Analiza proteinskih nizova iz CoViD-a 19

Mavrek, Iva

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Iva Mavrek

ANALIZA PROTEINSKIH NIZOVA IZ COVID-A 19

Diplomski rad

Voditelj rada:
doc. dr. sc. Pavle Goldstein

Zagreb, travanj 2022.

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. _____

Ovaj rad posvećujem prije svega sebi kao nagradu za borbu same sa sobom i svojim strahovima. Ponosna sam na sebe što ne odustajem kad je teško. Zahvaljujem mentoru doc. dr. sc. Pavlu Goldsteinu na strpljenju, razumijevanju i podršci u pisanju ovog rada. Hvala i najvećoj podršci koju čine moja obitelj, dečko i prijatelji. Da nema njih, ne bih ustajala nakon teških trenutaka i nastavljala hodati. Zato zapamti
"Hodaj, nebo strpljive voli."

Sadržaj

Uvod

Koronavirus već više od dvije godine hara stanovništvom čitavog svijeta. Krajem 2020. godine stigla su i cjepiva kao zaštita protiv koronavirusa. Koronavirus ili SARS-CoV-2 uzrokuje bolest dišnih puteva COVID-19, odnosno koronu. Koronavirus je RNK virus koji se sastoji od jedne molekule RNK s nekoliko gena i proteina koji štite tu srž virusa i omogućavaju da virus ulazi u ćelije živih bića. Koronavirus ima 4 virusna proteina: S, N, M i E. Diplomski rad bavit će se proučavanjem S proteina. On se naziva spike protein ili protein *šiljak*. Pomoću njega i ACE2 receptora na ćelijama virus može ući u stanice. Cjepiva se zasnivaju na S proteinu i zato je on interesantan za promatranje. Ovaj diplomski rad bavi se S proteinima u tri poglavlja. U prvom poglavlju navedeni su matematički pojmovi koji se koriste u ostalim poglavljima. U drugom poglavlju objašnjena je struktura sekvenciranih S proteina koji se koriste u radu. U trećem poglavlju navedeni su rezultati analiza provedenih na dva skupa, jedan su proteini za vrijeme karantene, a drugi skup čine proteini od početka pandemije pa do prosinca 2020. Vidjet ćemo koje mutacije su dobivene te ćemo generalizirati skup drugim skupom koji u nizovima sadrži crticu. Analiza proteinskih nizova rađena je u programskom jeziku Python.

Poglavlje 1

Definicije matematičkih pojmova

U ovom poglavlju navodimo definicije i propozicije iz linearne algebre koje se nalaze u [4], iz statistike u [5] i [6] te strojnog učenja u [8].

1.1 Linearna algebra

Definicija 1.1.1. *Neka su zadane binarne operacije zbrajanja* $+ : \mathbb{F} \times \mathbb{F} \to \mathbb{F}$ *i množenja* $\cdot : \mathbb{F} \times \mathbb{F} \to \mathbb{F}$ *na skupu* $\mathbb{F}$ *sa sljedećim svojstvima:*

(1) $\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma$, $\forall \alpha, \beta, \gamma \in \mathbb{R}$;
(2) postoji $0 \in \mathbb{R}$ *sa svojstvom* $\alpha + 0 = 0 + \alpha = \alpha$, $\forall \alpha \in \mathbb{R}$;
(3) za svaki $\alpha \in \mathbb{R}$ *postoji* $-\alpha \in \mathbb{R}$ *tako da je* $\alpha + (-\alpha) = -\alpha + \alpha = 0$;
(4) $\alpha + \beta = \beta + \alpha$, $\forall \alpha, \beta \in \mathbb{R}$;
(5) $\alpha(\beta\gamma) = (\alpha\beta)\gamma$, $\forall \alpha, \beta, \gamma \in \mathbb{R}$;
(6) postoji $1 \in \mathbb{R} \setminus \{0\}$ *sa svojstvom* $1 \cdot \alpha = \alpha \cdot 1 = \alpha$, $\forall \alpha \in \mathbb{R}$;
(7) za svaki $\alpha \in \mathbb{R}$, $\alpha \neq 0$, *postoji* $\alpha^{-1} \in \mathbb{R}$ *tako da je* $\alpha\alpha^{-1} = \alpha^{-1}\alpha = 1$;
(8) $\alpha\beta = \beta\alpha$, $\forall \alpha, \beta \in \mathbb{R}$;
(9) $\alpha(\beta + \gamma) = \alpha\beta + \alpha\gamma$, $\forall \alpha, \beta, \gamma \in \mathbb{R}$,

tada kažemo da je uređena trojka $(\mathbb{F}, +, \cdot)$ *polje.*

Definicija 1.1.2. *Neka je* V *neprazan skup na kojem su zadane binarna operacija zbrajanja* $+ : V \times V \to V$ *i operacija množenja skalarima iz polja* $\mathbb{F}$, $\cdot : \mathbb{F} \times V \to V$. *Kažemo da je uređena trojka* $(V, +, \cdot)$ *vektorski prostor nad poljem* $\mathbb{F}$ *ako vrijedi:*

(1) $a + (b + c) = (a + b) + c$, $\forall a, b, c \in V$;
(2) postoji $0 \in V$ *sa svojstvom* $a + 0 = 0 + a = a$, $\forall a \in V$;
(3) za svaki $a \in V$ *postoji* $-a \in V$ *tako da je* $a + (-a) = -a + a = 0$;

(4) $a + b = b + a$, $\forall a, b \in V$;
(5) $\alpha(\beta a) = (\alpha\beta)a$, $\forall \alpha, \beta \in \mathbb{F}$, $\forall a \in V$;
(6) $(\alpha + \beta)a = \alpha a + \beta a$, $\forall \alpha, \beta \in \mathbb{F}$, $\forall a \in V$;
(7) $\alpha(a + b) = \alpha a + \alpha b$, $\forall \alpha \in \mathbb{F}$, $\forall a, b \in V$;
(8) $1 \cdot a = a$, $\forall a \in V$.

Napomena 1.1.3. *Elementi vektorskog prostora nazivaju se vektorima, a elementi polja skalarima.*

Definicija 1.1.4. *Neka su* $m, n \in \mathbb{N}$. *Preslikavanje*

$$A : \{1, ..., m\} \times \{1, ..., n\} \to \mathbb{F}$$

naziva se matrica tipa (m,n) (ili $m \times n$*) s koeficijentima iz polja* $\mathbb{F}$. *Skup svih takvih matrica označavamo s* $M_{mn}(\mathbb{F})$.

Dakle, matrica A uređenom paru (i,j), $1 \leqslant$ i $\leqslant$ m, $1 \leqslant$ j $\leqslant$ n, pridružuje neki skalar iz polja $\mathbb{F}$. Uobičajeno je te funkcijske vrijednosti A(i,j) označiti s a_{ij} te ih zapisati u tablicu s m redaka i n stupaca na sljedeći način:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \dots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}.$$

Uređenu n-torku

$$(a_{i1}, a_{i2}, ..., a_{in})$$

nazivamo i-ti redak matrice A, a uređenu m-torku

$$(a_{1j}, a_{2j}, ..., a_{mj})$$

nazivamo j-ti stupac matrice A. Element a_{ij} nalazi se na presjeku i-tog retka i j-tog stupca, a čitavu matricu kraće zapisujemo kao A = $[a_{ij}]$ ili A = (a_{ij}).

Napomena 1.1.5. *Skup svih matrica tipa (m,n) označavamo s* $M_{mn}(\mathbb{F})$. *Lako se pokaže da je* $M_{mn}(\mathbb{F})$ *vektorski prostor nad* $\mathbb{F}$ *uz koordinatno definirane operacije zbrajanja i množenja skalarima. Matrice koje imaju samo jedan redak (tj.* $m = 1$*) nazivaju se retčane matrice, a one koje imaju samo jedan stupac (tj.* $n = 1$*) nazivaju se stupčane matrice. Skup svih retčanih matrica,* $M_{m1}(\mathbb{F})$, *također je vektorski prostor, kao i skup svih stupčanih matrica,* $M_{1n}(\mathbb{F})$. *Zato retčanu matricu nazivamo* ***vektor-redak****, a stupčanu* ***vektor-stupac****.*

Definicija 1.1.6. *Neka je V vektorski prostor nad poljem $\mathbb{F}$. Skalarni produkt na V je preslikavanje $\langle\cdot,\cdot\rangle : V \times V \to \mathbb{F}$ koje ima sljedeća svojstva:*

(1) $\langle x,x\rangle \geqslant 0$, $\forall x \in V$;
(2) $\langle x,x\rangle = 0 \Leftrightarrow x = 0$;
(3) $\langle x_1 + x_2,y\rangle = \langle x_1,y\rangle + \langle x_2, y\rangle$, $\forall x_1, x_2, y \in V$;
(4) $\langle \alpha x,y\rangle = \alpha\langle x,y\rangle$, $\forall \alpha \in \mathbb{F}$, $\forall x, x \in V$;
(5) $\langle x,y\rangle = \overline{\langle y, x\rangle}$, $\forall x, y \in V$.

Definicija 1.1.7. *Vektorski prostor na kojem je definiran skalarni produkt zove se unitaran prostor.*

Napomena 1.1.8. *U $\mathbb{R}^n$ skalarni, odnosno tzv. kanonski produkt definiran je s $\langle(x_1, ..., x_n),(y_1, ..., y_n)\rangle = \sum_{i=1}^{n} x_i y_i$.*

Definicija 1.1.9. *Neka je V unitaran prostor. Norma na V je funkcija $\|\cdot\| : V \to \mathbb{R}$ definirana s $\|x\| = \sqrt{\langle x, x\rangle}$.*

Propozicija 1.1.10. *Norma na unitarnom prostoru V ima sljedeća svojstva:*

(1) $\|x\| \geqslant 0$, $\forall x \in V$;
(2) $\|x\| = 0 \Leftrightarrow x = 0$;
(3) $\|\alpha x\| = |\alpha|\|x\|$, $\forall \alpha \in \mathbb{F}$, $\forall x \in V$;
(4) $\|x + y\| \leqslant \|x\| + \|y\|$, $\forall x, y \in V$.

