



אוניברסיטת בן-גוריון בנגב
Ben-Gurion University
of the Negev



פרויקט קורס סימולציה - בית החולים הווטרינרי



עידן דיוה 208385930
אדי אלימלך כהן 209496843
קרולינה פיין 318402930



תאריך הגשה : 7.4.24

תוכן עניינים :

3.....	3.תקציר
3.....	4.מבוא
3-4.....	4.1.תיאור המערכת הנחקרת
4.....	4.2.מטרות
5.....	5.מודל סימולציה
5-7.....	5.1.תיאור המודל
7.....	5.2.הנחות
7.....	6.ניתוח מצב קיים
7-8.....	6.1.בחירת מדדים לניתוח
9-32	7.נספחים

3. תקציר - בפרויקט זה, בוצע מידול סימולציה של בית החולים הוטרני האוניברסיטאי. המטרה

הייתה לבחון את יעילותו של בית החולים הוטרני השייך לפקולטה לחקלאות של האוניברסיטה העברית אשר מטפל בחיות מחמד, בחיות בר וסוסים. בנוסף, המטרה הייתה לזהות בעיות וצווארי בקבוק, ולהציע המלצות לשיפור. המודל פותח בשפת התכנות R, תוך שימוש בספריית SIMMER. בית החולים פעיל במשך 16 שעות (מ-7:00 עד 23:00) ומספק שירותי חירום, טיפול נמרץ, טיפולים קוסמטיים, טיפולים שגרתיים ואשפוז. המודל על בית החולים נבנה על סמך נתונים שנאספו על זמני המתנה, תשלומים ועוד. ניתוח המידע העלה מספר בעיות, ביניהן תורים ארוכים בעמדת הקבלה, עומס עבודה גבוה על הוטרנירים בחדר החירום וזמני המתנה ארוכים לאשפוז. כתוצאה מכך, הוצעו מספר המלצות לשיפור, כגון הגדלת כמות עמדות הקבלה, שיפור תיאום זמנים בין הוטרנירים והקמת מחלקה ייעודית לאשפוז. המודל פותח בהצלחה ומאפשר ניתוח מעמיק של התנהלות בית החולים. יישום ההמלצות עשוי לשפר משמעותית את יעילותו ואיכות השירות הניתן בבית החולים.

4. מבוא -

4.1 תיאור המערכת הנחקרת

המערכת הנחקרת היא בית החולים הוטרני האוניברסיטאי שהוא מוסד ייחודי במזרח התיכון המשלב טיפול בבעלי חיים, הוראה ומחקר. בית החולים מציע מגוון רחב של שירותים למגוון בעלי חיים: חיות מחמד, חיות בר וסוסים. קצב הגעת החיות לבית החולים שונה ביניהם (נספח 1).

צוות בית החולים מורכב מרופאים בכירים ומומחים וכולל מחלקות רבות, ביניהן חדר מיון, טיפול נמרץ, אשפוז ועוד. בנוסף לשירותי חירום 24/7, בית החולים מציע גם טיפולים קוסמטיים (כגון תספורות, רחצה וטיפול בציפורניים), וכן טיפולים שגרתיים כמו בדיקות רפואיות, מתן תרופות וחיסונים. יום עבודה בבית החולים אורך במשך 16 שעות, בין השעות 07:00-23:00 ולאחר מכן פועל מוקד לילה בין השעות 23:00-7:00.

ראשית, המטופלים עוברים בעמדת הקבלה. זמן הרישום מתפלג בדרך הבאה (נספח 2).

מלבד חיות הבר שלהן יש עדיפות וממשיכות לשירותי החירום בלבד, חיות המחמד והסוסים ממשיכים לשירותי חירום, טיפולים שגרתיים וטיפולים קוסמטיים עפ"י ההסתברויות הבאות (נספח 3).

בשירותים השגרתיים החיות מגיעות אל וטרינר ולאחר מכן ימשיכו לחיסונים/קבלת תרופות/שירותי חירום/סיכום ביקור ומשך הטיפול יהיה בהתאם לסוג הטיפול ובהתאם לסוג המטופל (נספח 4). חיות מחמד אשר ימתינו מעל חצי שעה בתור יעזבו את בית החולים. חיות הבר אשר ידרשו לקבל תרופות יכנסו לאחות בלי להמתין.

בטיפולים הקוסמטיים, קיימות שתי תכניות. לחיות מחמד ולסוסים המגיעים לטיפול קיימות הסתברויות שונות לתכניות (נספח 5). התכנית הראשונה כוללת 3 טיפולים - ציפורניים/פרסות, רחצה ותספורת. החיות בתכנית זו מחויבות לעבור בכל אחד מהטיפולים והסדר שלהם יקבע כדי לחסוך זמן. התכנית השנייה כוללת טיפול שיניים יסודי ומיד לאחריו חזרה לוטרינר. זמני כלל הטיפולים מתוארים כאן (נספח 6).

בשירותי החירום החיות יעברו תחילה אצל וטרינר חירום. לאחר מכן יתפצלו לאשפוז יום/חדר ניתוח ואשפוז לאחר מכן עפ"י ההסתברויות (נספח 7). במידה ומגיעים לאשפוז יום ואין מיטה פנויה, החיה תופנה לבית חולים אחר. לאחר האשפוז, הן עוברות בדיקת וטרינר כללי. 80% מחיות הבר יקבלו תרופות בהמשך, בעוד היתר יסיימו את הביקור. במידה ומגיעים לניתוח - חשוב לציין שזמן ההמתנה לניתוח משפיע על סיכויי הצלחתו, כאשר נוסחה מיוחדת (נספח 8) מחשבת את הסיכוי לכישלון בהתאם לזמן ההמתנה. במקרה של ניתוח כושל, החיה תועבר לבית חולים אחר והבעלים יהיה פטור מתשלום עבור הטיפולים. זמני הניתוח והאשפוז מפולגים כך (נספח 9).

לאחר כל סוגי הטיפולים מגיעים לסיכום הביקור הכולל תשלום וחתימה על טפסי שחרור. גם כאן יש עדיפות לחיות הבר אשר מטופלות בקבוצות של חמש, אלא אם כן עוברת יותר משעה ואז בקבוצות קטנות יותר. בנוגע לתשלום, אין צורך לסכום את התשלום של חיות הבר (משלמות מחיר קבוע) אך הסוסים וחיות המחמד משלמות עפ"י טבלת התשלומים (נספח 10). זמני סיכום הביקור שונים מחיה לחיה ומצוינים כאן (נספח 11).

4.2 מטרות

1. מטרות בית החולים הווטרינרי -

- יעילות ושיפור זרימת התהליכים בבית החולים: תוך קיצור זמני ההמתנה, ניהול יעיל של צוות רפואי וסגל עזר ושיפור תהליכי קבלה, רישום ותשלום.
- שיפור שביעות רצון הלקוחות: זמינות מיידית במקרים דחופים.
- רווח

2. מטרת הפרויקט - נרצה לבנות סימולציה על מנת להבין לעומק אלמנטים שנלמדו בקורס ויישומם בשפת קוד בתוכנת R-Studio. אנו נבצע ניתוחים חישוביים וסטטיסטיים של המערכת במצב הקיים, על ידי מדדים שונים אשר אותם נפרט בהמשך. כמו כן, אנו נתמודד עם מקרים מורכבים המצריכים שימוש בכלים אשר למדנו בקורס – הן ברמה התיאורטית והן ברמה הפרקטית, זאת על מנת שנקבל תוצאות אופטימליות בהיבט הרווחי ובהיבט התפעולי. נרצה לזהות בעיות, צרכים וצווארי בקבוק ולהציע המלצות לשיפור התנהלות בית החולים.

3. מטרת הסימולציה - הסימולציה תאפשר לבחון את התנהלות בית החולים הווטרינרי לעומק ולזהות דרכים לשיפור יעילות הטיפול בחיות – בדגש על בחינת המדדים המצוינים בסעיף 6.

