Matematički Fakultet
Univerzitet u Beogradu

Statistički softver 3

Seminarski rad

Student:
Ana Nikolić 45/2015

Asistent:
Danijel Subotić

1

Ugao rotacije pravougaonika u odnosu na x-osu se može odrediti tako što se pronađu dva susedna temena i nađe ugao pod kojim prava određena tim temenima seče x-osu. Za taj ugao se slika rotira i čuva u folderu u kome je i zadatak1.R

Temena se pronalaze tako što se posmatraju pikseli slike redom i prvi na koji se naiđe a da je različit od pozadine se čuva kao teme. To je mouće samo uz pretpostavku da je pozadina homogena. Jedno teme se može pronaći tako što se pretražuje redom po vrstama, a drugo po kolonama. Ta dva temena bi se poklopila samo u slučaju da je pravougaonik već paralelan sa x-osom i u tom slučaju je ugao 0.

Na nekim dobijenim rotiranim slikama, ukoliko je slika male rezolucije (manje od 100x100), pravougaonici ne deluju baš skroz paralelno sa x-osom, ali ako bi se greška od 2-3 piksela tolerisala onda su rezultati korektni :D

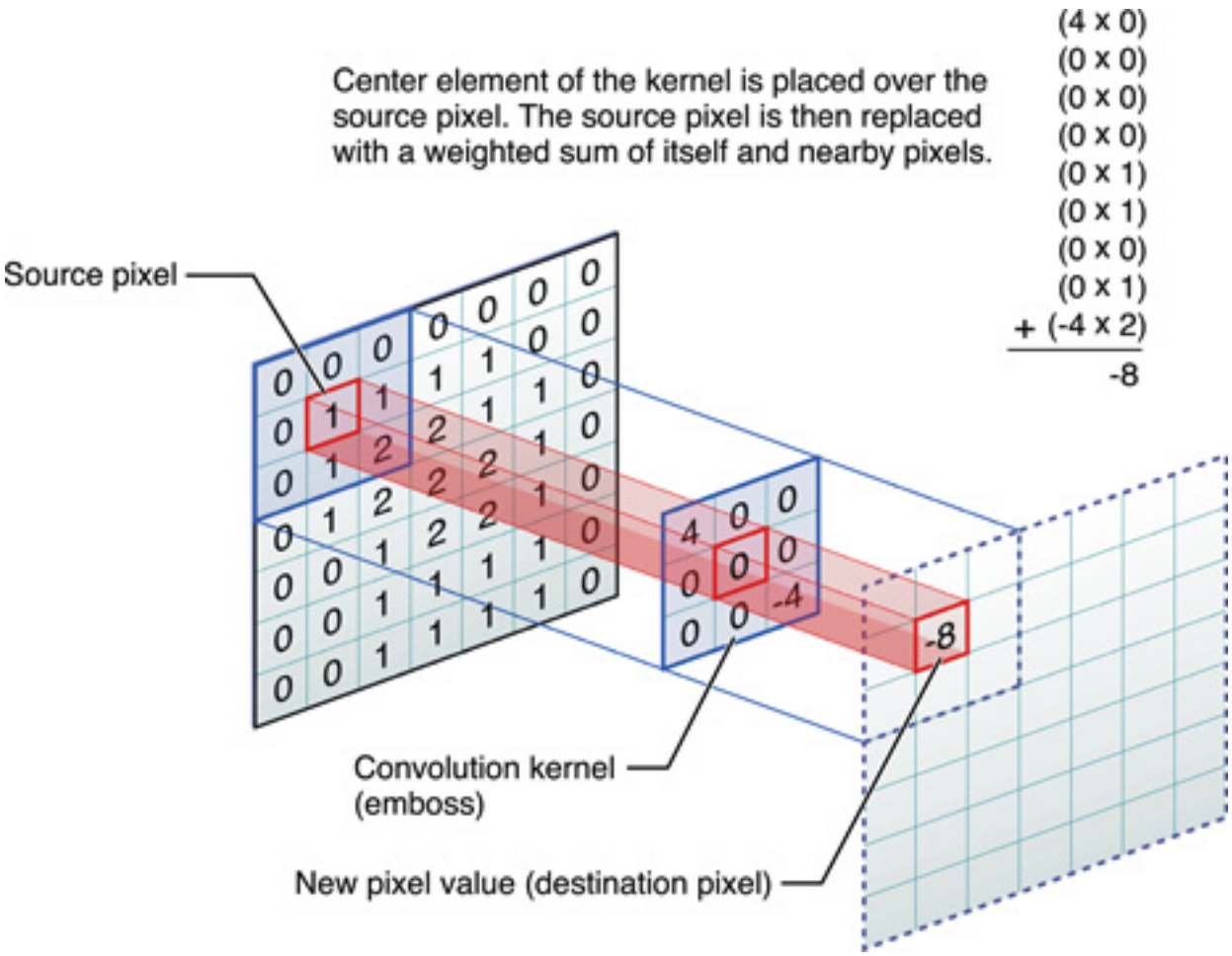
2

Zbog jednostavnosti učitavanja slika u R, slike su podeljene u foldere pravougaonici i krugovi.

Kao i malo pre, u slučaju da radimo sa pravougaonicima, potrebno je pronaći njihova temena. To bi bilo dosta prostije da, kao u prethodnom zadatku, važi pretpostavka o homogenosti pozadine, ali jedna od slika u primerima ima pozadinu koja se neprekidno menja od crne do bele. To je možda moglo da se reši tako što se porede svaka dva susedna piksela i posmatra se razlika u boji između njih. Ako je ona veća od nekog zadatog praga, koji bi se mogao odrediti testiranjem, onda smo naišli na ivicu. Međutim, ta ideja mi je pala na pamet kasno, kad je zadatak bio već skoro gotov na komplikovaniji ali zanimljiviji način :)

Sliku sa homogenom pozadinom je moguće dobiti ako se izdvoje ivice objekta na slici. Sve što nisu ivice će biti crno dok će samo ivice biti prikazane nekom bojom. Odatle je moguće naći koordinate temena na isti način kao u pretodnom zadatku. Jedan od načina za detekciju ivica je konvolucija!

