz, 1956; Wallace, 1972; Reljin, 2002a; Reljin, 2002b].

Često su glavne komponente (pravci u kojima ulazni podaci imaju najveće varijanse) smatrani za važne dok su komponente koje su vezane za sporedne komponente smatrane nevažnim ili povezivane sa šumom. U poslednje vreme je pokazano, da MCA (minor component analysis) predstavlja veoma koristan alat za procesiranje sekvenci podataka. MCA je korisna na primer kod: praćenja pokretnih meta [Klemm, 1987], procenjivanja pozicije emitera signala [Mathew, 1994] kao i procene parametara signala [Schmidt, 1986], analize i razumevanja bioloških podataka [Wiscott, 1998], potiskivanja šuma ili aproksimacije funkcija pomoću fitovanja [Xu, 1992].

Uopšteno govoreći, PCA je povezana i motivisana sa dva sledeća problema:

1. Za date slučajne vektore $\mathbf{x}(i) \in \mathcal{R}^K$, koji imaju konačne druge momente i srednju vrednost jednaku nuli, potrebno je naći linearni potprostor manje dimenzije N ($N < K$) za koji je rastojanje \mathbf{x} od tog potprostora minimalno. Ovaj problem se pojavljuje kod kompresije podataka gde je zadatak reprezentovanje svih podataka sa redukovanim brojem parametara pri čemu je potrebno obezbediti minimalnu distorziju usled projekcije.

2. Za dati slučajan vektor $\mathbf{x}(i) \in \mathcal{R}^K$, nađi N -dimenzioni linearni potprostor koji sadrži većinu varijanse podataka \mathbf{x} . Ovaj problem je vezan za ekstrakciju karakterističnih osobina signala, gde je cilj redukcija dimenzije podataka pri čemu je potrebno zadržati maksimalnu količinu informacija koja se sadrži u ulaznim podacima.

Pokazalo se da oba problema imaju isto optimalno rešenje (u smislu najmanje kvadratne greške) koje se bazira na statistici drugog reda, posebno, na sopstvenim vektorima kovarijanske matrice ulaznog signala \mathbf{x} i to je rešenje u suštini ekvivalentno Karhunen-Levovoj transformaciji koja se koristi kod analize slike i procesiranja signala. Drugim rečima, PCA je tehnika za izračunavanje sopstvenih vrednosti i sopstvenih vektora estimirane kovarijanske matrice

$$\mathbf{C} = E\{\mathbf{x}(i)\mathbf{x}(i)^T\} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T \in \mathcal{R}^{K \times K} \quad (3.1)$$

gde je $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_K\}$ dijagonalna matrica koja sadrži K sopstvenih vrednosti i $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_K] \in \mathcal{R}^{K \times K}$ je odgovarajuća ortonormalna ili unitarna matrica koja se sastoji od sopstvenih vektora jedinične dužine.

Karhunen-Levova transformacija određuje linearnu transformaciju ulaznog vektora \mathbf{x} kao

$$\mathbf{y}_p = \mathbf{V}_S^T \mathbf{x} \quad (3.2)$$

gde je $\mathbf{x} = [x_1(i), x_2(i), \dots, x_K(i)]^T$ ulazni vektor sa srednjom vrednošću nula, $\mathbf{y}_p = (y_1(i), y_2(i), \dots, y_N(i))^T$ izlazni vektor koji se naziva vektorom glavnih komponenti i $\mathbf{V}_S = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N] \in \mathcal{R}^{K \times N}$ predstavlja sopstvene vektore signalnog potprostora, gde su $\mathbf{v}_k = [v_{k1}, v_{k2}, \dots, v_{kK}]^T$ ortonormalni vektori. Vektori \mathbf{v}_k ($k = 1, 2, \dots, N$) su sopstveni vektori kovarijanske matrice, dok su varijanse glavnih komponentata y_k odgovarajuće sopstvene vrednosti. Sa druge strane $K-N$ sporednih komponentata je dato sa

$$\mathbf{y}_M = \mathbf{V}_N^T \mathbf{x}, \quad (3.3)$$

gde $\mathbf{V}_N = [\mathbf{v}_K, \mathbf{v}_{K-1}, \dots, \mathbf{v}_{K-N+1}]$ sadrži $(K-N)$ sopstvenih vektora povezanih sa najmanjim sopstvenim vrednostima.

Tako, osnovni problem koji pokušavamo da rešimo predstavlja standardni problem izračunavanja sopstvenih vektora-vrednosti koji se može formulirati sledećim skupom jednačina

$$\mathbf{C}\mathbf{v}_k = \lambda_k \mathbf{v}_k \quad (k = 1, 2, \dots, N) \quad (3.4)$$

gde su \mathbf{v}_k sopstveni vektori, a λ_k odgovarajuće sopstvene vrednosti, $\mathbf{C} = E(\mathbf{x}\mathbf{x}^T)$ kovarijansna matrica ulaznog signala \mathbf{x} čija je srednja vrednost nula i E je operator očekivanja. Zapažimo da se jednačina (3.4) može pisati i u formi $\mathbf{V}^T \mathbf{C} \mathbf{V} = \Lambda$, gde je Λ dijagonalna matrica sopstvenih vrednosti matrice \mathbf{C} .

U standardnim numeričkim pristupima određivanja glavnih komponenta, prvo se mora izračunati kovarijansna matrica \mathbf{C} i onda se pristupa određivanju sopstvenih vektora i odgovarajućih sopstvenih vrednosti pomoću nekog od poznatih algoritama. Međutim, u situacijama kada je ulazni vektor velikih dimenzija (na primer 1000 elemenata), kovarijansna matrica postaje veoma velikih dimenzija (10^6 ulaza) i tada izračunavanje sopstvenih vektora može postati veoma teško.

Upotrebom neuralnih mreža sa odgovarajućim adaptivnim zakonima učenja omogućava se izračunavanje sopstvenih vektora i, njima pridruženim, sopstvenih vrednosti na osnovu ulaznog signala $\mathbf{x}(i)$ bez potrebe za izračunavanjem ili estimiranjem veoma velike matrice \mathbf{C} . Takav pristup je posebno koristan u slučajevima kada je ulazni signal nestacionaran, na primer u slučajevima kada se zahteva praćenje sporih promena međuzavisnosti ulaznih podataka (signala) ili pri osvežavanju sopstvenih vektora novim odbircima. Izračunavanje kovarijansne matrice samo za sebe je računski veoma skupo. Dalje, direktna dijagonalizacija matrice ili *eigenvalue decomposition* mogu takođe biti veoma računski skupe jer je broj operacija koji je u tom slučaju potrebno izvesti $O(K^3)$. Većina adaptivnih algoritama koje ćemo sada predstaviti ne zahteva izračunavanje kovarijansne matrice i ima malu složenost.

3.1.2 Procena kovarijansne matrice

U praksi, tačna vrednost kovarijansne matrice \mathbf{C} nije nikada dostupna. Jedino što se može uraditi je procena \mathbf{C}_1 matrice \mathbf{C} koja se izračunava na osnovu konačnog broja odbiraka kao

$$\mathbf{C}_1 = \frac{1}{M} \sum_{i=1}^M \mathbf{x}(i)\mathbf{x}(i)^T. \quad (3.5)$$

U ovom slučaju pretpostavka je da se kovarijansna matrica ne menja (ili menja veoma sporo) na bloku ulaznih podataka dužine M . Alternativno, može se koristiti pristup koji izračunava *moving average* za *online* estimaciju kovarijansne matrice, a koji se izračunava kao

$$\mathbf{C}_1^{(i)} = (1 - \eta_0)\mathbf{C}_1^{(i-1)} + \eta_0\mathbf{x}(i)\mathbf{x}^T(i), \quad (3.6)$$

gde je $\eta_0 > 0$ faktor učenja (a faktor $1 - \eta_0$ faktor zaboravljanja) i koji se bira na osnovu stacionarnosti signala i tipično iznosi 0.01 do 0.1.

Alternativno, u aplikacijama u realnom vremenu, kovarijansna matrica se može rekurzivno izračunavati na osnovu sledeće formule

$$\begin{aligned} \mathbf{C}_M &= \frac{1}{M} \sum_{l=i-M+1}^i \mathbf{x}(l)\mathbf{x}^T(l) = \frac{1}{M} \left[\sum_{l=i-M+1}^{i-1} \mathbf{x}(l)\mathbf{x}^T(l) + \mathbf{x}(i)\mathbf{x}^T(i) \right] \\ &= \frac{M-1}{M} \mathbf{C}_{M-1} + \frac{1}{M} \mathbf{x}(i)\mathbf{x}^T(i), \end{aligned} \quad (3.7)$$

gde \mathbf{C}_M označava estimiranu vrednost kovarijansne matrice u i -tom trenutku i gde je

$$\mathbf{C}_{M-1} = \frac{1}{M-1} \sum_{l=i-M+1}^{i-1} \mathbf{x}(l)\mathbf{x}^T(l).$$

Rekurzivna formula u mnogo opštijoj formi je data sa

$$\mathbf{C}_M = \alpha \mathbf{C}_{M-1} + \Delta \mathbf{C} \quad (3.8)$$

gde je α parametar u intervalu 0 do 1, a $\Delta \mathbf{C}$ predstavlja simetričnu matricu čiji je rang mnogo manji od ranga matrice \mathbf{C}_{M-1} . U slučaju kada se radi o stacionarnim signalima obično se bira $\alpha = (M-1)/M$ i $\Delta \mathbf{C} = (1/M)\mathbf{x}(i)\mathbf{x}^T(i)$ čiji je rang 1, gde je $\mathbf{x}(i)$ vrednost ulaznog vektora u trenutku i . Sa druge strane u nestacionarnom slučaju, koristi se $\Delta \mathbf{C}$ ranga 1, pri čemu je $\alpha \ll$

1, a ΔC je $\mathbf{x}(i)\mathbf{x}^T(i)$. Alternativno, u nestacionarnom slučaju, koristi se ΔC čiji je rang 2, a $\alpha=1$.

3.2 Potprostori signala i šuma – automatski izbor dimenzije PCA

Veoma važan problem koji se pojavljuje u mnogim oblastima primene PCA jeste određivanje dimenzije signala kao i potprostora šuma. Drugim rečima, jedan od glavnih problema kod primene PCA je izbor broja glavnih komponentata koje je potrebno zadržati [Minka, 2001]. Da bismo rešili taj problem obično se koristi fundamentalna osobina PCA: PCA projektuje ulazne podatke $\mathbf{x}(i)$ iz njihovog originalnog K -dimenzionog prostora u N -dimenzioni izlazni potprostor $\mathbf{y}(i)$ (tipično $N \ll K$), pri čemu se redukcija dimenzije obavlja na takav način da se zadrži maksimalno moguća količina informacija koja se sadrži u ulaznim podacima. Drugim rečima, glavne komponente $y_k(i)=\mathbf{w}_k^T\mathbf{x}(i)$ se procenjuju na taj način da se, i pored značajne redukcije dimenzionalnosti, zadrži maksimalna količina informacija. U tom slučaju se ulazni podaci \mathbf{x} mogu rekonstruisati na osnovu izlaznih podataka \mathbf{y} pomoću transformacije $\mathbf{x}_p=\mathbf{W}\mathbf{y}$, tako da se minimizira odgovarajuća funkcija cene. Najčešće korišćeni kriterijum je minimizacija srednjeg kvadratnog odstupanja $\|\mathbf{x}-\mathbf{W}\mathbf{W}^T\mathbf{x}\|^2$.

PCA nam omogućava da podelimo merene, senzorne signale: $\mathbf{x}(i)=\mathbf{x}_s(i)+\mathbf{v}(i)$ na dva potprostora: signalni potprostor koji odgovara glavnim komponentama koje odgovaraju najvećim sopstvenim vrednostima i odgovarajućim sopstvenim vektorima i potprostor šuma koji odgovara sporednim komponentama koje su povezane sa najmanjim sopstvenim vrednostima. Signalni potprostor se tada može smatrati potprostorom koji je «bešuman». Jedan od problema koji se pojavljuje kod ovakvog pristupa problemu je kako korektno proceniti prag koji deli sopstvene vrednosti u dva podskupa, posebno u slučajevima kada je poznato da je nivo šuma visok.

Pretpostavimo da je model vektora $\mathbf{x}(i) \in \mathfrak{R}^K$ dat sa

$$\mathbf{x}(i) = \mathbf{H}\mathbf{s}(i) + \mathbf{v}(i) \quad (3.9)$$

gde je $\mathbf{H} \in \mathfrak{R}^{(K \times N)}$ matrica za mešanje ($K > N$) maksimalnog ranga, $\mathbf{s}(i) \in \mathfrak{R}^N$ vektor izvora koji zadovoljavaju Gausovu raspodelu sa srednjom vrednošću nula i čija je kovarijansna matrica $\mathbf{R}_{ss}=\mathbf{E}(\mathbf{s}(i)\mathbf{s}^T(i))$ nesingularna i gde je $\mathbf{v}(i) \in \mathfrak{R}^K$ vektor šuma koji zadovoljava Gausovu

raspodelu sa srednjom vrednošću nula i čija je kovarijansna matrica data sa $\mathbf{R}_{vv} = \sigma_v^2 \mathbf{I}_N$, pri čemu su vektori \mathbf{s} i \mathbf{v} nekorelisani. Ovakav model se često naziva i probabilistička PCA i obično se koristi u kontekstu mašinskog učenja (*machine learning*). Takođe, ovakav model se može smatrati posebnom formom faktor analize (FA) u kojoj je šum izotropan. Razlika u odnosu na FA je u tome što je kod FA kovarijansna matrica šuma, generalno gledano, dijagonalna matrica.

Za model (3.9), pod gore navedenim uslovima, kovarijansna matrica $\mathbf{x}(i)$ se može pisati kao

$$\begin{aligned} \mathbf{C} &= E\{\mathbf{x}(i)\mathbf{x}^T(i)\} = \mathbf{H}\mathbf{R}_{ss}\mathbf{H}^T + \sigma_v^2\mathbf{I}_N \\ &= [\mathbf{V}_S, \mathbf{V}_{NI}] \begin{bmatrix} \Lambda_S & 0 \\ 0 & \Lambda_{NI} \end{bmatrix} [\mathbf{V}_S, \mathbf{V}_{NI}]^T \\ &= \mathbf{V}_S\Lambda_S\mathbf{V}_S^T + \mathbf{V}_{NI}\Lambda_{NI}\mathbf{V}_{NI}^T, \end{aligned} \quad (3.10)$$

gde je $\mathbf{H}\mathbf{R}_{ss}\mathbf{H}^T = \mathbf{V}_S\Lambda_S\mathbf{V}_S^T$ matrica ranga K , $\mathbf{V}_S \in \mathfrak{R}^{K \times N}$ i sadrži sopstvene vektore koji odgovaraju N dominantnih sopstvenih vrednosti, koje čine dijagonalu matrice $\Lambda_S = \text{diag}\{\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N\}$ u opadajućem redosledu. Slično, matrica $\mathbf{V}_{NI} \in \mathfrak{R}^{K \times (K-N)}$ sadrži $(K-N)$ sopstvenih vektora koji odgovaraju sopstvenim vrednostima šuma $\Lambda_{NI} = \text{diag}\{\lambda_{N+1} \geq \lambda_{N+2} \geq \dots \geq \lambda_K\} = \sigma_v^2 \mathbf{I}_{K-N}$. To znači, teorijski, da je $(K-N)$ najmanjih sopstvenih vrednosti matrice \mathbf{C} jednako σ_v^2 , tako da se može odrediti dimenzija signalnog potprostora na osnovu višestrukosti sopstvene vrednosti čija je vrednost najmanja, pod pretpostavkom da je varijansa šuma relativno mala i da je moguća perfektna procena vrednosti kovarijansne matrice. Međutim, u praksi, kovarijansna matrica se procenjuje na osnovu ograničenog broja uzoraka i najmanje sopstvene vrednosti su obično različite, tako da određivanje dimenzije signalnog potprostora nije jednostavan zadatak.

Umesto da se prag između sopstvenih vrednosti koje reprezentuju signal i onih koje reprezentuju šum postavlja na osnovu nekakve heurističke procedure, mogu se koristiti i informacioni teoretski kriterijumi, kao što su Akaikeov informacioni kriterijum (AIC), minimalna dužina zapisa (*Minimum Description Length* - MDL) i Bajesov informacioni kriterijum (BIC) [Karhunen, 1997; Wax, 1985; Minka, 2001]. Primena ovih kriterijuma zahteva izračunavanje verovatnoće podataka za sve moguće dimenzije. Primenom AIC kriterijuma bira se model koji minimizira sledeću funkciju cene [Liavas, 1999]

$$\text{AIC} = -2 \log(p(\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(M) | \Theta_K)) + 2N, \quad (3.11)$$

gde je $p(\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(M) | \Theta_K)$ parametrizovana funkcija gustine verovatnoće, Θ_K je maksimum *likelihood* estimator parametarskog vektora Θ , i N je broj slobodno podesivih parametara.

MDL kriterijum selektuje model koji minimizira

$$\text{MDL} = -\log(p(\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(M) | \Theta_K)) + \frac{1}{2} N \log M. \quad (3.12)$$

Ako usvojimo da je vektor ulaza $\{\mathbf{x}(i)\}_{i=1}^M$ u stvari vektor čija je srednja vrednost nula i ako zadovoljava Gausovu raspodelu, može se pokazati [Wax, 1985] da se dimenzija signalnog potprostora $N \in (1, 2, \dots, K)$ dobija za ono N za koje je

$$\text{AIC}(N) = -2M(K - N) \log \rho(N) + 2N(2K - N), \quad (3.13)$$

$$\text{MDL}(N) = -M(K - N) \log \rho(N) + 0.5N(2K - N) \log(M) \quad (3.14)$$

minimalno. Ovde je M broj vektora podataka $\mathbf{x}(i)$ koji se koriste za estimiranje kovarijanske matrice \mathbf{C} i ρ je dato sa

$$\rho(N) = \frac{(\lambda_{N+1} \lambda_{N+2} \wedge \lambda_K)^{\frac{1}{K-N}}}{\frac{1}{K-N} (\lambda_{N+1} + \lambda_{N+2} + \Lambda + \lambda_K)} \quad (3.15)$$

i predstavlja odnos geometrijske i aritmetičke sredine najmanjih $(K-N)$ sopstvenih vrednosti. Estimirana vrednost za N se može izabrati primenom bilo AIC, bilo MDL kriterijuma.

Nažalost, oba kriterijuma daju samo približnu procenu dimenzije signalnog potprostora i oba su veoma osetljiva na odnos signal/šum (SNR) kao i na broj raspoloživih uzoraka ulaznog signala [Liavas, 1999]. Drugi problem koji se pojavljuje kod primene AIC ili MDL kriterijuma je taj što se mora pretpostaviti da ulazni vektor $\mathbf{x}(i)$ zadovoljava Gausovu raspodelu [Wax, 1985]. To je urađeno iz razloga što je izborom Gausove raspodele omogućeno dobijanje izraza u zatvorenoj formi.

Nedavno je Minka [Minka, 2001] predložio veoma efikasan kriterijum koji se koristi za estimaciju dimenzije signalnog potprostora, a koji se bazira na Bajesovoj selekciji modela (MIBS). Estimacija izračunava integral na Stifelovoj mnogostrukosti koji se aproksimira Laplasovom metodom. Za primenu ovog metoda potrebno je izabrati N , $1 \leq N \leq K$ tako da se maksimizira izraz

$$\text{MIBS}(N) = p(X|N) \approx p_N \left(\prod_{j=1}^N \lambda_j \right)^{-M/2} \sigma_N^{-M(K-N)} |A_N|^{-1/2} (2\pi)^{(d_N+N)/2} M^{-M/2}, \quad (3.16)$$

gde je

$$p_N = 2^{-N} \prod_{i=1}^N \Gamma\left(\frac{K-i+1}{2}\right) \pi^{-(K-i+1)/2},$$

$$|A_N| = \prod_{i=1}^N \prod_{j=i+1}^K (\lambda k_j^{-1} - \lambda k_i^{-1}) (\lambda_i - \lambda_j) M,$$

$$\sigma_N^2 = \left(\sum_{j=N+1}^K \lambda_j \right) / (K-N), \quad d_N = KN - N(N+1)/2$$

i gde je λk_j identično sa λ_j osim za $j > n$ kada je $\lambda k_j = \sigma_n$. Kako bi se odredila stvarna dimenzija podataka \mathbf{X} , biramo vrednost N tako se maksimizira aproksimacija $p(X|N)$ [Minka, 2001].

Jedna aproksimacija MIBS dovodi do Bajesovog informacionog kriterijuma (BIC) koji zanemaruje sve članove koji ne zavise od M :

$$\text{BIC}(N) = \left(\prod_{j=1}^N \lambda_j \right)^{-\frac{M}{2}} \sigma_N^{-M(K-N)/2} M^{-(d_N+N)/2}. \quad (3.17)$$

Eksperimenti koje je obavljao Minka su pokazali da je kriterijum robusan u odnosu na distribuciju izvora i veoma konzistentan čak i u slučaju relativno malog broja ulaznih podataka.

Sledeća dva uslova mogu dovesti do korektno estimacije broja izvora. Prvo, dimenzija ulaznog podatka mora biti veća od broja izvora. Ako je broj izvora jednak broju senzora, svi kriterijumi bez izuzetka procenjuju da je broj izvora za jedan manji nego što jeste. Drugi

uslov je da mora postojati barem mala količina šuma. To, takođe, garantuje da su najmanje sopstvene vrednosti koje odgovaraju šumu, različite od nule.

3.3 Osnovne osobine PCA

Sada ćemo navesti neke osobine glavnih komponentata $y_k(i) = \mathbf{w}_k^T \mathbf{x}(i)$, bez dokaza jer se svi dokazi mogu jednostavno izvesti.

1. Faktor $y_1(i) = \mathbf{w}_1^T \mathbf{x}(i)$ je glavna komponenta $\mathbf{x}(i)$ ako je varijansa $y_1(i)$ maksimalna, pod uslovom da je norma (dužina) vektora \mathbf{w}_1 konstantna [Oja, 1992]. Tada vektor \mathbf{w}_1 maksimizira sledeći kriterijum

$$J_1(\mathbf{w}_1) = E(y_1^2) = E(\mathbf{w}_1^T \mathbf{C} \mathbf{w}_1), \quad (3.18)$$

Pod ograničenjem $\|\mathbf{w}_1\|_2 = 1$. Ovaj kriterijum se može proširiti na N glavnih komponentata (N je bilo koji ceo broj veći od 1 i manji od K) kao

$$J_1(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N) = E\left(\sum_{k=1}^N y_k^2\right) = E\left(\sum_{k=1}^N (\mathbf{w}_k^T \mathbf{x})^2\right) = \sum_{k=1}^N \mathbf{w}_k^T \mathbf{C} \mathbf{w}_k \quad (3.19)$$

pod ograničenjem $\mathbf{w}_i^T \mathbf{w}_j = \delta_{ij}$.

2. Glavne komponente imaju srednju vrednost nula,

$$E(y_i) = 0, \quad \forall i. \quad (3.20)$$

3. Različite glavne komponente su međusobno nekorelisane

$$E(y_i y_j) = \delta_{ij} \lambda_j, \quad (i, j = 1, 2, \dots, N). \quad (3.21)$$

4. Varijansa i -te glavne komponente je jednaka i -toj sopstvenoj vrednosti kovarijanske matrice \mathbf{C}

$$\text{var}(y_k) = \sigma_{y_k}^2 = E(y_k^2) = E\left(\left(\mathbf{w}_k^T \mathbf{x}\right)^2\right) = \mathbf{w}_k^T \mathbf{C} \mathbf{w}_k = \lambda_k. \quad (3.22)$$

5. Osobina najbolje aproksimacije

Pretpostavimo da važi:

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K. \quad (3.23)$$

Za srednju kvadratnu grešku aproksimacije

$$\mathbf{x} \mathbf{p} = \sum_{k=1}^N y_k \mathbf{w}_k = \sum_{k=1}^N \mathbf{w}_k \mathbf{w}_k^T \mathbf{x}, \quad N < K \quad (3.24)$$

imamo

$$E\left(\|\mathbf{x} - \mathbf{x} \mathbf{p}\|_2^2\right) = E\left(\left\|\sum_{k=N+1}^K y_k \mathbf{w}_k\right\|_2^2\right) = \sum_{k=N+1}^K E\left(|y_k|^2\right) = \sum_{k=N+1}^K \lambda_k. \quad (3.25)$$

Uzimajući u obzir da važi (3.23), očigledno je da aproksimacija sa sopstvenim vektorima $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N$ koji odgovaraju najvećim sopstvenim vrednostima, dovodi do minimalne srednje kvadratne greške, odnosno do najbolje aproksimacije.

3.4 Ekstrakcija glavnih komponenta korišćenjem principa optimalne kompresije-rekonstrukcije

Jedan od najjednostavnijih i intuitivno najrazumljivijih pristupa za adaptivno izračunavanje glavnih komponentata (PCA) je baziran na samonadzoru (samoasocijaciji ili principu replikacije) [Amari, 1978; Cichocki, 1993; Cichocki, 1994]. Upotrebom takvog pristupa prvo se vrši kompresija ulaznog vektora $\mathbf{x}(k)$ na jednu promenljivu $y_1(k) = \mathbf{w}_1^T \mathbf{x}(k)$. Sada ćemo pretpostaviti da želimo da izračunamo glavne komponente sekvencijalno upotrebom principa samonadzora i hijerarhijske arhitekture neuralne mreže [Cichocki, 1993; Cichocki, 1994; Cichocki, 1996].

Ako koristimo jednostavnu procesnu jedinicu koja je prikazana na slici 2.2, imamo

$$y_1(i) = \mathbf{w}_1^T \mathbf{x}(i) = \sum_{p=1}^K w_{1p} x_p(i), \quad (3.26)$$

koja izračunava dominantnu glavnu komponentu, pri čemu je $\lambda_1 = E(y_1^2(i))$. Striktno govoreći, $y_1(i)$ predstavlja prvu, odnosno dominantnu glavnu komponentu $\mathbf{x}(i)$, ako je varijansa $y_1(i)$ maksimalna pod ograničenjem da je glavni sopstveni vektor \mathbf{w}_1 jedinične dužine.

Vektor $\mathbf{w}_1 = (w_{11}, w_{12}, \dots, w_{1K})^T$ treba da bude određen tako da rekonstruisani vektor $\mathbf{x}_p(k) = \mathbf{w}_1 y_1(k)$ rekonstruiše ulazni vektor $\mathbf{x}(k)$ što je bolje moguće, a u skladu sa odgovarajućim optimizacionim kriterijumom. Generalno govoreći, funkcija cene može biti prikazana u sledećem obliku

$$J_1(\mathbf{w}_1) = E \left\{ \left\| \mathbf{x}(i) - \mathbf{w}_1 \mathbf{w}_1^T \mathbf{x}(i) \right\|_2^2 \right\} \cong \sum_{i=1}^K \mu^{K-i} \left\| \mathbf{x}(i) - \mathbf{w}_1 \mathbf{w}_1^T \mathbf{x}(i) \right\|_2^2, \quad (3.27)$$

gde je μ faktor zaboravljanja.

Sada ćemo razmotriti uprošćenu (u kontinualnom vremenu) verziju funkcije cene koja će biti napisana kao

$$J_{p1} = \frac{1}{2} \left\| \mathbf{e}_1(t) \right\|_2^2 = \sum_{p=1}^K e_{1p}^2(t), \quad (3.28)$$

gde je

$$\mathbf{e}_1(t) = \mathbf{x}(t) - \mathbf{w}_1 y_1(t), \quad y_1 = \mathbf{w}_1^T \mathbf{x}(t).$$

Formulacija funkcije cene koja se izračunava, u ovom slučaju $J_{p_1}(\mathbf{w}_1)$, je ključni korak u postupku, jer nam omogućava transformaciju minimizacionog problema u skup diferencijalnih ili diferencnih jednačina koje onda određuju adaptivni zakon učenja [Cichocki, 1993; Cichocki, 1994].

Minimizacija funkcije cene (3.28) u skladu sa standardnim opadajućim gradijentnim postupkom [Cichocki, 1993] po vektoru \mathbf{w}_1 , rezultuje sledećim diferencijalnim jednačinama

$$\frac{d\mathbf{w}_{1p}}{dt} = -\gamma_1 \frac{\partial J_1(\mathbf{w}_1)}{\partial \mathbf{w}_{1p}} = \gamma_1 \left[y_1(t) \mathbf{e}_{1p}(t) + \mathbf{x}_p(t) \sum_{h=1}^K w_{1h}(t) e_{1h}(t) \right], \quad (p = 1, 2, \dots, K) \quad (3.29)$$

koje u matricnoj formi mogu biti napisane kao

$$\frac{d\mathbf{w}_1}{dt} = \gamma_1 [y_1 \mathbf{e}_1 + \mathbf{x} \mathbf{w}_1^T \mathbf{e}_1] \quad (3.30)$$

za bilo koje $\mathbf{w}_1(0) \neq 0$ i $\gamma_1 > 0$.

Algoritam (3.30) može dalje biti uprošćen

$$\frac{d\mathbf{w}_1}{dt} = \gamma_1 y_1 \mathbf{e}_1 = \gamma_1 y_1 [\mathbf{x} - \mathbf{w}_1 y_1] \quad (3.31)$$

jer drugi član na desnoj strani jednačine (3.30) koji može biti napisan kao $\mathbf{x} \mathbf{w}_1^T \mathbf{e}_1 = \mathbf{x} \mathbf{w}_1^T (\mathbf{x} - \mathbf{w}_1 y_1) = \mathbf{x} (\mathbf{1} - \mathbf{w}_1^T \mathbf{w}_1) y_1$, teži brzo ka nuli jer $\mathbf{w}_1^T \mathbf{w}_1$ teži 1 kako t teži beskonačnosti i tako može biti zanemaren. Ova osobina može biti potvrđena kroz simulacije.

Interesantno je primetiti da direktna realizacija ovog zakona učenja

$$\mathbf{w}_1(i+1) = \mathbf{w}_1(i) + \gamma_1(i) y_1(i) [\mathbf{x}(i) - \mathbf{w}_1(i) y_1(i)], \quad (i = 0, 1, 2, \dots, K) \quad (3.32)$$

u stvari predstavlja poznati Ojin algoritam [Oja, 1982]. Kao što je već pomenuto, nije baš previše poznato da je Amari [Amari, 1978] bio prvi koji je predložio algoritam u kome je normalizovani Hebov zakon učenja bio korišćen za izdvajanje dominantne glavne komponente.

Predloženi zakon učenja može biti proširen na izračunavanje proizvoljnog broja glavnih komponenta korišćenjem principa samonadziranja i hijerarhijske neuralne mreže (vidi na primer [Sanger, 1989]).

Sada ćemo predstaviti pristup za sekvencijalnu ekstrakciju glavnih komponenta koje odgovaraju realnom, slučajnom ulaznom signalu sa srednjom vrednošću nula, bez estimacije kovarijanske matrice. Ekstrakcija će se obavljati sekvencijalno i onoliko dugo dok je poslednja izračunata glavna komponenta veća od nekog izabranog praga. Biće usvojeno da sporedne komponente odgovaraju aditivnom šumu.

Zakon učenja za izdvajanje druge glavne komponente $\lambda_2 = E(y_2^2(i))$ je sličan onome koji je korišćen pri izdvajanju dominantne glavne komponente. Razlika se sastoji u tome što se algoritam ne primenjuje direktno na ulazni vektor već na vektor koji se formira kao

$$\mathbf{x}_2(i) = \mathbf{x}_1(i) - \mathbf{x}\mathbf{p}_1(i) = \mathbf{x}_1(i) - \mathbf{w}_1 y_1(i) \quad (3.33)$$

i gde je $y_2(i) = \mathbf{w}_2^T \mathbf{x}_2(i)$. Odatle sledi da se k -ta glavna komponenta može pisati u formi

$$\mathbf{w}_k(i+1) = \mathbf{w}_k(i) + \gamma_k(i) y_k(i) \mathbf{x}_{k+1}(i), \quad (3.34)$$

gde je

$$\mathbf{e}_k = \mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{w}_k y_k, \quad y_k = \mathbf{w}_k^T \mathbf{x}_k, \quad \mathbf{x}_1(i) = \mathbf{x}(i). \quad (3.35)$$

Izlazni signali $y_k(i)$ posle primene ovog algoritma će biti dekorelisani i odgovaraće sopstvenim vrednostima $\lambda_k = E(y_k^2(i))$, ($k=1,2,\dots,N$).

3.5 Osnovne funkcije cene i adaptivni algoritmi za PCA

3.5.1 Rejljev koeficijent – osnovne osobine

Većina adaptivnih algoritama za PCA i MCA mogu direktno ili indirektno biti definisane korišćenjem Rejljevog koeficijenta (RK) specifične kovarijanske matrice kao funkcije cene.

Rejljev koeficijent $r(\mathbf{w})$ je definisan za $\mathbf{w} \neq 0$, kao

$$r(\mathbf{w}) = r(\mathbf{w}, \mathbf{C}) = \frac{\mathbf{w}^T \mathbf{C} \mathbf{w}}{\mathbf{w}^T \mathbf{w}} \quad (3.36)$$

gde je $\mathbf{C} = \mathbf{E}(\mathbf{x}\mathbf{x}^T)$. Rejljev koeficijent ima sledeće osobine:

1. Stacionarnost i kritične tačke:

$$\lambda_1 = \max r(\mathbf{w}, \mathbf{C}) \quad (3.37)$$

$$\lambda_K = \min r(\mathbf{w}, \mathbf{C}) \quad (3.38)$$

gde su λ_1 i λ_K najveća i najmanja sopstvena vrednost matrice \mathbf{C} .

Generalno govoreći, kritične tačke funkcije $r(\mathbf{w}, \mathbf{C})$ su sopstveni vektori i sopstvene vrednosti matrice \mathbf{C} . Pretpostavimo da za sopstvene vrednosti kovarijanske matrice važi

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K. \quad (3.39)$$

2. Homogenost:

$$r(\alpha \mathbf{w}, \beta \mathbf{C}) = \beta r(\mathbf{w}, \mathbf{C}) \quad \forall \alpha \neq 0, \quad \beta \neq 0. \quad (3.40)$$

3. Invarijantnost u odnosu na translaciju

$$r(\mathbf{w}, \mathbf{C} - \alpha \mathbf{I}) = r(\mathbf{w}, \mathbf{C}) - \alpha \quad (3.41)$$

4. Minimalni ostatak

$$\|(\mathbf{C} - r(\mathbf{w}, \mathbf{C})\mathbf{I})\mathbf{w}\| \leq \|(\mathbf{C} - \alpha\mathbf{I})\mathbf{w}\|_2, \quad \forall \mathbf{w} \neq 0 \text{ i svaki skalarni koeficijent } \alpha. \quad (3.42)$$

5. Ortogonalnost

$$\mathbf{w} \perp (\mathbf{C} - r(\mathbf{w}, \mathbf{C})\mathbf{I})\mathbf{w}. \quad (3.43)$$

6. Hesijan Rejljevog koeficijenta je

$$H_r(\mathbf{w}, \mathbf{C}) = \left[\frac{\partial^2 r(\mathbf{w}, \mathbf{C})}{\partial w_i \partial w_j} \right] = \frac{2}{\|\mathbf{w}\|_2^2} (\mathbf{C} - r(\mathbf{w}, \mathbf{C})\mathbf{I}) \left(\mathbf{I} - \frac{4}{\|\mathbf{w}\|_2^2} \mathbf{w}\mathbf{w}^T \right). \quad (3.44)$$

Za sopstvene vrednosti $\lambda_1, \lambda_2, \dots, \lambda_K$ i odgovarajuće sopstvene vektore $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K$ Hesijan može biti izražen kao [Cirrincione, 1998]

$$H_r(\mathbf{w}_k) = 2(\mathbf{C} - \lambda_k\mathbf{I}) \quad (3.45)$$

$$H_r(\mathbf{w}_k)\mathbf{w}_j = \begin{cases} 0, & k = j \\ 2(\lambda_j - \lambda_k)\mathbf{w}_j, & k \neq j \end{cases} \quad (3.46)$$

$$\det H_r(\mathbf{w}_k) = \det[\mathbf{C} - \lambda_k\mathbf{I}] = 0 \quad (3.47)$$

$$\text{tj. } H_r \text{ je singularna za svaki vektor } \mathbf{w}_k. \quad (3.48)$$

Hesijan H_r ima iste sopstvene vektore kao i \mathbf{C} , ali za različite sopstvene vrednosti.

Zgodno je zapaziti da nije praktično koristiti Njutnov i kvazi Njutnov metod za minimizaciju Rejljevog koeficijenta, pošto je Hesijan H_r singularna matrica u tačkama ekstremuma pa tako u tim tačkama ne postoji inverzna matrica.

3.5.2 Osnovne funkcije cene za sekvencijalno izračunavanje glavnih i sporednih komponentata

Maksimalna i minimalna sopstvena vrednost kovarijansne matrice $\mathbf{C}=\mathbf{E}(\mathbf{xx}^T)$ mogu biti izračunate kao ekstremumi Rejljevog koeficijenta, tako da sledeća funkcija cene može biti korišćena

$$J_2(\mathbf{w}) = r(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{C} \mathbf{w}}{\mathbf{w}^T \mathbf{w}}, \quad \mathbf{w} \in \Re^K, \quad \|\mathbf{w}\|_2 \neq 0. \quad (3.49)$$

Da bismo pronašli ekstremne tačke, možemo izračunati gradijent kao

$$\nabla_{\mathbf{w}} r(\mathbf{w}) = \frac{\partial r(\mathbf{w})}{\partial \mathbf{w}} = 2 \frac{\mathbf{C} \mathbf{w} (\mathbf{w}^T \mathbf{w}) - \mathbf{w} (\mathbf{w}^T \mathbf{C} \mathbf{w})}{(\mathbf{w}^T \mathbf{w})^2}, \quad (3.50)$$

na osnovu čega sledi da stacionarne tačke koje odgovaraju $\nabla_{\mathbf{w}} r(\mathbf{w})=0$ zadovoljavaju

$$\mathbf{C} \mathbf{w} = \frac{\mathbf{w}^T \mathbf{C} \mathbf{w}}{\mathbf{w}^T \mathbf{w}} \mathbf{w}. \quad (3.51)$$

Ova jednačina je zadovoljena ako je \mathbf{w} sopstveni vektor (jedinične dužine) matrice \mathbf{C} i pri čemu je odgovarajuća sopstvena vrednost data sa $\lambda = \frac{\mathbf{w}^T \mathbf{C} \mathbf{w}}{\mathbf{w}^T \mathbf{w}}$. Očigledno, minimum će odgovarati minimalnoj sopstvenoj vrednosti matrice \mathbf{C} , dok će maksimum $r(\mathbf{w})$ odgovarati maksimalnoj sopstvenoj vrednosti. U opštem slučaju, nule $\nabla_{\mathbf{w}} r(\mathbf{w})$ odgovaraju sopstvenim vektorima \mathbf{c} , za koje se usvaja da imaju jedinične dužine.

Predloženi optimizacioni problem bez ograničenja se može formulisati i kao problem sa ograničenjem:

Maksimiziraj $J_3(\mathbf{w}) = \mathbf{w}^T \mathbf{C} \mathbf{w}$ pri čemu je $\mathbf{w}^T \mathbf{w} = 1$.

Lagranžijan ovog problema sa ograničenjem je

$$J_3(\mathbf{w}, l) = \mathbf{w}^T \mathbf{C} \mathbf{w} + l(1 - \mathbf{w}^T \mathbf{w}) \quad (3.52)$$

gde $l \in \mathfrak{R}$ predstavlja Lagranžov multiplikator. Potreban i dovoljan uslov da stacionarna tačka odgovara sopstvenoj vrednosti matrice \mathbf{C} je

$$\nabla_{\mathbf{w}} J_3(\mathbf{w}, l) = 0 \quad i \quad \nabla_l J_3(\mathbf{w}, l) = 0. \quad (3.53)$$

i odgovara jednačini

$$\mathbf{C} \mathbf{w} = \mathbf{w} \lambda, \quad \mathbf{w}^T \mathbf{w} = 1. \quad (3.54)$$

Alternativno, umesto Lagranžijana, može se primeniti metod «penala» i na taj način formulisati sledeća funkcija cene

$$J_4(\mathbf{w}) = \mathbf{w}^T \mathbf{C} \mathbf{w} - \alpha(1 - \mathbf{w}^T \mathbf{w})^2 \quad (3.55)$$

gde je α pozitivni koeficijent «penala».

Druga klasa funkcija cene, bazirana na informaciono-teoretskom kriterijumu, je predložena kao [Abed-Meraim, 2000], [Baldi, 1995]

$$J_5(\mathbf{w}) = \log \frac{\mathbf{w}^T \mathbf{C} \mathbf{w}}{\mathbf{w}^T \mathbf{w}}. \quad (3.56)$$

i

$$J_6(\mathbf{w}) = \log(\mathbf{w}^T \mathbf{C} \mathbf{w}) - \mathbf{w}^T \mathbf{w}. \quad (3.57)$$

Još jedna važna i relativno jednostavna funkcija cene za PCA, koja nije bazirana na Rejljevom koeficijentu je:

$$J_7(\mathbf{w}, c) = E \{ \| \mathbf{x} - \mathbf{w}c \|_2 \} \quad (3.58)$$

gde je c skalar [Diamantaras, 1996].

Različite funkcije cene su sumirane u Tabeli 3.1.

Interesantno je zapaziti da minimizacija ovih funkcija cene dovodi do adaptivnih algoritama koji u opštoj formi mogu biti zapisani kao:

$$\mathbf{w}_1(i+1) = \mathbf{w}_1(i) + \gamma_1(i)F(\mathbf{w}_1(i), \mathbf{C}^{(i)}), \quad (3.59)$$

gde funkcija F uzima različite forme (vidi Tabelu 3.2). Kovarijansna matrica može biti on-line estimirana kao

$$\mathbf{C}^{(i)} = (1 - \eta_0)\mathbf{C}^{(i-1)} + \eta_0\mathbf{x}(i)\mathbf{x}^T(i), \quad (3.60)$$

gde je η_0 faktor učenja.

Tabela 3.1 Osnovne kriterijumske funkcije za generisanje PCA algoritama

1.	$J_1(\mathbf{w}_1) = E\left\{\left\ \mathbf{x}(i) - \mathbf{w}_1\mathbf{w}_1^T\mathbf{x}(i)\right\ _2^2\right\} \cong \sum_{i=1}^K \gamma^{K-i}\left\ \mathbf{x}(i) - \mathbf{w}_1\mathbf{w}_1^T\mathbf{x}(i)\right\ _2^2$
2.	$J_2(\mathbf{w}) = r(\mathbf{w}) = \frac{\mathbf{w}^T\mathbf{C}\mathbf{w}}{\mathbf{w}^T\mathbf{w}}$
3.	$J_3(\mathbf{w}, l) = \mathbf{w}^T\mathbf{C}\mathbf{w} + l(1 - \mathbf{w}^T\mathbf{w})$
4.	$J_4(\mathbf{w}) = \mathbf{w}^T\mathbf{C}\mathbf{w} - \alpha(1 - \mathbf{w}^T\mathbf{w})^2$
5.	$J_5(\mathbf{w}) = \log \frac{\mathbf{w}^T\mathbf{C}\mathbf{w}}{\mathbf{w}^T\mathbf{w}}$
6.	$J_6(\mathbf{w}) = \log(\mathbf{w}^T\mathbf{C}\mathbf{w}) - \mathbf{w}^T\mathbf{w}$
7.	$J_7(\mathbf{w}, c) = E\left\{\left\ \mathbf{x} - \mathbf{w}c\right\ _2\right\}$

Tabela 3.2 Osnovni adaptivni algoritmi za PCA

Br.	Zakon učenja	Reference
1.	$\bar{\mathbf{w}}_1(i+1) = \mathbf{w}_1(i) + \gamma y(i)\mathbf{x}(i),$ $\mathbf{w}_1(i+1) = \frac{\bar{\mathbf{w}}_1(i+1)}{\ \bar{\mathbf{w}}_1(i+1)\ _2^2}$	[Amari, 1978]
2.	$\Delta \mathbf{w}_1 = \gamma \left(y_1 \mathbf{x} - y_1 ^2 \mathbf{w}_1 \right)$	[Oja, 1982]
3.	$\Delta \mathbf{w}_1 = \gamma \left(y_1 \mathbf{x} \mathbf{w}_1^T \mathbf{w}_1 - y_1 ^2 \mathbf{w}_1 \right)$	[Chen, 1998]
4.	$\Delta \mathbf{w}_1 = \gamma \left(y_1 \mathbf{x} - \ \mathbf{w}_1\ ^2 \mathbf{w}_1 \right)$	[Yullie, 1994]
5.	$\Delta \mathbf{w}_1 = \gamma g \left(\mathbf{w}_1^T \mathbf{w}_1 \right) \left(\mathbf{C} \mathbf{w}_1 - \frac{\mathbf{w}_1^T \mathbf{C} \mathbf{w}_1}{\mathbf{w}_1^T \mathbf{w}_1} \mathbf{w}_1 \right),$ $g(\mathbf{w}_1^T \mathbf{w}_1) = 1, \text{ ili } \left(\mathbf{w}_1^T \mathbf{w}_1 \right)^{-1}$	[Luo, 1996] [Chatterje, 1999]
6.	$\Delta \mathbf{w}_1 = \gamma \left(y_1 \mathbf{x} - y_1 ^2 \mathbf{w}_1 - \mathbf{w}_1 (1 - \mathbf{w}_1^T \mathbf{w}_1) \right)$	[Abed-Meraim, 2000]
7.	$\Delta \mathbf{w}_1 = \gamma \left(2y_1 \mathbf{x} - y_1 ^2 \mathbf{w}_1 - y_1 \mathbf{w}_1^T \mathbf{w}_1 \mathbf{x} \right)$	[Abed-Meraim, 2000], [Douglas, 1999], [Hua, 1999]
8.	$\Delta \mathbf{w}_1 = \gamma y_1 \Psi_1(\mathbf{x} - y_1 \mathbf{w}_1)$	[Cichocki, 1993]

3.5.3 Brzi PCA algoritam baziran na power metodi

Brzi algoritmi za izračunavanje PCA mogu biti izvedeni na bazi *power* metoda uz korišćenje osobina Rejljevog koeficijenta [Rao, 2002], [Roweis, 1998]. Ako usvojimo da dominantni sopstveni vektor \mathbf{w}_1 ima jediničnu dužinu, tj. $\mathbf{w}_1^T \mathbf{w}_1 = 1$, on može biti izračunat korišćenjem sledećeg iterativnog postupka

$$\mathbf{w}_1(l+1) = \frac{\mathbf{C} \mathbf{w}_1(l)}{\mathbf{w}_1^T(l) \mathbf{C} \mathbf{w}_1(l)} \quad (3.61)$$

Ako se uzme u obzir da je $y_1^{(l)}(i) = \mathbf{w}_1^T(l)\mathbf{x}(i)$ i da je $C_p = \langle \mathbf{x}(i) \mathbf{x}^T(i) \rangle$, možemo koristiti sledeću uprošćenu formulu

$$\mathbf{w}_1(l+1) = \frac{\sum_{i=1}^M y_1^{(l)}(i)\mathbf{x}(i)}{\sum_{i=1}^M [y_1^{(l)}(i)]^2}, \quad (3.62)$$

ili opštije, za veći broj glavnih komponentata možemo koristiti deflacioni postupak

$$\mathbf{w}_k(l+1) = \frac{\sum_{i=1}^M y_k^{(l)}(i)\mathbf{x}_k(i)}{\sum_{i=1}^M [y_k^{(l)}(i)]^2}, \quad (k = 1, 2, \dots, N) \quad (3.63)$$

gde je $y_k^{(l)}(i) = \mathbf{w}_k^T(l)\mathbf{x}_k(i)$. Nakon konvergencije vektora $\mathbf{w}_k(l)$ ka $\mathbf{w}_{k\#}$, možemo izvršiti deflaciju $\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{w}_{k\#}y_k$, $\mathbf{x}_1 = \mathbf{x}$.

