קמפוס טל

החוג לביואינפורמטיקה

פרוייקט גמר ביואינפורמטיקה - דו"ח סיכום

מקטעים שמורים בוירוסים

מגישה: חיה ברבולין
מנחים: ד"ר רונן חזן, ד"ר משה גולדשטיין, ד"ר אורי בריטברד

24/10/2018

תוכן עניינים

תקציר

בשנים האחרונות תחום המטא-דאטה מתפתח וניתן לו חשיבות רבה, ניתן לזהות את כל החיידקים הקיימים בדגימה בטכניקת PCR ע"י זיהוי רצף ה16S, שהינו חלק מרצף ה- RNA הריבוזומלי של החיידק. לעומת זאת בוירוסים אין עדיין אפשרות לזהות את סוגי הוירוסים הקיימים בדגימה באמצעות PCR היות שאין להם 16S. אולם, קיימות טכנולוגיות אחרות אך יש להן חסרונות משלהן. לכן, מחקר זה מתמקד במציאת קבוצת אוליגות היכולים לשמש כפריימרים לוירוסים שבעזרתם ניתן להגביר את כל הוירוסים בדגימה וכך לזהות אותם. בשונה מהגישה בה מצאו את הפריימרים בחיידקים (בחנו את הבעיה מהצד הביולוגי וכך זיהו את ה 16S ויחודיותו), במחקר זה ניגשנו לבעיה מהצד הביואינפורמטי, השתמשנו במידע הרב שהצטבר על רצפי הוירוסים וחיפשנו קבוצת אוליגות היכולים לשמש כפריימרים אוניברסליים עבורם. פיתחנו אלגוריתם שכקלט הוא מקבל את הוירוסים שאנו רוצים שיהיו עבורם פריימרים, וכפלט, האלגוריתם יוצר קבוצת אוליגות מינימלית שיכולים לשמש כפריימרים. כך, עבור כל וירוס שמתקבל בקלט, יש זוג פריימרים מקבוצת הפלט.

קיבלנו 255 פריימרים (כל אחד באורך 10 בסיסים) הפורשים 5,783 וירוסים.

מספר זה של פריימרים לא פרקטי לשימוש בטכניקת PCR, אך ניתן להשתמש בשלבים האחרונים של האלגוריתם על מנת למצוא פריימרים לתת-קבוצה של וירוסים, על פי רצון המשתמש, וכך לקבל קבוצת פריימרים קטנה המכסה את הוירוסים הרצויים וניתנת ליישום ב-PCR. תחום זה מעניין ופותח אפשרויות חדשות, היות שעד עכשיו בטכניקת PCR התאפשר לזהות מעט וירוסים באמצעות בדיקה בודדת, במידה ושייכים לאותה משפחה. כאן ישנה אפשרות לזיהוי וירוסים שאין להם קשר משפחתי.

ישנן מספר אפשרויות מעניינות להמשך מחקר בתחום זה, ביניהן לבדוק האם קיים קשר ביולוגי כלשהו בין וירוסים הנפרשים ע"י זוג פריימרים מסוים. בנוסף, ניתן לשכלל את תוצאות האלגוריתם ע"י קיבוץ פריימרים דומים עם mismatch נמוך ובכך להביא לקבוצת פריימרים קטנה יותר. מומלץ לקחת את תוצאות הפריימרים מהאלגוריתם להמשך מחקר במעבדה ולבדוק התאמה להגברת קטעי וירוסים ב-PCR.

הקדמה

זיהוי סוג הוירוס (virus) הוא נושא מרכזי במחקר בשנים האחרונות. החשיבות רבה היות וזיהוי סוג הוירוס מאפשר אבחנת מחלות אצל מטופל ובחירת טיפול התואם להדבקה, נותנת הבנה רחבה ועמוקה על המתרחש אצל הבריא והחולה, מאפשרת חקירת מחלות ופיתוח תרופות מתאימות, ואפילו פיתוח טיפול למחלות השונות באמצעות וירוסים.[3]

אבחון אוכלוסיית הוירוסים בדוגמאות של חולים ובריאים מאפשרת הבנה של רמות הוירוסים הנורמליות בפרט בריא, והבנת חוסר האיזון הנגרם בעקבות חולי, לאו דווקא ממקור נגיפי ושאיפה להחזרת האיזון עבור החולה.[2]

זיהוי סוג החיידק מתאפשר באמצעות ה - 16S, שהינו חלק מרצף ה - RNA הריבוזומלי של החיידק. בדיקת רצף שמור זה מתאפשרת בעזרת טכניקת PCR (Polymerase Chain Reaction), ע"י פריימרים (primers) ספציפיים ניתן להגביר קטעים קצרים מתוך רצף החיידק ובכך להגיע לזיהוי וכימות של החיידק[9]
.

היות ולוירוסים אין, 16S צריך להתאים פריימרים מיוחדים עבור זיהוי של כל וירוס בנפרד.

כיום, כאשר רוצים למצוא וירוס בדגימה ע"י PCR, ניתן לחפש וירוס ספציפי לנוכחות חיובית או שלילית.[4]

החיסרון הוא שכאשר נרצה למצוא נוכחות של קבוצת וירוסים, נצטרך לעשות מספר רב של ניסויים, ולמצוא עבור כל וירוס בנפרד האם קיים בבדיקה או לא. תהליך זה דורש זמן רב, כוח אדם ומשאבים. הפתרון שהוצע הוא לפתח פרוטוקול בו ניתן לזהות את הוירוסים ע"י ניסוי בודד, כך שיוביל לקבלת תוצאות מהירות, וחסכון במשאבים.

הגישה הנפוצה לזיהוי וכימות וירוסים היא בעזרת PCR היות שהיא מאוד רגישה, מהירה וחסכונית. שיטה זו מסורבלת כאשר רוצים לנתח דגימה בקנה מידה רחב כגון זיהוי כל הוירוסים הקיימים בה, היות שבמקרה זה צריך לעשות את התהליך עבור כל וירוס בנפרד. לכן, מציאת דרך שתאפשר זיהוי מספר רב של וירוסים בניסוי בודד היא חשובה ומהווה מקפצה בנושא.

