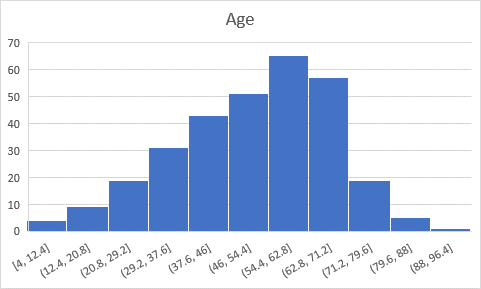
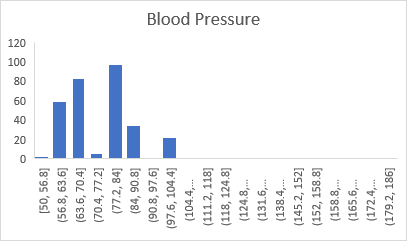
שאלה 1

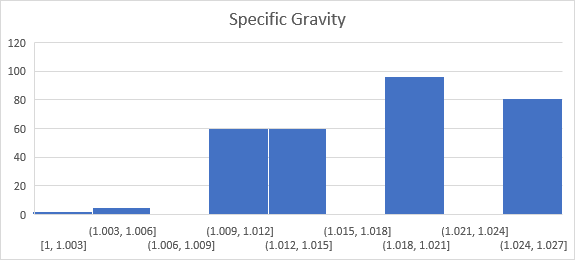
1. **מטרת כריית המידע:**מטרת כריית המידע היא לחזות האם לחולה יש CKD – Chronic Kidney Disease, על בסיס רשומות החולה ומערכת חיזוי המבוססת על 400 רשומות של פציינטים שחלקם (250) חולים והשאר (150) אינם חולים. הרשומות מכילות נתונים כגון גיל, לחץ דם, רמת המוגלובין בדם ונתונים נוספים. את כל הנתונים ופירוט עליהם ניתן לראות בסעיף ב. על מסד הנתונים שקיבלנו יש מספר הנחות יסוד.
   1. הנתונים מדויקים עד כדי תיקון ידני של תאים שזזו או ערכי שהשתנו במקצת (למשל הוספת רווח לערך yes או no).
   2. הנתונים הגיעו בפורמט arff. מכיוון שיותר נוח לעבוד עם פורמט xlsx, יש להניח שהמעבר מarff לcsv והמעבר מcsv לxlsx הוא פשוט ועמיד לטעויות.
   3. רשימת המידע על השדות מהאתר שממנו קיבלנו את מסד הנתונים מעודכנת ונכונה. (האתר מוזכר בסוף המסמך, בביבליוגרפיה).
2. **הנתונים בשימוש:**

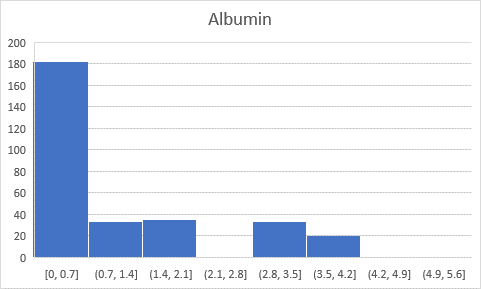
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **שם השדה** | **יחידות מידה** | **סוג הנתונים** | **תחומי ערכים** | **mean** | **median** | **mode** |
| **Age** | years | נומרי | 0-100 | 51.2 | 54 | 55 |
| **Blood pressure** | Mm/Hg | נומרי | 50-180 | 75.8 | 80 | 80 |
| **Specific gravity** |  | נומרי | 1 + 0.05k (0<=k<=5) | 1.02 | 1.02 | 1.02 |
| **Albumin** |  | קטגורי | 0-5 |  |  |  |
| **Sugar** |  | קטגורי | 0-5 |  |  |  |
| **Red blood cells** |  | בינארי | 1 אם תקין |  |  |  |
| **Pus cell** |  | בינארי | 1 אם תקין |  |  |  |
| **Puc cell clumps** |  | בינארי | 1 אם קיים |  |  |  |
| **Bacteria** |  | בינארי | 1 אם קיים |  |  |  |
| **Blood glucose random** | Mgs/dl | נומרי | 70-490 | 140.6 | 117 | 107 |
| **Blood urea** | על עמודה זו נוותר, כי רק 70 מתוך 400 הרשומות (כ17%) הוא בטווח התקין של הערכים (לפי חיפוש בגוגל), ולכן סביר שהערכים המוזנים שגויים. | | | | | |
| **Serum Creatinine** | Mgs/dl | נומרי | 0.4-10 | 1.8 | 1.2 | 1.2 |
| **Sodium** | mEq/L | נומרי | 104-163 | 138.5 | 139 | 135 |
| **Potassium** | mEq/L | נומרי | 2.5-7.6 | 4.34 | 4.4 | 5 |
| **Hemoglobin** | Gms | נומרי | 6-17.8 | 13.2 | 13.7 | 15 |
| **Packed cell volume** | Percentage | נומרי | 0-100 | 40 | 41 | 52 |
| **White blood cell count** | Cells/ cumm | נומרי | 2200-16700 | 8355 | 8100 | 9200 |
| **Red blood cell count** | Millions/ cumm | נומרי | 2.1-6.5 | 4.8 | 4.8 | 4.5 |
| **Hypertension** | על עמודה זו נוותר, כי העמודה blood pressure נותן לנו את אותו המידע באופן יותר מפורט. | | | | | |
| **Diabetes Mellitus** |  | בינארי | 1 אם חולה |  |  |  |
| **Coronary Artery Disease** |  | בינארי | 1 אם חולה |  |  |  |
| **Appetite** |  | בינארי | 1 אם יש |  |  |  |
| **Pedal Edema** |  | בינארי | 1 אם חולה |  |  |  |
| **Anemia** |  | בינארי | 1 אם חולה |  |  |  |
| **Classification** |  | בינארי | 1 אם חולה |  |  |  |