Ovakav algoritam može rigoroznije biti izveden u nešto modifikovanoj formi kao minimizacija sledeće funkcije cene:

$$\begin{aligned} J_1(\mathbf{w}, y^{(l)}) &= \sum_{i=1}^M \|\mathbf{x}(i) - y^{(l)}\mathbf{w}\|_2^2 \\ &= \sum_{i=1}^M \|\mathbf{x}(i)\|_2^2 + \|\mathbf{w}\|_2^2 \sum_{i=1}^M [y^{(l)}(i)]^2 - 2\mathbf{w}^T \sum_{i=1}^M y^{(l)}(i)\mathbf{x}(i), \end{aligned} \quad (3.64)$$

pri čemu je ograničenje $\|\mathbf{w}\|_2 = 1$. Gornja funkcija cene dostiže ravnotežnu tačku kada je gradijent J_1 nula, tj. u

$$\mathbf{w}_{\#} = \frac{\sum_{i=1}^M y_{\#}(i)\mathbf{x}(i)}{\sum_{i=1}^M y_{\#}^2(i)}. \quad (3.65)$$

To nam sugerira sledeću iterativnu formulu [Roweis, 1998], [Tipping, 1999]

$$\mathbf{w}^{(l+1)} = \frac{\sum_{i=1}^M y^{(l)}(i) \mathbf{x}(i)}{\left\| \sum_{i=1}^M y^{(l)}(i) \mathbf{x}(i) \right\|_2}. \quad (3.66)$$

Outline algoritma za brzo izračunavanje dominantne glavne komponente

1. Inicijalizacija: postavi $\mathbf{w}_1(0) \neq 0$ i $l = 0$
2. Izračunaj $y_1^{(l)}(i) = \mathbf{w}_1^T(l) \mathbf{x}(i)$, ($i=1, 2, \dots, M$)
3. Izračunaj

$$\mathbf{w}^{(l+1)} = \frac{\sum_{i=1}^M y^{(l)}(i) \mathbf{x}(i)}{\left\| \sum_{i=1}^M y^{(l)}(i) \mathbf{x}(i) \right\|_2}$$

4. Stani ako je

$$1 - \frac{J_1(\mathbf{w}_1^{(l+1)})}{J_1(\mathbf{w}_1^l)}$$

manje od nekakvog praga ε . Inače, dodeli l vrednost $l+1$ i idi na korak 2.

Ovde ćemo napomenuti da brzina konvergencije *power* algoritma zavisi od odnosa λ_2/λ_{\max} , gde je λ_2 druga najveća sopstvena vrednost matrice \mathbf{C} [Cichocki, 2002]. Ovaj odnos je generalno manji od 1, što omogućava adekvatnu konvergenciju algoritma. Međutim, ako je sopstvena vrednost $\lambda_1 = \lambda_{\max}$ bliska po vrednosti jednoj ili više sopstvenih vrednosti matrice \mathbf{C} , drugim rečima ako λ_1 pripada klasteru sopstvenih vrednosti, tada je odnos veoma blizak jedinici, što uslovljava veoma sporu konvergenciju, što za posledicu može imati nepreciznu estimaciju sopstvenog vektora \mathbf{w}_1 . U slučaju postojanja višestrukih sopstvenih vrednosti, *power* metod ne konvergira.

3.5.4 Inverzni iterativni power metod

Neke od mana *power* metoda se mogu delimično prevazići primenom *power* metoda na matricu $\mathbf{T}_{xx} = (\mathbf{C} - \sigma \mathbf{I})^{-1}$ umesto na matricu \mathbf{C} , gde je σ pozitivni koeficijent, koji se naziva pomeraj, i koji se specificira od strane korisnika. Ovaj metod konvergira ka sopstvenom vektoru koji odgovara sopstvenoj vrednosti λ_j koja je najbliža po vrednosti σ , a ne λ_{\max} . Ovaj metod se naziva inverzni *power* metod i na osnovu njega je moguće formulisati sledeći algoritam:

$$\mathbf{w}_T(l+1) = \frac{\mathbf{T}_{xx} \mathbf{w}_T(l)}{\|\mathbf{T}_{xx} \mathbf{w}_T(l)\|_2} = \frac{\mathbf{z}^{(l)}}{\|\mathbf{z}^{(l)}\|_2}, \quad (3.67)$$

Gde je $\mathbf{z}^{(l)} = (\mathbf{C} - \sigma \mathbf{I})^{-1} \mathbf{w}^{(l)}$. Kriterijum za zaustavljanje je dat sa

$$\|\mathbf{z}^{(l)} - \lambda_T(l) \mathbf{w}^{(l)}\|_2 \leq \varepsilon$$

gde je $\lambda_T = \mathbf{w}_T^T(l) \mathbf{z}^{(l)}$ i ε mali prag. Nakon konvergencije je $\mathbf{w}_j = \mathbf{w}_T / \lambda_T$ i $\lambda_j = \sigma + 1 / \lambda_T$.

Inverzni *power* metod konvergira ako $\mathbf{w}_T(0)$ nije normalno na \mathbf{w}_j . Brzina konvergencije je određena odnosom $|(\lambda_j - \sigma) / (\lambda_i - \sigma)|$ gde je λ_i sopstvena vrednost matrice \mathbf{C} takva da $|\lambda_i - \sigma|^{-1}$ predstavlja drugu najveću sopstvenu vrednost matrice \mathbf{T}_{xx} . Algoritam je veoma efikasan ako imamo dobru aproksimaciju sopstvene vrednosti za koju želimo da izračunamo odgovarajući sopstveni vektor. Ako izaberemo σ veoma blisko željenoj sopstvenoj vrednosti, algoritam može da konvergira veoma brzo [Bai, 2000]. Jedna od dobrih osobina inverznog *power* algoritma je njegova mogućnost da konvergira ka bilo kojoj sopstvenoj vrednosti koja je najbliža σ . Za druge efikasne i brze algoritme za izračunavanje sopstvenih vrednosti treba pogledati [Bai, 2000].

3.5.5 Robusna PCA

Algoritmi za učenje koji su predstavljeni u prethodnom paragrafu (3.29 i 3.30) su optimalni samo u slučaju kada ulazni podaci zadovoljavaju Gausovu raspodelu i osetljivi su na impulsni šum i autlajere (*outlier* – signali sa velikom amplitudom koji se retko javljaju).

Moguće je na više načina povećati robusnost PCA sa aspekta šuma i autlajera. Prvo, autlajeri mogu biti eliminisani iz ulaznih podataka; drugo mogu biti potisnuti ili zamenjeni nekim odgovarajućim vrednostima; i konačno, robusniji kriterijum može biti primenjen. Na primer, veoma robusan i efikasan metod primenjuje estimator minimalne determinante kovarijanse (MCD) [Rousseev, 1999]. MCD traži podskup od L uzoraka koji rezultuju najmanjom determinantom kovarijansne matrice koja je izračunata na osnovu tih uzoraka. Tipično, L je približno jednako $3M/4$ ili $M/2$ ako je M broj ukupnih uzoraka. Glavni nedostatak ovog metoda je što nije pogodan za primenjivanje kod problema sa velikim brojem dimenzija (kao što su problemi vezani za procesiranje slike ili kompjuterskog vida). Zato će u ovoj sekciji biti razvijen robusan algoritam baziran na M-estimatoru.

Kako bismo to uradili definišimo sledeću funkciju cene

$$J_{1\rho}(\mathbf{w}_1) = \rho(e_1) = \sum_{p=1}^K \rho_p(\mathbf{e}_{1p}), \quad (3.68)$$

gde je $\rho_p(\mathbf{e}_{1p})$ realna, tipično konveksna funkcija koja je u statistici poznata kao «robustna funkcija gubitka». Kako bi se redukovao uticaj autlajera predlagane su različite robustne funkcije gubitka $\rho(\mathbf{e})$. Ovde ćemo dati samo četiri primera [Cichocki, 1993]:

1. Apsolutna vrednost

$$\rho_A(e) = |e| \quad (3.69)$$

2. Huberova funkcija

$$\rho_H(e) = \begin{cases} e^2/2 & |e| \leq \beta \\ \beta|e| - \beta^2/2 & |e| > \beta, \end{cases} \quad (3.70)$$

3. Talvarova funkcija

$$\rho_T(e) = \begin{cases} e^2 / 2 & |e| \leq \beta \\ \beta^2 / 2 & |e| > \beta \end{cases} \quad (3.71)$$

4. Logistička funkcija

$$\rho_L(e) = \beta^2 \log(\cosh(e / \beta)), \quad (3.72)$$

gde je $\beta > 0$ parametar koji zavisi od vrste problema, nazvan parametar odsecanja (tipična vrednost mu je $1 < \beta < 3$). Tipične funkcije gubitka robusnosti i njihove funkcije aktivacije su definisane sa

$$\Psi_p(e_p) = \frac{\partial \rho_p}{\partial e_p} \quad (3.73)$$

i prikazane su u literaturi [Cichocki, 2002].

Generalno govoreći, izbor odgovarajuće funkcije gubitka zavisi od raspodele ulaznog vektora $\mathbf{x}(t)$. Primenjujući standardni opadajući gradijentni metod na energetska funkciju (3.68) dobijamo sledeći algoritam (predstavlja generalizaciju jednačine (3.29))

$$\frac{dw_{1p}}{dt} = \mu_1 \left[y_1 \Psi_p(e_{1p}) + \mathbf{x}_p \sum_{h=1}^K w_{1h} \Psi_h(e_{1h}) \right], \quad (3.74)$$

gde $\mu(t)$ pozitivno i

$$\Psi_p(e_{1p}) = \frac{\partial \rho_p(e_{1p})}{\partial e_{1p}}$$

U matricnoj formi ovaj algoritam se može pisati kao

$$\frac{d\mathbf{w}_1}{dt} = \mu_1 \left[y_1 \Psi(\mathbf{e}_1) + \mathbf{x} \mathbf{w}_1^T \Psi(\mathbf{e}_1) \right], \quad (3.75)$$

gde je

$$\Psi(\mathbf{e}_1) = [\Psi_1(e_{11}), \Psi_2(e_{12}), \dots, \Psi_N(e_{1N})]^T.$$

Uprošćena verzija ovakvog zakona učenja ima oblik

$$\frac{d\mathbf{w}_1}{dt} = \mu_1 y_1 \Psi(\mathbf{e}_1) = \mu_1 y_1 \Psi[\mathbf{x} - \mathbf{w}_1 y_1] \quad (3.76)$$

gde je $\mu > 0$.

Korišćenjem principa samonadziranja i kaskadnih hijerarhijskih neuralnih mreža, zakoni učenja (3.74), (3.75) i (3.76) mogu biti prošireni za dobijanje većeg broja značajnih glavnih komponenta. Drugim rečima, algoritam učenja za određivanje druge glavne komponente (komponente koja odgovara drugoj najvećoj sopstvenoj vrednosti) je veoma sličan onome za određivanje prve sopstvene vrednosti, pri čemu se proces učenja ne primenjuje direktno na ulaznu komponentu \mathbf{x} već na dostupnu grešku

$$\mathbf{x}p_1 = \mathbf{e}_1 = \mathbf{x} - \mathbf{w}_1 y_1 \quad (3.77)$$

i

$$y_2 = \mathbf{w}_2^T \mathbf{e}_1 \quad (3.78)$$

Ostale komponente se izračunavaju analogno. U tom slučaju se može definisati sekvencijalna funkcija cene kao

$$J_\rho(\mathbf{w}_k) = \rho(\mathbf{e}_k) = \sum_{p=1}^K \rho_p(\mathbf{e}_{kp}), \quad (k = 1, 2, \dots, N) \quad (3.79)$$

gde je $\mathbf{e}_k = \mathbf{e}_{k-1} - \mathbf{w}_k y_k$, $y_k = \mathbf{w}_k^T \mathbf{e}_{k-1}$, i gde je $\mathbf{e}_0(t) = \mathbf{x}(t)$. Minimizacija ove funkcije cene primenom metoda opadajućeg gradijenta dovodi do adaptivnog algoritma učenja

$$\frac{d\mathbf{w}_k}{dt} = \mu_i [y_k \Psi(\mathbf{e}_k) + \mathbf{e}_{k-1} \mathbf{w}_k^T \Psi(\mathbf{e}_k)] \quad (3.80)$$

za bilo koje $\mathbf{w}_k(0) \neq 0$ ($k=1, 2, \dots, N$) gde je

$$\mu_k(t) > 0$$

$$\Psi(\mathbf{e}_k) = [\Psi_1(e_{k1}), \Psi_1(e_{k2}), \dots, \Psi_1(e_{kK})]^T,$$

$$\Psi_p(e_{kp}) = \frac{\partial \rho_p(e_{kp})}{\partial (e_{kp})},$$

$$\mathbf{e}_0(t) = \mathbf{x}(t).$$

Obično je drugi član na desnoj strani jednačine (3.80) relativno mali i može biti zanemaren. Tada dobijamo uprošćenu verziju zakona učenja za ekstrakciju najvećih N glavnih komponentata

$$\frac{d\mathbf{w}_k(t)}{dt} = \mu_k(t) y_k(t) \Psi[\mathbf{e}_k(t)] \quad (3.81)$$

ili u diskretnoj formi

$$\mathbf{w}_k(i+1) = \mathbf{w}_k(i) + \gamma_k(i) y_k(i) \Psi[\mathbf{e}_k(i)] \quad (3.82)$$

gde je $\mathbf{w}_k(0) \neq 0$ i $\gamma_k(i) > 0$.

3.6 Adaptivni algoritmi za sekvencijalnu ekstrakciju sporednih komponentata

Na prvi pogled izgleda logično da se za ekstrakciju sporednih komponentata mogu koristiti algoritmi koji su slični onima za izračunavanje glavnih komponentata, ali jednostavna promena znaka faktora učenja kod velike većine algoritama izaziva numeričku nestabilnost. To dalje znači da je potrebno uvesti dodatne stabilizacione članove kako bi se obezbedila stabilnost algoritma. Na primer, Ojin zakon učenja za izračunavanje dominantne komponente može biti modifikovan da izračunava najmanju sporednu komponentu na sledeći način

$$\mathbf{w}(i+1) = \mathbf{w}(i) - \gamma(i) \left(y(i)\mathbf{x}(i) - y^2(i)\mathbf{w}(i) + (\mathbf{w}^T(i)\mathbf{w}(i) - 1)\mathbf{w}(i) \right) \quad (3.83)$$

Lako je uočiti da zakon učenja sadrži dodatni kazneni član $(\mathbf{w}^T\mathbf{w}-1)$ koji obezbeđuje stabilnost algoritma forsiranjem dužine sinaptičkog vektora na jediničnu vrednost.

Široka klasa MCA algoritama se može izvesti iz minimizacionog problema (bez ograničenja)

$$\min_{\mathbf{w} \in \mathbb{R}^k} r(\mathbf{w})/2 \quad \text{i} \quad r(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{C} \mathbf{w}}{\mathbf{w}^T \mathbf{w}}. \quad (3.84)$$

Primenjujući opadajući gradijentni metod direktno, dobijamo sistem nelinearnih običnih diferencijalnih jednačina

$$\frac{d\mathbf{w}}{dt} = -\gamma \nabla_{\mathbf{w}} r(\mathbf{w}) = -\gamma \frac{\partial r(\mathbf{w})}{\partial \mathbf{w}} = -\gamma \left[\frac{\mathbf{C} \mathbf{w} \mathbf{w}^T \mathbf{w} - \mathbf{w} (\mathbf{w}^T \mathbf{C} \mathbf{w})}{(\mathbf{w}^T \mathbf{w})^2} \right] \quad (3.85)$$

gde je $\gamma(t) > 0$ faktor učenja.

Važna osobina gornjeg zakona učenja je da je norma (dužina) vektora \mathbf{w} konstantna u toku učenja i jednaka jedinici. To je jednostavno dokazati na sledeći način, pošto je

$$\frac{d\|\mathbf{w}\|_2^2}{dt} = 2\mathbf{w}^T \frac{d\mathbf{w}}{dt} = 0, \quad (3.86)$$

pri čemu je neophodno da inicijalno \mathbf{w} nije 0. To znači da je

$$\|\mathbf{w}(t)\|_2 = \|\mathbf{w}(0)\|_2, \quad \forall t \geq 0. \quad (3.87)$$

Bez gubitka opštosti, može se usvojiti da je norma $\|\mathbf{w}(0)\| = 1$. To dalje znači da član $\mathbf{w}^T \mathbf{w}$ može biti zanemaren ili formalno apsorbovan u pozitivni faktor učenja $\gamma(t)$ tako da važi

$$\frac{d\mathbf{w}}{dt} = -\tilde{\gamma} \left(\mathbf{C}\mathbf{w} - \mathbf{w} \frac{\mathbf{w}^T \mathbf{C}\mathbf{w}}{\mathbf{w}^T \mathbf{w}} \right). \quad (3.88)$$

Može se pokazati da ova jednačina može biti interpretirana i kao specijalni slučaj Broketovog dvostrukog *bracket* toka [Brockett, 1991a], [Brockett, 1991b].

Na bazi jednačine (3.88), u literaturi je predloženo više modifikacija koje u generalnoj formi mogu biti napisane kao

$$\frac{d\mathbf{w}}{dt} = -\gamma(t) g(\mathbf{w}^T \mathbf{w}) \left(\mathbf{C}\mathbf{w} - \frac{\mathbf{w}^T \mathbf{C}\mathbf{w}}{\mathbf{w}^T \mathbf{w}} \mathbf{w} \right) \quad (3.89)$$

ili ekvivalentno

$$\frac{d\mathbf{w}}{dt} = -\gamma(t) \frac{g(\mathbf{w}^T \mathbf{w})}{\mathbf{w}^T \mathbf{w}} \left[\mathbf{C}\mathbf{w}\mathbf{w}^T \mathbf{w} - (\mathbf{w}^T \mathbf{C}\mathbf{w}) \mathbf{w} \right], \quad (3.90)$$

gde $g(\mathbf{w}^T \mathbf{w})$ uzima različite forme (na primer $(\mathbf{w}^T \mathbf{w})$, 1, $(\mathbf{w}^T \mathbf{w})^{-1}$).

U diskretnom domenu algoritam se u najjednostavnijoj formi može pisati kao

$$\mathbf{w}(i+1) = \mathbf{w}(i) - \gamma(i) \left[\mathbf{C}^{(i)} \mathbf{w}(i) - \frac{\mathbf{w}^T(i) \mathbf{C}^{(i)} \mathbf{w}(i)}{\mathbf{w}^T(i) \mathbf{w}(i)} \mathbf{w}(i) \right], \quad (3.91)$$

a njegova *online* verzija

$$\mathbf{w}(i+1) = \mathbf{w}(i) - \gamma(i)g\left(\|\mathbf{w}(i)\|_2^2\right)\left[y(i)\mathbf{x}(i) - \frac{y^2(i)}{\mathbf{w}^T(i)\mathbf{w}(i)}\mathbf{w}(i)\right], \quad (3.92)$$

gde je $y(i) = \mathbf{w}^T\mathbf{x}(i)$.

Nažalost, usled numeričkih aproksimacija gornji diskretni algoritam je nestabilan (algoritam divergira posle velikog broja iteracija, ukoliko se ne koristi periodična normalizacija na jediničnu dužinu). Da bi se sprečila nestabilnost, faktor učenja mora eksponencijalno da teži ka nuli ili moramo da ortonormalizujemo vektor $\mathbf{w}(i)$ posle svakih nekoliko iteracija:

$$\mathbf{w}(i) = \frac{\mathbf{w}(i)}{\|\mathbf{w}(i)\|_2}. \quad (3.93)$$

Posle ekstrakcije prve sporedne komponente, u cilju ekstrakcije ostalih sporednih komponentata, umesto eliminacije vektora \mathbf{w}_k iz kovarijansne matrice, pokušava se ekstrakcija dominantne glavne komponente nove kovarijansne matrice koja se definiše kao

$$\mathbf{C}^{(2)} = \mathbf{C}^{(1)} + \kappa_n \mathbf{w}_n \mathbf{w}_n^T, \quad (3.94)$$

gde je $\mathbf{C}^{(1)} = E\{\mathbf{xx}^T\}$ i κ_n je fiksna konstanta veća od λ_l .

Svi gore pomenuti algoritmi za MCA su vrlo spori i njihova konvergencija veoma zavisi od izabranog faktora učenja $\gamma(i)$.

Nedavno, Sakai i Šimizu su izvršili modifikaciju brzog PCA RLS algoritma za MCA [Sakai, 1997] (vidi takođe [Cichocki, 1993]):

$$y_k(i) = \mathbf{w}_k^T(i)\mathbf{x}_k(i), \quad (3.95)$$

$$\bar{\mathbf{w}}_k(i+1) = \mathbf{w}_k(i) - \frac{y_k(i)}{\eta_k^{-1}(i)}[\mathbf{x}_k(i) - y_k(i)\mathbf{w}_k(i)], \quad (3.96)$$

$$\eta_k^{-1}(i+1) = \gamma\eta_k^{-1}(i) + |y_k(i)|^2, \quad (3.97)$$

$$\bar{\mathbf{w}}_k \leftarrow \bar{\mathbf{w}}_k - \sum_{j=m}^{k+1} (\bar{\mathbf{w}}_k^T \bar{\mathbf{w}}_j) \bar{\mathbf{w}}_j, \quad \mathbf{w}_k(i) = \bar{\mathbf{w}}_k(i) (\bar{\mathbf{w}}_k^T(i) \bar{\mathbf{w}}_k(i))^{-1/2} \quad (3.98)$$

$$\mathbf{x}_{k-1}(i) = \mathbf{x}_k + \kappa_m y_k(i) \mathbf{w}_k(i) \quad k = m, m-1, K, \quad (3.99)$$

Tabela 3.3: Adaptivni algoritmi za MCA

$$\mathbf{x}_n(i) = \mathbf{x}(i), \tag{3.100}$$

κ_m je večje od λ_1 .

Glavna razlika između ovog algoritma i RLS PCA algoritma leži u promeni znaka faktora učenja, ortonormalizaciji vektora $\mathbf{w}_k(i)$ u svakoj iteraciji i u tome što se već ekstrahovane sporedne komponente premeštaju na glavne komponente.

Različiti adaptivni algoritmi za MCA su prikazani u Tabeli 3.3.

	Referenca
1. $\Delta \mathbf{w}_k(i+1) = -\gamma(i) [\mathbf{x}_k(i) \mathbf{x}^T(i) - \lambda_k \mathbf{w}_k(i) \mathbf{w}_k^T(i)] \mathbf{x}_k(i)$	[Luo, 1997]
2. $\Delta \mathbf{w}_k(i+1) = -\gamma(i) [\mathbf{x}_k(i) \mathbf{x}^T(i) - \frac{\gamma_k^2(i)}{\ \mathbf{w}_k(i)\ _2^2} \mathbf{w}_k(i) \mathbf{w}_k^T(i)] \mathbf{x}_k(i)$	[Xu, 1994a; Xu, 1994b]
3. $\Delta \mathbf{w}_k(i+1) = -\gamma(i) \lambda_k(i) [\mathbf{w}_k(i) \mathbf{w}_k^T(i)]^{-1} \mathbf{x}_k(i) - \lambda_k(i) \mathbf{w}_k(i)$	[Chen, 1996]
4. $\Delta \mathbf{w}_k(i+1) = -\gamma(i) \lambda_k(i) [\mathbf{w}_k(i) \mathbf{w}_k^T(i)]^{-1} \mathbf{x}_k(i) - \lambda_k(i) \mathbf{w}_k(i)$	[Douglas, 1999]
5. $\Delta \mathbf{w}_k(i+1) = -\gamma(i) [\mathbf{x}_k(i) \mathbf{x}^T(i) - \log\ \mathbf{w}_k(i)\ _p \mathbf{w}_k(i) \mathbf{w}_k^T(i)] \mathbf{x}_k(i)$	[Amari, 1998]
6. $\Delta \mathbf{w}_k(i+1) = -\gamma(i) [\mathbf{x}_k(i) \mathbf{x}^T(i) - (\alpha - \ \mathbf{w}_k(i)\ _p) \mathbf{w}_k(i) \mathbf{w}_k^T(i)] \mathbf{x}_k(i)$ $\forall \alpha \leq \alpha < \lambda_{\max}$	[Zhang, 2009]
7. $\Delta \mathbf{w}_k(i+1) = -\gamma(i) [\mathbf{w}_k(i) - \lambda_k(i) \mathbf{x}_k(i)] \mathbf{w}_k^T(i) \mathbf{x}_k(i)$	[Cibicki, 2002]
8. $\mathbf{x}_k(i) = \mathbf{w}_k^T(i) \mathbf{x}(i)$ $\mathbf{w}_k(i+1) = \mathbf{w}_k(i) - \frac{\lambda_k(i)}{\ \mathbf{w}_k(i)\ _2} [\mathbf{x}_k(i) - \lambda_k(i) \mathbf{w}_k(i)]$ $\sigma_k^2(i+1) = \sigma_k^2(i) + \gamma_k(i) \frac{\lambda_k(i)}{\ \mathbf{w}_k(i)\ _2^2} \mathbf{x}_k^2(i)$ $\bar{\mathbf{w}}_k \leftarrow \bar{\mathbf{w}}_k - \sum_{j=n}^{k+1} (\bar{\mathbf{w}}_j^T \bar{\mathbf{w}}_k) \bar{\mathbf{w}}_j, \quad \mathbf{w}_k(i) = \bar{\mathbf{w}}_k(i) (\bar{\mathbf{w}}_k^T(i) \bar{\mathbf{w}}_k(i))^{-1/2}$ $\mathbf{x}_{k+1}(i) = \mathbf{x}_k(i) + \gamma_{k+1} \mathbf{y}_k(i) \mathbf{w}_k(i), \quad k = m, m-1, \dots, K, \quad \mathbf{x}_m(i) = \mathbf{x}(i)$	[Sica, 1997]

Tabela 3.3 Adaptivni algoritmi za MCA

Br.	Algoritam	Reference
1.	$\Delta \mathbf{w}_k(i+1) = -\gamma(i) \left(y_k(i) \mathbf{x}(i) - y_k^2(i) \mathbf{w}(i) + (\mathbf{w}_k^T(i) \mathbf{w}_k(i) - 1) \mathbf{w}_k(i) \right)$	[Oja, 1992a]
2.	$\Delta \mathbf{w}_k(i+1) = -\gamma(i) g \left(\ \mathbf{w}_k(i)\ _2^2 \right) \left[y_k(i) \mathbf{x}(i) - \frac{y_k^2(i)}{\mathbf{w}_k^T(i) \mathbf{w}_k(i)} \mathbf{w}_k(i) \right],$ $g(\mathbf{w}^T \mathbf{w}) = 1, \text{ ili } \mathbf{w}_k^T \mathbf{w}_k \text{ ili } \left(\mathbf{w}_k^T \mathbf{w}_k \right)^{-1}$	[Oja, 1992a]. [Luo, 1997] [Cirincione, 1998]
3.	$\Delta \mathbf{w}_k(i+1) = -\gamma(i) \left[y_k(i) \mathbf{x}(i) - \frac{y_k^2(i)}{\ \mathbf{w}_k(i)\ _2^4} \mathbf{w}_k(i) \right]$	[Xu, 1994a; Xu, 1994b]
4.	$\Delta \mathbf{w}_k(i+1) = -\gamma(i) y_k(i) \left[\ \mathbf{w}_k(i)\ _2^2 \mathbf{x}(i) - y_k(i) \mathbf{w}_k(i) \right]$	[Chen, 1998]
5.	$\Delta \mathbf{w}_k(i+1) = -\gamma(i) y_k(i) \left[\ \mathbf{w}_k(i)\ _2^4 \mathbf{x}(i) - y_k(i) \mathbf{w}_k(i) \right]$	[Douglas, 1999]
6.	$\Delta \mathbf{w}_k(i+1) = -\gamma(i) \left[y_k(i) \mathbf{x}(i) - \log \ \mathbf{w}_k(i)\ _p \mathbf{w}_k(i) \right]$	[Amari, 1998]
7.	$\Delta \mathbf{w}_k(i+1) = -\gamma(i) \left[y_k(i) \mathbf{x}(i) - (d - \ \mathbf{w}_k(i)\ _p) \mathbf{w}_k(i) \right]$ gde je $d > \lambda_{\max}$	[Zhang, 2000]
8.	$\Delta \mathbf{w}_k(i+1) = -\gamma(i) \left[\mathbf{w}_k(i) - y_k(i) \mathbf{x}(i) \ \mathbf{w}_k(i)\ _p \right]$	[Cichocki, 2002]
9.	$y_k(i) = \mathbf{w}_k^T(i) \mathbf{x}_k(i),$ $\bar{\mathbf{w}}_k(i+1) = \mathbf{w}_k(i) - \frac{y_k(i)}{\eta_k^{-1}(i)} \left[\mathbf{x}_k(i) - y_k(i) \mathbf{w}_k(i) \right],$ $\eta_k^{-1}(i+1) = \gamma \eta_k^{-1}(i) + y_k(i) ^2,$ $\bar{\mathbf{w}}_k \leftarrow \bar{\mathbf{w}}_k - \sum_{j=m}^{k+1} \left(\bar{\mathbf{w}}_k^T \bar{\mathbf{w}}_j \right) \bar{\mathbf{w}}_j, \quad \mathbf{w}_k(i) = \bar{\mathbf{w}}_k(i) \left(\bar{\mathbf{w}}_k^T(i) \bar{\mathbf{w}}_k(i) \right)^{-1/2}$ $\mathbf{x}_{k-1}(i) = \mathbf{x}_k + \gamma_m y_k(i) \mathbf{w}_k(i) \quad k = m, m-1, \dots, K, \quad \mathbf{x}_n(i) = \mathbf{x}(i),$	[Sakai, 1997]

IV Poglavlje

Paralelni algoritmi za estimaciju glavnih komponentata, sporednih komponentata i njihovih potprostora

U prethodnom poglavlju su prikazani osnovni adaptivni algoritmi za sekvencijalnu ekstrakciju glavnih i sporednih komponentata. U ovom poglavlju će biti prikazan jedan mogući generalni pristup paralelnim algoritmima za ekstrakciju glavnih komponentata, sporednih komponentata ili njihovih potprostora. Prvo će biti analizirana funkcija cene koja dovodi do opšteg metoda za ekstrakciju više glavnih (PCA) ili sporednih (MCA) komponentata. Postoje slučajevi kada nije neophodno identifikovati glavne/sporedne komponente već samo potprostore koje oni definišu. U tom slučaju ćemo se baviti analizom glavnog (PSA) i sporednog potprostora (MSA). Detaljno će biti opisan jedan od najpoznatijih PSA algoritama – SLA algoritam. U većini slučajeva PCA/MCA su nastali modifikacijom nekog od PSA/MSA algoritama tako da ćemo se baviti generalnim principima koji omogućavaju formiranje algoritama bilo za PSA/MSA ili PCA/MCA. Navedeno je nekoliko najpoznatijih PSA/PCA i MSA/MCA algoritama. Na kraju je pokazana analiza stabilnosti za jednu klasu algoritama.

4.1 Funkcija cene za paralelnu ekstrakciju glavnih komponentata i njihovih potprostora

Prvo ćemo dati intuitivno objašnjenje ideje. Neka su $\{\lambda_1, \dots, \lambda_m\}$ i $\{d_1, \dots, d_m\}$ dva skupa od po m pozitivnih brojeva, pri čemu je $d_1 > \dots > d_m > 0$. Razmotrimo problem maksimizacije (odnosno minimizacije) sume

$$S = \sum_{i=1}^m \lambda_{i'} d_i, \quad (4.1)$$

pri čemu $\{\lambda_{1'}, \dots, \lambda_{m'}\}$ predstavlja pregrupisan skup nastao od $\{\lambda_1, \dots, \lambda_m\}$, gde je $\{1', \dots, m'\}$ permutacija $\{1, 2, \dots, m\}$. Jednostavno je uočiti da je S maksimalno kada je $\{\lambda_{m'}\}$ uređeno u opadajućem poretku ($\lambda_{1'} > \dots > \lambda_{m'}$), a minimalno kada je $\lambda_{1'} < \dots < \lambda_{m'}$.

Broket [Brockett, 1991a], [Brockett, 1991b] je generalizovao ovu ideju za slučaj matričnog računa. Neka je $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_m]$ ortogonalna matrica čije kolone zadovoljavaju sledeću jednačinu

$$\mathbf{v}_i^T \mathbf{v}_j = \delta_{ij} \quad (\mathbf{V}^T \mathbf{V} = \mathbf{I}_m). \quad (4.2)$$

Definišimo $J(\mathbf{V})$ kao

$$J(\mathbf{V}) = \text{tr}(\mathbf{D}\mathbf{V}^T\mathbf{C}\mathbf{V}) = \text{tr}(\mathbf{V}\mathbf{D}\mathbf{V}^T\mathbf{C}) \quad (4.3)$$

gde je $\mathbf{D} = \text{diag}(d_1, \dots, d_m)$. Ako se \mathbf{V} sastoji od m sopstvenih vektora matrice \mathbf{C} , tj. $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_m]$, tada je

$$\mathbf{V}^T \mathbf{C} \mathbf{V} = \text{diag}(\lambda_{1'}, \dots, \lambda_{m'}) \quad (4.4)$$

i jednačina (4.3) se svodi na $J(\mathbf{V}) = \sum d_i \lambda_{i'}$. U opštem slučaju kada je \mathbf{V} ortogonalna matrica $\mathbf{V}^T \mathbf{C} \mathbf{V}$ nije dijagonalna matrica. Međutim, i u tom slučaju važi sledeće tvrđenje.

Tvrđenje 4.1. Funkcija cene $J(\mathbf{V})$ je maksimalna kada je

$$\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_m], \quad (4.5)$$

a minimalno kada je

$$\mathbf{V} = [\mathbf{v}_m, \dots, \mathbf{v}_1], \quad (4.6)$$

ako sopstvene vrednosti zadovoljavaju $\lambda_1 > \dots > \lambda_m$. $J(\mathbf{V})$ nema ni lokalnih minimuma ni lokalnih maksimuma izuzev globalnih.

Razmotrimo sada slučaj kada je $d_1 > \dots > d_n > d_{n+1} = \dots = 0$. U tom slučaju

$$\begin{aligned} \mathbf{VDV}^T &= [\mathbf{v}_1, \mathbf{K}, \mathbf{v}_n, \mathbf{v}_{n+1}, \mathbf{K}, \mathbf{v}_m] \begin{bmatrix} d_1 & & & & & \\ & \mathbf{O} & & & & \\ & & d_n & & & \\ & & & & & \\ & & & & & \\ & & & & & 0 \end{bmatrix} [\mathbf{v}_1, \mathbf{K}, \mathbf{v}_n, \mathbf{v}_{n+1}, \mathbf{K}, \mathbf{v}_m]^T \\ &= [\mathbf{v}_1, \mathbf{K}, \mathbf{v}_n] \begin{bmatrix} d_1 & & & \\ & \mathbf{O} & & \\ & & d_n & \\ & & & \end{bmatrix} [\mathbf{v}_1, \mathbf{K}, \mathbf{v}_n]^T, \end{aligned} \tag{4.7}$$

tako da zadnjih $m-n$ kolona \mathbf{V} automatski iščezavaju. Uzmimo n proizvoljnih međusobno ortogonalnih jediničnih vektora $\mathbf{w}_1, \dots, \mathbf{w}_n$ koji su predstavljeni sledećom matricom

$$\mathbf{W} = [\mathbf{w}_1, \mathbf{K}, \mathbf{w}_n]. \tag{4.8}$$

Tada imamo

$$J(\mathbf{W}) = \text{tr}(\mathbf{D}\mathbf{W}^T\mathbf{C}\mathbf{W}) = \text{tr}(\mathbf{W}\mathbf{D}\mathbf{W}^T\mathbf{C}), \tag{4.9}$$

koja je definisana na isti način kao i $J(\mathbf{V})$. Odmah je jasno da je $J(\mathbf{W})$ maksimalno (minimalno) kada se \mathbf{W} sastoji od sopstvenih vektora koji odgovaraju n najvećih (najmanjih) sopstvenih vrednosti, pri čemu ne postoje lokalni minimumi i maksimumi osim globalnih. Znači, predložena funkcija cene je primenjiva i na PCA i na MCA, a takođe i na PSA i MSA.

Kada je $n=1$, $J(\mathbf{w})$ je

$$J(\mathbf{w}) = \mathbf{w}^T\mathbf{C}\mathbf{w} \tag{4.10}$$

pri čemu je $d_1 = 1$ i $\mathbf{w}^T\mathbf{w} = 1$.

Kada su $d_1 = d_2 = \dots = d_n = d$, $J(\mathbf{W})$ je maksimalno (minimalno) kada se \mathbf{W} sastoji od vektora koji predstavljaju linearne kombinacije sopstvenih vektora koji odgovaraju n najvećih

(najmanjih) sopstvenih vrednosti. Ne postoje drugi lokalni maksimumi i minimumi. U ovom slučaju, maksimizacijom ili minimizacijom funkcije cene se može identifikovati samo odgovarajući potprostor, ali se ne mogu odrediti sopstveni vektori. Takođe, postoji i drugi slučajevi kada je dovoljno izvršiti identifikaciju glavnog potprostora, a nije neophodno pronaći sopstvene vektore. U tom slučaju se primenjuju PSA algoritmi. Primer za to je potprostorni algoritam (*Subspace Learning Algorithm* - SLA) [Oja, 1989], koji ćemo ovde prikazati detaljnije.

Potprostorni kriterijum je definisan sledećom matricom

$$J^{\text{SLA}}(w_1, \dots, w_n) = \frac{1}{2} \left(\sum_{i=1}^n w_i^2 \right) - \frac{1}{2} \left(\sum_{i=1}^n w_i^2 \right)^2 \quad (4.17)$$

$$w_i^T w_j = \delta_{ij}$$

U kompaktnom zapisu imamo

$$J^{\text{SLA}}(w_1, \dots, w_n) = \text{trac}(W^T C W) \quad (4.18)$$

$$W^T W = I$$

gde je W matrica čija je n -ta kolona w_i , $i=1, \dots, n$ čineći $n \times n$ jediničnu matricu. Jednačina (4.18) implicira da je matrica $P = W W^T$ ortogonalna projekcijska matrica koja zadovoljava $P^2 = P$, $P^T = P$. Saša jednačina (4.18) može biti zapisana kao

$$\text{trac}(W^T C W) = \sum_i (P x_i)^2 \quad (4.19)$$

Udeležila vidimo da je originalni problem ekvivalentan za problemom pronalaska n -dimenzionog potprostora u E^n , tako da je postizvanje kvaliteta nećine projekcije vektora x na taj potprostor maksimalno veliko.

4.2 Potprostorni algoritam

Pretpostavićemo, kao i ranije, da je \mathbf{x} slučajni vektor sa srednjom vrednošću nula i da pripada R^K . Matrica \mathbf{C} je $K \times K$ kovarijansna matrica definisana sa

$$\mathbf{C} = E\{\mathbf{x}\mathbf{x}^T\}. \tag{4.11}$$

Potprostorni kriterijum je definisan sledećom jednačinom

$$\begin{aligned} J_N^{PCA}(\mathbf{w}_1, \dots, \mathbf{w}_N) &= E\left\{ \sum_{n=1}^N y_n^2 \right\} = E\left\{ \sum_{n=1}^N (\mathbf{w}_n^T \mathbf{x})^2 \right\} \\ &= \sum_{n=1}^N \mathbf{w}_n^T \mathbf{C} \mathbf{w}_n = \max, \end{aligned} \tag{4.12}$$

$$\mathbf{w}_m^T \mathbf{w}_n = \delta_{mn}.$$

U kompaktnom zapisu imamo

$$\begin{aligned} J_N^{PCA}(\mathbf{w}_1, \dots, \mathbf{w}_N) &= \text{trace}(\mathbf{W}^T \mathbf{C} \mathbf{W}), \\ \mathbf{W}^T \mathbf{W} &= \mathbf{I}, \end{aligned} \tag{4.13}$$

gde je \mathbf{W} matrica čija je n -ta kolona \mathbf{w}_n i \mathbf{I} predstavlja $N \times N$ jediničnu matricu. Jednačina (4.13) implicira da je matrica $\mathbf{P} = \mathbf{W}\mathbf{W}^T$ ortogonalna projekciona matrica koja zadovoljava $\mathbf{P}^2 = \mathbf{P}$ i $\mathbf{P}^T = \mathbf{P}$. Sada jednačina (4.13) može biti zapisana kao

$$\text{trace}(\mathbf{W}^T \mathbf{C} \mathbf{W}) = E\left\{ \|\mathbf{P}\mathbf{x}\|^2 \right\} \tag{4.14}$$

Oдавде vidimo da je originalni problem ekvivalentan sa problemom pronalaženja N -dimenzionog potprostora u R^K , tako da je očekivanje kvadrata norme projekcije vektora \mathbf{x} na taj potprostor maksimalno veliko.

Ovaj problem je rešen u [Ogawa, 1992], i rezultat pokazuje da je rešenje potprostor koji je definisan sa N sopstvenih vektora $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_N$, koji odgovaraju N najvećih sopstvenih vrednosti matrice \mathbf{C} . Ako uvedemo sledeću oznaku

$$\mathbf{U} = (\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_N), \quad (4.15)$$

ortogonalna projekciona matrica \mathbf{P} može biti izražena kao

$$\mathbf{P} = \mathbf{U}\mathbf{U}^T. \quad (4.16)$$

To znači da matrica \mathbf{W} u kriterijumu (4.13) nije definisana na jedinstven način. Imamo da je

$$\mathbf{W} = \mathbf{U}\mathbf{R}, \quad (4.17)$$

gde je \mathbf{R} proizvoljna $N \times N$ rotaciona matrica koja zadovoljava $\mathbf{R}^{-1} = \mathbf{R}^T$.

Problem kompresije u stvari predstavlja problem rekonstrukcije \mathbf{x} sa

$$\mathbf{x}_{rec} = \mathbf{P}\mathbf{x}.$$

Pošto srednja kvadratna greška rekonstrukcije može biti predstavljena sa

$$\|\mathbf{x}_{rec} - \mathbf{x}\|^2 = \|\mathbf{x}\|^2 - \|\mathbf{P}\mathbf{x}\|^2 = \min, \quad (4.18)$$

očekivana greška se minimizira kada se $E\{\|\mathbf{P}\mathbf{x}\|^2\}$ maksimizira.

Na osnovu potprostornog kriterijuma, može se konstruisati sledeći algoritam (baziran na [Oja, 1989]):

$$\begin{aligned} \mathbf{w}_k(i+1) &= \mathbf{w}_k(i) + \gamma_i \left(y_k(i) \left(\mathbf{x}(i) - \sum_{m=1}^N y_m(i) \mathbf{w}_m(i) \right) \right), \\ y_m(i) &= \mathbf{w}_m(i)^T \mathbf{x}(i), \\ m, k &= 1, \dots, N, \end{aligned} \quad (4.19)$$

gde i označava diskretne trenutke u vremenu ($i=0, 1, \dots$) i γ_i predstavlja faktor učenja. Ovaj algoritam je poznat kao potprostorni algoritam za učenje (*Subspace Learning Algorithm - SLA*).