פתרונות חלקיים לזיהוי וכימות מספר רב של וירוסים שקיימים בספרות הם עבור וירוסים מאותה משפחה או מספר וירוסים בודדים שהצליחו להגביר אותם באמצעות multiplex PCR (שימוש בשיטת PCR להגברת מספר רצפי DNA שונים בו זמנית, תוך שימוש במספר פריימרים) [2]. פתרונות אלו לא מספקים במידה ומספר הוירוסים לא מאותה משפחה, או שהשילוב המדובר לא נחקר קודם לכן - אלו פתרונות ספציפיים למדי.[1]

הגישות הקלאסיות לזיהוי קיום וירוס בדגימה מתבססות על סינון הוירוס מהתמיסה, הוספת נוגדן ובדיקה אם היה התפוצות של וירוסים או נתינת מצע פלורוסנטי ובדיקה למושבות. החסרון, כמובן, שלא ניתן לבדוק כמה וירוסים בו זמנית, רמת הדיוק לא מאוד גבוהה והזמן הנדרש ארוך. [4]

בין השיטות הראשונות לריצוף היא שיטת סנגר (Sanger), שבה משתמשים בשני סוגי נוקלאוטידים עבור יצירת מולקולות חדשות, סוג אחד של נוקלאוטידים רגיל והשני מונע את המשך הארכת המולקולה ובכך מביא לסיומה. תוצרי ההגברה הם מקטעים מהרצף המקורי באורכים שונים. מריצים את המקטעים בג'ל ומקבלים פרקציות שונות, קוראים לפי הצבע (לכל נוקלאוטיד שחוסם יש צבע משלו) .

היתרון הוא שאפשר לרצף רצפים ארוכים, אך החיסרון הוא העלות היקרה והופעה של שיטות אחרות שלא פחות טובות.

שיטה נוספת היא ה next generation sequencing (NGS)[3] המאפשרת קבלת מידע רב –זיהוי וכימות ע"י ריצוף רנדומלי של חלקים קטנים הקיימים בדגימה רפואית או סביבתית . היתרונות הם שלא צריך מידע מוקדם, וניתן לזהות וירוסים לא מוכרים בדרך זו, ואף ניתן לממש על כל סוגי הרצפים דו גדילי וחד גדילי. לדוגמא, MDA (Multiple Displacement Amplification) המזהה את כל הוירוסים שנמצאים בדגימה, אך תחום זה עדיין מתפתח ויש בעיות בשלב ההגברה שלעיתים יש בה סטיות[2], עדיין יקר יחסית ל PCR. ככל שהטכנולוגיה בתחום תתפתח יותר, והעלויות ירדו, השיטה תהיה יותר זמינה לשימוש במעבדות.[5]

המחקר המתואר בדו"ח הזה מתמקד בתת נושא בפיתוח שיטה לזיהוי מספר רב של וירוסים בבדיקה בודדת ע"י PCR.

כפי שקיימים פריימרים אוניברסליים בחיידקים (16S) [6], במחקר הנוכחי נחפש קבוצת רצפים יחודיים שיכולים לשמש כפריימרים אוניברסליים לוירוסים. כאשר יהיה מאגר של פריימרים אוניברסליים לוירוסים, יהיה ניתן לזהות את כל הוירוסים הקיימים בדגימה ע"י בדיקה בודדת בעזרת שיטת PCR בדומה לפרוטוקול בחיידקים.

בשונה מהדרך שבה מצאו את הפריימרים האוניברסליים בחיידקים (בחנו את הבעיה מהצד הביולוגי וכך זיהו את ה-16S ויחודיותו), בוירוסים, היות ואין להם 16S אין את זוג הפריימרים האוניברסליים ומתוך המידע על הרצפים לא מצאו זוג כזה. אך, יש זוגות פריימרים אוניברסליים עבור חלק ממשפחות הוירוסים אך זה לא מספק, במידה ונרצה לזהות וירוסים ממספר משפחות או וירוסים שעבור המשפחות שלהם עדיין לא זיהו פריימרים.

במחקר זה נקטנו בגישה הביואינפורמטית, ומתוך הרצפים של הוירוסים שהצטברו, חיפשנו קבוצת פריימרים אוניברסליים. פיתחנו אלגוריתם שכקלט הוא

יקבל את הוירוסים שאנו רוצים שיהיו עבורם פריימרים, וכפלט, האלגוריתם יצור קבוצת פריימרים מינימלית. כך, עבור כל וירוס שהתקבל בקלט, יהיה זוג פריימרים מקבוצת הפלט. מזה נובע שהפתרון לא יהיה תלוי במשפחות הוירוסים ויאפשר גמישות ומבחר עפ"י דרישות החוקר.

לצורך פיתוח הפרוטוקול, התמקדנו בבניית אלגוריתם המוצא קבוצת פריימרים מינימלית בוירוסים הנתונים כקלט.

מציאת קבוצת פריימרים אוניברסלית לוירוסים תאפשר מחקר מעמיק יותר ופיתוח פרוטוקול לזיהוי בו-זמני של כל הוירוסים הרצויים בדגימת מעבדה ע"י multiplex-PCR[2]. יתרונות השימוש ב-PCR פורטו לעיל ולכן אנו רוצים לאפשר את השימוש ב-PCR גם לזיהוי של הרבה וירוסים בבדיקה בודדת.

נגדיר:

א) אוליגו שפוטנציאלי לשמש כפריימר, הוא באורך 10-18 bp.

ב) מרחק בין זוג אוליגות היכולים לשמש כפריימרים, הוא 500-1500 bp (כדי לאפשר מקטע יחודי לכל וירוס).

