Iterativno traženje motiva i vreća fraza u genomu i proteomu

Čavka, Anamarija

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:217:357042

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Anamarija Čavka

ITERATIVNO TRAŽENJE MOTIVA I VREĆA FRAZA U GENOMU I PROTEOMU

Diplomski rad

Voditelj rada:
doc. dr. sc. Pavle Goldstein

Zagreb, studeni, 2019.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Zahvaljujem mentoru doc.dr.sc. Palvi Goldsteinu na savjetima, strpljenju i neopisivom entuzijazmu te dr.sc. Braslavu Rabaru na osnovnim kodovima i pojašnjenjima. Hvala mojim roditeljima, sestrama, bratu i teti Dragici na svoj podršci koju su mi pružili tijekom studiranja, a posebice pisanja ovog rada.

Posvećeno mojoj baki Anici.

Sadržaj

Uvod

U današnje vrijeme postoji velika potreba za računalnim analizama različitih nizova znakova te njihovoj identifikaciji s obzirom na pripadnost nekim klasama. U prirodnom jeziku tako će se pokušavati klasificirati tekstove koji pripadaju nekom području, a u bioinformatici pak će se pokušati razvrstati proteinske nizove po proteinskim familijama ili identificirati kodirajuće dijelove DNA zadužene za određenu proteinsku familiju. Time dolazimo do teme i cilja ovog rada.

Cilj ovog diplomskog rada je testirati algoritam iterativnog pretraživanja teksta u svrhu pronalaska ključnih fraza specifičnih za neku tekstualnu familiju. Testiranje provodimo na genomu te ga uspoređujemo s rezultatima sličnog algoritma na ekvivalentnom problemu na proteomu. Time pokušavamo ustvrditi kako se algoritam ponaša na različitim tipovima teksta.

Ovaj diplomski rad podijeljen je u 5 poglavlja. U prvom poglavlju ukratko navodimo pojmove te rezultate iz vjerojatnosti i statistike koje ćemo kasnije koristiti u radu. Drugo poglavlje se bavi definicijama osnovnih pojmova u smislu u kojem se one koriste u radu. Treće poglavlje pobliže objašnjava biološku terminologiju te postupke pripreme podataka za provođenje centralnog algoritma kojim se bavimo u četvrtom poglavlju. U četvrtom poglavlju također navodimo rezultate primjene algoritma na konkretnom primjeru. U petom poglavlju dan je kratak osvrt na glavne zaključke rada te usporedba s rezultatima sličnog algoritma provedenog na ekvivalentnom problemu.

Poglavlje 1

Pojmovi iz vjerojatnosti i statistike

1.1 Vjerojatnost

U ovom odjeljku navodimo neke osnovne definicije i rezultate teorije vjerojatnosti preuzete iz [7].

Definicija 1.1.1. *Pod **slučajnim pokusom** podrazumijevamo takav pokus čiji **ishodi**, odnosno **rezultati** nisu jednoznačno određeni uvjetima u kojima izvodimo pokus. Rezultate slučajnog pokusa nazivamo **događajima**.*

Definicija 1.1.2. *Neka je A događaj vezan uz neki slučajni pokus. Pretpostavimo da smo taj pokus ponovili n puta i da se u tih n ponavljanja događaj A pojavio točno n_A puta. Tada broj n_A zovemo **frekvencija** događaja A, a broj $\frac{n_A}{n}$ **relativna frekvencija** događaja A.*

Definicija 1.1.3. *Osnovni objekt u teoriji vjerojatnosti je neprazan skup Ω koji zovemo **prostor elementarnih događaja** i koji reprezentira skup svih ishoda slučajnih pokusa. Točke ω iz skupa Ω zvat ćemo **elementarni događaji**.*

Definicija 1.1.4. *Familija $\mathcal{F}$ podskupova od Ω ($\mathcal{F} \subset \mathcal{P}(\Omega)$) jest **$\sigma$-algebra skupova** (na Ω) ako je*

- $\emptyset \in \mathcal{F}$
- $A \in \mathcal{F} \implies A^c \in \mathcal{F}$
- $A_i \in \mathcal{F}, i \in \mathbb{N} \implies \bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$

Definicija 1.1.5. *Neka je $\mathcal{F}$ σ-algebra na Ω. Uređen par $(\Omega, \mathcal{F})$ se zove **izmjeriv prostor**.*

Definicija 1.1.6. *Neka je* $(\Omega, \mathcal{F})$ *izmjeriv prostor. Funkcija* $\mathbb{P} : \mathcal{F} \to \mathbb{R}$ *jest* ***vjerojatnost*** *ako vrijedi*

- $\mathbb{P}(\Omega) = 1$
- $\mathbb{P}(A) \geq 0, A \in \mathcal{F}$
- $A_i \in \mathcal{F}, i \in \mathbb{N}, A_i \cap A_j, i \neq j \implies \mathbb{P}\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mathbb{P}(A_i)$

Uređena trojka $(\Omega, \mathcal{F}, \mathbb{P})$ *gdje je* $\mathcal{F}$ σ*-algebra na* Ω *i* $\mathbb{P}$ *vjerojatnost na* $\mathcal{F}$ *zove se* ***vjerojatnosni prostor****.*
Neka je $(\Omega, \mathcal{F}, \mathbb{P})$ *vjerojatnosni prostor. Elemente* σ*-algebre zovemo* ***događaji****, a broj* $\mathbb{P}(A), A \in \mathcal{F}$ *se zove* ***vjerojatnost događaja*** *A.*

Definicija 1.1.7. *Neka je* $(\Omega, \mathcal{F}, \mathbb{P})$ *proizvoljan vjerojatnosni prostor i* $A \in \mathcal{F}$ *takav da je* $\mathbb{P}(A) > 0$*. Definiramo funkciju* $\mathbb{P}_A : \mathcal{F} \to [0, 1]$ *s*

$$\mathbb{P}_A(B) = \mathbb{P}(B|A) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(A)}, B \in \mathcal{F}.$$

Lako je provjeriti da je $\mathbb{P}_A$ *vjerojatnost na* $\mathcal{F}$ *i nju zovemo* ***vjerojatnost od*** *B* ***uz uvjet*** *A.*

Definicija 1.1.8. *Neka je* $(\Omega, \mathcal{F}, \mathbb{P})$ *proizvoljan vjerojatnosni prostor i* $A_i \in \mathcal{F}, i \in I$ *proizvoljna familija događaja. Kažemo da je to* ***familija nezavisnih događaja*** *ako za svaki konačan podskup različitih indeksa* $i_1, i_2, ..., i_k$ *vrijedi*

$$\mathbb{P}\left(\bigcap_{j=i}^{k} A_{i_j}\right) = \prod_{j=i}^{k} \mathbb{P}(A_{i_j}).$$

Označimo s $\mathcal{B}$ σ-algebru generiranu familijom svih otvorenih skupova na skupu realnih brojeva $\mathbb{R}$. $\mathcal{B}$ zovemo **Borelova σ-algebra skupova na** $\mathbb{R}$, a elemente σ-algebre $\mathcal{B}$ zovemo **Borelovi skupovi**.

Definicija 1.1.9. *Neka je* $(\Omega, \mathcal{F}, \mathbb{P})$ *vjerojatnosni prostor. Funkcija* $X : \Omega \to \mathbb{R}$ *je* ***slučajna varijable*** *(na* Ω*) ako je* $X^{-1}(B) \in \mathcal{F}$ *za proizvoljno* $B \in \mathcal{B}$*, odnosno* $X^{-1}(\mathcal{B}) \subset \mathcal{F}$*.*

Definicija 1.1.10. *Neka je X slučajna varijabla na* Ω*.* ***Funkcija distribucije*** *od X je funkcija* $F_X = F : \mathbb{R} \to [0, 1]$ *definirana s*

$$F(x) = \mathbb{P}\{X \leq x\} = \mathbb{P}\{\omega : X(\omega) \leq x\}, x \in \mathbb{R}.$$

Definicija 1.1.11. *Slučajna varijabla X je* ***diskretna*** *ako postoji konačan ili prebrojiv skup* $D \subset \mathbb{R}$ *takav da je* $\mathbb{P}\{X \in D\} = 1$*.*

Definicija 1.1.12. *Neka je X slučajna varijabla na vjerojatnosnom prostoru $(\Omega, \mathcal{F}, \mathbb{P})$ i neka je F_X njezina funkcija distribucije. Kažemo da je X* ***apsolutno neprekidna*** *ili, kraće,* ***neprekidna slučajna varijabla*** *ako postoji nenegativna realna Borelova funkcija f na $\mathbb{R}$ takva da je*

$$F_X(x) = \int_{\infty}^{x} f(t)d\lambda(t), x \in \mathbb{R}. \tag{1.1}$$

Ako je X neprekidna slučajna varijabla, tada se funkcija f iz formule 1.1 zove **funkcija gustoće vjerojatnosti od** X, to jest od njezine funkcije distribucije F_X ili, kraće, **gustoća od** X.