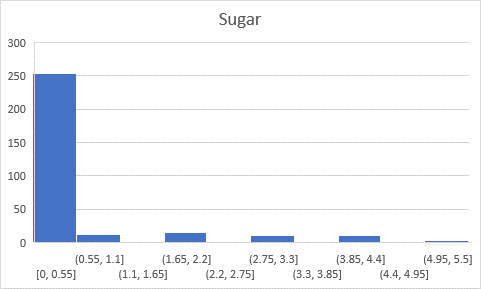
1. **שלבי הKDD:**
   1. איסוף נתונים: נעשה עבורנו (הורדנו את הנתונים מהאתר שבביבליוגרפיה).
   2. ניקוי נתונים:
      1. העברה מפורמט arff לcsv ואז לxlsx.
      2. מעבר ידני על מסד הנתונים: זיהוי בעיות הבולטות בעין (תיקון ההזזה בשורה 371).
      3. הסרת העמודות bu, hypertension.
      4. החלפת ערכים מחוץ לטווח בערך חסר.
      5. הסרת שורות מרובות חוסרים (5 או יותר ערכים חסרים – שורש מכמות השדות הקיימים).
      6. השלמת ערכים מספריים חסרים בהתאם לעמודה (ממוצע).
      7. השלמת ערכים בינאריים חסרים (עצי חיזוי פשוטים).
      8. המרת ערכים קטגוריים לבינאריים (איפה שניתן).
      9. דיסקרטיזציה וניפוי הנתונים לערכים שמישים בעזרת שיטות כמו binning – **נדחה לעת חישוב העץ.**
   3. בחירת שיטות לכריית מידע:
      1. הכנה של מסד הנתונים: excel.
      2. ניקוי הנתונים קוד פייתון בפיתוח שלי בשימוש בספריית pandas.
      3. חישוב עצי החלטה בשימוש בספריית sklearn.
      4. הצגת עצי ההחלטה בשימוש בספריית graphviz.
   4. הרצת כריית המידע
      1. שימוש בקוד פייתון מוכן מספריות שהוזכרו.
      2. עטיפה בקוד משלי להתאמות לצרכי המטלה.
   5. ניתוח התוצאות
      1. ניתוח סטטיסטי מבוסס על תוצאות הריצה.
      2. ניתוח סטטיסטי של "כלל הרוב".
      3. החלטה האם כריית המידע מוצלחת.
2. **חלופות לכריית המידע:**
   1. מיפוי אלגוריתמים שנלמדו:
      1. עץ החלטה ID3 – Information Gain  
         העץ מורכב על ידי שימוש במדד אנטרופיה וחישוב הרווח (הצמצום) של האנטרופיה מפיצול מסויים.  
         **יתרונות:** מדד gain שימושי מאוד ונותן תוצאות טובות לכריית מידע. העץ פשוט למימוש ולבנייה.  
         **חסרונות:** information למול ratio – במקרה של עץ שאינו בינארי, תינתן העדפה לתכונות מרובות ערכים. בנוסף, העץ המתקבל עלול להיות בומבסטי ולכן איטי בהרצתו בעת החיזוי.
      2. עץ החלטה C4.5 – Gain Ratio  
         העץ פועל בדומה לID3, רק שבחישוב הרווח נכנס גם היחס לכמות הערכים השונים לכל תכונה.  
         **יתרונות:** מדד gain שימושי מאוד ונותן תוצאות טובות לכריית מידע. העץ פשוט למימוש ולבנייה, מתמודד עם בעית הinformation למול ratio על מסדי נתונים שבהם קיימות מגוון תכונות עם כמויות ערכים שונות.  
         **חסרונות:** העץ המתקבל עלול להיות בומבסטי ולכן איטי בהרצתו בעת החיזוי. לפעמים ייתן תוצאות חלשות יותר מאשר ID3 בגלל הוספת היחס בעצים בינאריים.
      3. עץ החלטה CART – Gini  
         העץ מורכב לפי מדד ג'יני, שמאפשר התמודדות טובה גם לתכונות מרובות ערכים וגם לתכונות בינאריות. לאחר הרכבת העץ, מתבצע תהליך גיזום אשר מקטין את העץ בשביל לחסוך בזמן ריצה. תהליך הגיזום רץ עד שהוא מגיע לאיזון.  
         **יתרונות:** מדד gini גם הוא חזק ונותן תוצאות טובות לכריית מידע. העץ המתקבל קטן ולכן קל יותר להריץ עליו חיזויים.  
         **חסרונות:** לוקח הרבה יותר זמן לאמן עץ כזה בגלל הגיזום - pruning. העץ קטן יותר ולכן עלול להניב תוצאות פחות טובות מID3 – מעין tradeoff עם זמן הריצה של עץ.
      4. האצה אדפטיבית AdaBoost  
         היער מבוסס על עצי CART, רק שכל אחד מהם מקבל משקל בהצבע שבסוף התהליך. בנוסף, גם בעת הלמידה ניתנים משקלים שונים לערכים שונים.  