Napomena 1.1.11. *Svaka funkcija na vektorskom prostoru sa svojstvima iz 1.1.10 zove se norma. Norma koja potječe od standardnog skalarnog produkta na $\mathbb{R}^n$ dana je formulom $\|(x_1, ..., x_n)\| = \sqrt{\sum_{i=1}^{n} |x_i|^2}$. Ova norma zove se **euklidska**.*

Definicija 1.1.12. *Neka je V unitaran prostor. Preslikavanje $d : V \times V \to \mathbb{R}$ dano formulom $d(x,y) = \|x - y\|$ zove se metrika ili udaljenost vektora x i y.*

Propozicija 1.1.13. *Metrika na unitarnom prostoru ima sljedeća svojstva:*

(1) $d(x,y) \geqslant 0$, $\forall x, y \in V$;
(2) $d(x,y) = 0 \Leftrightarrow x = y$;
(3) $d(x,y) = d(y,x)$, $\forall x, y \in V$;
(4) $d(x,y) \leqslant d(x,z) + d(z,y)$, $\forall x, y, z \in V$.

1.2 Statistika

Neka je (Ω, $\mathcal{F}$, P) vjerojatnosni prostor i X realna funkcija na Ω. Ako je ishod pokusa reprezentiran točkom $\omega \in \Omega$, tada je tom ishodu pridružen realan broj X(ω). Vrlo je često važno znati vjerojatnost da je a < X(ω) < b, a, b $\in \mathbb{R}$, a < b, tj. da X padne u intervale (a,b) (a < b). Prema tome, želimo izračunati $P\{\omega \in \Omega; X(\omega) \in B\} = P\{X \in B\} = P(X^{-1}(B))$, gdje je $B = (a, b), a, b \in \mathbb{R}, a < b$. Da bi to bilo moguće, mora biti $X^{-1}(B) \in \mathcal{F}$ za svaki interval $B = (a, b)$. Budući da je svaki otvoren skup na $\mathbb{R}$ prebrojiva unija otvorenih intervala, lako je dokazati da vrijedi $\mathcal{B} = \sigma\{(a, b); a, b \in \mathbb{R}, a < b\}$. Dakle, ako je $X^{-1}((a, b)) \in \mathcal{F}$ za sve a,b $\in\mathbb{R}$, a ¡ b, tada je $X^{-1}(B) \in \mathcal{F}$, za svako $B \in \mathcal{B}$.

Definicija 1.2.1. *Neka je (Ω, $\mathcal{F}$, P) vjerojatnosni prostor. Funkcija $X : \Omega \to \mathbb{R}$ je slučajna varijabla (na Ω) ako je $X^{-1}(B) \in \mathcal{F}$ za proizvoljno $B \in \mathcal{B}$, tj. $X^{-1}(\mathcal{B}) \subset \mathcal{F}$.*

Definicija 1.2.2. *Neka je (Ω, $\mathcal{F}$, P) vjerojatnosni prostor. Kažemo da je X n-dimenzionalan slučajan vektor (na Ω) ako je $X^{-1}(B) \in \mathcal{F}$ za svako $B \in \mathcal{B}^n$, tj. $X^{-1}(\mathcal{B}^n) \subset \mathcal{F}$.*

Definicija 1.2.3. *Neka je X slučajna varijabla na (Ω, $\mathcal{F}$, P). X je jednostavna slučajna varijabla ako je njezino područje vrijednosti konačan skup.*

Lako je vidjeti da je X jednostavna slučajna varijabla ako i samo ako je

$$\mathrm{X} = \sum_{k=1}^{n} x_k \mathcal{K}_{A_k} ,$$

gdje su $x_1, ..., x_n$ realni brojevi, a $A_1, ..., A_n$ međusobno disjunktni događaji, $\cup_{k=1}^{n} A_k = \Omega$. Neka su $X_1, X_2 : \Omega \to \mathbb{R}$. Tada definiramo funkcije $X_1 \vee X_2$ i $X_1 \wedge X_2$ na Ω relacijama

$$(X_1 \vee X_2)(\omega) = max\{X_1(\omega), X_2(\omega)\}, \omega \in \Omega \tag{1.1}$$

i

$$(X_1 \wedge X_2)(\omega) = min\{X_1(\omega), X_2(\omega)\}, \omega \in \Omega. \tag{1.2}$$

Pomoću funkcije zadane s (1.2) definiramo pozitivan i negativan dio realne funkcije X na Ω:

$$X^+ = X \vee 0, X^- = (-X) \vee 0. \tag{1.3}$$

X^+ i X^- su nenegativne realne funkcije ($X^+, X^- \geqslant 0$) i lako je dokazati da vrijedi

$$X = X^+ - X^-, |X| = X^+ + X^-. \tag{1.4}$$

Korolar 1.2.4. *X je slučajna varijabla ako i samo ako su X^+ i X^- slučajne varijable.*

Definicija matematičkog očekivanja provodi se u tri koraka. Prvo se definira matematičko očekivanje jednostavne slučajne varijable, zatim nenegativne slučajne varijable i na kraju opće slučajne varijable. Neka je (Ω, $\mathcal{F}$, P) vjerojatnosni prostor. S $\mathcal{K}$ označimo skup svih jednostavnih slučajnih varijabli na Ω, a sa $\mathcal{K}_+$ skup svih nenegativnih funkcija iz $\mathcal{K}$. Neka je $X \in \mathcal{K}$,

$$X = \sum_{k=1}^{n} x_k \mathcal{K}_{A_k}, \tag{1.5}$$

gdje su $x_1, x_2, ..., x_n$ realni brojevi, a $A_1, A_2, ..., A_n$ međusobno disjunktni događaji.

Definicija 1.2.5. *Matematičko očekivanje od X u (1.5), koje označavamo s* $\mathbb{E}(X)$*, definira se s*

$$\mathbb{E}(X) = \sum_{k=1}^{n} x_k \mathbb{P}(A_k).$$

Neka je X sada nenegativna slučajna varijabla definirana na Ω. Tada postoji rastući niz (X_n, n $\in \mathbb{N}$) nenegativnih jednostavnih slučajnih varijabli takav da je

$$X = \lim_{n\to\infty} X_n. \tag{1.6}$$

Definicija 1.2.6. *Matematičko očekivanje od X u (1.6), koje označavamo s* $\mathbb{E}(X)$*, definira se s*

$$\mathbb{E}(X) = \lim_{n\to\infty} \mathbb{E}(X_n).$$

Definicija 1.2.7. *Kažemo da matematičko očekivanje od opće slučajne varijable X, koje označavamo s* $\mathbb{E}(X)$ *postoji ili da je definirano ako je barem jedna od veličina* $\mathbb{E}(X^+)$ *ili* $\mathbb{E}(X^-)$ *konačna, tj. ako vrijedi*

$$min\{\mathbb{E}X^+, \mathbb{E}X^-\} < \infty.$$

Tada je po definiciji

$$\mathbb{E}X = \mathbb{E}X^+ - \mathbb{E}X^-.$$

Definicija 1.2.8. *Neka je X slučajna varijabla na vjerojatnosnom prostoru (*Ω*,* $\mathcal{F}$*, P) i* $r > 0$. $\mathbb{E}(X^r)$ *zovemo r-ti moment od X, a* $\mathbb{E}(|X^r|)$ *zovemo r-ti apsolutni moment od X.*

Definicija 1.2.9. *Neka* $\mathbb{E}X$ *postoji (tj. konačno je). Tada* $\mathbb{E}[(X - \mathbb{E}X)^r]$ *zovemo r-ti centralni moment od X, a* $\mathbb{E}[|X - \mathbb{E}X|^r]$ *zovemo r-ti apsolutni centralni moment od X.*

Definicija 1.2.10. *Varijanca od X, koju označavamo s VarX ili* σ_X^2 *je drugi centralni moment od X, tj.*

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

Definicija 1.2.11. *Standardna devijacija od X, koju označavamo sa σ_X, je pozitivan drugi korijen iz varijance, tj.*

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

Navedimo neke značajke *opisne* ili *deskriptivne statistike* koje koristimo u radu. Deskriptivna statistika bavi se metodama prikaza skupova podataka pomoću tablica, grafikona i numeričkih pokazatelja. Podatke možemo podijeliti po tipu vrijednosti opažanog statističkog obilježja, odnosno varijable. Razlikujemo numeričke i kategorijalne varijable. Numeričke se dijele na diskretne i neprekidne. U ovom radu bavit ćemo se numeričkim diskretnim varijablama.

Postoji više različitih mjera centralnih tendencija podataka, a mi ćemo u radu koristiti aritmetičku sredinu.

Definicija 1.2.12. *Neka su $x_1, x_2, ..., x_n$ n vrijednosti numeričke slučajne varijable X. Aritmetička sredina tih vrijednosti (brojeva) je broj*

$$\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i \, .$$

Uz mjere lokacije, odnosno srednje vrijednosti skupa podataka, važno je svojstvo distribucije tih podataka to kako su podaci raspršeni, često u odnosu na neku srednju vrijednost. Najčešće korištena mjera raspršenja skupa numeričkih podataka je standardna devijacija. Ona je zapravo odraz količine varijabilnosti unutar danog skupa podataka.

Definicija 1.2.13. *Neka su $x_1, x_2, ..., x_n$ n vrijednosti numeričke slučajne varijable X. Standardna devijacija je srednje kvadratno odstupanje podataka od njihove aritmetičke sredine. Formulom*

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \overline{x})^2} \, .$$

Broj

$$s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \overline{x})^2$$

naziva se varijanca skupa podataka $x_1, x_2, ..., x_n$.

Najvažnije svojstvo slučajnih varijabli za primjene njihova je distribucija opisana gustoćom ili funkcijom distribucije.

Definicija 1.2.14. *Slučajna varijabla X ima normalnu razdiobu s parametrima μ i $\sigma^2 > 0$, i pišemo X $N(\mu, \sigma^2)$, ako je $ImX = \mathbb{R}$ i gustoća joj je*

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} .$$

Graf funkcije gustoće zove se Gaussova krivulja i ona je zvonolikog oblika.

