

פרויקט סימולציה

בחירות 2021



תאריך הגשה: 6.6.21

מגישות:

עמית גבאי - 316088525

מאי דניאל - 313177073

נוי כשר – 314963810

תוכן עניינים

1. תקציר	3
2. מבוא	3
2.1 תיאור המערכת הנחקרת	3
2.2 מטרות - הגדרת מטרת הארגון, מטרת הסימולציה ומטרות הפרויקט שגיאה! הסימניה אינה מוגדרת.	
3. מודל סימולציה	4
3.1 תיאור המודל	4
3.2 הנחות	4
4. ניתוח המצב הקיים והצעת חלופות לשיפור המצב הקיים:	7
4.1 בחירת מדדים לניתוח (מטרת הסימולציה)	7
4.2 סוג המערכת	7
4.3 ניתוח מצב קיים ביחס למדדים ולמטרות (הפרויקט/ הארגון) ..	7
4.4 תיאור החלופות	7
4.5 השוואה סטטיסטית בין מצב קיים לחלופות המוצעות	12
5. מסקנות	12
6. נספחים:	13
נספח 1- נתונים	13
1.1 נתוני המערכת הנחקרת	13
נספח 2- המסלולים במודל	14
נספח 3- הישויות המשאבים והתכונות שהוגדרו במודל	21
3.1 הישויות שהוגדרו במודל	
3.2 המשאבים שהוגדרו במודל	22
3.3 התכונות שהוגדרו במודל	23
נספח 4- בדיקת נכונות המודל:	24
4.1 ישות מצביע	24
4.2 מנהל צפון / מרכז / דרום	28
4.3 משקיף צפון / מרכז / דרום	29
4.4 ישות פיקטיבית אשר סוגרת את הקלפיות	29
נספח 5- מבחני טיב התאמה	30
נספח 6- חלופות	34
6.1 חלופה מספר 1	34
6.2 חלופה מספר 2	35
6.3 כמות הבוחרים המגיעים לקלפיות	35
6.4 טבלת כמות הנוטשים בדוכנים	35
6.5 תיעוד קוד חלופות	37
נספח 7 – חישוב מספר ריצות, תיעוד קוד והסבר מילולי וכמותי	40
נספח 8- הפונקציות במודל הסימולציה	44

8.1 הפונקציות במצב הקיים 44

1. תקציר

בפרויקט התבקשנו למדל באמצעות תוכנת ה-R STUDIO את תהליך הבחירות לכנסת ה-25. מערך הקלפיות פרוס בכל רחבי הארץ: צפון, מרכז ודרום. הקלפיות מכילות את בעלי התפקידים השונים, כך שבכל קלפי, כיאה למדינה דמוקרטית המפלגות מציבות את נציגיהן בדוכנים מחוץ לקלפיות במטרה לנסות ולשנות את דעת הבוחרים ולהגדיל את סיכויי הזכייה של המפלגה. בנינו מודל סימולציה הממדל את אזורי המדינה בהן מתבצעות הבחירות, את מערך הקלפיות ומימוש זכות הבחירה ואת דוכני המפלגות הנמצאים מחוץ לקלפיות שמטרתן לשכנע את הבוחרים לבחור במפלגה שלהן. בחנו את נכונות הסימולציה ואת המצב הקיים. לאחר מכן, חקרנו כיצד המדדים משתנים בהתאם לחלופות שנועדו להעלות את סיכויי **מפלגת שמאל בה בחרנו לזכות**. בחלק זה של בדיקת החלופות, עמד לרשותנו תקציב של 30,000 ₪ ועל כן התמקדנו בבניית החלופות שיעזרו בצמצום המדדים וצווארי הבקבוק שעלו מניתוח נתוני הסימולציה.

החלופה הראשונה כללה העברת שני נציגים ממתחם מרכז למתחם הדרום ללא תשלום, הוספת נציג נוסף למתחם צפון, הוספת דוכן למתחם מרכז והעברת סדנה לנציגים ממתחם צפון לשיפור יכולת השכנוע שלהם. עלות החלופה הנ"ל היא 30,000 ₪.

החלופה השנייה כללה העברת סדנה לנציגים ממתחם צפון לשיפור יכולת השכנוע שלהם והשקעה בתעמולה תקשורתית ביום הבחירות. עלות החלופה הנ"ל היא 27,000 ₪.

לאחר הרצת המצב הקיים כנדרש, קיבלנו שבחלופה השנייה אחוז המצביעים למפלגה שלנו הוא 36%, וזאת לעומת 30% בחלופה הראשונה ו-27% במצב הקיים. כמו כן, מדד אחוז הנוטשים בדוכנים בחלופה השנייה עמד על 17%, לעומת 18% בחלופה הראשונה ו-20% במצב הקיים. ניתן לראות כי על פי כל המדדים, **החלופה השנייה היא הטובה ביותר**, וכן גם על פי מבחני t מזווג בין החלופות – נבחר להמליץ בסופו של דבר על החלופה השנייה כחלופה הדומיננטית והמועדפת.

2. מבוא

2.1 תיאור המערכת

בפרויקט זה נחקר את מערך הקלפיות בבחירת לכנסת ה-25. מדינת ישראל הינה מדינה דמוקרטית ועל כן לכל בוחר במדינה יש את אפשרות הבחירה החופשית להציע למפלגה כראות עיניו ועל-פי עקרונותיו.

הקלפיות פרוסות ברחבי הארץ כך שכל בוחר יכול להצביע במיקום הקלפי הקרובה אליו בדרום, בצפון או במרכז. הפריסה היא כזו שבצפון הארץ מוצבות 3 קלפיות, במרכז מוצבות 9 קלפיות ובדרום מוצבות 6 קלפיות, כאשר קיימים שני סוגי קלפיות - קלפיות אשר נפתחות בשעה 7:00 ונסגרות בשעה 22:00 וקלפיות אשר נפתחות בשעה 8:00 ונסגרות בשעה 20:00. קלפיות שזמן פעולתן הוא הגדול יותר הינן קלפיות נגישות בהן גם אנשים בעלי מוגבלויות יכולים להצביע.

כלל הנתונים שנוגעים לזמנים ולקצבי ההגעה של הנבדקים במערכת הנחקרת מפורטים **בנספח 1.1**.

כאשר הבוחר מגיע למתחם הקלפי יש בידיו דעה ראשונית לגבי המפלגה אליה יצביע. בדרך למימוש זכות הבחירה ומחוך לקלפי, עובר הבוחר בדוכני המפלגות השונות וזאת על מנת שניציגי המפלגות יוכלו לנסות ולשנות את דעתו. בין אם הנציגים הצליחו לשנות את דעת הבוחר ובין אם לא, מגיע הבוחר אל הקלפי ומחכה לתורו להצביע. בכניסת הבוחר לקלפי, ולאחר שעובר את הבירוקרטיה ההצבעתית (בדיקת תעודת זהות ומתן מעטפה ריקה), משלשל הבוחר את פתק המפלגה בה בחר וכך - מצביע ומשפיע! במידה והבוחר בעל תעודת נכה הוא מקבל עדיפות להיכנס לקלפי ואינו מחכה בסוף התור ככל הבוחרים.

בכל שלוש שעות, מגיע אחראי אזור, לכל אחד משלושת האזורים, על מנת לפקח שתהליך הבחירה מתנהל כראוי וכן כדי להעביר תדריך לעובדים בקלפי. בזמן זה תהליך הבחירה נעצר עד לעזיבתו של האחראי.

עם זאת, למרות עקרונותיה הדמוקרטיים של המדינה ולמרות הניסיון של האחראי לפקח על המתרחש, לא כולם מתנהלים בצורה ישרה ונציגי המפלגות לפעמים מנסים לנצל את היחס דעתם של מזכירי הקלפיות ולהוסיף קולות למפלגה שלהם. בתום יום הבחירות הארוך, ותוך השתוקקות רבה, נספרים קולות הבוחרים.

2.2 מטרות - הגדרת מטרת הארגון, מטרת הסימולציה ומטרות הפרויקט

מטרת הארגון - הינה לספק תהליך הצבעה נוח ויעיל בכל המשתמע בכך: יום שבתון ביום ההצבעה ותחבורה ציבורית חינם.

מטרת הפרויקט - המטרת הפרויקט הינה לעזור למפלגה למקסם את סיכויי הזכייה שלה בבחירות, ע"י בחינת כל החלופות המוצעות ובחירת החלופה הטובה ביותר.

מטרת הסימולציה - בניית מודל שידמה לנו את המציאות בצורה קרובה ככל הניתן לתהליך ההצבעה של הבוחר והדרך אל הקלפי.

3. מודל סימולציה

3.1 תיאור המודל

את בניית מודל הסימולציה ביצענו באמצעות תוכנת R-studio, על ידי שימוש בחבילת Simmer. את מידול פעילות תהליך יום הבחירות ביצענו באמצעות יצירת מסלולים (נספח 2), משאבים, ישויות ותכונות (נספח 3), כאשר התורים מתנהלת בצורת FIFO, מלבד הישויות שמייצגות את המצביעים בעלי המוגבלות, להם יש עדיפות בתור.

הרצף הלוגי שהישויות יוצרות:

לאורך כל זמן הסימולציה נוצרים שלושה סוגים של ישויות בוחרים:

- **מצביעים בקלפי קטנה** שנוצרים בזמן התפלגות אקספוננציאלית עם $\lambda = \frac{1}{1.11}$,
 - **מצביעים שאינם בעלי מוגבלויות והולכים לקלפי נגישה (קלפי גדולה)** הנוצרים בזמן התפלגות גמא עם $\alpha = 1.9688$ ו- $\lambda = 0.6994$.
 - **מצביעים בעלי מוגבלות והולכים לקלפי נגישה (קלפי גדולה)** הנוצרים בזמן התפלגות גמא עם $\alpha = 2.0235$ ו- $\lambda = 2.8359$.
- הישויות שמיועדות להגיע לקלפי שאינה נגישה (קטנה) מתחילות להיווצר עם פתיחת הקלפיות הנ"ל (בשעה 08:00) ומפסיקות להיווצר עם סגירת הקלפיות הנ"ל (בשעה 20:00), כאשר שאר הישויות נוצרות לאורך כל זמן פתיחת הקלפיות הנגישות (מהשעה 07:00 עד 22:00).
- כמו כן, נוצרים משאבים המייצגים את הקלפיות ואת דוכני המפלגות באזורים השונים. בחרנו למדל את יצירת המשאבים בצורה **מודולרית** על ידי מתן אינדקסים לכל קלפי או דוכן ספציפי, כאשר האינדקס הראשון מציין את האזור בארץ, האינדקס השני מייצג את מספר הקלפי והאינדקס השלישי מייצג את המפלגה (במידה ומדובר בדוכן). **(נספח 3.2)**.

מסלולים:

מסלול הבוחרים:

ראשית, כל ישות נשלחת למסלול ייעודי לה שמעניק לה את התכונה המעידה על איזה סוג קלפי היא דורשת - קלפי מסוג 1 (קלפי קטנה) או קלפי מסוג 2 (קלפי גדולה/נגישות) **(נספח 2)**. לאחר מכן, כלל הישויות נשלחות אל מסלול אתחול התכונות הראשוניות. במסלול זה נקבע לאיזה **אזור** הולכת הישות (1) מסמל את אזור צפון, 2 מסמל את אזור מרכז ו-3 מסמל את אזור דרום). בנוסף, כידוע, בכל אזור יש מספר קלפיות שונה, ועל כן בשלב הבא הישויות מקבלות תכונה המעידה על **מספר הקלפי** האזורי שהן הולכות אליו (בצפון יש 3 קלפיות, במרכז 6 ובדרום 9).

בהמשך הישויות מקבלות את תכונות המעידות על **ההסתברות** שיבחרו במפלגה מסוימת, כאשר דעה זו מחושבת כפי שהתבקשנו בהוראות (חישוב וקטור דעה התחלתי ושקלול מידת הקיצוניות של הבוחר המחושבת מתוך פונקציית צפיפות). **(נספח 8.1)**.

כמו כן, הישויות מקבלות תכונה שמכילה את **הערך המקסימלי** מווקטור הדעות וזאת על מנת שיהיה קל לדעת בהמשך איזה דוכן מפלגתי הישות לא נוטשת.

תכונות נוספות שמקבלות הישויות הן **סדר** הגעתן אל דוכני המפלגות. כידוע, בוחר צריך להגיע לדוכן של המפלגה אותה הוא מעדיף לאחר שעבר בשאר בדוכנים (אותם הוא מעדיף פחות), ועל כן על ידי תכונות אלו אנחנו מגדירים לבוחרים מה יהיה סדר ההגעה אל הדוכנים (1 מייצג את דוכן השמאל, 2 מייצג את דוכן המרכז ו-3 מייצג את דוכן הימין) **(נספח 3.3)**.

בסיום הגדרת כלל תכונות אלו, הישויות ממשיכות אל עבר דוכני המפלגות (שכידוע מוגדר לנו בתכונות). בדוכן הראשון, במידה והסתברות ההצבעה למפלגה אותה מייצג הדוכן זהה להסתברות ההצבעה המקסימלית בוקטור הדעה, הבוחר לא יוכל לנטוש את הדוכן. אחרת, תינתן אופציה

לנטישת הדוכן. בשני המקרים, הבוחר ינסה לתפוס את המשאב המתאים לאזור ולמספר הקלפי של הבוחר על ידי פונקציית `get_stand_resource` (נספח 8). בסיום השירות בדוכן בו נמצא הבוחר, במידה והבוחר לא נטש את העמדה, דעת הבוחר תשתנה בהתפלגות מוגדרת לטובת המפלגה אותה ייצג הדוכן. אחרת, השתנות הדעה של הבוחר תקרה גם כן אך לרעת המפלגה אותה ייצג הדוכן. בדוכן השני הדבר זהה להתנהלות הדוכן הראשון, אך בדוכן השלישי, משום שדוכן זה מייצג את המפלגה האהובה ביותר, אין לבוחר אפשרות לנטוש ועל כן הוא ישמע את דברי הנציגים וישנה את דעתו בהתאם. לסיום, הבוחר נכנס אל הקלפי ומחכה לתורו להצביע, כאשר בעת ההצבעה הבוחר מוסיף את קולו אל תכונה גלובלית, בהסתברויות הזרות להסתברות שהוגדרו לאחר מעבר הבוחר בדוכני המפלגות.

המסלול של אחראי האזור:

בסימולציה ישנם שלושה אחראי אזור, כל אחד אחראי לאזור אחר בארץ. האחראיים מגיעים למסלולי האזורים באינטרוולים של שלוש שעות החל מזמן תחילת הסימולציה (זמן 0) כאשר כל אחראי מגיע למסלול שלו, הוא סוגר את הקלפיות באזורו למשך זמן המתפלג משולשית, בזמן זה מבצע את התדריך עבור מזכירי הקלפי ולאחר מכן פותח אותן בחזרה ועוזב (עד ההגעה הבאה).

המסלול של המשקיפים:

בסיפור ישנם משקיפים עבור כל מפלגה, אשר רק אחד מהם, ובהסתברות מסוימת, מעלה את מספר הקולות של מפלגתו ב-50 קולות. בחרנו למדל זאת על ידי יצירת שלוש ישויות הנוצרות בשעה 22:00, אחת לכל אזור, אשר בהסתברות של 40% מוגרלת המפלגה שעבורה הישות מרמה ומעלה את מספר הקולות.