Ovaj algoritam i njegova implementacija na potprostornoj mreži su detaljno analizirani u literaturi [Oja, 1989; Oja, 1992]. Rezultat primene ovog algoritma je takav da čak i u slučaju kada se primeni usrednjavanje primenom tehnike stohastičke aproksimacije, algoritam ne rezultuje asimptotski stabilnim rešenjima. Težinski vektori teže ka nekoj ortonormalnoj bazi u N -dimenzionom potprostoru koji je definisan dominantnim sopstvenim vektorima kovarijansne matrice ulaznog signala, ali ta baza ne mora biti ona koju definišu sami sopstveni vektori. To je u potpunosti u skladu sa rešenjem potprostornog kriterijuma koje je dato u jednačini (4.17) [Ogawa, 1992].

U matričnom zapisu, sistem (4.19) može biti prikazan kao

$$\frac{d\mathbf{W}}{dt} = \alpha(\mathbf{x} - \mathbf{W}\mathbf{W}^T\mathbf{x})\mathbf{x}^T\mathbf{W}. \quad (4.20)$$

Ako su ulazni podaci dobijeni uzorkovanjem stacionarnog signala čija je kovarijansna matrica \mathbf{C} definisana u jednačini (4.11), tada na jednačinu (4.20) može biti primenjena stohastička aproksimacija, pri čemu dobijamo:

$$\frac{d\mathbf{W}}{dt} = \alpha(\mathbf{C}\mathbf{W} - \mathbf{W}\mathbf{W}^T\mathbf{C}\mathbf{W}). \quad (4.21)$$

Ova jednačina je analizirana u [Oja, 1989]. U [Oja, 1989] je rečeno da, pošto jednačina (4.21) predstavlja matričnu jednačinu trećeg stepena, jednačina je teška za rešavanje, iako odgovarajuća jednačina po $\mathbf{W}\mathbf{W}^T$ može biti rešena u zatvorenoj formi, pod određenim uslovima. Sada ćemo pokazati da se određeni rezultati o prirodi sistema opisanog jednačinom (4.21) mogu dobiti jednostavnom analizom jednačine, bez pokušaja rešavanja iste. Jednačina (4.21) se može pisati kao sistem jednačina:

$$\frac{d\mathbf{W}^k}{dt} = \alpha(\mathbf{C}\mathbf{W}^k - \mathbf{W}\mathbf{W}^T\mathbf{C}\mathbf{W}^k), \quad (4.22)$$

gde \mathbf{W}^k ($k=1, \dots, N$) predstavlja k -tu kolonu matrice \mathbf{W} . U tom slučaju možemo izvući sledeće zaključke:

- Stacionarne tačke jednačine (4.22) predstavljaju jednačine čija su rešenja sopstveni vektori matrice $\mathbf{W}\mathbf{W}^T$.
- Očigledno je da će sistem pronaći sopstvene vektore ($\mathbf{C}\mathbf{W}^k$) koji odgovaraju sopstvenim vrednostima koje su jednake jedinici.
- Ako je $\text{rank}(\mathbf{W})=K$, tada će matrica $\mathbf{W}\mathbf{W}^T$ imati K različitih sopstvenih vektora koji svi odgovaraju sopstvenim vrednostima 1. U tom slučaju je jednostavno zaključiti da matrica $\mathbf{W}\mathbf{W}^T$ predstavlja projekcionu matricu. Ovaj slučaj je, takođe, analiziran u [Plumbley, 1991] – isti rezultat je dobijen na drugačiji način.
- Ako je $\text{rank}(\mathbf{W})=r < K$, tada će sistem pronaći r sopstvenih vektora koji odgovaraju nezavisnim kolonama matrice \mathbf{W} . Ostalih $K-r$ vektora će odgovarati sopstvenim vrednostima koje su jednake nuli. Ponovo, možemo zaključiti da je matrica $\mathbf{W}\mathbf{W}^T$, u stvari, projekciona matrica.

Na osnovu sprovedene analize, bez pokušaja da rešimo jednačinu (4.22), možemo zaključiti da rešenje \mathbf{W} sistema (4.22), formira matricu $\mathbf{W}\mathbf{W}^T$ koja je projekciona matrica.

4.3 Gradijent funkcije cene $J(\mathbf{W})$

Skup matrica \mathbf{W} , označićemo ga sa $O_{m,n}$, takvih da $\mathbf{W} \in \mathfrak{R}^{K \times N}$ i

$$\mathbf{W}^T \mathbf{W} = \mathbf{I}_N \quad (4.23)$$

i funkcija cene $J(\mathbf{W})$ se nazivaju Stifelovim potprostorom [Edelman, 1998]. Usvojimo da je $J(\mathbf{W})$ definisano sa (4.9). Izračunaćemo sada gradijent funkcije $J(\mathbf{W})$ na skupu $O_{m,n}$. Usvojimo da je $d\mathbf{W}$ mala promena \mathbf{W} i da je odgovarajuća promena $J(\mathbf{W})$ data sa $dJ(\mathbf{W}) = J(\mathbf{W} + d\mathbf{W}) - J(\mathbf{W})$. Tada imamo

$$dJ = \text{tr}(d\mathbf{W} \mathbf{D} \mathbf{W}^T \mathbf{C}) + \text{tr}(\mathbf{W} \mathbf{D} d\mathbf{W}^T \mathbf{C}). \quad (4.24)$$

Iz (4.23) imamo

$$d\mathbf{W}^T \mathbf{W} + \mathbf{W}^T d\mathbf{W} = 0. \quad (4.25)$$

Kako je

$$dJ = \frac{\partial J}{\partial \mathbf{W}} \cdot d\mathbf{W}, \quad (4.26)$$

gde je $\mathbf{A} \cdot \mathbf{B}$ definisano sa $\sum_{i,j} a_{ij} b_{ij} = \text{tr}(\mathbf{A} \mathbf{B}^T)$ i gde je gradijent matrica koja je data sa

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{W}} &= -\mathbf{W} \mathbf{D} \mathbf{W}^T \mathbf{C} \mathbf{W} + \mathbf{C} \mathbf{W} \mathbf{D} \mathbf{W}^T \mathbf{W} \\ &= -\mathbf{W} \mathbf{D} \mathbf{W}^T \mathbf{C} \mathbf{W} + \mathbf{C} \mathbf{W} \mathbf{D}. \end{aligned} \quad (4.27)$$

Gradijentni metod za dobijanje glavnih komponentata je dat sa

$$\Delta \mathbf{W}_P = \eta (\mathbf{C} \mathbf{W}_P \mathbf{D} - \mathbf{W}_P \mathbf{D} \mathbf{W}_P^T \mathbf{C} \mathbf{W}_P) \quad (4.28)$$

dok za sporedne komponente važi

$$\Delta \mathbf{W}_M = \eta (\mathbf{C} \mathbf{W}_M \mathbf{D} - \mathbf{W}_M \mathbf{D} \mathbf{W}_M^T \mathbf{C} \mathbf{W}_M) \quad (4.29)$$

gde je $\mathbf{W}_P = \mathbf{V}_S = [\mathbf{w}_1, \dots, \mathbf{w}_N]$ i $\mathbf{W}_M = \mathbf{V}_N = [\mathbf{w}_K, \dots, \mathbf{w}_{N+1}]$.

Razmotrimo specijalan slučaj $N = 1$. Ako stavimo $d_1 = 1$ jednačina (4.28) se svodi na

$$\Delta \mathbf{w} = \eta (\mathbf{C} \mathbf{w} - \mathbf{w} \mathbf{w}^T \mathbf{C} \mathbf{w}), \quad (4.30)$$

što predstavlja poznati algoritam čija je *on-line* verzija data kao

$$\Delta \mathbf{w} = \eta (y \mathbf{x} - y^2 \mathbf{w}). \quad (4.31)$$

Jasno je da je to algoritam koji je predložio Oja [Oja 1982].

U algoritmu koji je predložio Brocket [Brocket, 1991a; Brocket, 1991b] mora biti $N=K$, dok u algoritmu predloženom od strane Ksua mora biti $N < K$. Ako se u tim algoritmima stavi $d_1 = d_2 = \dots = d_N = 1$, dobijamo PSA algoritme.

Nekoliko najpoznatijih algoritama za paralelno izračunavanje PCA/PSA je prikazano u Tabeli 4.1. Tabela 4.2 sadrži nekoliko MCA/MSA algoritama.

Tabela 4.1 Adaptivni PSA-PCA algoritmi

Skraćenica	Algoritam za učenje	Referene	Tip
PLA1.	$\Delta \mathbf{W} = \eta (\mathbf{x} \mathbf{y}^T - \mathbf{W} \mathbf{y} \mathbf{y}^T)$	[Oja, 1985]	PSA
PLA2.	$\Delta \mathbf{W} = \eta (\mathbf{x} \mathbf{y}^T \mathbf{D} - \mathbf{W} \mathbf{D} \mathbf{y} \mathbf{y}^T \mathbf{D})$	[Brocket, 1991a]	PSA/PCA
PLA3.	$\Delta \mathbf{W} = \eta (\mathbf{x} \mathbf{y}^T - \mathbf{W} \mathbf{y} \mathbf{y}^T \mathbf{D})$	[Oja, 1992b]	PCA
PLA4.	$\Delta \mathbf{W} = \eta (\mathbf{x} \mathbf{y}^T \mathbf{W}^T \mathbf{W} - \mathbf{W} \mathbf{y} \mathbf{y}^T)$	[Chen, 1998]	PSA
PLA5.	$\Delta \mathbf{W} = \eta (\mathbf{x} \mathbf{y}^T \mathbf{D} \mathbf{W}^T \mathbf{W} - \mathbf{W} \mathbf{D} \mathbf{y} \mathbf{y}^T)$	[Chen, 2001]	PSA/PCA
PLA6.	$\Delta \mathbf{W} = \eta \mathbf{x} \mathbf{g} (\mathbf{y}^T) + \mu \mathbf{W} (\mathbf{I} - \mathbf{W}^T \mathbf{W})$	[Wang, 1996]	PCA

PLA1 u tabeli predstavlja već opisani PSA algoritam, SLA, koji je baziran na potprostornom kriterijumu. Jedan način da se taj algoritam pretvori u PCA algoritam je predložen od strane Broketa [Brocket, 1991a]. Ako je \mathbf{D} dijagonalna matrica sa pozitivnim striktno opadajućim dijagonalnim elementima algoritam će predstavljati PCA – ako su svi dijagonalni elementi isti i jednaki 1, algoritam u stvari koincidira sa SLA (kao što je već naglašeno, Broketov algoritam zahteva jednak broj ulaznih i izlaznih neurona). Drugi način da se formira PCA algoritam od SLA je predložen od strane Oje, Ogave and Wangviwatane [Oja, 1992b] – baziran je na težinskom potprostornom kriterijumu. Ovaj algoritam nije homogen sa aspekta pojedinačnog neurona. U cilju prevazilaženja nehomogenosti algoritma, isti autori su predložili uvođenje nelineranosti u originalni SLA [Oja, 1992c]. PLA4 predstavlja algoritam koji su predložili Čen i Amari [Chen, 1998]. Taj algoritam predstavlja PSA algoritam koji može biti shvaćen kao derivativ SLA. Ekstenzija ovog algoritma u PCA algoritam je predstavljena sa PLA5 (mora se naglasiti da u slučaju da \mathbf{D} predstavlja dijagonalnu matricu sa pozitivnim striktno opadajućim elementima algoritam će rezultovati PCA algoritmom – ako su svi članovi isti, algoritam koincidira sa PLA4). Ponovo ćemo naglasiti da ni ovaj algoritam nije homogen sa aspekta pojedinačnog neurona. PLA6 je PCA metod kod koga se zakon učenja sastoji od dva dela – prvi član zakona učenja usmerava sve težinske vektore ka dominantnom sopstvenom vektoru kovarijanske matrice, dok drugi član predstavlja kazneni član koji “zahteva” da vektori budu međusobno ortonormalni.

MLA1 u Tabeli 4.2, predstavlja ekstenziju PLA1 algoritma, koja može biti korišćena za MSA izračunavanja. Međutim, u literaturi [Yan, 1994] je pokazano da norma svakog vektora kolone matrice \mathbf{W} divergira sa vremenom. MLA2 predstavlja modifikaciju MLA1 koja je bazirana na uvođenju algoritma za učenje koji primenjuje prirodni gradijent [Chen, 1998]. U [Chen, 1998] je pokazano, sa čisto numeričke tačke gledišta, da algoritam “pati” od

Tabela 4.2 Adaptivni paralelni MSA/MCA algoritmi

Skr.	Zakon učenja	Reference	Tip
MLA1.	$\Delta \mathbf{W} = -\eta(\mathbf{x} - \mathbf{W}\mathbf{y})\mathbf{y}^T$	[Oja, 1985]	MSA
MLA2.	$\Delta \mathbf{W} = -\eta(\mathbf{x}\mathbf{y}^T \mathbf{W}^T \mathbf{W} - \mathbf{W}\mathbf{y}\mathbf{y}^T)$	[Chen, 1998]	MSA
MLA3.	$\Delta \mathbf{W} = -\eta(\mathbf{x}\mathbf{y}^T \mathbf{D}\mathbf{W}^T \mathbf{W} - \mathbf{W}\mathbf{D}\mathbf{y}\mathbf{y}^T + \mathbf{W}(\mathbf{D} - \mathbf{W}^T \mathbf{D}\mathbf{W}))$	[Chen, 2001]	MSA/ MCA
MLA4.	$\Delta \mathbf{W} = -\eta(\mathbf{x}\mathbf{y}^T \mathbf{W}^T \mathbf{W}\mathbf{W}^T \mathbf{W} - \mathbf{W}\mathbf{y}\mathbf{y}^T)$	[Douglas, 1998]	MSA
MLA5.	$\mathbf{z} = \mathbf{W}\mathbf{y}, \mathbf{p} = \mathbf{z} - \mathbf{x}, \phi = \frac{1}{\sqrt{1 + \eta^2 \ \mathbf{p}\ ^2 \ \mathbf{y}\ ^2}}, \tau = \frac{\phi - 1}{\ \mathbf{y}\ ^2},$ $\bar{\mathbf{p}} = -\tau \mathbf{z} / \eta + \phi \mathbf{p}, \mathbf{u} = \frac{\bar{\mathbf{p}}}{\ \bar{\mathbf{p}}\ ^2}, \mathbf{v} = \mathbf{W}^T \mathbf{u}, \Delta \mathbf{W} = -2\mathbf{u}\mathbf{v}^T$	[Abed- Meraim, 2000]	MSA
MLA6.	$\mathbf{F} = -4\kappa \mathbf{x}\mathbf{y}^T, \mathbf{P} = -\mu \mathbf{G}\mathbf{W}, \Delta \mathbf{G} = \frac{\eta}{2} \text{sk}[(\mathbf{F} + \mathbf{P})\mathbf{W}^T]$ $\Delta \mathbf{W} = -\eta \mathbf{G}\mathbf{W}$	[Fiori 2002]	MSA
MLA7.	$\Delta \mathbf{W} = -\eta \mathbf{x}\mathbf{g}(\mathbf{y}^T)_+ + \mu \mathbf{W}(\mathbf{I} - \mathbf{W}^T \mathbf{W})$	[Wang, 1996]	MCA

marginalne nestabilnosti koja se može prevazići povremenom normalizacijom kolona matrice \mathbf{W} . Kao heurističko poboljšanje MLA2 algoritma Douglas i ostali [Douglas, 1998] su predložili algoritam MLA4. Jedina razlika između algoritama MLA2 i MLA4 je prisustvo $(\mathbf{W}^T \mathbf{W})^2$ umesto $\mathbf{W}^T \mathbf{W}$ u prvom članu na desnoj strani jednačine, što ima za cilj da se pojača uticaj devijacije \mathbf{W} od ortonormalnosti na zakon učenja. MLA3 predstavlja MCA metod koji je nastao modifikacijom MLA2. MLA3 će predstavljati MCA algoritam u slučaju da su dijagonalni članovi matrice \mathbf{D} pozitivni i u striktno opadajućem rasporedu. Ukoliko su svi članovi isti algoritam predstavlja MSA algoritam. MLA5 metod je izveden iz algoritma MLA1 pri čemu je na specifičan način izvedena ortogonalizacija matrice \mathbf{W} . MLA6

predstavlja nedavno predloženi MSA algoritam koji se bazira na analogiji sa sistemom jednačina koji opisuje dinamiku apstraktnog krutog tela koje se nalazi u polju sila [Fiori, 2002]. MLA7 predstavlja varijantu PLA6 koja se koristi za MCA.

$$\frac{dW(t)}{dt} = -\gamma(CWD - WDW^T CW) \quad (4.28)$$

$$\frac{dW(t)}{dt} = -\gamma(CWD - WDW^T CW) \quad (4.29)$$

pri čemu su sufiksi P i M zanemareni.

Uvojećemo da je W matrica dimenzije $K \times N$. Uvojećimo nakoda

$$U(t) = W^T(t)W(t) \quad (4.34)$$

Sada ćemo pokazati da je $U(t)$ invarijantna pri zadatom početnom uslovu i u slučaju da $W(t)$ zadovoljava (4.32) ili (4.33), odnosno

$$\frac{dU}{dt} = 0 \quad (4.35)$$

Ovo se može dokazati direktnim izračunavanjem,

$$\begin{aligned} \frac{dU(t)}{dt} &= \frac{dW^T}{dt} W + W^T \frac{dW}{dt} \\ &= -\gamma(-W^T W) - \gamma(W^T W) = 0 \end{aligned} \quad (4.36)$$

pri čemu je inicijalni uslov: $U(0) = W^T(0)W(0) = I$. Znači, ako $W(0)$ u trenutku $t=0$ pripada Stiefelovoj podmnožici $O_{K \times N}$, tada je $U(0) = I$ i $U(t) = I$ o bilo kom trenutku t , što implicira da $W(t)$ uvek pripada $O_{K \times N}$. Kako globalni sistem ima N glavnih ili specijalnih komponenta, dinamičke jednačine (4.28) i (4.29) bi trebalo da konvergiraju ka tom rešenju, pri čemu $W(t)$ uvek pripada $O_{K \times N}$.

4.4 Analiza stabilnosti predloženog algoritma

U ovom paragrafu ćemo se pozabaviti analizom stabilnosti (4.28) i (4.29). Primenom stohastičke aproksimacije [Ljung, 1977] algoritmima (4.28) i (4.29) mogu biti pridružene sledeće diferencijalne jednačine

$$\frac{d\mathbf{W}(t)}{dt} = \gamma(\mathbf{C}\mathbf{W}\mathbf{D} - \mathbf{W}\mathbf{D}\mathbf{W}^T\mathbf{C}\mathbf{W}) \quad (4.32)$$

$$\frac{d\mathbf{W}(t)}{dt} = -\gamma(\mathbf{C}\mathbf{W}\mathbf{D} - \mathbf{W}\mathbf{D}\mathbf{W}^T\mathbf{C}\mathbf{W}) \quad (4.33)$$

pri čemu su sufiksi P i M zanemareni.

Usvojicemo da je \mathbf{W} matrica dimenzije $K \times N$. Usvojimo takođe

$$\mathbf{U}(t) = \mathbf{W}^T(t)\mathbf{W}(t). \quad (4.34)$$

Sada ćemo pokazati da je $\mathbf{U}(t)$ invarijantna pri zadatom početnom uslovu i u slučaju da $\mathbf{W}(t)$ zadovoljava (4.32) ili (4.33), odnosno

$$\frac{d\mathbf{U}}{dt} = 0. \quad (4.35)$$

Ovo se može dokazati direktnim izračunavanjem,

$$\begin{aligned} \frac{d\mathbf{U}(t)}{dt} &= \frac{d\mathbf{W}^T}{dt}\mathbf{W} + \mathbf{W}^T\frac{d\mathbf{W}}{dt} \\ &= \gamma(\mathbf{I} - \mathbf{W}^T\mathbf{W})\left(\mathbf{D}\mathbf{W}^T\mathbf{C}\mathbf{W} + \mathbf{W}^T\mathbf{C}\mathbf{W}\mathbf{D}\right) = 0 \end{aligned} \quad (4.36)$$

pri čemu je inicijalni uslov: $\mathbf{U}(0) = \mathbf{W}^T(0)\mathbf{W}(0) = \mathbf{I}$. Znači, ako $\mathbf{W}(0)$ u trenutku $t=0$ pripada Stifelovoj podmnogostrukosti $O_{m,n}$, tada je $\mathbf{U}(0) = \mathbf{I}$ i $\mathbf{U}(t) = \mathbf{I}$ u bilo kom trenutku t , što implicira da $\mathbf{W}(t)$ uvek pripada $O_{m,n}$. Kako globalni ekstremum daje N glavnih ili sporednih komponentata, dinamičke jednačine (4.28) i (4.29) bi trebale da konvergiraju ka tom rešenju, pri čemu $\mathbf{W}(t)$ uvek pripada $O_{m,n}$.

Međutim, kompjuterske simulacije su pokazale da primena algoritma (4.29) ne rezultuje ekstrakcijom sporednih komponenata iako je dinamika koje definiše (4.29) stabilna na $O_{m,n}$. U slučaju kada se algoritam proširi na ceo prostor $R^{K \times N}$ pojavljuju se problemi. Drugim rečima, kada $\mathbf{W}(t)$ počne da napušta $O_{m,n}$ usled šuma ili numeričke greške, devijacija će rasti i $\mathbf{W}(t)$ definitivno napušta $O_{m,n}$.

Kada $\mathbf{W}(t)$ odstupi ka $\mathbf{W}(t) + \delta\mathbf{W}(t)$, dinamika promene $\delta\mathbf{W}$ je definisana sledećom jednačinom:

$$\begin{aligned} \frac{d}{dt} \delta \mathbf{W}(t) &= \pm \gamma \delta (\mathbf{C} \mathbf{W} \mathbf{D} - \mathbf{W} \mathbf{D} \mathbf{W}^T \mathbf{C} \mathbf{W}) \\ &= \pm \gamma (\mathbf{C} \delta \mathbf{W} \mathbf{D} - \delta \mathbf{W} \mathbf{D} \mathbf{W}^T \mathbf{C} \mathbf{W} - \mathbf{W} \mathbf{D} \delta \mathbf{W}^T \mathbf{C} \mathbf{W}). \end{aligned} \quad (4.37)$$

Ova jednačina je analizirana od strane Čena i Amarija [Chen, 2001] kao i Čena i drugih [Chen, 1998] i rezultat je da je gornja jednačina stabilna u slučaju kada je gradijentna dinamika definisana jednačinom (4.32), a nestabilna kada je dinamika definisana jednačinom (4.33). Da bi jednačina bila stabilna u slučaju kada je dinamika definisana jednačinom (4.33) mora se stalno vršiti ortonormalizacija kolona \mathbf{W} .

Kako se ovaj problem može prevazići je dalje analizirano od strane Čena i Čena i Amarija u [Chen, 2001], [Chen, 1998].

Odavde je jasno da je od velike važnosti analizirati stabilnost pojedinačnih algoritama u slučaju da se definiše klasa algoritama koja zadovoljava neku kriterijumsku funkciju.

V Poglavlje

Biološki verovatan zakon učenja za izračunavanje

dominantne glavne komponente

U ovom i naredna dva poglavlja će biti prikazan originalni doprinos ove teze.

U ovom poglavlju će biti predstavljen ∞ OH metod koji predstavlja biološki verovatan model za izračunavanje dominantne glavne komponente kovarijanske matrice ulaznog signala. Biće pokazano da primena originalnog Hebovog zakona učenja ne dovodi do divergencije norme sinaptičkog vektora zahvaljujući usvojenoj strukturi mreže i usvojenom zakonu učenja. Takođe, biće objašnjeno zašto se ovakav zakon učenja može smatrati biološki verovatnim.

U VI poglavlju će biti analiziran biološki verovatna metoda MH (i njen derivat MHO) za određivanje dominantnog potprostora kovarijanske matrice ulaznog signala. Struktura mreže je inspirisana organizacijom dela retine kod riba. Mreža sadrži samo mali broj globalnih izračunavanja.

Sedmo poglavlje se bavi vremenski orijentisanim hijerarhijskim metodom (TOHM) koji predstavlja prvi generalni metod za transformaciju PSA/MSA algoritama u PCA/MCA algoritme. Biće prikazana glavna ideja koja predstavlja osnovu metode, jednostavna matematička analiza u opštem slučaju kao i detaljna analiza primene na jedan konkretan MSA i jedan konkretan PSA algoritam.

5.1 Jednostavan biološki verovatan ∞ OH model neurona

U ovom paragrafu će biti predstavljen novi pristup u formiranju jednoslojne samoorganizujuće neuralne mreže. Biće prikazan algoritam za učenje bez nadzora koji je baziran na Hebovom zakonu učenja. Biće analiziran jednostavan dinamički model neurona koji ima i uobičajene *feed-forward* ali i povratne veze između ulaza i izlaza. U stvari, predloženi algoritam se najkorektnije može nazvati samonadzirućim. Predloženo rešenje omogućava direktnu primenu Hebovog zakona bez ikakvih dodatnih stabilizacionih članova. To je omogućeno kako strukturom mreže tako i zakonom učenja. Predloženi zakon učenja za modifikaciju jačine sinapsi je proporcionalan pre-sinaptičkoj i usrednjennoj postsinaptičkoj aktivnosti. Šta se podrazumeva pod usrednjenom postsinaptičkom aktivnošću će biti jasnije iz predstojećeg teksta. Biće pokazano da je model sposoban da izdvoji dominantnu glavnu komponentu iz stacionarne ulazne vektorske sekvence. Primenjeni zakon učenja, u stvari, predstavlja originalni Hebov zakon učenja bez ikakvih dodatnih stabilizacionih članova – stabilizaciju obezbeđuje usvojena struktura mreže.

Kao što je već rečeno ranije, iako je teško očekivati da će do kompletnog razumevanja principa rada bioloških neuronskih mreža doći vrlo uskoro, može se očekivati da će biti moguće razumevanje pojedinih osnovnih funkcija na osnovu rezultata bioloških istraživanja. Osnovno pitanje koje se postavlja u slučaju takvog pristupa problemu je postavljeno od strane Linskera [Linsker,1988]:

“Na koji način se perceptivni sistem živih bića razvije tako da prepozna specifične osobine svog okruženja, a da mu pri tom nije rečeno koje karakteristike treba da analizira ili čak da li je identifikacija tih karakteristika korektna?”

Ovde će biti sugerisano da neki zakoni učenja za koje se smatra da pripadaju grupi zakona za učenje bez nadzora mogu biti aproksimirani samonadzirućim Hebovim zakonom učenja. Biće pokazano da metod koji se ovde predlaže može biti modifikovan tako da bude ekvivalentan sa Kohonenovim [Kohonen, 1982] i Ojinim [Oja, 1982] zakonima učenja za stabilizaciju originalnog Hebovog zakona učenja.

Analizirajmo još jednom deo Linskerovog pitanja u kome se kaže da “neuralnoj mreži nije rečeno koje karakteristike treba da analizira”. Izgleda sasvim prirodno da, na neki način, postoji specifikacija zadataka koje neuralna mreža treba da obavi i da je ekstrakcija specifičnih karakteristika posledica ostvarivanja tih zadataka. Ideja koja je osnova za formiranje metoda koji se predlaže je da neuralna mreža pokušava da neutralizuje izvor poremećaja koji se dešava na njenom ulazu.

Imajući u vidu originalni Hebov zakon učenja kao i modifikaciju koju je predložio Oja a koja dovodi do njegove stabilizacije, postoje tri interesantna aspekta koja se mogu dalje analizirati:

1. Većina radova, koji su objavljeni u poslednje vreme, a koji se bave stabilizacijom originalnog Hebovog zakona učenja su fokusirani na pronalaženje stabilizacionih članova u zakonu učenja pri čemu se smatra da je struktura mreže fiksna i da se sastoji samo od direktnih veza (*feed-forward*) između ulaznih i izlaznih neurona. U slučaju primene simetrične korekcije greške (SEC) (en. *Symmetrical Error Correction*) [Saund, 1995] postoje takođe i povratne veze. Međutim, potpuno je nejasno kako povratne veze utiču samo na vrednost sinapsi, a nemaju nikakav uticaj na potencijal izlaznih neurona kroz direktne veze. Jedna ideja za stabilizaciju koja je predložena od strane Fajfea i Bedlija [Fyfe, 1995a; Fyfe, 1995b] zahteva uvođenje takozvanih posrednih neurona. Drugi koji su davali određena rešenja u tom pravcu su Dajan i drugi [Dayan, 1995], Harpur [Harpur, 1997], Ksu [Xu, 1993] i Oujang i drugi [Ouyang, 2000]. Iz ovoga se može zaključiti da je opravdano pokušati stabilizaciju originalnog Hebovog zakona učenja otkrivanjem pogodne strukture neuralne mreže na kojoj se zakon učenja primenjuje.
2. Kao što je već ranije objašnjeno, generalno govoreći, potencijal izlaznog neurona je nelinearna funkcija ulaznih potencijala. Vizuelni neurofiziolozi su pokazali da je za veliki broj nervnih ćelija, aproksimacija odziva linearnom sumacijom trenutne vrednosti ulaznih neurona, zadovoljavajuća. (To, se naravno ne odnosi na oblast zasićenja koja postoji kod svake realne nervne ćelije.) Interesantan je sada sledeći aspekt: poznato je da se hemijski neurotransmiteri oslobađaju na sinapsama ili u područjima koja su blizu kontakata sa drugim ćelijama. Intuitivno, moglo bi se očekivati da prijemna ćelija tokom vremena obavlja nekakvu vrstu integracije neurotransmitera koje prima. Nekoliko autora kao na primer Bajnenštok i drugi (poznati Bajnenštok-Kuper-Monro (BCM) zakon učenja [Bienenstock, 1982]), Foldiak [Foldiak, 1992] i Harpur [Harpur, 1997] su za modifikaciju jačine sinapsi koristili, pored trenutne vrednosti presinaptičkog

signala i promenljivu, vremenski usrednjenu, vrednost postsinaptičkog signala. To ukazuje na mogućnost da se uključivanjem sporo promenljivog vremenski usrednjelog signala u osnovni Hebov zakon može doprineti njegovoj stabilizaciji.

3. Kao što je već rečeno, samoorganizujuće neuralne mreže pokušavaju da se organizuju na takav način da obezbede detekciju korisnih karakteristika ulaznog signala, bez ikakvog ili sa samo malo znanja o željenom rezultatu koji treba postići. U slučaju koji ćemo sada razmatrati smatraćemo da mreža ima jedan dodatni cilj – da potisne izvor promene na svom ulazu. U tom slučaju prisustvo povratnih veza u modelu je neizbežno.

Na osnovu gore navednog usvojicemo novu strukturu neurona (Slika 5.1) koja će dalje biti analizirana.

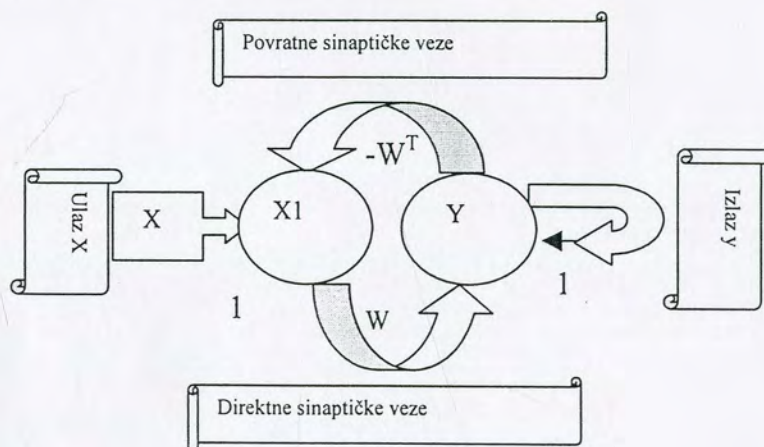
Novi zakon učenja je dat sa

$$\Delta \mathbf{w} \propto \mathbf{y} \cdot \mathbf{x1}^T, \tag{5.1}$$

pri čemu je, za razliku od ostalih delova teksta, \mathbf{w} red vektor i

$$y(i+1) = y(i) + \mathbf{w}(i) \cdot (\mathbf{x}(i+1) - \mathbf{w}(i)^T \cdot y(i)), \tag{5.2}$$

a $\mathbf{x1}$ je definisano sa $\mathbf{x1} = (\mathbf{x} - \mathbf{w}^T y)$. Veoma je lako videti da predloženi zakon učenja predstavlja direktnu implementaciju Hebovog zakona učenja.



Slika 5.1 Predložena struktura mreže

Sada će biti dokazano da predloženi algoritam, primenjen na predloženoj strukturi neuralne mreže, izdvaja dominantni sopstveni vektor ulazne kovarijanske matrice \mathbf{R}_x , i da izlazna ćelija predstavlja skoro linearnu kombinaciju ulaznih signala.

Predloženi metod je nazvan ∞ OH iz sledećih razloga:

znak za beskonačnost (∞) je korišćen iz dva razloga: grafički on veoma liči na strukturu usvojene mreže i, takođe, on ukazuje na dinamički karakter mreže.

O je skraćenica od Oja – očigledno je da je na neki način Ojin stabilizacioni član prisutan u predloženom zakonu učenja. Kasnije će biti pokazano kako se predloženi zakon učenja može transformisati u Ojin zakon učenja.

H je skraćenica od Heb – predloženi zakon učenja u stvari predstavlja Hebov tip učenja.

Ove jednačine se mogu napisati u sledećoj formi:

$$w_i(t+1) = w_i(t) + \eta(t) \cdot (x_i(t) - y(t)) \cdot x_i(t) \quad (5.3)$$

$$y(t+1) = A(w(t)) \cdot x(t) = \Delta(w(t)) \cdot x(t+1) \quad (5.4)$$

$$Q(w(t), x(t+1)) = x(t+1) - (x(t+1) - w(t)^T x(t+1))^2 \quad (5.5)$$

$$A(w(t)) = 1 - w(t) \cdot w(t)^T \quad (5.6)$$

$$B(w(t)) = \eta(t) \quad (5.7)$$

Nije teško uočiti da jednačine (5.3) i (5.4) predstavljaju specijalan slučaj algoritma koji je analizirao Ljung [Ljung, 1977]. Ako za jednačine (5.5) i (5.6) važe sledeće pretpostavke:

- A1: $x(t)$ je stacionarna sekvencija slučajnih promenljivih
- A2: $x(t)$ je ograničeno sa verovatnoćom 1.
- A3: Funkcija $Q(w, x)$ je konvexna diferencijabilna i za $w \neq 0$ i za $w \in D_x(t)$ je otvoren povezan podskup D_w gde se D_w definiše sa $D_w = \{w \mid A(w) \text{ ima nula}$

5.2 Matematička analiza ∞ OH algoritma

Predloženi zakon učenja, relacije (5.1) i (5.2), može biti opisan sledećim jednačinama

$$\mathbf{w}(i+1) = \mathbf{w}(i) + \gamma(i+1) \cdot y(i+1) \cdot (\mathbf{x}(i+1) - \mathbf{w}(i))^T \cdot y(i+1))^T, \quad (5.3)$$

$$y(i+1) = y(i) + \mathbf{w}(i) \cdot (\mathbf{x}(i+1) - \mathbf{w}(i))^T \cdot y(i). \quad (5.4)$$

Ove jednačine se mogu napisati i u sledećoj formi:

$$\mathbf{w}(i+1) = \mathbf{w}(i) + \gamma(i+1) \cdot Q(\mathbf{w}(i), \mathbf{x}(i+1)) \quad (5.5)$$

$$y(i+1) = A(\mathbf{w}(i)) \cdot y(i) + B(\mathbf{w}(i)) \cdot \mathbf{x}(i+1), \quad (5.6)$$

gde je

$$Q(\mathbf{w}(i), \mathbf{x}(i+1)) = y(i+1) \cdot (\mathbf{x}(i+1) - \mathbf{w}(i))^T \cdot y(i+1))^T \quad (5.7)$$

$$A(\mathbf{w}(i)) = 1 - \mathbf{w}(i) \cdot \mathbf{w}(i)^T \quad (5.8)$$

$$B(\mathbf{w}(i)) = \mathbf{w}(i). \quad (5.9)$$

Nije teško uočiti da jednačine (5.3) i (5.4) predstavljaju specijalan slučaj algoritma koji je analizirao Ljung [Ljung, 1977]. Ako za jednačine (5.5) i (5.6) važe sledeće pretpostavke:

- A1: $\mathbf{x}(i)$ je stacionarna sekvenca slučajnih promenljivih.
- A2: $\mathbf{x}(i)$ je ograničeno sa verovatnoćom 1.
- A3: Funkcija $Q(\mathbf{w}, \mathbf{x})$ je kontinualno diferencijabilna i po \mathbf{w} i po \mathbf{x} za $\mathbf{w} \in D_R$, (D_R je otvoren povezan podskup D_S , gde je D_S definisano sa $D_S = \{\mathbf{w} \mid A(\mathbf{w})$ ima sve

sopstvene vrednosti striktno unutar jediničnog kruga}). Izvodi su, za fiksno \mathbf{w} i \mathbf{x} , ograničeni u vremenu.

A4: Matrične funkcije $A()$ i $B()$ su kontinualne u Lipšicovom smislu u D_R .

A5: $\lim_{t \rightarrow \infty} EQ(\mathbf{w}, \mathbf{x})$ postoji za svako $\mathbf{w} \in D_R$ i označen je sa $h(\mathbf{w})$. Očekivanje je po \mathbf{x} .

A6: $\gamma(k)$ je niz pozitivnih realnih brojeva takvih da važi $\gamma \rightarrow 0$, $\sum_t \gamma(t)^p < \infty$ za neko p , i $\sum_t \gamma(t) = \infty$.

A7: T je lokalno asimptotski stabilan skup rešenja za diferencijalnu jednačinu

$$\frac{d\mathbf{w}}{dt} = \dot{\mathbf{w}} = h(\mathbf{w}), \quad (5.10)$$

sa domenom privlačenja $D\{T\}$, i

A8: $\mathbf{w}(t)$ posećuje kompaktan podskup $O \subset D\{T\}$ beskonačno često sa verovatnoćom 1.

Pod pretpostavkom da su ispunjene pretpostavke A1:A8 ispunjene može se primeniti Teorema 1 iz [Ljung, 1977]. Tada sa verovatnoćom 1

$$\lim_{t \rightarrow \infty} \mathbf{w}(t) \in T,$$

gde je $\mathbf{w}(t)$ definisano u (5.10). Ta teorema nam omogućava proučavanje ponašanja kao i pronalaženje stacionarnog stanja diferencne jednačine na osnovu analize pridružene diferencijalne jednačine.

Da bi zadovoljili pretpostavku A6 u teoremi usvojicemo da je $\gamma(i)=1/i$ (u tom slučaju, $p=2$ takođe zadovoljava A6). Takođe, nije teško uočiti da je D_S definisano kao $D_S(\mathbf{w}) = \{\mathbf{w} \mid 0 < \mathbf{w} \cdot \mathbf{w}^T < 2\}$.

Sledeći korak je izračunavanje $h(\mathbf{w})$ (za $\mathbf{w} \in D_R$). Na osnovu (5.6) sledi da je

$$y(i+1) = \left[\prod_{r=0}^{i+1} A(\mathbf{w}(r)) \right] y(0) + \sum_{j=1}^{i+1} \left[\prod_{l=j+1}^{i+1} A(\mathbf{w}(l-1)) \right] B(\mathbf{w}(j-1)) \mathbf{x}(j), \quad (5.11)$$

ili (ili ako usvojimo $y(0) = 0$ i $\mathbf{x}(0) = 0$)

$$y(i+1) = \sum_{j=1}^{i+1} \left[\prod_{r=j+1}^{i+1} A(\mathbf{w}(r-1)) \right] \cdot B(\mathbf{w}(j-1)) \cdot \mathbf{x}(j). \quad (5.12)$$

Sada $h(\mathbf{w})$ može biti izračunato kao:

$$\begin{aligned}
 h(\mathbf{w}) &= \lim_{t \rightarrow \infty} EQ(\mathbf{w}, \mathbf{x}) \\
 &= \lim_{i \rightarrow \infty} E \left(\sum_{j=1}^{i+1} \left(\prod_{r=j+1}^{i+1} A(\mathbf{w}) \right) B(\mathbf{w}) \mathbf{x}(j) \mathbf{x}(i+1)^T \right. \\
 &\quad \left. - \sum_{j=1}^{i+1} \left(\prod_{r=j+1}^{i+1} A(\mathbf{w}) \right) B(\mathbf{w}) \mathbf{x}(j) \cdot \sum_{j=1}^{i+1} \left(\prod_{r=j+1}^{i+1} A(\mathbf{w}) \right)^T \mathbf{x}(j)^T B(\mathbf{w})^T \mathbf{w} \right).
 \end{aligned} \tag{5.13}$$

Da bismo uprostiti jednačine u analizi koja predstoji usvojićemo pretpostavku da je srednja vrednost ulazne sekvence nula i da su ulazni vektori u različitim trenucima odabiranja međusobno nezavisni. U slučaju da važi $(x_i \cdot x_k) = 0$, za $i \neq k$ (x_i i x_k su i -ta i k -ta komponenta ulaznog vektora \mathbf{x}) imaćemo da je autokovarijansna matrica \mathbf{R}_x dijagonalna i da sopstveni vektori te matrice imaju samo po jednu komponentu koja je različita od nule. U tom slučaju model zadovoljava zahtev koji je specificiran u paragrafu 5.1, da ulazna mreža potpuno potiskuje promenu koja je izazvana na ulazu. Slučaj u kome je srednja vrednost različita od nule će biti razmatran na kraju poglavlja. Takođe, analiza može biti sprovedena i za slučaj kada ulazni vektori u različitim trenucima nisu međusobno nezavisni, samo su u tom slučaju jednačine složenije.

Sada jednačina (5.13) može biti napisana kao

$$h(\mathbf{w}) = \lim_{t \rightarrow \infty} EQ(\mathbf{w}, \mathbf{x}) = \lim_{i \rightarrow \infty} E(\mathbf{w} \mathbf{x}(i+1) \mathbf{x}(i+1)^T) - \sum_{j=1}^{i+1} \left(\prod_{r=j+1}^{i+1} A(\mathbf{w}) \right)^2 \mathbf{w} \mathbf{x}(j) \mathbf{x}(j)^T \mathbf{w}^T \mathbf{w}, \tag{5.14}$$

ili, posle par transformacija

$$h(\mathbf{w}) = \mathbf{w} \mathbf{R}_x - \frac{1}{1 - (\mathbf{w} \mathbf{w}^T)^2} \mathbf{w} \mathbf{R}_x \mathbf{w}^T \mathbf{w}. \tag{5.15}$$

Sada ćemo naći stabilne tačke diferencijalne jednačine

$$\dot{\mathbf{w}} = \mathbf{w}\mathbf{R}_x - \frac{1}{1 - (\mathbf{w}\mathbf{w}^T)^2} \mathbf{w}\mathbf{R}_x \mathbf{w}^T \mathbf{w} \quad (5.16)$$

Fiksne tačke postoje za svako \mathbf{w} (\mathbf{w} je red vektor) koje je jednako transponovanom sopstvenom vektoru (ili linearnoj kombinaciji sopstvenih vektora) matrice \mathbf{R}_x . Biće pokazano da rešenje konvergira ka dominantnom (glavnom) sopstvenom vektoru sa verovatnoćom 1. Analiza koja sledi će biti uprošćena ako usvojimo sledeću pretpostavku za sopstvene vrednosti matrice \mathbf{R}_x :

$$\lambda_0 > \lambda_1 > \dots > \lambda_{K-1} > 0. \quad (5.17)$$

Drugim rečima, prepostavićemo da \mathbf{R}_x ima K različitih striktno pozitivnih sopstvenih vrednosti koje odgovaraju pojedinim ortonormalnim sopstvenim vektorima. Analiza koja sledi se može izvesti i bez te pretpostavke ali su u tom slučaju izvođenja komplikovanija. Obično, u realnim situacijama (na primer kada je ulazna sekvenca rezultat odabiranja govornog signala kod procesiranja govora ili vektor koji sadrži stepen osvetljenosti pojedinih piksela u prozoru kod procesiranja slike), sopstvene vrednosti kovarijansne matrice će biti različite i striktno pozitivne.