Interpretacija parametara normalne razdiobe je da je $\mu = \mathbb{E}[X]$ i $\sigma^2 = Var[X]$. Linearna transformacija normalno distribuirane varijable je opet normalno distribuirana varijabla. Preciznije, ako je $X \sim N(\mu, \sigma^2)$ te ako su $a \neq 0$ i b realni brojevi, tada je $Y := aX + b \sim N(a\mu + b, a^2\sigma^2)$. Specijalno, **standardizirana verzija** normalne varijable X,

$$Z = \frac{X - \mu}{\sigma}, \tag{1.7}$$

je normalno distribuirana s očekivanjem 0 i varijancom 1. Kažemo da Z ima jediničnu normalnu razdiobu. Vrijednosti od Z su bezdimenzionalne (u smislu da nisu izražene u nekim fizikalnim jedinicama) i njima izražavamo koliko je standardnih devijacija pripadna vrijednost X udaljena (i na koju stranu) od svoje očekivane vrijednosti μ. Ako je $Z < 0$, tada je X za $|Z|$ standardnih devijacija manji od μ, a ako je $Z > 0$, tada je X za Z standardnih devijacija veći od μ.

Definicija 1.2.15. *Slučajni uzorak je niz nezavisnih jednako distribuiranih (n.j.d.) slučajnih varijabli. Označavamo ga s $\underline{X}$.*

Na primjer, ako se sastoji od n n.j.d. varijabli $X_1, X_2, ..., X_n$, $\underline{X}$ je slučajni vektor

$$\underline{X} = (X_1, X_2, ..., X_n).$$

Intuitivno, slučajni uzorak predstavlja niz mjerenja (opažanja) slučajnih vrijednosti izučavane varijable X na članovima (jedinicama) odabranim u uzorak na slučajan način iz populacije. Kažemo da se članovi biraju u uzorak na slučajan način ako svaki element iz populacije ima jednaku šansu da bude izabran u uzorak, neovisan o odabiru drugih članova u uzorku. Uz tu interpretaciju, dakle, varijable $X_1, X_2, ..., X_n$ su nezavisne i imaju distribuciju jednaku populacijskoj distribuciji varijable X.

Uređenu n-torku brojeva $\underline{x} = (x_1, x_2, ..., x_n)$, koja predstavlja realizaciju slučajnog uzorka $\underline{X}$, zovemo opaženi uzorak.

Definicija 1.2.16. *Statistika je funkcija slučajnog uzorka koja ne sadrži nepoznate parametre.*

Na primjer, **uzoračka sredina**

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

i **uzoračka varijanca**

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$$

su statistike.

Koristeći uzoračku sredinu i varijancu, (1.7) postaje formula

$$Z_i = \frac{X_i - \overline{X}}{S}. \tag{1.8}$$

1.3 Strojno učenje

Društvene, znanstvene i različite industrijske djelatnosti preplavljene su velikom količinom podataka koji se svakodnevno pohranjuju u bazama podataka. Pojavom tehnika i metoda strojnog učenja (eng. *machine learning*) odgovara se na problem kako iz raspoloživih skupova podataka, njihovim spremanjem, manipuliranjem i korištenjem, proizvesti novo znanje. Dakle, sustavi uče kroz iskustvo na temelju podataka. Strojno učenje ujedinjuje računalne discipline kao što su rudarenje podataka (eng. *data mining*), bioinformatika, robotika i mnoge druge.

Oblici strojnog učenja su:

1. Nadzirano učenje (eng. *Supervised learning*)
2. Nenadzirano učenje (eng. *Unsupervised learning*)
3. Učenje s podrškom (eng. *Reinforcement learning*).

Nadzirano učenje uzima eksplicitnu informaciju o primjerima i vrijednosti njihove ciljne varijable. Cilj je napraviti model koji će raditi predikcije na još neviđenim (novim) primjerima. Proces učenja kod nadziranog strojnog učenja dijeli se na dvije faze: treniranje i testiranje. Kako bi to bilo moguće, potrebno je skup podataka podijeliti u dvije zasebne cjeline, u podatke za treniranje i testiranje, gdje će najveći udio imati podaci za treniranje. U fazi treniranja podaci za treniranje uzimaju se kao ulazni te se njihove karakteristike uče uz pomoć odgovarajućeg algoritma. Glavni zadatak algoritma predvidjeti je izlazni podatak na temelju karakteristika ulaznog podatka, kako bi onda svoju predviđenu vrijednost mogao usporediti sa stvarnim izlaznim podatkom i pritom otkriti pogreške. Nakon toga se na temelju pronađene pogreške model modificira kako bi buduće predikcije mogle biti

preciznije, što zapravo daje svojstvo učenja. U fazi testiranja uzima se istrenirani model te se uz pomoć njega rade predikcije nad podacima za testiranje. S obzirom na to da podaci za testiranje nisu bili korišteni prilikom treniranja modela, rezultat predikcije nad njima poslužit će za evaluaciju, odnosno dobiva se informacija o tome je li istrenirani model dovoljno dobar za naše potrebe ili ga je potrebno dodatno usavršiti. U nadzirano učenje ubrajaju se klasifikacija, regresija i predikcija (eng. *Forecasting*).

Kod nenadziranog učenja uzimaju se samo primjeri bez ikakve anotacije ili povratne informacije o njihovoj kategorizaciji. Cilj je grupirati primjere, odnosno otkriti neku strukturnu pravilnost u podacima i projicirati podatke u niže-dimenzionalne prostore. Za razliku od nadziranog učenja, gdje je cilj naučiti preslikavanje iz ulaznog u izlazni podatak, u nenadziranom učenju nemamo izlazne podatke, već imamo ulazne podatke uz pomoć kojih želimo pronaći pravilnosti. U ovom je području čest pojam procjena gustoće, koji predstavlja pronalaženje uzoraka u prostoru ulaznih podataka koji se pojavljuju češće nego neki drugi uzorci. Jedna od najčešćih metoda za procjenu gustoće je klasteriranje odnosno grupiranje ulaznih podataka. U nenadzirano učenje, osim klasteriranja, ubrajaju se i otkrivanje, tj. detekcija iznimaka (eng. *Outlier detection*) i kompresija podataka.

Učenje s podrškom bavi se problemom pronalaženja i odabira odgovarajućih akcija koje je potrebno poduzeti kako bi se maksimizirala nagrada. Za razliku od nadziranog i nenadziranog učenja, ovdje nemamo podatke, već ih je potrebno saznati iz postupaka pokušaja i pogreške. U učenju s podrškom najčešće imamo niz stanja i akcija koje su u interakciji s okolinom. Često trenutna akcija utječe na nagradu u trenutnom vremenu i u nadolazećem vremenu, tj. ona može utjecati na odluku daljnjih akcija (učenje sekvenci akcija - roboti, igre). Glavno je obilježje podržanog učenja kompromis između istraživanja i eksploatacije. U istraživanju sustav isprobava nove akcije da utvrdi koliko su one efikasne, dok kod eksploatacije sustav koristi već ranije poznate akcije koje daju visoku nagradu.

K-means algoritam

Klaster analiza predstavlja statističku tehniku kojom se utvrđuju relativno homogene grupe objekata, tj. za dani skup podataka U, treba odraditi k podskupova (klastera) C_i, i = 1, 2, ..., k, koji su homogeni i/ili dobro separirani u odnosu na mjerne varijable. Elementi pojedinog klastera, C_i, sličniji su jedan drugome nego elementima izvan tog klastera.

Klasteriranje k-sredinama jedan je od najkorištenijih algoritama klasteriranja. On je vrsta nenadziranog učenja te se koristi kod obrade neobilježenih podataka, tj. podataka koji nemaju definirane kategorije. To je jednostavan iterativan proces koji služi za grupiranje podataka.

Neka su podaci reprezentirani skupom vektora $X = \{x_1, x_2, ..., x_n\} \subset \mathbb{R}^m$. Cilj postupka je pronaći optimalnu k-članu particiju $\{C_1, C_2, ..., C_k\}$ skupa X. Ovo se postiže minimi-

ziranjem funkcije cilja f, definirane u terminima klastera $C_1, C_2, ..., C_k$ i središta klastera $c_1, c_2, ..., c_k$. Dakle, funkcija f dana je s

$$f(C_1, C_2, ..., C_k, c_1, c_2, ..., c_k) = \sum_{i=1}^{k} \sum_{x \in C_i} d^2(x, c_i), \tag{1.9}$$

gdje je d(·, ·) Euklidska udaljenost dana u 1.1.12.

Procedura optimizacije k-means algoritma započinje inicijalnim odabirom k središta, $c_1, c_2, ..., c_k$, ili k klastera, $C_1, C_2, ..., C_k$, za skup vektora X i fiksni broj klastera k. Nakon inicijalizacije slijedi iteriranje sljedeća dva koraka:

1. korak: pridruživanje svake točke x klasteru s najbližim središtem,

$$C_i^{(t+1)} = \{x : d(x, c_i^{(t)}) \leqslant d(x, c_j^{(t)}), \forall j\} \tag{1.10}$$

2. korak: određivanje novih središta klastera uzimajući u obzir trenutnu particiju,

$$c_i^{(t+1)} = \frac{1}{|C_i^{(t)}|} \sum_{x \in C_i^{(t)}} x, \tag{1.11}$$

gdje su $C_i^{(t)}$ i $c_i^{(t)}$ i-ti klaster i i-to središte klastera u t-toj iteraciji, redom, a $|C_i^{(t)}|$ označava veličinu skupa $C_i^{(t)}$. U drugom koraku postavljamo središte c_i kao aritmetičku sredinu i-tog klastera C_i.

Inicijalna konfiguracija - bilo particija ili središta - može biti nasumično odabrana ili određena na neki drugi način. Algoritam se obično zaustavlja kad se stabiliziraju particije ili nakon unaprijed određenog broja koraka. Prvi i drugi korak smanjuju vrijednost funkcije cilja f. Štoviše, izbori u prvom i drugom koraku su *optimalni* izbori u smislu da maksimalno (lokalno) poboljšavaju danu konfiguraciju. Slijedi da je k-means pohlepni algoritam. Ako označimo s $f^{(i)}$ vrijednost od f u i-toj iteraciji, dobivamo padajući niz

$$f^{(1)} \geqslant f^{(2)} \geqslant ... \geqslant 0.$$

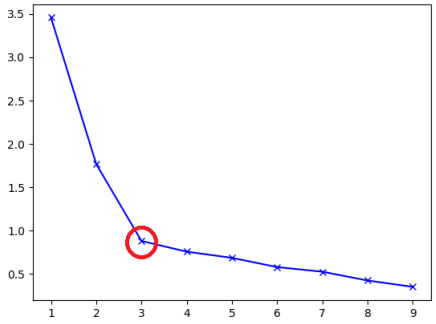
To ukazuje na to da će algoritam doseći minimum (od f) u konačnom broju koraka, ali i na to da će minimum često biti lokalan. Ovaj problem očituje se kroz osjetljivost k-means algoritma na početne uvjete, kao što je početni odabir klastera ili središta. To se u primjeni često rješava pokretanjem algoritma nekoliko puta s različitim početnim vrijednostima i biranjem najboljeg rješenja. Druga je mogućnost da se na bolji način konstruiraju početni klasteri ili središta.