         **יתרונות:** מדד gini גם הוא חזק ונותן תוצאות טובות לכריית מידע. היער המתקבל מורכב ממספר עצים שונים ומותאם לנתונים ברגישות גבוהה יותר.  
         **חסרונות:** לוקח הרבה יותר זמן לאמן יער כזה, גם בגלל הגיזום (במידה ובוחרים להכליל גיזום) וגם בגלל ריבוי העצים. זמן הריצה עלול להיות ארוך בעת חיזוי בגלל ריבוי העצים.
      5. יער אקראי Random Forest  
         היער מבוסס על עצי ID3, רק שכל אחד מהם מאומן על תכונות שונות. כל אחד מהם מקבל גם משקל שונה להצבעה שבחיזוי.  
         **יתרונות:** מדד gain שימושי מאוד ונותן תוצאות טובות לכריית מידע. היער המתקבל מורכב ממספר עצים שונים ומותאם לנתונים ברגישות גבוהה יותר.  
         **חסרונות:** לוקח הרבה יותר זמן לאמן יער כזה, בגלל ריבוי העצים. זמן הריצה גם הוא ארוך משמעותית בעת חיזוי בגלל ריבוי העצים הגדולים (אין גיזום).
   2. החלטה בהקשר הפרוייקט:
      1. מכיוון שרוב המסווגים הקטגוריים שלנו הם בינאריים, אין טעם ממשי להשתמש בGain Ratio על פני Information Gain, ולכן ויתרתי על C4.5.
      2. את שאר המודלים מימשתי והשוויתי בקוד, מכיוון שלכל אחד מהם יש יתרונות וחסרונות שעלולים להיות מועילים, ולכן רציתי להשוות ביניהם.
      3. בחרתי לבסוף ביער אקראי והאצה אדפטיבית, כי אפשר היה לוותר על ID3 מכיוון שהוא "מוכל" בתוך היער האקראי, ועל CART מכיוון שהוא "מוכל" בהאצה אדפטיבית. המימוש של העצים האחרים נשאר בקוד.
3. **שלבי הכנת הנתונים:**
   1. ערכים חסרים וערכים שגויים  
      במסד הנתונים התקבלו שני סוגים של נתונים בעייתיים: ערכים חסרים וערכים שגויים. התחלתי מטיפול בערכים השגויים, שאותם מצאתי בכך שהגדרתי טווחים הגיוניים לכל אחת מהשדות מחיפוש מהיר בגוגל, כל ערך שגוי הוחלף בערך חסר.  
      הטיפול בערכים החסרים התבצע בשלבים. ראשית, הסרתי את השורות שבהן חסר לפחות 5 ערכים. הגדרתי שמספר הערכים החסרים הוא לפחות שורש ממספר התכונות מכיוון שזה שלב שבו תלות גדולה מדי בין הערכים יכולה בהחלט להשפיע על החיזוי. שנית, השלמתי את הערכים המספריים החסרים בממוצע של התכונה הזו. לבסוף, את הערכים הבינאריים החסרים השלמתי בעזרת עצי חיזוי בסיסיים שהרכבתי על בסיס הערכים המספריים בעבור כל אחד מהערכים.  
      כמובן שאם היה חסר ערך בעמודת הCKD הייתי מוחק את אותה השורה כי היא לא מוסיפה לנו מידע, אבל לא היו שורות כאלו במסד הנתונים.
   2. ניקוי נתונים  
      כחלק מהאלגוריתמים השונים לבנית עצים בספריית sklearn מתבצע binning לערכים. תהליך זה ממיר מגוון גדול של ערכים לכדי טווחים, וזה מאפשר לעצים לעבוד בגרנולריות גבוהה יותר, מכיוון שלא צריך להתייחס לכל אחד מהערכים, במיוחד בשדות שבהן אין הבדל גדול בין ערכים עוקבים, כמו למשל גיל. בנוסף, הbinning מכליל את הממוצע שהכנסנו בשלב הערכים החסרים, ובכך חוסך את העיגול שהיינו צריכים לעשות לו אחרת.
   3. המרת נתונים  
      הנתונים הבינאריים היו שמורים בצורה מילולית, ובמקום זה הפכתי אותם לערכים של 0 ו1. המרה זו חשובה במיוחד במקרה של הclassification, שבו הערכים 1 ו0 והסדר שלהם קובע את ההבדל בין FP לFN. במקרה זה בחרתי ש1 יהיה שהמטופל חולה בCKD.  
      בנוסף, הסרתי את העמודות blood urea וhypertension מסיבות שמפורטות בטבלה.
   4. תצוגה גרפית  
      משום מה הexcel מתעקש לעשות binning בייצור הגרפים ולא הצלחתי לבטל את זה ועדיין שזה יראה נורמאלי