Ako usvojimo da je $\mathbf{m} = \mathbf{w}^T$, diferencijalna jednačina od interesa je

$$\dot{\mathbf{m}} = \frac{d\mathbf{m}}{dt} = \mathbf{R}_x \mathbf{m} - \frac{1}{1 - (\mathbf{m}^T \mathbf{m})^2} \mathbf{m} \mathbf{m}^T \mathbf{R}_x \mathbf{m} \quad (5.18)$$

Ako predstavimo \mathbf{m} pomoću ortonormalnog skupa sopstvenih vektora (\mathbf{e}_k), koji predstavljaju sopstvene vektore matrice \mathbf{R}_x dobijamo

$$\mathbf{m} = \sum_{k=0}^{K-1} \alpha_k \mathbf{e}_k, \quad (5.19)$$

gde je

$$\alpha_k = \mathbf{m}^T \mathbf{e}_k. \quad (5.20)$$

Imajući u vidu (5.19) i (5.20) dobijamo sledeće jednakosti

$$\mathbf{m}(\mathbf{m}^T \mathbf{R}_x \mathbf{m}) = \mathbf{m} \sum_{l=0}^{K-1} \lambda_l \alpha_l^2 = \sum_{l=0}^{K-1} \lambda_l \alpha_l^2 \left(\sum_{k=0}^{K-1} \alpha_k \mathbf{e}_k \right) \quad (5.21)$$

$$\mathbf{R}_x \mathbf{m} = \sum_{k=0}^{K-1} \lambda_k \alpha_k \mathbf{e}_k \quad (5.22)$$

$$\mathbf{m}^T \mathbf{m} = \sum_{k=0}^{K-1} \alpha_k^2. \quad (5.23)$$

Sada imamo

$$\dot{\mathbf{m}} = \sum_{k=0}^{K-1} \dot{\alpha}_k \mathbf{e}_k = \sum_{k=0}^{K-1} \lambda_k \alpha_k \mathbf{e}_k - \frac{1}{1 - \left(1 - \sum_{p=0}^{K-1} \alpha_p^2\right)^2} \left(\sum_{l=0}^{K-1} \lambda_l \alpha_l^2 \right) \sum_{i=0}^{K-1} \alpha_i \mathbf{e}_i. \quad (5.24)$$

Ako pomnožimo obe strane jednačine (5.24) sa \mathbf{e}_k^T za svako k , ortonormalnost vektora $\{\mathbf{e}_k\}$ implicira

$$\dot{\alpha}_k = \alpha_k \left(\lambda_k - \frac{1}{1 - \left(1 - \sum_{k=0}^{K-1} \alpha_k^2\right)^2} \sum_{l=0}^{K-1} \lambda_l \alpha_l^2 \right). \quad (5.25)$$

Ako uvedmo sledeću definiciju

$$\theta_k = \frac{\alpha_k}{\alpha_0} \quad (5.26)$$

pod pretpostavkom da je $\alpha_0 \neq 0$, što je tačno sa verovatnoćom jedan, za slučajno izabranu inicijalnu vrednost $\mathbf{w}(0)$, imamo

$$\dot{\theta}_k = \frac{1}{\alpha_0} (\dot{\alpha}_k - \theta_k \dot{\alpha}_0), \quad (5.27)$$

ili

$$\dot{\theta}_k = \frac{1}{\alpha_0} \left(\alpha_k \left(\lambda_k - \frac{1}{1 - \left(1 - \sum_{l=0}^{K-1} \alpha_l^2\right)^2} \sum_{j=0}^{K-1} \lambda_j \alpha_j^2 \right) - \theta_k \dot{\alpha}_0 \right) \quad (5.28)$$

$$\dot{\theta}_k = \theta_k (\lambda_k - \lambda_0). \quad (5.29)$$

Kako su sopstvene vrednosti označene u opadajućem redosledu, λ_0 je najveća sopstvena vrednost i lako je uočiti da $\theta_k \rightarrow 0$ kada $t \rightarrow \infty$ i $k > 0$.

Sa druge strane imamo

$$\dot{\alpha}_0 = \alpha_0 \left(\lambda_0 - \frac{1}{1 - \left(1 - \sum_{k=0}^{K-1} \alpha_k^2\right)^2} \sum_{l=0}^{K-1} \lambda_l \alpha_l^2 \right), \quad (5.30)$$

ili

$$\dot{\alpha}_0 = \alpha_0 \left(\lambda_0 - \frac{\left(\lambda_0 \alpha_0^2 + \alpha_0^2 \sum_{l=1}^{K-1} \lambda_l \theta_l^2 \right)}{1 - \left(1 - \alpha_0^2 - \alpha_0^2 \sum_{k=1}^{K-1} \theta_k^2\right)^2} \right). \quad (5.31)$$

Pošto $\theta_k \rightarrow 0$ kada $t \rightarrow \infty$ i $k > 0$, dobijamo sledeći izraz

$$\dot{\alpha}_0 = \lambda_0 \left(\alpha_0 - \frac{1}{1 - (1 - \alpha_0^2)^2} \alpha_0^3 \right), \quad (5.32)$$

ili

$$\begin{aligned} \dot{\alpha}_0 &= \frac{\lambda_0}{2 - \alpha_0^2} (\alpha_0 - \alpha_0^3) \\ &= \frac{\lambda_0 \alpha_0}{2 - \alpha_0^2} (1 - \alpha_0^2). \end{aligned} \quad (5.33)$$

Da bismo pokazali da gornja jednačina konvergira, zapazimo da je

$$V = (\alpha_0^2 - 1)^2, \quad (5.34)$$

funkcija Ljapunova, pošto je

$$\begin{aligned} \dot{V} &= 2(\alpha_0^2 - 1) \cdot 2\alpha_0 \dot{\alpha}_0 \\ &= -\frac{4\lambda_0}{2 - \alpha_0^2} \alpha_0^2 (\alpha_0^2 - 1)^2 \leq 0. \end{aligned} \quad (5.35)$$

V ima minimum za $\alpha_0 = \pm 1$. Znači, $\alpha_0 \rightarrow \pm 1$ kako $t \rightarrow \infty$. Pošto $\theta_k \rightarrow 0$ za $k > 0$, zaključujemo da $\alpha_k \rightarrow 0$ za $k > 0$. Znači za veliko t , jedina važna komponenta je α_0 , što znači da će \mathbf{m} konvergirati ka $\pm \mathbf{e}_0$.

Sada mora biti pokazano da postoji kompaktan podskup $O \subset D\{T\}$ vektora $\mathbf{w}(t)$ takvih da $\mathbf{w}(t) \in O$ beskonačno često sa verovatnoćom jedan, i da postoji slučajna promenljiva C takva da je $|y(i)| < C$ beskonačno često sa verovatnoćom jedan 1 za $\mathbf{w}(i) \in O$.

Da bismo to ostvarili razmotrimo sledeći algoritam:

$$w_k(i+1) = \frac{\tilde{w}_i(i+1)}{\psi[\tilde{w}_1(i+1), \dots, \tilde{w}_K(i+1)]} \quad (5.36)$$

i

$$\tilde{w}_k(i+1) = w_k(i) + \gamma \varphi[w_k(i), x_k(i+1), y(i+1)]. \quad (5.37)$$

Ovde $\varphi[]$ predstavlja inkrement u trenutku i ; normalizacija se ostvaruje pomoću neke ograničavajuće funkcije $\psi[]$. U slučaju koji nas interesuje usvajamo

$$\varphi[w_k(i), x_k(i+1), y(i+1)] = x_k(i+1) \cdot y(i+1) \quad (5.38)$$

$$\psi[\tilde{w}_1(i+1), \dots, \tilde{w}_K(i+1)] = \left[\sum_{k=1}^K \tilde{w}_k(i+1)^2 \right]^{1/2} \quad (5.39)$$

$$y(i+1) = (1 - \mathbf{w}(i) \cdot \mathbf{w}(i)^T) y(i) + \mathbf{w}(i) \mathbf{x}(i+1). \quad (5.40)$$

Implementacijom istih koraka koji su primenjeni u Ojinom radu [Oja, 1982] dobijamo (posle nekoliko jednostavnih transformacija)

$$\begin{aligned} w_k(i) &= w_k(i-1) + \gamma \varphi[x_k(i), y(i)] \\ &\quad - \gamma w_k(i-1) \frac{\partial \psi}{\partial \gamma} \Big|_{\gamma=0} + O(\gamma^2) \\ &= w_k(i-1) + \gamma x_k(i) \cdot y(i) - \gamma w_k(i-1) y(i)^2 + O(\gamma^2), \end{aligned} \quad (5.41)$$

što predstavlja zakon učenja koji je ovde i predložen (D_R je definisano ranije). Tada možemo reći da je sledeća jednačina zadovoljena:

$$\mathbf{w}(i) \cdot \mathbf{w}(i)^T = 1 + O(\gamma^2). \quad (5.42)$$

Sada se jednačina (5.12) može napisati na sledeći način

$$y(i+1) = \sum_{j=1}^{i+1} \left[\prod_{l=j+1}^{i+1} O(\gamma_{l-1}^2) \right] \cdot \mathbf{w}(j-1) \cdot \mathbf{x}(j). \quad (5.43)$$

Iz jednačine (5.42) zaključujemo da je $\mathbf{w}(k)$ ograničeno, i imajući u vidu pretpostavke A2 i A6, na osnovu (5.43) možemo zaključiti da postoji slučajna promenljiva C takva da je

$$|y(i)| < C \text{ beskonačno često sa verovatnoćom 1 za } \mathbf{w}(i) \in D_R.$$

Sada je prva pretpostavka za Teoremu 1 iz Ljungovog rada [Ljung, 1977], zadovoljena.

Sada ćemo dokazati da $\mathbf{w}(i) = \{\mathbf{w}_i\}$ posećuje skoro sigurno beskonačno često kompaktni podskup domena privlačenja jedne od asimptotski stabilnih tačaka, \mathbf{e}_0 ili $-\mathbf{e}_0$, diferencijalne jednačine (5.16). Da bismo to postigli pokazaćemo da postoji broj ε takav da se događaj $|\mathbf{w}_i \mathbf{e}_0| \geq \varepsilon$ dešava beskonačno često i skoro sigurno.

Iz jednačine (5.5) imamo

$$\mathbf{w}_i \cdot \mathbf{e}_0 = \mathbf{w}_{i-1} \cdot \mathbf{e}_0 + \gamma_i y_i \cdot (\mathbf{x}_i - \mathbf{w}_i^T y_i)^T \cdot \mathbf{e}_0. \quad (5.44)$$

Posle jednostavne transformacije dobijamo

$$\mathbf{w}_i \cdot \mathbf{e}_0 = (1 - \gamma_i^2 \gamma_i) \cdot \mathbf{w}_i \mathbf{e}_0 + \gamma_i (y_i \cdot \mathbf{x}_i^T) \cdot \mathbf{e}_0. \quad (5.45)$$

Imajući u vidu jednačinu (5.42) kao i to da počev od nekog i , član $(1 - \gamma_i^2 \gamma_i)$ postaje pozitivan (imajući u vidu da je pokazano da $y(i)$ uzima vrednosti iz konačnog intervala), primenjujući potpuno iste korake kao u dokazu Leme 3 Ojinog i Karhunenovog rada [Oja, 1985], može se pokazati da postoji broj ε takav da se događaj $|\mathbf{w}_i \mathbf{e}_0| \geq \varepsilon$ dešava beskonačno često i skoro sigurno. Tako, možemo zaključiti da je uslov A8 Ljungove teoreme zadovoljen.

Sad su svi uslovi Ljungove teoreme [Ljung, 1977] zadovoljeni. To znači da je dokazano da pod datim uslovima, jednačine (5.3) i (5.4) uslovljavaju da \mathbf{w} konvergira, sa verovatnoćom 1, ka transponovanom dominantnom sopstvenom vektoru matrice \mathbf{R}_x .

5.3. Poređenje sa nekim poznatim algoritmima za ekstrakciju dominantne glavne komponente

Kao što je već prikazano u glavi 3, postoji brojna literatura koja se bavi problemom implementacije PCA u neuralnim mrežama. Ovde ćemo razmotriti sličnosti i razlike predloženog ∞ OH metoda sa reprezentima pojedinih klasa algoritama.

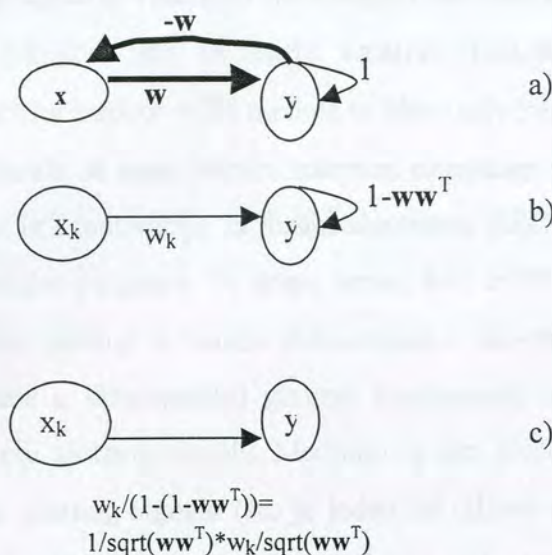
A. Ojin zakon učenja

Prvo će biti izvršeno poređenje predloženog zakona učenja i Ojinog zakona učenja. Poređenje sa pojedinim drugim poznatim zakonima, kao što je na primer SEC, nećemo raditi jer je pokazano [Baldi, 1995] da su mnogi poznati zakoni učenja u suštini ekvivalentni Ojinom zakonu učenja.

Na slici 5.2 je prikazan graf koji je ekvivalentan strukturi neuralne mreže koja se koristi za ∞ OH i neke ekvivalentne transformacije koje prevode taj graf u jednostavnije forme. Sa slike 5.2 se jasno mogu uočiti sličnosti i razlike između predloženog i Ojinog zakona učenja. Autor ne zna za postojanje matematičkog dokaza o ekvivalentnosti transformacija koje su prikazane na slici, ali intuitivno, ako je promena pojačanja grana grafa veoma spora u poređenju sa varijacijama veličina na ulazu grafa, rezultati bi trebalo da budu veoma slični sa slučajem kada pojačanja grana grafa imaju konstantne vrednosti. Za konstantne vrednosti pojačanja grana grafa vidi [Chen, 1971].

Poređenjem ∞ OH zakona učenja sa Ojinim zakonom učenja [Oja, 1982] može se uočiti nekoliko razlika.

Ojin zakon učenja ne primenjuje Hebov zakon učenja direktno, dok ∞ OH metod direktno primenjuje Hebov zakon učenja. Drugim rečima, zahvaljujući usvojenoj strukturi mreže predloženi metod koristi oba kraja sinapse simetrično, što nije slučaj sa Ojinim zakonom učenja. Takođe, ∞ OH metod ne zahteva nikakva dodatna lokalna preračunavanja – u slučaju Ojinog zakona učenja potrebna su dodatna izračunavanja (potrebno je izračunati kvadrat postsinaptičkog potencijala). Sa tehničke tačke gledišta (a čini se i sa biološke, takođe) ovo se može smatrati dobrom stranom predloženog algoritma.



Slika 5.2 Grafovska reprezentacija predloženog zakona učenja i njegove ekvivalentne transformacije koje dovode do struktura koje podsećaju na zakone učenja predložene u [Oja, 1982] i [Kohonen, 1982] (sa izuzetkom jednog dodatnog člana) (potrebno je uočiti prisustvo dodatnog člana $1/\sqrt{WW^T}$) – bazirano na rezultatima simulacija, ovaj dodatni član nema značajan uticaj na zakon učenja koji je predložio Oja [Oja, 1982])

Novi algoritam vrši vremensku integraciju, što nije slučaj sa Ojinim modelom. To se, takođe, može smatrati dobrom osobinom novog algoritma jer se čini da je postojanje integracije neurotransmitera sasvim razumna pretpostavka.

Kod Ojinog algoritma izlaz predstavlja linearnu kombinaciju trenutnog stanja ulaza mreže. U ∞ OH metodu to nije slučaj, striktno govoreći. Međutim, iz analize predloženog rešenja se može videti da nelinearnost iščezava sa vremenom.

B. Harpurova rekurentna korekcija greške (REC)

REC mreža [Harpur, 1997] u slučaju mreže sa samo jednim izlaznim neuronom, ima identičnu strukturu kao i ∞ OH mreža kada se izabere $\mu = 1$ i $\gamma = 1$. Takođe, i zakon učenja je identičan. ∞ OH metod je razvijen bez ranijeg uvida u postojanje REC metoda (na REC metod je autoru skrenuta pažnja tokom revizije rada [Jankovic, 2003a]). Međutim, postoji veoma bitna razlika u primeni zakona učenja. Ta razlika se ogleda u tome što primena REC metoda podrazumeva da se ulazni signal održava konstantnim dovoljno dugo dok izlazni potencijal i

proces modifikacije sinaptičkih vrednosti ne dostignu stacionarno stanje. Ta pretpostavka je nepotrebna za ∞ OH metod (što se može smatrati biološki mnogo verovatnijim), i konsekvntno matematička analiza ∞ OH metoda se bitno usložnjava.

U [Harpur, 1997], mreža sa samo jednim izlaznim neuronom nije bila posebno razmatrana. Jedan od razloga za to je i motivacija za dizajn algoritma. REC algoritam je usmeren na što bolju rekonstrukciju ulaznog signala. Sa druge strane, kod ∞ OH metoda to je samo jedan od ciljeva. Razlog za takav pristup je veoma jednostavan – ako karakteristike ulaznog signala nisu dobro lokalizovane u dominantnoj glavnoj komponenti onda ne možemo govoriti o kvalitetnoj rekonstrukciji ulaznog signala. Međutim, u tom slučaju još uvek možemo postići perfektno potiskivanje ulaznog signala (što je jedan od ciljeva ∞ OH metoda) – dominantna glavna komponenta će biti precizno rekonstruisana dok će preostale komponente biti ignorisane. Znači, pored načina primene REC i ∞ OH metod se razlikuju i u motivaciji koja je dovela do njihove konstrukcije.

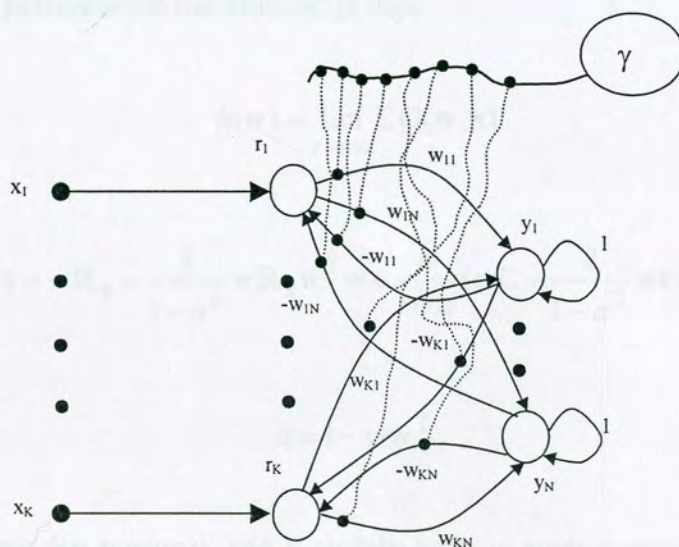
C: Pobednik-uzima-sve (*Winner-Take-All*, WTA) zakon učenja

“Pobednik-uzima-sve“ ideja je osnova za nekoliko samoorganizujućih sistema [Grossberg, 1988]. U slučaju kada je korelaciona matrica ulaznog signala dijagonalna (što znači da su ulazne komponente međusobno dekorelisane) metod predložen u ovom radu će rezultovati WTA zakonom učenja. Drugim rečima samo će jedna sinaptička vrednost biti različita od nule (u predloženom slučaju će biti jednaka 1). To dalje znači da će samo jedna ulazna komponenta biti predstavljena na izlazu i to ona koja odgovara dominantnom glavnom vektoru. Preostale komponente će biti ignorisane.

5.4. Generalizacija predloženog algoritma za slučaj mreža sa više izlaza

Predloženi metod se može generalizovati za slučaj kada neuralna mreža ima više izlaza. U slučaju GHA metoda [Sanger, 1989], generalizacija je jednostavna i u tom slučaju mreža će obavljati PCA. Takođe, moguće je izvršiti generalizaciju u smislu potprostornog kriterijuma [Oja, 1983][Oja, 1989] ili REC mreže sa više izlaza i u tom slučaju će mreža obavljati PSA (u tom slučaju, struktura mreže je prikazana na slici 5.3). U slučaju da se izvrši generalizacija slična težinskom potprostornom metodu [Oja, 1992] mreža će obavljati PCA.

Moguće je izvršiti generalizaciju i na nešto drugačiji način. Ako mreža ima više od jednog izlaza i lokalne povratne veze u ∞ OH algoritmu zamenimo globalnom povratnom vezom, to će rezultovati MH/MHO [Janković, 2001] PSA algoritmom. Taj metod će biti predmet analize u sledećem poglavlju.



Slika 5.3 REC neuralna mreža

5.5 Analiza u slučaju kada ulazna sekvenca nema srednju vrednost nula

Ovde ćemo ponoviti neke proračune za slučaj kada ulazna sekvenca nema nultu srednju vrednost. Biće usvojeno da za svako t , $\mathbf{x}(t) = \mathbf{x} + \mathbf{xs}(t)$ gde je \mathbf{x} fiksni vektor i $\mathbf{xs}(t)$ simetrično distribuiran vektor sa srednjom vrednošću nula. U tom slučaju je dominantni glavni vektor ulazne korelacione matrice \mathbf{R}_x vektor \mathbf{x} , i najveća sopstvena vrednost \mathbf{R}_x je $1+p$, gde je p varijansa vektora $\mathbf{xs}(t)$. Sada ćemo usvojiti oznake $\mathbf{R}_{xs} = E\{\mathbf{xs} \mathbf{xs}^T\}$ i $\mathbf{C} = \mathbf{xx}^T$. U tom slučaju

$$\begin{aligned} \lim_{t \rightarrow \infty} EQ(\mathbf{w}, \mathbf{x}) &= \lim_{i \rightarrow \infty} \sum_{j=1}^{i+1} \left(\prod_{p=j+1}^{i+1} A(\mathbf{w}) \right) B(\mathbf{w}) \mathbf{x}(j) \mathbf{x}(k+1)^T \\ &\quad - \sum_{j=1}^{i+1} \left(\prod_{p=j+1}^{i+1} A(\mathbf{w}) \right) B(\mathbf{w}) \mathbf{x}(j) \cdot \sum_{j=1}^{i+1} \left(\prod_{p=j+1}^{i+1} A(\mathbf{w}) \right)^T \mathbf{x}(j)^T B(\mathbf{w})^T \mathbf{w} \end{aligned}$$

što nakon nekoliko jednostavnih transformacija daje

$$h(\mathbf{w}) = \lim_{t \rightarrow \infty} EQ(\mathbf{w}, \mathbf{x})$$

$$h(\mathbf{w}) = \mathbf{wR}_x - \frac{1}{1-a^2} \mathbf{wR}_x \mathbf{w}^T \mathbf{w} + \frac{a}{1-a} (\mathbf{wC} - \frac{2}{1-a^2} \mathbf{wC} \mathbf{w}^T \mathbf{w})$$

$$a = 1 - \mathbf{w} \mathbf{w}^T.$$

Primenjujući potpuno isti postupak kao u slučaju kada je srednja vrednost ulazne sekvence vektora bila nula i zadržavajući iste oznake, dobijamo

$$\dot{\theta}_k = \theta_k (\lambda_k - \lambda_0 + 1 - \frac{1}{1-a}).$$

Ponovo možemo zaključiti (na osnovu prethodne analize i usvojene vrednosti za γ_i) da $\theta_k \rightarrow 0$ kada $t \rightarrow \infty$ i kada je $k > 0$. Posle par transformacija dobijamo sledeći izraz

$$\dot{\alpha}_0 = \frac{\lambda_0}{2 - \alpha_0^2} (\alpha_0 - \alpha_0^3) + \frac{a}{1 - a} \left(\alpha_0 - \frac{2}{1 - a^2} \alpha_0^3 \right),$$

ili

$$\dot{\alpha}_0 = (\lambda_0 - 1) \frac{\alpha_0 - \alpha_0^3}{2 - \alpha_0^2}.$$

Da bismo pokazali da gornja jednačina ima stacionarno stanje uočimo, ponovo, da je

$$V = (\alpha_0^2 - 1)^2,$$

funkcija Ljapunova, pošto je jednostavno proveriti da je (ako imamo u vidu da je $\lambda_0 > 1$)

$$\dot{V} < 0.$$

Znači, sada se može izvući potpuno isti zaključak kao i slučaju kada je srednja vrednost ulazne sekvenca bila nula.

VI Poglavlje

Modulisani Hebov (MH) metod – biološki verovatan metod za izračunavanje PSA

MH metod predstavlja novi biološki verovatan model koji vrši PSA analizu. Metod je inspirisan delom strukture retine kod riba.

U ovom, uvodnom, delu ćemo posmatrati mrežu koja ima jednak broj ulaza i izlaza ($N=K$). Taj slučaj nije od velikog praktičnog značaja ali omogućava jednostavno predstavljanje jednog zakona učenja koji će biti predstavljen u sledećem paragrafu. Biće pokazano da predložena mreža uz predloženi zakon učenja obezbeđuje da sinaptički vektori u stacionarnom stanju predstavljaju bazu potprostora koji je definisan korelacionom matricom ulazne sekvence slučajnog stacionarnog signala.

Da bismo predstavili novi zakon učenja ponovo ćemo analizirati jednačinu (4.18) koja je dobijena pod pretpostavkom da je $\mathbf{W}^T \mathbf{W} = \mathbf{I}_N$. Jednačina (4.18) može biti napisana kao

$$\|\mathbf{x}_{rec} - \mathbf{x}\|^2 = \sum_{k=1}^N x_k^2 - \sum_{k=1}^N y_k^2 = \min. \quad (6.1)$$

Kao što je već pomenuto u prethodnom poglavlju i objašnjeno u literaturi [Oja, 1992] očekivana greška se minimizira kada se član

$$\sum_{k=1}^N y_k^2$$

maksimizira. Već je objašnjeno kod SLA algoritma (poglavlje 4) da u slučaju direktnog generisanja zakona učenja na osnovu gradijentnog metoda, postoji potreba za dodatnim

članovima u zakonu učenja koji bi obezbedili ortogonalnost sinaptičkih vektora. Sada ćemo pokazati nešto drugačiji pristup koji će dovesti do novog zakona učenja.

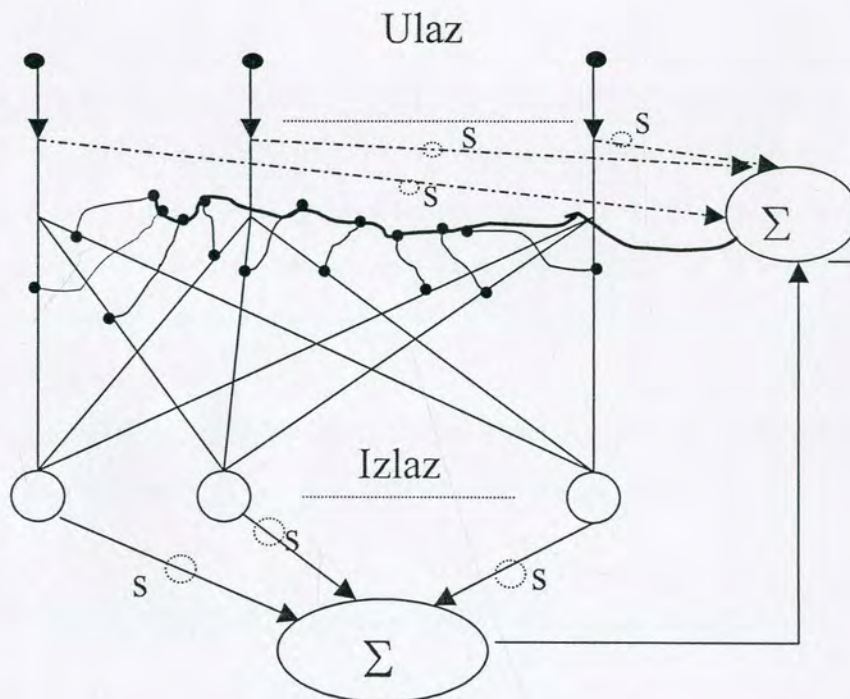
Jasno je da desna strana jednačine (6.1) (bez ikakvih dodatnih ograničenja), takođe, ne može dovesti do stabilnog silaznog gradijentnog metoda jer je u tom slučaju minimalna vrednost $-\infty$. Zato će sada biti razmotren novi kriterijum

$$J_N^{LoPCA} = E \left\{ \left(\sum_{k=1}^N x_k^2 - \sum_{k=1}^N y_k^2 \right)^2 \right\}, \quad (6.2)$$

na osnovu koga se može direktno generisati sledeći silazni gradijentni metod:

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \left(\sum_{k=1}^N x_k^2(i) - \sum_{k=1}^N y_k^2(i) \right) \mathbf{x}(i) \mathbf{y}^T(i). \quad (6.3)$$

Nazvaćemo ovaj metod Modulisanim Hebovim (MH) algoritmom. Razlog za to je jednostavan – iz jednačine se vidi da zakon učenja predstavlja Hebov član koji je modulisan jednim jedinim signalom koji predstavlja razliku energija ulaznog i izlaznog signala. Jasno se može uočiti da u ovoj jednačini ne postoje nikakva dodatna ograničenja koja obezbeđuju konvergenciju algoritma. Struktura neuralne mreže za implementaciju ovog zakona učenja je prikazana na slici 6.1.



Slika 6.1 Lokalno implementirana PSA neuralna mreža (s označava kvadratnu funkciju)

Sa slike 6.1 se uočava da predloženi metod ima osobine lokalnosti i homogenosti što predstavlja poželjne karakteristike mreže koja se realizuje u paralelnom hardveru. Ovde nećemo vršiti detaljno poređenje ovog metoda sa drugim poznatim metodima, jer će to biti urađeno za slučaj MHO algoritma koji predstavlja modifikaciju ovog algoritma u slučaju kada je broj izlaznih neurona manji od broja ulaznih neurona, a što je od mnogo većeg praktičnog interesa. Ovde ćemo samo istaći da se iz jednačine (6.3) može jednostavno uočiti da modifikacija pojedinih sinaptičkih težina ne zahteva informaciju o eksplicitnoj vrednosti bilo koje druge sinaptičke vrednosti. Takođe, može se uočiti da je broj sumatora koji obavljaju globalne kalkulacije uvek 2, bez obzira na broj ulaznih i izlaznih neurona.

6.1 Matematička analiza MH metoda

Sada ćemo pretpostaviti da slučajni ulazni vektor sa srednjom vrednošću nula, ostaje stacionaran u periodu koji je od interesa. U tom slučaju je moguće primeniti stohastičku aproksimaciju kako je to već rađeno ranije [Ljung, 1977] [Oja, 1985]. Usvojicemo pretpostavku da ulazni vektor ima međusobno nezavisne komponente, što znači da je autokorelaciona matrica \mathbf{C} dijagonalna. Za implementaciju algoritma nije neophodno da komponente vektora \mathbf{x} budu međusobno nezavisne, ali takva pretpostavka znatno uprošćava matematičku analizu koja sledi.

Usrednjavanjem jednačine (6.3) (detalji ovog postupka se mogu videti u [Ljung, 1977], [Oja, 1985]) i korišćenjem kompaktne notacije dobijamo sledeću diferencijalnu jednačinu

$$\frac{d\mathbf{W}}{dt} = \text{diag}(\text{kurt}(\mathbf{x}))\text{diag}(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{W} + \text{trace}(\mathbf{C} - \mathbf{W}\mathbf{W}^T\mathbf{C})\mathbf{C}\mathbf{W} + 2\mathbf{C}(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{C}\mathbf{W}, \quad (6.4)$$

gde $\text{diag}(\text{kurt}(\mathbf{x}))$ predstavlja dijagonalnu matricu čiji su dijagonalni elementi jednaki kurtozisu individualnih komponentata ulaznog signala i pri čemu je definicija kurtozisa data sa

$$\text{kurt}(x_i) = E(x_i^4) - 3E^2(x_i^2).$$

Iz jednačine (6.4) se može jednostavno zaključiti da jedno rešenje koje zadovoljava gornju jednačinu jeste \mathbf{W} takvo da važi $\mathbf{W}\mathbf{W}^T = \mathbf{I}$. Ovde nećemo striktno analizirati jednačinu (6.4). Koristeći metod koji je sličan onom koji sledi moguće je pokazati da stacionarno stanje zahteva da matrica $\mathbf{M} = \mathbf{W}\mathbf{W}^T$ bude ortogonalna ako se \mathbf{W} sastoji od N nezavisnih kolona. Ovde ćemo detaljno analizirati samo jedan specijalan slučaj jednačine (6.4) – usvojicemo da su ulazni signali Gausovog tipa.

U tom slučaju za svaku individualnu komponentu ulaznog signala imamo da joj je kurtozis nula, tako da sistem jednačina koji će biti analiziran možemo pisati kao

$$\frac{d\mathbf{W}^k}{dt} = \text{trace}(\mathbf{C} - \mathbf{W}\mathbf{W}^T\mathbf{C})\mathbf{C}\mathbf{W}^k + 2\mathbf{C}(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{C}\mathbf{W}^k, \quad (6.5)$$

gde \mathbf{W}^k ($k=1,2,\dots,N$) predstavlja k -tu kolonu matrice \mathbf{W} . Jednačine dostižu stacionarno stanje u tačkama u kojima važi

$$0 = \text{trace}\left(\left(\mathbf{I} - \mathbf{W}\mathbf{W}^T\right)\mathbf{C}\right)\mathbf{C}\mathbf{W}^k + 2\mathbf{C}\left(\mathbf{I} - \mathbf{W}\mathbf{W}^T\right)\mathbf{C}\mathbf{W}^k \quad (6.6)$$

ili

$$\mathbf{F}\mathbf{C}\mathbf{W}^k = -\frac{1}{2}\text{trace}(\mathbf{F})\mathbf{C}\mathbf{W}^k, \quad \text{gde je}$$

$$\mathbf{F} = 2\mathbf{C}\left(\mathbf{I} - \mathbf{W}\mathbf{W}^T\right).$$

Iz ove jednačine je jasno da sistem pokušava da nađe sopstvene vektore matrice \mathbf{F} i da su ti vektori predstavljeni sa $\mathbf{C}\mathbf{W}^k$. Sada će biti pokazano da ako matrica \mathbf{W} ima N nezavisnih kolona ($\text{rank}(\mathbf{W}) = N$) tada mora biti $\mathbf{W}^T\mathbf{W}=\mathbf{I}$. Iz jednačine (6.6) je jednostavno zaključiti da jedno od rešenja predstavlja i \mathbf{W} takvo da je $\mathbf{W}\mathbf{W}^T=\mathbf{I}$. U tom slučaju je $\mathbf{F}=0$. Sada ćemo pretpostaviti da postoji drugo \mathbf{W} takvo da je $\text{rank}(\mathbf{W}) = N$ i da je \mathbf{W} rešenje jednačine (6.6). Iz jednačine (6.6) tada vidimo da matrica \mathbf{F} ima N linearno nezavisnih sopstvenih vektora koji svi imaju sopstvene vrednosti $-0.5 \text{trace}(\mathbf{F})$. Sa druge strane, poznato je da suma svih sopstvenih vrednosti matrice \mathbf{F} mora biti jednaka $\text{trace}(\mathbf{F})$. Tada imamo

$$-\frac{N}{2}\text{trace}(\mathbf{F}) = \text{trace}(\mathbf{F})$$

a što je moguće jedino u slučaju

$$\text{trace}(\mathbf{F}) = 0.$$

U tom slučaju matrica \mathbf{F} ima sve sopstvene vrednosti jednake nuli i tada sledi da mora biti

$$\mathbf{F}=0.$$

Drugim rečima, ako je $\text{rank}(\mathbf{W}) = N$, tada matrica \mathbf{F} mora biti nula i to je moguće samo u slučaju

$$\mathbf{W}\mathbf{W}^T=\mathbf{I},$$

što znači da važi $\mathbf{W}^T\mathbf{W} = \mathbf{I}$.

Ako označimo $\mathbf{W}\mathbf{W}^T$ sa \mathbf{M} , tada na osnovu jednačine (6.4), a za slučaj ulaznog signala Gausovog tipa, dobijamo

$$\frac{d\mathbf{M}}{dt} = \delta\mathbf{C}\mathbf{M} + 2\mathbf{C}(\mathbf{I} - \mathbf{M})\mathbf{C}\mathbf{M} + \mathbf{M}\mathbf{C}\delta + 2\mathbf{M}\mathbf{C}(\mathbf{I} - \mathbf{M})\mathbf{C}, \quad (6.7)$$

gde je

$$\delta = \text{trace}(\mathbf{C} - \mathbf{W}\mathbf{W}^T\mathbf{C}).$$

Označimo sada sa \mathbf{m}_k k -tu kolonu matrice $\mathbf{M}(k) = \mathbf{W}(k)\mathbf{W}(k)^T$. Sada bi bilo potrebno dokazati da \mathbf{m}_k posećuje skoro sigurno beskonačno često kompaktan podskup domena privlačenja asimptotski stabilnih tačaka jednačine (6.7). Pronalaženje domena privlačenja i dokazivanje gornje osobine je veoma težak zadatak. Ono što će biti rečeno ovde (i što je često korišćeno u literaturi – na primer [Oja, 1992b]) je da u svim simulacijama u kojima je postignuto stacionarno stanje, ono koincidira sa stabilnim rešenjima jednačine (6.7).

Sada ćemo dokazati da $\mathbf{M}=\mathbf{I}$ predstavlja asimptotski stabilno rešenje jednačine (6.7). Da bismo to uradili primenićemo istu proceduru koja je primenjena kod dokaza teoreme 1 u [Oja, 1992]. Kao što je to već objašnjeno u prethodnom tekstu (poglavlje o ∞ OH) analiza koja sledi će biti uprošćena ako se pretpostavi da su sve sopstvene vrednosti matrice \mathbf{C} različite i pozitivne.

Da bismo dokazali asimptotsku stabilnost rešenja $\mathbf{M}=\mathbf{I}$, posmatraćemo male perturbacije $\mathbf{E}(t)=\{e_{ij}\}$ u odnosu na stacionarnu tačku $\mathbf{M}=\mathbf{I}$. Tada imamo

$$\mathbf{M}(t) = \mathbf{I} + \mathbf{E}(t). \quad (6.8)$$

Tada za $\mathbf{E}(t)$ važi,

$$\begin{aligned} \frac{d\mathbf{E}}{dt} &= \text{trace}(\mathbf{C}\mathbf{E})\mathbf{C}(\mathbf{I} + \mathbf{E}) - 2\mathbf{C}\mathbf{E}\mathbf{C}(\mathbf{I} + \mathbf{E}) - \text{trace}(\mathbf{C}\mathbf{E})(\mathbf{I} + \mathbf{E})\mathbf{C} - 2(\mathbf{I} + \mathbf{E})\mathbf{C}\mathbf{E}\mathbf{C} \\ &= 2(\text{trace}(\mathbf{C}\mathbf{E})\mathbf{C} + 2\mathbf{C}\mathbf{E}\mathbf{C}) - \text{trace}(\mathbf{C}\mathbf{E})(\mathbf{E}\mathbf{C} + \mathbf{C}\mathbf{E}) - 2\mathbf{C}\mathbf{E}\mathbf{C}\mathbf{E} - 2\mathbf{E}\mathbf{C}\mathbf{E}\mathbf{C} \end{aligned}$$

odnosno

$$\frac{d\mathbf{E}}{dt} = -2(\text{trace}(\mathbf{C}\mathbf{E})\mathbf{C} + 2\mathbf{C}\mathbf{E}\mathbf{C}) + O(\|\mathbf{E}\|^2), \quad (6.9)$$

gde $O(\|\mathbf{E}\|^2)$ predstavlja matricu koja se sastoji od članova drugog reda e_{ij} matrice \mathbf{E} .

Ako usvojimo da je $\|\mathbf{E}\|$ dovoljno malo, linearni deo (po \mathbf{E}) jednačine (6.9) određuje asimptotsku stabilnost. U tom slučaju, elementi matrice \mathbf{E} koji se nalaze van glavne dijagonale, zadovoljavaju sledeći skup jednačina:

$$\frac{de_{ij}}{dt} = -4c_i c_j e_{ij} \quad (6.10)$$

$$i \neq j,$$

gde c_i, c_j predstavljaju i -ti i j -ti dijagonalni element matrice \mathbf{C} , respektivno. Iz jednačine (6.10) je jasno da $e_{ij} \rightarrow 0$ (teži asimptotski ka nuli), kada se ima u vidu da su sve sopstvene vrednosti matrice \mathbf{C} pozitivne.

Dijagonalni elementi matrice \mathbf{E} zadovoljavaju sledeći skup jednačina:

$$\frac{de_{ii}}{dt} = -2c_i \sum_{j=1}^N c_j e_{jj} - 4c_i^2 e_{ii}, \quad (6.11)$$

$$i = 1, 2, \dots, N.$$

Definišimo V kao

$$V = \sum_{i=1}^N e_{ii}^2. \quad (6.12)$$

Kako je $V \geq 0$ i

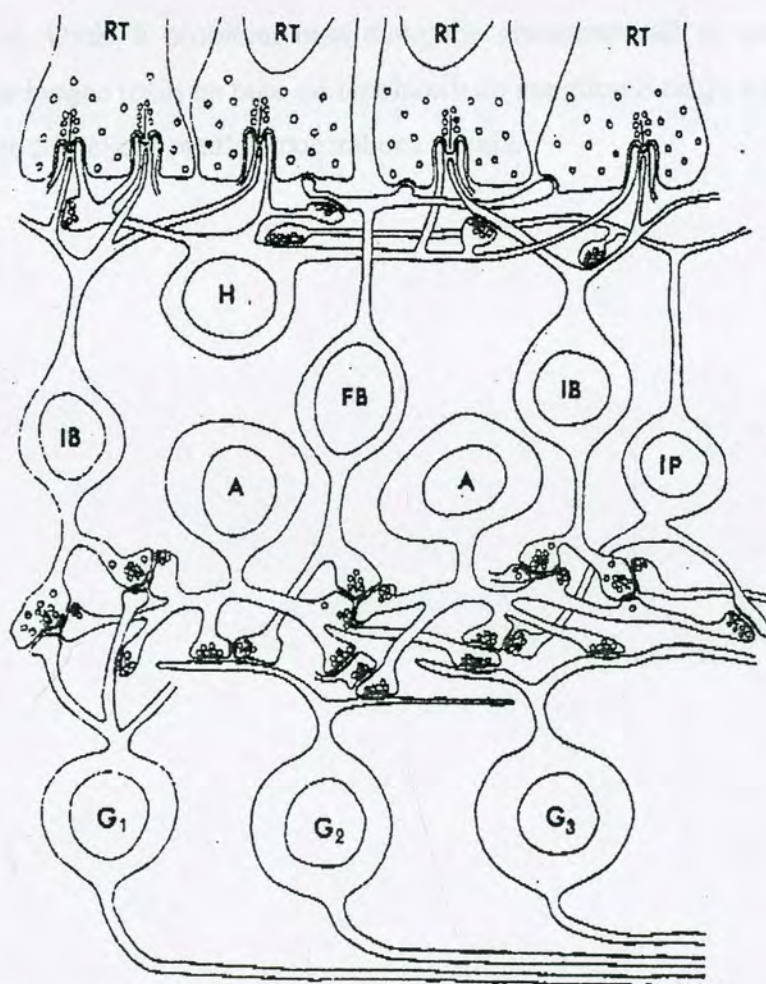
$$\begin{aligned} \frac{dV}{dt} &= -4 \left(\sum_{i=1}^N e_{ii} \left(c_i \sum_{j=1}^N c_j e_{jj} + 2c_i^2 e_{ii} \right) \right) \\ &= -4 \left(\sum_{i=1}^N e_{ii} c_i \left(\sum_{j=1}^N c_j e_{jj} \right) + 2 \sum_{i=1}^N c_i^2 e_{ii}^2 \right) \\ &= -4 \left(\left(\sum_{i=1}^N e_{ii} c_i \right)^2 + 2 \sum_{i=1}^N c_i^2 e_{ii}^2 \right) \leq 0, \end{aligned} \quad (6.13)$$

V predstavlja funkciju Ljapunova. Sada možemo zaključiti da (6.11) konvergira. Jednostavno se uočava da V konačno dostiže svoj minimum $V=0$ što znači da je $e_{ij} = 0$ za svako i, j . Drugim rečima, pokazali smo da, pod pretpostavkama koje su usvojene, predloženi algoritam dovodi do toga da je matrica \mathbf{M} ortonormalna.

6.2 Poređenje usvojene strukture sa delom retine kod riba

U literaturi postoji više algoritama za neuralne mreže za koje se tvrdi da predstavljaju biološki verovatne modele (videti [Foldiak, 1992], [Fyfe, 1995], [Harpur, 1997]). Neki od rezultata su bili direktno inspirisani psihofizičkim rezultatima dobijenim ispitivanjem retine (vidi [Atick, 1990; Atick, 1992; Atick, 1993], [Goodall, 1960]). U tim radovima, međutim, cilj autora nije bio da formiraju strukturu neuralne mreže koja bi imala sličnosti sa biološkim uzorom već samo ostvarivanje slične prenosne funkcije. Sa druge strane MH algoritam (i njegov derivat MHO) su zapravo inspirisani delom strukture retine kod riba.

Slika 6.2 predstavlja aproksimaciju dela retine ribe koja je prikazana u (Dowling, page 76, Fig. 3.17).



Slika 6.2 Uprošćen prikaz dela retine ribe (IB- invaginacione bipolarne ćelije; H- horizontalne ćelije; IP – interpleksiformne ćelije; A – amarkrine ćelije; FB – ravne bipolarne ćelije; G₁ – ganglione ćelije; RT – krajevi štapića)

Poređenjem sa strukturom MH mreže koja je prikazana na slici 6.1, možemo uočiti veoma veliki stepen sličnosti sa vezama između IB, H, A i IP ćelija. Iako autor nema nikakve rezultate koji bi ukazivali i na funkcionalnu sličnost, visok stepen strukturne sličnosti može ukazivati i na postojanje određene verovatnoće o postojanju sličnosti u funkcionalnom smislu. Ako imamo u vidu da je u referencama [Atick, 1992], [Deco, 1996] sugerisano da je cilj preprocesiranja koje se odvija u retini, u stvari transformisanje vizuelnog ulaza u što je moguće više dekorelisanu bazu (u model koji sadrži manji stepen redundantnosti ili generalnije, u statistički nezavisnu reprezentaciju u korteksu), možemo reći da predloženi metod zaslužuje da bude analiziran kao biološki moguć. U literaturi su opisana biološki moguća rešenja (na primer [Atick, 1993]) koja modeluju sveukupno procesiranje koje se obavlja u retini. Međutim, ta rešenja nisu bila usmerena na imitiranje strukture retine. U referenci [Atick, 1993] su analizirani različiti aspekti procesiranja koje se obavlja u retini – kao što su lokalizacija prijemnih polja koja bi odgovarala ganglionim ćelijama retine, ili problem simetrije. Ovde ti problemi nisu detaljnije analizirani ali se može reći da bi se problem simetrije mogao rešiti na putu od bipolarnih do ganglionih ćelija u nekom međusloju, ili delimičnom modifikacijom predloženog zakona učenja.