Koristimo poseban način izbora središta za k-means algoritam. Posebno, neka je D(x) oznaka za najmanju udaljenost od točke do najbližeg središta koje smo već izabrali. Tada dobivamo **k-means++ algoritam**, objašnjen u [3] kao sljedeće:

1. odabir jednog centra, c_1, na slučajan način iz X,
2. odabir novog centra, c_i, odabirući x $\in$ X s vjerojatnošću $\frac{D(x)^2}{\sum_{x \in X} D(x)^2}$.
3. ponavljanje 2. sve dok se ne odabere svih k središta.
4. koraci k-means algoritma.

Ideja je, dakle, nasumično odrediti jedan centar, a svaki sljedeći odabrati na način da je što dalje od ostalih centara. Ovakav izbor početnih centara ubrzava konvergenciju algoritma i značajno izbjegava lokalne minimume čime znatno smanjuje pogrešku grupiranja.

Metoda lakta (eng. *Elbow method*) je metoda za određivanje broja klastera. Ona promatra postotak objašnjene varijance kao funkciju broja klastera. Konačan broj klastera, k, je onaj za koji dodavanje još jednog klastera značajno ne poboljšava podjelu podataka. Preciznije, ako napravimo graf u kojem prikazujemo ovisnost broja klastera o postotku objašnjene varijance, prvi će klaster puno pridonijeti jer će objasniti veliki postotak varijance. Povećanjem broja klastera ta se dobit smanjuje. Tražimo za koji broj klalstera se prestane značajno smanjivati. Grafički to znači da ćemo dobiti kut blizu pravom kutu. Taj kut izgledom podsjeća na lakat pa se metoda zato naziva metodom lakta.



Slika 1.1: Primjer grafa metode lakta

Broj klastera gledamo na x-osi, a na y-osi je ciljna funkcija koja predstavlja varijabilnost podataka. Na slici 1.1 lakat je prisutan za $k = 3$ klastera.

Bioinformatika

Bioinformatika je znanost koja kombinira računalna znanja, informacijske tehnologije i genetiku. Bioinformatika, za razliku od ostalih bioloških znanosti, uglavnom barata samo s 20 ili čak samo s 4 različitih simbola za aminokiseline ili nukleotide. Priroda koristi taj mali broj molekula da u genetskom kodu zapiše poruke za život.

Definicija 1.3.1. *Neka je $S = S_1, S_2, ..., S_k$ skup od k proteinskih nizova. Poravnanje, M, nizova iz S je skup od k nizova jednake duljine, $M = S'_1, S'_2, ..., S'_k$, pri čemu vrijedi:*

1. *Svi $S'_1, S'_2, ..., S'_k$ jednake duljine.*
2. *Od S_i dobijemo S'_i tako da ubacimo crtice.*
3. *Nema nijedna pozicija u kojem su samo crtice.*

Poglavlje 2

Opis problema

2.1 Struktura podataka

Podaci će biti proteini iz korona virusa. On se sastoji od proteina M, N, S i E. Podaci za diplomski rad su višestruko poravnati nizovi aminokiselina. Pronađen je najdulji niz i onda su po parovima poravnati nizovi u odnosu na taj najdulji niz. Podaci se mogu podijeliti u dvije skupine. Prvu skupinu čine nizovi aminokiselina koji čine protein S, a koji su iz razdoblja od ožujka 2020. do lipnja 2020. i prema lokaciji su uzeti diljem svijeta. Drugu skupinu podataka čine nizovi proteina S, a uzeti su iz razdoblja od ožujka 2020. do studenog 2021., također iz cijelog svijeta. Druga skupina podataka iz vremena je kada je α varijanta virusa tek nastajala, a druge varijante nisu ni postojale. U prvoj skupini nema naznaka nikakvih varijanti virusa. Također, u prvoj skupini nije bilo miješanja virusa zbog mjera civilnog stožera, dok u drugoj skupini jest. S protein nazivamo i Spike protein ili šiljak. On omogućava virusu SARS-CoV-2 ulazak u stanice živih bića. Moderna cjepiva šalju genetičku poruku za ciljani protein, onaj koji najviše dolazi u dodir s površinom naših stanica i s kojim se naš organizam prvi susreće. To je u slučaju SARS-CoV-2 S protein. Zato je taj protein od velikog interesa.

U prvoj skupini želimo dobiti podjelu podataka na dva klastera kao što je to dobiveno u [7], ali uz prethodno standardiziranje podataka. U drugoj skupini primjenjujemo navedeno standardiziranje podataka jer uvodimo crticu (na mjesta gdje je preskočena aminokiselina u proteinu) te tražimo značajne mutacije prema rangu standardnih devijacija pozicija. U prvoj skupini imamo 3547 nizova duljine 1273, a u drugoj skupini imamo 9997 nizova duljine 1273.

U obje skupine podataka nalaze se aminokiseline A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W i Y, a u drugoj skupini imamo i nizove sa crticom (-). Budući da slova ne možemo uspoređivati jer nema metrike za usporedbu vrijednosti, definiramo preslikavanje u $\mathbb{R}^5$ koje "čuva" sve važne informacije o svojstvima aminokiselina. Dakle, svaku

aminokiselinu možemo pretvoriti u 5-dimenzionalan vektor numeričkih vrijednosti. Te koordinate nazivamo faktorima. Faktor I je bipolaran, faktor II je faktor sekundarne strukture, faktor III odnosi se na molekularni volumen ili veličinu aminokiseline, faktor IV odražava raznolikost kodona (relativnu kompoziciju aminokiselina u različitim proteinima), a faktor V označava elektrostatski naboj aminokiselina.

Aminokiselina	Kratica	Faktor I	Faktor II	Faktor III	Faktor IV	Faktor V
alanin	A	-0.591	-1.302	-0.733	1.570	-0.146
cistein	C	-1.343	0.465	-0.862	-1.020	-0.255
asparaginska kiselina	D	1.050	0.302	-3.656	-0.259	-3.242
glutaminska kiselina	E	1.357	-1.453	1.477	0.113	-0.837
fenilanin	F	-1.006	-0.590	1.891	-0.397	0.412
glicin	G	-0.384	1.652	1.330	1.045	2.064
histidin	H	0.336	-0.417	-1.673	-1.474	-0.078
izoleucin	I	-1.239	-0.547	2.131	0.393	0.816
lizin	K	1.831	-0.561	0.533	-0.277	1.648
leucin	L	-1.019	-0.987	-1.505	1.266	-0.912
metionin	M	-0.663	-1.524	2.219	-1.005	1.212
asparagin	N	0.945	0.828	1.299	-0.169	0.933
prolin	P	0.189	2.081	-1.628	0.421	-1.392
glutamin	Q	0.931	-0.179	-3.005	-0.503	-1.853
arginin	R	1.538	-0.055	1.502	0.440	2.897
serin	S	-0.228	1.399	-4.760	0.670	-2.647
treonin	T	-0.032	0.326	2.213	0.908	1.313
valin	V	-1.337	-0.279	-0.544	1.242	-1.262
triptofan	W	-0.595	0.009	0.672	-2.128	-0.184
tirozin	Y	0.260	0.830	3.097	-0.838	1.512
crtica	-	7.500	10.000	-6.000	-8.000	-1.000

Tablica 2.1: Faktori aminokiselina

Crtice, odnosno delecije puno su rijeđe nego supstitucije. One su često štetne jer zbog njih protein gubi funkciju. Iz tablice vidimo da faktori za crticu odskaču od ostalih. Zato će biti važno standardizirati podatke.

Kao što je već spomenuto, nizovi aminokiselina proteina S duljine su 1273. Na tim višestruko poravnatim nizovima svaku aminokiselinu zapišemo kao 5-dimenzionalan vektor. Duljina nizova postaje $1273 \cdot 5 = 6365$. Kako su od interesa one pozicije nizova na kojima dolazi do promjene aminokiseline, baratat ćemo s nizovima kraćim od 6365.

2.2 Priprema podataka

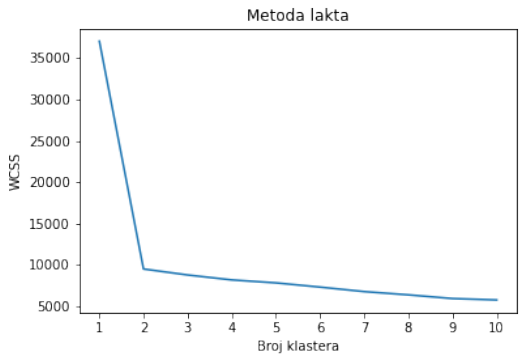
U obje skupine najviše nizova je duljine 1273. Zato ostale nizove, koji nisu duljine 1273, izbacimo. Također, izbacimo i nizove koji sadrže slovo koje nije oznaka jedne od 20 aminokiselina. U drugoj skupini nizova imamo i nizove koji sadrže crticu. Njih ostavljamo ako su duljine 1273. Iz prve skupine ostaje 2731 nizova, a iz druge 9988. Pretvorimo aminokiseline u 5-dimenzionalne vektore na način da matrica predstavlja sve nizove, tj. svaki je niz jedan redak matrice. Prvih pet stupaca prvog retka čini prva aminokiselina prvog niza, drugih pet stupaca prvog retka čini druga aminokiselina prvog niza, itd. Dakle, matrica je tipa 2731×6365. Radi očuvanja memorije, ali i da se brže provode algoritmi nad tim podacima, želimo smanjiti duljine nizova. To ćemo napraviti tako da promatramo samo one pozicije (indekse aminokiselina) u nizovima kod kojih je došlo do mutacije. Te pozicije pronalazimo pomoću matrice varijance. Prvo je izračunata matrica srednjih vrijednosti koja sadrži srednje vrijednosti po svakoj poziciji. Dobivena je matrica od 1 retka i 6365 stupaca. Zatim izračunamo varijancu po pozicijama i dobivamo matricu varijance iste veličine kao i matrica srednjih vrijednosti. Ako u stupcu imamo broj različit od nule, onda znači da je došlo do neke promjene i te pozicije spremamo. U prvoj skupini podataka broj pozicija u kojima su se dogodile promjene jednak je 153, a u drugoj skupini 512. Na ovaj način transformirali smo podatke u matrice u vektorskom prostoru kako bismo mogli provoditi analize nad njima.