מסלול של סוגרי הקלפיות:

מעבר לעצירת מחולל הישויות של ציבור הבוחרים, נוצרות שתי ישויות פיקטיביות, אחת בזמן 20:00 ואחת בזמן 22:00 (ישות אחת לכל סוג קלפי), שתפקידן למעשה לסגור את הקלפיות. סגירת הקלפיות משמעותה להפוך את קיבולת הקלפי והתור אליו ל 0. במילים אחרות, הבוחרים לא יכולים להצביע לאחר סגירת הקלפי, פרט לבוחרים הנמצאים כבר בשירות.

3.2 הנחות

ההנחות לגבי המערכת הנחקרת שעליהן מתבסס מודל הסימולציה:

- בעת סגירת הקלפיות, במידה ובתוך הקלפי יש בוחר שטרם מימש את זכות הבחירה, הוא יסיים את להצביע ורק לאחר מכן הקלפי תיסגר. לעומת זאת, בוחרים הנמצאים בתור לקלפי לא יקבלו שירות ויעזבו.
- בזמן סגירת הקלפיות הגדולה/הקטנה, לא ביצענו אכיפה עבור סגירת הדוכנים.

- זאת מכיוון שלא ניתן לבצע הצבעה כלל ולמרות זאת אנשים נשארים בתור.
- כאשר מגיע בוחר בעל מוגבלות לקלפי, וכידוע הוא בעל עדיפות להיכנס לקלפי ומתקדם לראש התור, אך אם יש בוחר שאינו בעל מוגבלות שנכנס כבר לקלפי, הוא אינו יוציא אותו משירות באמצע, כלומר preemptive = False.
 - מנהל האזור יגיע אחת ל-3 שעות לקלפיות ומתחיל את הסבב בזמן 0, ויסגור אותן ללא תלות באיזה סוג קלפי זו.
 - מנהל אזור יסגור את כל הקלפיות באזור באותו הזמן, והוא לא יסגור את הדוכנים.
 - בדוכן של מפלגה מסוימת, במידה וההסתברות לבחור במפלגה זו זהה להסתברות המקסימלית של וקטור הדעה ההתחלתית, הבוחר אינו נוטש את הדוכן בו הוא נמצא.

4. ניתוח המצב הקיים והצעת חלופות לשיפור המצב הקיים]

4.1 בחירת מדדים לניתוח (מטרת הסימולציה)

- ראשית, ביצענו בדיקה הנותנת אינדיקציה על נכונות המודל ([נספח 4 – בדיקת נכונות המודל](#)), כלומר בדקנו שכל ישות אכן מבצעת את הפעולות שהיא צריכה לבצע ובסדר שהיא צריכה לבצע אותן. לאחר מכן, בהתאם לאופי הסימולציה, חשבנו על מטרות הפרויקט ועל פי מטרות אלו הגדרנו את המדדים הבאים:
- **הקטנת אחוז הנוטשים בדוכן** - תעמולת בחירות היא מרכיב משמעותי ומרכזי בתהליך של כל מסע בחירות. לכל מפלגה צריכה להיות האפשרות להציג בפני הבוחרים את עמדותיהם ולנסות לשכנע אותם להצביע עבור המפלגה, על כן נרצה שהבוחרים יישארו בעמדות המפלגות ושהסתברות ההצבעה שלהן למפלגה מסוימת תהיה כזו שמשקפת באופן מקסימלי את דעת הבוחר.
 - **הגדלת סיכויי הזכייה** - המטרה העיקרית של המפלגה היא, בסופו של דבר, לזכות בבחירות. נשאף, אם כן, שהמפלגה שלנו תהיה המפלגה הזוכה, ונעשה זאת על ידי בחירת מדד המייצג את אחוז הבוחרים במפלגה שלנו מתוך כלל הבוחרים ביום הבחירות.

תיאור מתמטי:

- **הקטנת אחוז הנוטשים בדוכן** – בחרנו להוסיף לישויות הבוחרים תכונה המתארת האם הם נטשו את הדוכן של מפלגת שמאל או לא (1 מסמל נטישה, 0 אחרת). בעזרת שאילתת SQL נוכל לעשות פרופורציה של מספר הנוטשים חלקי מספר המבקרים במפלגה שלנו.

שמירת מספר המבקרים בדוכנים של מפלגת שמאל:

```
visitors_left <- sqldf("select replication, count(resource) as voters_left
from full_arrival_data_resource")
```

```

where resource in ('stand_111', 'stand_121', 'stand_131',
'stand_211','stand_221','stand_231','stand_241','stand_251','stand_261','stand_271','stand_28
1','stand_291',
'stand_311','stand_321','stand_331','stand_341','stand_351','stand_361')
group by replication")

```

שמירת מספר הישגיות אשר נטשו את דוכן מפלגת שמאל:

```

abandon_left <- sqldf("select replication, count(value) as abandoned
from full_attribute_data
where key = 'abandon' and value = 1
group by replication")

```

אחוז הנוטשים:

```

abandon_measure <- sqldf("select (abandoned*100.00)/(voters_left*100.00) as abandon_measure
from visitors_left as v join abandon_left as a on v.replication=a.replication")

```

- **הגדלת סיכויי הזכייה – בסוף כל יום בחירות יש בידינו את תוצאות הבחירות ואת מספר הקולות שכל מפלגה קיבלה. באמצעות נתון זה אנו יכולים לחשב את אחוז הקולות שקיבלה המפלגה שלנו מכלל הקולות.**

מספר הקולות שקיבלה מפלגת שמאל:

```

party_1 <- sqldf("select replication, max(value) as left_votes
from full_attribute_data
where key = 'party_1'
group by replication")

```

כלל הקולות:

```

total_votes <- sqldf ("select replication, sum(value) as total_votes from (select*
from full_attribute_data
where key in('party_1','party_2','party_3')
group by replication, key
having max(value)) group by replication")

```

אחוז הקולות למפלגת שמאל מכלל הקולות:

```

votes_measure <- sqldf("select left_votes/total_votes as votes_measure
from total_votes as t join party_1 as v on t.replication=v.replication")

```

4.2 סוג המערכת

המערכת שלנו היא מסוג מערכת מסתיימת - **System Terminating**, כיוון שבחרנו את פעילותה של הסימולציה לאורך יום עבודה אחד, כלומר יש לה תנאי סיום ולכן אין צורך בזמן חימום. יחידות הזמן במודל הוגדרו להיות בדקות, ומשך הזמן של ריצה בודדת הוא 15 שעות ו- 20 דקות

בסה"כ $\leftarrow 920$ דקות. בהתאם לכך, הגדרנו את יצירת כלל הישויות במערכת להיות בטווח ערכים זה, ובסוף יום הפעילות הישויות עוזבות את המערכת (כלומר בסיום זמן הריצה).

מספר הריצות הראשוני נקבע באופן שרירותי והינו $n_0 = 30$.

נקבע רמת מובהקות של $\alpha_{Total} = 0.1$.

רמת הביטחון של כל מדד תהיה רמת המובהקות חלקי מספר המדדים, מכאן ש $\alpha_i = 0.05$.

בחרנו את רמת הדיוק היחסי, באופן שרירותי, להיות $\gamma = 0.12$.

כדי למצוא את מס' הריצות המינימאלי נקבע את החסם: $\frac{\gamma}{1+\gamma} = 0.1071$, ונוודא שהדיוק היחסי

מקיים את התנאי הבא: $\frac{\delta(n,\alpha)}{\bar{x}} < \frac{\gamma}{1+\gamma} = 0.1071$.

תוצאות הדיוק היחסי לכל חלופה ומדד התקבלו על פי הנוסחה: $\frac{\delta}{\bar{x}} = \frac{t_{30-1,1-\frac{0.05}{2}} \cdot \frac{sd}{\sqrt{30}}}{avg}$.

הקטנת אחוז הנוטשים מכלל המבקרים בדוכן המפלגה		אחוז הקולות למפלגת שמאל	
מצב קיים	0.01561379	✓	0.02165837
חלופה 1	0.01229424	✓	0.01962857
חלופה 2	0.01568906	✓	0.01925472

ניתן לראות שכלל התוצאות עומדות ביחס הדיוק היחסי שהגדרנו (נספח 7), לכן מספר החדרות ההתחלתי יישאר אותו דבר.

4.3 ניתוח מצב קיים ביחס למדדים ולמטרות (הפרויקט / הארגון)

להלן תוצאות המדדים עבור 30 ריצות במצב הקיים:

אחוז נוטשים	אחוז מצביעים	
20.71148	27.21467	ממוצע
0.8660404	1.578516	סטיית תקן

ניתן לראות כי אחוז הנוטשים מדוכני המפלגה הוא גבוה, וכן כי אחוז המצביעים עבור מפלגת שמאל הוא נמוך. בהתאם לנתונים המוצגים בטבלה, נשאף שהחלופות שנבחר ישפרו את המדדים הנ"ל, מתוך אמונה שאלה המדדים שיכולים להעיד בצורה המובהקת ביותר על שיפור טיב השירות.

4.4 תיאור החלופות

בכדי לשפר את המצב הקיים נבחן שתי חלופות במסגרת התקציב 30,000 ₪. (בנספח 6).

חלופה מספר 1:

החלופה הראשונה מכילה את השינויים הבאים:

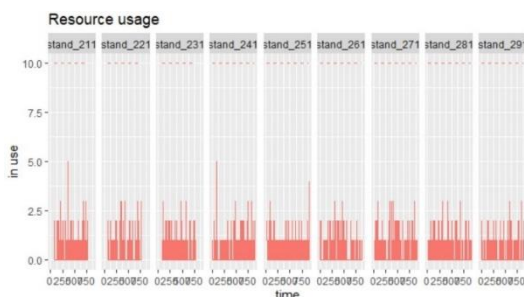
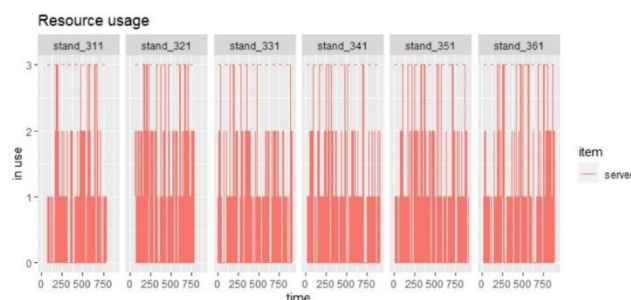
- **הוספת דוכן לאזור מרכז:** [נספח 6.3](#).

בדקנו כמה בוחרים מגיעים לכל מתחם לפי האינדקסים: 1-צפון, 2-מרכז ו-3 דרום. ניתן לראות **(נספח ג' תמונה 1.1)** כי למתחם מרכז רוב הבוחרים מגיעים. מכיוון שכמות האנשים באזור זה היא הגדולה ביותר דבר זה גורר כי כמות הבוחרים שיעברו בדוכן של מפלגת שמאל הוא הגבוה ביותר ביחס לשאר האזורים, והוספת דוכן נוסף תעלה את סיכויי המפלגה לזכות.

- **הוספת נציג למתחם:**

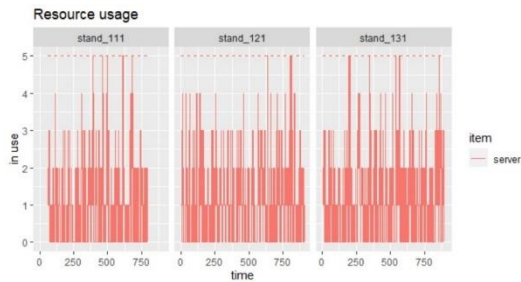
בכדי להחליט לאן להעביר נציגים מאזור לאזור והיכן להוסיף נציג נוסף, בדקנו את הניצולת של הנציגים בדוכני המפלגות בכל אזור בדוכן שמאל, בכדי לבדוק האם מגיעים אנשים לדוכנים אשר לא מקבלים שירות מנציגי המפלגות ועוזבים. לפי נתונים אלו הסקנו היכן יש להוסיף נציג ומאיפה ניתן להעביר נציג.

אזור דרום: כפי שניתן לראות בגרף, כמות הנציגים ממפלגת שמאל בדוכנים באזור דרום נמצאת בקיבולת המקסימאלית שלה מספר רב של פעמים, כלומר כל שלושת הנציגים תפוסים. מכך ניתן להסיק כי כמות רבה של בוחרים לא מדברים עם הנציגים ועוזבים את הדוכן בטרם קיבלו שירות. דבר זה גורר שההסתברות של בוחר מסוים לבחור במפלגת שמאל אינה עולה.



אזור מרכז: כפי שניתן לראות בגרף כמות

הנציגים ממפלגת שמאל בדוכנים באזור מרכז אינה נמצאת בקיבולת המקסימאלית שלה אף לא פעם אחת, וכל הנציגים נותנים שירות ולא קיימים אנשים שנוטשים את הדוכן.



אזור צפון: כפי שניתן לראות בגרף כמות הנציגים ממפלגת שמאל בדוכנים באזור צפון נמצאת בקיבולת המקסימאלית שלה מספר רב של פעמים (אך מועט ביחס לדרום) לאורך הסימולציה כלומר כל חמשת הנציגים תפוסים. מכך ניתן להסיק כי כמות רבה של בוחרים לא מדברים עם הנציגים ועוזבים את הדוכן. דבר זה גורר שההסתברות של בוחר מסוים לבחור במפלגת שמאל אינה עולה.

לסיום, נשים לב שבאזור המרכז יש 10 נציגים שהקיבולת שלהם אינה מגיעה למקסימום, ולעומת זאת בצפון ובדרום נציגי מפלגת שמאל מגיעה לשיאה במהלך הסימולציה. ישנם מצבים שרוב הנציגים תפוסים בצפון ובדרום ועל מנת להקטין את המדד של מספר הבוחרים שנדחים בחרנו לאייש בעוד נציגים. עשינו זאת ע"י העברת 2 נציגים מהמרכז לדרום וקינינו נציג נוסף לדוכן שמאל באזור צפון.

• העברת סדנא לנציגים ממתחם צפון לשיפור יכולת השכנוע שלהם:

בחרנו לבצע סדנא במפלגת שמאל במתחם צפון. הבחירה במתחם זה ספציפית נובעת מהטבלה (נספח 5.3). ניתן לראות שמספר הנוטשים הגבוה ביותר הוא בשלושת הדוכנים של המפלגה באזור הצפון ולכן נבחר קודם כל לטפל בה. נראה את המדדים בחלופה זו:

אחוז נוטשים	אחוז מצביעים	
17.95867	30.53955	ממוצע
0.005912816	0.01605351	סטיית תקן

חלופה מספר 2:

החלופה השנייה מכילה את השינויים הבאים:

• העברת סדנא לנציגים ממתחם צפון לשיפור יכולת השכנוע שלהם:

בדומה לחלופה 1 בחרנו לבצע סדנא במפלגת שמאל במתחם צפון. הבחירה במתחם זה ספציפית נובעת מהטבלה (נספח 5.3). ניתן לראות שמספר הנוטשים הגבוה ביותר הוא בשלושת הדוכנים של המפלגה באזור הצפון ולכן נבחר קודם כל לטפל בה.