6.3 Modulisani Heb-Oja zakon učenja

U slučaju kada je broj izlaznih neurona manji od broja ulaznih neurona MH metod će rezultovati ortogonalnim sinaptičkim vektorima ali ti vektori više neće biti ortonormalni. Da bi se održala ortonormalnost kolona sinaptičke matrice \mathbf{W} , potrebno je izvršiti modifikaciju zakona učenja. Jedna mogućnost je implementacija ∞ OH metoda. Međutim, u tom slučaju je matematička analiza veoma komplikovana. Pošto je kod analize ∞ OH metoda pokazano da postoji sličnost sa Ojnim zakonom učenja, ovde će kao dodatak MH metodu na svaki pojedinačni sinaptički vektor biti primenjen Ojini stabilizacioni član. U tom slučaju zakon učenja od interesa se izražava sledećom jednačinom:

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \left(\mathbf{x}(i) \mathbf{x}(i)^T - \mathbf{y}(i) \mathbf{y}(i)^T \right) \cdot \left(\mathbf{x}(i) \mathbf{y}(i)^T - \mathbf{W}(i) \text{diag} \left(\mathbf{y}(i) \mathbf{y}(i)^T \right) \right). \quad (6.14)$$

Ovaj zakon učenja ćemo nazvati Modulisanim Heb-Oja zakonom učenja (MHO). Primenjen na mreži koja je prikazana na slici 6.1, ovaj zakon učenja generiše sinaptičku matricu \mathbf{W} , čije kolone pokrivaju potprostor koji definiše N dominantnih sopstvenih vektora matrice \mathbf{C} .

6.4 Matematička analiza MHO metoda

U analizi koja sledi pretpostavićemo da je ulazni signal stacionaran i da mu je srednja vrednost nula. Takođe, da bi se analiza uprostila, pretpostavićemo da su komponente ulaznog vektora međusobno nezavisne (bez te pretpostvake analiza koja sledi bi bila znatno komplikovanija). Primenom stohastičke aproksimacije asimptotski stabilna rešenja jednačine (6.14) se mogu dobiti rešavanjem sledeće jednačine (koja je od jednačine (6.14) dobijena usrednjavanjem)

$$\begin{aligned} \frac{d\mathbf{W}}{dt} = & \left(\text{diag}(\text{kurt}(\mathbf{x}))\text{diag}(\mathbf{I} - \mathbf{W}\mathbf{W}^T) + \text{tr}(\mathbf{C} - \mathbf{W}\mathbf{W}^T\mathbf{C})\mathbf{C} + 2\mathbf{C}(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{C} \right)\mathbf{W} \\ & - \mathbf{W}\text{diag}\left(\mathbf{W}^T \left(\text{diag}(\text{kurt}(\mathbf{x}))\text{diag}(\mathbf{I} - \mathbf{W}\mathbf{W}^T) + \text{tr}(\mathbf{C} - \mathbf{W}\mathbf{W}^T\mathbf{C})\mathbf{C} + 2\mathbf{C}(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{C} \right)\mathbf{W}\right). \end{aligned} \quad (6.15)$$

U analizi koja sledi biće analizirane stabilne tačke jednačine (6.15) kao i njihova oblast privlačenja. Da bi se dokazalo da originalni diskretni algoritam (6.14) ima iste stacionarne tačke bilo bi potrebno dokazati da (6.14) posećuje skoro sigurno, beskonačno često kompaktan podskup domena privlačenja asimptotski stabilnih rešenja jednačine (6.15). Mora se reći da je takav dokaz veoma težak i da taj deo dokaza ne postoji za većinu poznatih algoritama. Na bazi iskustva, generalno se može reći da ako željena stabilna tačka diskretnog algoritma ne koincidira sa nekom od asimptotski stabilnih tačaka pridružene diferencijalne jednačine, tada do konvergencije diskretnog algoritma neće ni doći [Oja, 1992]. Ovde će biti naglašeno da su u svim simulacijama koje su rađene i u kojima je algoritam (6.14) konvergirao, stabilne tačke algoritma koincidirale sa asimptotski stabilnim rešenjima jednačine (6.15).

Prvo će biti pokazano da (6.15) obezbeđuje da u stacionarnim tačkama važi:

$$\mathbf{W}^T\mathbf{W} = \mathbf{I}_N. \quad (6.16)$$

Ako usvojimo da ulazni signali zadovoljavaju Gausovu raspodelu ($\text{kurt}(\mathbf{x}) = 0$), jednačina (6.15) poprima sledeći oblik

$$\begin{aligned} \frac{d\mathbf{W}}{dt} = & \left(\text{tr}(\mathbf{C} - \mathbf{W}\mathbf{W}^T\mathbf{C})\mathbf{C} + 2\mathbf{C}(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{C} \right) \mathbf{W} \\ & - \mathbf{W} \text{diag} \left(\mathbf{W}^T \left(\text{tr}(\mathbf{C} - \mathbf{W}\mathbf{W}^T\mathbf{C})\mathbf{C} + 2\mathbf{C}(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{C} \right) \mathbf{W} \right). \end{aligned} \quad (6.17)$$

Od ovog momenta ćemo analizirati samo slučaj kada ulazni signal zadovoljava Gausovu raspodelu jer će to jednačine učiniti manje komplikovanim i analizu koja sledi jednostavnijom. Analiza koja sledi se može primeniti i bez te pretpostavke ali jednačine postaju znatno složenije i analiza znatno komplikovanija. Iz jednačine (6.17) je jasno da sistem pokušava da pronađe sopstvene vektore matrice $\mathbf{F} = (\text{tr}(\mathbf{C} - \mathbf{W}\mathbf{W}^T\mathbf{C}) * 2\mathbf{C}(\mathbf{I} - \mathbf{W}\mathbf{W}^T)\mathbf{C})$ i da su ti sopstveni vektori predstavljeni kolonama matrice \mathbf{W} . Može se jednostavno zaključiti da ako matrica \mathbf{W} ima N nezavisnih kolona ($\text{rank}(\mathbf{W}) = N$) tada mora biti $\mathbf{W}^T\mathbf{W} = \mathbf{I}$ jer je matrica \mathbf{F} realna i simetrična. Očigledno da ista analiza može biti sprovedena i za jednačinu (6.15).

Sada će biti pokazano da rešenje \mathbf{W} ima *range* (potprostor definisan kolonama matrice \mathbf{W}) koji je ekvivalentan potprostoru koji je definisan sa N sopstvenih vektora matrice \mathbf{C} koje odgovaraju najvećim sopstvenim vrednostima, pod pretpostavkom da (6.16) važi aproksimativno (drugim rečima $\mathbf{W}^T\mathbf{W} \cong \mathbf{I}$).

Može se uočiti da se jednačina (6.14) može predstaviti sa

$$\begin{aligned} \mathbf{W}(i+1) &= \mathbf{W}(i) + \gamma(i)\xi(i)(\mathbf{x}(i)\mathbf{x}(i)^T) \mathbf{W}(i) - \mathbf{W}(i)\theta(i), \\ \xi(i) &= \mathbf{x}(i)^T \mathbf{x}(i) - \mathbf{y}(i)^T \mathbf{y}(i) = \text{tr} \left(\mathbf{x}(i)\mathbf{x}(i)^T - \mathbf{y}(i)\mathbf{y}(i)^T \right), \end{aligned} \quad (6.18)$$

što predstavlja Plamblijevu generalnu stohastičku aproksimaciju [Plumbley, 1991] sa izuzetkom člana $\xi(i)$. Biće pokazano da bilo koji algoritam iz klase (6.18) dovodi do identičnih uslova za konvergenciju ortogonalne projekcije $\mathbf{P}(i)$ na *range* od $\mathbf{W}(i)$, bez obzira na pojedinačni izbor $\theta(i)$, pod uslovom da je $\text{rank}(\mathbf{W})$ maksimalan za svako i . Biće pokazano da ta ortogonalna projekcija konvergira ka projekciji \mathbf{P}_N koja je projekcija na N -dimenzioni glavni potprostor matrice \mathbf{C} .

Ortogonalna projekcija na *range* $\mathbf{W}(i)$ je (pod pretpostavkom da je $\mathbf{W}(i)$ maksimalnog ranga za svako i)

$$\mathbf{P}(i) = \mathbf{W}(i) \left(\mathbf{W}(i)^T \mathbf{W}(i) \right)^{-1} \mathbf{W}(i)^T, \quad (6.19)$$

jer je $\mathbf{P}(i)^2 = \mathbf{P}(i)$, $\mathbf{P}(i)^T = \mathbf{P}(i)$, i *range* $\mathbf{P}(i)$ i $\mathbf{W}(i)$ su isti ([Plumbley, 1991]).

Teorema 6.1 Ako je $\mathbf{W}(i)^T \mathbf{W}(i)$ invertibilna matrica i $\|\mathbf{W}(i)^T \mathbf{W}(i)\|$ i $\|(\mathbf{W}(i)^T \mathbf{W}(i))^{-1}\|$ su ograničeni za svako i , algoritam (6.18) za $\mathbf{W}(i+1)$ rezultuje sledećim algoritmom za $\mathbf{P}(i+1)$

$$\mathbf{P}(i+1) = \mathbf{P}(i) + \gamma(i) \Delta \mathbf{P}(i) + O(\gamma(i)^2), \quad (6.20)$$

gde je

$$\begin{aligned} \Delta \mathbf{P}(i) &= (\mathbf{I} - \mathbf{P}(i)) \left(\text{tr} \left((\mathbf{I} - \mathbf{W}(i) \mathbf{W}(i)^T) \mathbf{C}(i) \right) \mathbf{C}(i) \right) \mathbf{P}(i) \\ &\quad + \mathbf{P}(i) \left(\text{tr} \left((\mathbf{I} - \mathbf{W}(i) \mathbf{W}(i)^T) \mathbf{C}(i) \right) \mathbf{C}(i) \right) (\mathbf{I} - \mathbf{P}(i)), \\ \mathbf{C}(i) &= \mathbf{x}(i) \mathbf{x}(i)^T. \end{aligned} \quad (6.21)$$

Dokaz Dokaz će biti izveden po sličnom postupku koji je primenjen za dokaz teoreme 3.3.1 u [Plumbley, 1991].

Napišimo (6.20) kao

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \Delta \mathbf{W}(i) + O(\gamma(i)^2)$$

gde je

$$\Delta \mathbf{W}(i) = \xi(i) (\mathbf{x}(i) \mathbf{x}(i)^T \mathbf{W}(i) - \mathbf{W}(i) \boldsymbol{\theta}(i)).$$

Sada označimo $\mathbf{R}(i) = \mathbf{W}(i)^T \mathbf{W}(i)$, tako da sledi

$$\begin{aligned} \mathbf{R}(i+1) &= \mathbf{W}(i+1)^T \mathbf{W}(i+1) \\ &= \left(\mathbf{W}(i) + \gamma(i) \Delta \mathbf{W}(i) + O(\gamma(i)^2) \right)^T \left(\mathbf{W}(i) + \gamma(i) \Delta \mathbf{W}(i) + O(\gamma(i)^2) \right) \\ &= \mathbf{R}(i) + \gamma(i) \Delta \mathbf{R}(i) + O(\gamma(i)^2), \end{aligned}$$

gde je

$$\Delta \mathbf{R}(i) = \mathbf{W}(i)^T \Delta \mathbf{W}(i) + (\Delta \mathbf{W}(i))^T \mathbf{W}(i).$$

Ako pretpostavimo da važi

$$\gamma(i) \|\mathbf{R}(i)^{-1}\| \ll 1,$$

za svako i ,

$$\begin{aligned}\mathbf{R}(i+1)^{-1}\mathbf{R}(i+1) &= \mathbf{I} \\ &= \mathbf{R}(i+1)^{-1}\left(\mathbf{R}(i) + \gamma(i)\Delta\mathbf{R}(i) + O(\gamma(i)^2)\right)\end{aligned}$$

što rezultuje sa

$$\mathbf{R}(i+1)^{-1} = \mathbf{R}(i)^{-1} - \gamma(i)\mathbf{R}(i)^{-1}(\Delta\mathbf{R}(i))\mathbf{R}(i)^{-1} + O(\gamma(i)^2),$$

a to dovodi do sledeće jednačine

$$\begin{aligned}\mathbf{P}(i+1) &= \mathbf{W}(i+1)\left(\mathbf{W}(i+1)^T\mathbf{W}(i+1)\right)^{-1}\mathbf{W}(i+1)^T \\ &= \left(\mathbf{W}(i) + \gamma(i)\Delta\mathbf{W}(i) + O(\gamma(i)^2)\right)\mathbf{R}(i+1)^{-1}\left(\mathbf{W}(i) + \gamma(i)\Delta\mathbf{W}(i) + O(\gamma(i)^2)\right)^T \\ &= \mathbf{P}(i) + \gamma(i)\Delta\mathbf{P}(i) + O(\gamma(i)^2),\end{aligned}$$

gde je

$$\begin{aligned}\Delta\mathbf{P}(i) &= \left(\mathbf{I} - \mathbf{W}(i)\mathbf{R}(i)^{-1}\mathbf{W}(i)^T\right)\Delta\mathbf{W}(i)\mathbf{R}(i)^{-1}\mathbf{W}(i)^T \\ &\quad + \mathbf{W}(i)\mathbf{R}(i)^{-1}\Delta\mathbf{W}(i)^T\left(\mathbf{I} - \mathbf{W}(i)\mathbf{R}(i)^{-1}\mathbf{W}(i)^T\right) \\ &= \left(\left(\mathbf{I} - \mathbf{P}(i)\right)\Delta\mathbf{W}(i)\mathbf{R}(i)^{-1}\mathbf{W}(i)^T\right) + \left(\left(\mathbf{I} - \mathbf{P}(i)\right)\Delta\mathbf{W}(i)\mathbf{R}(i)^{-1}\mathbf{W}(i)^T\right)^T.\end{aligned}$$

Sada, pošto je

$$\Delta\mathbf{W}(i) = \xi(i)\left(\mathbf{x}(i)\mathbf{x}(i)^T\mathbf{W}(i) - \mathbf{W}(i)\boldsymbol{\theta}(i)\right),$$

posle nekoliko jednostavnih transformacija dobijamo

$$\begin{aligned}\Delta\mathbf{P}(i) &= \left(\mathbf{I} - \mathbf{P}(i)\right)\left(\text{tr}\left(\left(\mathbf{I} - \mathbf{W}(i)\mathbf{W}(i)^T\right)\mathbf{C}(i)\right)\mathbf{C}(i)\right)\mathbf{P}(i) \\ &\quad + \mathbf{P}(i)\left(\text{tr}\left(\left(\mathbf{I} - \mathbf{W}(i)\mathbf{W}(i)^T\right)\mathbf{C}(i)\right)\mathbf{C}(i)\right)\left(\mathbf{I} - \mathbf{P}(i)\right).\end{aligned}$$

Ovim je teorema dokazana.

Primenom stohastičke aproksimacije, \mathbf{P} u (6.20) i (6.21) može biti pridružena sledeća diferencijalna jednačina

$$\begin{aligned} \frac{d\mathbf{P}}{dt} = & (\mathbf{I} - \mathbf{P})(\text{tr}((\mathbf{I} - \mathbf{P})\mathbf{C})\mathbf{C} + 2\mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C})\mathbf{P} \\ & + \mathbf{P}(\text{tr}((\mathbf{I} - \mathbf{P})\mathbf{C})\mathbf{C} + 2\mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C})(\mathbf{I} - \mathbf{P}), \end{aligned} \quad (6.22)$$

pod pretpostavkom da važi

$$\mathbf{W}\mathbf{W}^T \approx \mathbf{P}. \quad (6.23)$$

Pod pretpostvakama koje su analizirane na početku ove sekcije, \mathbf{P} u (6.22) i (6.23) teži skoro sigurno ka asimptotski stabilnim rešenjima jednačine (6.22).

Sada navedimo sledeću Lemu, koja će nam biti od koristi u analizi koja sledi.

Lema 6.1 ([Ogawa, 1987], [Brocket, 1991a]) Ako je \mathbf{X} $n \times n$ Hermitova matrica čiji je rang $\text{rang}(\mathbf{X})=r$ i neka je \mathbf{Q}_k neka $n \times k$ matrica, $k \leq r$, sa k ortonormalnih kolona. Tada, za dato \mathbf{X} , $\text{tr}(\mathbf{Q}_k^{\#} \mathbf{X} \mathbf{Q}_k)$ se maksimizira kada je $\mathbf{Q}_k = \mathbf{V} \mathbf{E}_k$ gde \mathbf{V} označava matricu ortogonalnih sopstvenih vektora matrice \mathbf{X} i \mathbf{E}_k predstavlja matricu sa k ortonormalnih kolona koja zadovoljava $\mathbf{E}_k \mathbf{E}_k^{\#} = \mathbf{I}_k$ (\mathbf{I}_k označava dijagonalnu matricu čijih je prvih k dijagonalnih elemenata jednako 1 dok su ostali dijagonalni elementi jednaki nuli).

Sada ćemo analizirati ponašanje diferencijalne jednačine (6.22). Prvo ćemo pokazati da važi sledeća teorema.

Teorema 6.2 U jednačini (6.22) \mathbf{P} izvodi najbrže opadajuće pretraživanje kvadrata greške u rekonstrukciji ulaznog signala, definisanog sa

$$S = \left(\mathbb{E} \left(\|\mathbf{x}(i) - \mathbf{P}\mathbf{x}(i)\|^2 \right) \right)^2. \quad (6.24)$$

Dokaz: Možemo napisati S kao:

$$S = \text{tr}^2((\mathbf{I} - \mathbf{P})\mathbf{C}). \quad (6.25)$$

Znamo da važi

$$(\mathbf{I} - \mathbf{P})^2 = (\mathbf{I} - \mathbf{P}).$$

Kako je $\mathbf{P} = \mathbf{P}^2$, imamo

$$\frac{d\mathbf{P}}{dt} = \left(\frac{d\mathbf{P}}{dt}\right)\mathbf{P} + \mathbf{P}\left(\frac{d\mathbf{P}}{dt}\right),$$

tako

$$(\mathbf{I} - \mathbf{P})\frac{d\mathbf{P}}{dt} = \left(\frac{d\mathbf{P}}{dt}\right)\mathbf{P}$$

i

$$(\mathbf{I} - \mathbf{P})\frac{d\mathbf{P}}{dt} = (\mathbf{I} - \mathbf{P})\left(\frac{d\mathbf{P}}{dt}\right)\mathbf{P}$$

Sada je jednostavno uočiti da važi

$$\frac{d\mathbf{P}}{dt} = (\mathbf{I} - \mathbf{P})\frac{d\mathbf{P}}{dt}\mathbf{P} + \mathbf{P}\frac{d\mathbf{P}}{dt}(\mathbf{I} - \mathbf{P}). \quad (6.26)$$

Tada važi

$$\frac{dS}{dt} = -2\text{tr}((\mathbf{I} - \mathbf{P})\mathbf{C})\text{tr}\left(\frac{d\mathbf{P}}{dt}\mathbf{C}\right). \quad (6.27)$$

Na osnovu jednačina (6.22) i (6.24) i posle nekoliko transformacija dobijamo

$$\begin{aligned} \frac{dS}{dt} &= -4\text{tr}^2((\mathbf{I} - \mathbf{P})\mathbf{C})\text{tr}((\mathbf{I} - \mathbf{P})\mathbf{C}\mathbf{P}\mathbf{C}) \\ &\quad - 8\text{tr}((\mathbf{I} - \mathbf{P})\mathbf{C})\text{tr}(\mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C}\mathbf{P}\mathbf{C}(\mathbf{I} - \mathbf{P})). \end{aligned} \quad (6.28)$$

Kako su $\mathbf{C}\mathbf{P}\mathbf{C}$ i $(\mathbf{I} - \mathbf{P})\mathbf{C}\mathbf{P}\mathbf{C}(\mathbf{I} - \mathbf{P})$ realne simetrične matrice i imajući u vidu Lemu 6.1 možemo zaključiti da

$$\frac{dS}{dt} \leq 0 \quad (6.29)$$

pri čemu se jednakost postiže samo u slučaju $\mathbf{CP}=\mathbf{PC}$ (što takođe znači i da je $d\mathbf{P}/dt=0$). Možemo zaključiti da \mathbf{P} i \mathbf{C} komutuju u stacionarnim tačkama algoritma (6.28) koje su u isto vreme i stacionarne tačke algoritma (6.22). To znači da \mathbf{P} ima iste sopstvene vektore kao i \mathbf{C} , što znači da \mathbf{P} mora biti ortogonalna projekcija na potprostor koji je definisan nekim od sopstvenih vektora matrice \mathbf{C} .

Korišćenjem informacionog kapaciteta sistema (u slučaju da smatramo da je sferni Gausov šum prisutan na ulazu) i odgovarajuće funkcije Ljapunova, pokazaćemo da je potprostor koji će biti pokriven u stvari potprostor koji je definisan dominantnim sopstvenim vektorima matrice \mathbf{C} . Metod za dokaz je veoma sličan metodu koji je korišćen u [Plumbley, 1991].

Informacioni kapacitet na izlazu, od interesa za naš slučaj, je definisana kao

$$\begin{aligned} I_c &= I_c(\mathbf{Y};\Omega) \\ &= \frac{1}{2} \left(\log(\det(\mathbf{W}^T \mathbf{C} \mathbf{W})) - \log(\det(\sigma_{\Phi_x}^2 \mathbf{W}^T \mathbf{W})) \right), \end{aligned} \quad (6.30)$$

gde \mathbf{Y} predstavlja izlaz mreže, Ω sferni aditivni Gausov šum na izlazu, σ_{Φ_x} je varijansa ulaznog šuma. Sada možemo predstaviti sledeću teoremu:

Teorema 6.3. Funkcija $(\max_{\mathbf{P}} I_c) - I_c$ je funkcija Ljapunova za algoritam (6.22). Algoritam će biti stabilan u tačkama u kojima je \mathbf{P} projekcija na neke od sopstvenih vektora matrice \mathbf{C} .

Dokaz Korišćenjem identiteta

$$\frac{d}{dt} \det(\mathbf{V}) = \text{tr} \left(\text{adj}(\mathbf{V}) \frac{d\mathbf{V}}{dt} \right), \quad (6.31)$$

gde je $\text{adj}(\mathbf{V})$ adjungovana matrica matrice \mathbf{V} , dobijamo

$$\frac{d}{dt} \log(\det(\mathbf{V})) = \frac{1}{\det(\mathbf{V})} \text{tr} \left(\text{adj}(\mathbf{V}) \frac{d\mathbf{V}}{dt} \right) = \text{tr} \left(\mathbf{V}^{-1} \frac{d\mathbf{V}}{dt} \right) \quad (6.32)$$

pod uslovom da je \mathbf{V} invertibilna matrica. Nalaženjem izvoda I_c u (6.30) imamo (posle nekoliko jednostavnih transformacija)

$$\frac{dI_c}{dt} = \text{tr} \left(\left(\mathbf{W}^T \mathbf{C} \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{C} (\mathbf{I} - \mathbf{P}) \frac{d\mathbf{W}}{dt} \right). \quad (6.33)$$

Pošto je $\mathbf{P}\mathbf{W} = \mathbf{W}$ dobijamo

$$\frac{d\mathbf{P}}{dt} \mathbf{W} + \mathbf{P} \frac{d\mathbf{W}}{dt} = \frac{d\mathbf{W}}{dt} \quad (6.34)$$

$$\frac{d\mathbf{P}}{dt} \mathbf{W} = (\mathbf{I} - \mathbf{P}) \frac{d\mathbf{W}}{dt} \quad (6.35)$$

i to dalje znači da važi

$$\frac{dI_c}{dt} = \text{tr} \left(\left(\mathbf{W}^T \mathbf{C} \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{C} \frac{d\mathbf{P}}{dt} \mathbf{W} \right). \quad (6.36)$$

Možemo jednostavno zaključiti da poslednja jednačina zavisi samo od forme algoritma za matricu \mathbf{P} , a ne i od forme algoritma za matricu \mathbf{W} . Zamenom (6.22) u (6.36) dobijamo

$$\begin{aligned} \frac{dI_c}{dt} = & \text{tr} \left(\left(\mathbf{W}^T \mathbf{C} \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{C} \mathbf{P} \mathbf{W} \right) \text{tr} \left((\mathbf{I} - \mathbf{P}) \mathbf{C} \right) \\ & + 2 \text{tr} \left(\left(\mathbf{W}^T \mathbf{C} \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{C} (\mathbf{I} - \mathbf{P}) \mathbf{C} (\mathbf{I} - \mathbf{P}) \mathbf{C} \mathbf{P} \mathbf{W} \right). \end{aligned} \quad (6.37)$$

Posle par transformacija to dovodi do

$$\begin{aligned} \frac{dI_c}{dt} = & \text{tr} \left((\mathbf{I} - \mathbf{P}) \mathbf{C} \right) \text{tr} \left((\mathbf{I} - \mathbf{P}) \mathbf{C} \mathbf{W} \left(\mathbf{W}^T \mathbf{C} \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{C} (\mathbf{I} - \mathbf{P}) \right) \\ & + \text{tr} \left(\mathbf{C} \left((\mathbf{I} - \mathbf{P}) \mathbf{C} \mathbf{W} \left(\mathbf{W}^T \mathbf{C} \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{C} (\mathbf{I} - \mathbf{P}) \right) \right) \geq 0, \end{aligned} \quad (6.38)$$

pri čemu se jednakost ostvaruje samo u slučaju da važi $(\mathbf{I} - \mathbf{P})\mathbf{C} = 0$ (pošto je $(\mathbf{W}^T \mathbf{C} \mathbf{W})^{-1}$ pozitivno definitna). Znači, predloženi zakon učenja, kada se razmatra u kontinualnom domenu, dovodi do toga da je informacioni kapacitet sistema neopadajuća funkcija u

vremenu, i dostiže stacionarno stanje samo u slučaju kada je \mathbf{P} projekcija na neke od sopstvenih vektora matrice \mathbf{C} .

Nažalost, to nam još uvek ne ukazuje na to koje su od stacionarnih tačaka stabilne i koji je njihov domen privlačenja. To znači da moramo postići “snažniji” rezultat. Taj rezultat se bazira na činjenici da \mathbf{P} nikada neće pronaći potprostor koji je definisan sa N dominantnih sopstvenih vektora matrice \mathbf{C} ako nema komponente u pravcu svake od njih na početku. Drugim rečima, ako je početna vrednost $\det(\mathbf{W}^T \mathbf{Q}_N \mathbf{W}) = 0$, gde je \mathbf{Q}_N projekcija na N -dimenzioni glavni sopstveni potprostor matrice \mathbf{C} , predložena klasa algoritama neće pronaći dominantni sopstveni potprostor. Ovde \mathbf{Q}_N predstavlja jednu vrstu pseudo-kovarijansne matrice koja daje jedinične težine za N dominantnih glavnih komponenta matrice \mathbf{C} , i težinu nula za preostalih $K-N$ komponenta.

Primetimo da pri izvođenju (6.38), nismo koristili nikakve posebne osobine matrice \mathbf{C} (osim da je matrica kvadratna i simetrična). Na osnovu (6.30) pravimo novu funkciju:

$$I_{Q_N} = \frac{1}{2} \left(\log \left(\det \left(\mathbf{W}^T \mathbf{Q}_N \mathbf{W} \right) \right) - \log \left(\det \left(\sigma_{\Phi_x}^2 \mathbf{W}^T \mathbf{W} \right) \right) \right) \quad (6.39)$$

u kojoj $\mathbf{W}^T \mathbf{Q}_N \mathbf{W}$ zamenjuje $\mathbf{W}^T \mathbf{C} \mathbf{W}$. Sada možemo predstaviti sledeću teoremu

Teorema 6.4. Funkcija $(\max_P(I_{Q_N}) - I_{Q_N})$ predstavlja Ljapunovu funkciju za algoritam (6.22). Algoritam će biti stabilan u tačkama u kojima je \mathbf{P} projekcija na glavni sopstveni potprostor matrice \mathbf{C} .

Dokaz. Ako je $\det(\mathbf{W}^T \mathbf{Q}_N \mathbf{W}) \neq 0$ imamo

$$\begin{aligned} \frac{d}{dt} I_{Q_N} &= \text{tr} \left(\left(\mathbf{W}^T \mathbf{Q}_N \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{Q}_N \frac{d\mathbf{P}}{dt} \mathbf{W} \right) \\ &= A + B, \quad \text{gde su} \\ A &= \text{tr} \left((\mathbf{I} - \mathbf{P}) \mathbf{C} \right) \left(\text{tr} \left(\left(\mathbf{W}^T \mathbf{Q}_N \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{Q}_N \mathbf{C} \mathbf{W} \right) - \text{tr} \left(\left(\mathbf{W}^T \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{C} \mathbf{W} \right) \right) \\ B &= 2 \text{tr} \left(\left(\mathbf{W}^T \mathbf{Q}_N \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{Q}_N (\mathbf{I} - \mathbf{P}) \mathbf{C} (\mathbf{I} - \mathbf{P}) \mathbf{C} \mathbf{W} \right). \end{aligned} \quad (6.40)$$

Ako imamo u vidu da je $Q_N^2 = Q_N$ i da Q_N iste sopstvene vektore kao C onda važi $Q_N C = Q_N (Q_N C) = Q_N C Q_N$. Pošto znamo da je Q_N ortogonalna projekcija na potprostor definisan sa N dominantnih sopstvenih vektora matrice C , možemo pisati $Q_N = \mathbf{q} \mathbf{q}^T$ gde je \mathbf{q} ortonormalna matrica čije su kolone N dominantnih sopstvenih vektora matrice C .

Na osnovu toga sledi $\mathbf{W}^T Q_N \mathbf{W} = \mathbf{W}^T \mathbf{q} \mathbf{q}^T \mathbf{W}$ gde je $\mathbf{W}^T \mathbf{q}$ $N \times N$ nesingularna kvadratna matrica, i

$$\det(\mathbf{W}^T \mathbf{q})^2 = \det(\mathbf{W}^T Q_N \mathbf{W}) \neq 0. \quad (6.41)$$

Tada je

$$A = \text{tr}((\mathbf{I} - \mathbf{P})\mathbf{C}) - \text{tr}(\mathbf{P}\mathbf{C}). \quad (6.42)$$

Iz prethodne analize (Lema 6.1), znamo da je $\text{tr}(\mathbf{P}\mathbf{C})$ maksimalno kad \mathbf{P} pokriva N dominantnih sopstvenih vektora matrice C , pa možemo zaključiti da je $A \geq 0$.

Takođe je (imajući u vidu (6.23))

$$\begin{aligned} B &= 2\text{tr}\left(\left(\mathbf{W}^T Q_N \mathbf{W}\right)^{-1} \mathbf{W}^T Q_N (\mathbf{I} - \mathbf{P})\mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C}\mathbf{W}\right) \\ &= 2\text{tr}\left(\mathbf{W}\left(\mathbf{W}^T Q_N \mathbf{W}\right)^{-1} \mathbf{W}^T Q_N \mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C} - \mathbf{P}\mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C}\right). \end{aligned} \quad (6.43)$$

Definišimo

$$\mathbf{P}\mathbf{1} = \mathbf{W}\left(\mathbf{W}^T Q_N \mathbf{W}\right)^{-1} \mathbf{W}^T Q_N. \quad (6.44)$$

Tada, nije teško uočiti da je $\mathbf{P}\mathbf{1}$ idempotentna matrica, pošto je $\mathbf{P}\mathbf{1}^2 = \mathbf{P}\mathbf{1}$. Imajući u vidu činjenicu da je $(\mathbf{I} - \mathbf{P})^2 = (\mathbf{I} - \mathbf{P})$, dobijamo

$$\begin{aligned} B &= 2\text{tr}((\mathbf{I} - \mathbf{P})\mathbf{C}\mathbf{P}\mathbf{1} - \mathbf{P}\mathbf{1}\mathbf{C}(\mathbf{I} - \mathbf{P})) - 2\text{tr}(\mathbf{W}^T (\mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C})\mathbf{W}) \\ &= B_1 - B_2 \end{aligned} \quad (6.45)$$

gde su

$$B_1 = 2\text{tr}((\mathbf{I} - \mathbf{P})\mathbf{C}\mathbf{P}\mathbf{1} - \mathbf{P}\mathbf{1}\mathbf{C}(\mathbf{I} - \mathbf{P})),$$

$$B_2 = 2\text{tr}(\mathbf{W}^T (\mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C})\mathbf{W}).$$

Nije teško zaključiti da je $B1 \geq 0$. Takođe, na osnovu Leme 6.1 znamo da je $B2$ maksimalno kada je \mathbf{W} takvo da $\mathbf{P} = \mathbf{W}\mathbf{W}^T$ pokriva N dimenzioni dominantni potprostor matrice \mathbf{C} . U tom slučaju \mathbf{P} i \mathbf{C} komutuju. Tada imamo da je maksimalna vrednost $B2$

$$B2_{\max} = 2\text{tr}(\mathbf{P}\mathbf{C}(\mathbf{I} - \mathbf{P})\mathbf{C}) = 2\text{tr}(\mathbf{C}\mathbf{P}(\mathbf{I} - \mathbf{P})\mathbf{C}) = 0. \quad (6.46)$$

To znači da je $B2 \leq 0$. Na osnovu toga zaključujemo da je (ako je $\det(\mathbf{W}^T\mathbf{Q}_N\mathbf{W}) \neq 0$)

$$\frac{d}{dt} I_{Q_N} \geq 0 \quad (6.47)$$

pri čemu se jednakost ostvaruje samo za $\mathbf{P} = \mathbf{Q}_N$.

Iz svega navedenog zaključujemo da je $(\max_P(I_{Q_N}) - I_{Q_N})$ naša funkcija Ljapunova.

Domen privlačenja je skup \mathbf{W} takav da za sve dominantne sopstvene vektore ζ_i matrice \mathbf{C} ($1 \leq i \leq N$), $\mathbf{W}^T\zeta_i \neq 0$.

Ukratko, pod pretpostavkama koje su napravljene na početku ove sekcije, pokazali smo da algoritam (6.22), a posledično i algoritam (6.17), konvergira ka \mathbf{W} koje pokriva N -dimenzioni dominantni potprostor matrice \mathbf{C} , pod uslovom da matrica $\mathbf{W}(i)$ ima maksimalni mogući rang za svako i .



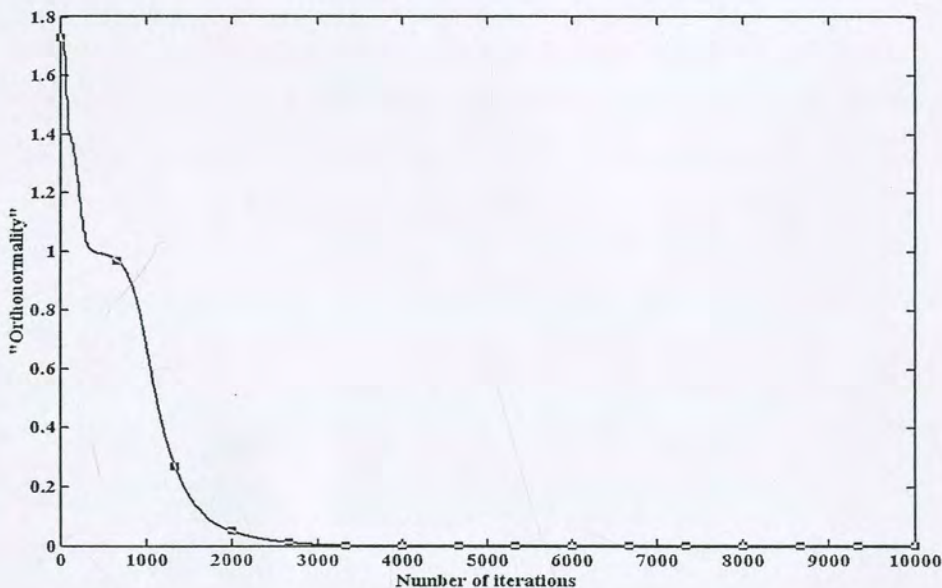
Slučaj 4.7. Mapa privlačnosti i konvergencija na svoj dominantni vektor linearnih korelacionih komponenti

6.5 Rezultati simulacija

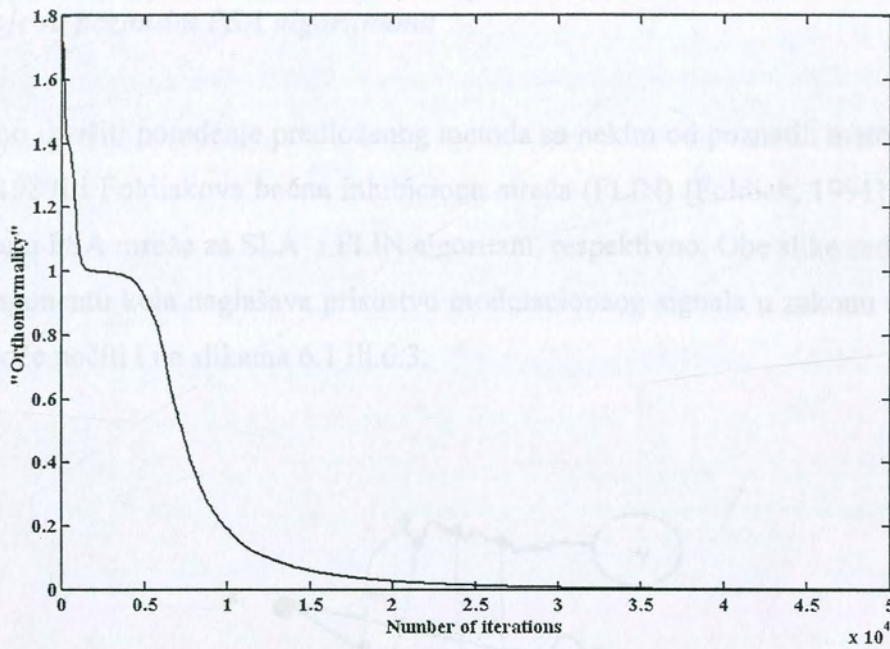
Sada ćemo prikazati rezultate simulacija za mali sistem. Broj ulaza je $K = 5$ a broj izlaznih neurona je $N = 3$. Ulazni vektor sa srednjom vrednošću nula i nekorelisanim komponentama je definisan sledećim jednačinama:

$$\begin{aligned} \mathbf{x}(1,i) &= .9 * \sin(i/2); \\ \mathbf{x}(2,i) &= .9 * ((\text{rem}(i,23) - 11)/9).^5; \\ \mathbf{x}(3,i) &= .9 * ((\text{rem}(i,27) - 13)/9); \\ \mathbf{x}(4,i) &= .9 * ((\text{rand}(1,1) < .5) * 2 - 1) * \log(\text{rand}(1,1) + .5); \\ \mathbf{x}(5,i) &= -.5 + \text{rand}(1,1); \end{aligned}$$

Slika 6.3 prikazuje kako se menja “ortonormalnost” matrice \mathbf{W} u odnosu na broj iteracija za MHO algoritam (inicijalna vrednost matrice \mathbf{W} je $\mathbf{W}_{init} = 0.1 * \text{rand}(5,3)$; usvojen je konstantan faktor učenja $\gamma = 0.015$). Rezultati predstavljeni na slici 6.4 prikazuju kako se menja ortonormalnost matrice \mathbf{W} u odnosu na broj iteracija za MHO algoritam pri čemu komponente uzlaznog vektora nisu dekorelisane već predstavljaju linearnu kombinaciju komponenta vektora $\mathbf{x}_m(i) = \mathbf{mix} * \mathbf{x}(i)$ (inicijalna vrednost matrice \mathbf{W} je $\mathbf{W}_{init} = 0.1 * \text{rand}(5,3)$; matrica koja definiše linearnu kombinaciju je $\mathbf{mix} = -.15 + .3 * \text{rand}(5,5)$; usvojen je konstantan faktor učenja $\gamma = 4.0$). Ortonormalnost matrice \mathbf{W} je definisana kao $\|\mathbf{I} - \mathbf{W}^T \mathbf{W}\|_F$, gde F predstavlja Frobenijusovu normu.



Slika 6.3. Mera ortonormalnosti u odnosu na broj iteracija (ulazni vektor ima dekorelisane komponente)

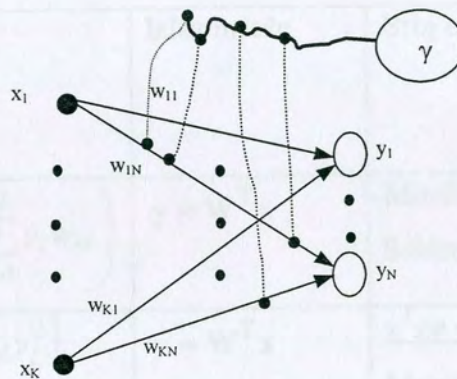


Slika 6.4 Mera ortonormalnosti u odnosu na broj iteracija
(komponente ulaznog vektora su međusobno korelisane)

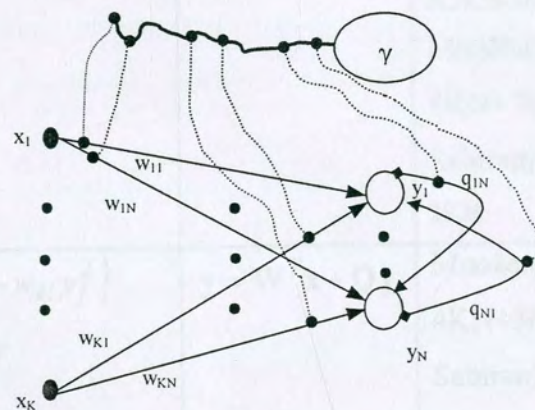
Može se uočiti da algoritam brže konvergira u slučaju kada su komponente ulaznog vektora dekokorelisane.

6.6 Poređenje sa poznatim PSA algoritmima

Sada ćemo izvršiti poređenje predloženog metoda sa nekim od poznatih metoda kao što su SLA [Oja, 1989] i Foldiakova bočna inhibiciona mreža (FLIN) [Foldiak, 1991]. Sledeće dve slike prikazuju PSA mreže za SLA i FLIN algoritam, respektivno. Obe slike sadrže “paukova mreža” komponentu koja naglašava prisustvo modulacionog signala u zakonu učenja, a koji se takođe može uočiti i na slikama 6.1 ili 6.3.



Slika 6.5 SLA neuralna mreža



Slika 6.6 Foldiakova bočna inhibiciona neuralna mreža

Tabela 6.1 sadrži jednačine koje se koriste za izračunavanje vrednosti potencijala izlaznih neurona za odgovarajuću PSA mrežu, kao i jednačine za izračunavanje vrednosti pojedinih sinapsi na bazi predloženog zakona učenja za dati PSA algoritam. Individualne komponente vektora $\mathbf{x}(i)$ i $\mathbf{y}(i)$ su označene sa x_n (n -ta individualna komponenta) i y_n , dok su članovi pojedinih matrica označeni sa w_{kl} . Faktor učenja za SLA i FLIN je $\gamma = \gamma(i)$, dok je za MHO slučaj $\gamma = \gamma(i)(\mathbf{x}^T \mathbf{x} - \mathbf{y}^T \mathbf{y}) = \gamma(i, \mathbf{x}, \mathbf{y})$.

Tabela 6.1 Pregled nekih PSA algoritama

Skr.	Zakon učenja	Izlaz mreže	Broj operacija	Broj kola za globalana izračunavanja
SLA	$\Delta w_{kl} = \gamma y_l \left(x_k - \sum_{i=1}^N y_i w_{ki} \right)$	$\mathbf{y} = \mathbf{W}^T \mathbf{x}$	Množenje: 4KN Sabiranje: 2KN-N	K
MHO	$\Delta w_{kl} = \hat{\gamma} (x_k y_l - w_{kl} y_l^2)$ $\hat{\gamma} = \gamma \cdot (\mathbf{x}^T \mathbf{x} - \mathbf{y}^T \mathbf{y})$	$\mathbf{y} = \mathbf{W}^T \mathbf{x}$	<u>$\mathbf{x}^T \mathbf{x} \neq \text{CONST}$</u> Množenje: 4KN + K + N + 1 Sabiranje: 2KN + K - 1 <u>$\mathbf{x}^T \mathbf{x} = \text{CONST}$</u> Množenje: 4KN + N + 1 Sabiranje: 2KN	2
FLIN	$\Delta w_{kl} = \gamma (x_k y_l - w_{kl} y_l^2)$ $\Delta q_{jp} = -\gamma y_j y_p$	$\mathbf{y} = \mathbf{W}^T \mathbf{x} + \mathbf{Qy}$	Množenje: 4KN + 3N(N-1)/2 Sabiranje: 2KN + N(N-3)	0

Poređenjem sa SLA algoritmom možemo videti da MHO algoritam ne zahteva podatak o eksplicitnoj vrednosti sinapse koja je vezana za bilo koji drugi neuron. Takođe, broj kola za globalna izračunavanja je 2 u slučaju MHO, a K u slučaju SLA algoritma. Jedina globalna

izračunavanja koja su potrebna u slučaju MHO algoritma su izračunavanja energije koja se sadrži u ulaznom i izlaznom signalu. Sa druge strane, broj izračunavanja koje zahteva SLA algoritam je manji od broja kalkulacija koji se zahteva za realizaciju MHO algoritma.