5. מודל סימולציה

5.1 תיאור המודל

לצורך הדמיית פעילות בית החולים הווטרינרי, נעשה שימוש בחבילת Simmer בתוכנת R-Studio. חבילה זו מאפשרת יצירת ישויות, משאבים, מסלולים ותכונות. בבית החולים מוגדרים שלושה מסלולים עיקריים: טיפולים שגרתיים, טיפולים קוסמטיים וטיפולי חירום. הסימולציה מתחילה עם תחילת יום העבודה ב-07:00 ונמשכת עד 23:00.

שלושה סוגי ישויות מגיעים לבית החולים: חיות מחמד, סוסים וחיות בר. אתחול הישויות מתבצע עם תחילת הריצה, ובנוסף מוגדרת ישות "שומר" שתפקידה לסגור את דלת הכניסה לטיפולי חירום בשעה 19:00.

כל ישות מופנית למסלול המרכזי המתאים לה בהתאם לסיווג. לאחר הגעתן למסלול, מאותחלות תכונותיהן: זמן הגעה, סוג (type) וסך תשלום. המחיר מתעדכן בהתאם למשאבים בהם עוברת הישות.

זמני הטיפוליים בכל משאב מחושבים באמצעות פונקציות עזר, המקבלות כקלט את סוג הישות ומחזירות את זמני הטיפול בהתאם. תחילה, כל החיות מופנות למסלול הקבלה שם נשלחת למסלולים שונים לפי סוג החיות מסלול שמפורט בהמשך בהרחבה.

מסלול חיות בר: חיות הבר מתחילות את טיפוליהן במסלול דלפק הקבלה (דלפק הקבלה של כל החיות חוץ מזוגות סוסים (reception_not_two_horses) ומנסות להגיע לפקיד הקבלה (לחיות הבר יש עדיפות בתור הנ"ל).

לאחר מכן, מועברות חיות הבר ישירות למסלול וטרינר החירום (emergencyvet) ללא פיצולים או אפשרויות נוספות. במסלול וטרינר החירום, חיית הבר מנסה להגיע לוטרינר החירום.

אצל וטרינר החירום, נקבע האם חיית הבר זקוקה לניתוח או לא.

במידה והחיה זקוקה לניתוח, היא מופנת למסלול חדר הניתוח (operationroom), שם החיה עוברת את הניתוח וצולחת אותו בהסתברות מסוימת התלויה בזמן שהחיה חיכתה בתור לניתוח (חיית בר שהניתוח שלה לא מצליח מועברת למסלול otherhospital ומסיימת את דרכה בסימולציה).

במידה וחיית הבר מצליחה את הניתוח, היא מצטרפת אל חיות הבר שלא היו צריכות לעבור את הניתוח וכולן מנסות לתפוס את אחת מ-15 מיטות האשפוז הזמינות. במידה ואין מיטת אשפוז פנויה, חיית הבר מועברת למסלול visitsumwild, מסלול סיכום הביקור של חיות הבר (עליו נפרט בהמשך).

במידה וחיית הבר מצאה מיטת אשפוז פנויה, היא תופסת אותה ולאחר מכן עוברת למסלול

(ishpuzetimegeneral), מסלול האשפוז הכללי, אשר מפנה את החיה מיד למסלול

ishpuzetimewild. במסלול זה חיית הבר מקבלת טיפול מוטרינר כללי אשר מיועד לטפל בחיות בר.

לאחר טיפול זה, חיית הבר חוזרת למסלול האשפוז הכללי ומשם עוברת למסלול סיכום ביקור visitsumwild. במסלול זה, נמחק החשבון של חיית הבר, אשר מחכה עד שעה להתאספות של 5 חיות בר כיוון שעל כולן לעבור סיכום יחד.

מסלול חיות מחמד וסוסים : חיות המחמד והסוסים מתחילות את טיפוליהן במסלול דלפק הקבלה (דלפק הקבלה של כל החיות חוץ מזוגות סוסים reception_not_two_horses) ומנסות להגיע לפקיד הקבלה (ולחיות הבר יש עדיפות בתור הנ"ל).

לאחר מכן, מועברות החיות לאחד משלושת המסלולים בהסתברות מסוימת (שירותי חירום-emergencyvet, טיפולים קוסמטיים-first_group_treatment או second_group_treatment וטיפולים שגרתיים-regular_treatment_pets).

סוסים יתחילו בבדיקה לאיזה מסלול הם הולכים משום שהם בהכרח מגיעים בזוגות, אלא אם כן מדובר במקרה חירום. ב-33% יעברו למסלול טיפולי חירום וב-67% ישלחו לקבלה ולטיפול שגרתי או קוסמטי.

במסלול וטרינר החירום, החיה מנסה להגיע לוטרינר החירום. אצל וטרינר החירום, נקבע האם החיה זקוקה לניתוח או לא. במידה והחיה זקוקה לניתוח, היא מופנת למסלול חדר הניתוח (operationroom) שם החיה עוברת את הניתוח וצולחת אותו בהסתברות מסוימת התלויה בזמן שהחיה חיכתה בתור לניתוח (חיה שהניתוח שלה לא מצליח מועברת למסלול otherhospital ומסיימת את דרכה בסימולציה).

במידה והחיה מצליחה את הניתוח, היא מצטרפת אל אילו שלא היו צריכות לעבור את הניתוח וכולן מנסות לתפוס את אחת מ-15 מיטות האשפוז הזמינות. במידה ואין מיטת אשפוז פנויה, החיה מועברת למסלול visitsum, מסלול סיכום הביקור של חיות הבר (עליו נפרט בהמשך).

במידה והחיה מצאה מיטת אשפוז פנויה, היא תופסת אותה ולאחר מכן עוברת למסלול (ishpuzetimegeneral), מסלול האשפוז הכללי, אשר מפנה את החיה מיד למסלול ishpuzetimeother.

במסלול זה החיה מקבלת טיפול מוטרינר כללי. לאחר טיפול זה, החיה חוזרת למסלול האשפוז הכללי ומשם עוברת למסלול סיכום ביקור visitsum. במידה ונשלחה לטיפול קוסמטי בסיכוי מסוים תלך לטיפול הראשון (first_group_treatment) והמשלים לשני (second_group_treatment).

במסלול הראשון החיה צריכה לעבור בשלוש תחנות שונות כאשר תמיד תעדיף ללכת לפי סדר שתראה שבו התור הכי פחות מלא.

במסלול השני תעבור טיפול שיניים. לאחר סיום אחד משתי האופציות הטיפול תועבר לסיכום טיפול (visitsum). במידה ונשלחה החיה למסלול טיפול שגרתי (regular_treatment_horses, regular_treatment_pets), היא תעבור אצל וטרינר כללי שיבחן את מצבה ויחליט האם החיה צריכה לקבל חיסון ואז תשלח לתור קבלת תרופה אצל האחות (nursesgivemeds) ומשם לסיכום ביקור, או לתור לקבלת חיסון אצל האחות

(nursesgivevaccine) וכמובן משם לסיכום הביקור.
בתור הסיכום ביקור (visitsum) החיות יגיעו וינסו לתפוס את הפקיד בסיום חיית המחמד/הסוס ימתין עד שעה להתאספות 5 חיות (לצורך סיכום משותף).

נספח: תיאור מפורט של הישויות, המשאבים והתכונות ניתן למצוא בנספחים 13 ו-14.

5.2 הנחות

1. משך הסימולציה: 960 דקות .
2. חיות המגיעות לטיפול חירום לאחר 19:00: יעברו רישום בקבלה לפני העברה למוקד הלילה.
3. בדיקה אצל רופא כללי: לאחר אשפוז יום או ביקור אצל רופא שיניים, החיה תעבור בבדיקה אצל רופא כללי בלבד, ללא אפשרות להמשיך למסלולים נוספים.
4. משאב דלת חירום: קיים במערכת שומר שתפקידו לסגור את דלתות שרותי החירום בשעה. בכל כניסה לשירותי חירום, חיה תנסה תחילה להשיג את משאב דלת החירום.
5. התפלגות זמן הגעת חיות בר: מעריכית עם קצב של 0.2636298.
6. זמן ההמתנה לניתוח: נמדד מרגע הכניסה של החיה לבית החולים ועד תחילת הניתוח.