• השקעה בתעמולה תקשורתית ביום הבחירות:

בחרנו בחלופה זו בכדי להעלות את סיכויי המפלגה לזכות ביום הבחירות, לחשוף כמה שיותר בוחרים למצע המפלגה. שינוי זה משנה את הדעה ההתחלתית של הבוחרים בסיכוי של 0.12, כלומר הסיכוי הראשוני להצביע למפלגה עולה, וכך גם סיכויי הזכייה עולים. נראה את המדדים בחלופה זו:

אחוז נוטשים		אחוז מצביעים
מוצע	17.0118	36.08599
סטיית תקן	0.007147691	0.01860777

4.5 השוואה סטטיסטית בין מצב קיים לחלופות המוצעות

עבור כל מדד וחלופה ביצענו מבחן t מזווג, על מנת לבחון את רווח הסמך, ובהתאם לכך נבחר עבור כל מדד את החלופה הטובה ביותר. **תוצאות ההרצות מפורטות בנספח.**

מדד	השוואה	גבול עליון	גבול תחתון	תוצאות המבחן	החלופה הנבחרת
אחוז נוטשים	מצב קיים וחלופה 1	0.03146662	0.2358971	חלופה 1	חלופה 2
	מצב קיים וחלופה 2	0.04268777	0.03130587	חלופה 2	
	חלופה 1 וחלופה 2	0.014005742	0.004931562	חלופה 2	
אחוז מצביעים	מצב קיים וחלופה 1	-0.02247094	-0.04402456	חלופה 1	חלופה 2
	מצב קיים וחלופה 2	-0.0766574	-0.1007669	חלופה 2	
	חלופה 1 וחלופה 2	-0.04388611	-0.06704271	חלופה 2	

7. מסקנות

המטרה העיקרית של הפרויקט היא שיפור ביצועיה של מפלגה מסוימת לבחירתנו, כך שסיכוייה לניצחון ביום הבחירות יגדלו, וכן שמספר האנשים הנוטשים את דוכן המפלגה יקטן (מה שיגרום בסופו של דבר להגדלת סיכויי הזכייה).

לאחר ביצוע החלופות ובחינתן, התוצאות שהתקבלו במחקר שערכנו מראות באופן חד משמעי כי **חלופה 2 עדיפה על פני המצב הקיים וכן על פני חלופה 1**. כמו כן, בנוסף לכך שמבחינת המדדים שהגדרנו היא החלופה העדיפה, היא גם **החלופה הזולה ביותר** מבין שתי החלופות, ויכולה אף לחסוך למפלגה כסף (למרות שזהו אינו מדד), ואת יתרת הסכום תוכל המפלגה להשקיע במקום אחר בו הוא נדרש.

בהתאם לתוצאות המחקר שלנו, נמליץ למפלגה להעביר סדנה לנציגים ממתחם צפון לשיפור יכולת השכנוע שלהם ובכך העלאת סיכויי ההעדפה של הבוחרים במפלגה. כמו כן נמליץ למפלגה על השקעה בתעמולה תקשורתית ביום הבחירות, וזאת על מנת להעלות את סיכויי הזכייה גם כן.

את תוצאות המדדים קיבלנו לאחר 30 ריצות של כל אחת מהחלופות והמצב הקיים, כאשר עמדנו בכל מדד בדיוק היחסי הרצוי. על כן ניתן להניח שהמודל אמין וכי ניתן להסתמך על תוצאותיו ולבצע את השינויים הנ"ל במפלגה.

8. נספחים:

נספח 1- נתונים1.1 נתוני המערכת הנחקרת

זמן	פעולה
$\Gamma \sim (\alpha = 1.9688, \lambda = 0.6994)$	קצב הגעת בוחרים לקלפיות גדולות
$\Gamma \sim (\alpha = 2.0235, \lambda = 2.8359)$	קצב הגעת בוחרים בעלי מוגבלויות
Exp(mean=1.11 min)	קצב הגעת בוחרים לקלפיות הקטנות
U[2,6]	זמן שיחה עם נציג מפלגה
U[3.5,6.5]	זמן לאיבוד סבלנות בשיחה עם נציג
Exp(mean =40 seconds)	זמן זיהוי בפני ועדת הקלפי
Norm(1 min,6 sec)	זמן הצעה
משולשית a=3 b=5 c=9	התפלגות אזור מגורי הבוחרים
30% מאזור הצפון 40% מאזור המרכז 30% מאזור הדרום	התפלגות אזור מגורי הבוחרים

נספח 2- המסלולים במודל**מסלול הבוחרים:**

הישויות בסימולציה נוצרות בהתפלגויות הבאות:

```
elections%%
# small kalpi is open between 08:00-20:00
add_generator("small_kalpi_voter", traj_small_kalpi, from_to(start_time = 60, stop_time = 780, dist = function () rexp(1, 1/1.11)), priority = 0, mon = 2)%%
# large kalpi is open between 07:00-22:00
add_generator("large_kalpi_voter", traj_large_kalpi, from_to(start_time = 0, stop_time = 900, dist = function () rgamma(1, 1.9688020, 0.6994063)), priority = 0, mon = 2)%%
# negishut kalpi is the large kalpi, therefore it's open between 07:00-22:00
add_generator("negishut_kalpi_voter", traj_negishut_kalpi, from_to(start_time = 0, stop_time = 900, dist = function () rgamma(1, 2.023509, 2.835970)), priority = 1, mon = 2)%%
# assumption : the manager will come at the start of the simulation (time = 0) in intervals of 3 hours,
# and it doesn't matter what is the type of the kalpi.
# moreover, there is no need to arrive at 22:00 (i.e time = 900)
add_generator("north_manager", traj_north_manager, distribution = at(0, 180, 360, 540, 720), mon=2)%%
add_generator("center_manager", traj_center_manager, distribution = at(0, 180, 360, 540, 720), mon=2)%%
add_generator("south_manager", traj_south_manager, distribution = at(0, 180, 360, 540, 720), mon=2)%%
# a fictive entity that closes the kalpis, each one at it's time
# assumption : the stands stay open and only the opportunity to vote is closed.
# assumption : although the kalpis are closed, voters still wait in queue.
add_generator("fictive_small", traj_close_small_kalpi, distribution = at(780), mon = 0)%%
add_generator("fictive_large", traj_close_large_kalpi, distribution = at(900), mon = 0)%%
# index --> 1 - north area, 2 - center, 3 - south
add_generator("mashkif_1", traj_mashkif_1, distribution = at(900), mon = 2)%%
add_generator("mashkif_2", traj_mashkif_2, distribution = at(900), mon = 2)%%
add_generator("mashkif_3", traj_mashkif_3, distribution = at(900), mon = 2)
```

ישויות הבוחרים ממשיכות במסלולן אל עבר קבלת התכונות שמעידות על אופייה של כל ישות מבחינת סוג הקלפי:

```
# trajectories to initialize what type of kalpi the voters will go - small or large
traj_small_kalpi <- trajectory("traj_small_kalpi")%%
set_attribute(key = "kalpi_type", value = 1)%%
join(traj_initialize)

traj_large_kalpi <- trajectory("traj_large_kalpi")%%
set_attribute(key = "kalpi_type", value = 2)%%
join(traj_initialize)

traj_negishut_kalpi <- trajectory("traj_negishut_kalpi")%%
set_attribute(key = "kalpi_type", value = 2)%%
join(traj_initialize)
```

לאחר מכן כלל ישויות הבוחרים עוברות אל מסלול בו הן מקבלות את שאר התכונות, הנוגעות בדעה הראשונית, מידת הקיצוניות, התפלגות Z ווקטור ההעדפות הסופי המחושב:

```
# trajectory to initialize the specific kalpi the voter will go, according to it's area and type
traj_initialize <- trajectory("traj_initialize")%%
set_attribute(key = "area", value = function () my_area ())%%
set_attribute(key = "kalpi_num", value = function () my_kalpi (get_attribute(elections, key = "area"), get_attribute(elections, key = "kalpi_type"))
# voter's agenda
set_attribute(key = "type", value = function () rdiscrte(1, c(0.4,0.1,0.5), c(1,2,3)))%%
set_attribute(key = "Z", value = function () densitz Z ())%% # voter's extreme level
# opinion including the extreme level
set_attribute(keys = c("opinion_1", "opinion_2", "opinion_3"), values = function () opinion_with_Z (get_attribute(elections, key = "type"), get_s
set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () opinion_with_Z (get_attribute(elections, key = "t
# the order of stands of a specific voter
set_attribute(keys = c("stands_order_1", "stands_order_2", "stands_order_3"), values = function () stand_order (get_attribute(elections, key="opi
set_attribute(key = "max_opinion", value = function () set_max_opinion (get_attribute(elections, key="opinion_1"), get_attribute(elections, key=
join(traj_stand_1)
```

בשלב הגדרת תכונות אלו, אנו משתמשים בפונקציות לצורך קביעת ערכי התכונות.

הפונקציה my_kalpi מגדירה עבור כל ישות לאיזה מספר קלפי ילך על פי האזור בו הוא נמצא:

```
my_kalpi <- function (area, kalpi_type) {
  kalpi <- switch(
    paste0(area, kalpi_type),
    "11" = 1,
    "12" = sample(2:3, 1),
    "21" = sample(1:3, 1),
    "22" = sample(4:9, 1),
    "31" = sample(1:2, 1),
    "32" = sample(3:6, 1)
  ) # switch
  return (kalpi)
} # returns the number of kalpi of each voter, according to area and type of kalpi
```

הפונקציה densityZ מחשבת עבור כל ישות את מידת הקיצוניות שלו. פונקציית הצפיפות חושבה על ידי שיטת הסופרפוזיציה כפי שנלמד בהרצאות:

```
densityZ <- function () { # calculations in the final report
  u <- runif(1,0,1)
  if (u < 0.5)
    return (u/8)^(1/3)
  else
    return (u/2)
} # a function that calculates the extreme level of a voter - COMPOSITION of f(x)
```

הפונקציה opinion_with_Z מחשבת עבור כל ישות את הדעה ההתחלתית שלו, בהתאם למידת הקיצוניות והתפלגות Z:

```
opinion_with_Z <- function (type, Z) {
  opinion <- c(0.25, 0.5, 0.25) # initial opinion
  for (i in 1 : 3) {
    if (i == type)
      opinion[i] <- opinion[i] + Z
    else opinion[i] <- opinion[i] - (Z/2)
  } # for
  return (opinion)
} # a function that calculates the opinion vector of a voter with the Z
```

הפונקציה stand_order מגדירה עבור הישויות את הסדר של הדוכנים בהם יבקרו בהגיעם למתחם הקלפי:

```
stand_order <- function (opinion_1, opinion_2, opinion_3) {
  opinion_vec <- c(opinion_1, opinion_2, opinion_3)
  order_vec <- c(1,3,2) # default order, center voter
  max_index <- which(opinion_vec == max(opinion_vec), arr.ind = TRUE) # המפלגה שהדעה שלה מועדפת ביותר
  if (max_index == 1)
    order_vec <- c(3,2,1) # left voter
  if (max_index == 3)
    order_vec <- c(2,1,3) # right voter
  return(order_vec)
} # a function that returns the order of the stands of a voter
```

לאחר הגדרת התכונות, הישויות ממשיכות אל עבר מתחם הקלפי וכן אל דוכן המפלגה אליה אמורה ללכת על פי וקטור ההעדפות שלה. ראשית אנו בודקים האם הדוכן הראשון אליה הישות הולכת שווה בהסתברות שלו אל הדוכן המועדף על ידי הישות, וזאת על מנת לבדוק האם הישות יכולה לנטוש את הדוכן או לא:

```
# here we'll check if there's equal probabilities between stand 1 and stand 3 (which the highest)
traj_stand_1 <- trajectory("traj_stand_1") %>%
  branch(option = function () isEqual (get_attribute(elections, key="max_opinion"),
    get_attribute(elections, key=paste0("opinion_", get_attribute(elections, key="stands_order_1")))), continue = c(FALSE, FALSE), stand1_no_renege, stand1_renege)
```

הבדיקה מתבצעת על ידי שימוש בפונקציה isEqual שמחשבת את השוויון (או אי-שוויון) בין ההסתברויות:

```
isEqual <- function (max_opinion, opinion) {
  if (max_opinion == opinion)
    return(1)
  else return(2)
} # returns T/F if there is equal probabilities with the highest party of a voter
```

לאחר בדיקה זו, כל ישות שמגיעה אל הדוכן הראשון תתפצל אל עבר שני מסלולים, מסלול אחד שניתן לנטוש אותו ואחד שלא ניתן. בחרנו למדל את המעבר בין הדוכנים בצורה מודולרית על ידי שימוש בפונקציית `paste0` שגורמת לכל ישות לנטוש את המשאב המותאם לקלפי באזור בו היא נמצאת:

```
# מסלול נטוש במחנה
stand1_abandon <- trajectory("stand1_abandon")%>%
  set_attribute(key = "abandon", value = function () aband_left(get_attribute(elections, "stands_order_1")))%>%
  set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () updated_values_renege(
    join(traj_stand_2)

stand1_reject <- trajectory("stand1_reject")%>%
  set_attribute(key = "reject", value = function () rejected_left(get_attribute(elections, "stands_order_1")))%>%
  join(traj_stand_2)

stand1_no_renege <- trajectory("stand1_no_renege")%>%
  simmer::select(resources = function () get_stand_resource(get_attribute(elections, key = "area"), get_attribute(electic
    seize_selected(amount = 1, id = 12, continue = FALSE, reject = stand1_reject)%>%
    timeout(function () runif(1, 2, 6))%>%
    release_selected(amount = 1, id = 12)%>%
    set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () updated_values(get_attri
      join(traj_stand_2)

stand1_renege <- trajectory("stand1_renege")%>%
  simmer::select(resources = function () get_stand_resource(get_attribute(elections, key = "area"), get_attribute(electic
    seize_selected(amount = 1, id = 11, continue = FALSE, reject = stand1_reject)%>%
    renege_in(t = function () runif(1, 3.5, 6.5), out = stand1_abandon)%>%
    timeout(function () runif(1, 2, 6))%>%
    renege_abort()%>%
    release_selected(amount = 1, id = 11)%>%
    # מעלה הסתברות
    set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () updated_values(get_attri
      join(traj_stand_2)
```

בתמונה ניתן לראות כי קיימים מספר מסלולים – מסלול `renege` אליו ישלחו הישויות שיכולות לנטוש את הדוכן, מסלול `no renege` אותו לא ניתן לנטוש, מסלול `reject` אליו נשלחות ישויות אשר ניסו לתפוס את המשאב ולא הצליחו, וכן מסלול `abandon`, אליו מגיעות הישויות שנטשו את העמדה אליה הגיעו.

