

# AI PROJECT – NBA FANTASY

## PREDICTIONS AND DRAFT GAME AGENTS

**מגישים:**

- מעין קינן – 203815451 – [maayankeinan@campus.technion.ac.il](mailto:maayankeinan@campus.technion.ac.il)
- רון עבאדי – 204803589 – [ronabadie@campus.technion.ac.il](mailto:ronabadie@campus.technion.ac.il)
- אופיר גורדון – 204025233 – [ofirgo@campus.technion.ac.il](mailto:ofirgo@campus.technion.ac.il)

**מנחה:** גיא קושילביץ

**תאריך:** 28.09.19



## תוכן עניינים

4	הקדמה
6	הגדרת הבעיה
6	משחק ה-Fantasy NBA
9	דגשים, גישות ואסטרטגיות למשחק הפנטזי
13	הגדרת הבעיה עבור הפרויקט
15	מוטיבציה
22	<b>חלק 1: חיזוי סטטיסטיקה לשחקני NBA</b>
22	תיאור פתרון הבעיה
22	סידור והכנת המידע
33	שיטות ומטריקות להערכת ביצועים
36	אלגוריתמי למידה לשלב החיזוי
41	תיאור מבנה מערכת החיזוי
44	ניסויים למערכת החיזוי
44	מתודולוגיה ניסויית
47	שלב 1 – תוצאות הניסויים – ניתוח ומסקנות
52	שלב 2 – תוצאות הניסויים – ניתוח ומסקנות
54	תוצאות ומסקנות סופיות למערכת החיזוי
56	<b>חלק 2: סוכן למשחק דראפט הפנטזי</b>
56	תיאור פתרון הבעיה
56	חוקי משחק הדראפט
56	הנחות מקלות
57	שימוש בנתוני החיזוי
58	הערכת תוצאות משחק הדראפט
60	אסטרטגיות להתמודדות במשחק דראפט מרובה סוכנים
66	תיאור מבנה המערכת
69	מעטפת המשחק
70	מימוש סוכני משחק הדראפט
72	מימוש ההיוריסטיקות ופונקציות יעילות (Utility)
74	ניסויים למערכת סוכן משחק הדראפט
74	מתודולוגיה ניסויית
77	שלב הניסויים הסטנדרטיים – תוצאות ומסקנות
81	שלב הניסויים המתקדמים – תוצאות ומסקנות
85	שלב הניסוי המסכם – תוצאות ומסקנות
88	ניסויים משניים – תוצאות ומסקנות
90	<b>סיכום</b>

93	נספח א' – תוצאות מודלים נבחרים מניסוי החיזוי שלב 1
94	נספח ב' – ניסוי החיזוי שלב 2 – פרטי המודלים
96	נספח ג' – מערכת החיזוי – תוצאות סופיות
100	נספח ד' – תיאור הניסויים הסטנדרטים – מערכת סוכן הדראפט
103	נספח ה' – תיאור הניסויים המתקדמים – מערכת סוכן הדראפט
106	מקורות מידע

## הקדמה

ליגת ה-NBA (National Basketball Association) הינה ליגת הכדורסל הבכירה בארה"ב, הנחשבת בעיני רבים כליגת הכדורסל הטובה בעולם. במסגרת העונה הסדירה של הליגה מתחרות 30 קבוצות במשחקים חוזרים ויום-יומיים כדי לקבוע מי תדורג גבוה יותר בטבלה בסיום העונה.

**משחק הפנטזי NBA (Fantasy NBA)**, להלן: "פנטזי" הינו משחק המתנהל בין אנשים באמצעות מספר פלטפורמות על-גבי רשת האינטרנט במקביל לעונה הסדירה של ליגת ה-NBA. במשחק זה, כל שחקן צריך להרכיב קבוצה (תחת הגבלות שונות), המורכבת משחקני NBA אמיתיים. הקבוצות בליגת פנטזי מתחרות ב-matchups שבועיים אחת נגד השניה, כאשר הניקוד מושפע מהתוצאות אותן השיגו שחקני ה-NBA במשחקיהם היומיים בליגה האמיתית, ובפרט מהישגיהם הסטטיסטיים כגון קליעת נקודות. הרכבת הקבוצה מתבצעת בתחילת העונה, במסגרת תחרות בין משתתפי הליגה הנקראת "דראפט" (לפירוט נרחב על החוקים ואופי המשחק ראו פרק תיאור "משחק הפנטזי NBA"). **לצורך הצלחה במשחק הפנטזי ובהרכבת קבוצה מנצחת, יש לאמוד את טיב שחקני ה-NBA, ולנסות לחזות את מידת הצלחתם בעונה הסדירה, וכן, לקחת בחשבון משתנים שונים בעת בניית הקבוצה בדראפט**, תחת מגבלות חוקי המשחק והעובדה שהדראפט הינו למעשה תחרות מול שחקנים נוספים, אשר מנסים גם הם להשיג את התוצאה הטובה ביותר.