U slučaju FLIN algoritma sva izračunavanja koja su vezana za modifikaciju sinaptičkih vrednosti su strogo lokalnog karaktera. Cena koja je plaćena za lokalnost izračunavanja je: prisustvo anti-Hebove mreže koja povezuje sve izlazne neurone kao i povećanje broja operacija koje je neophodno za izračunavanje vrednosti izlaza. Nije teško uočiti da realizacija MHO algoritma zahteva manje memorijskih lokacija nego u slučaju FLIN algoritma (FLIN algoritam zahteva dodatne memorijske elemente za čuvanje elemenata matrice Q). Takođe, u mnogim slučajevima, realizacija MHO algoritma zahteva manju složenost izračunavanja (ili hardvera) nego realizacija FLIN algoritma. To važi za slučajeve kada ne postoji veoma značajna razlika u broju ulaznih i izlaznih neurona ($N^2 > K$). U slučaju da se energija ulaznog signala održava konstantnom na nekom određenom nivou, uz pomoć nekakvog spoljnog kola, MHO algoritam uvek ima manju kompleksnost izračunavanja nego FLIN algoritam bez obzira na broj izlaznih neurona. To se jasno može uočiti iz Tabele 6.1.

Interesantno je zapaziti da MHO algoritam u formi koja je predložena, u stvari obavlja Ojin zakon učenja na pojedinačnim sinaptičkim vektorima uz samo jednu razliku – faktor učenja γ je specifično “programiran” od strane same mreže. U ovom slučaju γ ima inherentnu osobinu da se smanjuje kako vreme odmiče (u slučaju MH algoritma faktor učenja teži nuli). U slučaju SLA i FLIN algoritma faktor učenja γ ne poseduje tu osobinu.

Sada ćemo izvršiti poređenje gore pomenutih algoritama na primeru male neuralne mreže. Broj ulaza je $K = 5$, a broj izlaznih neurona je $N = 3$. Ulazni vektor sa srednjom vrednošću nula i nekorelisanim komponentama je definisan sledećim jednačinama:

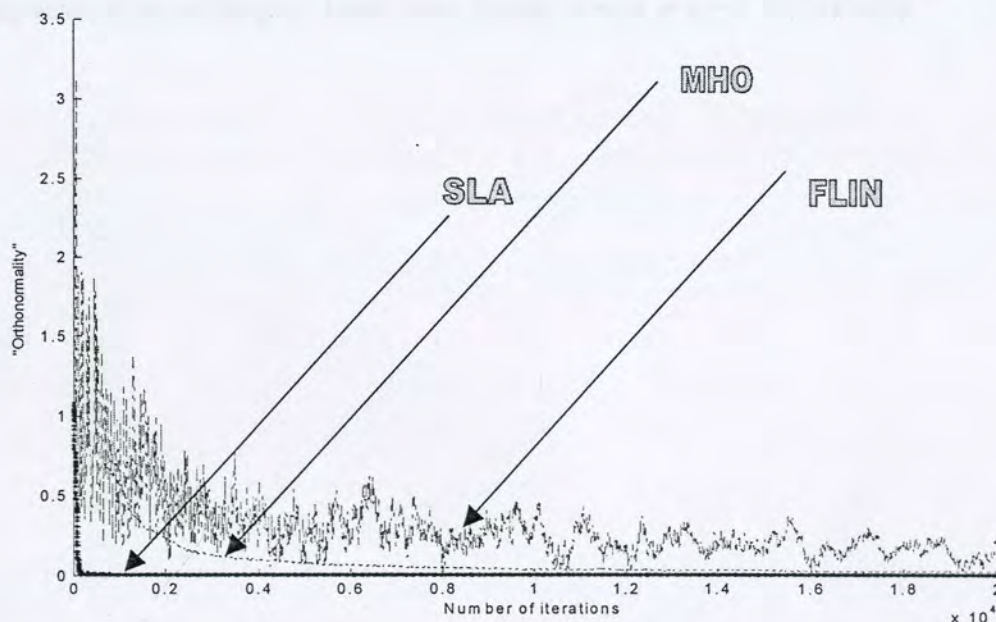
$$\begin{aligned} \mathbf{x}(1, i) &= .45 * \sin(i/2); \\ \mathbf{x}(2, i) &= .45 * ((\text{rem}(i, 23) - 11)/9).^5; \\ \mathbf{x}(3, i) &= .145 * ((\text{rem}(i, 27) - 13)/9); \\ \mathbf{x}(4, i) &= .145 * ((\text{rand}(1, 1) < .5) * 2 - 1) * \log(\text{rand}(1, 1) + .5); \\ \mathbf{x}(5, i) &= -.5 + \text{rand}(1, 1). \end{aligned}$$

U svim primerima koji slede inicijalna vrednost matrice \mathbf{W} je $\mathbf{W}_{\text{init}} = -.5 + \text{rand}(5, 3)$. Ortonormalnost matrice \mathbf{W} je korišćena kao mera brzine konvergencije algoritma. Ortonormalnost matrice \mathbf{W} je definisana sa $\|\mathbf{I} - \mathbf{W}^T \mathbf{W}\|_F$, gde F predstavlja Frobeniusovu normu. Rezultati koji će biti prikazani predstavljaju karakteristično ponašanje algoritama.

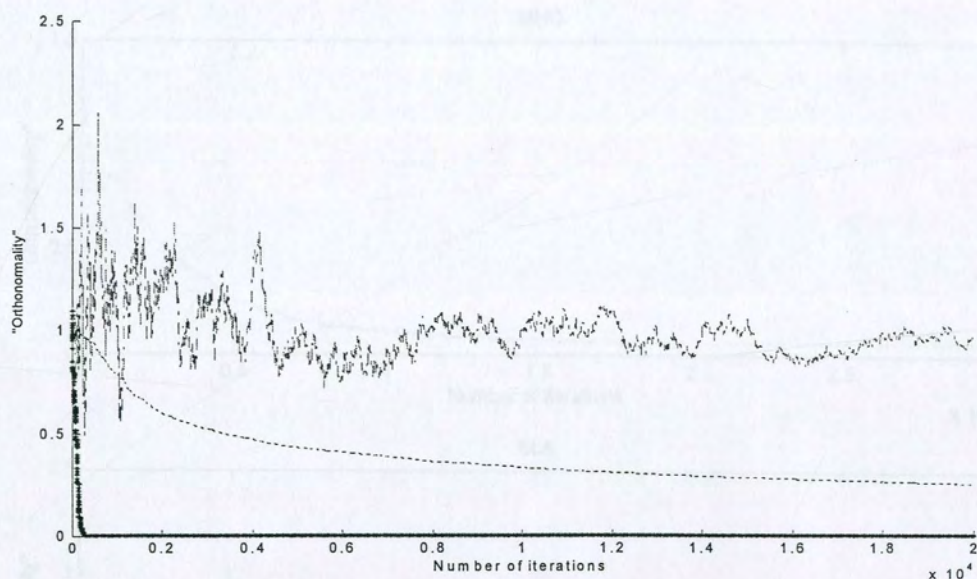
Slika 6.7 pokazuje kako se menja ortonormalnost matrice \mathbf{W} sa brojem iteracija, za SLA, FLIN i MHO algoritme. Izabrani faktor učenja je $\gamma_i = 0.8/(2+i/400)$. U posmatranom slučaju komponente ulaznog vektora su međusobno nekorelisane. Sa slike se može zaključiti da je SLA algoritam najbrži, dok MHO i FLIN algoritam konvergiraju približno istom brzinom.

Na slici 6.8 ulazni vektor je $\mathbf{xm}=\mathbf{mix}*\mathbf{x}$, gde je \mathbf{mix} matrica za mešanje definisana sa $\mathbf{mix} = -.5 + \text{rand}(5,5)$. Jasno, u ovom slučaju komponente ulaznog vektora su međusobno korelisane. Izabrani faktor učenja je $\gamma_i = 0.8/(2+i/400)$. Sa slike jasno da je SLA algoritam najbrži, dok je FLIN algoritam najsporiji.

Slike 6.9 i 6.10 sadrže rezultata simulacije za slučaj kada je $\mathbf{xm}=\mathbf{mix}*\mathbf{x}$, $\mathbf{mix} = -.5 + \text{rand}(5,5)$. Faktor učenja je konstantna $\gamma_i = 2.3$, u prvih 18000 iteracija, a nakon toga se postavlja na $\gamma_i = 0.8/(2+(i-18000)/400)$. Sa slika se jasno može uočiti da SLA i FLIN algoritam neće konvergirati ka stabilnom rešenju sve dok je faktor učenja konstantan. Sa druge strane, MHO algoritam će konvergirati ka stabilnom rešenju čak i slučaju kada je faktor učenja konstantan.

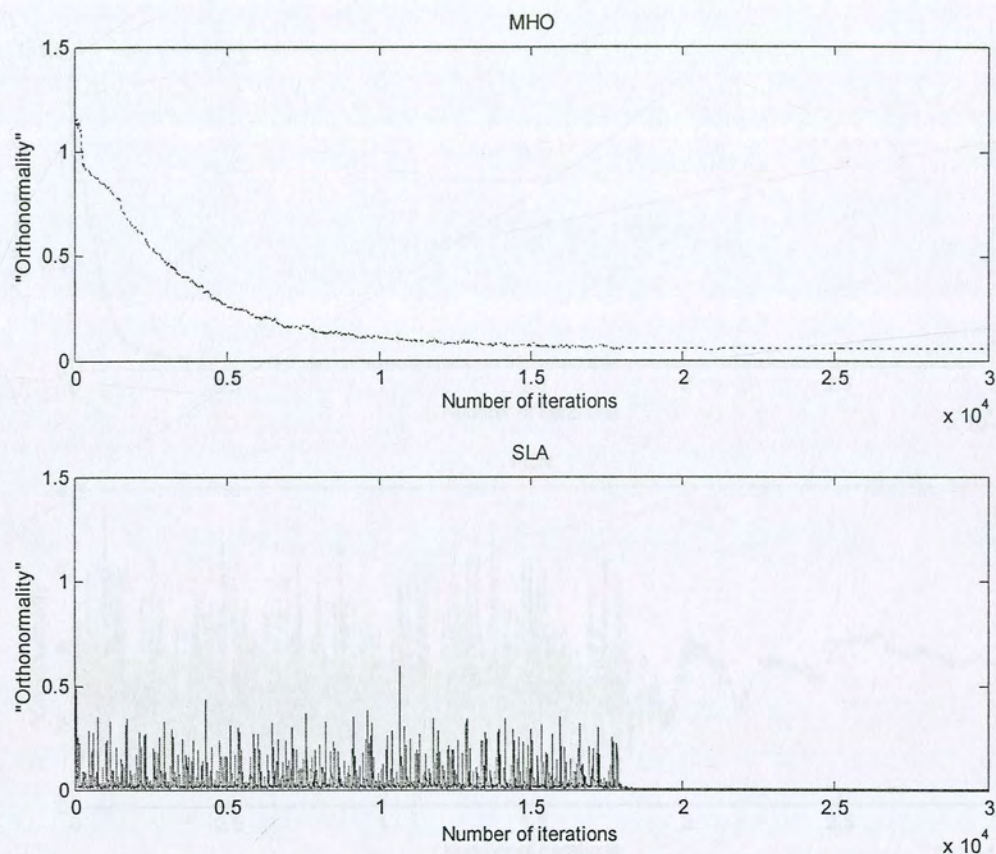


Slika 6.7 “Orthonormalnost” u zavisnosti od broja iteracija (tačkasta linija – MHO algoritam; isprekidana linija – FLIN algoritam; debela puna linija – SLA algoritam). Komponente ulaznog vektora su međusobno nekorelisane. Faktor učenja je $\gamma_i = 0.8/(2+i/400)$.



Slika 6.8 “Ortonormalnost” u zavisnosti od broja iteracija (tačkasta linija – MHO algoritam; isprekidana linija – FLIN algoritam; debela puna linija – SLA algoritam). Komponente ulaznog vektora su međusobno korelisane. Faktor učenja je $\gamma_i = 0.8/(2+i/400)$.

Slika 6.9 “Ortonormalnost” u zavisnosti od broja iteracija. Komponente ulaznog vektora su međusobno korelisane. Faktor učenja je $\gamma_i = 2.3$, za $i < 18000$, i $\gamma_i = 0.8/(2+(i-18000)/400)$ za $i > 18000$. Gornja polovina slike prikazuje ponašanje MHO algoritma. Donja polovina slike prikazuje ponašanje SLA algoritma.



Slika 6.9 “Ortonormalnost” u zavisnosti od broja iteracija. Komponente ulaznog vektora su međusobno korelisane. Faktor učenja je $\gamma_i = 2.3$, za $i < 18000$, i $\gamma_i = 0.8/(2+(i-18000)/400)$ za $i \geq 18000$. Gornja polovina slike predstavlja ponašanje MHO algoritma. Donja polovina slike prikazuje ponašanje SLA algoritma.

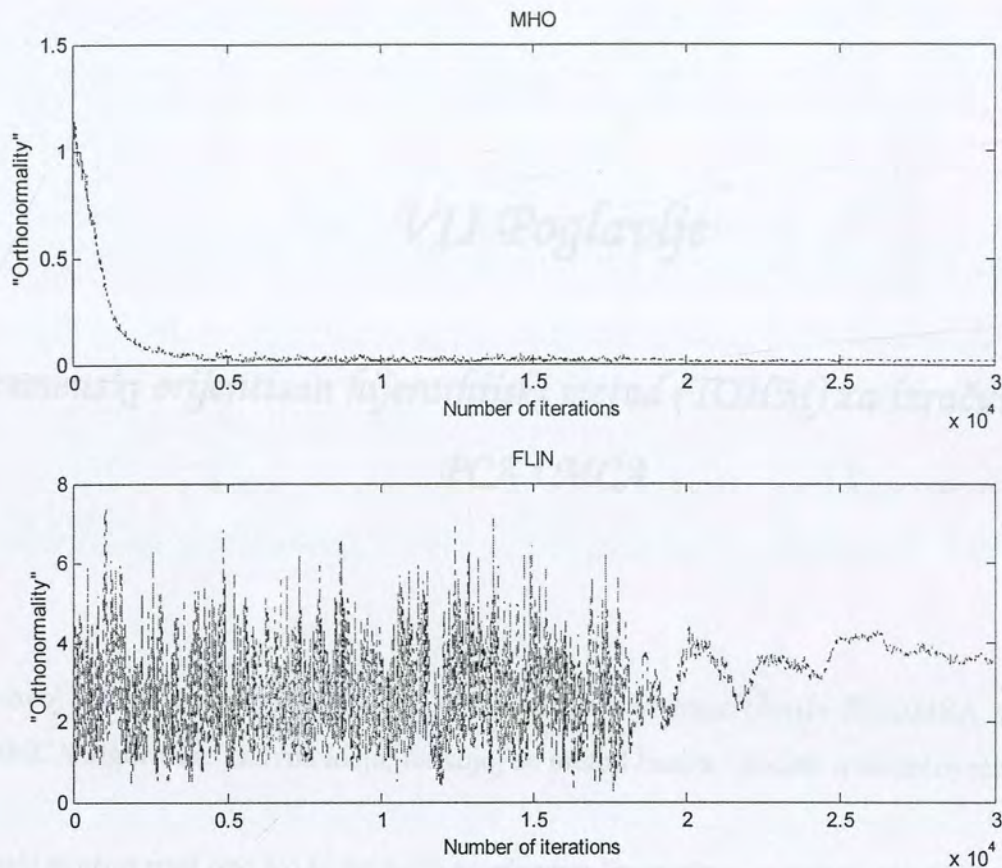
Na osnovu prethodne slike i empirijskih rezultata zaključuje se da je SLA algoritam najbrži, najjeftiniji i najlakše implementirati, ali je drugi algoritam. Najveći broj kole za globalna iznalaženja i ne konvergira ka stabilnoj rešenju u slučaju kada je faktor učenja konstantan.

SLA algoritam ne zahteva globalna iznalaženja, ali je sporiji od SLA i MHO algoritama, i ne konvergira ka stabilnom rešenju kada je faktor učenja konstantan.

MHO algoritam zahteva samo 1 kole za globalna iznalaženja, sporiji je od SLA algoritma i stabilniji je dok i u slučaju kada je faktor učenja konstantan.

SLA algoritam ne zahteva globalna iznalaženja, ali je sporiji od SLA i MHO algoritama, i ne konvergira ka stabilnom rešenju kada je faktor učenja konstantan.

MHO algoritam zahteva samo 1 kole za globalna iznalaženja, sporiji je od SLA algoritma i stabilniji je dok i u slučaju kada je faktor učenja konstantan.



Slika 6.10 “Ortonormalnost” u zavisnosti od broja iteracija Komponente ulaznog vektora su međusobno korelisane. Faktor učenja je $\gamma_i = 2.3$, za $i < 18000$, i $\gamma_i = 0.8/(2+(i-18000)/400)$ za $i \geq 18000$. Gornja polovina slike predstavlja ponašanje MHO algoritma. Donja polovina slike prikazuje ponašanje FLIN algoritma.

Na osnovu prethodne analize i simulacionih rezultata možemo zaključiti sledeće:

- SLA algoritam je najbrži, njegova realizacija zahteva najmanji broj operacija, ali sa druge strane, zahteva veliki broj kola za globalna izračunavanja i neće konvergirati ka stabilnom rešenju u slučaju kada je faktor učenja konstantan.
- FLIN algoritam ne zahteva globalna izračunavanja, ali je sporiji od the SLA i MHO algoritama, i ne konvergira ka stabilnom rešenju kada je faktor učenja konstantan.
- MHO algoritam zahteva samo 2 kola za globalna izračunavanja, sporiji je od SLA algoritma i stabilan je čak i u slučaju kada je faktor učenja konstantan.

VII Poglavlje

Vremenski orijentisan hijerarhijski metod (TOHM) za izračunavanje PCA i MCA

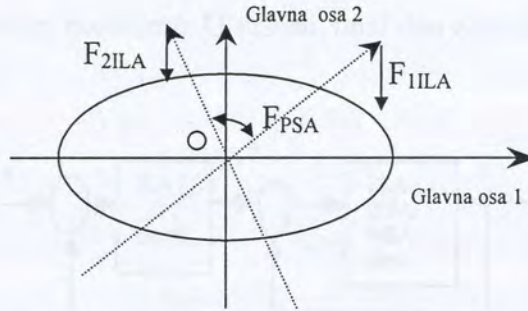
U ovoj sekciji će biti predstavljen opšti metod koji transformiše PSA/MSA algoritme u PCA/MCA algoritme. Glavna ideja, na kojoj se metod bazira, je data u sledećoj rečenici:

Svaki neuron radi ono što je najbolje za njegovu “porodicu” a potom ono što je najbolje za njega samog.

Ovu ideju ćemo nazvati “porodični princip”. Algoritam se sastoji od dva dela: jedan deo je odgovoran za ostvarivanje porodičnog cilja (*family-desirable feature learning*), a drugi je odgovoran za ostvarivanje želja pojedinačnih neurona (*individual-neuron-desirable feature learning*). Drugi deo zakona učenja se uzima sa težinskim koeficijentom koji je po apsolutnoj vrednosti manji od 1. To drugim rečima znači da se u algoritam uvodi vremenski orijentisana hijerarhija kod realizacije “porodičnog” i “individualnog” dela zakona učenja.

Na slici 7.1 je prikazana gustina verovatnoće multivarijabilnog Gausovog signala. Poznato je da su u tom slučaju glavne ose hiperelipsoida paralelne sopstvenim vektorima kovarijanske matrice ulaznog signala kada je srednja vrednost ulaznog signala nula.

U slučaju kada se primenjuje PSA/MSA algoritam, sinaptički vektori, koji su na slici prikazani isprekidanim linijama, se ne poklapaju sa osama hiperelipsoida. Oni su za izvestan ugao rotirani u odnosu na njih. Vektori su međusobno ortonormalni pod uticajem “sile” PSA algoritma – ta “sila” je na slici označena sa F_{PSA} . Sada se postavlja pitanje koju dodatnu “silu” treba primeniti da bi se sinaptički vektori dodatno rotirali do osa hiperelipsoida. Ovde će biti predloženo da se u PSA/MSA algoritam doda još jedan član. Taj novi član bi imao zadatak da doda “diferencijalni momenat” koji je na slici označen “silama” F_{1ILA} i F_{2ILA} .



Slika 7.1 Gustina verovatnoće multivariabilnog Gausovog signala

Za realizaciju “porodičnog principa” predložimo sledeći opšti metod, koji transformiše PSA/MSA algoritam, označen sa $FLA_{PSA/MSA}$ u PCA/MCA algoritam, označen sa $LA_{PCA/MCA}$:

$$LA_{PCA/MCA} = FLA_{PSA/MSA} + \alpha ILA \left(\max \left(E \left\{ \left(\mathbf{y}^T \mathbf{y} \right) \left| \begin{array}{l} \mathbf{w}_k^T \mathbf{w}_k = 1, \\ k = 1, 2, \dots, N \end{array} \right. \right\} \right) \right), \quad (7.1)$$

gde je α konstanta takva da važi $|\alpha| < 1$. ILA označava individualni deo zakona učenja. ILA predstavlja algoritam za maksimizaciju $E(\mathbf{y}^T \mathbf{y})$ pod ograničenjem $\mathbf{w}_k^T \mathbf{w}_k = 1$ for $k=1, 2, \dots, N$. Drugim rečima, novi PCA/MCA algoritam učenja ima sledeću formu

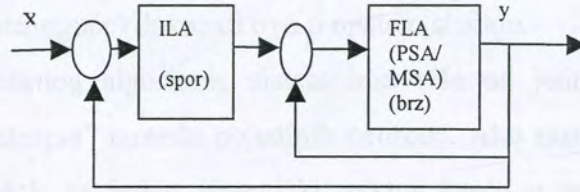
$$\Delta \mathbf{W}_{PCA/MSA} = \Delta \mathbf{W}_{FLA} + \alpha \Delta \mathbf{W}_{ILA}, \quad (7.2)$$

gde $\Delta \mathbf{W}_{FLA}$ označava doprinos porodičnog dela zakona učenja, dok $\Delta \mathbf{W}_{ILA}$ označava doprinos individualnog dela zakona učenja. Lako je uočiti da ako se koristi homogeni PSA/MSA algoritam rezultujući PCA/MCA algoritam će takođe biti homogen.

Pošto je $|\alpha| < 1$, porodični deo zakona učenja se primenjuje brže nego individualni deo zakona učenja. Jednačina (7.2) je uprošćeno prikazana na slici 7.2.

Iako je usvojeno $|\alpha| < 1$, sada ćemo se pozabaviti uprošćenom analizom u slučaju kada je $|\alpha|$ mnogo manje od 1. U tom slučaju, individualni deo jednačine (7.2) skoro da nema nikakvog uticaja na porodični deo. Tada porodični deo obezbeđuje dostizanje glavnog/sporednog potprostora. To znači da je $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i da \mathbf{W} pokriva glavni/sporedni

potprostor. Drugim rečima, usled uvođenja dve vremenske skale možemo pristupiti problemu PCA/MCA kao optimizacionom problemu. U suštini, onaj deo zakona učenja koji se odvija



Slika 7.2 Blok šema realizacije predloženog metoda

na bržoj vremenskoj skali, i koji obavlja PSA/MSA, predstavlja ograničenje za sporiji (individualni) deo zakona učenja. Znači, ako usvojimo da je PSA/MSA deo zakona učenja dovoljno brz, možemo reći da on predstavlja striktno ortonormalno ograničenje na PSA/MSA potprostoru. U tom slučaju, za individualni deo u jednačini (7.2) možemo pisati

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \left(\mathbf{x}(i)\mathbf{y}(i)^T - \mathbf{W}(i) \text{diag} \left(y_1(i)^2, \dots, y_N(i)^2 \right) \right) \alpha, \quad (7.3)$$

gde je α takvo da je $|\alpha| < 1$,

$\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i pokriva glavni/sporedni podprostor.

Odgovarajuća diferencijalna jednačina je [Ljung, 1977; Oja, 1985]

$$\frac{d\mathbf{W}}{dt} = \left(\mathbf{C}\mathbf{W} - \mathbf{W} \text{diag} \left(\mathbf{W}^T \mathbf{C}\mathbf{W} \right) \right) \alpha, \quad (7.4)$$

gde $\text{diag}(\mathbf{W}^T \mathbf{C}\mathbf{W})$ predstavlja dijagonalnu matricu koja se dobija kada se svi elementi van glavne dijagonale matrice $\mathbf{W}^T \mathbf{C}\mathbf{W}$, anuliraju. Ako napišemo jednačinu za pojedinačne kolone \mathbf{w}_k , imamo

$$\frac{d\mathbf{w}_k}{dt} = \alpha (\mathbf{C}\mathbf{w}_k - \lambda_k \mathbf{w}_k), \quad (7.5)$$

gde je λ_k k -ti dijagonalni element matrice $\text{diag}(\mathbf{W}^T \mathbf{C}\mathbf{W})$. Možemo lako zaključiti da su stacionarne tačke ove jednačine sopstveni vektori matrice \mathbf{C} . Pošto \mathbf{w}_k pokrivaju

dominantni/sporedni potprostor matrice \mathbf{C} , onda su jedino moguća rešenja dominantne/sporedne komponente. Ako $\mathbf{w}_k(i)$ pridružene diferencne jednačine posećuje beskonačno često skoro sigurno, kompaktan podskup domena privlačenja rešenja jednačine (7.5), tada rešenje jednačine (7.5) predstavlja rešenje diskretne jednačine (7.3). Naravno, veoma je teško (ako ne i nemoguće) dokazati ovo u opštem slučaju.

Usled simetrije predloženog algoritma, sistem ima više od jednog rešenja. To može prouzrokovati “konflikt interesa” između pojedinih neurona. Ako razmotrimo slučaj na slici 7.1 može se uočiti da dok se jedan sinaptički vektor kreće u pravcu pozitivnog dela dominantne ose 1, u isto vreme drugi sinaptički vektor može da ima “nameru” da se kreće ka negativnom delu dominantne ose 1. To znači da algoritam može da postane osetljiv na izbor koeficijenta α . Osim toga, značajno je proučiti i slučaj u kome realizacija algoritma u paralelnom hardveru dovodi do toga da koeficijent α nije perfektno jednak za sve vektore, pa se u tom slučaju postavlja pitanje stabilnosti algoritma.

Sada ćemo analizirati slučaj kada je u algoritam uneta nekakva vrsta asimetrije. Drugim rečima, daćemo pojedinim vektorima drugačije šanse za dostizanje dominantnog/najsporednijeg sopstvenog vektora. U tom slučaju, predloženi metod se može opisati sledećom jednačinom:

$$LA_{PCA/MCA} = FLA_{PSA/MSA} + ILA \left(\max \left(E \left\{ \left((\mathbf{D}\mathbf{y})^T \mathbf{y} \right) \left| \begin{array}{l} \mathbf{w}_k^T \mathbf{w}_k = 1, \\ k = 1, 2, \dots, N \end{array} \right. \right\} \right) \right), \quad (7.6)$$

gde je \mathbf{D} dijagonalna matrica sa nenultim dijagonalnim elementima d_n takvim da je $|d_n| < 1$. Unošenje asimetrije dovodi do dodatne vremenske hijerarhije u realizaciji individualnih delova zakona učenja. Ako su svi d_n jednaki α , ova jednačina se svodi na homogeni slučaj.

Ponovo ćemo sprovesti uprošćenu analizu za slučaj kada je $|d_n|$ mnogo manje od 1. U tom slučaju, individualni deo u jednačini (7.6) praktično nema uticaj na porodični deo. Tada porodični deo obezbeđuje pronalaženje glavnog/sporednog potprostora. To znači da je $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i da \mathbf{W} pokriva glavni/sporedni potprostor. Kao što je već ranije objašnjeno možemo pristupiti problemu PCA/MCA kao optimizacionom problemu. U stvari brži deo zakona učenja, koji obezbeđuje izračunavanja PSA/MSA predstavlja ograničenje za sporiji (individualni) deo zakona učenja. To znači da ako pretpostavimo da je PSA/MSA deo zakona učenja dovoljno brz možemo reći da u tom slučaju taj deo zakona učenja predstavlja

ortonormalno ograničenje na PSA/MSA potprostoru. U tom slučaju, individualni deo u jednačini (7.6) možemo pisati kao

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \left(\mathbf{x}(i)\mathbf{y}(i)^T - \mathbf{W}(i) \text{diag}(y_1(i)^2, \dots, y_N(i)^2) \right) \mathbf{D}, \quad (7.7)$$

gde je \mathbf{D} dijagonalna matrica sa dijagonalnim elementima d_k

takvim da je $|d_k| < 1$ i $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i pokriva glavni (sporedni) podprostor.

Odgovarajuća diferencijalna jednačina je [Ljung, 1977], [Oja, 1985]

$$\frac{d\mathbf{W}}{dt} = \left(\mathbf{C}\mathbf{W} - \mathbf{W} \text{diag}(\mathbf{W}^T \mathbf{C}\mathbf{W}) \right) \mathbf{D}, \quad (7.8)$$

gde je $\text{diag}(\mathbf{W}^T \mathbf{C}\mathbf{W})$ definisana kao i ranije. Ako napišemo jednačinu za svaku kolonu \mathbf{w}_k , imamo

$$\frac{d\mathbf{w}_k}{dt} = d_k (\mathbf{C}\mathbf{w}_k - \lambda_k \mathbf{w}_k), \quad (7.9)$$

gde je λ_k k -ti dijagonalni element matrice $\text{diag}(\mathbf{W}^T \mathbf{C}\mathbf{W})$. Nije teško zaključiti da su stacionarne tačke ove jednačine sopstveni vektori matrice \mathbf{C} . Ako $\mathbf{w}_k(i)$ pridruženog diskretnog algoritma posećuje beskonačno često skoro sigurno kompaktni podskup domena privlačenja jednačine (7.9), tada rešenje jednačine (7.9) predstavlja i rešenje algoritma (7.6). Ponovo ćemo naglasiti, da je veoma teško pokazati tu osobinu diskretnog algoritma u opštem slučaju.

7.1 Matematička analiza predloženog principa

U ovom paragrafu će biti pokazano kako predloženi TOHM metod može biti predstavljen i kao učenje na približno Stifel/Grasmanovoj (Stiefel/Grassman) podmnogostrukosti. Biće predstavljen algoritam koji izvodi učenje na Grasmanovoj dominantnoj/sporednoj podmnogostrukosti i kasnije će biti pokazano kako se on može povezati sa TOHM metodom. Potpuno analogno može se izvesti analiza za slučaj učenja na Stifelovoj dominantnoj/sporednoj podmnogostrukosti.

Sada ćemo neformalno ponoviti definiciju Grasmanove mnogostrukosti koja je data u [Edelman, 1998], [Fiori, 2002]:

Prostor matrica $\mathbf{W} \in O^{K \times N} \subset R^{K \times N}$ ($N \leq K$) takvih da je $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i homogena funkcija $J: O^{K \times N} \rightarrow R$ takva da je $J(\mathbf{W}) = J(\mathbf{W}\mathbf{Q})$ za bilo koju $N \times N$ ortogonalnu matricu \mathbf{Q} se nazivaju Grasmanovom mnogostrukošću.

Algoritmi učenja koji se primenjuju u neuralnim mrežama se često mogu shvatiti kao algoritmi koji maksimiziraju/minimiziraju funkciju cene pod nekim ograničenjem, pri čemu je to ograničenje najčešće ortonormalnost sinaptičke matrice.

Takvi zakoni učenja mogu predstavljati rešenje sledećeg problema:

$$\text{Nađi } \mathbf{W}_{eks} \text{ takvu da važi: } J(\mathbf{W}_{eks}) = \max/\min J(\mathbf{W}), \mathbf{W} \in R^{K \times N}.$$

Standardan način za dobijanje željenog rešenja je definisanje Lagranžove funkcije

$$J_l(\mathbf{W}) = J(\mathbf{W}) + l \operatorname{tr}(\mathbf{W}^T \mathbf{W} - \mathbf{I}),$$

gde je l takozvani Lagranžov multiplikator, i traženje ekstremuma funkcije $J_l(\mathbf{W})$, na primer upotrebom tehnike uzlaznih/silaznih gradijenata,

$$\frac{d\mathbf{W}}{dt} = \eta \nabla J_l.$$

Kako bi se osigurala ortonormalnost sinaptičkih kolona, može se primeniti iterativna ortogonalizacija kolona matrice \mathbf{W} , na primer upotrebom Gram-Šmit (Gram-Schmidt) metoda

ortogonalizacije matrice čiji se elementi menjaju na osnovu gradijentne optimizacije funkcije $J(\mathbf{W})$ ili pomoću projekcije na ortonormalnu grupu [Fiori, 2001]. Međutim, iterativna primena ortonormalnosti može biti problematična u praksi (see e.g. [Fiori, 2001]). To je razlog zašto su istraživači počeli da proučavaju zakone učenja koji održavaju ortogonalnost sinaptičke matrice u svakom momentu. Takvi algoritmi su poznati kao SOC (Orthonormal Strongly-Constrained – striktno ortnormalno ograničeni) algoritmi. Nekoliko algoritama iz te klase je analizirano u [Fiori, 2001].

Prvo analizirajmo sledeći algoritam:

$$E(y^2) = \max \quad (7.10)$$

$$y = \frac{\mathbf{w}^T}{\sqrt{\mathbf{w}^T \mathbf{w}}} \mathbf{x}, \quad (7.11)$$

gde \mathbf{x} predstavlja ulazni vektor za jednoslojnu mrežu sa jednim izlaznim neuronom (y) i \mathbf{w} predstavlja sinaptički vektor. Jednostavno je uočiti da jednačinu (7.10) možemo napisati i kao

$$y^2 = \frac{\mathbf{w}^T \mathbf{x} \mathbf{x}^T \mathbf{w}}{\mathbf{w}^T \mathbf{w}} = \max. \quad (7.12)$$

Ako sada direktno generišemo uzlazni gradijentni algoritam u odnosu na y^2 , pri čemu se gradijent računa po \mathbf{w} , imamo:

$$\mathbf{w}(i+1) = \mathbf{w}(i) + \gamma(i) \frac{xy \sqrt{\mathbf{w}(i)^T \mathbf{w}(i)} - y^2 \mathbf{w}(i)}{\mathbf{w}(i)^T \mathbf{w}(i)}. \quad (7.13)$$

Jednostavno se uočava da zamenom $\mathbf{w}(i)^T \mathbf{w}(i) = 1$ dobijamo, direktno, poznati Ojin zakon učenja [Oja, 1982]. Ako generalizujemo ovaj algoritam na slučaj sa više izlaznih neurona, imamo

$$E(\text{tr}(\mathbf{y}\mathbf{y}^T)) = \max$$

$$y_k = \frac{\mathbf{w}_k^T}{\sqrt{\mathbf{w}_k^T \mathbf{w}_k}} \mathbf{x},$$

gde y_k predstavlja k -ti izlazni neuron i \mathbf{w}_k predstavlja k -tu kolonu sinaptičke matrice \mathbf{W} . Zakon učenja za modifikaciju sinaptičkih vektora je u tom slučaju

$$\mathbf{w}_k(i+1) = \mathbf{w}_k(i) + \gamma(i) \frac{\mathbf{x}y_k \sqrt{\mathbf{w}_k(i)^T \mathbf{w}_k(i)} - y_k^2 \mathbf{w}_k(i)}{\mathbf{w}_k(i)^T \mathbf{w}_k(i)}. \quad (7.14)$$

Sada ćemo definisati Grasmanovu dominantnu podmnogostrukost.

Prostor matrica $\mathbf{W} \in O^{K \times N} \subset R^{K \times N}$ ($N \leq K$) takvih da je $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i \mathbf{W} pokriva dominantni potprostor matrice \mathbf{C} , i funkcija $J: O^{K \times N} \rightarrow R$ takva da je $J(\mathbf{W}) = J(\mathbf{W}\mathbf{Q})$ za bilo koju $N \times N$ ortonormalnu matricu \mathbf{Q} ćemo nazvati Grasmanovom dominantnom podmnogostrukošću.

Ako sada primenimo algoritam (7.14) pod striktnim ograničenjem za $\mathbf{W} - \mathbf{W}$ je takvo da je $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i \mathbf{W} pokriva dominantni potprostor matrice \mathbf{C} , u stvari imamo sledeći algoritam

$$\begin{aligned} \mathbf{w}_k(i+1) &= \mathbf{w}_k(i) + \gamma(i) (\mathbf{x}y_k - y_k^2 \mathbf{w}_k(i)), \\ y_k &= \mathbf{w}_k^T \mathbf{x}. \end{aligned}$$

Funkcija cene u kompaktnoj notaciji je data sa

$$E(\text{tr}(\mathbf{y}\mathbf{y}^T)) = E(\text{tr}(\mathbf{W}^T \mathbf{x}\mathbf{x}^T \mathbf{W})) = \text{tr}(\mathbf{W}^T \mathbf{C}\mathbf{W}) = \max$$

Nije teško uočiti da se radi o funkciji za koju je ispunjeno $J(\mathbf{W}\mathbf{Q}) = J(\mathbf{W})$. Drugim rečima možemo reći algoritam (7.14) vrši učenje na Grasmanovoj dominantnoj podmnogostrukosti. Algoritmu (7.14) se može pridružiti sledeća diferencijalna jednačina (u kompaktnoj notaciji)

$$\frac{d\mathbf{W}}{dt} = \left(\mathbf{C}\mathbf{W} - \mathbf{W} \text{diag}(\mathbf{W}^T \mathbf{C}\mathbf{W}) \right), \quad (7.15)$$

gde je $\text{diag}(\mathbf{W}^T \mathbf{C}\mathbf{W})$ dijagonalna matrica definisana kao i ranije. Ako napišemo (7.15) za svaku kolonu \mathbf{w}_k , imamo

$$\frac{d\mathbf{w}_k}{dt} = (\mathbf{C}\mathbf{w}_k - \lambda_k \mathbf{w}_k), \quad (7.16)$$

gde je λ_k k -ti element glavne dijagonale matrice $\text{diag}(\mathbf{W}^T \mathbf{C}\mathbf{W})$. Lako je zaključiti da su stacionarne tačke ovih jednačina sopstveni vektori matrice \mathbf{C} . Pošto se učenje obavlja na

dominantnoj Grasmanovoj podmnogostrukosti rezultujući vektor \mathbf{w}_k će biti jednak nekom od dominantnih sopstvenih vektora matrice \mathbf{C} . Ako $\mathbf{w}_k(i)$ pridruženog diskretnog algoritma posećuje beskonačno često kompaktan podskup domena privlačenja rešenja jednačine (7.16), tada su rešenja jednačine (7.16) takođe rešenja pridruženog diskretnog algoritma (7.11). Naravno, kao što je već rečeno, ovu osobinu algoritma (7.11) je teško dokazati.

Na sličan način se može definisati Grasmanova sporedna podmnogostrukost.

Prostor matrica $\mathbf{W} \in O^{K \times N} \subset R^{K \times N}$ ($N \leq K$) takvih da je $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i \mathbf{W} pokriva sporedni potprostor matrice \mathbf{C} , i funkcija $J: O^{K \times N} \rightarrow R$ takva da je $J(\mathbf{W}) = J(\mathbf{W}\mathbf{Q})$ za bilo koju $N \times N$ ortonormalnu matricu \mathbf{Q} će biti nazvani Grasmanovom sporednom podmnogostrukošću.

Ako se algorithm (7.14) primeni pod striktnim ograničenjem za \mathbf{W} , \mathbf{W} je takvo da je $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i \mathbf{W} pokriva sporedni potprostor matrice \mathbf{C} , i primenimo analizu kao u slučaju dominantne Grasmanove podmnogostrukosti kolone sinaptičke matrice \mathbf{W} će biti jednake sporednim sopstvenim vektorima matrice \mathbf{C} .

Potpuno analogno, mogu se definisati Stifelova dominantna i sporedna podmnogostrukost, za slučaj kada algoritam nije potpuno simetričan (vidi [Jankovic, 2005a, 2005b]).

7.2 Jedan primer primene TOHM metoda za PCA

U ovom pragrafu će biti pokazano kako se TOHM metod može primeniti za transformaciju SLA metoda [Oja, 1989] u PCA metod. U poglavlju 4 je već objašnjeno da je SLA metod PSA metod.

Primenom TOHM metoda na SLA (jednačina (4.19)) dobijamo sledeći PCA algoritam:

$$\begin{aligned} \mathbf{w}_k(i+1) = & \mathbf{w}_k(i) + \gamma(i) \left[y_k(i) \mathbf{x}(i) - y_k(i) \sum_{m=1}^N y_m(i) \mathbf{w}_m(i) \right] \\ & + \gamma(i) d_k \left[y_k(i) \mathbf{x}(i) - \mathbf{w}_k(i) y_k^2(i) \right], \end{aligned} \quad (7.17)$$

$m, k = 1, \dots, N,$

gde su d_k parametri različiti od nule takvi da važi $|d_k| < 1$. U kompaktnoj notaciji imamo

$$\begin{aligned} \mathbf{W}(i+1) = & \mathbf{W}(i) + \gamma(i) \left(\mathbf{x}(i) \mathbf{y}(i)^T - \mathbf{W}(i) \mathbf{y}(i) \mathbf{y}(i)^T \right) \\ & + \gamma(i) \left(\mathbf{x}(i) \mathbf{y}(i)^T - \mathbf{W}(i) \text{diag} \left(\mathbf{y}(i) \mathbf{y}(i)^T \right) \right) \mathbf{D}, \end{aligned} \quad (7.18)$$

gde je $\mathbf{W}(i) = (\mathbf{w}_1(i) \dots \mathbf{w}_N(i))$, $\mathbf{y}(i) = (y_1(i) \dots y_N(i))^T$, \mathbf{D} dijagonalan matrica sa dijagonalnim elementima d_k ($k=1, 2, \dots, N$), i $\text{diag}(\mathbf{F})$ označava dijagonalnu matricu dobijenu od matrice \mathbf{F} u kojoj su svi elementi van dijagonale matrice \mathbf{F} postavljeni na 0. Primenom stohastičke aproksimacije ovoj diferencnoj jednačini se može pridružiti odgovarajuća diferencijalna jednačina :

$$\frac{d\mathbf{W}}{dt} = \mathbf{C}\mathbf{W} - \mathbf{W}\mathbf{W}^T\mathbf{C}\mathbf{W} + \left(\mathbf{C}\mathbf{W} - \mathbf{W} \text{diag} \left(\mathbf{W}^T \mathbf{C}\mathbf{W} \right) \right) \mathbf{D}. \quad (7.19)$$

Kao što je već ranije rečeno, pod uslovima koji su dati u [Ljung, 1977] [Oja, 1985] rešenja stohastičke diferencne jednačine (7.18) teže asimptotski ka stabilnim rešenjima jednačine (7.19). Ključna pretpostavka za konvergenciju je da rešenje $\mathbf{W}(i)$ originalnog diskretnog algoritma (7.18) posećuje beskonačno često sa verovatnoćom 1 kompaktan podskup domena privlačenja asimptotski stabilnog rešenja jednačine (7.19). Nažalost, dokazivanje ove osobine je veoma teško. Ovde možemo reći da je u svim simulacijama koje su izvedene stabilno rešenje jednačine (7.19) predstavljalo i rešenje jednačine (7.18), pod uslovom da je matrica \mathbf{D} bila izabrana na odgovarajući način.

Sada ćemo pronaći asimptotski stabilno rešenje jednačine (7.19), ako isto postoji. Prvo uvedimo sledeću oznaku

$$\mathbf{U} = (\mathbf{c}_1, \dots, \mathbf{c}_N). \quad (7.20)$$

Tada je

$$\mathbf{C}\mathbf{U} = \mathbf{U}\mathbf{\Lambda}, \quad \mathbf{U}^T \mathbf{C}\mathbf{U} = \mathbf{\Lambda}, \quad (7.21)$$

gde je $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_N)$ i $\lambda_1 > \lambda_2 > \dots > \lambda_N > \lambda_{N+1} \geq \lambda_{N+2} \geq \dots \geq \lambda_K > 0$.

Nije teško uočiti da $\mathbf{W}=\mathbf{U}$ ili \mathbf{W} koje je jednako nekoj permutaciji od \mathbf{U} predstavlja rešenje jednačine (7.19). Zaista, ako zamenimo $\mathbf{W}=\mathbf{U}$ u jednačinu (7.19), tada je

$$\begin{aligned} \left. \frac{d\mathbf{W}}{dt} \right|_{\mathbf{W}=\mathbf{U}} &= \mathbf{C}\mathbf{U} - \mathbf{U}\mathbf{U}^T \mathbf{C}\mathbf{U} + (\mathbf{C}\mathbf{U} - \mathbf{U} \text{diag}(\mathbf{U}^T \mathbf{C}\mathbf{U}))\mathbf{D} \\ &= \mathbf{U}\mathbf{\Lambda} - \mathbf{U}\mathbf{\Lambda} + (\mathbf{U}\mathbf{\Lambda} - \mathbf{U}\mathbf{\Lambda})\mathbf{D} = \mathbf{0}. \end{aligned} \quad (7.22)$$

Takođe, ako zamenimo $\mathbf{W}=\mathbf{U}\mathbf{P}$ u jednačinu (7.19), gde \mathbf{P} označava permutacionu matricu, imaćemo isti rezultat.