מטרת המחקר:

בניית אלגוריתם שימצא את קבוצת הפריימרים המינימלית שפורשת קבוצת וירוסים נתונה (יכול להיות כל ה DB של הוירוסים או תת קבוצה מה DB).

שיטות

בכדי למצוא את קבוצת הפריימרים בוירוסים, השתמשנו ברצפים הקיימים בבסיס הנתונים refseq (DB refseq) [1]. בעזרת אלגוריתם שפיתחנו, מצאנו ב DB הקיים רצפים יחודיים העומדים בדרישות שלעיל (אורך האוליגו והמרחק בין זוג אוליגות), ולכן יכולים לשמש כפריימרים.

<u>השלבים</u>

1) מורידים את כל הרצפים של הוירוסים, המצויים בבסיס נתונים של NCBI (https://www.ncbi.nlm.nih.gov/), ב refseq
2) בשלב הראשוני נרצה למצוא, עבור כל מקטע קצר באורך X ,
מה מספר המופעים שלו במאגר. לצורך כך יוצרים מילון שמכיל את כל הפרמוטציות הקיימות עבור 20 בסיסים (קיימות 4^20)
היות שהיה מגבלת מקום, כוח וזמן ריצה, בחרנו מקטע באורך 10bp ונלקחו 33,227 פרמוטציות שהופקו מתוך רצף של וירוס.

$$\frac{33,227}{4^{10}} \sim 0.03$$

מה שנבחר הוא כ 3% מכלל הפרמוטציות, אך ראוי לציין שהפרמוטציות שנבחרו הן מתוך רצף קיים.
בנוסף, המילון הנ"ל הוא קלט לאלגוריתם, ובהינתן כוח חישוב וזמן, ניתן לתת כקלט את כל הפרמוטציות עפ"י רצון המשתמש.
מפעילים אלגוריתם שמחשב עבור כל פרמוטציה מספר המופעים ב DB
3) לוקחים את Z התוצאות עם המופעים הכי גבוהים (ערך הסף המדויק ייקבע בהמשך), ומרכיבים מתוכם תת-קבוצה הפורשת את ה DB

פירוט השלבים:

ניתן לראות את הקוד והתוצאות של כל שלב בקישור המצורף. עבור כל שלב יש תיקייה נפרדת לצורך הנוחות. [7]

<u>Stage 1: יצירת ה DB (נעשה באמצעות סקריפט בשפת python)</u>

חילצנו את רשימת ה ids מתוך קובץ של viruses הנמצא ב ftp ncbi בקטגוריה genome reports .

הורדנו את כל ה ids בעזרת ספריית entrez.

<u>זמני ריצה</u>

נגדיר a עבור כל החישובים

a - מספר הוירוסים ב DB : 8581

$$\theta(a)$$

[1] Reference Sequence (RefSeq) אוסף רצפים מקיף ללא כפילויות ומאומת, המכיל רצפי DNA, RNA וחלבונים

Stage 2: עיבוד ראשוני של ה DB

הקלט: DB של הוירוסים עבורם נדרש למצוא קבוצת אוליגות (אוליגונוקלאוטיד - oligonucleotide) [2] מינימלית

הפלט: קובץ csv הממפה, עבור כל פרמוטציה, מספר מופעים בוירוסים, באיזה וירוס ומיקום הפרמוטציה בוירוס.

הסבר האלגוריתם

הכנת תשתית:

יצירת X דוגמאות של אוליגות באורך K שמשמשים כבסיס לאלגוריתם, הוכנסו ככותרות של עמודות.

בהרצה שבוצעה, X = 33227, K = 10.

מילוי הטבלה:

עבור כל רצף שקיים ב DB:

שמו יהיה בשם השורה. נעבור על כל הקטעים באורך K שקיימים ברצף :

אם קטע זה קיים בבסיס אוליגות פוטנציאליים שהכנסנו לעמודות בהתחלה, נכניס את האינדקס של המקטע (מיקום תחילתו ברצף) בעמודה המתאימה. אם יש כמה אינדקסים, נכניס באותו תא עם תו " " (רווח) מפריד ביניהם.

סיכום תוכן הטבלה:

2 שורות סיכום:

1) עבור כל עמודה – אוליגו פוטנציאלי, נסכום את מספר המופעים הכולל; כלומר, אם עבור רצף מסוים יש 3 מופעים, נעלה את הספירה ב 3.
2) עבור כל עמודה – אוליגו פוטנציאלי, נסכום את מספר הרצפים שיש להם אינדקס בעמודה זו (אם יש לפחות מופע אחד ברצף T, נעלה את הספירה ב 1). זאת אומרת שאם קיבלנו Y, יש מופע של אוליגו זה ב Y רצפים ב DB.

הפלט של הסקריפט יהיה קובץ csv שבכותרות (ציר x) יש את רצף הפרמוטציה, כל שורה מייצגת רצף ב DB של הוירוסים. במידה ולוירוס זה יש מופע עבור עמודה x, יתבטא באינדקס של תחילת המופע. שתי השורות האחרונות מסכמות; חשוב לחדד שיש סיכום שבו מחשיבים כל מופע (בו נחשיב מספר מופעים באותו וירוס עבור אותה פרמוטציה) ויש סיכום שבו נסכום את מספר הוירוסים בהם פרמוטציה x מתקיימת.