6. ניתוח מצב קיים

6.1 בחירת מדדים לניתוח

שלושת המדדים בהם בחרנו להתמקד הינם: (1) זמן שהייה ממוצעת בבית החולים (2) אורך התור הממוצע אצל וטרינר החירום (3) כמה כל לקוח משלם בממוצע.

(1) מדד 1 - זמן שהייה ממוצעת בבית החולים:

זמן שהייה ממוצע נמוך בבית חולים לחיות חשוב מסיבות רבות:

חויות לקוח:

- זמן שהייה ארוך בבית החולים גורם לבעלי חיות לחוות לחץ, חוסר סבלנות ותסכול.

תחרותיות:

- מטופלים רבים בוחרים בית חולים לפי זמן שהות בבית חולים. תורים קצרים שישפיעו על זמן

שהייה קצר בבית החולים יהוו יתרון תחרותי משמעותי עבור בית החולים.

הדרך בה הגענו לזמן השהייה הממוצע מופיעה בנספח 15.

$avg(ActivityTime)$

(2) מדד 2 - אורך התור אצל וטרינר החירום:

חשיבות בדיקת אורך התור אצל וטרינר החירום חשוב כי:

- זמן המתנה גבוה עלול להצביע על בעיות בתהליך הטיפול, כמו מחסור בוטרינרי חירום, עומס על מערכת החירום, או תכנון לקוי של זמני קבלת קהל.
- ניטור זמן ההמתנה מאפשר זיהוי נקודות תורפה במערכת ונקיטת צעדים לשיפור יעילות הטיפול.

בעייתיות זמן המתנה גבוה:

- זמן המתנה גבוה גורם למטופלים לחוות לחץ וחרדה, במיוחד במצבים חירום רפואיים.
 - המתנה ממושכת עלולה להחמיר את מצבם של מטופלים, במיוחד במקרים דחופים.
 - זמן המתנה ארוך עלול להוביל לעזיבה של מטופלים לפני קבלת טיפול, מה שעלול להזיק לבריאותם.
 - זמן המתנה גבוה פוגע בתדמית מערכת הבריאות וגורם לאי שביעות רצון בקרב המטופלים.
- הקוד שבו השתמשנו נמצא בנספח 16. בנוסף ישנה פונקציית עזר בנספח 16.1.

$$avgQueue = \frac{1}{T_{max}} \int_0^{T_{max}} avgQueue(t) dt$$

(3) מדד 3 - ממוצע תשלום לכל לקוח:

- מעקב אחר תשלומים של לקוחות מאפשר לך לעקוב אחר ההכנסות שלך ולנהל את תזרים המזומנים שלך בצורה יעילה.
- ידע מדויק לגבי סכומים ששולמו, תשלומים עתידיים ותשלומים באיחור תורם לתכנון עסקי נכון וקבלת החלטות אסטרטגיות.
- ניתוח נתוני תשלומים יכול לחשוף מגמות, דפוסי צריכה והתנהגות לקוחות, ומאפשר לך להתאים את האסטרטגיה העסקית שלך בהתאם.

הקוד נמצא בנספח 17.

$$avg(Total_To_Pay)$$

7. נספחים :

נספח 1 – קצב הגעת החיות :

התפלגות	
$X \sim \text{Exp}(4)$	חיות מחמד
$X \sim U(20,40)$	סוסים
$X \sim \text{Exp}(1/0.2636298)$	חיות בר

נספח 2 – זמן רישום בעמדת קבלה :

$$f(x) = \begin{cases} \frac{5x}{150} + \frac{2}{3}, & 3 \leq x \leq 4 \\ \frac{13}{120}, & 4 < x \leq 6 \end{cases}$$

אלגוריתם הדגימה בקוד :

```
#countertime calculates the time each entity waits at the counter
countertime <- function() {
  # random number between 0-1
  u <- runif(1, 0, 1)

  # create countertime according to function
  if (u <= 47 / 60) {
    countertime <- - 20 + sqrt(529 + 60 * u)
  }
  else{
    countertime <- (120 * u - 42) / 13
  }
  return(countertime)
}
```

נספח 3 – התפלגויות הגעת החיות לטיפולים השונים :

שירותי חירום	טיפולים קוסמטיים	טיפולים שגרתיים	
31%	39%	30%	חיות מחמד
33%	46%	21%	סוסים

נספח 4 – התפלגויות טיפולי המשך מהוטרינר הכללי :

סיכום ביקור	שירותי חירום	מתן תרופות	חיסונים	
35%	13%	25%	27%	חיות מחמד וסוסים
20%	-	80%	-	חיות בר

נספח 5 – התפלגויות לתכניות לטיפול קוסמטיים :

תכנית ראשונה	תכנית שנייה	
79%	21%	חיות מחמד וסוסים
89%	11%	חיות בר

נספח 6 – התפלגויות זמני קבלת הטיפולים הקוסמטיים :

התפלגות	פעולה
$X \sim \text{Exp}(15)$	טיפול שיניים
$X \sim N(90,20)$	תספורת
$X \sim N(60,15)$	רחצה
$X \sim N(15,2)$	ציפורניים

נספח 7 – התפלגות הפניות לשירותי חירום :

אשפוז יום	חדר ניתוח	
87%	13%	כלל החיות

נספח 8 – נוסחה לחישוב כישלון בניתוח בהתאם לזמן המתנה :

$$f(t) = \begin{cases} \frac{1}{1 + e^{-(3t-8)}} , & 0 \leq t \leq 16 \\ 0 , & \text{else} \end{cases}$$

נספח 9 – התפלגות זמני שירותי החירום :

התפלגות	
$X \sim U(30,50)$	ניתוח
$X \sim N(240,30)$	אשפוז

נספח 10 – טבלת תשלומים על הטיפולים השונים :

התפלגות	פעולה
₪ 30	מתן חיסון
₪ 200	מתן תרופות
₪ 150 לכל בדיקה	בדיקת וטרינר (כללי/חירום)
₪ 600	מסלול 1 קוסמטי
₪ 700	מסלול 2 קוסמטי
₪ 2000	ניתוח
₪ 1500	אשפוז

נספח 11 – התפלגות זמני סיכום ביקור :

התפלגות	
$X \sim N(4, 0.5)$	ניתוח
$X \sim N(20, 4)$	אשפוז

Distribution of wild animals

2024-03-05

התקנת חבילה קריאה מקובץ אקסל

```
install.packages("readxl")

# Libraries
library(fitdistrplus)

## Warning: package 'fitdistrplus' was built under R version 4.3.3
## Loading required package: MASS
## Loading required package: survival

library(magrittr)

## Warning: package 'magrittr' was built under R version 4.3.3

library(readxl)

## Warning: package 'readxl' was built under R version 4.3.3

dataset <- read_excel(file.choose())

## New names:
## • `` -> `...2`
## • `` -> `...3`
## • `` -> `...4`
## • `` -> `...5`
## • `` -> `...6`

# Convert the "datasets בר חיות" column to POSIXct format
dataset$`בר חיות` <- as.POSIXct(dataset$`בר חיות`, format = "%Y-%m-%d
%H:%M:%S")

# Calculate the time difference between consecutive rows in the
"datasets בר חיות" column in minutes
time_diff <- diff(dataset$`בר חיות`, units = "mins")

# Add a new column named "Time DiffMinutes" to the dataset and populate
it with the time difference in minutes
dataset$Time.DiffMinutes <- c(0, time_diff / 60)

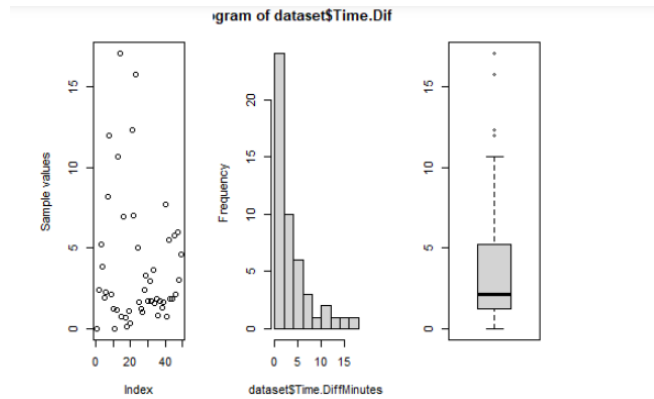
# Print the "Time DiffMinutes" column
print(dataset$Time.DiffMinutes)

## [1] 0.0000000 2.4000000 5.2500000 3.8500000 1.9500000
2.2666667
## [7] 8.1833333 12.0000000 2.1333333 1.2166667 0.0000000
1.1333333
## [13] 10.6500000 17.0333333 0.7333333 6.9166667 0.7166667
0.1166667
## [19] 1.0666667 0.3166667 12.3166667 7.0000000 15.7833333
5.0000000
## [25] 1.6666667 1.2333333 1.0333333 2.4166667 3.3000000
1.7333333
## [31] 2.9833333 1.7333333 3.6166667 1.5500000 1.8500000
0.8500000
## [37] 1.7333333 1.3000000 1.6166667 7.7000000 0.7833333
5.4833333
## [43] 1.8333333 1.8833333 5.8000000 2.1333333 5.9500000
3.0166667
## [49] 4.6333333
```