Uzmimo matricu neparne dimanzije, u ovom slučaju 3x3, koja predstavlja filter. Taj filter se kreće po matrici, koja predstavlja sliku koju obrađujemo. Nad svakom mogućom 3x3 podmatricom primenimo sledeću operaciju: Elementi filtera se množe sa elementima te podmatrice redom (prvi sa prvim drugi sa drugim itd.), onda svi tako dobijeni brojevi saberu i dobije se jedna vrednost koja se upisuje u novu matricu na odgovarajuće mesto. Slikovito objašnjeno ispod:



Za različite filtere se postižu različiti rezultati. Izoštravanje slike, zamućivanje, detekcija ivica i razne druge stvari se mogu postići primenom odgovarajućih filtera.

Za ovaj konkretan zadatak su se mnogo lepši rezultati dobili tako što se prvo izdvoje posebno vertikalne ivice, a posebno horizontalne ivice, pa se tražena slika dobije kombinacijom prethodne dve. Filter za detekciju vertikalnih ivica je najjednostavniji, jedinice levo, minus jedinice desno i nule u sredini, dok je transponat toga korišćen za detekciju horizontalnih.

3

Ako se dve normirane slike oduzmu, trebalo bi da pikseli čija je vrednost različita od nule budu mesta na kojima se te dve slike razlikuju. Međutim, može da se desi da se provuku i neke tačke koje ne bi trebalo da budu različite. Zato se traže one tačke u kojima je razlika veća od 10 jer se boja, na mestima na kojima su dve slike iste, svakako razlikuje za manje od 10.

4

Da bi se odredilo koji je broj u potpolju prvo je potrebno izdvojiti to potpolje zbog čega je prvi korak odrediti ivice. Pošto su slike međusobno slične, dovoljno je gledati kako se ponaša samo prva slika. Posmatranjem matričnog zapisa slike se može uočiti da su vrednosti matrice koje odgovaraju pikselima koji čine granicu manje od 0,1. Želimo intervale koji predstavljaju poziciju granica na slici, ali tako da oni budu što širi, da bi kad se iseče sve što nije ivica, ostalo polje u koje ne upadaju delovi ivica, ali opet ne preširoki jer zapravo treba izdvojiti polje. Čini se razumnim pretpostavka da ako je na jednoj vertikali preko 40% piksela crno (vrednost manja od 0,1) da se ta vertikala može smatrati delom ivice polja.