Poglavlje 3

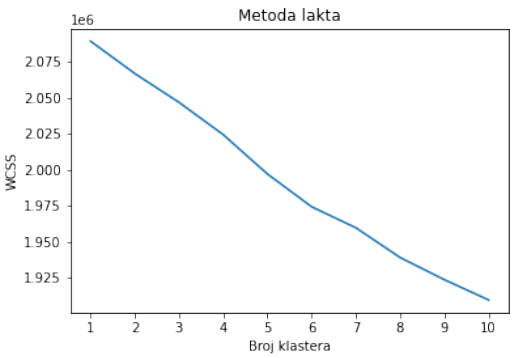
Grupiranje standardiziranih podataka k-means algoritmom i otkrivanje značajnih pozicija

3.1 Metoda lakta na standardiziranim podacima

U 1.3 je opisana metoda za procjenu najboljeg broja klastera, metoda lakta. U diplomskom radu [7] dobiven je graf metode lakta na danim podacima prikazan u 3.1. Cilj ovog diplomskog rada generalizirati je dobiven lakat na nizove sa crticom. Zato uvodimo standardizaciju podataka koja je opisana u 1.8.



Slika 3.1: Metoda lakta na nestandardiziranim podacima



Slika 3.2: Metoda lakta na standardiziranim podacima

Na x-osi grafova nalazi se broj klastera, a na y-osi je *WCSS*, što je engleska skraćenica za *Within-Cluster Sum of Square*. To je suma kvadratnih udaljenosti između svake točke

i središta klastera. Tu sumu želimo minimizirati dodavanjem klastera. Primijetimo da je na slici 3.1 jasan lakat za dva klastera, dok na slici 3.2 nema naznake lakta. Iz te slike ne možemo ništa zaključiti o najboljem broju klastera. Budući da nakon standardiziranja podataka varijanca unutar klastera pada jednoliko kako povećavamo broj klastera. Za standardizirane podatke ne postoji broj klastera nakon kojeg količina objašnjene varijance prestaje biti značajna. U ovom slučaju ona je jednoliko objašnjena dodavajući klastere. Zaključujemo da je prije standardizacije bila jasna podjela na dva klastera, a nakon standardizacije razlike u klasterima u potpunosti su se obrisale. Zašto je došlo do toga?

Pogledamo li sljedeću tablicu nizova aminokiselina koje su standardizirane, čini se kao da svaka pozicija sadrži po jedan broj, odnosno da su na prvoj poziciji vrijednosti svakog retka iste, na drugoj poziciji vrijednosti svakog retka iste, itd.

	0	1	2	3	4	5	6	...	758	759	760	761	762	763	764
0	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
1	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
2	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
3	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
4	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2726	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
2727	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
2728	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
2729	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203
2730	0.027072	0.027072	0.027072	-0.027072	0.027072	-0.123457	-0.123457	...	-0.019139	-0.019139	0.054203	0.054203	-0.054203	-0.054203	-0.054203

Slika 3.3: Tablica nizova aminokiselina nakon standardiziranja

Pozicija	Broj izuzetaka	Standardizirana vrijednost	Vrijednost izuzetaka	U kojim nizovima
6	1	-0.019139	52.24940191045314	765
7	1	-0.019139	52.24940191023518	894
8	2	-0.027072	36.93913913452995	48, 49
9	1	0.019139	-52.24940191045187	696
10	2	0.024339	-17.033030664208727	55, 1261
11	1	-0.019139	52.249401910447155	2538
31	3	-0.033162	30.155154341065657	1747, 2540, 2585
56	1	-0.019139	-52.24940191045159	576
91	4	0.038299	-26.110342778295607	68, 2719, 2720, 2721

Tablica 3.1: Tablica izuzetaka

U tablici 3.1 prikazani su izuzetci na nekim pozicijama u nizovima. Pozicije 6, 7, 8, 9, 10 odnose se na 5 faktora druge aminokiseline u poravnatim nizovima, pozicija 31 prvi je faktor sedme aminokiseline, itd. Naime, na svakoj poziciji postoje izuzetci. Iz trećeg stupca tablice možemo uočiti da su ti izuzetci u drugačijim nizovima proteina za svaku poziciju. Tako na primjer 6. pozicija, koja predstavlja prvi faktor druge aminokiseline u nizu, ima izuzetak u 765. nizu, a nijedna od ostalih pozicija nema izuzetak u tom nizu. Također, 7. pozicija ima izuzetak u 894. nizu, a nijedna druga pozicija nema izuzetak u tom nizu, itd. Funkcijom u Pythonu provjerena je tvrdnja da se za pojedinu poziciju ne ponavljaju nizovi u kojima su izuzetci. Za svaku su poziciju drugi nizovi.

Vrijednosti izuzetaka na pojedinoj poziciji razlikuju se od uzoračke sredine te pozicije. Broj izuzetaka na pojedinoj poziciji zanemariv je u odnosu na broj nizova u kojima promatramo tu poziciju. Zbog toga je uzoračka sredina pojedine pozicije približno jednaka vrijednosti koja je izuzetak, a na toj je poziciji, tj. $\overline{X} \approx X_i$, za većinu i. Dakle, u formuli za standardizaciju, $Z_i = \dfrac{X_i - \overline{X}}{S}$, za vrijednosti koje nisu izuzetci brojnik će biti ≈ 0. S obzirom na to da je većina vrijednosti na istoj poziciji jednakog iznosa, uzoračka standardna devijacija također je broj blizu nuli, $S \approx 0$. Dakle i brojnik i nazivnik su brojevi blizu nule. Nakon standardizacije po pozicijama, vrijednosti koje nisu izuzetci dobivaju vrijednosti blizu nuli, tj. $Z_i \approx 0$. Znači da je brojnik ipak bliže nuli nego nazivnik pa imamo:

$$Z_i = \frac{X_i - \overline{X}}{S},$$

$$X_i - \overline{X} \approx 0, S \in \langle -1, 1 \rangle, X_i - \overline{X} \ll S\,.$$

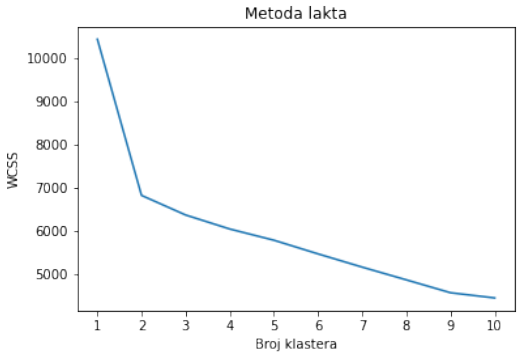
Za izuzetke vrijedi nešto drugačije. Nakon standardizacije dobivamo vrijednosti izuzetaka koje su velike, a da bismo to dobili, brojnik mora biti veći od nazivnika. Za nazivnik, odnosno uzoračku standardnu devijaciju, S, napomenuli smo da je ona broj blizu nuli s obzirom na to da je većina vrijednosti na istoj poziciji jednaka. Međutim, sada brojnik u $Z_i = \dfrac{X_i - \overline{X}}{S}$ nije više ≈ 0 kao prije jer se izuzetci razlikuju od uzoračke sredine. Zato jer je nazivnik ≈ 0, slijedi da je $|Z_i| > 1$:

$$Z_i = \frac{X_i - \overline{X}}{S}$$

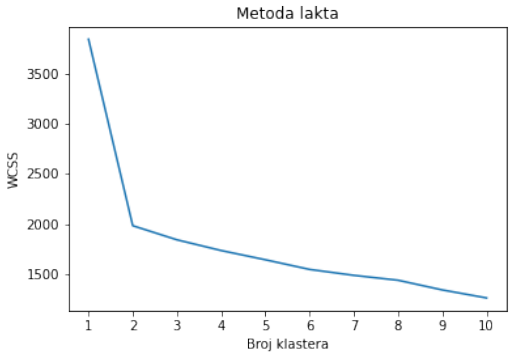
$$X_i - \overline{X} \in \langle -1, 1 \rangle, S \approx 0, S \ll X_i - \overline{X}.$$

Dobiven je suprotan efekt od očekivanog. Standardizacija podatke obično skupi, a u ovom slučaju ih je raspršila u svim smjerovima. Do toga je došlo zbog izuzetaka koji svojim vrijednostima odskaču od ostalih vrijednosti i to u različitim smjerovima jer su oni u različitim

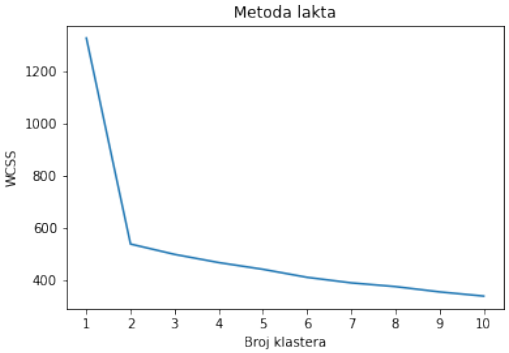
nizovima na različitim pozicijama. Tako su se podaci izmiješali pa ih ne možemo klasterirati. Kada bismo dodali 1-icu (ili neki veći broj) standardnoj devijaciji, $Z_i = \frac{X_i - \overline{X}}{S + 1}$, kod izuzetaka ne bismo više broj različit od nule dijelili s brojem približno nula, nego s brojem većim od 1. Tako možemo izbjeći odskakanje vrijednosti izuzetaka. Pogledajmo metode lakta dobivene za standardizirane podatke uz izmjenu S u S+1, S+2, S+4, S+6 i S+8.