זמני ריצה

בניית בסיס 4^20 הפרמוטציות

*היות שבדיקת מספר אקספוננציאלי של פרמוטציות דורשת זמן רב, הרצתי עם בסיס בגודל 33227

מספר הוירוסים ב DB : 8581

מילוי הטבלה: 1<i<8581, n = 33,227 (שהוא מספר האוליגות)

[2] אוליגונוקלאוטיד (Oligonucleotide) - מולקולות DNA/RNA קצרות, בדו"ח נכתב כאוליגו.

$$\theta(a * n * |\text{size of virus file } X_i|)$$

סיכום העמודות:

$$\theta(a * n)$$

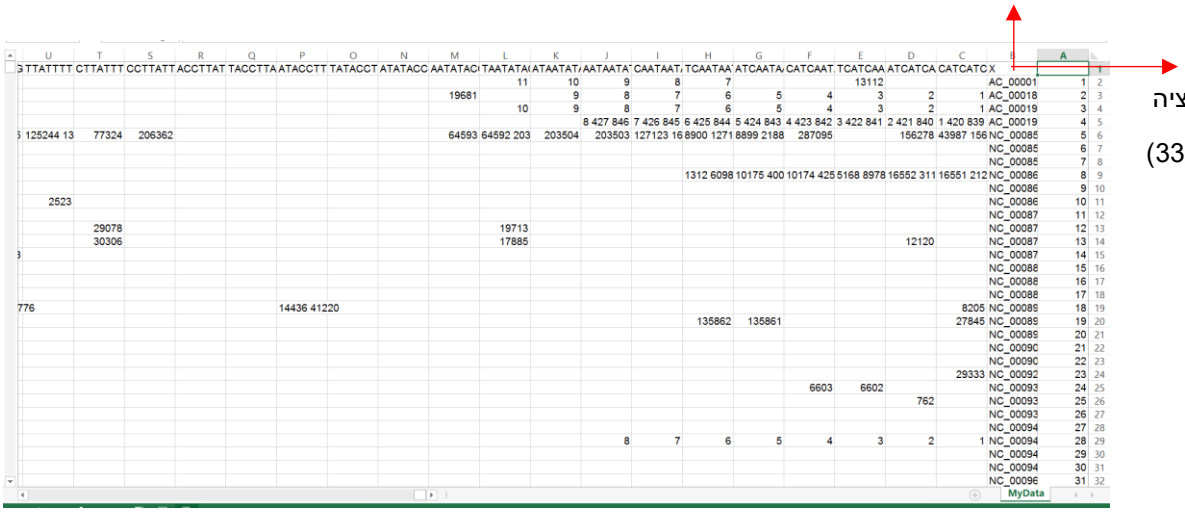
כפי שניתן לראות, שלב מילוי הטבלה הוא שלב שדורש זמן רב - עוברים על התוכן של כל ה DB ומסווגים לעמודה הנכונה. מכאן והלאה נתעסק באינדקסים ובמידע שיש בטבלה (המופק בשלב זה). שלב זה יחסוך פתיחת הרצפים וחיפוש בהם בהמשך, כל המידע הרלוונטי נכנס לטבלה.

תוצאות - הפלט

לפנינו צילום (איור 1) של חלק מקובץ הפלט (שבפורמט csv) בתצוגת excel - זהו צילום של תחילת הגיליון.

שם הרצף (יש 8581)

רצף הפרמוטציה (יש 33227)



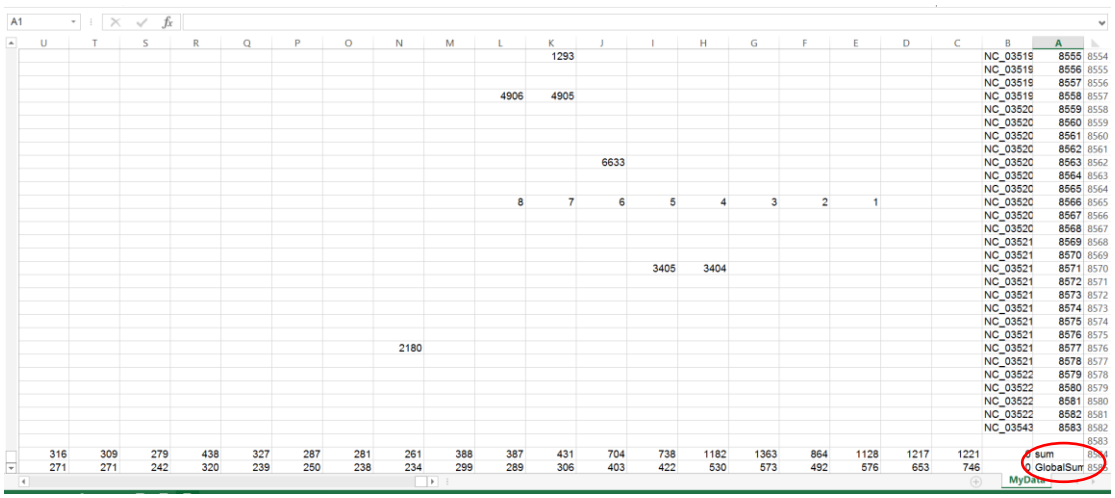
U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A	
3 TTATTTT	CTTATTT	CCTTATT	ACCTTAT	TACCTTA	ATACCTT	TATACCT	ATATACC	AATATAC	TAATATA	ATAATAT	AATAATA	CAATAAT	TCAATAA	ATCAATA	CATCAAT	TCATCAA	ATCATCA	CATCATC	X		
									11	10	9	8	7			13112			AC_00001	1	2
								19681		9	8	7	6	5	4	3	2	1	AC_00018	2	3
									10	9	8	7	6	5	4	3	2	1	AC_00019	3	4
											8 427 846	7 426 845	6 425 844	5 424 843	4 423 842	3 422 841	2 421 840	1 420 839	AC_00019	4	5
5 125244 13	77324	206362						64593	64592 203	203504	203503	127123 16	8900 1271	8899 2188	287095		156278	43987 156	NC_00085	5	6
																			NC_00085	6	7
																			NC_00085	7	8
													1312 6098	10175 400	10174 425	5168 8978	16552 311	16551 212	NC_00086	8	9
2523																			NC_00086	9	10
																			NC_00086	10	11
																			NC_00087	11	12
	29078								19713										NC_00087	12	13
	30306								17885								12120		NC_00087	13	14
3																			NC_00087	14	15
																			NC_00088	15	16
																			NC_00088	16	17
																			NC_00088	17	18
776					14436 41220													8205	NC_00089	18	19
													135862	135861				27845	NC_00089	19	20
																			NC_00089	20	21
																			NC_00090	21	22
																			NC_00090	22	23
																		29333	NC_00092	23	24
															6603	6602			NC_00093	24	25
																	762		NC_00093	25	26
																			NC_00093	26	27
																			NC_00094	27	28
											8	7	6	5	4	3	2	1	NC_00094	28	29
																			NC_00094	29	30
																			NC_00094	30	31
																			NC_00096	31	32