Definicija 1.1.13. *Neka je X diskretna slučajna varijabla i neka je skup D iz definicije diskretne slučajne varijable, $D = \{x_1, x_2, ...\}$ i neka za svako k vrijedi $\mathbb{P}(\{x_k\}) = p_k$. Tada je* ***očekivanje diskretne slučajne varijable*** *X dano s*

$$\mathbb{E}X = \sum_k x_k p_k.$$

Definicija 1.1.14. *Neka je X neprekidna slučajna varijabla s funkcijom distribucije F_X.* ***Očekivanje neprekidne slučajne varijable*** *X dano je relacijom*

$$\mathbb{E}X = \int_{\Omega} X d\mathbb{P} = \int_{\mathbb{R}} x dF_X(x).$$

Neka je $g : \mathbb{R} \to \mathbb{R}$ Borelova funkcija. Vrijedi

$$\mathbb{E}[g(X)] = \int_{\Omega} g(X) d\mathbb{P} = \int_{\mathbb{R}} g(x) dF_X(x).$$

Definicija 1.1.15. *Neka $\mathbb{E}X$ postoji. Tada $\mathbb{E}[(X - \mathbb{E}(X))^r]$ zovemo* ***r-ti centralni moment od*** *X.*

Definicija 1.1.16. *Drugi centralni moment od X zovemo* ***varijanca od*** *X i označavamo je s $VarX$ ili σ_X^2. Odnosno,*

$$VarX = \mathbb{E}[(X - \mathbb{E}(X))^2].$$

Pozitivan drugi korijen iz varijance zovemo ***standardna devijacija od*** *X i označavamo je s σ_X*

Definicija 1.1.17. *Neka je $(X_n, n \in \mathbb{N})$ niz slučajnih varijabli definiran na vjerojatnosnom prostoru $(\Omega, \mathcal{F}, \mathbb{P})$. Kažemo da niz $(X_n, n \in \mathbb{N})$ slučajnih varijabli konvergira* ***gotovo sigurno (g.s.)*** *prema slučajnoj varijabli X ako je*

$$\mathbb{P}(\omega \in \Omega : X(\omega) = \lim_{n\to\infty} X_n(\omega)) = 1.$$

Konvergenciju označavamo s $(g.s.) \lim_{n\to\infty} X_n = X$ i takav je limes g.s. jedinstven.

Definicija 1.1.18. *Neka je* $(X_n, n \in \mathbb{N})$ *niz slučajnih varijabli definiran na vjerojatnosnom prostoru* $(\Omega, \mathcal{F}, \mathbb{P})$*. Kažemo da niz* $(X_n, n \in \mathbb{N})$ *slučajnih varijabli konvergira* ***po vjerojatnosti*** *prema slučajnoj varijabli* X *ako za svaki* $\epsilon > 0$ *vrijedi*

$$\lim_{n\to\infty} \mathbb{P}(|X_n - X| \geq \epsilon) = 0.$$

Konvergenciju označavamo s $(\mathbb{P}) \lim_{n\to\infty} X_n = X$ *i takav je limes g.s. jedinstven.*

Definicija 1.1.19. *Neka je* $(X_n, n \in \mathbb{N})$ *niz slučajnih varijabli definiran na vjerojatnosnom prostoru* $(\Omega, \mathcal{F}, \mathbb{P})$*. Kažemo da niz* $(X_n, n \in \mathbb{N})$ *slučajnih varijabli konvergira* ***po distribuciji*** *prema slučajnoj varijabli* X *ako je*

$$\lim_{n\to\infty} F_{X_n}(x) = F_X(x), x \in C(F_X)$$

gdje je $C(F_X)$ *skup svih točaka neprekidnosti funkcije* F_X*.*
Konvergenciju označavamo s $(\mathcal{D}) \lim_{n\to\infty} X_n = X$*.*

Među navedenim tipovima konvergencije vrijede implikacije

$$(g.s.) \lim_{n\to\infty} X_n = X \implies (\mathbb{P}) \lim_{n\to\infty} X_n = X$$

i

$$(\mathbb{P}) \lim_{n\to\infty} X_n = X \implies (\mathcal{D}) \lim_{n\to\infty} X_n = X.$$

Kako bi u radu lakše pokazali određene veze među nekim slučajnim varijablama, koristit ćemo neka od svojstava karakterističnih funkcija. Naime, metoda karakterističnih funkcija je jedno od osnovnih sredstava analitičkog aparata teorije vjerojatnosti ponajviše zbog činjenice da postoji $1-1$ korespondencija između skupa karakterističnih funkcija i skupa funkcija distribucije. Stoga, definiramo karakterističnu funkciju te navodimo teorem koji pokazuje postojanje $1-1$ korespondencije karakterističnih funkcija i funkcija distribucije.
Neka je F ograničena funkcija distribucije na $\mathbb{R}$.

Definicija 1.1.20. ***Karakteristična funkcija od*** F *jest funkcija* ρ *definirana s*

$$\rho(t) = \int_{-\infty}^{\infty} e^{itx} dF(x) = \int_{-\infty}^{\infty} cos(tx) dF(x) + i \int_{-\infty}^{\infty} sin(tx) dF(x), t \in \mathbb{R}.$$

Za svako $t \in \mathbb{R}$ funkcija $x \mapsto e^{itx}$ je neprekidna i budući da je $|e^{itx}| = 1$, ρ je dobro definirana, tj. imamo $\rho : \mathbb{R} \to \mathbb{C}$.

Definicija 1.1.21. *Neka je* X *slučajna varijabla s funkcijom distribucije* F_X*.* ***Karakteristična funkcija*** ρ_X ***od*** X *je karakteristična funkcija od* F_X*.*

Ako je X neprekidna slučajna varijabla s gustoćom f_X, tada vrijedi

$$\rho_X(t) = \int_{-\infty}^{\infty} e^{itx} f_X(x) d(x).$$

Propozicija 1.1.22. *(a) Ako je X slučajna varijabla i $a, b \in \mathbb{R}$, tada vrijedi*

$$\rho_{aX+b}(t) = e^{ibt}\rho_X(at), t \in \mathbb{R}. \tag{1.2}$$

(b) Ako su $X_1, X_2, ..., X_n, n \in \mathbb{N}$ nezavisne slučajne varijable, tada vrijedi

$$\rho_{\sum_{i=1}^{n} X_i}(t) = \prod_{i=1}^{n} \rho_{X_i}(t), t \in \mathbb{R}. \tag{1.3}$$

Teorem 1.1.23. *(Teorem jedinstvenosti) Neka su F_1 i F_2 funkcije distribucije na $\mathbb{R}$ i neka one imaju istu karakterističnu funkciju, tj. za sve $x \in \mathbb{R}$ vrijedi*

$$\int_{-\infty}^{\infty} e^{itx} dF_1(x) = \int_{-\infty}^{\infty} e^{itx} dF_2(x).$$

Tada je $F_1 = F_2$.

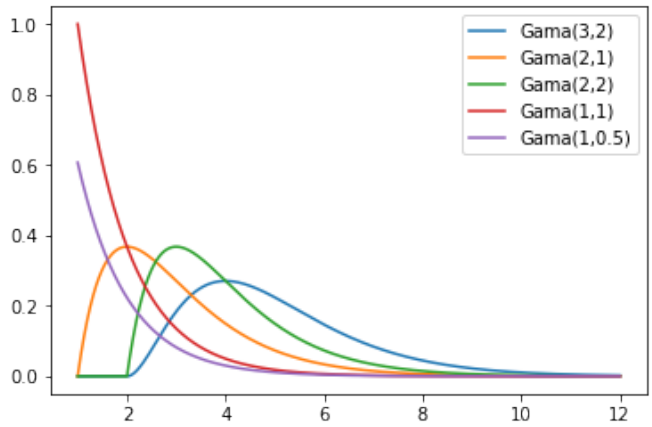
1.2 Primjeri slučajnih varijabli

U ovom odjeljku, navodimo neke primjere slučajnih varijabli koje će se direktno ili indirektno spominjati u ovom radu. Definicije i formule preuzimamo uglavnom iz [2] i [8] uz neke modifikacije.