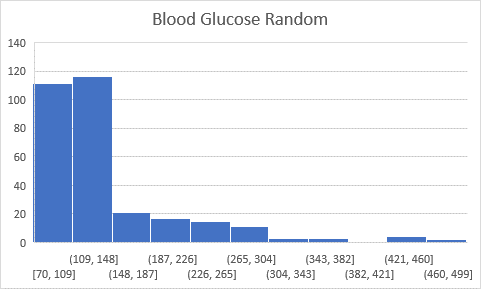


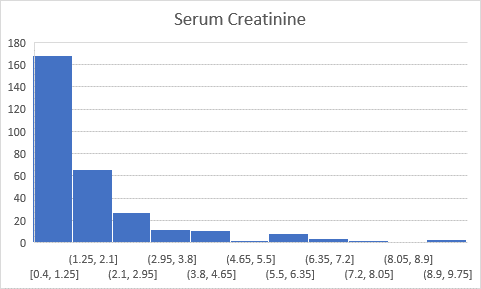


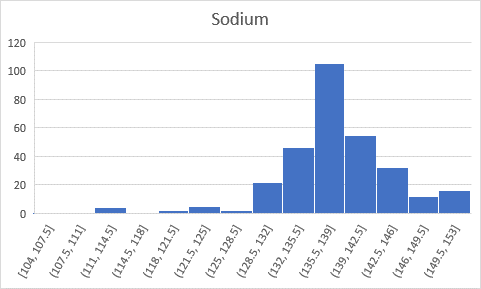


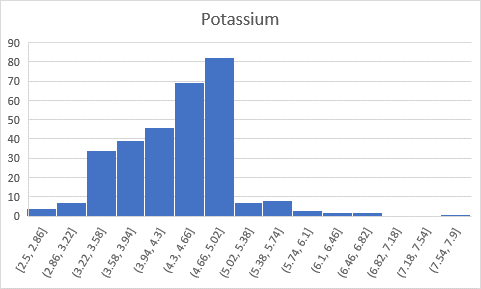


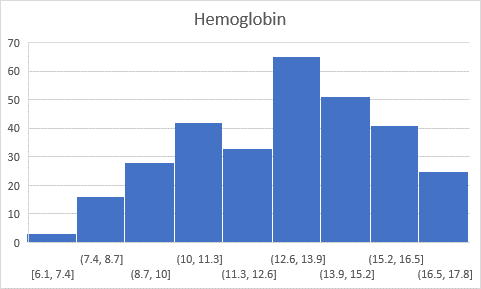


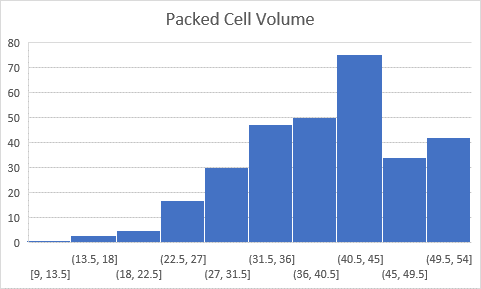


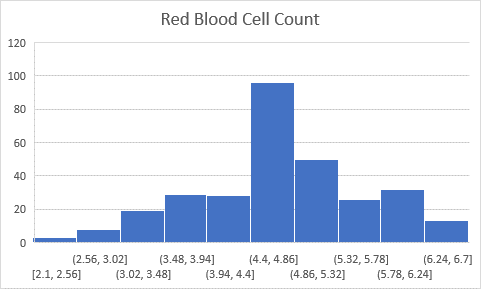