MyData

איור 1

בצילום הבא (איור 2) ניתן לראות את סוף הגיליון (אנכית):

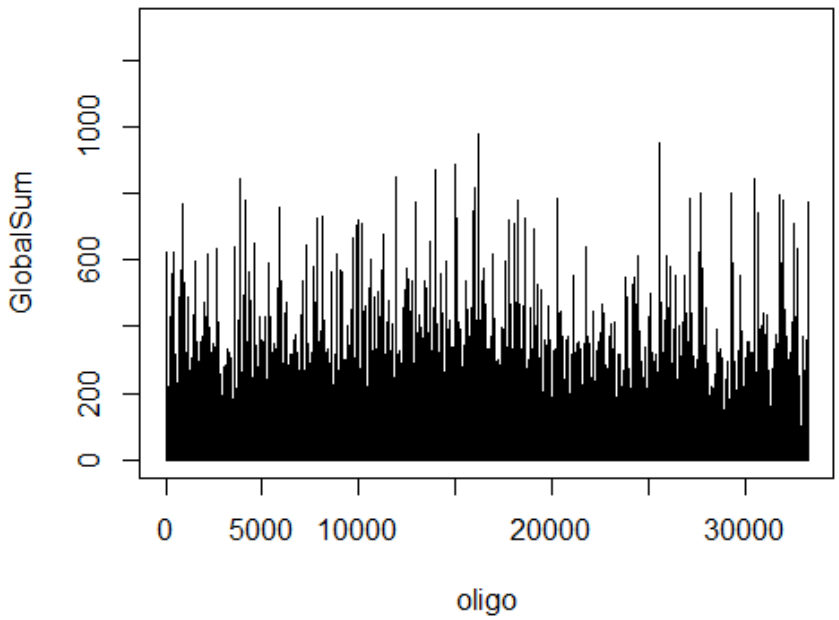
1) יש שורת סיכום sum שעבור כל עמודה סוכמת את כל מופעי הפרמוטציה כפי שפורט לעיל.
2) יש שורת סיכום GlobalSum שעבור כל עמודה סוכמת את מספר הרצפים בהם הפרמוטציה הופיעה כפי שפורט לעיל.



איור 2

לדוגמה: עבור הפרמוטציה המופיעה בעמודה C, יש לנו סה"כ 1221 מופעים בכל ה DB של הוירוסים. הפרמוטציה מופיעה ב 746 רצפי וירוסים; כלומר, בחלקם הפרמוטציה מופיעה יותר מפעם אחת.

בכדי להמחיש את התוצאות, יצרתי את ההיסטוגרמה שלהלן (איור 3):
ציר X = הפרמוטציות.
ציר Y = מספר הרצפים בהם הפרמוטציה מופיעה (שורת ה GlobalSum).
בציר ה X יש 33227 ערכים; ניתן לראות את ערכי הY משתנים בין 0 ל~1000.



איור 3

בעזרת ההיסטוגרמה ניתן לבחור ערך cutoff של ה globalSum , ובכך לצמצם את זמני הריצה של האלגוריתם שנעשה בהמשך.

לדוגמה: אוכל לבחור את הערך 200, ולנתח רק את הפרמוטציות שהמופעים עבורן גדולים מערך זה (טוב כי פרמוטציות עם ערך נמוך מופיעות במעט רצפים ולכן כנראה שלא נרצה לבחור בפרמוטציה זו לפריימר כי הדרישה שלנו היא קבוצה קטנה ככל האפשר שתפרוש את כל הרצפים הנתונים).

<u>Stage 3: הרכבת זוגות של פרמוטציות (אוליגות)</u>

קלט: קובץ csv מ stage 2.

פלט: קובץ csv המתאר זוגות של פריימרים פוטנציאליים ומופעים שלהם בוירוסים עם פירוט באיזה וירוס והמיקום ברצף. הזוגות עומדים בתנאים של מרחק בין אוליגו לאוליגו (בהרצה נבחר בין 500-1500bp)

<u>הסבר האלגוריתם</u>

יצירת קובץ עם פחות פרמוטציות היות שבשלב זה נרכיב זוגות והיות והנוסחה המתאימה היא

$$\frac{n * (n-1)}{2}$$

הערך עבור n=33227 הוא 552000151, כדי לאפשר את ההרצה צמצמנו את n ע"י cutoff = 600 , כפי שהוסבר לעיל.

בשלב זה ממשיכים עם הקובץ המצומצם שבו קיבלנו 257 אוליגות פוטנציאליים לפריימרים.

האלגוריתם עובר על תוכן הרשומה של כל וירוס, נזכיר שעבור כל וירוס יש רשומה עם האוליגות והאינדקסים שבהם מופיעים ברצף, האלגוריתם מרכיב את כל הקומבינציות של מופעי האוליגות. עבור כל קומבינציה האלגוריתם בודק האם המרחק בין זוג האוליגות עומד בתנאי הסף של מרחק בין זוג פריימרים, שהוא בין 500 ל 1500 . במידה ועמד בסף, האלגוריתם מכניס את זוג האוליגות למבנה שמחזיק את הרצפים של זוג האוליגות כשם עמודה, ואת שמות הוירוסים שיש להם מופע של זוג אוליגות זה כשם השורה, והאינדקסים שבהם מופיעים ברצף כתוכן.

לדוגמא:

שם עמודה: ATATATATAT-GCGCGCGCGC (זוג האוליגות הנבחר)

שם שורה: AC12345

ערך: (521,1521) 1000 (בערך יכול להיות רשימה של מפתחות וערכים מתאימים, במידה ויש יותר ממופע אחד של הזוג ברצף).