First Impressions

For that we can use the following plots:

```
par(mfrow=c(1,3)) # defines 2 plots on the same window
plot(dataset$Time.DiffMinutes, ylab="Sample values") # plot ( left graph
)
hist(dataset$Time.DiffMinutes, ylab="Frequency ") # histogram (right
graph)
boxplot(dataset$Time.DiffMinutes)
```



According to the plots shown, we chose to check a normal distribution

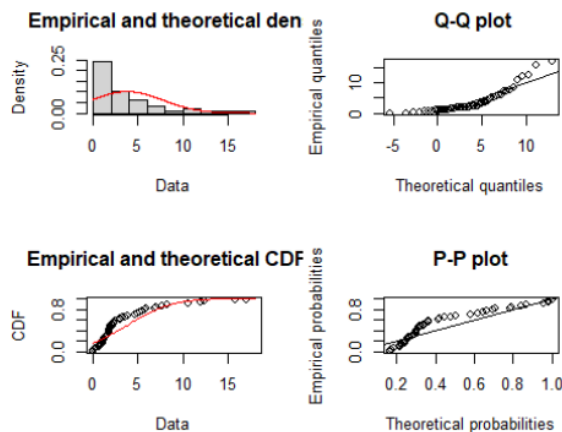
```
normFit<-fitdist(dataset$Time.DifMinutes,"norm") # fitting a normal
distribution
summary(normFit) # summary function displays the results
```

```
## Fitting of the distribution ' norm ' by maximum likelihood
## Parameters :
##      estimate Std. Error
## mean 3.793197  0.5640321
## sd   3.948225  0.3988308
## Loglikelihood: -136.818   AIC: 277.6361   BIC: 281.4197
## Correlation matrix:
##      mean sd
## mean  1  0
## sd    0  1
```

The estimate are R estimates of the paramters of a normal distribution - mean and sd.

```
normFit$estimate[1] # is the first paramter- mean
normFit$estimate[2] # is the second paramter- sd
```

Now we will use plotdist to plot our data versus the theoretical distribution



Now we have chosen to test an exponential distribution

```
expFit<-fitdist(dataset$Time.DifMinutes,"exp") # fitting a exponential
distribution
summary(expFit) # summary function displays the results
```

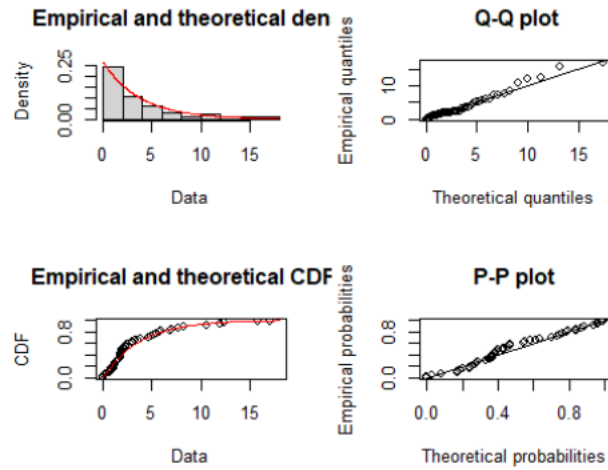
```
## Fitting of the distribution ' exp ' by maximum likelihood
## Parameters :
##      estimate Std. Error
## rate 0.2636298  0.03766086
## Loglikelihood: -114.3273   AIC: 230.6545   BIC: 232.5463
```

We can reach the paramter directly from code using the following:

```
expFit$estimate[1] #rate
```

Now we will use plotdist to plot our data versus the theoretical distribution

```
plotdist(dataset$Time.DifMinutes,"exp",para=list(rate=expFit$estimate[
1]))
```



נספח 13 - משאבים :

שם המשאב	קיבולת	אורך תור
קבלה	3	Inf
וטרינר כללי חיות בר	1	Inf
וטרינר כללי חיות מחמד וסוסים	1	Inf
וטרינר חירום	1	Inf
מיטת ניתוח	15	0
חדר ניתוח	2	Inf
דלת		Inf
סיכום ביקור	2	Inf
אחות - תרופות	1	Inf
אחות - חיסונים	1	Inf
תספורת	1	Inf
רחצה	1	Inf
ציפורניים	1	Inf
טיפול שיניים	1	Inf

תכונה	ערכים	הסבר
end1start1	ערך מספרי	תכונה לצורך חישוב הסיכוי להצלחת הניתוח
hair_cuter / washer / nail_cuter	בינארי – 0 או 1	מספור לטובת סידור הטיפולים בתכנית קוסמטית 1
type	1- חיות בר 2- חיות מחמד & סוסים	תכונה נדרשת עבור בחירת הוטריןר הרלוונטי
Total_To_Pay	ערך מספרי	העלות הכוללת של כל חיה
vaccine_time", "medication_time", "vet_check_time", "summary_time"	ערך מספרי	תכונה לצורך קביעת זמן הטיפול לכל סוג של חיה

```
# Average time at the hospital
paste("avg time at hospital: ", sqldf("
SELECT
  AVG(activity_time) AS avg_activity_time
FROM data_per_animal
"))
```

```
#Average time in emergency queue
time<-as.matrix(sqldf("SELECT time FROM data_for_resource_per_resource WHERE resource='emergency_vet'"))
queueLength <- as.matrix(sqldf("SELECT queue FROM data_for_resource_per_resource WHERE resource='emergency_vet'"))
avgResQueue <- avgQueue(time, queueLength, simulationTime)
paste("avg time at emergency vet queue: ", avgResQueue)
```

נספח 16.1 - פונקציית עזר לאורך תור ממוצע לוטרינר חירום :

```
avgQueue <- function(time, queueLength, simTime){
  Lavg = 0;
  L = queueLength[1];
  Tnow = time[1];
  Llast = time[1];
  TL = 0;
  Tmax = simTime;
  if (length(time) == length(queueLength)){
    for (i in 2:length(time)){
      if(queueLength[i] != queueLength[i-1]){
        Tnow = time[i];
        TL = TL+L*(Tnow-Llast);
        L = queueLength[i];
        Llast = Tnow;
      }
    }
  }
  TL=TL+L*(Tmax-Llast);
  Lavg = TL/Tmax;
  return (Lavg);
}
```

נספח 17 – תשלום ממוצע של לקוח לבית החולים :

```
# Average payment for an animal
paste("avg payment in hospital:"
      ,sqldf("
SELECT
  AVG(value)
FROM data_per_attribute
WHERE key = 'Total_To_Pay'
"))
```

נספח 18 – הוכחת נכונות המודל :