Gama distribucija

Neka je $\alpha > 0, \beta > 0$ i $\Gamma(x) = \int_0^{\infty} e^{-t}t^{x-1}dt, x > 0$ gama funkcija. Neprekidna slučajna varijabla ima **gama distribuciju** s paramterima α i β ako joj je funkcija gustoće f dana s

$$f(x) = \begin{cases} \frac{1}{\Gamma(\alpha)\beta^{\alpha}} x^{\alpha-1} e^{\frac{-x}{\beta}}, x > 0 \\ 0, x \leq 0 \end{cases}. \tag{1.4}$$

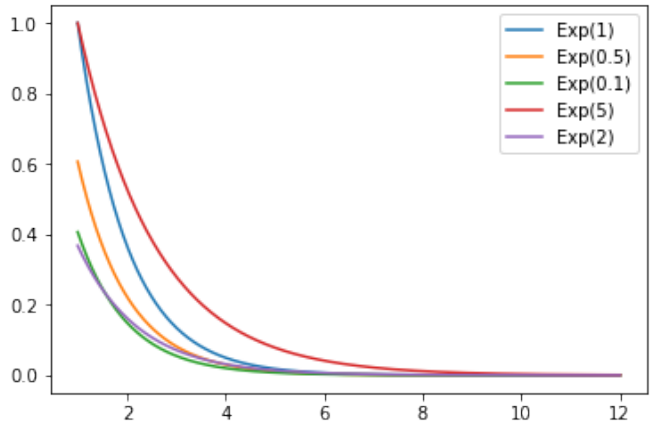


Slika 1.1: Funkcije gustoće gama distribucije s različitim parametrima

Eksponencijalna distribucija

Za neprekidnu slučajnu varijablu koja ima gama distribuciju s parametrima $\alpha = 1$ i $\beta = \frac{1}{\lambda}$ kažemo da ima **eksponencijalnu distribuciju** s parametrom λ. Funkcija gustoće eksponencijalne distribucije s parametrom λ je

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, x > 0 \\ 0, x \leq 0 \end{cases} . \tag{1.5}$$

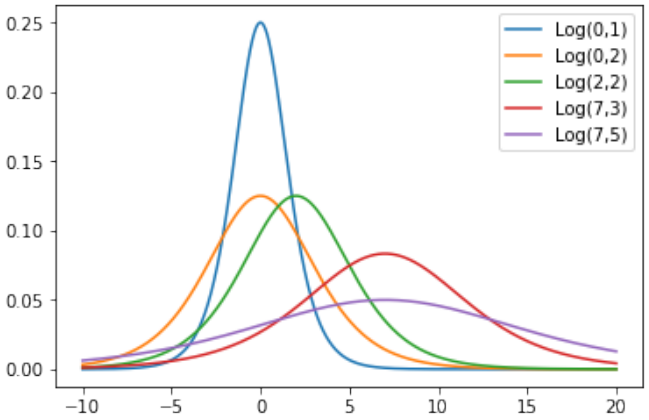


Slika 1.2: Funkcije gustoće eksponencijalne distribucije s različitim parametrima

Logistička distribucija

Neka je $\mu, \beta \in \mathbb{R}, \beta > 0$. Neprekidna slučajna varijabla X ima **logističku distribuciju** s parametrima μ i β ako joj je funkcija gustoće dana s

$$f(x) = \frac{e^{\frac{x-\mu}{\beta}}}{\beta(1+e^{\frac{x-\mu}{\beta}})^2}, x \in \mathbb{R}. \tag{1.6}$$



Slika 1.3: Funkcije gustoće logističke distribucije s različitim parametrima

Neka je $p, q > 0$. Slučajna varijabla X ima **generaliziranu logističku distribuciju** ako joj je funkcija gustoće dana s

$$f(x) = \frac{\Gamma(p+q)}{\Gamma(p)\Gamma(q)} \frac{e^{-qx}}{(1+e^{-x})^{p+q}}, x \in \mathbb{R}. \tag{1.7}$$

Budući da će nam biti bitno u daljnjem radu, navodimo i karakterističnu funkciju generalizirane logističke slučajne varijable X koja glasi

$$\rho_X(t) = \frac{\Gamma(p+it)\Gamma(q-it)}{\Gamma(p)\Gamma(q)}. \tag{1.8}$$

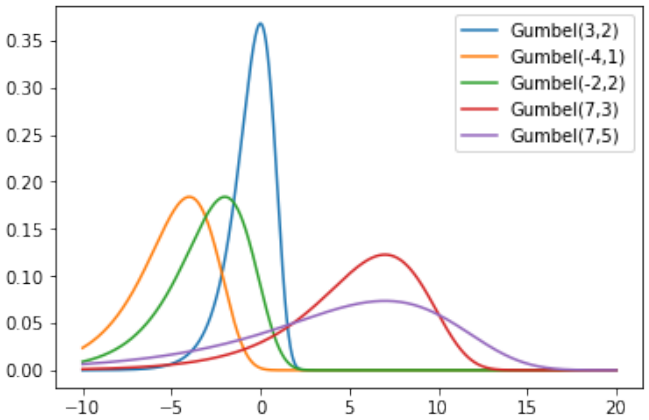
1.3 Teorija ekstremnih vrijednosti

U ovom odjeljku bavimo se pojmovima bitnima kod analize distribucija maksimalnih ocjena. Kao glavni izvor definicija i relacija koristimo [2]. Pritom su nam najvažniji rezultati korolar 1.3.1 i Fisher - Tippett - Gnedenkov teorem.

Gumbleova distribucija

Neka su $\mu \in \mathbb{R}$ i $\beta > 0$. Neprekidna slučajna varijabla X ima **Gumbelovu distribuciju** s parametrima μ i β ako joj je funkcija gustoće dana s

$$f(x) = \frac{1}{\beta} e^{-\frac{x-\mu}{\beta} - e^{-\frac{x-\mu}{\beta}}}, x \in \mathbb{R}. \tag{1.9}$$



Slika 1.4: Funkcije gustoće Gumbel distribucije s različitim parametrima

Neka je $p > 0$. Slučajna varijabla X ima **generaliziranu Gumbelovu distribuciju** ako joj je funkcija gustoće dana s

$$f(x) = \frac{1}{\Gamma(p)} e^{-px} e^{-e^{-x}}, x \in \mathbb{R}. \tag{1.10}$$

Pripadna karakteristična funkcija dana je s

$$\rho_X(t) = \frac{\Gamma(p - it)}{\Gamma(p)}, t \in \mathbb{R}. \tag{1.11}$$

Od posebnog interesa nam je veza između dvije nezavisne Gumbel distribuirane slučajne varijable budući da nam to garantira logističku distribuiranost naših maksimalnih ocjena u kasnijim poglavljima. Korolar i dokaz direktno preuzimamo iz [2], a navodimo ih u cijelosti radi potpunosti.

Korolar 1.3.1. *Neka su X_1 i X_2 nezavisne generalizirane Gumbel distribuirane slučajne varijable. Slučajna varijabla $Y = X_1 - X_2$ ima generaliziranu logističku distribuciju s parametrima p i q.*

Dokaz. Odgovarajuće karakteristične funkcije jednake su

$$\rho_{X_1}(t) = \frac{\Gamma(p - it)}{\Gamma(p)} \tag{1.12}$$

i

$$\rho_{X_2}(t) = \frac{\Gamma(q - it)}{\Gamma(q)}. \tag{1.13}$$

Koristeći relacije (1.2) i (1.3), vrijedi

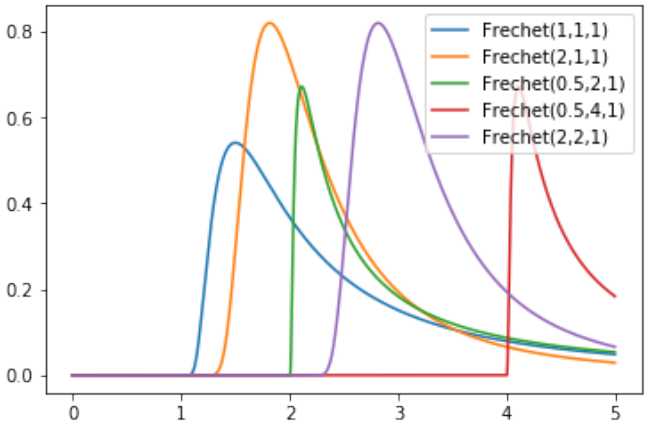
$$\rho_Y(t) = \rho_{X_1 - X_2}(t) = \rho_{X_1}(t)\rho_{X_2}(-t) = \frac{\Gamma(p - it)\Gamma(q + it)}{\Gamma(p)\Gamma(q)}. \tag{1.14}$$

Primijetimo da smo dobili karakterističnu funkciju generalizirane logističke slučajne varijable. Tvrdnja sada slijedi iz teorema jedinstvenosti. □

Fréchetova distribucija

Neka su $\alpha > 0, \beta > 0$ i $\mu \in \mathbb{R}$. Slučajna varijabla X ima **Fréchetovu distribuciju** ako joj je funkcija gustoće dana s

$$f(x) = \frac{\alpha}{\beta}\left(\frac{\beta}{x - \mu}\right)^{\alpha+1} e^{-(\frac{\beta}{x-\mu})^{\alpha}}, x \in \mathbb{R}. \tag{1.15}$$

