

Upravljanje omejenega maloprodajnega prostora
za osnovne izdelke
Kratko poročilo

Peter Dolenc, Jan Rems

2017

1 Uvod

V nalogi se bova soočila z naslednjim problemom. Zamislimo si, da imamo v ravnini neko *konveksno množico* C , ki vsebuje *konveksno podmnožico* C' . Najina naloga bo določiti verjetnost dogodka, da naključna premica, ki seka množico C seka tudi podmnožico C' . Gre za klasični problem s področja stohastične geometrije, ki se ga da posplošiti tudi na druge veje verjetnostne teorije. Preko klasičnega rezultata integralske geometrije pričakujemo, da bo verjetnost zgoraj omenjenega dogodka enaka razmerju obsegov konveksnih množic. Najina glavna naloga bo, izvedba eksperimenta, kjer se bova z generiranjem naključnih premic in konveksnih množic, poizkusila čim bolj približati željenemu rezultatu. To nameravamo doseči na eni strani z razumevanjem teoretičnega ozadja problema, na drugi pa z izvedbo čim natančnejšega eksperimenta.

2 Analitični pristop k problemu

Kot bomo videli, se izkaže, da je iskanje preseka premice s konveksno množico, smiselno prevesti na problem iskanja presečišč premice in sklenjene regularne krivulje, ki omejuje dano konveksno množico. Tako nas namesto razmerja obsegov, zanima razmerje dolžine krivulj, kjer ena omejuje konveksno množico C , druga pa njeno konveksno podmnožico C' . Tudi krivulji bomo ustrezno poimenovali γ in γ' . Krivuljam, ki omejujejo konveksne množice pravimo tudi konveksne krivulje. Za začetek, pa moramo definirati množico premic v ravnini.

Definicija 2.1 *Množico vseh neorientiranih premic na \mathbb{R}^2 definiramo kot:*

$$\{(p, \theta) | 0 \leq \theta \leq 2\pi, p \geq 0\},$$

kjer p predstavlja oddaljenost premice od izhodišča, θ pa kot premice glede na x os.

Sedaj ko imamo definirano množico vseh premic na \mathbb{R}^2 , lahko na tej množici podamo tudi mero s sledečim predpisom:

$$\int \int d\theta dp$$

Izrek 1 (Cauchy-Croftonova formula) *Naj bo c regularna ravninska krivulja in $n_c(p, \theta)$ število točk pri katerih premica parametrizirana na način, kot je podan v definiciji seka krivuljo c . Potem za dolžino krivulje $L(c)$ veja:*

$$L(c) = \frac{1}{2} \int \int n_c(p, \theta) d\theta dp$$

Preko sklicevanja na zgornji izrek, bova torej v teoretičnem delu naloge pokazala, da je verjetnost dogodka

3 Načrt dela

Projekta se bova lotila programersko in sicer z metodami Monte Carlo. To so simulacijske metode, ki s pomočjo naključnih števil in velikega števila izračunov in ponavljanja omogočajo predvidevanje obnašanja zapletenih matemtičnih sistemov. Ideja je preprosta: generirati dve naključni konveksni množici in pa naključno premico, ki seka večjo od obeh ter ugotoviti, ali premica seka tudi manjšo. Če tak poskus ponovimo velikokrat, bomo lahko ocenili verjetnost, da naključna premica, ki seka večjo konveksno množico, seka tudi manjšo.

Na začetku bova poskušala preveriti najino hipotezo na bolj enostavnih konveksnih množicah - elipsah, kasneje pa se bova spopadla tudi z bolj splošnimi konveksnimi množicami. Pri delu z elipsami je cilj sprogramirati program, ki bo sposoben obvladovanja osnovnih prijemov: generiranje množic in premic in pa prepoznavanja, kdaj te premice res sekajo naše objekte. Ta orodja nama bodo pomagala kasneje, ko bova iz enostavnih elips želela preiti na bolj splošne in zapletene konveksne množice.

Drugi del izziva bo predstavljala generiranje naključnih premic. Za to obstaja nekaj različnih enakovrednih načinov. Eden takšnih je, da na x osi izberemo naključno točko skozi katero bo premica potekala in pa kot med 0 in 180° , ki predstavlja naklon premice. Težava takega generiranja premic je, da premice morda ne bodo sekale niti večjega od naših likov. Zato bova poskusila tudi tako, da v večji od najinih množic naključno generirava eno točko, drugo pa generirava poljubno. Taki dve točki enolično določita premico, ki gre skozi njiju. Končno bova premice generirala tako, kot je parametrizirano v definiciji, z določitvijo parametrov p in θ , kjer p predstavlja razdaljo premice do izhodišča, θ pa kot med premico in x osjo.

Delo bo potekalo v programskem okolju Matlab.