עפ"י הדוגמא, לוירוס AC12345 יש את האוליגו ATATATATAT באינדקס 521 ואת האוליגו GCGCGCGCGC באינדקס 1521 והמרחק ביניהם הוא 1000 בסיסים.

בשלב זה יש שימוש ב dataframe[8] , וכאשר יש במבנה 1000 שורות של וירוסים, הנתונים של המבנה מועברים לקובץ כדי לא להעמיס על הזיכרון של התוכנית בזמן הריצה.
סה"כ נקבל 9 קבצים עבור 8581 הוירוסים שיש לנו ב DB.

<u>מבנה הקובץ</u>:
כותרת של עמודות, אלו זוגות האוליגות.
כותרת של שורות, אלו שמות הוירוסים.
אם קיים צירוף של זוג אוליגות ספציפי עבור וירוס מסוים, במשבצת המתאימה, יהיה המרחק והאינדקסים בין האוליגות בוירוס זה.

<u>זמני ריצה</u>

n - מספר האוליגות שמתקבלות כקלט

$$a * O((n-1) * \frac{n}{2}) = a * O(n^2)$$

עבור הערכים שלנו נקבל $8581 * O(257^2)$

<u>תוצאות הפלט</u>
התקבלו 32,224 זוגות של אוליגות (יכלו להתקבל עד 32,385)

<u>Stage 4: בחירה סופית של אוליגות לפריימרים</u>

<u>קלט</u>: פלט של stage 3, 8 קבצי csv .

<u>פלט</u>: 3 קבצי txt כדלהלן :
descriptionRun = נתונים על הריצה; כל שורה היא סבב נפרד, ונתון מספר הפריימרים שיש בסיבוב זה וכמה וירוסים מכוסים ע"י פריימרים אלו.
האלגוריתם בנוי כך שבכל סבב מתווסף זוג פריימרים בודד.

finishViruses = עבור כל סבב שנעשה, האלגוריתם רושם זוג הפריימרים שנבחרו, מספר הוירוסים שהם מכסים, ופרוט שמות הוירוסים.

oligo_list = פרוט הפריימרים הבודדים שנבחרו (תצוגה לא כזוגות).

<u>הסבר האלגוריתם</u>
1) בשלב הראשון האלגוריתם מאחד את הקבצים לקובץ בודד וסוכם, עבור כל זוג אוליגות, את מספר המופעים.
2) בשלב השני, נעשית בחירת הפריימרים, תוך שימוש באלגוריתם חמדני [10] (בוחרים את האפשרות הטובה ביותר הנראית לעין בשלב הנוכחי). בכל סבב, האלגוריתם בוחר את זוג האוליגות עם מספר מופעים הכי גבוה, מכניס את הנתונים הרלוונטיים לקבצי הפלט, מסמן את שורות הוירוסים שכבר כוסו ע"י

פריימר שנבחר כלא רלוונטיות, סוכם שוב את מופעי זוגות האוליגות (בכל סבב יש פחות כי ישנם רשומות של וירוסים שכבר לא נכנסות לחישוב) וחוזר חלילה עד שלא נשארים וירוסים ללא סימון.

<u>זמני ריצה</u>

<u>שלב ראשון</u>:
a - מספר הוירוסים ב DB : 8581

n - מספר זוגות האוליגו שהתקבלו בשלב הקודם, אצלנו 32,224

$$O(a * n)$$

<u>שלב שני</u>:
סכימה של עמודות

$$O(a * n)$$

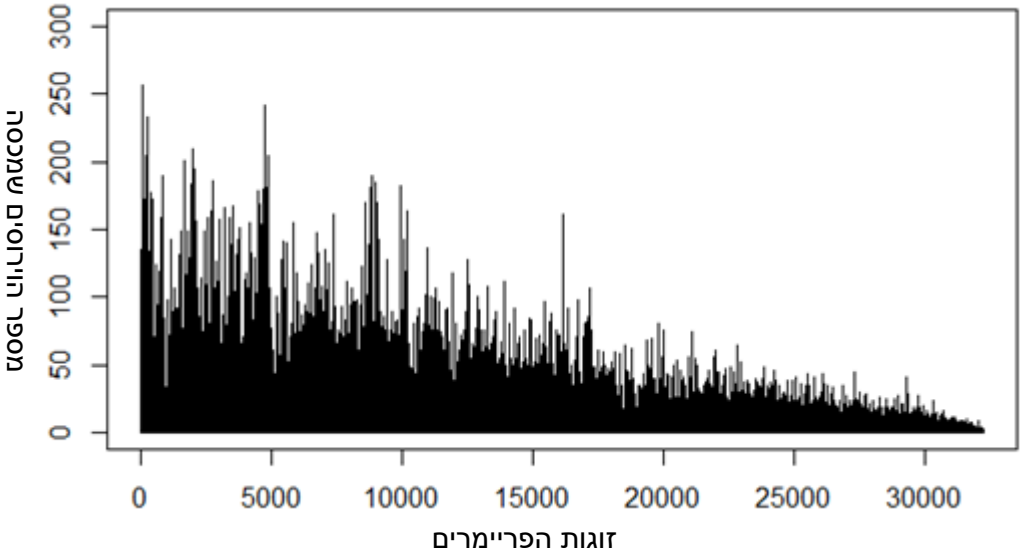
בחירת המקסימלי $O(1)$ (שומרים את המקסימלי כבר בחיפוש).
שמירת הנתונים הרלוונטיים $O(1)$ (טיפול בקבועים).
חזרה על כל התהליך עד x פעמים, כאשר x = מספר הרשומות ב DB, 8581