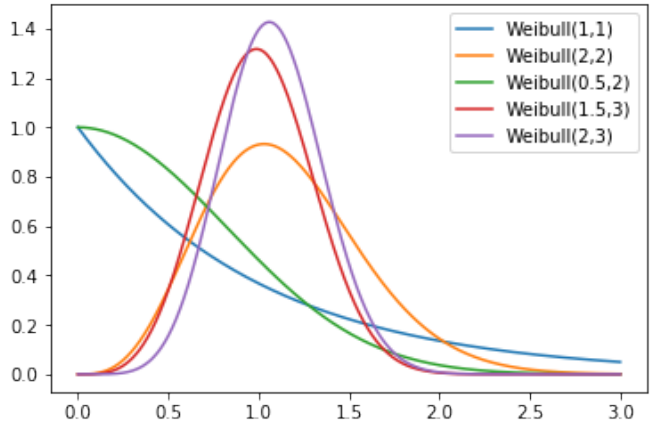


Slika 1.5: Funkcije gustoće Fréchetove distribucije s različitim parametrima

Weibullova distribucija

Neka su $\alpha > 0, \beta > 0$. Slučajna varijabla X ima **Weibullovu distribuciju** ako joj je funkcija gustoće dana s

$$f(x) = \begin{cases} \frac{\alpha}{\beta}(\frac{x}{\beta})^{\alpha-1} e^{-(\frac{x}{\beta})^{\alpha}}, x \geq 0 \\ 0, x < 0 \end{cases} . \tag{1.16}$$



Slika 1.6: Funkcije gustoće Weibullove distribucije s različitim parametrima

Fisher - Tippett - Gnedenkov teorem

Teorem navodimo bez dokaza.

Teorem 1.3.2. *(Fisher-Tippett, 1928; Gnedenko, 1943.) Neka su $X_1, X_2, ..., X_n$ jednako distribuirane slučajne varijable i neka je $M_n = max\{X_1, X_2, ..., X_n\}$. Ako postoji $a_n > 0$ i $b_n \in \mathbb{R}$ tako da*

$$\lim_{n\to\infty} \mathbb{P}\left(\frac{M_n - b_n}{a_n} \leq x\right) = F(x),$$

gdje je F nedegenerirana distribucija, tada granična distribucija F pripada Gumble, Fréchet ili Weibull distribuciji.

Gumble, Fréchet i Weibull distribucije možemo zapisati u generaliziranom obliku s funkcijom gustoće

$$f(x;\alpha,\beta,\mu) = \begin{cases} \frac{1}{\beta}(1+\alpha z)^{-1-\frac{1}{\alpha}} e^{-(1+\alpha z)^{-\frac{1}{\alpha}}}, \alpha \neq 0 \\ \frac{1}{\beta} e^{-z-e^{-z}}, \alpha = 0 \end{cases} \tag{1.17}$$

gdje je $z = \frac{x-\mu}{\beta}$ za $\alpha, \mu \in \mathbb{R}, \beta > 0$ parametre oblika, lokacije i mjere.

1.4 Specifičnost i osjetljivost

U ovom odjeljku definirat ćemo pojmove osjetljivosti, specifičnosti, pozitivne prediktivne vrijednosti i negativne prediktivne vrijednosti testa. Definicije su preuzete iz [4].

Osjetljivost testa definirana je s

$$\text{osjetljivost} = \frac{\text{broj stvarno pozitivnih}}{\text{broj stvarno pozitivnih + broj lažno negativnih}}. \tag{1.18}$$

Specifičnost testa definirana je s

$$\text{specifičnost} = \frac{\text{broj stvarno negativnih}}{\text{broj stvarno negativnih + broj lažno pozitivnih}}. \tag{1.19}$$

Pozitivna prediktivna vrijednost testa (PPV) definirana je s

$$\text{PPV} = \frac{\text{broj stvarno pozitivnih}}{\text{broj stvarno pozitivnih + broj lažno pozitivnih}}. \tag{1.20}$$

Negativno prediktivna vrijednost testa (NPV) definirana je s

$$\text{NPV} = \frac{\text{broj stvarno negativnih}}{\text{broj stvarno negativnih + broj lažnonegativnih}}. \tag{1.21}$$

Rezultati se često prikazuju tablicom, odnosno takozvanom matricom konfuzije.

		Predviđeno stanje		
		pozitivno stanje	negativno stanje	
Stvarno stanje	pozitivno stanje	stvarno pozitivno	lažno negativno	osjetljivost
	negativno stanje	lažno pozitivno	stvarno negativno	specifičnost
		PPV	NPV	

Tablica 1.1: Matrica konfuzije

Poglavlje 2

Pojmovnik

Budući da se ovaj rad bavi metodama koje se podjednako mogu primjenjivati na različite vrste podataka, od interesa je uvesti neke pojmove odmah na početku u smislu u kojem se oni koriste u ovom radu.

Osnovni pojam je tako pojam slova. Slovo je jednostavno rečeno proizvoljan znak koji ima neko značenje u specifičnom kontekstu konkretne primjene. Konačni skup svih korištenih slova čini alfabet. Niz slova čini redak, a više redaka čini tekst. Podniz čini bilo koji segment slova unutar retka.

Upitom smatramo skup koji se sastoji od jednog ili više nizova jednake ili različitih duljina ovisno o specifikacijama korištenog algoritma. Odgovor je skup koji se sastoji od jednog ili više nizova jednake ili različitih duljina koje algoritam identificira kao nizove dovoljno slične upitu.

Prijevod je preslikavanje s jednog alfabeta na drugi.

Poglavlje 3

Proteom, genom i genetski kod

3.1 Biološki koncepti

Proteom je skup svih proteina određenog organizma ili stanice koji nastaju kao posljedica ekspresije gena u određenom trenutku. Proteini su biološke makromolekule sastavljene od aminokiselina. Aminokiselinski niz proteina određuje njegovu trodimenzionalnu strukturu i funkciju (v. [1]). U tablici 3.1 dan je popis standardnih aminokiselina. Motiv proteinskog niza je kratak niz aminokiselina, u pravilu 5 do 20, koji je ostao djelomično sačuvan selekcijskim pročišćavanjem ili evolucijom i ima neko biološko značenje. U principu, motiv prepoznajemo time što ima specifičan supstitucijski uzorak.

Kratica	Naziv	Kratica	Naziv
A	Alanin	M	Metionin
C	Cistein	N	Asparagin
D	Asparaginska kiselina	P	Prolin
E	Glutaminska kiselina	Q	Glutamin
F	Fenilalanin	R	Araginin
G	Glicin	S	Serin
H	Histidin	T	Treonin
I	Izoleucin	V	Valin
K	Lizin	W	Triptofan
L	Leucin	Y	Tirozin

Tablica 3.1: Standardne aminokiseline

Genom je genetski materijal organizma. Za razliku od proteoma, genom je isti za svaku stanicu. Slično kao što je informacija računalne datoteke pohranjena u obliku niza

nula i jedinica, tako je genetska informacija organizama pohranjena u nizu koji čine četiri nukleotidnih baza (Tablica 3.2) koje sačinjavaju molekule DNA (v. [1]).

Kratica	Naziv
A	Adenin
C	Citozin
G	Guanin
T	Timin

Tablica 3.2: Nukleotidne baze

Genom se grubo može podijeliti na dva dijela, kodirajući dijelovi te nekodirajuća DNA. Od posebnog interesa su nam kodirajući dijelovi koji sudjeluju u biosintezi proteina. Set pravila po kojima se informacija pohranjena u kodirajućim djelovima genoma prevodi u proteine naziva se genetski kod. Prema genetskom kodu (Tablica 3.3) triplet nukleotidnih baza (kodon) kodira jednu aminokiselinu.

Kodon	Amino.	Kodon	Amino.	Kodon	Amino.	Kodon	Amino.	Kodon	Amino.
ATG	M	GTC	V	GCG	A	ACA	T	CAA	Q
ATT	I	GTA	V	GGT	G	ACG	T	CAG	Q
ATC	I	GTG	V	GGC	G	TCT	S	AAT	N
ATA	I	TTT	F	GGA	G	TCC	S	AAC	N
CTT	L	TTC	F	GGG	G	TCA	S	CAT	H
CTC	L	TGT	C	CCT	P	TCG	S	CAC	H
CTA	L	TGT	C	CCC	P	AGT	S	GAA	E
CTG	L	TGC	C	CCA	P	AGC	S	GAG	E
TTA	L	GCT	A	CCG	P	TAT	Y	GAT	D
TTG	L	GCC	A	ACT	T	TAC	Y	GAC	D
GTT	V	GCA	A	ACC	T	TGG	W	AAA	K
AGG	R	AGA	R	CGG	R	CGA	R	AAG	K
CGC	R	CGT	R						

Tablica 3.3: Genetski kod

Napomenimo kako nam je od posebnog interesa kodon ATG, odnosno aminokiselina M budući da ona označava početak kodiranja. Također, iz tablice je potrebno uočiti kako postoje kodoni koji ne kodiraju aminokiseline (TAA, TGA, TAG). Naime, radi se o kodonima koji označavaju kraj translacije. Nadalje, od posebna značaja u radu će nam biti činjenica kako kodoni iz tablice 3.3 kodiraju svaki po jednu aminokiselinu, ali kako jednu aminokiselinu može kodirati više različitih kodona.