18.1 חייט מחמד עוזבת אחרי חצי שעה בדיוק והactivity_time שלה הוא 0 :

	name	start_time	end_time	activity_time
438	pets114	439.75167	469.75167	0.000000

18.2 דלת בית החולים נסגרת ב19:00 :

בית החולים נפתח ב7:00, 720 דקות לאחר מכן (12 שעות) הדלת נסגרת.

	name	start_time	end_time	activity_time	finished	replication
349	doorman0	720.000000	720.000000	0.000000	TRUE	1

18.3 סוסים וחיות מחמד עוברות את שלושת הטיפולים הקוסמטיים בתכנית 1 (תספורת, רחצה וציפורניים):

ניתן לראות שמדובר באותה היישות ושכל פעולה מתחילה כשהקודמת מסתיימת

769	horse1	416.78175	489.84863	17.205871	washer
791	horse1	489.84863	502.92689	13.078257	hair_cuter
822	horse1	502.92689	523.08717	15.993048	nail_cuter

18.4 שני סוסים מתקבצים יחד על מנת לעבור את הדלפק הראשוני:
ניתן לראות כי סוס 1 וסוס 2 נכנסו יחד לדלפק וסיימו יחד (תוך שהתאחדו).

1404	horse18	943.4693	946.6183	3.149047	counter	1
1405	horse21	943.4693	946.6183	3.149047	counter	1

943.469: horse21: i arrived to horse traj
943.469: horse21: we, two horses, arrived to reception
943.469: batch_twoHorses: we are now two horses together

סוס 21 הגיע ולאחר מכן נוצרה הישות המשותפת של שני הסוסים

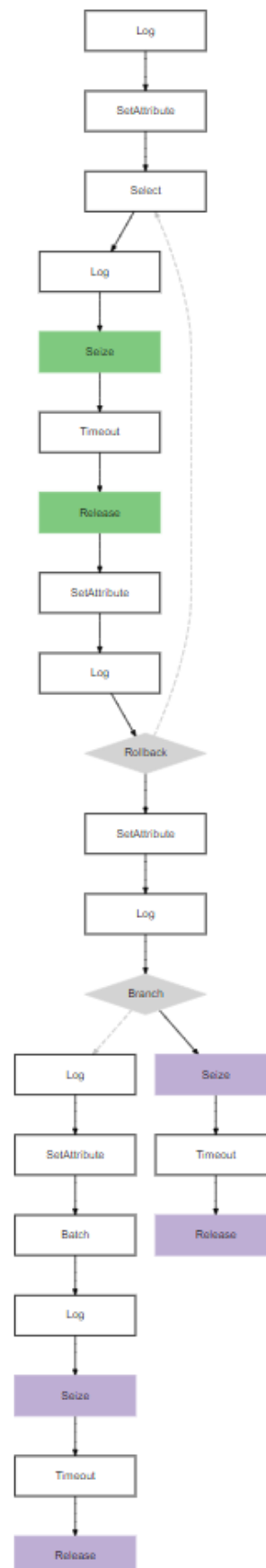
18.5 חמש חיות בר מתקבצות יחד לקראת דלפק הסיכום:

261	wildanimals15	192.4885	204.8111	12.32261	sum_counter	1
262	wildanimals16	192.4885	204.8111	12.32261	sum_counter	1
263	wildanimals17	192.4885	204.8111	12.32261	sum_counter	1
264	wildanimals18	192.4885	204.8111	12.32261	sum_counter	1
265	wildanimals19	192.4885	204.8111	12.32261	sum_counter	1

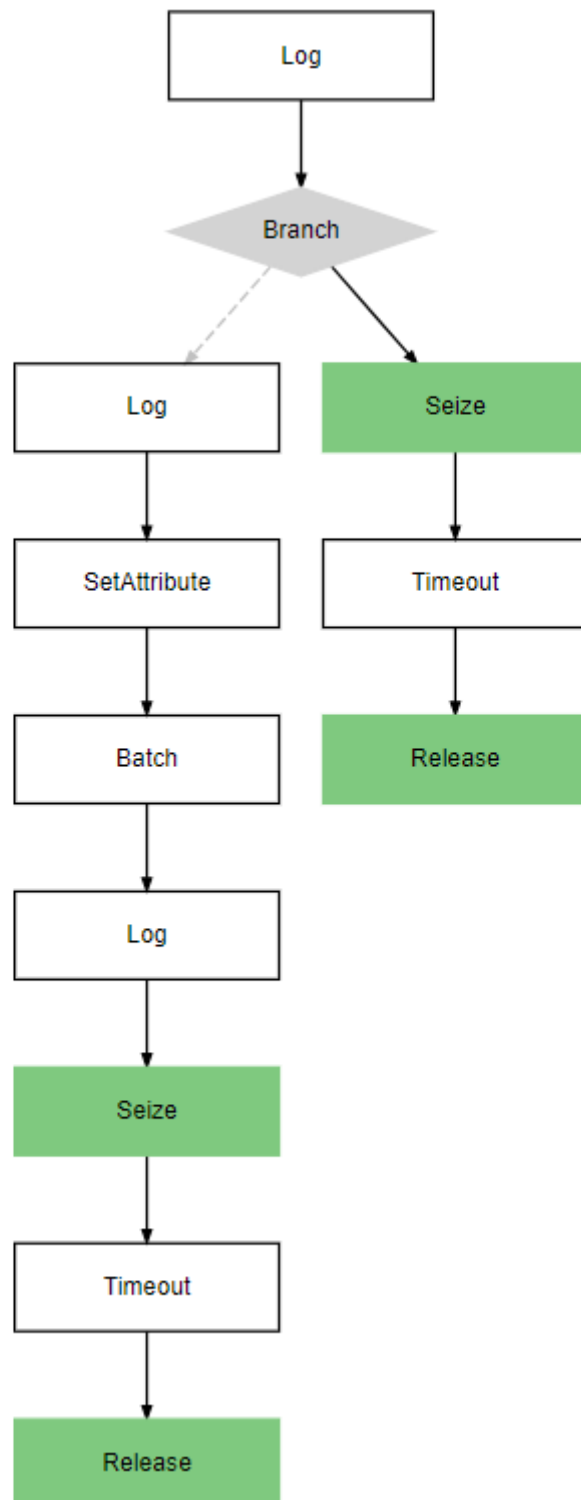
18.6 הסוס הראשון מגיע בשעה 12:00

421	horse0	300.0000	303.8813	3.881301	counter	1
-----	--------	----------	----------	----------	---------	---

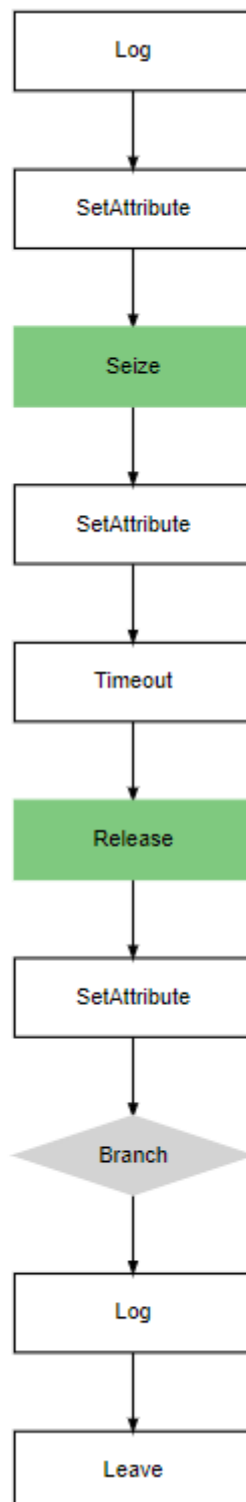
18.7 פלוט שמציג את השתלשלות האירועים במסלול הטיפול הראשון-



18.8 פלוט שמראה את השתלשלות האירועים במסלול סיכום ביקור של חיות מחמד וסוסים:



18.9 פלוט שמציג את השתלשלות האירועים במסלול של חדר הניתוח:



```
.1 -----##all functions-----

#countertime calculates the time each entity waits at the counter
countertime <- function() {
  # random number between 0-1
  u <- runif(1, 0, 1)

  # create countertime according to function
  if (u <= 47 / 60) {
    countertime <- - 20 + sqrt(529 + 60 * u)
  }
  else {
    countertime <- (120 * u - 42) / 13
  }

  return(countertime)
}

#chooses the first treatment pets go to after reception
first_treatment_pets<-function() {
  typenumber<-function()runif(1, min =0, max = 1)
  if (typenumber()<0.31) {
    # emergency_vet=1
    print(paste("I should go to emeregency vet"))
    return(1)
  }
  if (typenumber()>0.61) {
    typenumber2<-function()runif(1, min =0, max = 1)
    if(typenumber2()<0.79) {
      # first_group treatmt=2
      return(2)
    }
    else {
      # second_group treatment=3
      return(3)
    }
  }
  else {
    # regular vet pets=4
    return(4)
  }
}

#chooses the first treatment horses go to after reception
first_treatment_horses<-function() {
  typenumber<-function()runif(1, min =0, max = 1)
  if (typenumber()<0.33) {
    return(1)
  }
  if (typenumber()>0.79) {
```

```

# regular vet horses=5
  return(5)
{
  else}
  typenumber3<-function()runif(1, min =0, max = 1)
  if(typenumber3()<0.89){
    return(2)
  }
  else}
  return(3)
{
{
{

#chooses the first treatment wild animals go to- only emergency possible
first_treatment_wild<-function() {
  return(1)
{

#determine whether the surgery will succeed
surgery_fail_rate <- function(ZooHospital){
  start1<- get_attribute(ZooHospital, "Start_time") #the time animal entered the queue
  end1<- get_attribute(ZooHospital, "End time") #the time animal started the surgery
  t<- (end1-start1)/60 #convert to hours
  if (t >= 0 & t <= 16){
    return(1 / (1 + exp(-(3 * t - 8))))
  }
  else}
  return(0)
{
{

#Determine the order of the 3 possibilities- hair cut,wash,nail treatment
Cosmeticresource<-function(ZooHospital){
  hair_cuter <- get_attribute(ZooHospital,"hair_cuter")
  washer <- get_attribute(ZooHospital,"washer")
  nail_cuter <- get_attribute(ZooHospital,"nail_cuter")
  return_vec<- character(0) #empty vector, all possible treatments (that animal didn't do yet) are added

  if(identical(washer,0)){
    return_vec<-c(return_vec,"washer") #if washer wasn't used yet, its added to the possibilities vector
  }
  if(identical(hair_cuter,0)){
    return_vec<-c(return_vec,"hair_cuter") #if hair cutter wasn't used yet, its added to the possibilities
vector
  }
  if(identical(nail_cuter,0)){
    return_vec<-c(return_vec,"nail_cuter") #if nail cutter wasn't used yet, its added to the possibilities
vector
  }
  return(return_vec)
{

```

```

#return how long hair cut,wash or nail treatment take, according to what's needed
cosmetic_first_group_treatments_timeout <- function(resource_name){
  if (identical(resource_name,c("hair_cuter"))){
    return(function () rnorm(1,mean = 90,sd=20)) # Timeout for hair_cuter
  } else if (identical(resource_name,c("washer"))){
    return(function () rnorm(1,mean = 60,sd=15)) # Timeout for washer
  } else{
    return(function () rnorm(1,mean = 15,sd=2)) # Timeout for nail_cuter
  }
}

```

```

#According to animal type choose the relevant vet- 1 for wild, 2 for pets and horses
what_vet <- function(ZooHospital){
  animal<-get_attribute(ZooHospital,"type")
  if (animal == 3){
    return(1)
  }
  else{
    return(2)
  }
}

```

```

#determine what visit summary trajectory to go to- 1 is for wild animals and 0 for others
what_sum <- function(ZooHospital){
  animal<-get_attribute(ZooHospital,"type")
  if (animal == 3){
    return(1)
  }
  else{
    return(0)
  }
}

```

```

#Function to help get emergency vet queue
avgQueue <- function(time, queueLength, simTime){
  Lavg = 0;
  L = queueLength[1]
  Tnow = time[1]
  Llast = time[1]
  TL = 0;
  Tmax = simTime;
  if (length(time) == length(queueLength)){
    for (i in 2:length(time)){
      if(queueLength[i] != queueLength[i-1]){
        Tnow = time[i];
        TL = TL+L*(Tnow-Llast);
        L = queueLength[i];
        Llast = Tnow;
      }
    }
  }
}

```

```

{
{
{
  TL=TL+L*(Tmax-Llast);
  Lavg = TL/Tmax;
  return (Lavg);
}

.2 -----##all simulation parameters-----

simulationTime<-16*60

.3 -----##Init Simulation and add all resources-----
-----
ZooHospital<- simmer%<%()
# regular treatment & starting counter resources
add_resource("counter",capacity=3,queue_size=Inf)%<%
add_resource("regular_vet_wild",capacity=1,queue_size=Inf)%<%
add_resource("regular_vet_other",capacity=1,queue_size=Inf)%<%

# emergency resources
add_resource("emergency_vet",capacity=1,queue_size=Inf)%<%
add_resource("emergency_bed",capacity=15,queue_size=0) %<%
add_resource("emergency_operation_room",capacity=2,queue_size=Inf)%<%
add_resource("door", capacity = Inf)%>% #When time is over, this will lock patients out

# visit summary resources
add_resource("sum_counter",capacity=2,queue_size=Inf)%<%

# nurse resources
add_resource("nurse_meds",capacity=1,queue_size=Inf,preemptive=TRUE)%<%
add_resource("nurse_vaccine",capacity=1,queue_size=Inf,preemptive=TRUE)%<%

# cosmetic first group treatments resources
add_resource("hair_cuter",capacity=1,queue_size=Inf,preemptive=FALSE)%<%
add_resource("washer",capacity=1,queue_size=Inf,preemptive=FALSE)%<%
add_resource("nail_cuter",capacity=1,queue_size=Inf,preemptive=FALSE)%<%

# cosmetic second group treatments resources
add_resource("teeth_cleaner",capacity=1,queue_size=Inf,preemptive=FALSE)

.4 -----##All trajectories, start from main trajectory and add sub-trajectories
ABOVE IT it-----

#wild animals visit summary trajectory
visitsumwild->
  trajectory("visitsum wild path")%<%
  log_("i arrived to visitsum wild animals")%<%
  set_attribute("Total_To_Pay", value = 0)%<%
  batch(n=5, timeout=60, permanent=FALSE,name="fiveWildAnimals")%>% #batch 5 wild animals and
  seize the sum counter together
  log_("we finished the batch")%<%
  seize("sum_counter",1)%<%

```



```

timeout(function() rnorm(1,mean = 20,sd=4))%<%
release("sum_counter",1)

#visit summary for all animals
visitsum->
  trajectory("visitsum path")%<%
  log_("i arrived to visitsum")%<%
  branch(option = function() what_sum(ZooHospital,(
    continue=c(FALSE),visitsumwild)%>% #Wild animals are sent to another traj inorder to batch
there
  seize("sum_counter",1)%<%
  timeout_from_attribute("summary_time")%<%
  release("sum_counter",1)

#Three possible treatments that are chosen by the first available- wash, hair cut and nail treatment
first_group_treatment->
  trajectory("first_group path") %<%
  log_("i arrived to first_group")%<%
  set_attribute(keys = c("washer","hair_cuter","nail_cuter"), values= c(0,0,0))%>% #this is used in the
function
  simmer::select(resources = function() Cosmeticresource(ZooHospital,(
    policy = "first-available",id = 0) %>% #choose the treatment
  log_("I chose first resource")%<%
  seize_selected(amount = 1, id = 0)%<%
  timeout(cosmetic_first_group_treatments_timeout(function() get_selected(ZooHospital)))%>% #use
function to find the correct time for timeout
  release_selected(amount = 1, id = 0)%<%
  set_attribute(keys = function()(get_selected(ZooHospital)), value = 1) %>% # that way this treatment
can't be chosen again
  log_("I had treatment")%<%
  simmer::rollback(target = 7, times = 2)%<%
  set_attribute("Total_To_Pay", value = 600, mod= "+" )%<%
  join(visitsum) #After this you go to visit sum traj to pay and finish

#second cosmetic treatment- teeth treatment
second_group_treatment->
  trajectory("second_group path")%<%
  log_("i arrived to second_group") %<%
  set_attribute("Total_To_Pay", value = 700, mod= "+" )%<%
  seize("teeth_cleaner",amount = 1)%<%
  timeout(function() exp(15))%<%
  release("teeth_cleaner",amount = 1)%<%
  seize("regular_vet_other",1)%>% #After this treatment you see a regular doctor, so seize from here
  timeout_from_attribute("vet_check_time")%<%
  release("regular_vet_other",1)%<%
  join(visitsum) #After finishing the treatment and seeing doctor, go to visit sum traj to pay and finish

#Nurse gives the animal medicine
nursesgivemed->