Sada će biti pokazano da ne postoje druga rešenja. Prvo će biti pokazano da rešenje jednačine (7.19) pokriva N -dimenzioni glavni potprostor matrice \mathbf{C} .

Jednačina (7.18) se može napisati i na sledeći način:

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \left(\mathbf{x}(i) \mathbf{y}(i)^T \right) (\mathbf{I} + \mathbf{D}) - \gamma(i) \mathbf{W}(i) \left(\mathbf{y}(i) \mathbf{y}(i)^T + \text{diag}(y_1^2(i), \dots, y_N^2(i)) \mathbf{D} \right), \quad (7.23)$$

ili

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \cdot \left(\mathbf{x}(i) \mathbf{x}(i)^T \mathbf{W}(i) \right) (\mathbf{I} + \mathbf{D}) - \gamma(i) \cdot \mathbf{W}(i) \Theta_i, \quad (7.24)$$

gde je \mathbf{I} jedinična matrica i Θ_i matrica dimenzije $N \times N$ definisana sa

$$\Theta_i = \mathbf{y}(i) \mathbf{y}(i)^T + \text{diag}(y_1^2(i), \dots, y_N^2(i)) \mathbf{D}.$$

U homogenom slučaju (kada su svi dijagonalni članovi matrice \mathbf{D} jednaki) možemo pisati $\mathbf{D} = \alpha \mathbf{I}$, pri čemu je $\alpha > -1$. Tada imamo

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) (1 + \alpha) \left(\mathbf{x}(i) \mathbf{x}(i)^T \mathbf{W}(i) \right) - \gamma(i) (1 + \alpha) \mathbf{W}(i) \Theta_i / (1 + \alpha). \quad (7.25)$$

Nije teško uočiti da ova jednačina (sa izuzetkom člana $1 + \alpha$) predstavlja jednačinu koja je analizirana od strane Plumblija [Plumbley, 1991]. U [Plumbley, 1991] je analizirana sledeća klasa zakona učenja:

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \left(\mathbf{x}(i) \mathbf{x}^T(i) \mathbf{W}(i) + \mathbf{W}(i) \Theta(i) \right) \quad (7.26)$$

gde je $\Theta(i)$ nekakva $N \times N$ matrica koja može biti funkcija od $\mathbf{W}(i)$ i/ili $\mathbf{x}(i)$. Pod uslovima koji su definisani u [Ljung, 1977] i [Oja, 1985], rešenje stohastičke diferencne jednačine (7.26) teži asimptotski stabilnom rešenju sledeće jednačine:

$$\frac{d\mathbf{W}}{dt} = \mathbf{C}\mathbf{W} + \mathbf{W}\Theta. \quad (7.27)$$

U [Plumbley, 1991] je dokazana sledeća teorema:

Teorema 7.1. Ako težinska matrica \mathbf{W} ima maksimalni rang za svako t , ordinarna diferencijalna jednačina (7.27) je asimptotski stabilna na skupu gde kolone matrice \mathbf{W} pokrivaju potprostor definisan sa N dominantnih sopstvenih vektora matrice \mathbf{C} . Oblast privlačenja ovog skupa je skup \mathbf{W} takav da je za sve dominantne sopstvene vektore \mathbf{c}_i ($1 \leq i \leq N$) matrice \mathbf{C} , ispunjeno $\mathbf{W}^T \mathbf{c}_i \neq 0$.

Ukratko, u [Plumbley, 1991] je pokazano da klasa algoritama (7.26) konvergira ka matrici \mathbf{W} koja pokriva N -dimenzionalni dominantni potprostor matrice \mathbf{C} , pri čemu se smatra da je matrica $\mathbf{W}(i)$ maksimalnog ranga. Konvergencija ne zavisi od $\Theta(i)$, iako je jasno da $\Theta(i)$ mora da bude izabrano tako da matrica $\mathbf{W}(i)$ ostaje nesingularna u toku primene algoritma [Plumbley, 1991].

Kako je $1+\alpha > 0$ (jer je usvojeno $\alpha > -1$), član $1+\alpha$ ne utiče na analizu koja je izvedena u [Plumbley, 1991]. To znači da rešenje jednačine (7.25) prekriva N -dimenzioni glavni potprostor matrice \mathbf{C} .

Korišćenjem sličnog postupka moguće je pokazati, da i kada su dijagonalni članovi matrice \mathbf{D} nejednaki, rešenje jednačine (7.18) pokriva glavni potprostor matrice \mathbf{C} . To se može pokazati analizom sledećeg algoritma:

$$\mathbf{W}(i+1) = \mathbf{W}(i) + \gamma(i) \left(\mathbf{x}(i) \mathbf{x}^T(i) \mathbf{W}(i) \mathbf{D}_1 + \mathbf{W}(i) \Theta(i) \right) \quad (7.28)$$

gde je $\Theta(i)$ neka $N \times N$ matrica koja može biti funkcija od $\mathbf{W}(i)$ i/ili $\mathbf{x}(i)$, i \mathbf{D}_1 je dijagonalna matrica čiji su svi dijagonalni elementi pozitivni. Opet, pod uslovima koji su ranije definisani, rešenje stohastičke diferencne jednačine (7.28) teži asimptotski stabilnom rešenju jednačine:

$$\frac{d\mathbf{W}}{dt} = \mathbf{C}\mathbf{W}\mathbf{D}_1 + \mathbf{W}\Theta. \quad (7.29)$$

Sada ćemo predstaviti sledeću teoremu koja predstavlja generalizaciju Teoreme 7.1.

Teorema 7.2. Ako težinska matrica \mathbf{W} ima maksimalni rang za svako t , ordinarna diferencijalna jednačina (7.29) je asimptotski stabilna na skupu gde kolone \mathbf{W} pokrivaju N -dimenzioni dominantni potprostor \mathbf{C} . Domen privlačenja ovog skupa je skup \mathbf{W} takav da za sve dominantne sopstvene vektore \mathbf{c}_i ($1 \leq i \leq N$) matrice \mathbf{C} važi $\mathbf{W}^T \mathbf{c}_i \neq 0$.

Dokaz se može izvesti analogno dokazu Teoreme 7.1, a koji je prikazan u [Plumbley, 1991].

Kako moguće rešenje pokriva dominantni potprostor matrice \mathbf{C} , jedino rešenje koje nije matrica čije su kolone sopstveni vektori matrice \mathbf{C} , je matrica koja predstavlja rotaciju matrice čije su kolone dominantni sopstveni vektori matrice \mathbf{C} . Označimo sa \mathbf{R} rotacionu matricu. Označimo sa \mathbf{W}_s matricu koja se od matrice \mathbf{U} dobija rotacijom

$$\mathbf{W}_s = \mathbf{U}\mathbf{R}, \quad \mathbf{R}^{-1} = \mathbf{R}^T. \quad (7.30)$$

Ako zamenimo \mathbf{W}_s u jednačinu (7.19) imamo

$$\begin{aligned} \frac{d\mathbf{W}}{dt} \Big|_{\mathbf{W} = \mathbf{W}_s} &= \mathbf{C}\mathbf{W}_s - \mathbf{W}_s \mathbf{W}_s^T \mathbf{C}\mathbf{W}_s + \left(\mathbf{C}\mathbf{W}_s - \mathbf{W}_s \text{diag}(\mathbf{W}_s^T \mathbf{C}\mathbf{W}_s) \right) \mathbf{D} \\ &= \mathbf{C}\mathbf{U}\mathbf{R} - \mathbf{U}\mathbf{R}\mathbf{R}^T \mathbf{U}^T \mathbf{C}\mathbf{U}\mathbf{R} + \left(\mathbf{C}\mathbf{U}\mathbf{R} - \mathbf{U}\mathbf{R} \text{diag}(\mathbf{R}^T \mathbf{U}^T \mathbf{C}\mathbf{U}\mathbf{R}) \right) \mathbf{D} \\ &= \mathbf{U}\mathbf{\Lambda}\mathbf{\Lambda} - \mathbf{U}\mathbf{\Lambda}\mathbf{\Lambda} + \left(\mathbf{U}\mathbf{\Lambda}\mathbf{\Lambda} - \mathbf{U}\mathbf{R} \text{diag}(\mathbf{R}^T \mathbf{\Lambda}\mathbf{R}) \right) \mathbf{D} \\ &= \mathbf{U} \left(\mathbf{\Lambda}\mathbf{R} - \mathbf{R} \text{diag}(\mathbf{R}^T \mathbf{\Lambda}\mathbf{R}) \right) \mathbf{D}. \end{aligned} \quad (7.31)$$

Nije teško uočiti da se stabilne tačke dobijaju kada je $\Lambda R - R \text{diag}(R^T \Lambda R) = 0$. Ovo je moguće samo kada je R jedinična ili permutaciona matrica, pošto je Λ dijagonalna matrica, i kolone R moraju biti jednake nekim od sopstvenih vektora matrice Λ . To dalje znači da su jedina moguća rešenja permutacije matrice U .

Stabilne tačke U u jednačini (7.19) zadržavamo

$$W = U + A$$

Tada imamo

$$\begin{aligned} \frac{dA}{dt} &= \frac{dW}{dt} \Big|_{W=U+A} \\ &= C(U+A) - (U+A)(U+A)^T C(U+A) \\ &\quad + (C(U+A) - (U+A) \text{diag}(U+A)^T C(U+A))U \\ &= CA - UA^T CU - AU^T CU - UU^T CA \\ &\quad + (CA - U \text{diag}(U^T CA + A)) - A \text{diag}(U^T CU)U + \dots \end{aligned} \tag{7.20}$$

Ako uočimo da je $\|A\|$ malo, možemo kao (7.3) jediničnu (7.32) određiti asimptotičku stabilnost. Za n -ti kolonu δ_n matrice A , linearni dio jednačine (7.32) daje

$$\begin{aligned} \frac{d\delta_n}{dt} &= C\delta_n - \sum_{m=1}^n \lambda_{m,n} \delta_m \delta_n - \sum_{m=1}^n \lambda_{m,n} \delta_m \delta_n \\ &\quad - \lambda_{n,n} \delta_n \delta_n - \dots \end{aligned} \tag{7.21}$$

7.3 Analiza asimptotske stabilnosti rešenja

Sada pošto smo pronašli rešenja jednačine (7.19) pokazaćemo da ta rešenja predstavljaju asimptotski stabilna rešenja. Da bismo to uradili, razmotrimo malo odstupanje $\Delta = (\delta_1 \dots \delta_N)$ od stabilne tačke \mathbf{U} . U jednačini (7.19) zamenaimo

$$\mathbf{W} = \mathbf{U} + \Delta.$$

Tada imamo

$$\begin{aligned} \frac{d\Delta}{dt} &= \frac{d\mathbf{W}}{dt} \Big|_{\mathbf{W} = \mathbf{U} + \Delta} \\ &= \mathbf{C}(\mathbf{U} + \Delta) - (\mathbf{U} + \Delta)(\mathbf{U} + \Delta)^T \mathbf{C}(\mathbf{U} + \Delta) \\ &\quad + \left(\mathbf{C}(\mathbf{U} + \Delta) - (\mathbf{U} + \Delta) \text{diag} \left((\mathbf{U} + \Delta)^T \mathbf{C}(\mathbf{U} + \Delta) \right) \right) \mathbf{D} \\ &= \mathbf{C}\Delta - \mathbf{U}\Delta^T \mathbf{C}\mathbf{U} - \Delta \mathbf{U}^T \mathbf{C}\mathbf{U} - \mathbf{U}\mathbf{U}^T \mathbf{C}\Delta \\ &\quad + \left(\mathbf{C}\Delta - \mathbf{U} \text{diag} \left(\mathbf{U}^T \mathbf{C}(\Delta + \mathbf{U}) \right) - \Delta \text{diag} \left(\mathbf{U}^T \mathbf{C}\mathbf{U} \right) \right) \mathbf{D} + O(\|\Delta\|^2). \end{aligned} \quad (7.32)$$

Ako usvojimo da je $\|\Delta\|$ malo, linearni deo (po Δ) jednačine (7.32) određuje asimptotsku satbilnost. Za n -tu kolonu δ_n matrice Δ , linearni deo jedančine (7.32) daje

$$\begin{aligned} \frac{d\delta_n}{dt} &= \mathbf{C}\delta_n - \sum_{m=1}^N \lambda_m \mathbf{c}_m (\mathbf{c}_m^T \delta_n) - \sum_{m=1}^N \lambda_n \mathbf{c}_m (\delta_m^T \mathbf{c}_n) \\ &\quad - \lambda_n \delta_n + d_n \left(\mathbf{C}\delta_n - \lambda_n \delta_n - 2\mathbf{c}_n \lambda_n (\mathbf{c}_n^T \delta_n) \right). \end{aligned} \quad (7.33)$$

Sada će biti analizirano šta se dešava sa projekcijom δ_n na različite sopstvene vektore \mathbf{c}_r , $r=1, \dots, K$. Prvo, usvojimo $r > N$. Tada je, usled ortonormalnosti sopstvenih vektora $\{\mathbf{c}_r\}$,

$$\begin{aligned} \frac{d(\mathbf{c}_r^T \delta_n)}{dt} &= \lambda_r (\mathbf{c}_r^T \delta_n) - \lambda_n (\mathbf{c}_r^T \delta_n) + d_n (\lambda_r (\mathbf{c}_r^T \delta_n) - \lambda_n (\mathbf{c}_r^T \delta_n)) \\ &= (1 + d_n)(\lambda_r - \lambda_n) \mathbf{c}_r^T \delta_n. \end{aligned} \quad (7.34)$$

Tako, ako je $d_n > -1$, tada $\mathbf{c}_r^T \delta_n$ teži asimptotski nuli kako $t \rightarrow \infty$, pošto je po pretpostavci $\lambda_r < \lambda_n$.

Ako usvojimo $r \leq N$, tada je

$$\begin{aligned} \frac{d(\mathbf{c}_r^T \delta_n)}{dt} &= \lambda_r (\mathbf{c}_r^T \delta_n) - \lambda_r (\mathbf{c}_r^T \delta_n) - \lambda_n (\delta_r^T \mathbf{c}_n) - \lambda_n (\mathbf{c}_r^T \delta_n) \\ &\quad + d_n (\lambda_r (\mathbf{c}_r^T \delta_n) - \lambda_n (\mathbf{c}_r^T \delta_n) - \delta(r, n) \lambda_n (\mathbf{c}_n^T \delta_n)), \end{aligned} \quad (7.35)$$

gde je $\delta(r, n)$ Kronekerov delta simbol. Lako je uočiti da su funkcije $\mathbf{c}_r^T \delta_n$ dekuplovane u smislu da $\mathbf{c}_r^T \delta_n$ i $\mathbf{c}_n^T \delta_r$ zavise jedna od druge ali ne i od ostalih funkcija. Slučaj $r=n$ je najjednostavniji i rezultuje sledećom jednačinom

$$\begin{aligned} \frac{d(\mathbf{c}_n^T \delta_n)}{dt} &= \lambda_n (\mathbf{c}_n^T \delta_n) - \lambda_n (\mathbf{c}_n^T \delta_n) - \lambda_n (\mathbf{c}_n^T \delta_n) - \lambda_n (\mathbf{c}_n^T \delta_n) \\ &\quad + d_n (\lambda_n (\mathbf{c}_n^T \delta_n) - \lambda_n (\mathbf{c}_n^T \delta_n) - 2\lambda_n (\mathbf{c}_n^T \delta_n)) \\ &= -2(1 + d_n) \lambda_n (\mathbf{c}_n^T \delta_n). \end{aligned} \quad (7.36)$$

To implicira da $\mathbf{c}_n^T \delta_n$ teži asimptotski nuli kako $t \rightarrow \infty$ ako je $d_n > -1$.

Konačno, usvojimo $r \neq n$, i $r \leq N$. Sada ćemo morati da analiziramo sistem dve jednačine koji opisuje ponašanje $\mathbf{c}_r^T \delta_n$ i $\mathbf{c}_n^T \delta_r$

$$\begin{aligned}\frac{d(\mathbf{c}_r^T \boldsymbol{\delta}_n)}{dt} &= (1+d_n)(\lambda_r - \lambda_n) \mathbf{c}_r^T \boldsymbol{\delta}_n - \lambda_r (\mathbf{c}_r^T \boldsymbol{\delta}_n) - \lambda_n (\mathbf{c}_n^T \boldsymbol{\delta}_r), \\ \frac{d(\mathbf{c}_n^T \boldsymbol{\delta}_r)}{dt} &= (1+d_r)(\lambda_n - \lambda_r) \mathbf{c}_n^T \boldsymbol{\delta}_r - \lambda_n (\mathbf{c}_n^T \boldsymbol{\delta}_r) - \lambda_r (\mathbf{c}_r^T \boldsymbol{\delta}_n).\end{aligned}\quad (7.37)$$

Uvedimo sledeće oznake

$$\begin{aligned}\mathbf{X} &= \begin{bmatrix} \mathbf{c}_r^T \boldsymbol{\delta}_n \\ \mathbf{c}_n^T \boldsymbol{\delta}_r \end{bmatrix} \\ \mathbf{A} &= \begin{bmatrix} (1+d_n)(\lambda_r - \lambda_n) - \lambda_r & -\lambda_n \\ -\lambda_r & (1+d_r)(\lambda_n - \lambda_r) - \lambda_n \end{bmatrix}.\end{aligned}\quad (7.38)$$

Tada se jednačina (7.37) može pisati kao

$$\frac{d\mathbf{X}}{dt} = \mathbf{A}\mathbf{X}.\quad (7.39)$$

Poznato je da rešenje jednačine (7.39) teži asimptotski ka nuli ako i samo ako oba korena karakterističnog polinoma imaju striktno negativne realne delove. Karakteristični polinom se može dobiti iz sledeće jednačine

$$\det(s\mathbf{I} - \mathbf{A}) = 0.\quad (7.40)$$

U našem slučaju,

$$(s + \lambda_r - (1+d_n)(\lambda_r - \lambda_n))(s + \lambda_n - (1+d_r)(\lambda_n - \lambda_r)) - \lambda_r \lambda_n = 0.\quad (7.41)$$

Posle nekoliko algebarskih manipulacija dobijamo

$$s^2 + s(\lambda_r + \lambda_n - (d_n - d_r)(\lambda_r - \lambda_n)) - (\lambda_n(1 + d_n) - \lambda_r(1 + d_r))(\lambda_r - \lambda_n) - (1 + d_n)(1 + d_r)(\lambda_r - \lambda_n)^2 = 0. \quad (7.42)$$

Prvo ćemo razmotriti slučaj kada su svi d_i jednaki α . U tom slučaju je jednačina od interesa

$$s^2 + s(\lambda_r + \lambda_n) - (1 + \alpha)\alpha(\lambda_r - \lambda_n)^2 = 0. \quad (7.43)$$

U tom slučaju su koreni jednačine dati sa

$$s_{1/2} = \frac{-(\lambda_r + \lambda_n)}{2} \pm \frac{\sqrt{(\lambda_r + \lambda_n)^2 + 4\alpha(1 + \alpha)(\lambda_r - \lambda_n)^2}}{2}. \quad (7.44)$$

Nije teško uočiti da mora biti $\alpha < 0$. Uistinu, ako je $\alpha > 0$, jedan koren će imati pozitivan realni deo. Znači, mora biti $0 > \alpha > -1$. Šta se može reći o izboru α ? Iz jednačine (7.44) možemo videti da ako je α blisko 0 ili -1 imamo jedan koren karakterističnog polinoma čiji je realni deo veoma blizak nuli. To znači da u tom slučaju, algoritam sporo konvergira. Analizom (7.44) se može pronaći ono α koje obezbeđuje maksimalnu brzinu konvergencije.

U slučaju kada su svi d_i različiti jednačina (7.42) rezultuje sa:

$$s^2 + s(\lambda_r + \lambda_n - (d_n - d_r)(\lambda_r - \lambda_n)) - (d_n - d_r)(\lambda_r - \lambda_n)\lambda_n - (1 + d_r)d_n(\lambda_r - \lambda_n)^2 = 0. \quad (7.45)$$

Uvedimo sledeću oznaku

$$B = -(\lambda_r + \lambda_n - (d_n - d_r)(\lambda_r - \lambda_n)).$$

Koreni jednačine (7.45) su dati sa

$$s_{1/2} = \frac{B}{2} \pm \frac{\sqrt{B^2 + 4(d_n - d_r)(\lambda_r - \lambda_n)\lambda_n + 4d_n(1 + d_r)(\lambda_r - \lambda_n)^2}}{2}.$$

Ako je $B < 0$, oba korena će imati negativan realan deo u slučaju

$$4(d_n - d_r)(\lambda_r - \lambda_n)\lambda_n + 4d_n(1 + d_r)(\lambda_r - \lambda_n)^2 < 0. \quad (7.46)$$

Iz ove nejednakosti se vidi da vektor \mathbf{d} može biti izabran na mnogo načina tako da nejednakost (7.46) bude zadovoljena, ili drugim rečima da algoritam ima stabilno rešenje. Interesantno je zapaziti, ako su svi d_i pozitivni i različiti, i pošto za sopstvene vrednosti matrice \mathbf{C} važi

$$\lambda_1 > \lambda_2 > \dots > \lambda_N > \lambda_{N+1} \geq \lambda_{N+2} \geq \dots \geq \lambda_K > 0,$$

da je jedino rešenje ono u kojem većem d_i odgovara veće λ_i (samo u tom slučaju je $(d_n - d_r)(\lambda_r - \lambda_n) < 0$ za svako r i svako n).

7.4 Primer primene TOHM metoda za MCA

Sada ćemo primeniti TOHM metod na MSA metod predstavljen u [Douglas, 1998]:

$$\mathbf{W}(i+1) = \mathbf{W}(i) - \gamma(i) \left(\mathbf{x}(i) \mathbf{y}(i)^T \mathbf{W}(i)^T \mathbf{W}(i) \mathbf{W}(i)^T \mathbf{W}(i) - \mathbf{W}(i) \mathbf{y}(i) \mathbf{y}(i)^T \right), \quad (7.47)$$

gde $\mathbf{x}(i)$ predstavlja ulazni vektor sa srednjom vrednošću nula, $\mathbf{y}(i) = \mathbf{W}^T(i) \mathbf{x}(i)$ predstavlja izlazni vektor i $\mathbf{W}(i)$ predstavlja težinsku (sinaptičku) matricu.

U referenci [Douglas, 1998] je pokazano da je algoritam stabilan, tako da nije neophodno vršiti renormalizaciju vektora $\mathbf{w}_k(i)$, koji predstavlja k -tu klonu matrice $\mathbf{W}(i)$. Nije teško uočiti da je predloženi algoritam homogen sa aspekta pojedinačnog neurona.

Ako sada primenimo TOHM metod na algoritam (7.47), dobijamo sledeći zakon učenja

$$\begin{aligned} \mathbf{W}(i+1) = \mathbf{W}(i) - \gamma(i) \left(\mathbf{x}(i) \mathbf{y}(i)^T \mathbf{W}(i)^T \mathbf{W}(i) \mathbf{W}(i)^T \mathbf{W}(i) - \mathbf{W}(i) \mathbf{y}(i) \mathbf{y}(i)^T \right) \\ - \alpha \gamma(i) \left(\mathbf{x}(i) \mathbf{y}(i)^T - \mathbf{W}(i) \text{diag} \left(\mathbf{y}(i) \mathbf{y}(i)^T \right) \right), \end{aligned} \quad (7.48)$$

gde $\text{diag}(\mathbf{A})$ postavlja sve članove matrice \mathbf{A} , koji nisu na glavnoj dijagonali, na nulu i gde je $-1 < \alpha < 0$. Ovaj algoritam predstavlja MCA algoritam. Koristeći stohastičku aproksimaciju [Ljung, 1977], [Oja, 1985] znamo da algoritam (7.48) može biti povezan sa sledećom ordinarnom diferencijalnom jednačinom

$$\frac{d\mathbf{W}}{dt} = - \left(\mathbf{C} \mathbf{W} \mathbf{W}^T \mathbf{W} \mathbf{W}^T \mathbf{W} - \mathbf{W} \mathbf{W}^T \mathbf{C} \mathbf{W} \right) - \alpha \left(\mathbf{C} \mathbf{W} - \mathbf{W} \text{diag} \left(\mathbf{W}^T \mathbf{C} \mathbf{W} \right) \right). \quad (7.49)$$

Ova jednačina može biti korišćena za proučavanje ponašanja algoritma (7.48).

Dve teoreme koje slede opisuju ponašanje algoritma (7.49).

Teorema 7.3: Moguće je izabrati α tako da kolone \mathbf{W} pokrivaju sporedni podprstor matrice \mathbf{C} pod uslovom da su svi elementi matrice \mathbf{W} ograničeni.

Dokaz teoreme 7.3:

Neka su $\mathbf{E}_1 \in R^{K \times N}$ i $\mathbf{E}_2 \in R^{K \times (K-N)}$ matrice koje sadrže N sporednih i $(K-N)$ glavnih sopstvenih vektora matrice \mathbf{C} , respektivno. Možemo izraziti $\mathbf{W}(t)$ u (7.49) kao

$$\mathbf{W} = \mathbf{E}_1 \mathbf{A}_1 + \mathbf{E}_2 \mathbf{A}_2, \quad (7.50)$$

gde su $\mathbf{A}_1 = \mathbf{A}_1(t) \in R^{N \times N}$ i $\mathbf{A}_2 = \mathbf{A}_2(t) \in R^{(K-N) \times N}$. Analiza koja sledi će biti uprošćena ako usvojimo da važi sledeća nejednakost za sopstvene vrednosti matrice \mathbf{C} :

$$0 < \lambda_1 < \lambda_2 < \dots < \lambda_N < \lambda_{N+1} \leq \lambda_{N+2} \leq \dots \leq \lambda_K.$$

Drugim rečima, biće usvojeno da matrica \mathbf{C} ima K različitih striktno pozitivnih sopstvenih vrednosti sa odgovarajućim ortonormalnim vektorima. Analiza koja sledi može se obaviti i bez te pretpostavke ali se onda ista usložnjava. Obično su, u realnim situacijama kao što su situacije kada ulazni vektor \mathbf{x} predstavlja uzorke signala govora ili vektore nivoa sivog u prozorima kod procesiranja slike, sopstvene vrednosti kovarijanske matrice ulaznog signala različite i striktno pozitivne.

Napišimo (7.49) kao

$$\frac{d\mathbf{W}}{dt} = -\left(\mathbf{C}\mathbf{W}(\mathbf{W}^T \mathbf{W} \mathbf{W}^T \mathbf{W} + \alpha \mathbf{I}) - \mathbf{W} \Theta_N\right), \quad (7.51)$$

gde je \mathbf{I} jedinična matrica odgovarajućih dimenzija ($N \times N$) i

$$\Theta_N = \mathbf{W}^T \mathbf{C} \mathbf{W} + \alpha \text{diag}(\mathbf{W}^T \mathbf{C} \mathbf{W}). \quad (7.52)$$

Množenjem obe strane jedančine (7.51) sa \mathbf{E}_1^T dobijamo

$$\frac{d\mathbf{A}_1}{dt} = -\mathbf{\Lambda}_1 \mathbf{A}_1 (\mathbf{B} + \alpha \mathbf{I}) + \mathbf{A}_1 \Theta_N \quad (7.53)$$

gde je

$$\mathbf{\Lambda}_1 = \text{diag}\{\lambda_1, \dots, \lambda_N\} \text{ i}$$

$$\mathbf{B} = \mathbf{B}(t) = \left(\mathbf{A}_1^T \mathbf{A}_1 + \mathbf{A}_2^T \mathbf{A}_2\right)^2.$$

Množenjem obe strane jednačine (7.51) sa \mathbf{E}_2^T

$$\frac{d\mathbf{A}_2}{dt} = -\Lambda_2 \mathbf{A}_2 (\mathbf{B} + \alpha \mathbf{I}) + \mathbf{A}_2 \Theta_N \quad (7.54)$$

gde je

$$\Lambda_2 = \text{diag}\{\lambda_{N+1}, \dots, \lambda_K\}.$$

Razmotrimo matricu $\mathbf{S}(t) = \mathbf{A}_2(t)\mathbf{A}_1^{-1}(t)$. Ako je $\mathbf{A}_1(0)$ nesingularna, tada je $\mathbf{B}(0)$ pozitivno definitna, i tada je, na osnovu (7.53) $\mathbf{A}_1(t)$ nesingularna, pod pretpostavkom da je α izabrano na odgovarajući način (što je uvek moguće imajući u vidu rezultate u referenci [Douglas, 1998] – dokaz je moguće izvesti kontrapozicijom) i pod pretpostavkom da su komponente matrice \mathbf{W} ograničene. Sada možemo pisati

$$\frac{d\mathbf{S}}{dt} = \frac{d\mathbf{A}_2}{dt} \mathbf{A}_1^{-1} - \mathbf{A}_2 \mathbf{A}_1^{-1} \frac{d\mathbf{A}_1}{dt} \mathbf{A}_1^{-1}. \quad (7.55)$$

Zamenom (7.53) i (7.54) u (7.55) i uprošćavanjem, posle par transformacija dobijamo

$$\frac{d\mathbf{S}}{dt} = (-\Lambda_2 \mathbf{S} + \mathbf{S} \Lambda_1) \mathbf{A}_1 (\mathbf{B} + \alpha \mathbf{I}) \mathbf{A}_1^{-1} = [\Lambda \circ \mathbf{S}] \mathbf{M}, \quad (7.56)$$

gde je $\mathbf{M} = \mathbf{M}(t) = \mathbf{A}_1 (\mathbf{B} + \alpha \mathbf{I}) \mathbf{A}_1^{-1} \in R^{N \times N}$ i Λ° linearni operator koji deluje na $\mathbf{S}(t)$. Kako je $\mathbf{B}(t)$ kvadratna pozitivno definitna simetrična matrica, $\mathbf{B} + \alpha \mathbf{I}$ i $\mathbf{M}(t)$ su takođe pozitivno definitne matrice ukoliko je α izabrano na odgovarajući način (nije teško uočiti da α mora biti po apsolutnoj vrednosti, manje od najmanje sopstvene vrednosti matrice \mathbf{B}).

Sada ćemo pokazati da je Λ° negativan operator. Možemo smatrati da $\mathbf{S}(t)$ predstavlja vektor i onda možemo tražiti sopstvene vrednosti Λ° . Sopstvene vrednosti Λ° su $\lambda_{kl} = \lambda_k - \lambda_l$, $k \in \{1, \dots, N\}$, $l \in \{K-N+1, \dots, K\}$, i odgovaraju sopstvenim vektorima (vektorizovanim matricama) $\mathbf{S}_{kl}(t)$ čije su sve komponente nula izuzev elementa (k,l) koji ima jediničnu vrednost, jer je

$$\Lambda \circ \mathbf{S}_{kl} = (\lambda_k - \lambda_l) \mathbf{S}_{kl}. \quad (7.57)$$

Kako je po pretpostavci $\lambda_k - \lambda_l < 0$ za svaki par $\{k, l\}$, Λ° je negativan operator, i sve sopstvene vrednosti $[\Lambda^\circ] \mathbf{M}(t)$ imaju negativne realne delove. Tada je

$$\lim_{t \rightarrow \infty} \mathbf{S}(t) = 0,$$

i $\text{span}[\mathbf{W}] \rightarrow \text{span}[\mathbf{E}_1]$.

Teorema 7.4: Ako je $\text{rank}[\mathbf{W}] = N$, tada prostor lokalno asimptotski stabilnih tačaka jednačine (7.49) zadovoljava $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ i kolone matrice \mathbf{W} će biti jednake sopstvenim vektorima matrice \mathbf{C} .

Dokaz teoreme 7.4:

Stacionarne tačke jednačine (7.49) zadovoljavaju sledeću jednačinu

$$\mathbf{C} \mathbf{W} \mathbf{W}^T \mathbf{W} \mathbf{W}^T \mathbf{W} - \mathbf{W} \mathbf{W}^T \mathbf{C} \mathbf{W} + \alpha \mathbf{C} \mathbf{W} = \alpha \mathbf{W} \text{diag}(\mathbf{W}^T \mathbf{C} \mathbf{W}). \quad (7.58)$$

Ako je $\text{rank}[\mathbf{W}] = N$, tada iz teoreme 7.3 znamo da su potencijalne stacionarne tačke (na osnovu (7.50)) date u formi $\mathbf{W} = \mathbf{E}_1 \mathbf{A}_1$, gde $\mathbf{A}_1 \in R^{N \times N}$ predstavlja proizvoljnu nesingularnu matricu i \mathbf{E}_1 je definisano kao i ranije. Zamenom ove forme u (7.58) i posle određenog uprošćavanja dobijamo

$$\left(\Lambda_1 (\mathbf{A}_1 \mathbf{A}_1^T)^2 - \mathbf{A}_1 \mathbf{A}_1^T \Lambda_1 + \alpha \Lambda_1 \right) \mathbf{A}_1 = \alpha \mathbf{A}_1 \text{diag}(\mathbf{A}_1^T \Lambda_1 \mathbf{A}_1). \quad (7.59)$$

Množenjem sa desne strane (7.59) sa \mathbf{A}_1^T i posle grupisanja, dobijamo

$$\Lambda_1 (\mathbf{A}_1 \mathbf{A}_1^T)^3 + \alpha \Lambda_1 \mathbf{A}_1 \mathbf{A}_1^T = \mathbf{A}_1 \mathbf{A}_1^T \Lambda_1 \mathbf{A}_1 \mathbf{A}_1^T + \alpha \mathbf{A}_1 \text{diag}(\mathbf{A}_1^T \Lambda_1 \mathbf{A}_1) \mathbf{A}_1^T. \quad (7.60)$$

Očigledno je da desna strana jednačine (7.60) predstavlja simetričnu matricu. To znači da i leva strana jednačine (7.60) mora predstavljati simetričnu matricu i u tom slučaju mora biti:

$$\Lambda_1 \left(\left(\mathbf{A}_1 \mathbf{A}_1^T \right)^3 + \alpha \mathbf{A}_1 \mathbf{A}_1^T \right) = \left(\left(\mathbf{A}_1 \mathbf{A}_1^T \right)^3 + \alpha \mathbf{A}_1 \mathbf{A}_1^T \right) \Lambda_1. \quad (7.61)$$

Iz (7.61) nije teško zaključiti da je matrica

$$\mathbf{D} = \left(\mathbf{A}_1 \mathbf{A}_1^T \right)^3 + \alpha \mathbf{A}_1 \mathbf{A}_1^T \quad (7.62)$$

dijagonalna matrica jer je komutativna sa dijagonalnom matricom Λ_1 . Množenjem sa desne strane obe strane jednačine (7.62) sa $\mathbf{A}_1 \mathbf{A}_1^T$ dobijamo

$$\left(\mathbf{A}_1 \mathbf{A}_1^T \right)^4 + \alpha \left(\mathbf{A}_1 \mathbf{A}_1^T \right)^2 = \mathbf{D} \left(\mathbf{A}_1 \mathbf{A}_1^T \right). \quad (7.63)$$

Sada imamo da je matrica na levoj strani jednačine (7.63) simetrična što znači da i matrica na desnoj strani (7.63) mora predstavljati simetričnu matricu. To znači da važi

$$\mathbf{D} \left(\mathbf{A}_1 \mathbf{A}_1^T \right) = \left(\mathbf{A}_1 \mathbf{A}_1^T \right) \mathbf{D}, \quad (7.64)$$

što dalje znači da je matrica $\mathbf{A}_1 \mathbf{A}_1^T$ dijagonalna matrica. Ako označimo $\mathbf{A}_1 \mathbf{A}_1^T = \mathbf{D}_A$ i zamenimo u (7.59) imamo

$$\left(\Lambda_1 \mathbf{D}_A^2 - \mathbf{D}_A \Lambda_1 + \alpha \Lambda_1 \right) \mathbf{A}_1 = \mathbf{A}_1 \alpha \text{diag} \left(\mathbf{A}_1^T \Lambda_1 \mathbf{A}_1 \right). \quad (7.65)$$

Nije teško uočiti da jednačina (7.65) predstavlja jednačinu za određivanje sopstvenih vrednosti i sopstvenih vektora matrice $\mathbf{F} = \Lambda_1 \mathbf{D}_A^2 - \mathbf{D}_A \Lambda_1 + \alpha \Lambda_1$ pri čemu kolone matrice \mathbf{A}_1 predstavljaju sopstvene vektore te matrice koja je (očigledno) dijagonalana matrica. To znači da su kolone matrice \mathbf{A}_1 u obliku $[0 \dots a_k \dots 0]^T$. Sada ćemo pokazati da a_k može imati isključivo vrednosti -1 ili 1 .

Ako napišemo (7.65) za k -tu kolonu matrice \mathbf{A}_1 dobijamo sledeću jednačinu

$$\left(\lambda_k a_k^4 - \lambda_k a_k^2 + \alpha \lambda_k \right) a_k = \alpha \lambda_k a_k^3 \quad (7.66)$$

ili, posle određenih transformacija

$$a_k^4 - (1 + \alpha)a_k^2 + \alpha = 0. \quad (7.67)$$

Nije teško uočiti da (7.67) predstavlja bikvadratnu jednačinu čija su rešenja $a_k^2=1$ i/ili $a_k^2=\alpha$. Kako je $\alpha < 0$, jedino rešenje je $a_k^2=1$, što znači da je $a_k=\pm 1$. To dalje znači da su kolone matrice \mathbf{W} jednake sporednim sopstvenim vektorima matrice \mathbf{C} i posledično $\mathbf{W}^T\mathbf{W}=\mathbf{I}$.

Sada ćemo analizirati stabilnost u okolini stacionarnih tačaka. Prvo označimo ($\mathbf{c}_i=\mathbf{e}_i$)

$$\mathbf{U} = (\mathbf{c}_1, \dots, \mathbf{c}_N). \quad (7.68)$$

Tada je

$$\mathbf{C}\mathbf{U} = \mathbf{U}\mathbf{\Lambda} \quad \mathbf{U}^T\mathbf{C}\mathbf{U} = \mathbf{\Lambda}, \quad (7.69)$$

gde je $\mathbf{\Lambda}_1=\mathbf{\Lambda}=\text{diag}(\lambda_1, \dots, \lambda_N)$, i $\mathbf{\Lambda}_1$ je već ranije definisano. Sada ćemo pokazati da je rešenje jednačine (7.49) $\mathbf{W}=\mathbf{U}$, asimptotski stabilno rešenje. Da bismo to uradili, razmotrimo malo odstupanje $\mathbf{\Delta}=(\delta_1, \dots, \delta_N)$ od stacionarne tačke \mathbf{U} . U jednačini (7.49) stavimo

$$\mathbf{W} = \mathbf{U} + \mathbf{\Delta}.$$

Tada imamo

$$\begin{aligned} \frac{d\mathbf{\Delta}}{dt} = \frac{d\mathbf{W}}{dt} \Big|_{\mathbf{W} = \mathbf{U} + \mathbf{\Delta}} &= -\left(\mathbf{C}\mathbf{\Delta} + 2\mathbf{C}\mathbf{U}\mathbf{\Delta}^T\mathbf{U} + 2\mathbf{C}\mathbf{U}\mathbf{U}^T\mathbf{\Delta} - \mathbf{\Delta}\mathbf{\Lambda} - \mathbf{U}\mathbf{U}^T\mathbf{\Delta} - \mathbf{U}\mathbf{\Delta}^T\mathbf{C}\mathbf{U}\right) \\ &\quad - \alpha\left(\mathbf{C}\mathbf{\Delta} - \mathbf{\Delta}\mathbf{\Lambda} - 2\mathbf{U}\text{diag}(\mathbf{\Delta}^T\mathbf{C}\mathbf{U})\right) + O(\|\mathbf{\Delta}\|^2). \end{aligned} \quad (7.70)$$

Ako pretpostavimo da je $\|\mathbf{\Delta}\|$ malo, linearni član (po $\mathbf{\Delta}$) jednačine (7.70) određuje asimptotsku stabilnost. Za n -tu kolonu δ_n matrice $\mathbf{\Delta}$, linearni deo jednačine (7.70) daje

$$\frac{d(\delta_n)}{dt} = - \left(\mathbf{C}\delta_n + \sum_{m=1}^N \lambda_m \mathbf{c}_m (\mathbf{c}_m^T \delta_n) + \sum_{m=1}^N (2\lambda_m - \lambda_n) \mathbf{c}_m (\delta_m^T \mathbf{c}_n) - \lambda_n \delta_n \right) - \alpha (\mathbf{C}\delta_n - \lambda_n \delta_n - 2\mathbf{c}_n \lambda_n (\mathbf{c}_n^T \delta_n)). \quad (7.71)$$

Sada ćemo analizirati šta se dešava sa projekcijama vektora δ_n na različite sopstvene vektore \mathbf{c}_r , $r=1, \dots, K$. Prvo ćemo analizirati slučaj $r > N$. Tada je, usled ortonormalnosti sopstvenih vektora $\{\mathbf{c}_r\}$,

$$\frac{d(\mathbf{c}_r^T \delta_n)}{dt} = -(1 + \alpha)(\lambda_r - \lambda_n) \mathbf{c}_r^T \delta_n. \quad (7.72)$$

Pošto je $\alpha > -1$, tada $\mathbf{c}_r^T \delta_n$ teži asimptotski nuli kako $t \rightarrow \infty$, pošto je po pretpostavci $\lambda_r > \lambda_n$.

Sada ćemo analizirati slučaj $r \leq N$. Tada je

$$\begin{aligned} \frac{d(\mathbf{c}_r^T \delta_n)}{dt} = & -((2\lambda_r - \lambda_n)(\mathbf{c}_r^T \delta_n) + (2\lambda_r - \lambda_n)(\mathbf{c}_n^T \delta_r)) \\ & - \alpha((\lambda_r - \lambda_n)(\mathbf{c}_r^T \delta_n) - 2\delta(r, n)\lambda_n(\mathbf{c}_n^T \delta_n)). \end{aligned} \quad (7.73)$$

gde je $\delta(r, n)$ Kronekerov delta simbol. Možemo uočiti da su funkcije $\mathbf{c}_r^T \delta_n$ razdvojene u smislu da $\mathbf{c}_r^T \delta_n$ i $\mathbf{c}_n^T \delta_r$ zavise jedna od druge ali ne i od drugih funkcija. Slučaj $r = n$ je najjednostavniji i daje

$$\frac{d(\mathbf{c}_n^T \delta_n)}{dt} = -2(1 - \alpha) \mathbf{c}_n^T \delta_n \lambda_n. \quad (7.74)$$

To implicira da $\mathbf{c}_n^T \delta_n$ teži asimptotski nuli kako $t \rightarrow \infty$ pošto je $\alpha < 0$.

Konačno, razmotrićemo i slučaj kada je $r \neq n$, i kada su i r i n manji od N . Sada ćemo analizirati sledeći sistem od dve diferencijalne jednačine

$$\begin{aligned} \frac{d(\mathbf{c}_r^T \delta_n)}{dt} = & -((2 + \alpha)\lambda_r - (1 + \alpha)\lambda_n)(\mathbf{c}_r^T \delta_n) - (2\lambda_r - \lambda_n)(\mathbf{c}_n^T \delta_r) \\ \frac{d(\mathbf{c}_n^T \delta_r)}{dt} = & -((2 + \alpha)\lambda_n - (1 + \alpha)\lambda_r)(\mathbf{c}_n^T \delta_r) - (2\lambda_n - \lambda_r)(\mathbf{c}_r^T \delta_n). \end{aligned} \quad (7.75)$$

Neka je

$$\mathbf{X} = \begin{bmatrix} \mathbf{c}_r^T \boldsymbol{\delta}_n \\ \mathbf{c}_n^T \boldsymbol{\delta}_r \end{bmatrix} \quad (7.76)$$

$$\mathbf{A} = \begin{bmatrix} -((2+\alpha)\lambda_r - (1+\alpha)\lambda_n) & -(2\lambda_r - \lambda_n) \\ -(2\lambda_n - \lambda_r) & -((2+\alpha)\lambda_n - (1+\alpha)\lambda_r) \end{bmatrix}.$$

Tada jednačinu (7.75) možemo pisati kao

$$\frac{d\mathbf{X}}{dt} = \mathbf{A}\mathbf{X}. \quad (7.77)$$

Dobro je poznato da rešenje jednačine (7.77) teži asimptotski ka nuli ako i samo ako su realni delovi oba korena karakterističnog polinoma striktno negativni. Karakteristični polinom se može dobiti iz sledeće jednačine

$$\det(s\mathbf{I} - \mathbf{A}) = 0. \quad (7.78)$$

U našem slučaju

$$\det(s\mathbf{I} - \mathbf{A}) = (s + (2+\alpha)\lambda_r - (1+\alpha)\lambda_n) \\ (s + (2+\alpha)\lambda_n - (1+\alpha)\lambda_r) - (2\lambda_r - \lambda_n)(2\lambda_n - \lambda_r) = 0, \quad (7.79)$$

ili posle određenih transformacija

$$s^2 + (\lambda_r + \lambda_n)s - \alpha(\alpha+3)(\lambda_r - \lambda_n)^2 = 0. \quad (7.80)$$

Koreni ove jednačine su dati sa

$$s_{1/2} = \frac{-(\lambda_r + \lambda_n) \pm \sqrt{(\lambda_r + \lambda_n)^2 + 4\alpha(\alpha+3)(\lambda_r - \lambda_n)^2}}{2}. \quad (7.81)$$

Nije teško uočiti da pošto je $-1 < \alpha < 0$ oba korena imaju negativan realan deo što dalje znači da sporedni vektori matrice C predstavljaju asimptotski stabilna rešenja jednačine (7.49).