3.2 Problem

Zahvaljujući napretku u računalnim resursima, danas se sve više pokušavaju analizirati različiti biološki nizovi računalnim metodama te primjenama string matching metoda na biološke nizove, a u svrhu bolje detekcije srodnih bioloških nizova (primjerice, identifikacije proteina iste proteinske familije). U radu [5] je tako prikazan rezultat primjene iterativnog traženja proteinskih familija GDSL lipaza na proteomima različitih organizama polazeći od jednog motiva specifičnog za tu proteinsku familiju (FVFGDSLSDA).

Budući da je proteom specifičan na razini stanice, a postoji veza između njega i genoma koji je univerzalan za svaku stanicu organizma, od interesa je istražiti može li se ta metoda uz nužne modifikacije prenijeti na analiziranje samog genoma.

Prvi problem koji susrećemo jest da nemamo nikakav popis analogona motiva proteinske familije na genomu. Samim time, nemamo dobru polaznu točku od koje bi mogli izvršiti pretraživanje. Stoga, moramo pronaći način kako prenijeti motive s proteoma na genom. Prvo rješenje koje se nameće jest da prevedemo postojeće motive na proteomu koristeći tablicu genetskog niza.

Međutim, kako smo već napomenuli u ranijem poglavlju, jedan kodon iz tablice 3.3 kodira jednu aminokiselinu, ali jednu aminokiselinu može kodirati više različitih kodona pa genetski kod nije bijektivan. Stoga naš prijevod ne bi bio jedinstven. Ukoliko bi uzeli sve moguće prijevode, upitno je koliko bi oni zaista imali biološko značenje ili se čak uopće pojavljivali u dijelu genoma koji zaista kodira danu proteinsku familiju. Stoga smo morali osmisliti način kako doskočiti tom problemu te postići neki konsenzus prijevoda za koji bi mogli očekivati kako ima biološko značenje.

3.3 Konsenzus prijevoda

U svrhu pronalaska konsenzusa, iskoristit ćemo genetski kod iz tablice 3.3 kako bi preveli kodirajuću DNA u aminokiselinski niz te potražiti proteine odabrane proteinske familije u tom prijevodu. Proces se komplicira činjenicom kako nećemo moći egzaktno pronaći sve nizove zbog bioloških procesa poput delecije, insercije i mutacije koje se odvijaju na proteinskim nizovima te oni neće odgovarati u potpunosti prijevodu genoma. Stoga ćemo odabrati samo najbolje prijevode ignorirajući spomenute biološke procese. Potom ćemo na njima potražiti specifični motiv (FVFGDSLSDA) te zapamtiti nizove i pozicije u kojima se nalaze nizovi koji su mu najviše slični. Na kraju ćemo u u originalnom genom pogledati koji nizovi odgovaraju pronađenima.

Kako bi mogli spariti prijevod i protein te motiv i dovoljno slične kratke nizove, potrebne su nam neke metode sličnosti. Prirodno bi nam se nametnule metode poravnanja bioloških nizova poput Needleman-Wunsch ili Smith-Waterman algoritma. Međutim, proteinski i genetski nizovi s kojima baratamo su izuzetno dugi i ima ih mnogo te je stoga

implementacija takvog računanja prespora. Naime, u primjeru krumpira na kojem ćemo demonstrirati razultate primjene metoda ovog rada, kodirajuća DNA se sastoji od preko 50000 nizova od kojih su neki duljine preko 1500 nukleotidnih baza. Stanje je nešto bolje na proteinskoj familiji krumpirovih GDSL lipaza gdje imamo 123 proteina duljina koje variraju od svega 50 do preko 400 aminokiselina dok se sam proteom krumpira sastoji od preko 35000 proteina. Dodatnu komplikaciju na kompleksnost izračuna predstavlja činjenica da ćemo poklapanje tražiti na nizovima čiji je alfabet duljine 20.

Stoga se okrećemo metodama string i pattern matchinga koristeći modifikaciju Hammingove udaljenosti čime ćemo značajno ubrzati traženje odgovarajućih prijevoda.

Hammingova udaljenost i Hammingov vektor

Neka je dan ulazni niz $x = x_1x_2...x_n$ duljine n i niz $y = y_1y_2...y_m$ također duljine n. Hammingova udaljenost niza x od niza y definira se kao broj pozicija na kojima se niz y razlikuje od zadanog niza x. Označimo tu udaljenost s $d(y, x)$.

Kako je naš cilj tražiti najsličnije nizove, koristimo modifikaciju Hammingove udaljenosti, tzv. Hammingovu bliskost, gdje umjesto broja pozicija na kojima se niz y razlikuje od niza x promatramo broj pozicija na kojima se niz y poklapa sa nizom x te tu vrijednost proglašavamo mjerom sličnosti dva nizga. Kako su dvije mjere potpuno analogne i dalje ćemo koristiti oznaku iz $d(y, x)$. Tu mjeru ilustriramo sljedećim primjerom.

X1	F	V	F	G	D	S	L	S	D	A
X2	F	V	C	A	A	M	D	L	I	K
X3	F	I	F	G	D	S	L	S	D	V

Pripadne Hammingove bliskosti su redom

$$d(X1, X2) = 2 \tag{3.1}$$

$$d(X1, X3) = 8 \tag{3.2}$$

Budući da ćemo mi tražiti najbolje sličnosti na cijelom tekstu koji je značajno dulji od upita, uvest ćemo odmah i pojam Hammingova vektora koji sadrži Hammingove udaljenosti odabranog niza sa svim podnizovima teksta koji su iste duljine kao i odabrani niz krećući se po tekstu metodom klizećeg prozora. Označimo taj vektor s H. Ponovno koristimo modifikaciju kakvu smo koristili i kod Hammingove udaljenosti gdje u Hammingovu vektoru gledamo bliskost umjesto udaljenosti. Vektor ilustriramo sljedećim primjerom.

Uzmimo motiv FVFGDSLSDA kao naš niz, a neka je naš tekst MATPFVFGDSLVDSG.

Niz	F	V	F	G	D	S	L	S	D	A
Podniz1	M	A	T	P	F	V	F	G	D	S
Podniz2	A	T	P	F	V	F	G	D	S	L
Podniz3	T	P	F	V	F	G	D	S	L	V
Podniz4	P	F	V	F	G	D	S	L	V	D
Podniz5	F	V	F	G	D	S	L	V	D	S
Podniz6	V	F	G	D	S	L	V	D	S	G

Pripadni Hammingov vektor H tada je jednak $(1, 0, 2, 0, 8, 0)$.

Postupak i primjena

Neka je T tekst koji se sastoji od k nizova $S_1, S_2, ..., S_k$ redom duljina $m_1, m_2, ..., m_k$. Neka je U upit, niz duljine n čiji najbolje poklapanje želimo pronaći u tekstu T. Metodom kliznog prozora za upit U računamo njegov Hammingov vektor (imajući na umu da se radi o modifikaciji Hammingova vektora iz odjeljka 3.3). Za brzo i efikasno računaje Hammingova vektora koristimo algoritam iz [6] čiji je pseudokod dan u algoritmu 1.

Algorithm 1 Izračun Hammingova vektora

```
procedure PSEUDOKOD
    int* PositionList[|A|];
    int MatchVector[ProteinLength];
    int HammingDistanceVector[ProteinLength];
    for i=0 to ProteinLength-1 do
        add i to PositionList[protein[i]];
    for i=0 to ProteinLength-1 do
        for j in PositionList[peptide[i]] do
            if j-i ≥ 0 then
                MatchVector[j-i] ++;
    for i=0 to ProteinLength-PeptideLength do
        HammingDistanceVector[i]=PeptideLength-MatchVector[];
```

Potom na svakom od izračunatih Hammingovih vektora pronađemo maksimalnu vrijednost te zapamtimo poziciju unutar niza na kojoj počinje podniz niza kojem odgovara ta maksimalna vrijednost.

Taj postupak ponovimo za svaki upit te tako dobijemo k maksimalnih vrijednosti i k pozicija za svaki od upita. Potom, za svaki od upita promatramo maksimume od tih k maksimalnih vrijednosti i pamtimo njihove pozicije unutar niza na kojoj počinje podniz niza

kojem odgovara ta maksimalna vrijednost. Upravo podnizove koji počinju na zapamćenim pozicijama smatramo kandidatima koje možemo proglasiti dovoljno sličnima našem upitu.

Navedeni algoritam primjenjujemo dva puta.