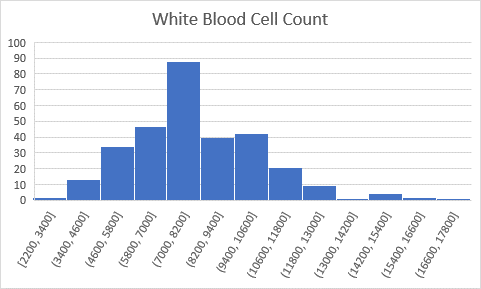


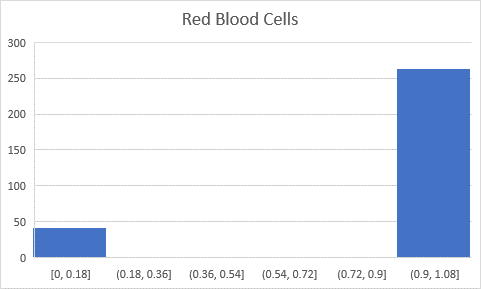


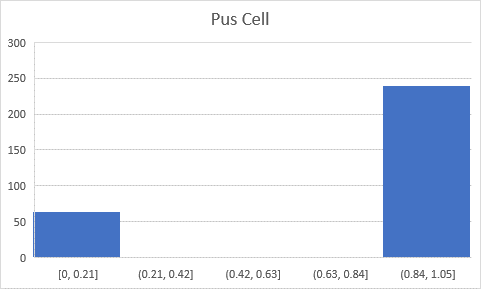


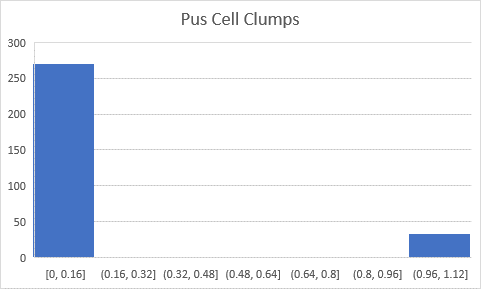


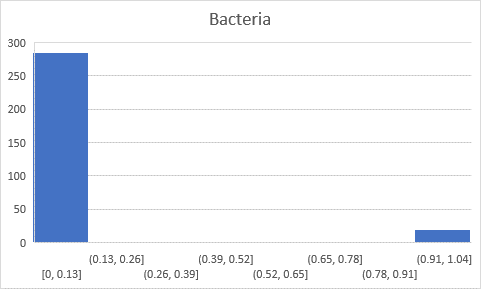


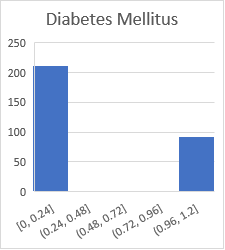
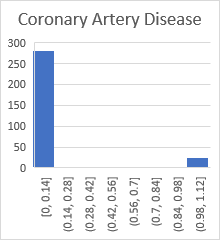