Sada ćemo analizirati ponašanje algoritma na bazi simulacije. Razmotrićemo simulaciju sistema malih dimenzija čije je ponašanje prikazano na slici 7.3. Broj ulaznih neurona je $K = 5$ i broj izlaznih neurona je $N = 3$. Vektori sa srednjom vrednošću nula i međusobno nekorelisanim komponentama su generisani na osnovu sledećih jednačina:

$$x(1,i) = .2 * \sin(i/2);$$

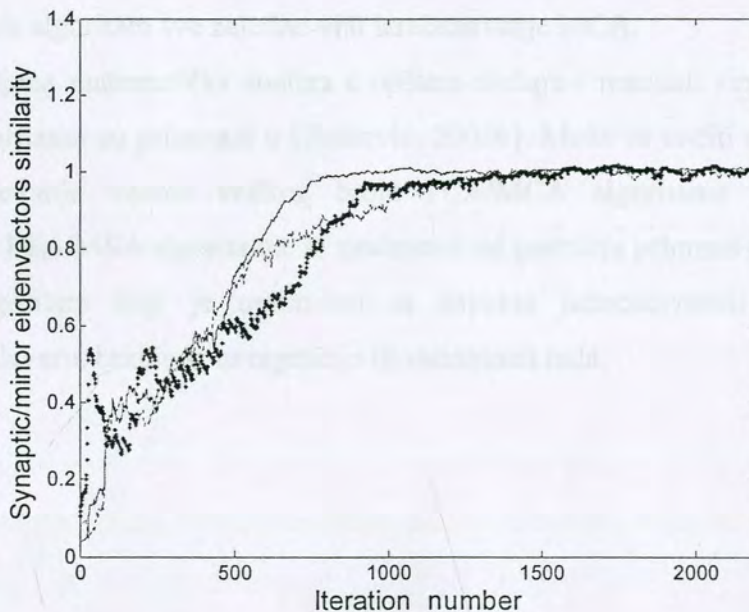
$$x(2,i) = .2 * ((\text{rem}(i,23) - 11) / 9) .^5;$$

$$x(3,i) = ((\text{rem}(i,27) - 13) / 9);$$

$$x(4,i) = ((\text{rand}(i,1) < .5) * 2 - 1) . * \log(\text{rand}(i,1) + .5);$$

$$x(5,i) = 1 * (-.5 + \text{rand}(i,1)).$$

U tom slučaju su tri sporedna vektora $\mathbf{c}_1 = (00001)^T$, $\mathbf{c}_2 = (01000)^T$ i $\mathbf{c}_3 = (10000)^T$. Inicijalna vrednost \mathbf{W} je bila $\mathbf{W}_{\text{init}} = 0.3 * \text{rand}(5,3)$, dok je faktor učenja bio $\gamma = 1 / (i/2000 + 6)$. Sličnost sinaptičkih vektora i odgovarajućih sopstvenih vektora je merena na osnovu vrednosti $\|\mathbf{w}_i^T \mathbf{c}_i\|$, $i = 1, 2, 3$, i prikazana je na slici 7.3.



Slika 7.3 Sličnost sinaptičkih i sopstvenih vektora u zavisnosti od broja iteracija (tačkasta linija predstavlja $\mathbf{w}_1^T \mathbf{c}_1$, debela puna linija predstavlja $\mathbf{w}_2^T \mathbf{c}_2$ i tanka puna linija predstavlja $\mathbf{w}_3^T \mathbf{c}_3$)

7.5 Generalizovani TOHM

Nedavno je predložena generalizacija TOHM metoda [Jankovic, 2005b] koja je nazvana GTOHM. Generalizovani zakon učenja za izračunavnje PCA/MCA može biti predstavljen sledećom jednačinom:

$$LA_{\substack{\text{PCA/} \\ \text{MCA}}} = FP_{\substack{\text{PSA/} \\ \text{MSA}}} + \mathbf{D}(i)IP_{\text{SPCAorSMCA}} \quad (7.82)$$

gde $LA_{\text{PCA/MCA}}$ definiše $\Delta\mathbf{W}_{\text{PCA/MCA}}$ (vidi (7.2)), FP definiše porodični deo zakona učenja (koji je PSA ili MSA zakon učenja) $\Delta\mathbf{W}_{\text{PSA/MSA}}$, $IP_{\text{SPCAorSMCA}}$ predstavlja individualni deo zakona učenja (PCA ili MCA algoritam za izračunavanje dominantne ili najsporednije glavne komponente) i $\mathbf{D}(i)$ je dijagonalna matrica čiji dijagonalni elemnti mogu biti u funkciji od vremena. Ako porodični deo zakona učenja predstavlja PSA algoritam tada je rezultujući algoritam PCA algoritam. Ako pak FP predstavlja MSA algoritam tada je rezultujući algoritam MCA tipa. Interesantno je zapaziti da individualni deo može predstavljati algoritam za izračunavanje najsporednije glavne komponente dok algoritam ukupno predstavlja PCA, kao i da individualni deo može biti algoritam za izračunavanje dominantne glavne komponente, dok algoritam sve zajedno vrši izračunavanje MCA.

Pojednostavljena matematička analiza u opštem slučaju i rezultati simulacija za nekoliko specifičnih algoritama su prikazani u [Jankovic, 2005b]. Može se uočiti da predloženi metod omogućava kreiranje veoma velikog broja PCA/MCA algoritama na bazi dostupnih SPCA/SMCA i PSA/MSA algoritama. U zavisnosti od područja primene može se konstruisati PCA/MCA algoritam koji je optimalan sa aspekta jednostavnosti implementacije u paralelnom hardveru, brzine konvergencije ili stabilnosti rada.

VIII Poglavlje

Zaključak

Ovaj rad je posvećen jednostavnim biološki verovatnim algoritmima za ekstrakciju glavnih komponenata i/ili njihovih potprostora iz kovarijanske matrice ulaznog signala, kao i pronalaženju generalnog metoda za transformaciju metoda za analizu glavnih i sporednih potprostora u metode za analizu glavnih i sporednih komponenata. U radu su proučavani jednostavni modeli koji, naravno, ne mogu da daju odgovor na osnovno pitanje na koji način se formira reprezentacija sveta u našem mozgu. Predloženi modeli u mnogo čemu ignorišu poznate činjenice o genetski determinisanim osobinama i anatomskim ograničenjima mozga, ali oni demonstriraju neke principe koji, možda, leže u osnovi našeg nervnog sistema, koji je za nas još uvek velika nepoznanica. Predloženi modeli su jednostavni, bazirani na lokalnim izračunavanjima i omogućavaju formiranje homogenih struktura, što sve zajedno daje dobru osnovu za jednostavnu implementaciju u paralelnom hardveru.

Izračunavanje glavnih/sporednih komponenata korišćenjem neuralnih mreža je problem koji je ispitivan u brojnim radovima u poslednjih dvadesetak godina. Oblast interesovanja istraživača je bila podeljena na nekoliko celina.

Jedan od pravaca istraživanja je bilo pronalaženje struktura mreža koje bi bile biološki verovatne. Ta vrsta mreža bi bila korišćena za ispitivanje hipoteza o mogućim računarskim konceptima koje primenjuje mozak. Jedan od osnovnih zadataka za istraživače koji su se bavili ovim problemom je bio rad na otkrivanju biološki verovatne strukture koja bi mogla da bude primenjena za modelovanje pojedinačnog realnog neurona. U literaturi je predloženo nekoliko pristupa koji su prikazani u trećem poglavlju, a detaljnije analizirani u petom poglavlju. Uglavnom su se sve metode bazirale na Hebovom zakonu učenja koji je bio primenjivan na mreži koja je sadržala samo direktne veze od ulaza ka izlazu. Kao što je i

opisano u poglavlju pet, u tom slučaju je neophodno da se izvrši modifikacija Hebovog zakona učenja kako bi se izvršila njegova stabilizacija. Manji broj radova se bavio drugim mogućim načinima stabilizacije Hebovog zakona učenja, kao što je izmena strukture mreže na kojoj se zakon primenjuje, ili promena načina na koji se izračunava potencijal izlaznog neurona.

U petom poglavlju je predstavljen ∞ OH metod, koji predstavlja linearni model pojedinačnog neurona koji vrši ekstrakciju dominantne glavne komponente kovarijanske matrice stacionarnog ulaznog signala. Primena originalnog Hebovog zakona učenja u ∞ OH neuralnoj mreži, ne dovodi do divergencije norme sinaptičkog vektora zahvaljujući usvojenoj strukturi mreže i usvojenom načinu izračunavanja vrednosti potencijala izlaznog neurona. Predloženi metod koristi oba kraja sinapse simetrično, i ne zahteva nikakva dodatna lokalna preračunavanja. Sa tehničke tačke gledišta ovo se može smatrati dobrom stranom predloženog algoritma. Originalna ideja, koja je korišćena pri konstrukciji strukture mreže, je da neuron pokušava da neutrališe (potisne) izvor promene na svom ulazu. U tom slučaju je prisustvo povratnih veza neizbežno.

Druga oblast istraživanja, u oblasti izračunavanja glavnih komponentata na bazi neuralnih mreža, je vezana za realizaciju efikasnih algoritama za izračunavanje glavnog potprostora kovarijanske matrice ulaznog signala čija je srednja vrednost nula. Pored nastojanja da se koriste biološki verovatni zakoni učenja, dva aspekta su posebno naglašavana kod realizacije takvih mreža, a to su brzina konvergencije algoritma i broj globalnih kalkulacija koje mreža koristi. U principu je pokazano da mreže koje sadrže samo lokalna preračunavanja za modifikaciju vrednosti sinaptičke matrice konvergiraju mnogo sporije ka konačnom rešenju od mreža koja koriste globalna izračunavanja. U literaturi nije dostupna informacija da je neko od predloženih rešenja bazirano na imitaciji neke od struktura koja se nalazi u realnim nervnim sistemima. Takođe, da bi se obezbedio stabilan rad ovih mreža, uglavnom je neophodno da se faktor učenja sa protokom vremena smanjuje i teži nuli. Ovi algoritmi su opisani u četvrtom poglavlju. Pojedini algoritmi su detaljnije analizirani na kraju šestog poglavlja.

Šesto poglavlje opisuje MH i MHO metod koji se mogu koristiti za analizu glavnog potprostora kovarijanske matrice ulaznog signala. Ova dva metoda su inspirisana strukturom dela retine kod riba. MH metod je baziran na ekstenziji ∞ OH metoda. «Stabilizacija» originalnog Hebovog zakona učenja je postignuta uvođenjem samo jednog dodatnog modulišućeg signala. Originalna ideja je bila da se umesto velikog broja lokalnih povratnih

veza, koristi globalna povratna veza koja vrši stabilizaciju zakona učenja. Kod MHO metoda je potrebna primena i lokalnih povratnih veza kako bi norma sinaptičkih vektora bila 1. U oba metoda zakon učenja je baziran na lokalnim izračunavanjima. Jedina globalna izračunavanja koja su potrebna u slučaju MH i MHO algoritma su izračunavanja energije koja se sadrži u ulaznom i izlaznom signalu. Modifikacija pojedinih sinapsi u MH i MHO algoritmu ne zahteva podatak o eksplicitnoj vrednosti sinapse koja je vezana za bilo koji drugi neuron. Ovo se može smatrati dobrom osobinom algoritma. Činjenica da se može uočiti velika strukturna sličnost sa delom retine kod riba ukazuje na to da ovaj algoritam zaslužuje dalju analizu. Na kraju šestog poglavlja je pokazano, na bazi simulacija, da su MH i MHO algoritmi stabilni čak i u slučaju kada je faktor učenja konstantan u toku vremena, što je veoma povoljna osobina za eventualnu hardversku realizaciju algoritma.

Treće važno područje koja je istraživano u oblasti izračunavanja glavnih/sporednih komponenata u neuralnim mrežama je vezano za otkrivanje algoritama/mreža koji će omogućiti paralelnu ekstrakciju većeg broja glavnih/sporednih komponenata istovremeno. Za razliku od velikog broja algoritama za izračunavanje PSA/MSA broj algoritama za izračunavanje PCA/MCA je relativno mali. Uglavnom su PCA/MCA algoritmi dobijani modifikacijom određenog PSA/MSA algoritma u smislu uvođenja određene nesimetrije ili nelinearnosti, pri čemu je ta modifikacija uvek bila specifična i vezana za izabrani PSA/MSA algoritam. Uvođenje nesimetrije i nelinearnosti je dalje značilo da ti algoritmi nisu pogodni za implementaciju u paralelnom hardveru. Jedini poznati algoritam za formiranje homogenih struktura za derivaciju PCA/MCA je bigradijentni algoritam. Taj algoritam transformiše algoritme za izračunavanje jedne glavne/sporedne komponente u algoritme za ekstrakciju više glavnih/sporednih komponenata uvođenjem tzv. penala, koji forsira međusobnu ortogonalnost sinaptičkih vektora koji čine sinaptičku matricu. Međutim, taj algoritam po svojoj strukturi ne može biti korišćen za modelovanje biološki verovatnih mreža.

Generalni metod za transformaciju PSA/MSA metoda u PCA/MCA metode je prikazan u sedmom poglavlju. Metod je baziran na originalnoj ideji o porodičnom principu koji kaže da svaki neuron pokušava da uradi ono što je najbolje za njegovu porodicu i zatim ono što je najbolje za njega samog. Primena ove ideje je rezultovala generalnim algoritmom koji je nazvan vremenski orijentisan hijerarhijski metod (TOHM). Primena ovog metoda omogućava formiranje homogenih (simetričnih) struktura za realizaciju PCA/MCA u paralelnom hardveru. Uvođenje vremenske hijerarhije u realizaciju algoritma dovodi do zanimljive hipoteze o neophodnosti postojanja više od jedne vrste neurotransmitera, kao i hipoteze da različite vrste neurotransmitera odgovaraju procesima na različitim vremenskim skalama. U

radu je prikazana uprošćena matematička analiza u opštem slučaju. Stabilnost metoda je detaljno analizirana na jednom PCA i jednom MCA metodu. Ono što je potrebno naglasiti je da primena ovog metoda dovodi do formiranja više stotina (pa i hiljada) novih PCA/MCA algoritama. U zavisnosti od željenog područja primene može se formirati optimalan algoritam korišćenjem poznatih algoritama za PSA/MSA i odgovarajućih algoritama za izračunavanje pojedinačne glavne/sporedne komponente.

Teorijski doprinos ovog rada se ogleda u sledećem:

- pokazano je da direktna primena Hebovog zakona ne dovodi do divergencije sinaptičkog vektora ako se taj zakon primeni na mreži odgovarajuće strukture;
- predložena je struktura neuralne mreže koja je u mnogo čemu slična sa strukturom dela retine kod riba;
- prikazan je generalni metod koji transformiše PSA/MSA metode u PCA/MCA metode i tako omogućava formiranje veoma velikog broja novih PCA/MCA algoritama. Korišćenjem ove transformacije moguće je formiranje homogenih algoritama na bazi Hebovog zakona učenja, koji koriste samo lokalno dostupne podatke za modifikaciju vrednosti sinaptičke matrice, i koji bi onda mogli biti smatrani za biološki verovatne.

Praktična primena originalnih metoda koje su predložene u ovom radu je moguća pri rešavanju bilo kog problema gde se zahteva izračunavanje glavnih/sporednih komponentata ili njihovih potprostora u neuralnim mrežama. Međutim, njihova glavna primena se može naći u mogućem modelovanju računskih principa koji se koriste u realnim neuralnim mrežama i kod jednostavne realizacije PSA/MSA ili PCA/MCA algoritama u paralelnom hardveru.

Radovi koji su objavljeni u vezi sa temom ove teze su:

A. Radovi vezani za ∞ OH metod

[1] Jankovic, M., "A new simple ∞ OH neuron model as a principal component analyzer", CCECE '01, Toronto, Canada, May 2001, pp. 981-986, (CD ROM code: f45ta207).

[2] Jankovic, M., "A simple biologically inspired principal component analyzer – ModH neuron model", NEUREL-2002, Belgrade, Yugoslavia, September 2002, pp. 23-26.

[3] Jankovic, M., "A new simple ∞ OH neuron model as a biologically plausible principal component analyzer", IEEE Trans. on Neural Networks, vol. 14, pp. 853-859, 2003.

B. Radovi vezani za MH i MHO metod

[1] Jankovic, M., "A new modulated Hebbian learning rule – method for local computation of a principal subspace", ICONIP2001, Shanghai, China, November 2001, Vol. 1, pp. 470-475.

[2] Jankovic, M. and Ogawa, H. "A new modulated Hebb learning rule – Biologically plausible method for local computation of principal subspace", *Int. J. Neural Systems*, Vol.13, no.4, pp. 215-224, 2003.

[3] Jankovic, M., Ogawa, H., "Modulated Hebb Oja learning rule – a method for principal subspace analysis", *IEEE Trans. on Neural Networks*, vol. 17, no. 2, pp. 345-356, 2006.

C. Radovi vezani za TOHM metod

[1] Jankovic, M. and Ogawa, H. "Method for transformation of principal subspace algorithms to principal components algorithms", ICEST' 04, pp. 65-68, Bitola, Macedonia, 2004.

[2] Jankovic, M. and Ogawa, H., "Time-oriented hierarchical method for computation of principal components using subspace learning algorithm", *Int. J. Neural Systems*, Vol.14, no.5, pp.313-324, 2004.

[3] Jankovic, M. and Ogawa, H., "Time-oriented hierarchical method for computation of minor components", ICANNGA '05, pp. 38-41, Portugal, 2005.

[4] Jankovic, M. and Reljin, B., "Neural learning on Grassman/Stiefel principal/minor submanifold", EUROCON 2005, pp. 249-252, Serbia and Montenegro, 2005.

[5] Jankovic, M. and Reljin, B., "A new minor component analysis method based on Douglas-Kung-Amari minor subspace analysis method", *IEEE Signal Processing Letters*, vol. 12, pp. 859-862, 2005.

Dalja analiza može biti usmerena na sledeće probleme:

1. Matematička analiza ∞ HO metoda u opštijem slučaju.
2. Matematička analiza MHO metoda u opštijem slučaju.
3. Pokušaj daljeg smanjenja globalnih izračunavanja u MHO modelu modifikacijom strukture neuralne mreže.
4. Dalja generalizacija TOHM metoda kako sa aspekta primenjenog individualnog zakona učenja, tako i načina na koji se taj zakon primenjuje. Na primer, umesto uzimanja individualnog člana zakona učenja sa težinom koja je manja od jedan, mogao bi se primeniti i princip drugačijeg vremena odabiranja koji bi se primenio na taj deo zakona učenja (odnosno taj deo zakona učenja bi se ubacivao u formulu svaki n -ti put).
5. Eventualna analiza stabilnosti TOHM u opštem slučaju.
6. Analiza stabilnosti PCA/MCA algoritama dobijenih primenom TOHM metoda na PSA/MSA metode koji su bazirani na učenju na ortogonalnim grupama ili Stifelovoj mnogostrukosti.

Literatura

- [1] K. Abed-Meraim, A. Chkief and Y. Hua (2000) "Fast orthonormal PAST algorithm", *IEEE Signal Processing Letters*, vol. 7, no. 3, pp. 60-62.
- [2] S.-I. Amari (1978) "Neural theory of association and concept formation", *Biological Cybernetics*, vol. 26, pp. 175-185.
- [3] J.J. Atick and A.N. Redlich (1990) "Towards a theory of early visual processing", *Neural Computation*, vol. 3, pp. 308-320.
- [4] J.J. Atick and A.N. Redlich (1992) "What does the retina know about natural scenes?", *Neural Computation*, vol. 4, pp. 196-210.
- [5] J.J. Atick and A.N. Redlich (1993) "Convergent algorithm for sensory receptive field development", *Neural Computation*, vol. 5, pp. 45-60.
- [6] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe and H. van der Vorst (editors), (2000) *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia.
- [7] P. Baldi and K. Hornik (1995) "Learning in linear neural networks: A survey", *IEEE Trans. Neural Networks*, vol. 6, pp. 837-858.
- [8] S. Bannour and M.R. Azimi-Sadjadi (1989) "Principal component extraction using recursive least squares learning", *IEEE Trans. on Neural Networks*, vol. 6, pp. 456-469.
- [9] E.L. Bienenstock, L.N. Cooper and P.W. Munro (1982) "Theory for the Development of Neuron Selectivity: Orientation Specificity and Binocular Interaction in Visual Cortex", *The Journal of Neuroscience*, vol. 2, No. 1, pp. 32-48.
- [10] R.W. Brockett (1991) "Dynamical systems that learn subspaces", *Mathematical system theory: The influence of R.E. Kalman*, pp. 579-592, Springer, Berlin.

- [11] R.W. Brockett (1991) "Dynamical systems that sort lists, diagonalize matrices and solve linear programming problems", *Linear Algebra Applications*, vol. 146, pp. 79-91.
- [12] T.H. Brown, E. W. Kairiss and C. L. Keenan (1990) "Hebbian synapses: Biophysical mechanisms and algorithms", *Annu. Rev. Neurosci.*, pp. 475-511.
- [13] A. Cattaneo, L. Maffei and C. Morrone (1981) "Patterns in the discharge of simple and complex visual cortical cells", *Proc. R. Soc. London B*, vol. 212, pp. 279-297.
- [14] G.A. Carpenter and S. Grossberg (1988) "The ART of adaptive pattern recognition by a self-organizing neural network", *Computer*, 21, pp. 77-88.
- [15] W.K. Chen (1971) *Applied Graph Theory*, John Willey and Sons, New York.
- [16] T.-P. Chen and S. Amari (2001) "Unified stabilization approach to principal and minor components", *Neural Networks*, vol. 14, pp. 1377-1387.
- [17] T.-P. Chen, S. Amari, and Q. Lin (1998) "A unified algorithm for principal and minor components extraction", *Neural Networks*, vol. 11, pp. 385-390.
- [18] T. Chen, Y. Hua and W. Yan (1998) "Global Convergence of Oja's Subspace Algorithm for Principal Component Extraction", *IEEE Trans. Neural Networks*, vol. 9, no. 1, pp.58-67.
- [19] C. Chatterjee, Z. Kung and V.P. Roychowdhury (2000) "Algorithms for Accelerated Convergence of Adaptive PCA", *IEEE Trans. Neural Networks*, vol. 11, no. 2, pp. 338-355.
- [20] C. Chatterjee, V.P. Roychowdhury and E.K.P. Chong (1998) "On relative convergence properties of principal component analysis algorithms", *IEEE Trans. on Neural Networks*, vol. 9, no. 2, pp. 319-329.
- [21] C. Chatterjee, Z. Kung and V.P. Roychowdhury (2000) "Algorithms for accelerated convergence of adaptive PCA", *IEEE Transactions on Neural Networks*, vol. 11, no. 2, pp. 338-355.
- [22] A. Cichocki, S.-I. Amari, (2002) *Adaptive Blind Signal and Image Processing – Learning Algorithms and Applications*, John Wiley and Sons, New York.
- [23] A. Cichocki, R.R. Gharieb, and T. Hoya (2001) "Efficient extraction of evoked potentials by combination of independent component analysis and cumulant-based matched filtering", in *Proc. 11th IEEE Signal Processing Workshop on Statistical Signal Processing*, pp. 237-240, Singapore.
- [24] A. Cichocki and R. Unbehauen (1993) "Robust estimation of principal components in real time", *Electronics Letters*, vol. 29, no. 21, pp. 1869-1870.

- [25] A. Cichocki and R. Unbehauen (1994) *Neural Networks for Optimization and Signal Processing*, John Wiley & Sons, New York.
- [26] A. Cichocki, R. Swiniarski and R.E. Bogner (1996) "Hierarchical neural networks for robust PCA of complex-valued signals", in *Proc. World Congress on Neural Networks, WCNN-96, USA*, pp. 818-821.
- [27] G. Cirrincione (1998) "A Neural Approach to the Structure from Motion Problem", Ph.D. thesis, INPG Grenoble, France.
- [28] J. E. Dayhoff and G. L. Gerstein (1983) "Favored patterns in spike trains. II", *Application. Journal of Neurophysiology*, vol. 49, pp. 1349-1363.
- [29] P. Dayan, G.E. Hinton, R.M. Neal and R.S. Zemel (1995) "The Helmholtz machine", *Neural Computation* 7(5), pp. 889-904.
- [30] G. Deco and D. Obradovic (1996) *An Information-Theoretic Approach to Neural Computing*, Springer-Verlag New York Inc.
- [31] K.I. Diamantaras (1996) "Robust principal component extracting neural networks", *ICNN'96, Washington, D.C., USA*, pp. 74-77.
- [32] K.I. Diamantaras and S.Y. Kung (1996) *Principal Component Neural Networks – Theory and Applications*, John Wiley & Sons Inc., New York.
- [33] R. J. Douglas and K. A. C. Martin (1991) "Opening the gray box", *Trends in Neuroscience*, vol. 14, pp. 286-293.
- [34] R. J. Douglas, K. A. C. Martin and D. Whitteridge (1988) "Selective responses of visual cortical cells do not depend on shunting inhibition", *Nature*, vol. 332, pp. 642-644.
- [35] S.C. Douglas, S.Y. Kung and S. Amari (1998) "A self-stabilized minor subspace rule", *IEEE Signal Processing Letters*, vol. 5, no. 12, pp. 328-330.
- [36] S.C. Douglas, S.-Y. Kung, S. Amari (1999) "On the numerical stabilities of principal, minor and independent component analysis algorithms", in *Proc. IEEE International Conference on Independent Component Analysis and Signal Separation, France*, pp. 419-425.
- [37] J. Dowling (1987) *The Retina: An Approachable Part of the Brain*, The Belknap Press of Harvard University Press.
- [38] A. Edelman, T.A. Arias and S.T. Smith (1998) "The geometry of algorithms with orthogonality constraints", *SIAM Journal on Matrix Analysis Applications*, vol. 20, no. 2, pp. 303-353.
- [39] C. Enroth-Cugell and J.G. Robson (1966) "The contrast sensitivity of retinal ganglion cells of the cat", *Journal of Physiology*, 187, pp. 517-552.

- [40] S. Fiori (2002) "A minor subspace algorithm based on neural Stiefel dynamics", *International Journal of Neural Systems*, vol. 12, no. 5, pp. 339 - 350.
- [41] S. Fiori (2003) "A Neural Minor Component Analysis Approach to Robust Constrained Beamforming", *IEE Proceedings - Vision, Image and Signal Processing*, vol. 150, pp. 205-218.
- [42] S. Fiori (2001) "A theory for learning by weight flow on Stiefel-Grassman manifold", *Neural Computation*, vol. 13, pp. 1625-1647.
- [43] L. Fox (1962) *Numerical Solution of Ordinary and Partial Differential Equations*, Pergamon Press.
- [44] C. Fyfe and R.J. Baddeley (1995a) "Finding compact and sparse distributed representations of visual images", *Network: Computation in Neural Systems* vol. 6, no.3, pp. 334-344.
- [45] C. Fyfe and R.J. Baddeley (1995b) "Nonlinear data structure extraction using simple Hebbian networks", *Biological Cybernetics*, vol. 72, no.6, 533-541.
- [46] P. Foldiak (1992) "Models of Sensory Coding", Technical Report CUED/F-INFENG/TR 91, Cambridge University Engineering Department, UK.
- [47] P. Foldiak (1991) "Learning invariance from transformation sequence", *Neural Computation*, vol. 3, pp. 194-200.
- [48] M.C. Goodall (1960) "Performance of stochastic net", *Nature* 185, pp. 557-558.
- [49] S. Grossberg (1973) "Contour enhancement, short-term memory, and constancies in reverberating neural networks", *Studies in Applied Mathematics*, vol. 52, pp. 213-257.
- [50] S. Grossberg (1988) "Nonlinear neural networks: principles, mechanisms, and architectures", *Neural Networks*, vol. 1, pp. 17-61.
- [51] G.F. Harpur (1997) "Low entropy coding with unsupervised neural networks", Ph.D. thesis, Cambridge University Engineering Department, UK.
- [52] G.F. Harpur and R.W. Prager (1994) "Experiments with simple Hebbian-based learning rules in pattern classification tasks", Technical report CUED/F-INFENG/TR 168, Cambridge University Engineering Department, UK.
- [53] D.O. Hebb (1949) *The Organisation of Behaviour*, New York: Wiley.

- [54] G.E. Hinton (1987) "Learning translation invariant recognition in a massively parallel network", In G. Goos & J. Hartmanis (Eds.), *PARLE: Parallel Architectures and Languages Europe* (pp. 1-13). Berlin: Springer-Verlag.
- [55] H. Hotelling (1933) "Analysis of a complex of statistical variables into principal components", *Journal of Educational Psychology*, vol. 24, pp. 417-441.
- [56] Y. Hua, Y. Xiang, T.-P. Chen, K. Abed-Meraim and Y.Miao (1999) "A new look at the power method for fast subspace tracking", *Digital Signal Processing*, vol. 9, pp. 297-314.
- [57] H. Hubel and T. N. Wiesel, (1974) "Uniformity of monkey striate cortex: A parallel between field size, scatter, and magnification factor", *J. Comu. Neurol.*, vol. 158, pp. 295-305.
- [58] N. Intrator and L.N. Cooper (1992) "Objective function formulation of the BCM theory of visual cortical plasticity: statistical connections, stability conditions", *Neural Networks*, vol.5, pp. 3-17.
- [59] M. Jankovic (2001a) "A new simple ∞ OH neuron model as a principal component analyzer", in *Proc.CCECE '01*, Toronto, Canada, pp. 981-986, (CD ROM code: f45ta207).
- [60] M. Jankovic (2001b) "A new modulated Hebbian learning rule – method for local computation of a principal subspace", in *Proc. ICONIP2001*, Shanghai, China, vol. 1, pp. 470-475.
- [61] M. Jankovic (2002) "A simple biologically inspired principal component analyzer – ModH neuron model", in *Proc. NEUREL-2002*, Belgrade, Yugoslavia, pp. 23-26.
- [62] M. Jankovic (2003a) "A new simple ∞ OH neuron model as a biologically plausible principal component analyzer", *IEEE Trans. on Neural Networks*, vol. 14, pp. 853-859.
- [63] M. Jankovic and H. Ogawa (2003b) "A new modulated Hebb learning rule – Biologically plausible method for local computation of principal subspace", *Int. J. Neural Systems*, vol.13, no.4, pp. 215-224.
- [64] M. Jankovic and H. Ogawa (2004a) "Method for transformation of principal subspace algorithms to principal components algorithms", in *Proc. ICEST' 04*, pp. 65-68, Bitola, Macedonia.
- [65] M. Jankovic and H. Ogawa (2004b) "Time-oriented hierarchical method for computation of principal components using subspace learning algorithm", *Int. J. Neural Systems*, vol.14, no.5, pp.313-324.

- [66] M. Jankovic and H. Ogawa (2005a) "Time-oriented hierarchical method for computation of minor components", in *Proc. ICANNGA '05*, pp. 38-41, Portugal.
- [67] M. Jankovic and B. Reljin (2005b) "Neural learning on Grassman/Stiefel principal/minor submanifold", in *Proc. EUROCON 2005*, pp. 249-252, Serbia and Montenegro.
- [68] M. Jankovic and B. Reljin (2005c) "A new minor component analysis method based on Douglas-Kung-Amari minor subspace analysis method", *IEEE Signal Processing Letters*, vol. 12, no. 12, pp. 859-862, 2005.
- [69] M. Jankovic and H. Ogawa (2006) "Modulated Hebb Oja learning rule – a method for principal subspace analysis", *IEEE Trans. on Neural Networks*, vol. 17, no. 2, pp. 345-356, 2006.
- [70] K. Karhunen (1947) "Über lineare Methoden in der Wahrscheinlichkeitsrechnung," *Ann. Acad. Sci. Fenn., Ser. A.1.: Math.-Phys.*, vol. 37, pp. 3–79.
- [71] J. Karhunen, A. Cichocki, W. Kasprzak, and P. Pajunen (1997) "On neural blind separation with noise suppression and redundancy reduction", *Int. Journal on Neural Systems*, vol. 8, no. 2, pp. 219-237.
- [72] R. Klemm (1987) "Adaptive airborne MTI: an auxiliary channel approach", *IEE Proceedings*, vol. 134, pp. 269-276.
- [73] T.P.Krasulina (1970) "Method of stochastic approximation in the determination of the largest eigenvalue of the mathematical expectation of random matrices," *Automat. Remote Contr.*, vol. 2, pp. 215-221.
- [74] T. Kohonen (1982) "Self-organization of Topologically Correct Feature Maps", *Biological Cybernetics*, vol. 43, pp. 59-69.
- [75] T. Kohonen (1995) *Self-Organizing Maps*, Springer, Berlin (3rd Edition, 2001)
- [76] T. Kohonen (1982) "Self-organization of topologically correct feature maps", *Biological Cybernetics*, vol. 43, pp. 59-69.
- [77] S.Y. Kung, K. Diamantaras and J. Taur (1991) "Neural networks for extracting pure/constrained/oriented principal components", in J.R. Vaccaro, editor, *SVD and Signal Processing*, pp. 57-81, Elsevier Science, Amsterdam.
- [78] A.P. Liavas, P.A. Regalia and J.-P. Delmas (1999) "Blind channel approximation: Effective channel order determination", *IEEE Trans. Signal Processing*, vol. 47, pp. 3336-3344.
- [79] R. Linsker (1988) "Self-organization in a perceptual network", *Computer*, vol. 21, pp. 105-117.

- [80] L. Ljung (1977) "Analysis of recursive stochastic algorithms", *IEEE Trans. Automat. Contr.*, vol. 22, pp. 551-575.
- [81] M. Loève (1948) "Fonctions aléatoires du second ordre," in *Processus stochastiques et mouvements Browniens* (P. Lévy, ed.), Paris, France: Gauthier-Villars.
- [82] E.N. Lorenz (1956) "Empirical orthogonal functions and weather prediction", Sci. Rept. No. 1, Contract AF19(604)-1566, Dept. of Meteorology, M.I.T., 49.
- [83] H. Luo, R.-W. Liu and Y. Li (1997) "Internal structure identification for layered medium", in *Proc. IEEE ISCAS*, pp. 161-164, USA.
- [84] G. Mathew and V. Reddy (1994) "Orthogonal eigensubspace estimation using neural networks", *IEEE Trans. on Signal Processing*, vol. 42, pp. 1803-1811.
- [85] C. Metin and D. O. Frost (1989) "Visual responses of neurons in somatosensory cortex of hamsters with experimentally induced retinal projections to somatosensory thalamus", *Proc. Natl. Acad. Sci. USA*, 86, 357-361.
- [86] T.P. Minka (2001) "Automatic choice of dimensionality for PCA", In NIPS 13 in Todd K. Leen, Thomas G. Dietterich and Volker Tresp, editors, *Advances in Neural Information Processing Systems*, pp. 598-604, MIT Press.
- [87] J.A. Movshon, I.D. Thompson and D.J. Tolhurst (1978) "Spatial summation in the receptive fields of simple cells in the cat's striate cortex", *Journal of Physiology*, vol. 283, pp. 53-77.
- [88] H. Ogawa (1992) "Karhunen-Loève subspace", in *Proc. International Conference on Pattern Recognition*, Hague, The Netherlands, pp. 75-78.
- [89] E. Oja (1982) "A simplified neuron model as a principal component analyzer", *J. Math. Biol.*, vol. 15, pp. 267-273.
- [90] E. Oja (1983) *Subspace Method of Pattern Recognition*, Research Studies Press and J. Wiley, Letchworth.
- [91] E. Oja and J. Karhunen (1985) "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix", *J. Math. Anal., Appl.*, 106, pp. 69-84.
- [92] E. Oja (1989) "Neural networks, principal components, and subspaces", *Int. J. Neural Systems*, 1, pp. 61-68.
- [93] E. Oja (1992a) "Principal components, minor components and linear neural networks", *Neural Networks*, vol. 5, pp. 927-935.

- [94] E. Oja, H. Ogawa and J. Wangviwattana (1992b) "Principal component analysis by homogeneous neural networks, Part I: The weighted subspace criterion", *IEICE Trans. Inf.&Syst.*, E75-D, 3, pp. 366-375.
- [95] E. Oja, H. Ogawa and J. Wangviwattana (1992c) "Principal component analysis by homogeneous neural networks, Part II: Analysis and extensions of the learning algorithm", *IEICE Trans. Inf.&Syst.*, E75-D, 3, pp. 376-382.
- [96] E. Oja, J. Karhunen and A. Hyvarinen (1997) "From neural principal components to neural independent components", in *Proc. Int. Conference on Artificial Neural Networks*, Lausanne, Switzerland.
- [97] E. Oja, S. Kaski (Eds.) (1999) "Kohonen Maps", Elsevier, Amsterdam.
- [98] S. Ouyang, Z. Bao and G. Liao (2000) "Robust recursive Least Squares Learning Algorithm for Principal Component Analysis", *IEEE Trans. Neural Networks*, vol. 11, no. 1, pp. 215-221.
- [99] K. Pearson (1901). *On lines and planes of closest fit to systems of points in space*, The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science, Sixth Series 2, 559-572.
- [100] M. Plumbley (1991) "On information theory and unsupervised neural networks", Technical Report CUED/F-INFENG/TR. 78, Cambridge University Engineering Department, UK.
- [101] Y. Rao and J.C. Principe (2002) "Time series segmentation using novel adaptive eigendecomposition algorithm", *Journal of VLSI Signal Processing*, vol. 32(1/2), pp. 7-17.
- [102] I. Reljin, G. Jovanovic, B. Reljin (2002a) "The use of SOM neural network for precipitation pattern recognition in Yugoslavia", in *Proc. 18th Int. Conf. on Carpathian Meteorology*, pp. 24-25, Belgrade, Oct. 7-11.
- [103] I. Reljin, B. Reljin, G. Jovanovic (2002b) "Clustering of climate data in Yugoslavia by using SOM neural network", in *Proc. 2002 6th Seminar NEUREL*, pp. 203-206, Belgrade, Sept. 26-28.
- [104] H. Ritter, T. Martinetz, K. Schulten (1992) *Neural Computation and Self-Organizing Maps: An Introduction*, Addison-Wesley, Reading, MA.
- [105] J. Rockel, R. W. Horns and T. I. S. Powell (1980) "The basic uniformity in structure of the neocortex", *Brain*, vol. 103, pp. 221-244.
- [106] P.J. Rousseeuw and D.K. van Driessen (1999) "A fast algorithm for the minimum covariance determinant estimator", *Technometrics*, vol. 41, pp. 212-223.
- [107] S.T. Roweis (1998) "EM algorithms for PCA and SPCA", in *Proc. Advances in Neural Information processing Systems, NIPS-98*, 10:452-456.

- [108] H. Sakai and K. Shimizu (1997) "Two improved algorithms for adaptive subspace filtering", in *Proc. 11th IFAC Symposium System Identification (SYSID '97)*, pp. 1689-1694, Kitakyushu, Japan.
- [109] T.D. Sanger (1989) "Optimal unsupervised learning in a single-layer linear feedforward neural network", *Neural Networks*, vol. 2, no. 6, pp. 459-473.
- [110] E. Saund (1995) "A multiple cause mixture model for unsupervised learning", *Neural Computation*, vol. 7, pp. 51-71.
- [111] R. Schmidt (1986) "Multiple emitter location and signal parameter estimation", *IEEE Trans. On Antennas and Propagation*, vol. 34, pp. 276-280.
- [112] T.J. Sejnowski and C. R. Rosenberg (1987) "Parallel networks that learn to pronounce English text", *Complex Systems*, L 145-168.
- [113] J. Sherry, D. L. Barrow and W. R. Klemm (1982). "Brain", *Research Bulletin*, S, 163-169.
- [114] M. Sur, I. E. Garraghty and A. W. Roe (1988) "Experimentally induced visual projections into auditory thalamus and cortex", *Science*, vol. 242, pp. 1437-1441.
- [115] M.E. Tipping, and C.M. Bishop (1999) "Probabilistic principal component analysis", *Journal of the Royal Statistical Societe*, Series B, vol. 3, pp. 600-616.
- [116] M. Turk and A. Pentland (1991) "Eigenfaces for recognition", *Journal of Cognitive Neuroscience*, vol. 3, pp. 71-86.
- [117] M. VanHulle (2002) *Faithful Representations and Topographic Maps*, Wiley, New York.
- [118] J. Wallace, R. Dickinson (1972) "Empirical orthogonal representation of time series in the frequency domain. Part I: Theoretical considerations", *Journal of Applied Meteorology*, vol. 11, no. 6, pp. 887-892.
- [119] L. Wang, and J. Karhunen (1996) "A unified neural bigradient algorithm for robust PCA and MCA", *Int. J. of Neural Systems*, vol. 7, no. 1, pp. 53-67.
- [120] M. Wax and T. Kailath (1985) "Detection of signals by information theoretic criteria", *IEEE Trans. Acoustic, Speech, Signal Processing*, 33:387-392.
- [121] A Weingessels and K. Hornik (2000) "Local PCA algorithms", *IEEE Trans. Neural Networks*, vol. 11, no. 6, pp. 1242-1250.
- [122] L. Wiscott (1998), "Learning invariance manifolds", in *Proc. International Conference on Artificial Neural Networks (ICANN)*, pp. 555-560.

- [123] L. Xu (1993) "Least mean square error reconstruction principle for self-organizing neural nets", *Neural Networks*, vol. 6, pp. 627-648, 1993.
- [124] L. Xu, E. Oja, C.Y. Suen (1992) "Modified Hebbian learning for curve and surface fitting", *Neural Networks*, vol. 5, pp. 441-457.
- [125] L. Xu and A. Yuille (1993) "Self organizing rules for robust principal component analysis", in Proc. *Advances in Neural Information Processing Systems*, NIPS 5, pp. 467-474.
- [126] W. Yan, U. Helmke and J.B. Moore (1994) "Global Analysis of Oja's flow for Neural Networks", *IEEE Trans. on Neural Networks*, vol. 5, no. 5, pp. 674-683.
- [127] M.P. Young, K. Tanaka and S. Yamane (1991) "On oscillating neuronal responses in the visual cortex of the monkey" In University Laboratory of Physiology, Oxford, manuscript.
- [128] A.L. Yuille, D.M. Kammen and D.S. Cohen (1989) "Quadrature and development of orientation selective cortical cells by Hebb rules", *Biological Cybernetics*, vol. 61, pp. 183-194.
- [129] Q. Zhang and Y. Leung (2000) "A class of learning algorithms for principal component analysis and minor component analysis", *IEEE Trans. on Neural Networks*, vol. 11, no. 2, pp. 529-533.



Прилог 1.

Изјава о ауторству

Потписани-а Јанковић, Марко В.

Изјављујем

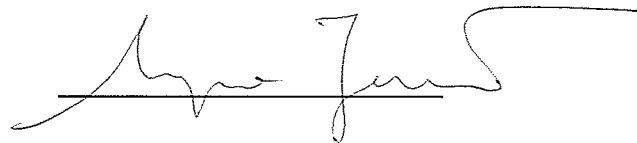
да је докторска дисертација под насловом

Самоорганизујуће неуралне мреже за анализу главних компонената

- резултат сопственог истраживачког рада,
- да предложена дисертација у целини ни у деловима није била предложена за добијање било које дипломе према студијским програмима других високошколских установа,
- да су резултати коректно наведени и
- да нисам кршио/ла ауторска права и користио интелектуалну својину других лица.

Потпис

У Београду, 11.02.2014.



Прилог 2.

Изјава о коришћењу

Овлашћујем Универзитетску библиотеку „Светозар Марковић“ да у Дигитални репозиторијум Универзитета у Београду унесе моју докторску дисертацију под насловом:

Самоорганизујуће неуралне мреже за анализу главних компонената

која је моје ауторско дело.

Дисертацију са свим прилозима предао/ла сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигитални репозиторијум Универзитета у Београду могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (*Creative Commons*) за коју сам се одлучио/ла.

1. Ауторство

2. Ауторство - некомерцијално

3. Ауторство – некомерцијално – без прераде

4. Ауторство – некомерцијално – делити под истим условима


5. Ауторство – без прераде

6. Ауторство – делити под истим условима

(Молимо да заокружите само једну од шест понуђених лиценци, кратак опис лиценци дат је на полеђини листа).

Потпис

У Београду, 11.02.2014.



1. Ауторство - Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце, чак и у комерцијалне сврхе. Ово је најслободнија од свих лиценци.
2. Ауторство – некомерцијално. Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела.
3. Ауторство - некомерцијално – без прераде. Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца не дозвољава комерцијалну употребу дела. У односу на све остале лиценце, овом лиценцом се ограничава највећи обим права коришћења дела.
4. Ауторство - некомерцијално – делити под истим условима. Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца не дозвољава комерцијалну употребу дела и прерада.
5. Ауторство – без прераде. Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, без промена, преобликовања или употребе дела у свом делу, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце. Ова лиценца дозвољава комерцијалну употребу дела.
6. Ауторство - делити под истим условима. Дозвољаваате умножавање, дистрибуцију и јавно саопштавање дела, и прераде, ако се наведе име аутора на начин одређен од стране аутора или даваоца лиценце и ако се прерада дистрибуира под истом или сличном лиценцом. Ова лиценца дозвољава комерцијалну употребу дела и прерада. Слична је софтверским лиценцама, односно лиценцама отвореног кода.