אנשים שלא נטשו את הדוכן אליו הגיעו, ישנו את וקטור ההעדפות שלהם באמצעות הפונקציה הבאה:

```
updated_values <- function (opinion_1, opinion_2, opinion_3, stand_party) {
  x <- runif(1, 0, 0.5)
  y <- x/2
  if (stand_party == 1)
    return(c(opinion_1 + x, opinion_2 - y, opinion_3 - y))
  if (stand_party == 2)
    return(c(opinion_1 - y, opinion_2 + x, opinion_3 - y))
  if (stand_party == 3)
    return(c(opinion_1 - y, opinion_2 - y, opinion_3 + x))
} # if a voter didn't abandon a stand, we will raise the probabilities of this stand
```

אנשים אשר נטשו את הדוכן אליו הגיעו, ישנו את וקטור ההעדפות שלהם גם כן באמצעות הפונקציה הבאה:

```
updated_values_renege <- function (opinion_1, opinion_2, opinion_3, stand_party) {
  x <- runif(1, 0, 0.25)
  y <- x/2
  if (stand_party == 1)
    return(c(opinion_1 - x, opinion_2 + y, opinion_3 + y))
  if (stand_party == 2)
    return(c(opinion_1 + y, opinion_2 - x, opinion_3 + y))
  if (stand_party == 3)
    return(c(opinion_1 + y, opinion_2 + y, opinion_3 - x))
} # if a voter abandon a stand, we will decline the probabilities of the other stands
```


בהמשך ממשיכות הישויות אל עבר הדוכן השני, ומבצעות את אותו המסלול בדומה לדוכן הראשון שהוצג מעלה:

```
stand2_abandon <- trajectory("trajofRenege2")%>%
  set_attribute(key = "abandon", value = function () aband_left(get_attribute(elections, "stands_order_2")))%>%
  set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () updated_values_renege(get
  join(traj_stand_3))

stand2_reject <- trajectory("stand2_reject")%>%
  set_attribute(key = "reject", value = function () rejected_left(get_attribute(elections, "stands_order_2")))%>%
  join(traj_stand_3)

stand2_no_renege <- trajectory("stand2_no_renege")%>%
  simmer::select(resources = function () get_stand_resource(get_attribute(elections, key = "area"), get_attribute(election
  seize_selected(amount = 1, id = 22, continue = FALSE, reject = stand2_reject)%>%
  timeout(function () runif(1, 2, 6))%>%
  release_selected(amount = 1, id = 22)%>%
  set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () updated_values(get_attrib
  join(traj_stand_3))

stand2_renege <- trajectory("stand2_renege")%>%
  simmer::select(resources = function () get_stand_resource(get_attribute(elections, key = "area"), get_attribute(election
  seize_selected(amount = 1, id = 21, continue = FALSE, reject = stand2_reject)%>%
  renege_in(t = function () runif(1, 3.5, 6.5), out = stand2_abandon)%>%
  timeout(function () runif(1, 2, 6))%>%
  renege_abort()%>%
  release_selected(amount = 1, id = 21)%>%
  set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () updated_values(get_attrib
  join(traj_stand_3))

traj_stand_2 <- trajectory("traj_stand_2")%>%
  branch(option = function () isEqual(get_attribute(elections, key = "max_opinion"), get_attribute(elections, key = paste0("opi
```

לבסוף מגיעות הישויות אל הדוכן השלישי, אותו, כידוע, הן לא יכולות לנטוש (על פי ההוראות). הישויות מנסות לתפוס את המשאב, ואם הן לא מצליחות הן עוברות אל מסלול ייעודי:

```
stand3_reject <- trajectory("stand3_reject")%>%
  set_attribute(key = "reject", value = function () rejected_left(get_attribute(elections, "stands_order_3")))%>%
  join(traj_vote)

traj_stand_3 <- trajectory("traj_stand_3")%>%
  simmer::select(resources = function () get_stand_resource(get_attribute(elections, key = "area"), get_attribute(elections, key = "kalpi_num"),
  seize_selected(amount = 1, id = 3, continue = FALSE, reject = stand3_reject)%>%
  timeout(function () runif(1, 2, 6))%>%
  release_selected(amount = 1, id = 3)%>%
  set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () updated_values(get_attribute(elections, key = "r
  # נכנסת לאזור ההצבעה
  join(traj_vote)
```

לאחר ביקור בדוכן השלישי ושינוי וקטור העדיפויות, הישויות ממשיכות אל עבר מתחם הקלפי, לא לפני שאנו מבצעים בדיקה של וקטור ההעדפות שסוכם ל1 וכן שאין בו ערכים קטנים מ0 או גדולים מ-1.

```
traj_vote <- trajectory("traj_vote")%>%
  set_attribute(keys = c("final_opinion_1", "final_opinion_2", "final_opinion_3"), values = function () opinion_check(get_attribute(elections, key = "new_opinion_1"), get_attribute(elections, key = "new_opinion_2"),
  simmer::select(resources = function () get_kalpi_resource(area = get_attribute(elections, key = "area"), kalpi_num = get_attribute(elections, key = "kalpi_num"), id = 0)%>%
  seize_selected(amount = 1, id = 0)%>%
  timeout(function () exp(1, 60/40))%>% # identification time
  timeout(function () rlnormore(mu = 1, sd = 6/60))%>% # voting time
  # נכנסת לאזור ההצבעה
  set_global(key = function () my_vote(get_attribute(elections, "final_opinion_1"), get_attribute(elections, "final_opinion_2"), get_attribute(elections, "final_opinion_3"), value = 1, mod = "+", init = 0)%>%
  release_selected(amount = 1, id = 0)
```

בדיקת וקטור העדיפויות מתבצע על ידי נרמול הערכים בפונקציה הבאה:

```
opinion_check <-function (opinion_1, opinion_2, opinion_3){
  vec <- c(opinion_1, opinion_2, opinion_3)
  if (vec[1] < 0)
    vec[1] <- 0
  if (vec[2] < 0)
    vec[2] <- 0
  if (vec[3] < 0)
    vec[3] <- 0
  return(vec/sum(vec))
} # a function that normalize the vector of opinions
```

כפי שהזכרנו כבר, ניסיון תפיסת הישויות מתבצע על ידי פונקציה ייעודית שתפקידה להחזיר את המשאב אותה הישות צריכה לתפוס. הפונקציה מקבלת את האזור ואת מספר הקלפי של כל ישות, וכן הפונקציה של הדוכנים מקבלת גם את המפלגה בה נמצאת הישות בדוכן הרלוונטי, ומחזירה את שם המשאב שיש לנסות לתפוס:

```
get_kalpi_resource <- function (area, kalpi_num){
  return (paste0("kalpi_", area, kalpi_num))
} # a function that returns the kalpi resource

get_stand_resource <- function (area, kalpi_num, party) {
  return (paste0("stand_", area, kalpi_num, party))
} # a function that returns the stand resource
```

מסלול מנהלי האזורים

מנהלי האזורים מגיעים בכל שלוש שעות וסוגרים את הקלפיות באזורים לזמן המשתנה בין הגעה להגעה.

יצרנו את ישויות המנהלים כך:

```
# assumption : the manager will come at the start of the simulation (time = 0) in intervals of 3 hours,
#               and it doesn't matter what is the type of the kalpi.
#               moreover, there is no need to arrive at 22:00 (i.e time = 900)
add_generator("north_manager", traj_north_manager, distribution = at(0, 180, 360, 540, 720), mon=2)%>%
add_generator("center_manager", traj_center_manager, distribution = at(0, 180, 360, 540, 720), mon=2)%>%
add_generator("south_manager", traj_south_manager, distribution = at(0, 180, 360, 540, 720), mon=2)%>%
```

לאחר יצירת מנהלי האזורים, כל מנהל הולך לאזור שלו, בו נקבע לכמה זמן יסגור את הקלפיות באזור:

```
!
# -----managers-----
# assumption : the manager closes all the kalpiot in his area, for an equal time
#               and he doesn't close the stages.

traj_north_manager <- trajectory("traj_north_manager")%>% # north = 1
  setCapacity(area = 1, kalpi_num = 3, capacity_value = 0)%>%
  timeout(function () rtriangle(1, 3, 15, 9))%>%
  setCapacity(area = 1, kalpi_num = 3, capacity_value = 1)

traj_center_manager <- trajectory("traj_center_manager")%>% # center = 2
  setCapacity(area = 2, kalpi_num = 9, capacity_value = 0)%>%
  timeout(function () rtriangle(1, 3, 15, 9))%>%
  setCapacity(area = 2, kalpi_num = 9, capacity_value = 1)

traj_south_manager <- trajectory("traj_south_manager")%>% # south = 3
  setCapacity(area = 3, kalpi_num = 6, capacity_value = 0)%>%
  timeout(function () rtriangle(1, 3, 15, 9))%>%
  setCapacity(area = 3, kalpi_num = 6, capacity_value = 1)
```

זמן סגירת הקלפיות נקבע על ידי דגימת מספר מהתפלגות משולשית.

עם סיום סגירת הקלפיות, ולאחר פתיחתן, מסיים מנהל האזור את תפקידו בסימולציה.

מסלול פיקטיבי

בנוסף לישויות שהצגנו עד כה, בחרנו לייצר שתי ישויות פיקטיביות שתפקידן לסגור את הקלפיות עצמן בשעות המוגדרות (את הקלפי הגדולה בשעה 22:00 ואת הקלפי הקטנה בשעה 20:00):

```
# a fictive entity that closes the kalpis, each one at it's time
# assumption : the stands stay open and only the opportunity to vote is closed.
# assumption : although the kalpis are closed, voters still wait in queue.
add_generator("fictive_small", traj_close_small_kalpi, distribution = at(780), mon = 0)%>%
add_generator("fictive_large", traj_close_large_kalpi, distribution = at(900), mon = 0)%>%
```

```
# ----- close the kalpi -----
traj_close_small_kalpi <- trajectory("traj_close_small_kalpi")%>%
  close_small_kalpi ()

traj_close_large_kalpi <- trajectory("traj_close_large_kalpi")%>%
  close_large_kalpi ()

close_small_kalpi <- function (trajectory) {
  updatedPath <- set_capacity(trajectory, resource = "kalpi_11", value = 0)%>%
    set_capacity(resource = "kalpi_21", value = 0)%>%
    set_capacity(resource = "kalpi_22", value = 0)%>%
    set_capacity(resource = "kalpi_23", value = 0)%>%
    set_capacity(resource = "kalpi_31", value = 0)%>%
    set_capacity(resource = "kalpi_32", value = 0)%>%

  set_queue_size(resource = "kalpi_11", value = 0)%>%
  set_queue_size(resource = "kalpi_21", value = 0)%>%
  set_queue_size(resource = "kalpi_22", value = 0)%>%
  set_queue_size(resource = "kalpi_23", value = 0)%>%
  set_queue_size(resource = "kalpi_31", value = 0)%>%
  set_queue_size(resource = "kalpi_32", value = 0)

  return(updatedPath)
}# a function that set the capacity and the queue size of the small kalpis to 0

close_large_kalpi <- function (trajectory) {
  updatedPath <- set_capacity(trajectory, resource = "kalpi_12", value = 0)%>%
    set_capacity(resource = "kalpi_13", value = 0)%>%
    set_capacity(resource = "kalpi_24", value = 0)%>%
    set_capacity(resource = "kalpi_25", value = 0)%>%
    set_capacity(resource = "kalpi_26", value = 0)%>%
    set_capacity(resource = "kalpi_27", value = 0)%>%
    set_capacity(resource = "kalpi_28", value = 0)%>%
    set_capacity(resource = "kalpi_29", value = 0)%>%
    set_capacity(resource = "kalpi_33", value = 0)%>%
    set_capacity(resource = "kalpi_34", value = 0)%>%
    set_capacity(resource = "kalpi_35", value = 0)%>%
    set_capacity(resource = "kalpi_36", value = 0)%>%

  set_queue_size(resource = "kalpi_12", value = 0)%>%
  set_queue_size(resource = "kalpi_13", value = 0)%>%
  set_queue_size(resource = "kalpi_24", value = 0)%>%
  set_queue_size(resource = "kalpi_25", value = 0)%>%
  set_queue_size(resource = "kalpi_26", value = 0)%>%
  set_queue_size(resource = "kalpi_27", value = 0)%>%
  set_queue_size(resource = "kalpi_28", value = 0)%>%
  set_queue_size(resource = "kalpi_29", value = 0)%>%
  set_queue_size(resource = "kalpi_33", value = 0)%>%
  set_queue_size(resource = "kalpi_34", value = 0)%>%
  set_queue_size(resource = "kalpi_35", value = 0)%>%
  set_queue_size(resource = "kalpi_36", value = 0)
}
```

מסלול משקיפי המפלגות

משקיפי הקלפיות הן הישויות האחרונות בסימולציה שלנו. ישויות אלה נוצרות בסוף יום הבחירות, ומטרתן היא לנסות "לרמות" בתהליך הבחירות ולהוסיף למפלגתן עוד 50 קולות. בשלב זה בחרנו למדל 3 ישויות (ישות אחת לכל אזור):

```
# index --> 1 = north area, 2 = center , 3 = south
add_generator("mashkif_1", traj_mashkif_1, distribution = at(900), mon = 2)%>%
add_generator("mashkif_2", traj_mashkif_2, distribution = at(900), mon = 2)%>%
add_generator("mashkif_3", traj_mashkif_3, distribution = at(900), mon = 2)
|
```

בהסתברות של 60% הישות תעזוב את המסלול ולא תעשה דבר, ובהסתברות המשלימה היא תגיע לכדי לרמות בבחירות.

```
# ----- mashkifim -----
traj_mashkif_1 <- trajectory("traj_mashkif_1")%>%
  leave(prob = 0.6)%>%
  set_global(key = function () cheat (), value = 50, mod = "+")

traj_mashkif_2 <- trajectory("traj_mashkif_2")%>%
  leave(prob = 0.6)%>%
  set_global(key = function () cheat (), value = 50, mod = "+")

traj_mashkif_3 <- trajectory("traj_mashkif_3")%>%
  leave(prob = 0.6)%>%
  set_global(key = function () cheat (), value = 50, mod = "+")
```

המפלגה אליה מוסיפה הישות את הקולות מחושבת על ידי פונקציית CHEAT:

```
cheat <- function () {
  party <- rdiscrete(1, c(1/3, 1/3, 1/3), c(1,2,3))
  return(paste0("party_", party))
} # returns the party to which we add 50 votes
```

*** הפניה נוספת 3- הישויות המשאבים והתכונות שהוגדרו במודל

3.1 הישויות שהוגדרו במודל

- **בוחר בקלפי קטנה** - הבוחרים מגיעים לקלפי שפתוח בין השעות 8:00-20:00, קצב הגעת הבוחרים לקלפי הקטנה מתפלג אקספוננציאלית עם קצב של $\lambda = \frac{1}{1.11}$.
- **בוחר בקלפי גדולה (נגישה)** - הבוחרים מגיעים לקלפי שפתוח בין השעות 7:00-22:00, קצב הגעת הבוחרים לקלפי הגדולה מתפלג גמא עם $\alpha = 1.9688$ ו- $\lambda = 0.6994$.
- **בוחר בעל מוגבלות בקלפי גדולה (נגישה)** - הבוחרים מגיעים לקלפי שפתוח בין השעות 7:00-22:00, קצב הגעת הבוחרים לקלפי הגדולה מתפלג גמא עם $\alpha = 2.0235$ ו- $\lambda = 2.8359$.
- **אחראי אזור** - יצרנו 3 ישויות של אחראי אזור (אחד לכל אזור) אשר עוצר את תהליך הבחירות על מנת לבצע בדיקת תקינות ותדרוך לוועדת הקלפיות.
- **ישות פיקטיבית שסוגרת את הקלפיות הקטנות** - ישות זו סוגרת את הקלפי בשעה 20:00.
- **ישות פיקטיבית שסוגרת את הקלפיות הגדולות (הנגישות)** - ישות זו סוגרת את הקלפי בשעה 22:00.
- **ישות משקיף** - ישות זו תפקידה בעת ספירת הקולות להוסיף בהסתברות של 40% 50 קולות למפלגה שאליה שייכת. יצרנו 3 משקיפים אחד לכל אזור אשר בהסתברות 1/3 מרמים לטובת כל אחת מהמפלגות (ימין/ מרכז/ שמאל).