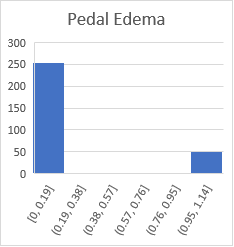
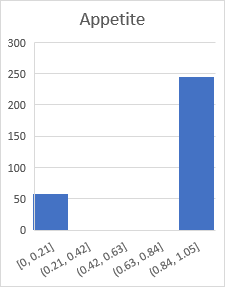


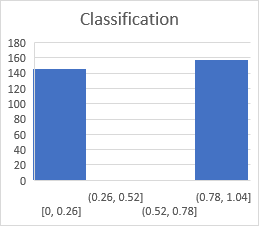
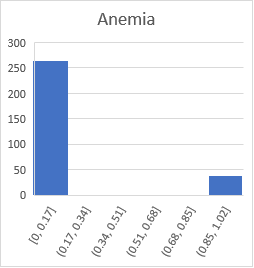






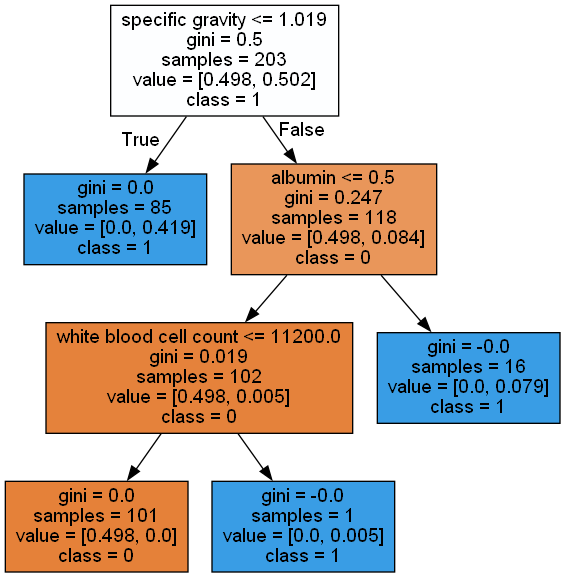
 

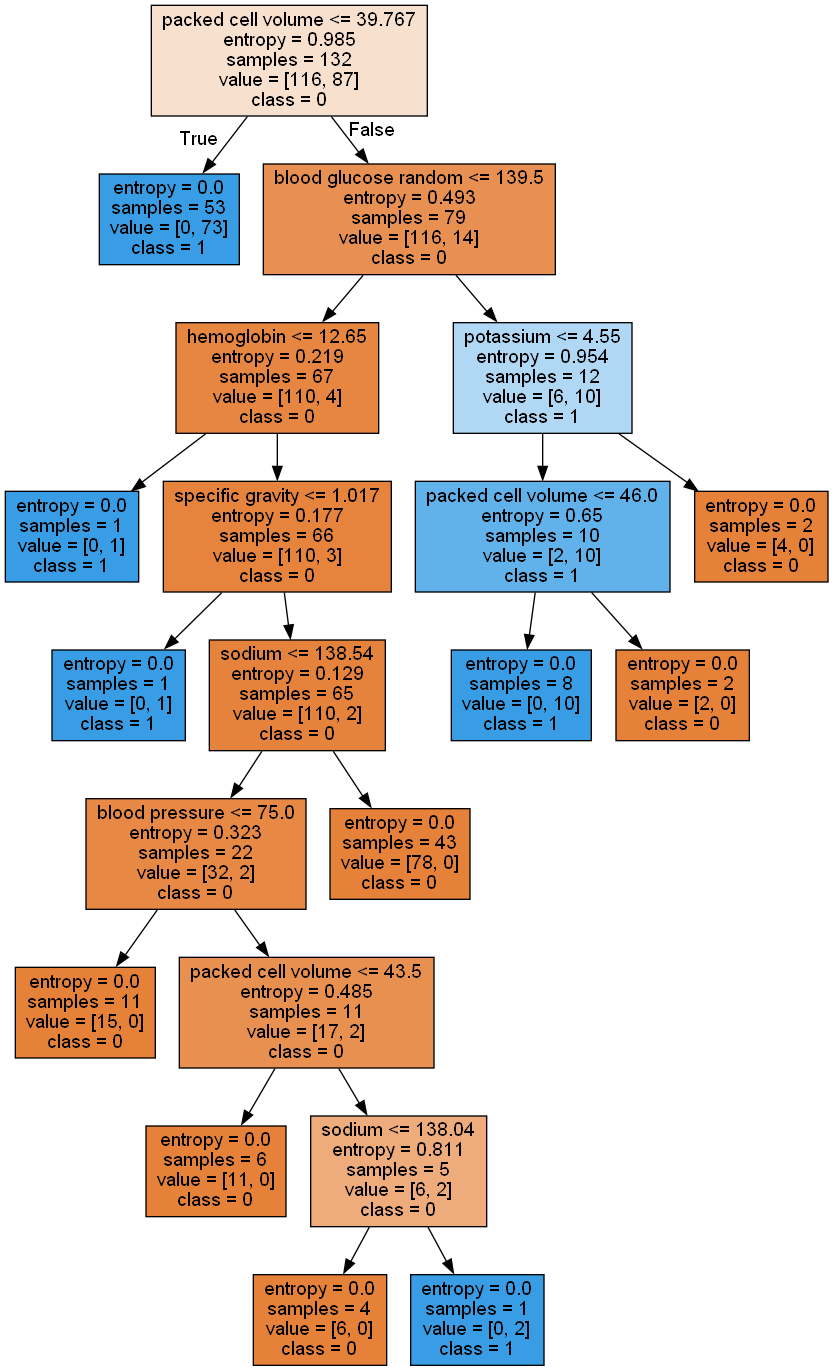




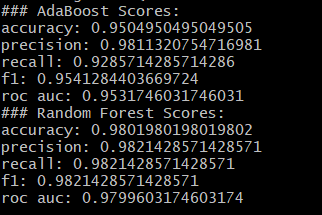
שאלה 2

1. **בחירת שיטות לסיווג נתונים:**כפי שהוזכר בשאלה 1 סעיף ד, בחרנו בשתי שיטות ensamble לצורכי סיווג הנתונים – AdaBoost וRandom Forest. שתי השיטות הללו נבחרו על פני שיטות סיווג שונות בגלל החוזק שבensamble. שיטות ensamble מייצרות מספר עצים, ואז נותנות לבחירת הרוב להכריע את הסיווג (בין אם רוב ממושקל או הוגן). היכולת הזו של הצבעה מונעת overfitting, ומשפרת את הביצועים של הסיווג.
2. **תיאור שיטות הסיווג:**
   1. שיטת AdaBoost:  