Kada se izdvoje polja, za njih se može izračunati zastupljenost plave boje i disperzija crno bele varijante polja. To se može uzeti za prediktore pomoću kojih se klasifikuju polja, tj. pomoću kojih se određuje da li je polje otvoreno i ako jeste, da li je neki broj u polju, i koji, ili mina. Za klasifikaciju

polja iskorišćene su Multinomna regresija, Linearna diskriminatorna analiza i Kvadratna diskriminatorna analiza.

Multinomna regresija je generalizacija Logističke regresije na problem klasifikacije u više od dve klase.

LDA za računanje verovatnoće pripadanja svakoj klasi koristi Bajesovu formulu pa je potrebno oceniti apriorne verovatnoće pripadanja klasama i uslovnu funkciju gustine, pod uslovom da zavisna promenljiva pripada određenoj klasi. Apriorna raspodela se određuje empirijski, iz uzorka, dok je za ocenu druge funkcije potrebno pretpostaviti da za svaku klasu j, uslvna gustina ima normalnu raspodelu sa očekivanjem μ_j i disperzijom σ^2 koja je ista za sve klase. U slučaju klasifikacije za više od dve klase, očekivanje je vektor koji može biti različit za sve klase, dok je kovarijaciona matrica uvek ista.

KDA je dosta slična kao LDA, osim jedna bitne razlike, a to je da sada više disperzija ne mora biti konstantna po klasama.

Korišćenjem ove tri metode samo na prvoj slici iz skupa za trening dobije se da sve tri imaju poteškoća sa identifikacijom broja 3. Pošto je broj tri crven, to je bila motivacija da se doda još jedan prediktor 'crveno', koji radi isto što i 'plavo' samo za crvenu boju. Korišćenjem sva tri prediktora, sve tri metode uspeju bez greške da klasifikuju sva polja iz kontrolnog skupa, dok samo sa prva dva prediktora Multinomna regresija i QDA rade odlično, dok LDA radi nešto slabije ali i dalje dosta dobro (sa 93,8% uspeha).

5

Još jedna metoda koja postoji je klasifikacija pomoću stabla odlučivanja. Prvi korak u primeni te metode je podela prostora prediktora na regione. Prostor se deli redom, na višedimenzione pravougaonike tako da greška klasifikacije bude najmanja. U svakom koraku se neka od oblasti nastala u prethodnom koraku podeli na dva dela. Te podele se mogu lepo predstaviti stablom zbog čega se metoda tako i zove. Podele se prekidaju kada se dodje do nekog od kriterijuma zaustavljanja. Međutim, često klasifikacija pomoću ovako dobijenog stabla ne daje dobre rezultate na test setu zbog

preprilagođenosti podacima koji su korišćeni za treniranje. Zbog toga se prvo formira jedno veliko stablo, koje se kasnije skraćuje da se dobije podstablo koje bi dalo najbolje moguće rezultate. Ovaj postupak se naziva 'pruning' i ne bih se upuštala u pokušaj prevoda tog izraza. Cilj je naći takvo podstablo koje bi dovelo do najniže greške pri testiranju. Kako je testiranje na svakom mogućem podstablu komplikovano, treba izdvojiti samo podskup podstabala koje će se uzeti u obzir. Posmatramo samo podskup podstabala indeksiran parametrom podešavanja α. Za svako α postoji podstablo takvo da je

$$\sum_{m=1}^{|T|} \sum_{x_i \in R_m} (y_i - \hat{y}_{Rm})^2 + \alpha |T|$$

najmanje moguće. Parametar α u ovom izrazu pravi kompromis između prilagođenosti podacima za treniranje i kompleksnosti stabla. Izbor parametra α se može dobti krosvalidacijom, i pomoću njega se odredi potreban podskup.
Kada je prostor prediktora podeljen u regione, klasifikacija se vrši tako što se svakoj opservaciji iz skupa za testiranje dodeli ona klasa koja je najčešća među podacima za treniranje u regionu kome pripada data opservacija.