לכ ן:

$$O(a^2 * n)$$

תוצאות

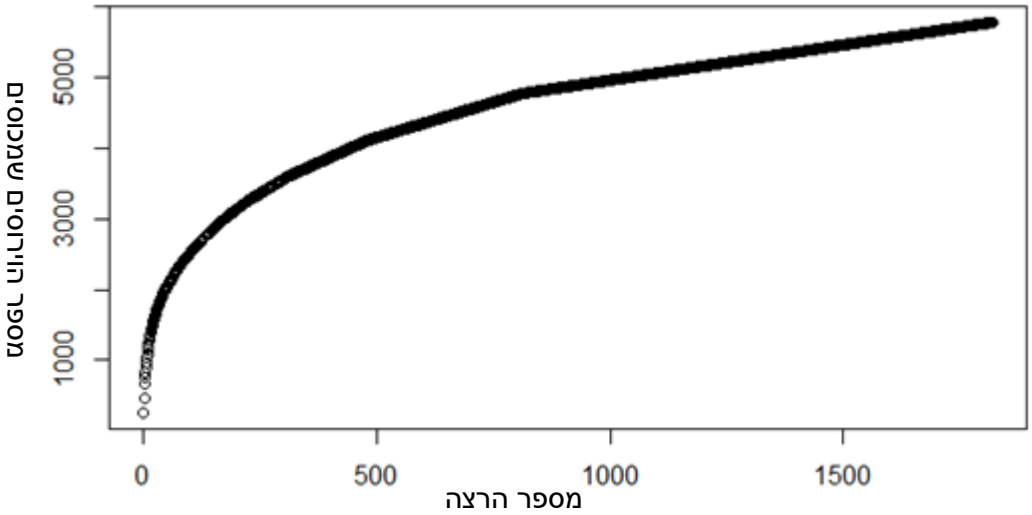
לפנינו היסטוגרמה (איור 4) המשקפת את הקשר בין זוגות האוליגו לוירוסים שהם מכסים.
ציר X: אינדקס של זוגות האוליגו עפ"י הסדר בקובץ הפלט של שלב 4.1.
ציר Y: מספר הוירוסים שזוג אוליגו מכסה.



איור 4

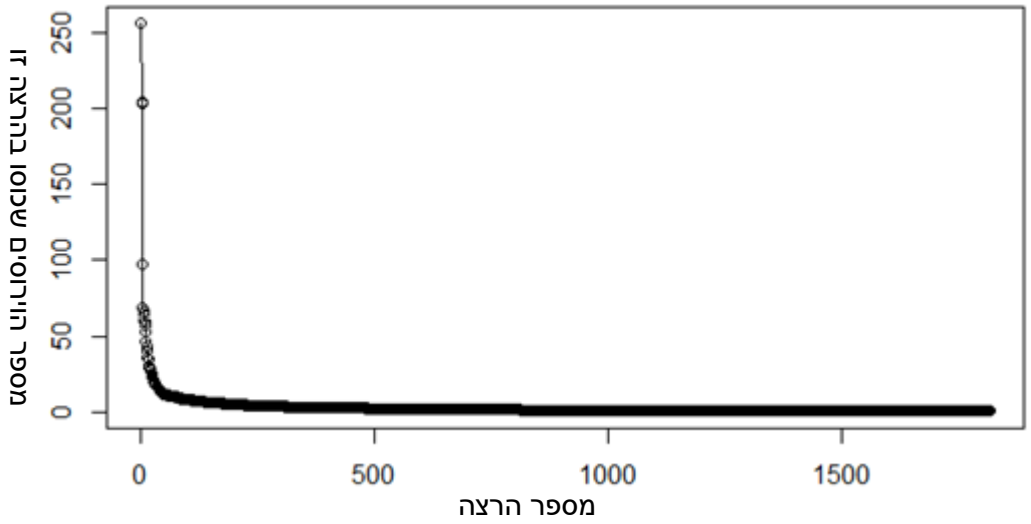
ניתן לראות שיש מעט זוגות עם פרישה של 200 עד 250 וירוסים . זהו המצב ההתחלתי של שלב 4.2 שבו יש אלגוריתם עם בחירה חמדנית.
ההיסטוגרמה מאפשרת את חיזוי התוצאות, היות שהפרישה של הזוגות דיי נמוכה, מספר זוגות הפריימרים שיבחרו יהיה גבוה בכדי להגיע לפרישת כל הוירוסים.

התקבלו 255 אוליגות שנבחרו לשמש כפריימרים, מספר הקומבינציות שחושבו בכדי לכסות את הוירוסים הם 1,820.
רצפי ה 255 הפריימרים שנבחרו נמצאים בקובץ oligo_list.txt בתיקיית שלב 4 [7].



איור 5

באיור 5, ש מתבסס על המידע בקובץ descriptionRun.txt, ניתן לראות את היחס בין מספר ההרצה וכמות הוירוסים שמכוסים עד לשלב זה של ההרצה. מספר ההרצה הוא גם מספר זוגות האוליגות שנבחרו היות שבכל הרצה נבחר זוג יחיד. אפשר להבחין בקלות שבהתחלה יש עליה חדה, היות שעבור כל זוג שנבחר יש הרבה וירוסים ובהמשך נבחרו זוגות שיש להם פחות וירוסים. הגרף משקף טוב את הבחירה החמדנית.



איור 6

איור 6 מחזק את הנקודה של איור 5.
שונה מאיור 5 בכך שעבור כל הרצה, בחירת זוג אוליגות, ניתן לראות כמה וירוסים זוג זה מוסיף. רואים שככל שמתקדמים בהרצות, מתווסף פחות ופחות, מהרצה ~200 נוספים עבור כל בחירה וירוסים בודדים, 1-5 וירוסים.

בקובץ finishViruses.txt יש פרוט שמות הוירוסים שנוספו בכל בחירת זוג אוליגות – בכל סבב באלגוריתם מידע זה יכול להיות מעניין במידה ורוצים להשתמש בחלק מהפריימרים ובכך לקבל פרישה חלקית של ה DB , ניתן להסיק בקלות אילו וירוסים בדיוק מכוסים ע"י חלק מהאוליגות.