**מטרתנו בפרויקט זה הינה לספק מערכת לומדת אשר חוזה את איכות שחקני ה-NBA בעונה הקרובה**, על ידי מתן הערכה למידת הצלחתם הסטטיסטית, **ובנוסף, בניית שחקן אוטונומי**, אשר על בסיס החיזוי הסטטיסטי, ינהל משחק דראפט מול יריביו במטרה להרכיב קבוצה עם סיכויים גבוהים לנצח בליגת הפנטזי.

**הפרויקט כולל שני חלקים אשר יוצגו בדו"ח זה:**

### 1. חלק החיזוי – פיתוח מערכת לומדת לחיזוי הנתונים הסטטיסטיים של שחקני ה-NBA ב-9

**קטגוריות שונות**, וחיזוי דירוג פנטזי כללי של כל שחקן. לכל אחת מהקטגוריות נבחר estimator מתאים, אשר במסגרת הניסויים השיג את התוצאות הטובות ביותר עבור הקטגוריה, ויחד הם מהווים מערכת שלמה אשר מאפשר לחזות, בהינתן נתונים שונים על ביצועי השחקנים בעונה מסוימת (ומידע נוסף), את הסטטיסטיקה שלהם בעונה הבאה.

### 2. חלק שחקן הדראפט – פיתוח שחקן עצמאי המנהל משחק דראפט, כלומר, מרכיב קבוצת שחקני

NBA לפי הפרמטרים של ליגת פנטזי נתונה (מספר קבוצות מתחרות, כמות השחקנים בכל קבוצה וכו'). במסגרת הפיתוח נבחנו אלגוריתמים שונים לחיפוש במרחב המצבים אשר הוגדר עבור משחק הדראפט. התוצאות שהושגו מאפשרות בחירה של אסטרטגיות משחק עדיפות, בהתאם לפרמטרי המשחק השונים.

כלל הפרויקט (מערכת החיזוי והשחקן) נכתב בשפת **Python** ועם שימוש נרחב בסיפריות הבאות :

- **pandas** – ספריית עזר לעיבוד ואנליזה של מידע. הסיפרייה מספקת כלים שונים לקריאה, עיבוד וניתוח של מידע טבלאי באמצעות מבנה הנתונים Data Frame.
- **sklearn** – ספריית עזר המספקת כלים יעילים לקריאה ואנליזה של מידע. הסיפרייה מספקת כלים רבים לאימון מערכת באמצעות מודלי למידה שונים.

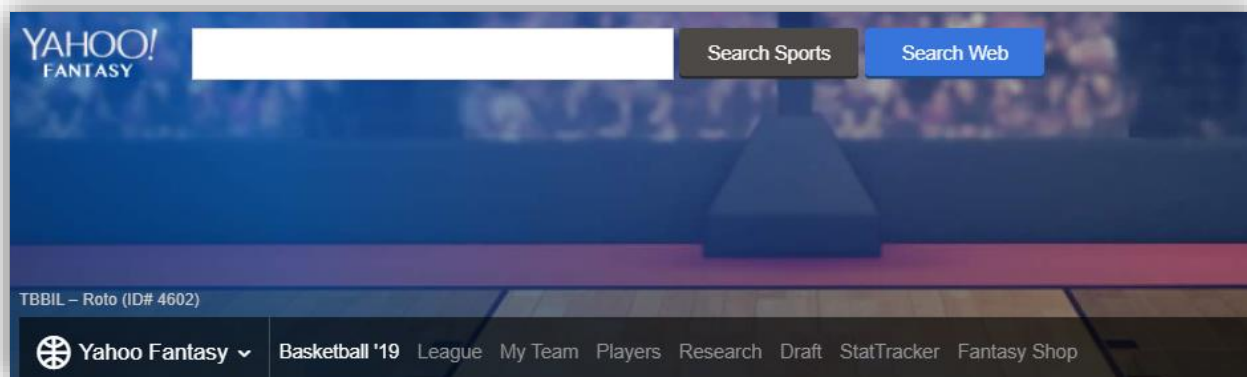
**הרעיון לפרויקט עלה מתוך עניין אישי** – לחלק מחברי הצוות יש ניסיון של מעל 10 שנים במשחק בליגות פנטזי שונות, וכן עניין רב בליגת ה-NBA ובסטטיסטיקות השונות של המשחק. הניסיון במשחק מסוג זה תרם רבות במהלך הפרויקט, כשעלה הצורך לאפיין את התכונות הנדרשות מהמערכת, הגדרת אסטרטגיות אפשריות לשחקן ופיתוח שיטות הערכה להצלחת מערכת החיזוי והשחקן האוטונומי.