3.2 המשאבים שהוגדרו במודל

בכתיבת הקוד ניסינו למדל את הסימולציה באופן גנרי ככל האפשר. על מנת לבצע זאת יצרנו את המשאבים לפי מוטיב חוזר בו כל משאב מלווה באינדקסים אחריו שמייצגים את הנתונים הנדרשים בכדי לאפיין משאב זה.

משאב קלפי:

שם המשאב	הסבר	קיבולת (capacity)	אורך תור מקסימלי
kalpi_1i	3 קלפיות באזור צפון	1	Inf
kalpi_2i	9 קלפיות באזור מרכז	1	Inf
kalpi_3i	6 קלפיות באזור דרום	1	Inf

משאב דוכן:

שם המשאב	קיבולת (capacity)	אורך תור מקסימלי
stand_ijk	1	Inf

- i - מייצג את האזור: 1- צפון, 2- מרכז, 3- דרום
- j - מייצג את מספר הקלפי בכל אזור: בצפון 3 קלפיות, במרכז 9 קלפיות ובדרום 6 קלפיות.
- k - מייצג את סוג הדוכן (שמאל/ ימין/ מרכז): 1- שמאל, 2-מרכז, 3- ימין.

3.3 התכונות שהוגדרו במודל

תכונה	הסבר קצר
area	תכונת המציינת את האזור אליו נשלח הבוחר להצביע.
kalpi_num	תכונה המציינת את מספר הקלפי אליו נשלח הבוחר.
type	תכונה המציינת את אג'נדת הבוחר
Z	תכונה המציינת את מידת הקיצוניות של בוחר.
c("opinion_1", "opinion_2", "opinion_3")	<p>וקטור הדעה ההתחלתית של מצביע, ההסתברות לבחור בכל אחת מן המפלגות.</p> <p>opinion_1 - ההסתברות הראשונית של בוחר לבחור במפלגת שמאל.</p> <p>opinion_2 - ההסתברות הראשונית של בוחר לבחור במפלגת מרכז.</p> <p>opinion_3 - ההסתברות הראשונית של בוחר לבחור במפלגת ימין.</p>
c("new_opinion_1", "new_opinion_2", "new_opinion_3")	<p>וקטור הדעה ההתחלתית של מצביע, הסתברות לבחור בכל אחת מן המפלגות.</p> <p>new_opinion_1 - ההסתברות החדשה של בוחר לבחור במפלגת שמאל.</p> <p>new_opinion_2 - ההסתברות החדשה של בוחר לבחור במפלגת מרכז.</p> <p>new_opinion_3 - ההסתברות החדשה של בוחר לבחור במפלגת ימין.</p>
c("stands_order_1", "stands_order_2", "stands_order_3")	וקטור של סדר הדוכנים אליו הולך הבוחר.
max_opinion	הדעה המועדפת והמשוכללת של בוחר (עם מידת הקיצוניות).
kalpi_type	תכונה המציינת לאיזה סוג קלפי הבוחר ילך (גדולה/קטנה)

נספח 4- בדיקת נכונות המודל:

לצורך בדיקת נכונות המצב הקיים, נציג תיעוד של כל המסלולים ואת הישגיות שעובדות בהם, לפי סוגי הישגיות. נציג את כל המקרים האפשריים במודל ונציג את התאמתם להנחיות ולהנחות שלנו.

4.1 ישות מצביע

* קריאת האינדקסים מתבצעת משמאל לימין.

מצביע רגיל:

לשם ההמחשה נבחר להציג בוחר שנשלח לקלפי הגדולה.

	time	name	key	value	replication
50801	699.4898	large_kalpi_voter240	kalpi_type	2.00000000	1
50802	699.4898	large_kalpi_voter240	area	3.00000000	1
50803	699.4898	large_kalpi_voter240	kalpi_num	5.00000000	1
50804	699.4898	large_kalpi_voter240	type	3.00000000	1
50805	699.4898	large_kalpi_voter240	Z	0.05511739	1
50806	699.4898	large_kalpi_voter240	opinion_1	0.22244131	1
50807	699.4898	large_kalpi_voter240	opinion_2	0.47244131	1
50808	699.4898	large_kalpi_voter240	opinion_3	0.30511739	1
50809	699.4898	large_kalpi_voter240	new_opinion_1	0.22244131	1
50810	699.4898	large_kalpi_voter240	new_opinion_2	0.47244131	1
50811	699.4898	large_kalpi_voter240	new_opinion_3	0.30511739	1
50812	699.4898	large_kalpi_voter240	stands_order_1	1.00000000	1
50813	699.4898	large_kalpi_voter240	stands_order_2	3.00000000	1
50814	699.4898	large_kalpi_voter240	stands_order_3	2.00000000	1

בשלב הראשוני מוגדרות כל התכונות של הבוחר:

- ראשית, לפי שדה $Area = 3$ ניתן לראות שהבוחר נמצא במתחם דרום.
- מאחר ומדובר בבוחר ששייך לקלפי גדולה ($kalpi_type = 2$), ובמתחם דרום יש 6 קלפיות כאשר 4 מהן גדולות, הבוחר יכול לבחור בקלפיות 3-6 בלבד. בסופו של דבר הקלפי שנבחרה עבורו (שדה $kalpi_num$) היא קלפי מספר 5.
- לפי שדה $Type = 3$ ניתן לראות שהבוחר ימני בדעותיו, עם זאת מידת הקיצוניות $z = 0.05$ קטנה יחסית ועל כן, נצפה שדעתו נוטה לכיוון מרכז.
- השדה $Opinion_i$ כאשר $i = 1, 2, 3$ מציין את הדעה הראשונית של הבוחר. כפי שציפינו לראות, הבוחר נוטה לכיוון מפלגת מרכז משום ש: $Opinion_2 = 0.472$ שזו ההסתברות המקסימלית.
- נצפה שהסדר שבו הבוחר יבקר בדוכנים יהיה שמאל \leftarrow ימין \leftarrow מרכז. ואכן, שדה $stand_order_i$ המציין את סדר המעבר בדוכני המפלגות של בוחר מסוים הוא $1 \leftarrow 3 \leftarrow 2$ כפי שציפינו, כלומר הבוחר עובר בדוכנים שמאל, ימין ומרכז.

כעת נבדוק שהבוחר אכן בוחר את המשאבים המתאימים לו על פי תכונות אלו.

	name	start_time	end_time	activity_time	resource	replication
7207	large_kalpi_voter240	699.4898	702.1692	2.679462	stand_351	1
7252	large_kalpi_voter240	702.1692	705.6593	3.490103	stand_353	1
7320	large_kalpi_voter240	705.6593	711.5490	5.889629	stand_352	1
7331	large_kalpi_voter240	711.5490	712.8372	1.288276	kalpi_35	1

ראשית הבוחר הולך ל: stand_351 שהוא דוכן בדרום, בקלפי מספר 5 השייך למפלגת שמאל. לאחר מכן הבוחר עובר ל stand_353 שהוא דוכן בדרום, בקלפי מספר 5 השייך למפלגת ימין. הדוכן השלישי והאחרון הינו stand_352 שהוא דוכן בדרום, מספר 5 השייך למפלגת מרכז (המפלגה המועדפת עליו עפ"י וקטור ההסתברויות ההתחלתי). ניתן לראות שזמן הפעילות בכל דוכן גדול מאפס לכן נסיק שהיו נציגים פנויים כאשר הבוחר הגיע לדוכנים. לבסוף הבוחר הולך ל kalpi_35 הממוקמת באזור הדרום ומספרה 5.

בנוסף, דעתו של הבוחר משתנה תוך כדי המעבר בין הדוכנים השונים כפי שניתן לראות מטה

בטבלה:

50924	702.1692	large_kalpi_voter240	new_opinion_1	0.28786253	1
50925	702.1692	large_kalpi_voter240	new_opinion_2	0.43973069	1
50926	702.1692	large_kalpi_voter240	new_opinion_3	0.27240677	1
51275	705.6593	large_kalpi_voter240	new_opinion_1	0.17831006	1
51276	705.6593	large_kalpi_voter240	new_opinion_2	0.33017822	1
51277	705.6593	large_kalpi_voter240	new_opinion_3	0.49151172	1
51690	711.5490	large_kalpi_voter240	new_opinion_1	-0.03069946	1
51691	711.5490	large_kalpi_voter240	new_opinion_2	0.74819727	1
51692	711.5490	large_kalpi_voter240	new_opinion_3	0.28250219	1
51693	711.5490	large_kalpi_voter240	final_opinion_1	0.00000000	1
51694	711.5490	large_kalpi_voter240	final_opinion_2	0.72591216	1
51695	711.5490	large_kalpi_voter240	final_opinion_3	0.27408784	1

דעה של בוחר

ניתן לראות שבשום שלב אין שוויון בהסתברויות ועל כן הבוחר יכול לנטוש את הדוכנים פרט לדוכן של המפלגה המועדפת עליו, כלומר יכול לנטוש את הדוכנים שמאל וימין.

בזמן 699.4898 הבוחר מגיע למסלול המייצג את הדוכן הראשון שמאפשר נטישה (ישנו מסלול המייצג את הדוכן הראשון שאינו מאפשר זאת).

699.49: large_kalpi_voter240: im in stand 1 and i can aband it
700.618: large_kalpi_voter240: im in stand 1 and i can aband

על מנת לבדוק האם הבוחר נטש את הדוכן, נבדוק בהדפסות האם הוא נטש בין זמן ההגעה שלו לדוכן הראשון לבין זמן ההגעה שלו לדוכן השני.
ניתן לראות שהבוחר סיים את השירות בדוכן הראשון במלואו כלומר לא נטש.
ואכן ניתן לראות שבטבלת דעה של בוחר, דעתו על מפלגת שמאל עלתה מ- 0.222 ל- 0.287 לאחר השיחה עם הנציג בדוכן.

```
699.477: negishut_kalpi_voter987: i left stand 1
700.038: negishut_kalpi_voter990: i left stand 1
701.817: negishut_kalpi_voter989: i left stand 1
701.861: negishut_kalpi_voter991: i left stand 1
702.21: negishut_kalpi_voter992: i left stand 1
703.247: negishut_kalpi_voter995: i left stand 1
```

בזמן 702.1692 הבוחר מגיע למסלול המייצג את הדוכן השני שמאפשר נטישה.

```
702.169: large_kalpi_voter240: in in stand 2 and i can aband it
702.21: negishut_kalpi_voter992: in in stand 2 and i can aband it
```

גם כאן נבדוק האם הבוחר נטש לפני זמן הגעתו לדוכן השלישי.

```
701.44: negishut_kalpi_voter983: i left stand 2
702.229: large_kalpi_voter237: i left stand 2
703.076: negishut_kalpi_voter988: i left stand 2
704.127: negishut_kalpi_voter986: i left stand 2
706.088: large_kalpi_voter238: i left stand 2
```

כפי שציפינו הוא לא נטש את הדוכן, ואכן ניתן לראות שבטבלת דעה של בוחר, דעתו על מפלגת ימין עלתה מ- 0.272 ל- 0.491 לאחר השיחה עם הנציג בדוכן.

עבור כל דוכן קיימים שני מסלולים, אחד שמאפשר נטישה ואחד שלא, לעומת זאת בדוכן השלישי יש רק מסלול אחד שלא מאפשר נטישה. לכן, אנו יודעים בוודאות כי הבוחר לא נטש את הדוכן האחרון.
ואכן ניתן לראות שבטבלת דעה של בוחר, דעתו על מפלגת מרכז עלתה מ- 0.33 ל- 0.748 לאחר השיחה עם הנציג בדוכן.

- יש לציין שבסוף הדוכן מתבצע נרמול עבור וקטור הדעה של בוחר כך שיסכום ל-1.

מצביע ההולך לקלפי נגישה:

מצביע זה זהה לשאר המצביעים פרט לעובדה שהינו בעל עדיפות, ועל כן יכול לעקוף את התור לקלפי, נבדוק שזה אכן מתקיים.

לפי טבלת המשאבים, ניתן לראות כי מגיעות שתי ישויות חדשות.

	resource	time	server	queue	capacity	queue_size	system	limit	replication
143	kalpi_12	14.078494	1	0	1	Inf	1	Inf	1
148	kalpi_12	14.367908	1	1	1	Inf	2	Inf	1
175	kalpi_12	15.449126	1	2	1	Inf	3	Inf	1

שתי הישויות הללו הינן בוחר רגיל ובוחר נגישות, והן מגיעות בסדר הזה. נצפה לראות שבוחר הנגישות יעזוב את הקלפי לפני הבוחר הרגיל, ואכן הוא עוזב בזמן 19.377 שזה מוקדם יותר מזמן העזיבה של הבוחר הרגיל אשר עוזב בזמן 25.483.

	name	start_time	end_time	activity_time	resource	replication
127	large_kalpi_voter0	14.36791	25.48311	2.6063008	kalpi_12	1
81	negishut_kalpi_voter4	15.44913	19.37742	3.7173084	kalpi_12	1

4.2 מסלול של מנהל צפון / מרכז / דרום

מנהל אזור מגיע לקלפי אחת ל-3 שעות מתחילת הסימולציה לעשות תדרוך לעובדי הקלפיות. לשם בדיקת הנכונות של מודל הסימולציה נבדוק את המסלול אשר מנהל אזור צפון עושה ונשליך על שאר האחראים.

ראשית, מהטבלה שלפנינו אכן ניתן לראות כי אחראי על אזור הצפון מגיע חמש פעמים ביום הבחירות החל מזמן תחילת הסימולציה, כלומר החל מהשעה 7:00 בבוקר (השעות בהן הוא מגיע הן 7:00 , 10:00 , 13:00 , 16:00 , 19:00) נחدد כי בשעה 22:00 אחראי האזור לא מגיע מכיוון שהגענו לזמן סגירת הקלפיות. זמן השהות שלו במערכת שונה מהגעה להגעה, מכיוון שהוא תלוי בזמן תדרוך הקלפי שמתפלג משולשית.

	name	start_time	end_time	activity_time	finished	replication	waiting
2	north_manager0	0	7.579082	7.579082	TRUE	1	0.000000e+00
405	north_manager1	180	190.547234	10.547234	TRUE	1	-7.105427e-15
883	north_manager2	360	366.432837	6.432837	TRUE	1	-1.243450e-14
1336	north_manager3	540	549.544692	9.544692	TRUE	1	-3.552714e-14
1810	north_manager4	720	728.061973	8.061973	TRUE	1	3.197442e-14

ניתן לראות כי בזמן 540 האחראי קלפי צפון משנה את ה-Capacity ל-0.

	resource	time	server	queue	capacity	queue_size	system	limit	replication
11076	kalpi_13	535.8931	0	0	1	Inf	0	Inf	1
11087	kalpi_11	536.2842	1	0	1	Inf	1	Inf	1
11095	kalpi_12	536.7797	1	0	1	Inf	1	Inf	1
11106	kalpi_12	537.0642	1	1	1	Inf	2	Inf	1
11118	kalpi_11	537.7910	0	0	1	Inf	0	Inf	1
11125	kalpi_11	538.1922	1	0	1	Inf	1	Inf	1
11141	kalpi_11	539.3421	0	0	1	Inf	0	Inf	1
11143	kalpi_12	539.5012	1	0	1	Inf	1	Inf	1
11167	kalpi_11	540.0000	0	0	0	Inf	0	Inf	1
11168	kalpi_12	540.0000	1	0	0	Inf	1	Inf	1
11169	kalpi_13	540.0000	0	0	0	Inf	0	Inf	1

כפי שראינו בטבלה הקודמת זמן הפעילות של אחראי אזור הוא 10.520593, ובסיום זמן הפעילות שלו ניתן לראות שאכן הוא סוגר את הקלפיות כפי שמצופה.