```

```

trajectory("nurses_give_meds path")%<%
log_("i arrived to nurses_give_meds")%<%
set_attribute("Total_To_Pay", value = 200, mod= "+" )%<%
seize("nurse_meds",1)%<%
timeout_from_attribute("medication_time")%>% #different attribute for each type of animal
release("nurse_meds",1)%<%
join(visitsum) #After finishing the treatment and seeing doctor, go to visit sum traj to pay and finish

```

#nurse gives vaccine to animal

nursesgivevaccine->

```

trajectory("nurses_give_vaccine path")%<%
log_("i arrived to nurses_give_vaccine")%<%
set_attribute("Total_To_Pay", value = 30, mod= "+" )%<%
seize("nurse_vaccine",1)%<%
timeout_from_attribute("vaccine_time")%<%
release("nurse_vaccine",1) %<%
join(visitsum) #After finishing the treatment and seeing doctor, go to visit sum traj to pay and finish

```

#Animals that there surgery doesn't succeed get here, their payment becomes 0 and they leave compensation->

```

trajectory("compensation path")%<%
log_("delete price before leaving")%<%
set_attribute("Total_To_Pay", value = 0)%<%
leave(1)

```

#Wild animals that come to rest come here for there timeout and back to the general ishpuzetraj

ishpuzetimetwild->

```

trajectory("ishpuzetimetwild path")%<%
log_("i arrived to ishpuzetimetwild")%<%
seize("regular_vet_wild",1)%<%
timeout_from_attribute("vet_check_time")%<%
release("regular_vet_wild",1)

```

#Pets and horses that come to rest come here for their timeout and back to the general ishpuzetraj

ishpuzetimeother->

```

trajectory("ishpuzetimeother path")%<%
log_("i arrived to ishpuzetimeother")%<%
seize("regular_vet_other",1)%<%
timeout_from_attribute("vet_check_time")%<%
release("regular_vet_other",1)

```

#send each type of animal to rest and timeout, after that they go to visitsum traj to pay and finish

ishpuzetimegeneral->

```

trajectory("ishpuzetimegeneral path")%<%
log_("i arrived to ishpuzetimegeneral")%<%
branch(option=function() what_vet(ZooHospital),continue= c(TRUE,
TRUE),ishpuzetimetwild,ishpuzetimeother)%<%

```

all animals go to there specific ishpuzetraj and come back here after their timeout

```

join(visitsum)

```

#whoever is sent to another hospital gets here and finishes their journey

```

otherhospital->
  trajectory("otherhospital")%<%
  log_("i am in the otherhospital")%<%
  leave(1)

#Animals who need operation get here
operationroom->
  trajectory("operationroom <-")%<%
  log_("i am in the operationroom")%<%

  set_attribute(key = "Start_time", function() now()) %>% #time of arrival
  seize("emergency_operation_room")%<%
  set_attribute(key = "End time", function() now()) %>% #time when surgery starts
  timeout(function() runif(1, 30, 50))%<%
  release("emergency_operation_room")%<%

  set_attribute(key = "Total_To_Pay,"
    values = function() { 2000 + get_attribute(ZooHospital, "Total_To_Pay") }%<% (

branch)
  option = function} ()
  prob <- surgery_fail_rate(ZooHospital)
  rdiscrete(1, c(1 - prob,prob), c(0,1))
,{
  continue = c(FALSE),
  otherhospital
# ( animals that their surgery doesn't succeed leave the hospital, others return to emergencyvet traj
that they came from

#animals that come late are locked out and leave
lockedout->
  trajectory("lockedout path")%<%
  log_("i'm locked out")%<%
  leave(1)

#Lets us know if door is open or not
opendoor->
  trajectory("opendoor path")%<%
  log_("the door is open so I'm in")

#Lets us know if bed was take or not
bedismine->
  trajectory("bedismine path")%<%
  log_("I took the bed")

#animals are sent here in different paths
emergencyvet->
  trajectory("emergencyvet path")%<%
  log_("i arrived to emergency ")%<%
  seize("door",1, continue = c(TRUE,FALSE), post.seize = opendoor, reject = lockedout) %<%

```

```

# Animals can only enter if it's early, if too late door is not available and they go to lockedout traj and leave
release("door",1)%<%
log_("I took and returned the door")%<%
set_attribute("Total_To_Pay", value = 150, mod= "+" )%<%
seize("emergency_vet",1)%>% #Animals are first checked by vet
timeout_from_attribute("vet_check_time")%<%
release("emergency_vet",1)%<%
branch(option=function() rdiscrete (1, c(0.87,0.13),c(0,1)),continue= c(TRUE),operationroom)%<%
# Some of the animals are sent to surgery, other try to catch a bed
log_("i arrived to getting a bed ")%<%
seize("emergency_bed", 1, continue = c(TRUE,FALSE), post.seize=bedismine, reject = visitsum)%<%
# If bed is caught continue here, if not go to visitsum traj, pay and leave
timeout(function() rnorm(1, mean = 240, sd = 30))%<%
release("emergency_bed", 1)%<%
log_("I returned my bed")%<%
set_attribute("Total_To_Pay", value = 1500, mod= "+" )%<%
join(ishpuzetimegeneral) #Once done with emergency treatment, animal goes to ishpuze

#Some pets start their journey from regular treatment
regular_treatment_pets->
trajectory("regular_treatment pets")%<%
log_("i arrived to regular_treatment_pets ")%<%
renew_in%<%(30)
# If pet is in line for over 30 minutes it leaves
log_("i am leaving")%<%
seize("regular_vet_other",1)%<%
renew_abort %<%()
timeout_from_attribute("vet_check_time") לשנות רגיל ואז בפנים %<%
release("regular_vet_other",1)%<%
set_attribute("Total_To_Pay", value = 150, mod= "+" )%<%
branch(option=function() rdiscrete (1, c(0.27,0.13,0.25,0.35),c(1,2,3,4)),continue= c(FALSE, FALSE,FALSE,FALSE),nursesgivevaccine ,emergencyvet,nursesgivemeds,visitsum)
# Animal continues to its next traj according to distribution

#Some horses start their journey from regular treatment
regular_treatment_horses->
trajectory("regular_treatment horses")%<%
log_("i arrived to regular_treatment_horses ")%<%
seize("regular_vet_other",1)%<%
# The horses seize the vet and timeout according to their function
timeout_from_attribute("vet_check_time")%<%
release("regular_vet_other",1)%<%
set_attribute("Total_To_Pay", value = 150, mod= "+" )%<%
branch(option=function() rdiscrete (1, c(0.27,0.13,0.25,0.35),c(1,2,3,4)),continue= c(FALSE, FALSE,FALSE,FALSE),nursesgivevaccine ,emergencyvet,nursesgivemeds,visitsum)
# Animal continues to its next traj according to distribution

#Some wild animals start their journey from regular treatment
regular_treatment_wild->
trajectory("regular_treatment wild")%<%
log_("i arrived to regular_treatment_wild ")%<%