Ova metoda se može primeniti na bazi Carseats ako se napravi zavisna promenljiva High koja je indikator da li je vrednost promenljive Sales veća od 8. Korišćenjem ove metode prvo pomocu celog stabla a zatim pomocu podstabla utvrđuje se da procenat uspešne klasifikacije sa 70.5% poraste na 75%.

Ova metoda se može primeniti i na prethodni zadatak. U tom slučaju se klasifikuje u više od dve klase. To je moguće primenom funkcije rpart iz paketa rpart. Pomoću istih prediktora kao u prošlom zadatku dobije se opet predviđanje sa 100% uspeha.

6

Klasterizacija je metoda nevođenog učenja koja grupiše međusobno slične podatke, a razdvaja one različite. Stoga bi mogla da se primeni u razdvajanju polja tabele minolovca. Ako je u pitanju $K - means$ klasterizacija, rezultat

je K centroida (za K klastera) dobijenih na sledeći način:

$$C = arg \min_{c_1,\ldots,c_K} \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} ||X_i - c_k||^2 I\{c_i \text{ najbliže } X_i\}.$$

Pošto imamo 7 različitih tipova polja, razdvajaćemo podatke u 7 klastera pomoću prediktora iz četvrtog zadatka.
Ova metoda ume prazno polje i broj 4 lepo da prepozna, dok ostale brojeve smešta u više od jednog klastera.

Da bi se rezultati mogli grafički predstaviti potrebno je redukoviti broj dimenzija, što je učinjeno metodom t-SNE (t-Distributed Stochastic Neighbor Embedding).

7

Da bi tabla minolovca bila ispravna mora da zadovoljava nekoliko uslova:
1. Broj mina na tabli odgovara zadatom broju mina
2. Ne sme postojiati mina pored praznog polja
3. Oko svakog polja koje je neki broj, mora postojati baš taj broj mina.
Ako su ovi uslovi ispunjeni, funkcija *prava_matrica* vraća 1, u suprotnom 0.

Da bi se generisala ispravna tabla minolovca potrebno je prvo odrediti gde će se nalaziti mine. One se raspoređuju proizvoljno. Tabla je na početku popunjena nulama, zatim se za polja u kojima je mina, za svako polje u okolini mine dodaje 10 na već postojeći broj. Time se postiže da svaki broj upisan u tabelu predstavlja broj okolina mina u kojima je to polje sadržano, samim tim i broj mina u okolini tog polja. Na kraju se u polja koja su ostala 0, tj. ona na koje mine ne utiču, upisuje 100 i predstavljaju otvorena polja.

skrivanje_polja radi tako što prvo zatvori sva polja u kojima su mine, a zatim od preostalih zatvori jos onliko koliko je traženo nasumičnih polja.

Funkcija *popuni* je pomoćna funkcija koja delimično popunjenu tablu popunjava do kraja. Ako je popunjena tabla validna, vraća pozicije mina u novoj tabli kojih nije bilo u prvobitnoj. Te pozicije se pamte pri svakoj od N iteracija funkcije $MK\_simulacija$. Mesto na kome je najviše puta bila mina je pozicija sa najvećom verovatnoćom da je sadrži.

8

Optimalnu strategiju igre Determinanta je moguće simulirati Monte Karlo simulacijom. Prvi igrač treba da izabere broj i poziciju u matrici gde će da upiše taj broj. To radi tako što se za svaku od N permutacija table broji koji igrač je pobedio. Na taj način se prati za svaki broj i za svaku poziciju na kojoj se našao taj broj, koliko je puta svaki od igrača pobedio za tako odigran potez. Rezultati se upisuju u matricu n^2 x n^2 jer ima n^2 brojeva i svaki od njih je moguće rasporediti na n^2 pozicija. Kada prvi igrač odigra, drugi bira jedan od preostalih brojeva i smešta ga na jednu od preostalih pozicija na isti način kao i prvi igrač u svom potezu, sem što on sad pravi permutacije samo od preostalih neupotrebljenih brojeva.