11346	kalpi_11	550.5206	1	1	1	Inf	2	Inf	1
11347	kalpi_12	550.5206	1	2	1	Inf	3	Inf	1
11348	kalpi_13	550.5206	1	2	1	Inf	3	Inf	1

4.3 משקיף צפון/ מרכז/דרום

בזמן ספירת הקולות, משקיפי המפלגות עשויים להוסיף קולות למפלגה שלהם בסיכוי של 0.4 אחוז ופעם אחת בכל אזור. הצורה בה מדלנו את המשקיפים היה ע"י משקיף גנרי לכל אזור, שאם וכאשר הוא מוסיף קולות למפלגה שלו נגריל באופן שווה בין שלושת המפלגות הקיימות. כפי שניתן לראות בטבלה שלפנינו בזמן בשעה 10:00 נוצרים שלושה משקיפים גנריים לכל אזור, כמו כן ניתן לראות שהמקרה היחיד שבו בוצעה רמאות היא באזור דרום.

	name	start_time	end_time	activity_time	finished	replication	waiting
2249	mashkif_10	900	900	0	FALSE	1	0
2250	mashkif_20	900	900	0	FALSE	1	0
2251	mashkif_30	900	900	0	TRUE	1	0

על מנת לדעת עבור איזו מפלגה נוספו קולות במחוז דרום נסתכל על הטבלה שלפנינו:

62878	899.8755		party_1	640	1
62879	900.0000		party_1	690	1

בתמונה ניתן לראות כי אכן בוצעה רמאות ונוספו 50 קולות למפלגת שמאל במחוז דרום.

4.4. ישות פיקטיבית אשר סוגרת את הקלפיות

בשעה 20:00 ו-22:00 נוצרו 2 ישויות פיקטיביות אשר סוגרות את הקלפיות הגדולות והקטנות בהתאמה. עקב כך שפרמטר $Mon = 0$ לא ניתן לראות אותן בטבלת ההגעה אבל נשתמש בטבלת המשאבים ונראה כי אכן הדבר קורה. בסה"כ יש בידינו 6 קלפיות קטנות בכל האזורים. בטבלה שלפנינו בזמן סגירתן כלומר בשעה 20 בערב הקיבולת שלהן משתנה ל-0 כמו כן גם אורך התור.

16175	kalpi_11	780.0000	1	0	0	Inf	1	Inf	1
16176	kalpi_21	780.0000	1	0	0	Inf	1	Inf	1
16177	kalpi_22	780.0000	0	0	0	Inf	0	Inf	1
16178	kalpi_23	780.0000	1	0	0	Inf	1	Inf	1
16179	kalpi_31	780.0000	0	0	0	Inf	0	Inf	1
16180	kalpi_32	780.0000	0	0	0	Inf	0	Inf	1
16181	kalpi_11	780.0000	1	0	0	0	1	0	1
16182	kalpi_21	780.0000	1	0	0	0	1	0	1
16183	kalpi_22	780.0000	0	0	0	0	0	0	1
16184	kalpi_23	780.0000	1	0	0	0	1	0	1
16185	kalpi_31	780.0000	0	0	0	0	0	0	1
16186	kalpi_32	780.0000	0	0	0	0	0	0	1

נספח 5- מבחני טיב התאמה**מבחני טיב התאמה להתפלגות הגעת הבוחרים****טיפול בנתונים**

ראשית, חישבנו את ההפרש בין זמני ההגעה של הבוחרים. בשלב השני, על מנת לקבל את פרמטר הקצב, עשינו 1 חלקי ההפרש.

התאמה להתפלגות ידועה

בצענו מבחני טיב התאמה להתפלגויות ההגעה של בעלי זכות הבחירה - בוחרים בקלפיות קטנות ובוחרים בקלפיות גדולות.

נשים לב שהתפלגויות זמני ההגעה אינן יכולות להיות:

- התפלגות בטא, זאת משום שהערכים אינם באינטרוול $[0, 1]$.
- פואסון / בינומית / גיאומטרית, זאת משום שההתפלגות הינה רציפה.

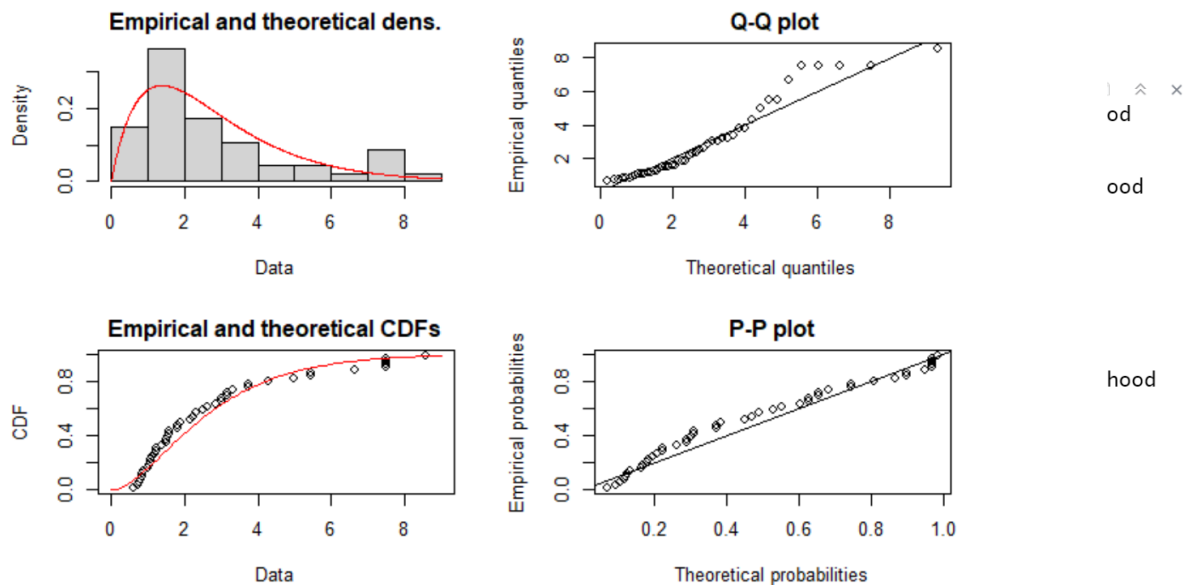
כיוון שמדובר בזמני הגעה חשדנו כי הנתונים מגיעים מהתפלגות מעריכית, מהתפלגות גמא (המקרה הכללי של התפלגות מעריכית) או מהתפלגות נורמלית. באמצעות R נאמוד את הפרמטרים של ההתפלגויות הנבחנות, כאילו הנתונים אכן מגיעים מהתפלגויות אלו.

1. התפלגות זמני ההגעה של הבוחרים מהקלפיות הקטנות

```
85 `r fitting1, include=TRUE}
86 expFit<-fitdist(final_large_rate,"exp") # fitting a exp distribution
87 summary(expFit) # summary function displays the results
88
89 normFit<-fitdist(final_large_rate,"norm")
90 summary(normFit)
91
92 gammaFit<-fitdist(final_large_rate,"gamma")
93 summary(gammaFit)
94
95 `r
```

נרצה מדד likelihood גבוה ומדדי AIC/BIC נמוכים. נסיק שהתפלגות גמא עם הפרמטרים $\text{shape}=1.9688020$ and $\text{rate}=0.6994063$ מתאימה לנתונים שלנו בצורה הטובה ביותר.

נוכל לראות זאת בצורה ויזואלית על ידי הגרפים הבאים:



2. התפלגות זמני ההגעה של הבוחרים מהקלפיות הגדולות

```

[[r fitting2, include=TRUE]
expFit2<-fitdist(final_mugbalut_rate,"exp") # fitting a exp distribution
summary(expFit2) # summary function displays the results

normFit2<-fitdist(final_mugbalut_rate,"norm")
summary(normFit2)

gammaFit2<-fitdist(final_mugbalut_rate,"gamma")
summary(gammaFit2)

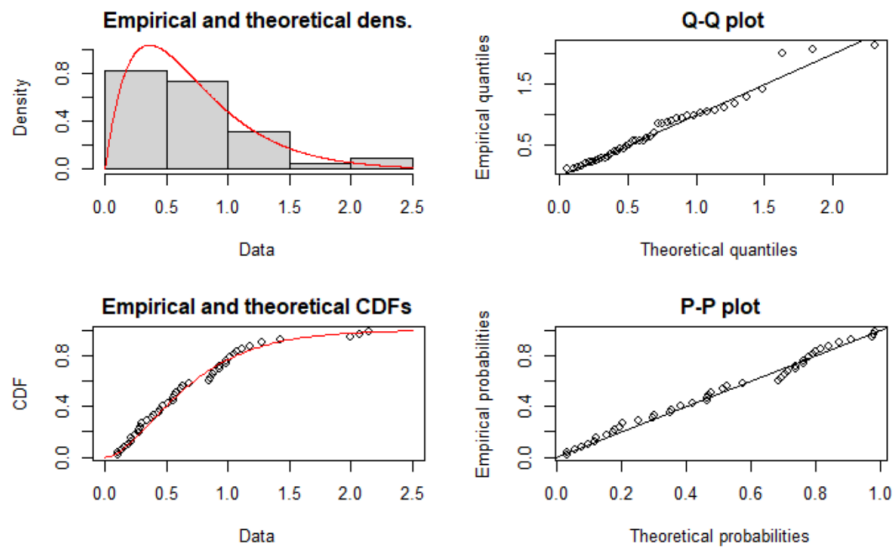
```

NaNs producedNaNs producedFitting of the distribution ' exp ' by maximum likelihood
 Parameters :
 Loglikelihood: -29.14852 AIC: 60.29704 BIC: 62.08123
NaNs producedNaNs producedFitting of the distribution ' norm ' by maximum likelihood
 Parameters :
 Loglikelihood: -32.34738 AIC: 68.69476 BIC: 72.26314
 Correlation matrix:
 mean sd
 mean 1 0
 sd 0 1

NaNs producedNaNs producedFitting of the distribution ' gamma ' by maximum likelihood
 Parameters :
 Loglikelihood: -23.89626 AIC: 51.79252 BIC: 55.3609
 Correlation matrix:
 shape rate
 shape 1.0000000 0.8817921
 rate 0.8817921 1.0000000

גם כאן, נרצה מדד likelihood גבוה ומדדי AIC/BIC נמוכים. נסיק שהתפלגות גמא עם הפרמטרים $\text{shape} = 2.023509$ and $\text{rate} = 2.835970$ מתאימה לנתונים שלנו בצורה הטובה ביותר.

נוכל לראות זאת בצורה ויזואלית על ידי הגרפים הבאים:



ביצוע מבחני טיב התאמה על ההתפלגויות הנבחות

לאחר שמצאנו התפלגויות מתאימות, נבחנו את ההשערה שהנתונים שלנו אכן הגיעו מהתפלגויות אלו על ידי מבחן סטטיסטי. נבחר במבחן קולמוגורוב סמירנוב (KS) משום שהוא יותר מדויק. נרשום את ההשערות (רלוונטי עבור הבחורים מהקלפיות הקטנות וכן מהקלפיות הגדולות):

H_0 : arrival \sim gamma(shape, rate)

H_1 : NOT

1. מבחן טיב התאמה לקלפיות הקטנות

```
##{r, goodness1 KS}
gammagof<- gofstat(gammaFit)%>%print()
gammagof$skstest%>%print()
```

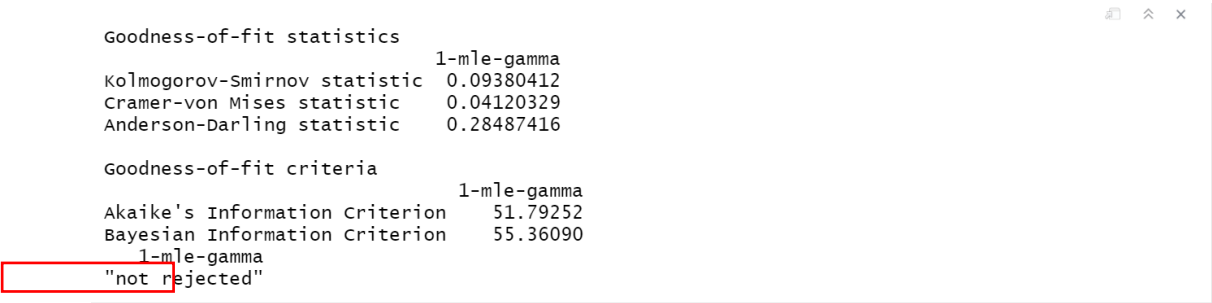
Goodness-of-fit statistics	
	1-mle-gamma
Kolmogorov-Smirnov statistic	0.135102
Cramer-von Mises statistic	0.164034
Anderson-Darling statistic	1.103976

Goodness-of-fit criteria	
	1-mle-gamma
Akaike's Information Criterion	184.8527
Bayesian Information Criterion	188.5530

1-mle-gamma
"not rejected"

השערת האפס אינה נדחית, לפיכך נסיק כי הנתונים אכן הגיעו מהתפלגות גמא.
p-value של המבחן הוא 0.388470762226611.

2. מבחן טיב התאמה לקלפיות הגדולות



```
Goodness-of-fit statistics      1-mle-gamma
Kolmogorov-Smirnov statistic  0.09380412
Cramer-von Mises statistic    0.04120329
Anderson-Darling statistic    0.28487416

Goodness-of-fit criteria      1-mle-gamma
Akaike's Information Criterion 51.79252
Bayesian Information Criterion 55.36090
1-mle-gamma
"not rejected"
```

השערת האפס אינה נדחית, לפיכך נסיק כי הנתונים אכן הגיעו מהתפלגות גמא.
p-value של המבחן הוא 0.315559767944528.