```

```

# The wild animals seize the vet and timeout according to their function
seize("regular_vet_wild",1)%<%
timeout_from_attribute("vet_check_time")%<%
release("regular_vet_wild",1)%<%
set_attribute("Total_To_Pay", value = 150, mod= "+" )%<%
branch(option=function() rdiscrete (1, c(0.8,0.2),c(1,2)),continue= c(FALSE,
FALSE),nursesgivevaccine,visitsum)
# Animal continues to its next traj according to distribution

#special reception designed to connect horses
reception_two_horses->
  trajectory("two horses reception path")%<%
  log_("we, two horses, arrived to reception")%<%
  batch(n=2, timeout=Inf, permanent=FALSE, name="twoHorses")%<%
# horses batch together to go through counter together
log_("we are now two horses together")%<%
seize("counter",1)%<%
timeout(function () countertime())%<%
release("counter",1)%<%
separate%<%()
# horses continue seperatly according to distribution
branch(option = function() get_attribute(ZooHospital,"first_treatment_num,("

continue=c(FALSE,FALSE,FALSE,FALSE,FALSE),emergencyvet,first_group_treatment,second_group_treatment,
  regular_treatment_pets,regular_treatment_horses(

#Entering the hospital-no 2 horses
reception_not_two_horses->
  trajectory("reception path")%<%
  log_("i arrived to reception")%<%
  seize("counter",1)%<%
  timeout(function () countertime())%<%
  release("counter",1)%<%
  branch(option = function() get_attribute(ZooHospital,"first_treatment_num,("

continue=c(FALSE,FALSE,FALSE,FALSE,FALSE),emergencyvet,first_group_treatment,second_group_treatment,
  regular_treatment_pets,regular_treatment_horses(
# All other animals (including horses for emergency) go to the next treatment according to ditribution

#Trajectory for each type of animal to set all relevant attributes
#pets trajectory, pets is 1
pet_traj->
  trajectory("Pet path")%<%
  log_("i arrived to pet traj")%<%

set_attribute(c("type","vaccine_time","medication_time","vet_check_time","summary_time","Total_To_Pay,("
  value=c(1, rnorm(1,mean = 15,sd=4), runif(1, min =7, max = 13,(

```

```

,10          rnorm(1,4,0.5),0%<% ((
set_attribute("first_treatment_num",value=function() first_treatment_pets())%<%
join(reception_not_two_horses)

#horses trajectory, horses is 2
horse_traj->
  trajectory("horse path")%<%
  log_("i arrived to horse traj")%<%

set_attribute(c("type","vaccine_time","medication_time","vet_check_time","summary_time","Total_To_Pay,("
  value=c(2, rnorm(1,mean = 12,sd=3), runif(1, min =13, max = 16,(
,14          rnorm(1,4,0.5),0%<% ((
set_attribute("first_treatment_num",value=function() first_treatment_horses())%<%
branch(option = function() get_attribute(ZooHospital,"first_treatment_num,("

continue=c(FALSE,FALSE,FALSE,FALSE,FALSE),reception_not_two_horses,reception_two_horses,r
eception_two_horses,
  reception_two_horses,reception_two_horses(
# If the horse is not supposed to go to emergency, go to 2 horses reception to batch

#wild_animal_trajectory, wild is 3
wild_animal_traj->
  trajectory("wild animal path")%<%
  log_("i arrived to wild animal traj")%<%
  set_attribute(keys=c("type","medication_time","vet_check_time","summary_time","Total_To_Pay,("
    value=c(3, rexp(1, 10,(
,9          rnorm(1,20,4),0%<% ((
set_attribute("first_treatment_num",value=function() first_treatment_wild())%<%
join(reception_not_two_horses)

#doorman comes to this traj to close the emergency services when it's late
closeDoor <- trajectory("door")%<%
log_( function() paste("the door is closed"))%<%
set_capacity("door",0)%<%
set_queue_size("door",0)

.5 -----##All Generators, ALWAYS LAST-----
-----

ZooHospital%<%

  add_generator(name="doorman", trajectory=closeDoor,at(720),mon=2)%>% #Doorman will lock the
door and stop emergencyvet at 19:00
  add_generator(name="pets",trajectory=pet_traj,distribution=function() rexp(1,
0.25),priority=1,preemptible=2,restart = FALSE, mon=2)%<%
  add_generator("horse", trajectory=horse_traj, from_to(300,960, function ()runif(1,20,40), arrive =
TRUE),priority=1,preemptible=2,restart = FALSE,mon=2)%<%

```

```
add_generator(name="wildanimals",trajectory=wild_animal_traj,distribution=function()
rexp(1,0.2636298),priority=2,preemptible=3,restart = FALSE,mon=2)
```

```
.6 -----##reset, run, plots, outputs-----
```

```
set.seed(456)
reset(ZooHospital)%>%run(until=simulationTime)
data_per_animal<-get_mon_arrivals(ZooHospital)
data_per_resource <- get_mon_arrivals(ZooHospital,per_resource = TRUE)
data_for_resource_per_resource<-get_mon_resources(ZooHospital)
data_per_attribute<-get_mon_attributes(ZooHospital)
```

```
#Measures
```

```
#Average payment for an animal
paste("avg payment in hospital":
```

```
, sqldf")
SELECT
  AVG(value)
FROM data_per_attribute
WHERE key = 'Total_To_Pay'
```

```
(("
```

```
#Average time at the hospital
paste("avg time at hospital: ", sqldf")
SELECT
  AVG(activity_time) AS avg_activity_time
FROM data_per_animal
(("
```

```
#Average time in emergency queue
time<-as.matrix(sqldf("SELECT time FROM data_for_resource_per_resource WHERE
resource='emergency_vet'"))
queueLength <- as.matrix(sqldf("SELECT queue FROM data_for_resource_per_resource WHERE
resource='emergency_vet'"))
avgResQueue <- avgQueue(time, queueLength, simulationTime)
paste("avg time at emergency vet queue: ", avgResQueue)
```

```
#Plots to show the chain of events in 3 different trajectories
plot(first_group_treatment)
plot(visitsum)
plot(operationroom)
```

```
-----##libraries##-----
```

```

#install.packages("rlang", repos = "http://cran.us.r-project.org")
#install.packages("MASS", repos = "http://cran.us.r-project.org")
#install.packages("fitdistrplus", repos = "http://cran.us.r-project.org")
#install.packages("magrittr", repos = "http://cran.us.r-project.org")
#install.packages("simmer", repos = "http://cran.us.r-project.org")
#install.packages("simmer.plot", repos = "http://cran.us.r-project.org")
#install.packages("dplyr", repos = "http://cran.us.r-project.org")
#install.packages("lazyeval", repos = "http://cran.us.r-project.org")
#install.packages("parallel", repos = "http://cran.us.r-project.org")
#install.packages("e1071", repos = "http://cran.us.r-project.org")
#install.packages("plotly", repos = "http://cran.us.r-project.org")
#install.packages("ggplot2", repos = "http://cran.us.r-project.org")
#install.packages("triangle", repos = "http://cran.us.r-project.org")
#install.packages("sqldf", repos = "http://cran.us.r-project.org")
#install.packages("knitr", repos = "http://cran.us.r-project.org")
#install.packages("rmarkdown", repos = "http://cran.us.r-project.org")
#install.packages("readxl", repos = "http://cran.us.r-project.org")
#install.packages("reprex")
#install.packages("lubridate")
#install.packages("RSQLite")
#
#
#library(rlang)
#library(MASS)
#library(fitdistrplus)
#library(magrittr)
#library(dplyr)
#library(lazyeval)
#library(parallel)
#library(e1071)
#library(plotly)
#library(ggplot2)
#library(triangle)
#library(sqldf)
#library(readxl)
#library(knitr)
#library(rmarkdown)
#library(simmer)
#library(simmer.plot)
#library(reprex)
#library(lubridate)
#library(sqldf)

```