      מרכיבים עצים בשיטת CART – מנסים להקטין את מדד הgini בכל שלב ושלב, וגוזמים את העץ במידת הצורך. כל עץ מתאמן על חלק אחד של הנתונים (למשל בעזרת חלוקת k-fold), ולכן מתקבלים עצים שונים. כשמסיימים עם הרכבת העצים משנים את העצים בעזרת למידה מטעויות. נותנים משקל גדול יותר לנתונים שבהם טעינו, ובכך מאמנים את העצים להשתפר בחיזוי של הטעויות שלהם. כשמסיימים את הרכבת העצים נותנים ציונים לעצים לפי החוזק שלהם על מידע הבדיקה. בעת חיזוי נקבע החיזוי הכולל לפי שיקלול של {מה עץ מסוים חזה, כמה הוא בטוח בחיזוי זה ומה משקל העץ} על כל העצים.
   2. שיטת Random Forest:  
      מרכיבים עצים בשיטת ID3/C4.5 – מנסים להגדיל את רווח הידע על בסיס אנטרופיה, בין אם רווח ישיר או רווח יחסי, ומוסיפים צמתים בעץ בהתאם. כל עץ מתאמן על חלק אחד של נתונים (למשל בעזרת חלוקת k-fold), ורק על חלק מהמאפיינים (העמודות בטבלה). כך נוצרים עצים המבוססים על מאפיינים שונים בחיזוי שלהם. בעת חיזוי נקבע החיזוי הכולל לפי דעת הרוב של העצים. התלות בהרבה עצים המתבססים על מאפיינים שונים עוזר להסיר overfitting.
3. **תוצאות ונתונים:**