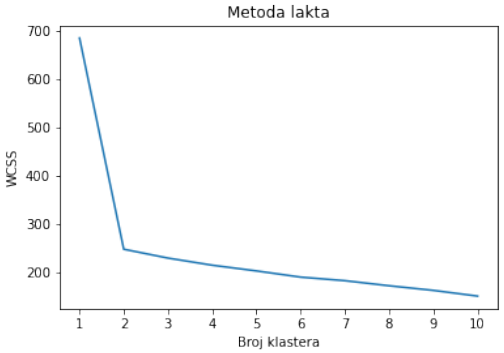
Slika 3.4: Metoda lakta na standardiziranim podacima uz S + 1



Slika 3.5: Metoda lakta na standardiziranim podacima uz S + 2

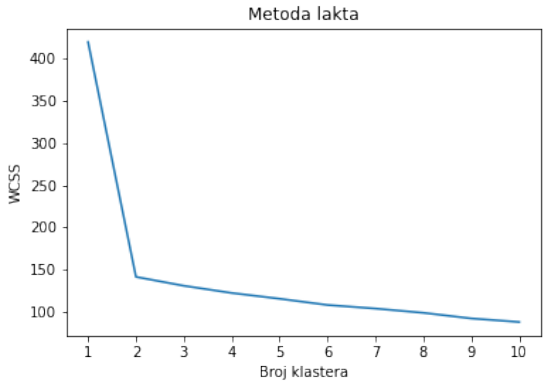


Slika 3.6: Metoda lakta na standardiziranim podacima uz S + 4



Slika 3.7: Metoda lakta na standardiziranim podacima uz S + 6

Iz grafa 3.4 slijedi da dodavanjem 1-ice uzoračkoj standardnoj devijaciji rješavamo problem sa grafa 3.1. Dodavanjem broja 2 standardnoj devijaciji lakat postaje oštriji, kao i dodavanjem 4, 6 i 8. Lakat pokazuje da je najbolji izbor za broj klastera k = 2. Dakle, prilagodbom podataka ne brišemo lakat koji je dobiven na originalnim podacima. Time je generalizirana metoda lakta i tehnikom standarizacije podataka ponovljeni su rezultati dobiveni na originalnim podacima u diplomskom radu [7].



Slika 3.8: Metoda lakta na standardiziranim podacima uz S + 8

Iz grafova, koji prikazuju metodu lakta za različite pribrojnike koji su pribrojeni uzoračkoj standardnoj devijaciji, mogu se uočiti približni postotci objašnjene varijance prelaskom s jednog na dva klastera. Postotci objašnjene varijance prelaskom s jednog na dva klastera navedeni su u tablici 3.2. Iz tablice 3.2, ali i iz grafova 3.7 i 3.8 vidimo da dodavanjem standardnoj devijaciji 6 ili 8 ne dobivamo veliku razliku u prononsiranosti lakta. Dodavanjem 6 ($\approx 1 - \frac{250}{700}$) ili 8 ($\approx 1 - \frac{145}{420}$) otprilike 65 % varijance objasni se prelaskom s jednog klastera na dva. To znači da daljnjim dodavanjem pribrojnika, nakon što smo dodali 6, ne dobivamo značajno oštriji lakat. Zaključujemo da je dovoljno uzeti S + 6.

S + ___	postotak objašnjene varijance
S + 1	$\approx 35\%$
S + 2	$\approx 50\%$
S + 4	$\approx 60\%$
S + 6	$\approx 65\%$
S + 8	$\approx 65\%$

Tablica 3.2: Postotak objašnjene varijance prelaskom s jednog na dva klastera

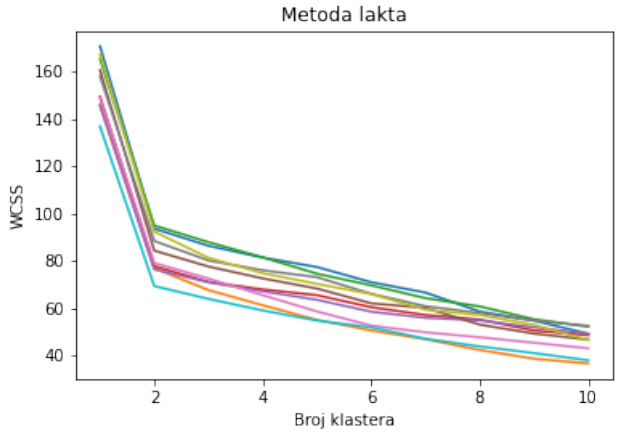
Izuzetke iz tablice 3.1 usporedimo s iznosima nakon nove standardizacije (S + 6 umjesto S). Oni su zapisani u tablici 3.3. Uočimo da sada te vrijednosti više ne odskaču kao ranije. Iz toga, a i lakta koji je na grafu vidljiv za isti k kao i u originalnim podacima, zaključujemo da smo sačuvali karakter podataka u novoj metodi.

Stabilnost podjele na dva klastera provjerimo tako da 10 puta uzmemo slučajni uzo-

Pozicija	Standardizirana vrijednost	Izuzetci	Izuzetci nakon S + 6
7	-0.019139	52.24940191023518	0.00216578351268354
9	0.019139	-52.24940191045187	-0.06938227889802291
11	-0.019139	52.249401910447155	0.060407912944636405
31	-0.033162	30.155154341065657	0.012647444656919042
56	-0.019139	-52.24940191045159	-0.20032188338908385
91	0.038299	-26.110342778295607	-0.09864724031491745

Tablica 3.3: Tablica izuzetaka nakon standardizacije sa S + 6

rak od 1000 nizova (2731 ukupno) i za svaki uzorak provedemo metode lakta. Iz slike 3.9 vidimo da je podjela na dva klastera stabilna nakon standardizacije podataka sa S + 6 umjesto uzoračke standardne devijacije, S. Iz stabilnosti možemo zaključiti da se standardiziran S protein grupira u dva klastera.

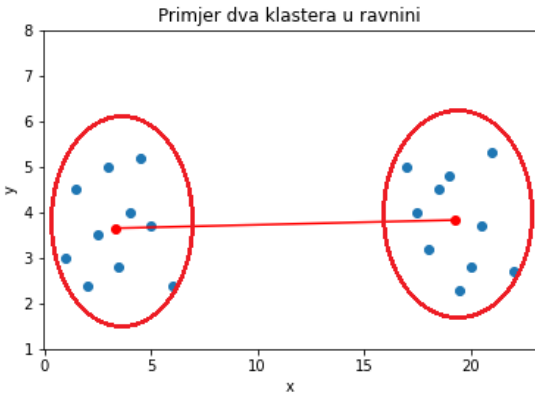


Slika 3.9: Stabilnost metode lakta za standardizirane podatke uz S + 6

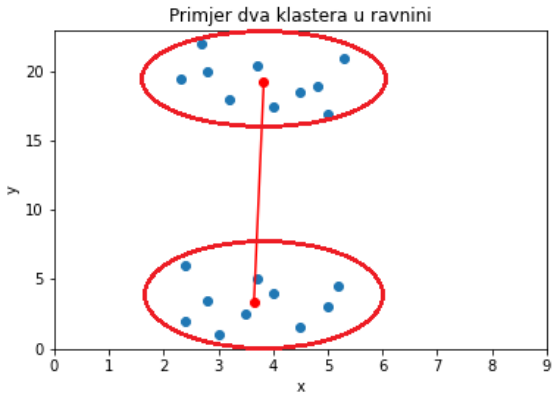
3.2 Otkrivanje značajnih pozicija za klasteriranje uz pomoć geometrije

Promatrajući dvodimenzionalan prostor i dva klastera u njemu, na temelju geometrijskog prikaza ta dva klastera intuitivno možemo zaključiti koja je koordinata značajna za klasteriranje, odnosno po kojoj koordinati se dijele na jedan i drugi klaster. Ako su u ravnini

klasteri jedan pored drugog, razlikuju se po prvoj koordinati (x), a ako su jedan iznad drugog, razlikuju se po drugoj koordinati (y). Ako napravimo spojnicu između centara tih dvaju klastera (dužinu $\overline{c_1c_2}$) i promatramo koordinate kao razlike koordinata centara $((c_{1,1} - c_{2,1}, c_{1,2} - c_{2,2}))$, najveća koordinata spojnice po apsolutnoj vrijednosti bit će upravo ona po kojoj se podaci dijele u dva klastera. Dakle, bit će najznačajnija za klasteriranje. Na slici 3.10 crvenom točkom nacrtani su centri klastera. Ako ih spojimo, spojnica će biti gotovo paralelna s x-osi. Znači da će x koordinata spojnice biti najveća po apsolutnoj vrijednosti, a y koordinata gotovo zanemariva. Analogno možemo zaključiti za spojnicu centara sa slike 3.11 gdje je najveća koordinata spojnice y koordinata.



Slika 3.10: Dva klastera koji se razlikuju po x koordinati



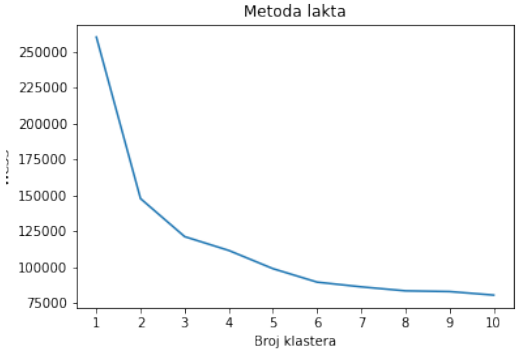
Slika 3.11: Dva klastera koji se razlikuju po y koordinati

Vodeći se navedenim, napravimo spojnicu dvaju centara klastera, odnosno gledamo apsolutne razlike po koordinatama centara i pogledamo koja je koordinata spojnice najveća. Dobivamo da je najveća koordinata 416., a to je 84. aminokiselina u pozicijama u kojima je došlo do mutacije, a 614. promatramo li sve pozicije. To je pozicija s D-G mutacijom. Ta je mutacija prva važna mutacija spike proteina jer je među prvim mutacijama. Dobiven je isti rezultat kao i u diplomskom radu [7], a na jednostavniji način. U tom diplomskom radu za svaku poziciju bio je promatran omjer varijance jednog klastera i zbroja varijanci svakog od dva klastera. Na poziciji gdje je omjer najveći, tamo je podjela na dva klastera najviše pridonijela. Umjesto toga promatramo spojnicu i time zauzimamo manje memorije (jednostavniji izračun) i manja je mogućnost za numeričke greške.