## הגדרת הבעיה

### משחק ה-Fantasy NBA

ליגות "פנטזי ספורט" הינן משחק נפוץ בקרב אוהדי ספורט ברחבי העולם, וכוללות פלטפורמות שונות ומגוונות עבור סוגי ספורט קבוצתי שונים כגון כדורסל, כדורגל, פוטבול, בייסבול ועוד.

משחק הפנטזי NBA הינו תחרות עונתית בין אנשים המשחקים בפלטפורמות שונות על-גבי רשת האינטרנט. ישנם אתרים רבים המספקים פלטפורמה לניהול ליגות פנטזי, כאשר המובילים שבהם הינם אתרי הספורט של Yahoo<sup>i</sup> ו-ESPN<sup>ii</sup>, **בפרויקט זה אנו מתמקדים בליגות המשוחקות בשיטה המוצעת על-ידי הפלטפורמה של Yahoo שהינה הפופולרית ביותר**. כמו כן, ישנן מספר שיטות משחק וניקוד בליגות הפנטזי, אנו נתמקד בשיטת המשחק המכונה Head-2-Head עם 9 קטגוריות משחק, כפי שיפורט בהרחבה בהמשך.



איור 1: פורטל משחק הפנטזי באתר Yahoo

לפני שנצלול לעומק חוקי המשחק, נתעכב על מספר הגדרות טרמינולוגיות:

- **קבוצה** – כל שחקן (אנושי) המשתתף בליגת הפנטזי הוא למעשה "בעלים" של קבוצה. מעתה נתייחס למשתתפי ליגת הפנטזי בתור "קבוצות".
- **שחקן** – הכוונה לשחקן בליגת ה-NBA.
- **קטגוריות** – הינן קטגוריות סטטיסטיות אשר לפיהן נמדדים שחקני ה-NBA ועליהן מתחרות הקבוצות בליגת הפנטזי.

## חוקי משחק הפנטזי NBA בשיטת Head-2-Head עם 9 קטגוריות

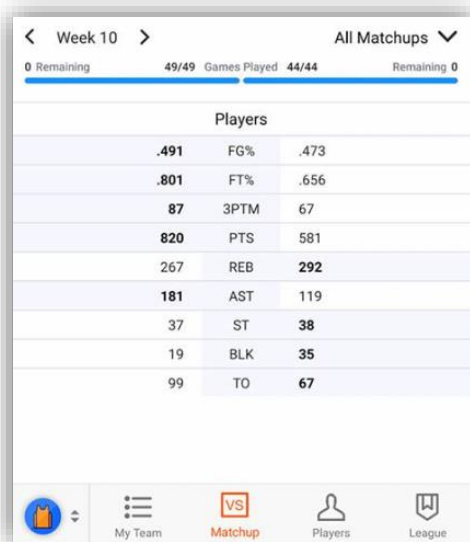
1. בתחילת העונה (סמוך לתחילת עונת המשחקים בליגת ה-NBA) מרכיבה כל קבוצה בליגת הפנטזי קבוצת שחקנים, במסגרת משחק הדראפט אשר חוקיו יפורטו בהמשך.

2. בכל שבוע, ישנם matchups שבועיים בין קבוצות הליגה, אשר מתנהלים באופן הבא :

א. בכל ערב, לפני תחילת המשחקים בליגת ה-NBA, מרכיב מנהל הקבוצה את הסגל לערב המשחקים, כלומר, בוחר אילו שחקנים מקבוצתו ייכללו בסגל לערב זה. השחקנים שנבחרו יצברו עבור הקבוצה את הסטטיסטיקות ששחקן ה-NBA האמיתי השיג במשחקו באותו ערב.