נספח 6- חלופות ומדדים לניתוח**6.1 חלופה מספר 1**

שיפור	תיאור	עלות
הוספת נציג למתחם מסוים	הוספת נציג לאזור צפון.	5000 ₪
העברת נציג המפלגה ממתחם אחד למתחם אחר	העברת 2 נציגים מאזור מרכז לאזור דרום.	ללא עלות
הוספת דוכן למתחם הצבעה מסוים (העלות היא ללא נציגים).	מעלה את הסיכוי בדעה ההתחלתית לבחור במפלגה ב1.0. הוספת דוכן לאזור מרכז.	10,000 ₪
העברת סדנא לנציגים ממתחם מסוים לשיפור יכולת השכנוע שלהם.	זמן עד שהבוחר מאבד סבלנות עולה ב2 דקות והסיכוי של הבוחר להצבעה למפלגה, במידה ולא עזב את הנציג, עולה בין 1.0 ל-4.0 סדנא לנציגי אזור מרכז.	15,000 ₪ למתחם
סה"כ		30,000 ₪

6.2 חלופה מספר 2

עלות	תיאור	שיפור
15,000 ₪ למתחם	זמן עד שהבוחר מאבד סבלנות עולה ב2 דקות והסיכוי של הבוחר להצבעה למפלגה, במידה ולא עזב את הנציג, עולה בין 1.0 ל- 4.0 סדנא לנציגי אזור מרכז.	העברת סדנא לנציגים ממתחם מסוים לשיפור יכולת השכנוע שלהם
12,000 ₪	מעלה אחוזים ב12.0 את הדעה ההתחלתית והסיכוי להצביע למפלגה.	השקעה בתעמולה תקשורתית ביום הבחירות
27,000 ₪		סה"כ

6.3. כמות הבוחרים המגיעים לקלפיות

	value	count(value)
1	1	751
2	2	761
3	3	745

6.4 טבלת כמות הנוטשים בדוכנים

	resource	count(value)
3	stand_131	62
2	stand_121	50
1	stand_111	43
18	stand_361	35
15	stand_331	28
16	stand_341	25
17	stand_351	23
5	stand_221	22
14	stand_321	22
8	stand_251	20
13	stand_311	19
10	stand_271	18
7	stand_241	17
4	stand_211	16
6	stand_231	13
9	stand_261	13
12	stand_291	12
11	stand_281	11

6.5 תיעוד קוד חלופות

חלופה 1:

עם תקציב של 30,000, ולאחר בדקת נכונות המודל ומציאת צווארי הבקבוק (סעיף 6.4), מצאנו לנכון להגדיר את השינויים הבאים למפלגת שמאל:

- הזזת שני נציגים מאזור מרכז אל אזור הדרום (ללא תשלום).
 - קניית נציג נוסף לאזור הצפון בעלות של 5000 ₪.
- הביטוי לשני שינויים אלה מגיע ביצירת המשאב, כאשר אנו מגדירים שבאזור הדרום יש בשני נציגים יותר, ובאזור המרכז יש שני נציגים פחות:

```
simulation_time <- 15*60 + 20
north_agents <- c(6, 5, 4) # number of agents at each stand of each kalpi of each area
south_agents <- c(5, 5, 5)
```

- הוספת דוכן לאזור מרכז משום שלאחר בדיקה, רוב האנשים מגיעים לשם - בעלות של 10,000 ₪.

הוספת דוכן מעלה את הסיכוי בדעה ההתחלתית לבחור במפלגה ב-0.1, ועל כן ביצענו בדיקה בשלב הגדרת הדעה ההתחלתית – האם הבוחר באזור בו בחרנו להציב דוכן נוסף, ואם כן, וקטור הדעה ההתחלתית יהיה שונה, ולמפלגת שמאל יהיה יתרון של עוד 0.1:

```
set_attribute(keys = c("opinion_1", "opinion_2", "opinion_3"), values = function () opinion_with_z
(get_attribute(elections, key = "type"), get_attribute(elections, key = "z"), get_attribute(elections, key = "area")))%>%
set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () opinion_with_z
(get_attribute(elections, key = "type"), get_attribute(elections, key = "z"), get_attribute(elections, key = "area")))%>%
```

```
opinion_with_z<-function (type, z, area){
  opinion <- c(0.25, 0.5 , 0.25) # initial opinion
  if (area == 2)
    opinion <- c(0.35, 0.45 , 0.20) # initial opinion if the voter is in the center area
  for (i in 1 : 3) {
    if (i == type)
      opinion[i] <- opinion[i] + z
    else opinion[i] <- opinion[i] - (z/2)
  } # for
  return (opinion)
} # a function that calculates the opinion vector of a voter with the z
```

- סדנה לשיפור יכולות השכנוע של הנציגים במפלגה באזור צפון בעלות של 15,000 ₪.
- באמצעות הסדנה, זמן עד שהבוחר מאבד סבלנות עולה ב-2 דקות והסיכוי של הבוחר להצבעה למפלגה, במידה ולא עזב את הנציג, עולה בין 0.1 ל-0.4.
- על כן, בשלב זה, בדקנו עבור כל הגעה של בוחר אל דוכן, האם הוא בדוכן של מפלגת שמאל באזור הצפון. לאחר בדיקה זו, שלחנו את הישויות העונות על הקריטריון למסלול ייחודי בו זמן הנטישה שונה מזה של שאר הישויות, וגם פונקציית שינוי וקטור הדעה:

```
stand1_renege<- trajectory ("stand1_renege")%>%
  branch(option = function () isLeft (get_attribute(elections, key="stands_order_1"), get_attribute(elections, key="area")), continue = FALSE, stand1_renege_left)%>%
```

```
isLeft<- function (stand, area){
  if (stand ==1 & area==1)
    return (1)
  else return (0)
} #Checks if a voter is in the stand of the left party
```

בעת הגעה אל מסלול עם אפשרות נטישה, נבדוק תחילה האם העמדה הראשונה בסבב העמדות של החובר היא הדוכן של מפלגת שמאל. אם כן, פונקציית isLeft בbranch תשלח את הבוחר אל מסלול אחר שבו זמן הנטישה שונה:

```
stand1_renege_left<- trajectory ("stand1_renege_left")%%
simmer::select(resources = function () get_stand_resource(get_attribute(elections, key = "area"), get_attribute(elections, key = "kalpi_num"), get_attribute(elections, key = "stands_order_1")), id = 11)%%
seize_selected(amount = 1, id = 11, continue = FALSE, reject = stand1_reject)%%
renege_init = function () runif(1, 3.5, 8.5) , out = stand1_abandon)%%
timeout(function () runif(1, 2, 6))%%
renege_abort()%%
release_selected(amount = 1, id = 11)%%
#סיום תהליך
set_attribute(keys = c("new_opinion_1", "new_opinion_2", "new_opinion_3"), values = function () updated_values_left(get_attribute(elections, key = "new_opinion_1"), get_attribute(elections, key = "new_opinion_2"))
join(traj_stand_2)
```

בנוסף, אם הבוחר לא נטש את העמדה, לבסוף יגיע לשורה בה אנו משנים את וקטור העדיפויות שלו, ובמקרה זה פונקציית שינוי וקטור העדיפויות תהיה שונה משל שאר הבוחרים:

```
updated_values_left <- function (opinion_1, opinion_2, opinion_3, stand_party) {
  x <- runif(1, 0.1, 0.4)
  y <- x/2
  return(c(opinion_1 + x, opinion_2 - y, opinion_3 - y))
} # if a voter didn't abandon a stand, we will raise the probabilities of this stand
```

חלופה 2:

בחלופה זו, לאחר בדיקת נכונות המודל ומציאת צווארי הבקבוק (סעיף 6.4), מצאנו לנכון להגדיר

את השינויים הבאים למפלגת שמאל:

- השארת מספר הנציגים בכל מתחם כמו שהוא.
 - סדנה לשיפור יכולות השכנוע של הנציגים במפלגה באזור צפון – בעלות של 15,000 ₪ ביצענו בדומה לחלופה 1.
 - השקעה בתעמולה תקשורתית ביום הבחירות בעלות של 12,000 ₪ מעלה ב0.12 את הדעה ההתחלתית והסיכוי להצביע למפלגה.
- מידלנו שיפור זה על ידי כך שבשלב קביעת התכונות הראשוניות ווקטור הדעה הראשוני, יתבצעו השינויים בהתאם לשיפור:

```
opinion_with_z<-function (type, z){  
  opinion <- c(0.37, 0.44 , 0.19) # initial opinion  
  for (i in 1 : 3) {  
    if (i == type)  
      opinion[i] <- opinion[i] + z  
    else opinion[i] <- opinion[i] - (z/2)  
  } # for  
  return (opinion)  
} # a function that calculates the opinion vector of a voter with the z
```

נספח 7 – חישוב מספר ריצות, תיעוד קוד והסבר מילולי וכמות

תיעוד קוד:

ראשית, נבצע קריאה של שלושה קבצי אקסל, אחד עבור כל חלופה, כאשר החלופות הן המצב הקיים, חלופה א' וחלופה ב'. בכל קובץ מצויים שני וקטורים המתאימים לשני המדדים שבחרנו, עבור מספר ריצות התחלתי של $n_0 = 30$ ריצות.

בנוסף, הגדרנו את הפרמטרים הנדרשים: $\gamma = 0.12, \alpha = 0.10, \alpha_i = 0.05, n = 30$.

```
## Current state
current_state <- read.csv(choose.files(), header=TRUE)

## Alternative 1
Alternative_1 <- read.csv(choose.files(), header=TRUE)

## Alternative 2
Alternative_2 <- read.csv(choose.files(), header=TRUE)

## Data for calculations
gamma<-0.12
alfa<-0.1
alfa_i<-alfa/2
n<-30
```

עבור כל מדד של כל חלופה (סה"כ 6) חישבנו את הדיוק היחסי $\frac{\delta}{\bar{x}} = \frac{t_{30-1, 1-\frac{0.05}{2}} \cdot \frac{sd}{\sqrt{30}}}{avg}$ בצורה הבאה:

1. חישוב \bar{x} של הוקטור.
2. חישוב δ , רווח סמך חלקי, על ידי לקיחת הגבול העליון של רווח הסמך והחסרת \bar{x} ממנו.
3. חישוב הדיוק היחסי על ידי חילוק התוצאה המתקבלת בסעיף 2 בזו שמתקבלת בסעיף 1.
4. אם הדיוק היחסי קטן מ- $\frac{\gamma}{1+\gamma}$, מספר הריצות ההתחלתי תקין.
5. אחרת, מספר הריצות הנדרש הינו: $n_0 * (\delta_0 / \delta_{total})^2$.

לשם ההמחשה, להלן תיעוד קוד של בדיקת הדיוק היחסי עבור שלוש המדדים של המצב הקיים.

```
#----- current state -----#
# abandon
test_abandon_current <- t.test(x= current_state$abandon_measure,y=NULL, alternative="two.sided",conf.level=0.95)
avg_current_abandon<-test_abandon_current$estimate
# דיוק סמך
delta_current_abandon<-test_abandon_current$conf.int[2]-avg_current_abandon
# דיוק יחסי
relative_accuracy_current <- delta_current_abandon/avg_current_abandon
print (relative_accuracy_current)
if(relative_accuracy_current<=gamma/(1+gamma)) {print("amount of runs is OK")}
if (relative_accuracy_current>(gamma/(1+gamma))) paste("amount of runs needed is: ",
,n*(delta_current_abandon/(avg_current_abandon*gamma/(1+gamma)))^2)

# votes
test_votes_current <- t.test(x= current_state$votes_measure,y=NULL, alternative="two.sided",conf.level=0.95)
avg_current_votes<-test_votes_current$estimate
# דיוק סמך
delta_current_votes<-test_votes_current$conf.int[2]-avg_current_votes
# דיוק יחסי
relative_accuracy_current3 <- delta_current_votes/avg_current_votes
print (relative_accuracy_current3)
if(relative_accuracy_current3<=gamma/(1+gamma)) {print("amount of runs is OK")}
if (relative_accuracy_current3>(gamma/(1+gamma))) paste("amount of runs needed is: ",
,n*(delta_current_votes/(avg_current_votes*gamma/(1+gamma)))^2)
```

קוד זה נכתב עבור שתי החלופות הנותרות.

הסבר התוצאות:

אלטרנטיבה/ מדד	הקטנת אחוז הנוטשים מכלל המבקרים בדוכן המפלגה	הגדלת אחוז הקולות למפלגת שמאל מכלל הקולות
מצב קיים	0.01561379 ✓	0.02165837 ✓
חלופה 1	0.01229424 ✓	0.01962857 ✓
חלופה 2	0.01568906 ✓	0.01925472 ✓

ניתן לראות שעבור $n_0 = 30$ כל התוצאות שהתקבלו עומדות בדיוק היחסי שהוגדר קודם לכן:

$$\frac{\gamma}{1 + \gamma} = 0.1071$$

לכן, מספר הריצות תקין ונשאר בעינו.

השוואת מודלים (מבחן t מזווג):**מבחן השערות:**

$$H0: x - y = 0$$

$$H1: x - y \neq 0$$

במבחן זה נרצה לקבל את ההשערה האלטרנטיבית, זאת משום שאנו רוצים שיהיה שוני ניכר בין החלופות.

נניח מספר הנחות:

1. יש תלות בין ריצות מקבילות בשתי הסדרות – מתקבל על ידי הרצת `set.seed()`. זהה.
2. אין תלות בין ריצות בכל חלופה בפני עצמה – מתקבל על ידי הרצת `set.seed(const + i)` שונה בכל הרצה.
3. ניתן להניח שוויון שוניות בין הסדרות.
4. התוצאות מתפלגות נורמאלית –
 - 4.1. עבור מדד אחוז הנוטשים – זהו מדד מסוג פרופורציה המחושב בכל הרצה עבור כל ישות, ולכן כאשר מספר הריצות רב ניתן להניח נורמאליות.
 - 4.2. עבור מדד אחוז הקולות – כנ"ל.

סדר פעולות לביצוע המבחן:

1. נחשב את ההפרש בין תוצאות של כל שתי הרצות מקבילות על ידי חיסור שני וקטורים.

```
# step 1 - calculate the difference between each to parallel runs:
# מספר ההשונות הכולל = מספר ההשונות * מספר המדדים = 6 = 2 * 3

dif_abandon_p_a1 <- find_difference_vec(current_state[,2], Alternative_1[,2])
dif_votes_p_a1 <- find_difference_vec(current_state[,4], Alternative_1[,4])

dif_abandon_p_a2 <- find_difference_vec(current_state[,2], Alternative_2[,2])
dif_votes_p_a2 <- find_difference_vec(current_state[,4], Alternative_2[,4])

dif_abandon_a1_a2 <- find_difference_vec(Alternative_1[,2], Alternative_2[,2])
dif_votes_a1_a2 <- find_difference_vec(Alternative_1[,4], Alternative_2[,4])
```

2. נחשב ממוצע וסטיית תקן להפרש:

```

mean_abandon_p_a1 <- mean(dif_abandon_p_a1)
sd_abandon_p_a1 <- sd(dif_abandon_p_a1)

mean_votes_p_a1 <- mean(dif_votes_p_a1)
sd_votes_p_a1 <- sd(dif_votes_p_a1)

mean_abandon_p_a2 <- mean(dif_abandon_p_a2)
sd_abandon_p_a2 <- sd(dif_abandon_p_a2)

mean_votes_p_a2 <- mean(dif_votes_p_a2)
sd_votes_p_a2 <- sd(dif_votes_p_a2)

mean_abandon_a1_a2 <- mean(dif_abandon_a1_a2)
sd_abandon_a1_a2 <- sd(dif_abandon_a1_a2)

mean_votes_a1_a2 <- mean(dif_votes_a1_a2)
sd_votes_a1_a2 <- sd(dif_votes_a1_a2)

```

3. חילוק אלפא במספר ההשוואות שאנו עושים, כאשר מספר ההשוואות הכולל הינו מספר החלופות (מצב קיים, חלופה 1, חלופה 2) * מספר המדדים (אחוז נוטשים, אחוז קולות למפלגת שמאל), כלומר 6 השוואות.

```

# step 3 - calculate alpha i -> alpha_total/num of comparisions
alpha_i <- 0.1/6

```

4. חישוב רווח הסמך עבור כל אחד מהמדדים:

```

paired_test1 <- t.test(x=current_state[,2],y=Alternative_1[,2],alternative = "two.sided", paired = T, var.equal = T, conf.level = 1-alpha_i)
print(paired_test1)
paired_test2 <- t.test(x=current_state[,4],y=Alternative_1[,4],alternative = "two.sided", paired = T, var.equal = T, conf.level = 1-alpha_i)
print(paired_test2)
paired_test3 <- t.test(x=current_state[,2],y=Alternative_2[,2],alternative = "two.sided", paired = T, var.equal = T, conf.level = 1-alpha_i)
print(paired_test3)
paired_test4 <- t.test(x=current_state[,4],y=Alternative_2[,4],alternative = "two.sided", paired = T, var.equal = T, conf.level = 1-alpha_i)
print(paired_test4)
paired_test5 <- t.test(x=Alternative_1[,2],y=Alternative_2[,2],alternative = "two.sided", paired = T, var.equal = T, conf.level = 1-alpha_i)
print(paired_test5)
paired_test6 <- t.test(x=Alternative_1[,4],y=Alternative_2[,4],alternative = "two.sided", paired = T, var.equal = T, conf.level = 1-alpha_i)
print(paired_test6)

```

כללי ההחלטה שהנחנו אותנו האם לבחור חלופה מסוימת התקבלו על סמך רווח הסמך:

- אם רווח הסמך חיובי בכל תחום ערכיו – חלופה 1 גדולה מחלופה 2.
- אם רווח הסמך שלילי בכל תחום ערכיו – חלופה 2 גדולה מחלופה 1.
- אם רווח הסמך כולל בתוכו את הערך 0, נהיה אישיים לקבלת ההחלטה באיזו חלופה לבחור.