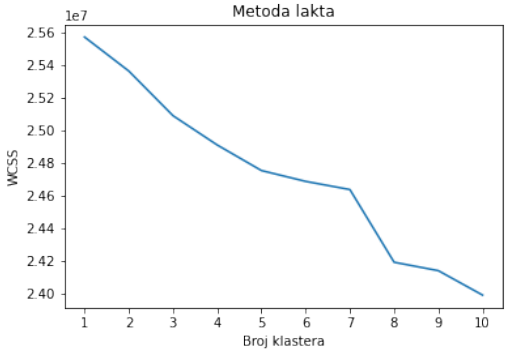
3.3 Metoda lakta na generaliziranim podacima

Uz prvotnu standardizaciju podataka s uzoračkom standardnom devijacijom S + 6, k-means algoritam primjenjujemo na generaliziranim podacima. Ti podaci imaju crticu koja je puno

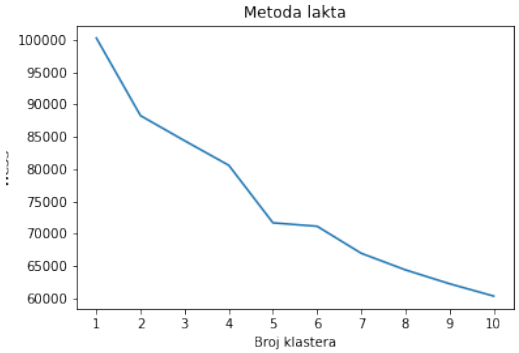
manje vjerojatna kod mutacije aminokiselina nego supstitucija. Sada radimo na 9988 nizova S proteina opisanih na stranici 14 u drugom poglavlju. Promotrimo grafove metode lakta za ove podatke.



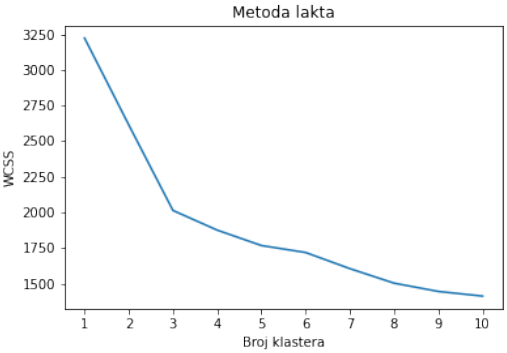
Slika 3.12: Metoda lakta na nestandardiziranim podacima



Slika 3.13: Metoda lakta na standardiziranim podacima



Slika 3.14: Metoda lakta na standardiziranim podacima uz S + 1

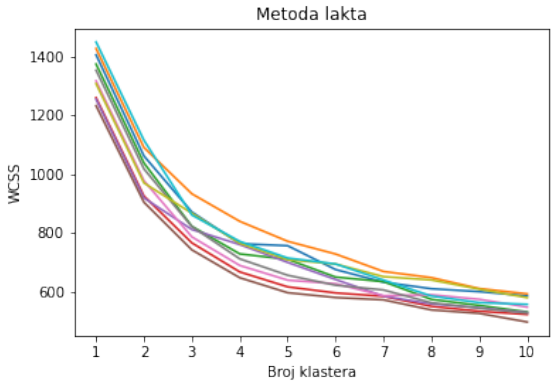


Slika 3.15: Metoda lakta na standardiziranim podacima uz S + 6

Ni bez standardizacije ni s njom ne možemo donijeti zaključke o potrebnom broju klastera u k-means grupiranju. Lakat na nestandardnim podacima može sugerirati 3 klastera, ali lakat nije oštar kao što nije ni na slici 3.2. Kod standardiziranih podataka metoda lakta može sugerirati 8 klastera. Usporedimo to s onime što dobivamo koristeći metodu standardizacije uz S + 6.

Na slici 3.15 lakat sugerira odabir 3 klastera. Na prethodnim podacima i slici 3.4 vidjeli smo da se već za pribrojen 1 standardnoj devijaciji nazire lakat. Ovdje je na grafu za S + 1 lakat vidljiv za odabir 5 klastera. S obzirom na to da se dodavanjem 1 ili 6

standardnoj devijaciji mijenja potreban broj klastera s 8 na 5 i 3 (slike 3.13, 3.14 i 3.15), a na nestandardiziranim podacima nema prononsiran lakat (slika 3.12), ne možemo zaključiti o značajnom broju klastera. Provjerimo stabilnost metode lakta za verziju standardizacije koju smo uveli na prethodnim podacima (uz S + 6).



Slika 3.16: Stabilnost metode lakta za standardizirane podatke uz S + 6

Iz stabilnosti metode lakta ne vidimo nikakvu jasnu podjelu na neki broj klastera. Zbog toga ne ćemo raditi grupiranje podataka k-means algoritmom. Boje grafova na slici 3.16 pokazuju grafove dobivene za različite uzorke i vidimo da se ti grafovi bitno razlikuju oblikom. To pokazuje da metoda lakta nije stabilna. Dakle, k-means klasteriranje bit će loše na ovom uzorku i nema ga smisla raditi. Ono što možemo promatrati i bez grupiranja su interesantne pozicije.

3.4 Standardnom devijacijom do interesantnih pozicija

Promatramo li standardne devijacije po pojedinom stupcu, očekujemo da će najveća standardna devijacija biti u onom stupcu gdje je najveća varijabilnost, odnosno gdje je najviše mutacija. Vodeći se time, očekujemo da ćemo rangiranjem standardnih devijacija po stupcu dobiti interesantne (značajne) pozicije za k-means klasteriranje. Uzmemo li standardnu devijaciju po zbroju po pet stupaca, dobit ćemo standardnu devijaciju za pojedinu aminokiselinsku poziciju u nizu. Rangirajući te standardne devijacije od najveće prema najmanjoj, dobivamo sljedećih najvećih 10:

Na web-stranicama [1] i [2] nalaze se popisi nekih značajnih mutacija spike proteina. Provjerimo jesu li neke od tih mutacija pozicije s najvećom standardnom devijacijom.

Pozicija	Standardna devijacija
477	35.692661
70	32.436773
253	32.435409
69	32.133859
614	31.843903
145	31.657257
138	31.026909
4	30.944608
142	30.854191
780	30.844337

Tablica 3.4: Tablica 10 najvećih standardnih devijacija po poziciji

477. pozicija	
S	6820
N	3162
I	5
R	1

70. pozicija	
V	9933
-	50
F	3
I	2

253. pozicija	
D	9690
G	291
V	7

69. pozicija	
H	9925
-	50
Y	9
P	3
R	1

614. pozicija	
D	9821
G	167

145. pozicija	
Y	9964
-	22
H	2

Navedene tablice s frekvencijama aminokiselina predstavljaju najvećih 6 pozicija po standardnoj devijaciji. Sve njih, osim 614. pozicije, možemo pronaći u [1] i [2]. Pritom su 69. i 70. pozicija mutacije karakteristične za alfa varijantu koronavirusa. Uspjevamo ih naći među najvećim standardnim devijacijama unatoč činjenici da je frekvencija crtica samo 50, dok je frekvencija aminokiselina V i H preko 9900 od ukupno 9988.

Na web-stranicama [1] i [2] nalaze se prve 4 pozicije koje su najveće po standardnoj devijaciji. One su vidljive u tablicama na stranici 26. Pozicija 477. je očekivano prva po standardnoj devijaciji jer je frekvencija aminokiseline S jednaka 6820, a N 3162. Dakle, aminokiseline N ima otprilike pola koliko ima S pa je broj mutiranih velik na toj poziciji, a iz toga slijedi da je veća varijabilnost. Za ostale tablice nemamo veliku varijabilnost. Npr.

70. pozicija druga je po standardnoj devijaciji dok ima samo 53 mutacije unutar ukupno 9988 nizova. Iz tih brojki ponovno zaključujemo da je uzorak loš. Uzorak nije nepristran jer nije izbalansiran po zemljama. Također, nisu uzeti svi nizovi, nego su uzeti oni koji su stariji i prvi sekvencirani, znači pristranost je prema prvim sekvenciranim nizovima. Nismo uzeli cijeli dokument s nizovima nego prvih 9988 te se uzorak ne sastoji od nizova koji su sekvencirani u zadnjih nekoliko mjeseci. Oni su sekvencirani prije dvije godine i prije godinu i pola. Dakle, iako je uzorak loš, uspjeli smo u vrhu standardnih devijacija pronaći mutacije koje su izlistane kao neke od značajnih mutacija. Ova analiza otkriva da bi pozicije s velikom standardnom devijacijom mogle biti među značajnim pozicijama za klasteriranje.

Stabilnost rangiranja pozicija po standardnoj devijaciji

U tablici 3.5 vidimo prvih 6 pozicija s najvećim standardnim devijacijama.

	cijeli uzorak
(max std) 1.	477
2.	70
3.	253
4.	69
5.	614
(min std) 6.	145

Tablica 3.5: Pozicije s najvećim standardnim devijacijama

	prvi uzorak	drugi uzorak	treći uzorak	četvrti uzorak	peti uzorak
(max std) 1.	477	477	477	477	477
2.	70	70	253	253	70
3.	253	253	70	70	253
4.	69	69	145	69	69
5.	614	145	614	614	614
(min std) 6.	145	614	69	145	145

Tablica 3.6: Stabilnost pozicija s najvećim standardnim devijacijama

Stabilnost rangiranja pozicija po standardnoj devijaciji provjeravamo tako da uzmemo 5 puta uzorak od 3000 nizova (od ukupno njih 9988). Za svaki uzorak izračunamo standardne devijacije pojedine pozicije i rangiramo ih od najveće prema najmanjoj. U tablici

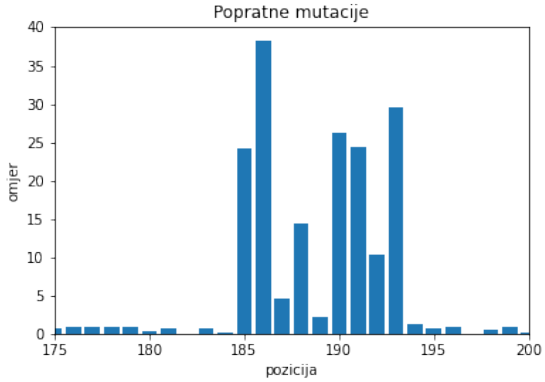
3.6 uočimo da su u prvih pet najvećih pozicija po standardnoj devijaciji uvijek iste pozicije (bez obzira na različitost uzoraka) što ukazuje na stabilnost rangiranja standardne devijacije pozicija.