בתור שיטות ensamble, כל אחת מהשיטות מייצרת יער. בגלל שיער קשה להצגה, אציג עץ לדוגמה מכל אחת מהשיטות.  
שיטת Adaboost:  


שיטת Random Forest:ניתן לראות בעץ את בכל צומת הדברים הבאים:

* 1. תנאי – התנאי לפיצול של הצומת. אם התנאי מתקיים נפנה שמאלה, אם התנאי לא מתקיים נפנה ימינה. אם הוא מתקיים נפנה שמאלה. בעלים שורה זו לא קיימת. שורה זו גורמת לצורה של עץ בינארי.
  2. גודל המדד – בין אם משתמשים במדד אנטרופיה או ג'יני, אפשר לראות את הערך הנוכחי של המדד בהתאם למקום בעץ. ערך של 0 אומר שבהכרח הצומת הוא עלה.
  3. כמות הנתונים – כמות הנתונים מתוך הטבלה המקורית שמקיימים את כל התנאים להגעה לצומת הזה. ערך של 1 אומר שבהכרח הצומת הוא עלה.
  4. ערך – לא רלוונטי, שדה לשימוש פנימי של הספריות.
  5. זיהוי – הזיהוי המקומי בעבור הצומת הנוכחי – האם חולה בCKD או לא. ערך 1 (או צבע רקע כחול) זה חולה, וערך 0 (או צבע רקע כתום) זה לא חולה.

1. **חישובי דיוק:**חישבתי בעבור כל אחד מהעצים דיוקים במדדים שונים, בהתאם לחלוקה של אימון-בדיקה. התוצאות שהתקבלו הן התוצאות הבאות:  
   
2. **השוואה ומסקנות:**  
   ראשית, נתחיל מהשוואה לחוק הרוב. אם נלך לפי חוק הרוב, רוב הרשומות הן של חולים (classification=1), והן 250 רשומות מתוך 400. כלומר אם ננחש על אדם מסויים שהוא חולה לפי חוק הרוב נהיה צודקים ב62.5% מהמקרים. בבירור, שתי השיטות שלנו עברו את הביצועים הללו בקלות, ולכן אפשר להגדיר את כריית המידע כהצלחה.  
     
   שנית, נשווה בין שתי השיטות השונות. כמובן שבגלל שהן שיטות ensamble נקבל עליהן תוצאות טובות, אבל במקרה הזה לRandom Forest יש תוצאות טובות יותר בכל אחד מהמדדים. מכיוון שבמקרה כזה דיוק הניתוח חשוב בהרבה ממהירות הניתוח, ככל הנוגע לרפואה, כמובן שנעדיף את המודל שמביא תוצאות יותר טובות, גם במקרה שהוא איטי יותר. לכן, נבחר במודל Random Forest בתור המודל המועדף.  
     
   לבסוף, נבחן מה היה יכול להתבצע אחרת בשביל להוביל לתוצאות יותר טובות או אמינות. לדעתי, מסד הנתונים שקיבלנו הוא די קטן ומנוון, ומעבר לכך היו חסרים בו הרבה נתונים. זה הוביל לכמות קטנה של נתונים, ובגלל השלמת הנתונים הרבה מהשדות היו דומים בערכיהם. על כן, הצלחנו להגיע לתוצאות יחסית טובות בעזרת יערות יחסית קטנים. להבא, נכון להביא מסד נתונים גדול בהרבה, ועם ערכים יותר מדוייקים. זה יסיר מליבי את החשש למעין overfitting שיכול להיות שנוצר פה. בנוסף, ייתכן שהיו נתונים מדעיים נוספים שהיו יכולים לעזור לחיזוי, אך מפאת חוסר הידע הרפואי והמדעי שלי לא אדע להמליץ על נתונים כאלה להבא.

ביבליוגרפיה

מקור בסיס הנתונים והמידע על הנתונים: <https://archive.ics.uci.edu/ml/datasets/chronic_kidney_disease>

ספריות פייתון שבהן השתמשתי:

Sklearn: <https://scikit-learn.org/stable/>

Graphviz: <https://graphviz.org/>

Pandas: <https://pandas.pydata.org/>