U prvoj primjeni, kao upite koristimo proteine proteinske familije GDSL lipaza krumpira, a kao tekst koristimo prijevod krumpirova genoma dobiven primjenom genetskog koda iz tablice 3.3. Svrha same primjene jest pronaći one nizove u genomu koji kodiraju proteine zadane proteinske familije.

Zbog otegotnih okolnosti opisanih u 3.2, moramo odabrati koje prijevode smatramo dovoljno dobrima. U tu svrhu, gledamo u kojem se postotku protein poklapa s prijevodom niza iz genoma koji smo identificirali kao najbolje poklapanje. Odabiremo ona poklapanja koja su dulja od 200 aminokiselina i koja se sa sparenim proteinom poklapaju u postotku većem od medijana svih postotaka nizova duljih od 200 (približna vrijednost: 0.72).

Time smo izabrali 46 nizova genoma koje smo identificirali kao kodirajuće nizove za neke od proteina iz proteinske familije GDSL lipaza. Iz tih 46 nizova izvlačimo upravo one segmente niza koji kodiraju proteine.

Potom, primjenjujemo gore opisani algoritam koristeći motiv FVFGDSLSDA kao upit, a koristeći gore dobivenih 46 segmenata nizova prijevoda genoma kao tekst. Koristeći kriterij poklapanja od 0.80, time dobijemo šest nizova koje detektiramo kao dovoljno slične motivu.

Budući da smo zapamtili poziciju tih šest nizova u segmentu, a ranije smo zapamtili i poziciju segmenata u prijevodu genoma, lako dolazimo do šest podnizova genoma koji kodiraju upravo te kratke nizove. U tablici 3.4 navodimo pronađene nizove i dijelove genoma koji ih kodiraju.

Aminokiselinski niz	Genom
FVFGDSLVDS	TTTGTGTTTGGTGATTCACTTGTTGATAGT
FVFGDSLVDT	TTTGTGTTTGGAGATTCATTAGTTGATACT
FIFGDSLSDV	TTCATCTTTGGAGACTCTCTTTCAGATGTT
FVFGDSLFDP	TTTGTGTTTGGTGATTCTCTCTTTGATCCT
FVFGDSLFDP	TTCGTGTTCGGCGATTCATTGTTCGATCCC
FVFGDSLVDN	TTTGTGTTTGGTGACTCTCTTGTTGACAAT

Tablica 3.4: Genetski kod

Pomnije promotrivši rezultate, na aminokiselinskim nizovima možemo primijetiti veliku očuvanost na pozicijama. Naime, svi nizovi se poklapaju na 7 od 10 pozicija. Dakle, očuvanost je 70%. Na genomu je pak ukupna očuvanost na svega 14 od 30 pozicija. Odnosno, očuvanost je približno 47%.

Tih šest nizova iz genoma uzimamo kao konsenzus segmenta genoma koji kodiraju motive dovoljno slične motivu FVFGDSLSDA te ih koristimo kao upit iterativnog pretraživanja na genomu, o čemu će više riječi biti u sljedećem poglavlju.

Poglavlje 4

Generiranje profila motiva

4.1 Algoritam iterativnog pretraživanja

Kao što je već spomenuto, primarni cilj ovog rada jest primjena algoritma iterativnog pretraživanja na danom tekstu polazeći od konkretnog upita u svrhu identifikacije nizova od interesa. Stoga pobliže opisujemo korišteni algoritam s naglaskom na izgradnji modela u prvoj iteraciji, odnosno opisujemo modifikaciju algoritma iz [5] korištenu o ovom diplomskom radu.

Neka je $\mathcal{A} = \{a_1, a_2, ..., a_l\}$ alfabet duljine $l \in \mathbb{N}$ gdje su $a_1, a_2, ..., a_l$ slova alfabeta $\mathcal{A}$. Neka je $T = \{t_1, t_2, ..., t_n\}, n \in \mathbb{N}$ tekst gdje $t_1, t_2, ..., t_n$ označavaju pojedine retke iz teksta redom duljina $m_1, m_2, ..., m_n \in \mathbb{N}$ sačinjene od slova alfabeta $\mathcal{A}$. Za svaki $j \in \{1, 2, ..., l\}$ računamo inicijalnu distribuciju slova u tekstu i označavamo je s Δ. Inicijalnu distribuciju čine relativne frekvencije pojedinog slova a alfabeta $\mathcal{A}$ u tekstu T.

Neka je $U^0 = \{u_1^0, u_2^0, ..., u_{k_0}^0\}$ početni upit gdje $u_i^0, i \in \{1, 2, ..., k_0\}, k_0 \in \mathbb{N}$ označavaju nizove slova alfabeta $\mathcal{A}$ jednake duljine $d \in \mathbb{N}$. Neka je $U^i = \{u_1^i, u_2^i, ..., u_{k_i}^i\}$ odgovor koji dobijemo nakon i-te iteracije algoritma. Neka je $E_j^0(a)$ frekvencija slova $a \in \mathcal{A}$ u stupcima upita $\{(u_k^0)_j : k = 1, 2, ..., k_0\}$, a neka je a_j^0 broj različitih slova u stupcima $\{(u_k^0)_j : k = 1, 2, ..., k_0\}$.

Za svaki $k = 1, 2, ..., k_0$ računamo težine prema formuli

$$\widetilde{w}_d^0 = \sum_{j \in \{1,2,...,d\}, E_j^0((u_k^0)_j) \neq 0} \frac{1}{a_j^0 E_j^0((u_k^0)_j)} \tag{4.1}$$

koje potom normaliziramo

$$w_k^0 = \frac{\widetilde{w}_k^0}{\sum_{k=1}^{k=k_0} \widetilde{w}_k^0}. \tag{4.2}$$

Za svako slovo $a \in \mathcal{A}$ potom računamo težinski model

$$\widetilde{\mathcal{W}}_j^0(a) = \sum_{k\in\{1,2,...,k_0\},(u_k^0)_j=a} w_k^0 \tag{4.3}$$

te ga blago modificiramo formulom

$$\mathcal{W}_j^0(a) = \frac{\widetilde{\mathcal{W}}_j^0(a) + \frac{1}{100k_0}}{1 + 1/5k_0}. \tag{4.4}$$

Na samom kraju, model finaliziramo kao

$$\mathcal{L}_j^0(a) = log(\mathcal{W}_j^0(a)) - log(\Delta(a)) \tag{4.5}$$

gdje $\Delta(a)$ označava onaj element vektora Δ koji odgovara relativnoj frekvenciji slova a u tekstu T.

Neka je t niz slova iz alfabeta $\mathcal{A}$ duljine m. Tada log-odds vektor $v(t) \in \mathbb{R}^{m-d+1}$ na poziciji i definiramo s

$$v(t)_i = \mathcal{L}_1^0(t_i) + \mathcal{L}_2^0(t_{i+1}) + ... + \mathcal{L}_l^0(t_{i+d-1}). \tag{4.6}$$

Formula je ekvivalentna formuli

$$v(t)_i = \sum_{h=0}^{d-1} log\frac{P(x_{i+h}|y_{h+1})}{P(x_{i+h}|q)}, i = 1, 2, ..., m - d + 1 \tag{4.7}$$

gdje su $\{y_1, y_2, ..., y_d\}$ i q distribucije koje određuju pozicijski profil motiva.

Log-odds vektor računamo za svaki redak teksta T

$$\{v(t_i) : i = 1, ..., n\},$$

a potom računamo maksimume na svakom retku $\widetilde{v}_i$ =max$\{v(t_i)_h : h = 1, ..., m - d + 1\}$. Kako bi ubrzali računanje log-odds vektora ponovno se pozivano na brzi algoritam iz [6].

Budući da su $\widetilde{v}_i$ približno logistički distribuirani (v. korolar 1.3.1 i [3]), računamo aritmetičku sredinu μ i skalu s. Aritmetička sredina μ je pritom upravo aritmetička sredina od $\widetilde{v}_i, i = 1, ..., n$ dok se skala s računa po formuli

$$s = \frac{\sqrt{3}}{\pi}\sigma \tag{4.8}$$

gdje je σ standardna devijacija uzorka.

Potom računamo prag $p = \mu + K \cdot s$ gdje je K parametar koju direktno zadajemo. Svaki podniz z duljine d proizvoljnog retka t iz teksta T čiji je pripadni $v(z) \geq p$ dodajemo u odgovor U^1.

Naravno, i-ta iteracija se provodi poput prve iteracije uz korištenje odgovora U^{i-1} iz $i-1$ iteracije kao upita i modifikaciju finalnog modela. Naime, nakon što izračunamo $\mathcal{L}_j^i(a)$, definiramo

$$\widetilde{\mathcal{L}}_j^i(a) = log\left(\frac{1}{2}\Big(exp\mathcal{L}_j^i(a) + exp\mathcal{L}_j^0(a)\Big)\right) \tag{4.9}$$

te ga u daljnjim izračunima koristimo umjesto $\mathcal{L}_j^i(a)$. Ostatak algoritma ostaje nepromijenjen.