דיון

כפי שהזכרנו בתוצאות, התקבלו 255 אוליגונוקלאוטידים שיכולים לשמש כפריימרים. כמות כזאת לא מעשית למימוש ב PCR בגלל מגבלות השיטה ואי כדאיות כספית. בכל זאת, ניתן לבדוק שימוש של חלק מהפריימרים שיתקבלו על מנת לקבל כיסוי חלקי של הוירוסים. מעבר לכך, מומלץ להריץ פעם נוספת את התוכניות על בסיס אוליגות גדול יותר; רצוי 4^{10} אוליגות פוטנציאליים כדי לקבל תוצאות יותר מדויקות, וכן ניתן לא לעשות cutoff ובכך להגדיל את זמן הריצה ולקבל תוצאות עוד יותר מדויקות.

דרך נוספת שכדאי לשקול כדי לצמצם את מספר הפריימרים שמתקבלים, אך לא לשלם במחיר זמן ריצה וזכרון, היא לקחת את כל הוירוסים שבמיוחד עבורם נוסף זוג אוליגו בודד שישמש רק אותם, ולעשות להם את כל התהליך מחדש עם בסיס אוליגות חדש – שונה ממה שעבדנו ולכן יש אפשרויות חדשות שויתרנו עליהם בהרצה הקודמת מעצם העובדה שלא לקחנו את כל הפרמוטציות אלא רק 3%.

לקבלת תוצאות מדויקות ומתאימות אישית לזיהוי וירוסים מקבוצת וירוסים רצויה, ניתן להריץ את חלק 4 בקוד על רשימת הוירוסים הרלוונטית (לוירוסים הלא רלוונטים נותנים סימון מיוחד בקובץ וכך נתעלם מהם). כך הבחירה החמדנית תהיה נכונה (לא רצוי להשתמש בתוצאות שנעשו על כלל ה DB, היות שהבחירה החמדנית החשיבה גם וירוסים שלא נרצה להחשיב במקרה של תת-קבוצה בוירוסים).

שלב נוסף לסינון האוליגות שהתקבלו כמתאימות לשמש כפריימרים, הוא בדיקה שהאוליגות שהתקבלו הם יחודיים לוירוסים ולא מופיעים ברצפים של חיידקים, אחרת בהגברה בשיטת PCR נקבל הגברה של קטעים לאו דווקא ממקור נגיפי.

האפשרות של קבלת פריימרים בהתאם לתת קבוצה עפ"י רצון המשתמש, היא אפשרות מעניינת ויכולה להיות לעזר רב עבור חוקרים בתחום. לדוגמה, הכנסת קלט שהוא קבוצת הוירוסים הקיימים באדם ובכך לקבל קבוצת אוליגות שבעזרתה ניתן לזהות את הוירוסים הקיימים באדם, נקבל קבוצה יותר קטנה מהתוצאות הנוכחיות.

להמשך המחקר ניתן לייצר תוכנית המקבצת אוליגות דומים עם mismatch קטן (לדוגמא 1 או 2) ובכך להוריד את מספר הפריימרים הסופיים.

נקודה נוספת שיהיה מעניין לבדוק היא האם יש קשר ביולוגי בין וירוסים שקיבלנו עבורם זוג אוליגות מסוים משותף, והאם התוצאות מחלקות במידה מסוימת את הוירוסים עפ"י המשפחות שלהם.

במידה ורוצים להשתמש בחלק מהפריימרים לזיהוי וירוסים, צריך לבדוק שהחלק בין הפריימרים משתנה בין וירוס לוירוס בכדי שיהיה תועלת בשיטת PCR.

תוצר חשוב של עבודה זו הוא הקוד עצמו, היות שניתן לשנות בקלות פרמטרים וקלט ולקבל תוצאות מותאמות לדרישות שונות, וכך כל מעבדה או חוקר יוכלו להתאים את מחקר זה לצרכיהם.

ביבליוגרפיה (REFERENCES)

[1] Elfath M. Elnifro, Ahmed M. Ashshi, Robert J. Cooper and Paul E. Klapper. Multiplex PCR: Optimization and Application in Diagnostic Virology. Clinical Microbiology Reviews. 2000 Oct; 13(4): 559–570.

[2] Linlin Li, Xutao Deng, Edward T. Mee, Sophie Collot-Teixeira, Rob Anderson, Silke Schepelmann, Philip D. Minor, and Eric Delwart. Comparing viral metagenomics methods using a highly multiplexed human viral pathogens reagent. J Virol Methods. 2015; 213: 139–146.

[3] Charles Y Chiu. Viral pathogen discovery. Curr Opin Microbiol. 2013; 16(4): 468–478.

[4] Sibnarayan Datta, Raghvendra Budhauliya, Bidisha Das, Soumya Chatterjee, Vanlalhmuaka, and Vijay Veer. Next-generation sequencing in clinical virology: Discovery of new viruses. World J Virol. 2015; 4(3): 265–276.

[5] Shea N Gardner, Crystal J Jaing, Kevin S McLoughlin, and Tom R Slezak. A microbial detection array (MDA) for viral and bacterial detection. BMC Genomics. 2010; 11: 668.

[6] J. Michael Janda and Sharon L. Abbott. 16S rRNA Gene Sequencing for Bacterial Identification in the Diagnostic Laboratory: Pluses, Perils, and Pitfalls. J Clin Microbiol. 2007; 45(9): 2761–2764.

[7] https://drive.google.com/open?id=1NaagBXh2-HkONl6hqtygmd1KIn-kk_e2

האלגוריתם שנכתב במסגרת עבודה זו

[8] pandas DataFrame docs

[9] Sanschagrin, S., Yergeau, E. Next-generation Sequencing of 16S Ribosomal RNA Gene Amplicons. *J. Vis. Exp.* (90), e51709, doi:10.3791/51709 (2014).

[10] https://www.encyclopediaofmath.org/index.php/Greedy_algorithm