Popratne mutacije

Ako je došlo do mutacije na jednoj poziciji, znači da se nešto promijenilo. Međutim, ako ta mutacija ima više popratnih mutacija, onda je veća promjena. Ako izgubimo popratnu mutaciju, onda ona nije važna, a ako se očuva, onda bi trebala biti važna. Kako bismo vidjeli funkcionira li računanje popratnih mutacija i na ovom generalnijem uzorku, a ne samo na urednom uzorku poput uzorka iz [7], podijelimo uzorak po nekoj od pozicija za koju smo pronašli da je značajna. Nakon podjele, provjerimo koje pozicije prate tu podjelu preko omjera. Omjer je statistika koja mjeri je li određena koordinata bolje opisana s jednim centrom ili s dva centra dviju grupa na koje smo podijelili uzorak. Omjer možemo promatrati po svakoj koordinati vektora ili po nizu od 5 koordinata (faktora), tj. po aminokiselini. Mi ćemo promatrati po faktorima.

$$O(j) = \frac{\sum_{i=1}^{br}(x_{i,j} - \overline{x_j})^2}{\sum_{k_1 \in K_1}(x_{k_1,j} - \overline{x_{j,k_1}})^2 + \sum_{k_2 \in K_2}(x_{k_2,j} - \overline{x_{j,k_2}})^2}, j = 1, 2, ..., l \tag{3.1}$$

U formuli (3.1) l je duljina vektora, br je oznaka za broj vektora koji su sudjelovali u k-means++ algoritmu, dok su K_1 i K_2 redom oznake prvog, odnosno drugog klastera. Budući da su 69. i 70. pozicija karakteristične i značajne mutacije za α varijantu COVID-a 19, pogledajmo imaju li one neke popratne mutacije. Promatramo li samo mutirane pozicije, 69. i 70. pozicija zapravo su 38. i 39. mutirana aminokiselina. Podijelimo nizove po te dvije pozicije, tj. u jednoj su grupi nizovi bez mutacija na pozicijama 38. i 39., a u drugoj su grupi nizovi s mutacijama na pozicijama 38. i 39. Dobivamo sljedeći prikaz omjera.

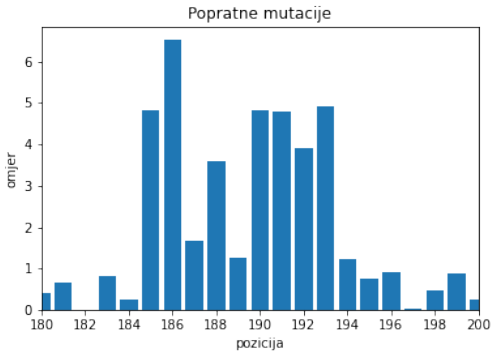


Slika 3.17: Popratne mutacije 38. i 39. pozicije

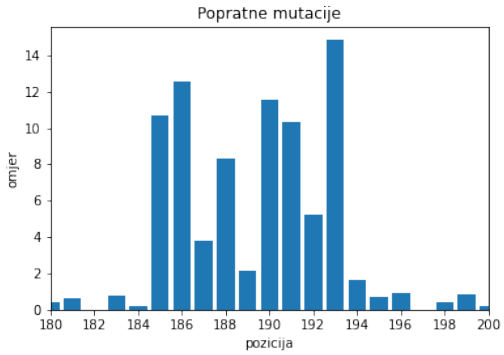
Ističu se stupići koji predstavljaju faktore 38. i 39. aminokiseline. Gledajući 5-dimenzionalne vektore, to su pozicije od 185. do 195. Te pozicije nemaju popratnih mutacija.

5-dimenzionalni vektori predstavljaju faktore. Faktori odgovaraju različitim svojstvima aminokiselina. Iz omjera vidimo da nije svako svojstvo aminokiseline jednako značajno. Odskok pojedinog faktora razlikuje se od drugih faktora. U 38. aminokiselini 2. faktor je značajan, dok je 5. faktor gotovo irelevantan. U 39. aminokiselini 4. faktor postaje puno značajniji nego 4. faktor 38. aminokiseline.

Mutacije 38. i 39. aminokiseline karakteristične su za α varijantu COVID-a 19, zato provjerimo jesu li one međusobno prateće. Podijelimo li sada uzorak po 38. aminokiselini ili po 39. aminokiselini, dobivamo omjere vidljive u 3.18 i 3.19.



Slika 3.18: Popratne mutacije 38. pozicije



Slika 3.19: Popratne mutacije 39. pozicije

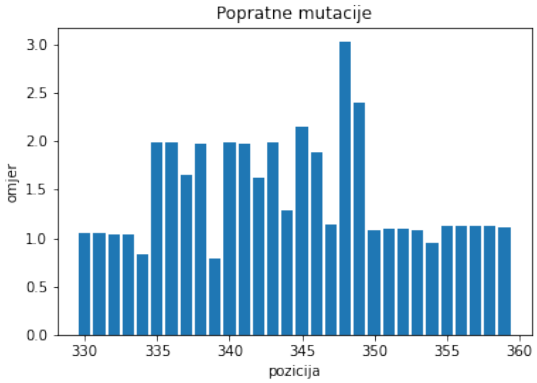
Zaključujemo da je mutacija 38. aminokiseline (185. do 190. pozicija promatrajući aminokiseline kao 5-dimenzionalne vektore) prateća mutaciji 39. aminokiseline (190. do 195. pozicija promatrajući aminokiseline kao 5-dimenzionalne vektore). Dakle, te mutacije često dolaze zajedno. Od ostalih pozicija, koje su najveće po standardnoj devijaciji, nisu pronađene popratne mutacije. Međutim, to ne znači da možda pozicija s malom standardnom devijacijom nema popratne mutacije.

Na web-stranici [1] ima popis značajnih mutacija spike proteina. Na primjer, 143. pozicija je po rangiranju standardnih devijacija 239. Provjerimo ima li ona popratne pozicije. Frekvencija te pozicije prikazana je u tablici 3.7.

143. pozicija	
V	9978
-	5
F	5

Tablica 3.7: Frekvencije 143. pozicije

Pogledajmo kakve omjere dobivamo dijeljenjem uzorka po toj poziciji. Napomenimo da promatramo omjere pozicija iz skupa mutiranih aminokiselina. U tom skupu je 143. aminokiselina zapravo 70. aminokiselina. Promatrajući aminokiseline kao 5-dimenzionalne vektore, 70. aminokiselina je vektor na pozicijama 345. do 349. Omjeri su prikazani na slici 3.20. Zaključujemo da 68. i 69. aminokiselina prate 70.



Slika 3.20: Popratne mutacije 70. pozicije

U skupu svih aminokiselina, 141. i 142. prate 143. aminokiselinu. Dakle, to su popratne mutacije. Frekvencije tih pozicija nalaze se u sljedećim tablicama.

141. pozicija	
L	9980
-	5
F	2
V	1

Tablica 3.8: Frekvencije 141. pozicije

142. pozicija	
G	9980
-	5
S	2
V	1

Tablica 3.9: Frekvencije 141. pozicije

Dakle, pronađena je pozicija koja ima popratne pozicije koje su značajne. Omjeri funkcioniraju na ovim primjerima ovog skupa podataka.

Bibliografija

[1] *Neke značajne mutacije spike proteina*, https://github.com/cov-lineages/constellations/blob/main/constellations/misc/spike_mutations.csv.

[2] *Značajne mutacije za alfa varijantu koronavirusa*, https://virological.org/t/preliminary-genomic-characterisation-of-an-emergent-sars-cov-2-lineage-in-t 563.

[3] D. Arthur i S. Vassilvitskii, *k-means++: The Advantages of Careful Seeding*, (2006), https://theory.stanford.edu/~sergei/papers/kMeansPP-soda.pdf.

[4] D. Bakić, *Linearna algebra*, Školska knjiga, 2008.

[5] M. Huzak, *Vjerojatnost i matematička statistika*, predavanja, Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, Matematički odsjek, 2006.

[6] N. Sarapa, *Teorija vjerojatnosti*, Školska knjiga, 2008.

[7] H. Tušek, *Analiza proteinskih nizova iz COVID-a 19*, diplomski rad, Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, Matematički odsjek, 2021.

[8] T. Šmuc, *Strojno učenje: Uvod u strojno učenje*, predavanja, Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet, Matematički odsjek, 2021/2022.

Sažetak

Tema ovog diplomskog rada je analiza S proteina iz koronavirusa. Tehnika strojnog učenja i statističke analize generaliziraju se na nizovima sa crticom.

Na početku su dani matematički pojmovi potrebni za razumijevanje termina koji su korišteni u radu i objašnjena je struktura podataka. Na pripremljenim podacima primjenjuje se k-means++ klasteriranje, traženje najznačajnijih pozicija za klasteriranje, rangiranje po standardnim devijacijama pojedine pozicije, traženje popratnih mutacija pomoću omjera.

Sve analize u diplomskom radu (grafovi, stupčasti dijagrami, tablice) napravljene su u programskom jeziku Python.

Summary

The topic of this thesis is the analysis of S protein from coronavirus. Machine learning techniques and statistical analysis are generalized on strings with a dash.

At the beginning, the mathematical concepts needed to understand the terms used in the paper are given and the data structure is explained. On the prepared data, k-means ++ clustering is applied, as well as searching for the most important positions for clustering, ranking according to standard deviations of individual positions, searching for accompanying mutations using ratios.

All the analyses in the thesis (graphs, bar charts, tables) are made in the Python programming language.

Životopis

Rođena sam 31. svibnja 1996. godine u Varaždinu. Školovanje započinjem u II. osnovnoj školi Varaždin. Nakon dvije godine započinjem i školovanje u Glazbenoj školi u Varaždinu, smjer violina. 2011. godine upisujem Prvu gimnaziju Varaždin, opći smjer te Srednju glazbenu školu u Varaždinu, smjer solopjevanje. Nakon završetka srednjoškolskog obrazovanja, 2015. godine upisujem Prirodoslovno-matematički fakultet u Zagrebu, preddiplomski sveučilišni studij Matematike. Po završetku preddiplomskog studija, 2018. godine, upisujem diplomski sveučilišni studij Matematičke statistike.