Algoritam se iterira dok se ne dostigne unaprijed zadan broj iteracija ili dok odgovor ne ostane nepromijenjen između dvije iteracije.

4.2 Primjena algoritma

Kao što je već najavljeno, opisani algoritam primjenjujemo na genomu krumpira. U ovom slučaju alfabet se sastoji od 4 slova koja predstavljaju nukleotidne baze i definira se kao $\mathcal{A} = \{A, C, G, T\}$. Kao tekst T koristimo nizove kodirajuće DNA krumpira. Sam algoritam primjenjujemo više puta uzmajući različite početne upite U^0 i zadajući različite parametre K. Kao početne upite U^0 uzimamo podnizove genoma identificirane u odjeljku 3.3, prvo ih sve koristeći kao početni upit, a potom pojedinačno. Maksimalni broj iteracija smo pritom ograničili na 10.

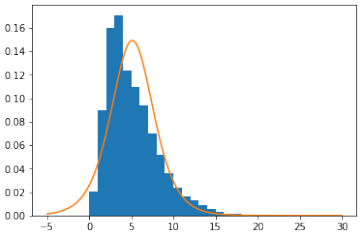
Cilj je algoritmom identificirati one nizove genoma koji kodiraju proteinsku familiju GDSL lipaza krumpira. Međutim, tu ponovno nailazimo na prepreku. Naime, zbog loše anotiranosti bioloških nizova, gdje se koriste različite oznake za proteom i genom, ne znamo koji od danih nizova genoma zaista kodiraju proteinsku familiju GDSL lipaza krumpira. Stoga je kvalitetu našeg odgovora teško provjeriti.

Kako bi stoga evaluirali kvalitetu našeg odgovora, morali smo ponovno posegnuti za prijevodom s genoma na proteom pomoću genetskog koda iz tablice 3.3 budući da na proteomu znamo točnije koji su to nizovi proteinske familije GDSL lipaza krumpira. Ponovno se pozivamo na algoritam iz odjeljka 3.3 te prevedene nizove tražimo na proteomu krumpira. U tom slučaju, koristimo prijevod zadnjeg odgovora iterativnog pretraživanja U^{10} kao upit, dok nam tekst čini cijeli proteom krumpira.

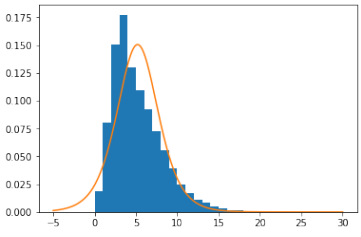
Važno je napomenuti da smo time potencijalno pogoršali rezultate našeg odgovora budući da se prijevod ponovno aproksimira. Međutim, bez konzistentnosti anotacije na biološkim nizovima, trenutno nemamo boljeg rješenja za evaluaciju našeg odgovora.

Usporedba rezultata

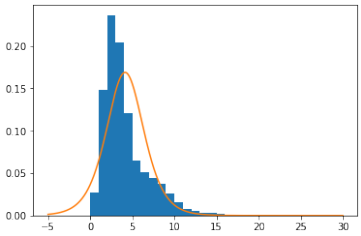
Pogledajmo sada neke konkretne grafove i rezultate primjene algoritma.



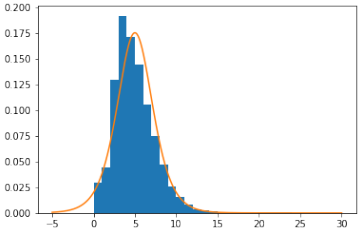
Slika 4.1: Puni upit
Parametar 7



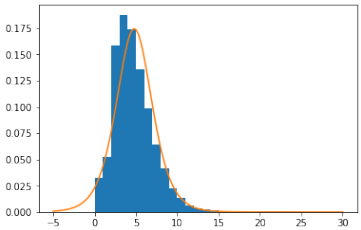
Slika 4.2: Puni upit
Parametar 8



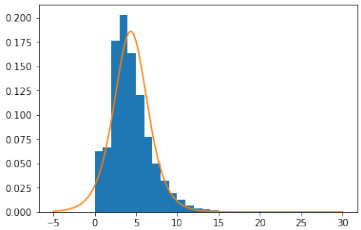
Slika 4.3: Puni upit
Parametar 9



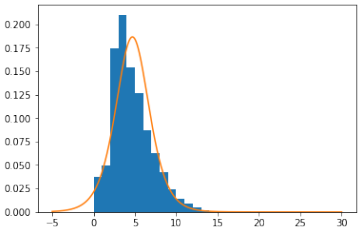
Slika 4.4: Niz 1
Parametar 7



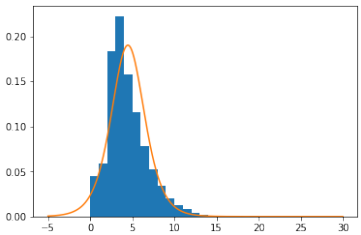
Slika 4.5: Niz 1
Parametar 8



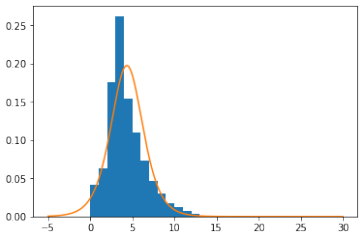
Slika 4.6: Niz 1
Parametar 9



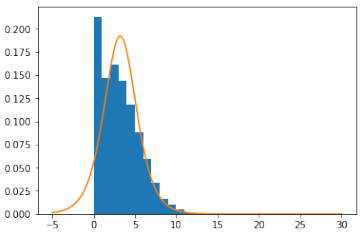
Slika 4.7: Niz 2
Parametar 7



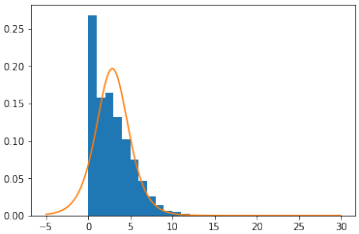
Slika 4.8: Niz 2
Parametar 8



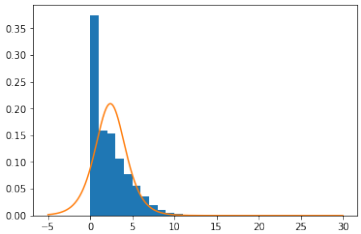
Slika 4.9: Niz 2
Parametar 9



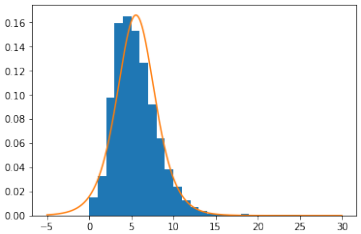
Slika 4.10: Niz 3
Parametar 7



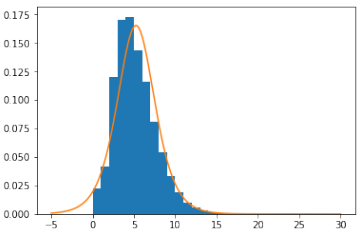
Slika 4.11: Niz 3
Parametar 8



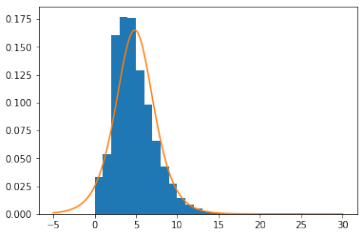
Slika 4.12: Niz 3
Parametar 9



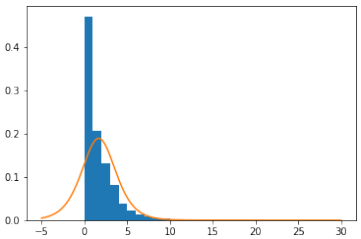
Slika 4.13: Niz 4
Parametar 7



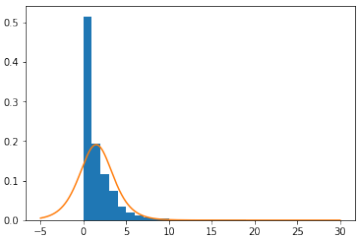
Slika 4.14: Niz 4
Parametar 8



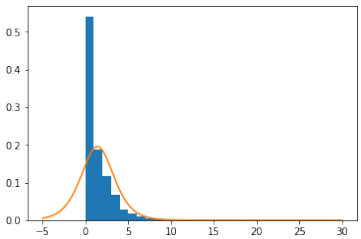
Slika 4.15: Niz 4
Parametar 9



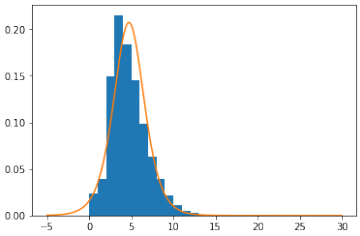
Slika 4.16: Niz 5
Parametar 7



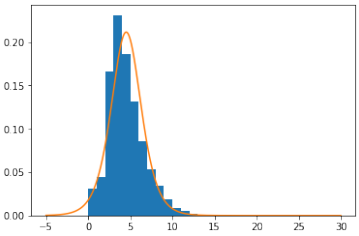
Slika 4.17: Niz 5
Parametar 8



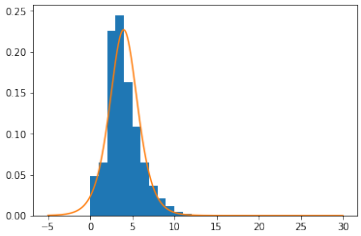
Slika 4.18: Niz 5
Parametar 9



Slika 4.19: Niz 6 Parametar 7



Slika 4.20: Niz 6 Parametar 8



Slika 4.21: Niz 6 Parametar 9

Na slikama 4.1 do 4.21 su dani histogrami $\widetilde{v}_i$ vrijednosti svih podnizova našeg grafa u posljednoj iteraciji po različitim upitima zajedno s funkcijom gustoće teoretske distribucije.