את המדד הראשון, אחוז הנוטשים, נרצה למזער ועל כן נבחר בחלופה הקטנה. לעומת זאת, את המדד השני, אחוז הקולות במפלגת שמאל, נרצה למקסם ועל כן נבחר בחלופה הגדולה.

תוצאות המבחן, כפי שפורטו בסעיפים הקודמים של הדו"ח, הכריעו באופן מובהק על חלופה 2 כהחלופה העדיפה על פני חלופה 1 והמצב הקיים.

נספח 8- הפונקציות במודל הסימולציה**8.1 הפונקציות במצב הקיים**פונקציית trimmedNorm

השתמשנו בפונקציה זו בכדי לבצע דגימה של זמני ההצבעה, הפונקציה מקבלת סטיית תקן וממוצע ודוגמת מספר מהתפלגות הנורמלית הקטומה.

```
trimmedNorm <- function(mu,sd){
  while(TRUE){
    sample <- rnorm(1, mu, sd)
    if (sample > 0)
      return (sample)
  }
} # trimmedNorm
```

פונקציית set_max_opinion

פונקציית set_max_opinion מקבלת את כל הסתברויות של בוחר לפי סדר:

1-opinion- הסיכוי של הבוחר לבחור במפלגת שמאל.

2-opinion- הסיכוי של הבוחר לבחור במפלגת מרכז.

3-opinion- הסיכוי של הבוחר לבחור במפלגת ימין.

הפונקציה מחזירה את הדעה המקסימלית מבין כולן.

```
set_max_opinion <- function (opinion_1, opinion_2, opinion_3){
  opinion_vec <- c(opinion_1, opinion_2, opinion_3)
  max_opinion <- max(opinion_vec)
  return(max_opinion)
} # returns the highest opinion of a voter
```

פונקציית my_area

הפונקציה מחזירה בהסתברות שווה את האזור אליו הבוחר הולך.

```
my_area <- function () {
  area <- sample(1:3, 1)
  return (area)
} # returns the area of the voter, equal probability to each area
```

פונקציית my_kalpi

הפונקציה מקבלת אזור ואת סוג הקלפי (גדולה/ קטנה), ומחזירה את מספר הקלפי בהתאם לאזור

ולסוג הקלפי.

```
my_kalpi <- function (area, kalpi_type) {
  kalpi <- switch(
    paste0(area, kalpi_type),
    "11" = 1,
    "12" = sample(2:3, 1),
    "21" = sample(1:3, 1),
    "22" = sample(4:9, 1),
    "31" = sample(1:2, 1),
    "32" = sample(3:6, 1)
  ) # switch
  return (kalpi)
} # returns the number of kalpi of each voter, according to area and type of kalpi
```

פונקציית get kalpi resource

הפונקציה מקבלת אזור ואת סוג הקלפי (גדולה/ קטנה), ומחזירה את משאב הקלפי כך: kalpi_ij
כאשר:

i- מסמן את האזור.

j- מסמן את מספר הקלפי באותו אזור.

```
get_kalpi_resource <- function (area, kalpi_num){
  return (paste0("kalpi_", area, kalpi_num))
} # a function that returns the kalpi resource
```

פונקציית get stand resource

הפונקציה מקבלת אזור, מספר קלפי ומפלגה ומחזירה את משאב הדוכן כך: stand_xij
כאשר x- מסמן את האזור בו ממקום הדוכן.


i- מסמן את מספר הקלפי בה ממקום הדוכן.

j- מסמן את מספר המפלגה.

```
get_stand_resource <- function (area, kalpi_num, party) {
  return (paste0("stand_", area, kalpi_num, party))
} # a function that returns the stand resource
```

פונקציית opinion with z

הפונקציה מחשבת את סוג הבוחר ואת מידת הקיצוניות שלו ומחזירה את הדעה שלו לפי הדעה ההתחלתית שלו ומידת הקיצוניות.

```
opinion_with_Z<-function (type, Z){
  opinion <- c(0.25, 0.5 , 0.25) # initial opinion
  for (i in 1 : 3) { # for
  return (opinion)
} # a function that calculates the opinion vector of a voter with the Z
```

פונקציית density z

הפונקציה מחשבת את מידת הקיצוניות של הבוחר.

```
densityZ <- function (){ # calculations in the final report
  u <- runif(1,0,1)
  if (u < 0.5)
    return (u/8)^(1/3)
  else
    return (u/2)
} # a function that calculates the extreme level of a voter - COMPOSITION of f(x)
```

פונקציית stand order

הפונקציה מקבלת את שלושת ההסתברויות של בוחר לבחור בכל אחת מהמפלגות, ומחזירה וקטור של סדר הדוכנים בהם בוחר יעבור, הבוחר ילך קודם כל למפלגה שההסתברות שלה לבחור בו הקטן ביותר, יעבור אח"כ למפלגה השנייה ולבסוף הבוחר יעבור למפלגה המועדפת עליו.

```
stand_order <- function (opinion_1, opinion_2, opinion_3){
  opinion_vec <- c(opinion_1, opinion_2, opinion_3)
  order_vec <- c(1,3,2) # default order, center voter
  max_index <- which(opinion_vec == max(opinion_vec), arr.ind = TRUE) # המפלגה שהדעה שלו מוטית לכיוונה כלומר הסיכוי הכי גבוה
  if (max_index == 1)
    order_vec <- c(3,2,1) # left voter
  if (max_index == 3)
    order_vec <- c(2,1,3) # right voter
  return(order_vec)
} # a function that returns the order of the stands of a voter
```

פונקציית is equal

הפונקציה מקבלת את הדעה של בוחר והסתברות המקסימלית מווקטור ההסתברויות ומחזירה 1 אם אכן מדובר בדעה המקסימלית או 2 המסמן false המציין כי לא מדובר בדעה המקסימלית.

```
isEqual <- function (max_opinion, opinion){
  if (max_opinion == opinion)
    return(1)
  else return(2)
} # returns T/F if there is equal probabilities with the highest party of a voter
```

פונקציית updated values

הפונקציה מקבלת את ההסתברות של בוחר לבחור בכל אחת מן המפלגות (ימין/ מרכז/ שמאל), ואת דוכן המפלגה הנוכחית בו נמצא הבוחר. הפונקציה מחזירה וקטור הסתברויות חדש בו נעלה את ההסתברות של הבוחר לבחור במפלגה שהדוכן נמצא בבעלותה ובהתאם להוריד את הסתברויות לבחור בשאר המפלגות.

```
updated_values <- function (opinion_1, opinion_2, opinion_3, stand_party) {
  x <- runif(1, 0, 0.5)
  y <- x/2
  if (stand_party == 1)
    return(c(opinion_1 + x, opinion_2 - y, opinion_3 - y))
  if (stand_party == 2)
    return(c(opinion_1 - y, opinion_2 + x, opinion_3 - y))
  if (stand_party == 3)
    return(c(opinion_1 - y, opinion_2 - y, opinion_3 + x))
} # if a voter didn't abandon a stand, we will raise the probabilities of this stand
```

פונקציית updated values renege

הפונקציה נקראת כאשר בוחר נוטש את הדוכן שבו היה. הפונקציה מקבלת את ההסתברות של בוחר לבחור בכל אחת מן המפלגות (ימין/ מרכז/ שמאל), ואת דוכן המפלגה הנוכחית בו נמצא הבוחר. הפונקציה מחזירה וקטור הסתברויות חדש בו נוריד את ההסתברות של הבוחר לבחור במפלגה שהדוכן נמצא בבעלותה ובהתאם תעלה את הסתברויות לבחור בשאר המפלגות.

```
updated_values_renege <- function (opinion_1, opinion_2, opinion_3, stand_party) {
  x <- runif(1, 0, 0.25)
  y <- x/2
  if (stand_party == 1)
    return(c(opinion_1 - x, opinion_2 + y, opinion_3 + y))
  if (stand_party == 2)
    return(c(opinion_1 + y, opinion_2 - x, opinion_3 + y))
  if (stand_party == 3)
    return(c(opinion_1 + y, opinion_2 + y, opinion_3 - x))
} # if a voter abandon a stand, we will decline the probabilities of the other stands
```

פונקציית -opinion check

הפונקציה מקבלת את ההסתברות של בוחר לבחור בכל אחת מן המפלגות (ימין/ מרכז/ שמאל), ומנרמלת את הערכים. במידה ויש ערך שקטן מ-0, ההסתברות לבחור במפלגה זו יהיה 0. הפונקציה מחזירה וקטור דעה מנורמל.

```
opinion_check <-function (opinion_1, opinion_2, opinion_3){
  vec <- c(opinion_1, opinion_2, opinion_3)
  if (vec[1] < 0)
    vec[1] <- 0
  if (vec[2] < 0)
    vec[2] <- 0
  if (vec[3] < 0)
    vec[3] <- 0
  return(vec/sum(vec))
} # a function that normalize the vector of opinions
```

פונקציית set capacity

פונקציה זו מקבלת מסלול, אזור, מספר קלפי ואת קיבולת הקלפי הספציפית, ומעדכנת את קיבולת קלפי זו.

```
setCapacity <- function (trajectory, area, kalpi_num, capacity_value){
  updatedPath <- NA
  for (i in 1:kalpi_num){
    updatedPath <- set_capacity(trajectory, resource = paste0("kalpi_", area, i), value = capacity_value)
  }
  return(updatedPath)
} # a function that set the capacity of the kalpi, called by the area manager.
} # assumption: if there is someone in service, he will finish first and then the kalpi will closed.
```

פונקציית my vote

הפונקציה מקבלת את שלושת ההסתברויות החדשות של בוחר לבחור בכל אחת מן המפלגות, הפונקציה מחזירה את המפלגה המועדפת של הבוחר.

```
my_vote <- function (final_opinion_1, final_opinion_2, final_opinion_3){
  final_opinion_vec <- c(final_opinion_1, final_opinion_2, final_opinion_3)
  vote <- rdiscrete(1, probs = final_opinion_vec, values = 1:3)
  return(paste0("party_", vote))
} # returns the winner party of a voter
```

פונקציית close small kalpi

הפונקציה שמעדכנת את הקיבולות ואורך תור של הקלפיות הקטנות ל-0, בסוף יום ע"י הישות הפיקטיבית שיצרנו.

```
close_small_kalpi <- function (trajectory) {
  updatedPath <- set_capacity(trajectory, resource = "kalpi_11", value = 0)%>%
    set_capacity(resource = "kalpi_21", value = 0)%>%
    set_capacity(resource = "kalpi_22", value = 0)%>%
    set_capacity(resource = "kalpi_23", value = 0)%>%
    set_capacity(resource = "kalpi_31", value = 0)%>%
    set_capacity(resource = "kalpi_32", value = 0)%>%

  set_queue_size(resource = "kalpi_11", value = 0)%>%
  set_queue_size(resource = "kalpi_21", value = 0)%>%
  set_queue_size(resource = "kalpi_22", value = 0)%>%
  set_queue_size(resource = "kalpi_23", value = 0)%>%
  set_queue_size(resource = "kalpi_31", value = 0)%>%
  set_queue_size(resource = "kalpi_32", value = 0)

  return(updatedPath)
}# a function that set the capacity and the queue size of the small kalpis to 0
```

פונקציית close large kalpi

הפונקציה שמעדכנת את הקיבולות ואורך תור של הקלפיות הגדולות ל-0, בסוף יום ע"י הישות הפיקטיבית שיצרנו.

```
close_large_kalpi <- function (trajectory) {
  updatedPath <- set_capacity(trajectory, resource = "kalpi_12", value = 0)%>%
    set_capacity(resource = "kalpi_13", value = 0)%>%
    set_capacity(resource = "kalpi_24", value = 0)%>%
    set_capacity(resource = "kalpi_25", value = 0)%>%
    set_capacity(resource = "kalpi_26", value = 0)%>%
    set_capacity(resource = "kalpi_27", value = 0)%>%
    set_capacity(resource = "kalpi_28", value = 0)%>%
    set_capacity(resource = "kalpi_29", value = 0)%>%
    set_capacity(resource = "kalpi_33", value = 0)%>%
    set_capacity(resource = "kalpi_34", value = 0)%>%
    set_capacity(resource = "kalpi_35", value = 0)%>%
    set_capacity(resource = "kalpi_36", value = 0)%>%

  set_queue_size(resource = "kalpi_12", value = 0)%>%
  set_queue_size(resource = "kalpi_13", value = 0)%>%
  set_queue_size(resource = "kalpi_24", value = 0)%>%
  set_queue_size(resource = "kalpi_25", value = 0)%>%
  set_queue_size(resource = "kalpi_26", value = 0)%>%
  set_queue_size(resource = "kalpi_27", value = 0)%>%
  set_queue_size(resource = "kalpi_28", value = 0)%>%
  set_queue_size(resource = "kalpi_29", value = 0)%>%
  set_queue_size(resource = "kalpi_33", value = 0)%>%
  set_queue_size(resource = "kalpi_34", value = 0)%>%
  set_queue_size(resource = "kalpi_35", value = 0)%>%
  set_queue_size(resource = "kalpi_36", value = 0)

  return(updatedPath)
}# a function that set the capacity and the queue size of the large kalpis to 0
```


פונקציית cheat

הפונקציה מעלה בהסתברות של $1/3$ במידה ומשקיפי המפלגות עלולים לנצל את העובדה שמזכירי הקלפי מוסחים, ולהוסיף 50 קולות למפלגה שלהם.

```
cheat <- function () {
  party <- rdiscrete(1, c(1/3, 1/3, 1/3), c(1,2,3))
  return(paste0("party_", party))
} # returns the party to which we add 50 votes
```

פונקציית rejected left

הפונקציה הינה פונקציית עזר להגדרת התכונה האם בוחר לא הצליח לקבל שירות בדוכן של מפלגת שמאל, כלומר כלל הנציגים היו תפוסים.

```
rejected_left <- function (party){
  if (party == 1)
    return (1)
  else return (0)
}
```

פונקציית aband left

הפונקציה הינה פונקציית עזר להגדרת התכונה האם בוחר נטש את מפלגת שמאל בה בחרנו או לא.

```
aband_left <- function (party){
  if (party == 1)
    return (1)
  else return (0)
}
```