Primijetimo kako se ponašanje koje najviše odudara od teoretske distribucije uočava u slikama 4.10 do 4.12 i 4.16 do 4.18 koje odgovaraju primjeni algoritma gdje je početni upit U^0 sadržavao nizove 3 i 5 respektivno. Valja napomenuti kako su nam svi upiti podjednako dobar kandidat genomskog niza koji kodira motiv FVFGDSLSDA te ne možemo reći koji je od njih najbolji a priori. Stoga se u daljnjim razmatranjima ponajviše bavimo rezultatima upita koji sadrži svih šest nizova identificiranim u odjeljku 3.3.

Promotrimo sad rezultate evaluacije odgovora u vidu matrica konfuzije za upit U^0 koji sadrži svih šest nizova identificiranih u odjeljku 3.3 za različite parametre K.

		Testiranje		
		pozitivni	negativni	
Stvarno	pozitivni	54	69	osjetljivost=0.44
	negativni	253	34628	specifičnost=0.99
		PPV=0.18	NPV=0.998	

Tablica 4.1: Matrica konfuzije, Parametar 7

		Testiranje		
		pozitivni	negativni	
Stvarno	pozitivni	44	79	osjetljivost=0.36
	negativni	45	34836	specifičnost=0.999
		PPV=0.49	NPV=0.998	

Tablica 4.2: Matrica konfuzije, Parametar 8

		Testiranje		
		pozitivni	negativni	
Stvarno	pozitivni	39	84	osjetljivost=0.32
	negativni	82	34799	specifičnost=0.998
		PPV=0.32	NPV=0.998	

Tablica 4.3: Matrica konfuzije, Parametar 9

Primijetimo kako su specifičnost i negativna prediktivna vrijednost konzistentno visoke za sve izbore parametra K što je i očekivano zbog relativno malog broja proteina iz familije GDSL lipaza krumpira (svega 123) naspram velikog broja nizova u samom proteomu krumpira (preko 35000).

U ovom slučaju, puno su važnije mjere osjetljivost i pozitivna prediktivna vrijednost koje pokazuju najbolje rezultate za vrijednost parametra $K = 7$ i $K = 8$, no one su i dalje preniske da bi mogli zaključiti kako naš algoritam dobro identificira one nizove genoma koji kodiraju proteinsku familiju GDSL lipaza krumpira.

Poglavlje 5

Osvrt i moguća poboljšanja algoritma

Na samom kraju, osvrnimo se na rezultate iz [5] koji se bavi rezultatima na samom proteomu. Naime, uz parametar $K = 7.5$ postiže se pozitivna prediktivna vrijednost testa koja iznosi 0.72. Ta vrijednost je značajno bolja od najbolje pozitivne prediktivne vrijednosti testa postignute u ovom radu, a koja iznosi 0.49 za parametar $K = 8$.

Takav rezultat smo mogli donekle i očekivati. Naime, više puta u ovom radu se okrećemo određenim aproksimacijama stvarnog stanja te stoga unosimo određenu dozu dodatnog kaosa. Drugi element koji nam je mogao implicirati lošije rezultate na genomu je upravo konsenzus prijevoda iz odjeljka 3.3 koji je pokazao kako je ponašanje na genomu kaotičnije od ponašanja na proteomu. Naime, prisjetimo se kako je očuvanost na pozicijama na proteomu bila 70% dok je očuvanost na pozicijama na genomu za ekvivalentne nizove bila svega 47%.

Postoji prostor mogućem poboljšanju rezultata ovog rada u smislu pronalaska načina za bolju anotaciju nizova genoma čime bi izbjegli prijevod na proteom u svrhu evaluacije odgovora. Na taj način također bi izbjegli i ranije izbore dovoljno dobrih prijevoda što bi olakšalo traženje dovoljno dobrih genomskih nizova koji kodiraju nizova dovoljno slične motivu FVFGDSLSDA. Također bi se mogao poboljšati model uvođenjem matrice evolucije genoma, svojevrsnog pandana $PAM120$ matrice korištene u izgradnji modela u [5].

Takva pak razmatranja ostavljamo za neke buduće radove i istraživanja.

Bibliografija

[1] B. Alberts, A. Johnson i J. Lewis, *Molecular Biology of the Cell, 6th edition*, Garland Science, 2015.

[2] M. Cigula, *Iterativna optimizacija modela i pretraživanje proteoma*, Diplomski rad, Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, Matematički odsjek, 2016.

[3] L. de Haan i A. Ferreira, *Extreme Value Theory: An Introduction*, Springer, 2006.

[4] A. Medved, *Lokalno poravnanje i prepoznavanje motiva*, Diplomski rad, Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, Matematički odsjek, 2016.

[5] B. Rabar, S. Ristov, M. Zagorščak, M. Rosenzweig i P. Goldstein, *IGLOSS: Iterative Gapless Local Similarity Search*, (2018), `https://arxiv.org/abs/1807.11862`.

[6] S. Ristov, *A Fast and Simple Pattern Matching with Hamming Distance on Large Alphabets*, Journal of Computational Biology **11** (2016), br. 23, 874–876.

[7] N. Sarapa, *Teorija vjerojatnosti*, Školska knjiga, 2002.

[8] A. Đurić, *Analiza tehnika traženja proteinskim motiva*, Diplomski rad, Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, Matematički odsjek, 2018.

Sažetak

Cilj ovog diplomskog rada bio je testirati algoritam iterativnog pretraživanja te analizirati rezultate ovisno o parametrima. Također, važno je bilo usporediti te rezultate s rezultatima sličnog algoritma na analognom problemu. Za analizu smo odabrali genom i proteom krumpira te proteinsku familiju GDSL lipaza krumpira. Na tim nizovima genoma proveli smo postupak traženja specifičnog motiva proteinske familije GDSL lipaza krumpira kojeg smo prethodno preveli u alfabet genoma. Cilj je bio identificirati sve one nizove na genomu koji kodiraju proteinsku familiju GDSL lipaza. Analizirali smo naše rezultate te ih usporedili s rezultatima na proteomu. Zaključili smo kako je ponašanje na našem primjeru poprilično loše. Međutim, postoje indicije kako algoritam sam po sebi nije problem. Problem vjerojatno leži u nedostatku konzistentne anotacije na biološkim nizovima, različitim biološkim procesima zbog kojih naše aproksimacije unose prevelik kaos te u većoj varijablinosti na genomu naspram proteoma.

Summary

This thesis is concerned with an iterative search algorithm and the analysis of its results with respect to various choices of parameters. Furthermore, it was important to compare these results with results of a similar algorithm on an analogous problem. For the analysis, we chose the potato genome and proteome and the GDSL lipase protein family. We applied the iterative search algorithm on genome sequences in order to find the specific motif of the GDSL lipase protein family which was translated to the alphabet of the genome earlier in the thesis. The goal was to identify all those genome sequences which code the potato's GDSL lipase protein family. Finally, we analyzed our results in comparison to those obtained by proteome analysis. We concluded that the results were rather poor. However, there are indications that the algorithm itself is not as problematic as the results would imply. It is likely that the problem lies in the lack of consistent annotation of biological sequences, different biological processes due to which our approximations create far too much chaos, and in the greater variability of the genome when compared to proteom.

Životopis

Rođena sam 08.02.1993. godine u Splitu. Osnovnoškolsko obrazovanje pohađala sam u Osnovnoj školi Domovinske zahvalnosti u Kninu od 1999. do 2007. godine. Nakon toga, od 2007. do 2011. pohađala sam III. gimnaziju Split. Po završetku srednjoškolskog obrazovanja, 2011. godine upisujem preddiplomski studij matematike na Prirodoslovno-matematičkom fakultetu u Zagrebu. 2016. završavam preddiplomski studij te iste godine upisujem diplomski sveučilišni studij Matematička statistika na istom fakultetu.