ב. לדוגמא, נניח שבליגת הפנטזי ישנן שתי קבוצות A ו-B. הקבוצה A מחזיקה בשחקן "לברון ג'יימס", והקבוצה B מחזיקה בשחקן "קובי בראיינט", אשר נבחרו בדראפט בתחילת העונה (השחקנים הללו הינם שחקנים אמיתיים בליגת ה-NBA). בשבוע כלשהו במהלך העונה, יתנהל matchup בין קבוצה A ל-B, ונניח כי בערב נתון במהלך שבוע זה השחקן לברון ג'יימס שיחק (בליגת ה-NBA האמיתית) וקלע 30 נקודות, והשחקן קובי בראיינט שיחק גם הוא וקלע 25 נקודות. נקודות אלו מצטרפות למניין הנקודות הכולל שקלעו שחקני הקבוצות A ו-B (בהתאמה) באותו שבוע.

ג. בסוף ה-matchup נמדד ההישג המצטבר של כל קבוצה ב-9 הקטגוריות הסטטיסטיות (באופן שתואר לעיל), והקבוצה שמובילה במירב הקטגוריות מנצחת את ה-matchup. כלומר, עליונות בלפחות 5 קטגוריות מבטיחה ניצחון שבועי.



Week 10		All Matchups	
0 Remaining	49/49 Games Played	44/44	Remaining 0
Players			
.491	FG%	.473	
.801	FT%	.656	
87	3PTM	67	
820	PTS	581	
267	REB	292	
181	AST	119	
37	ST	38	
19	BLK	35	
99	TO	67	

איור 2: תוצאות Matchup שבועי בליגת פנטזי באתר Yahoo

3. תשע הקטגוריות הנבחנות בליגת Head-2-Head סטנדרטית, ונלקחו בחשבון במסגרת הפרויקט, הינן –

קטגוריה	תרגום	קיצור	שם הקטגוריה	תיאור נוסף
נקודות	Points	נק'	PTS	
אסיסטים	Assists	אס'	AST	
ריבאונדים	Rebounds	ריב'	TRB	
חטיפות	Steals	חט'	STL	
חסימות	Blocks	חסי'	BLK	
אחוזי קליעה מהשדה	Field Goal %	-	FG%	כמות הקליעות ששחקן קלע מתוך כמות הזריקות הכוללת שלו
אחוזי קליעה מהעונשין	Free Throw %	-	FT%	כמות הקליעות ששחקן קלע מקו העונשין, לאחר שבוצעה עליו עבירה, מתוך כמות הזריקות הכוללת שלו מקו העונשין
כמות קליעות לשלוש	3 Points Made	-	3P	כמות הקליעות של שחקן מעבר לקו שלוש הנקודות
איבודי כדור	Turnovers	איב'	TOV	זוהי קטגוריה שלילית – יש להשיג כמה שפחות איבודים

4. למשחק ישנם אלמנטים נוספים, כגון החלפת שחקנים בין קבוצות במהלך העונה (טריידים) או הרמת שחקנים חופשיים (שחקני NBA שלא נבחרו לאף קבוצה בליגת הפנטזי). אלמנטים אלו לא נבחנו במסגרת הפרויקט ועל כן לא נרחיב עליהם.

5. כמו כן, ישנן שיטות משחק נוספות, בהן למשל ניצחון ב-matchup אינו מזכה בנקודה בודדת (ניצחון שבועי) אלא לאורך כל העונה נסכמות מספר הקטגוריות בהן הושג ניצחון (למשל, ניצחון שבועי ב-6 קטגוריות מזכה ב-6 נקודות). עוד שיטה פופולרית הינה ליגות בשיטת Rotisserie, בהן הסטטיסטיקות שמשיגים השחקנים בליגת ה-NBA נסכמות כל ערב עבור קבוצת הפנטזי שלהן, ובסוף העונה מתבצעת השוואה בין כל קבוצות ליגת הפנטזי בכל קטגוריה, וכך נקבעת זהות הקבוצה המנצחת של העונה.

### חוקי משחק הדראפט

1. כפי שהוזכר לעיל, הרכבת קבוצת הפנטזי מתבצעת בתחילת העונה במסגרת תחרות בין קבוצות הליגה במשחק בחירות הנקרא "דראפט". הדראפט הינו חלק משמעותי מאוד מהעונה, ושחקני פנטזי מנוסים מגיעים אליו כשהם מוכנים עם אסטרטגיות שונות ומהלכים לביצוע בזמן אמת בהתאם לתוצאות המשתנות של המשחק.



2. לצורך תיאור המשחק נניח ליגת פנטזי עם 3 קבוצות (A, B, C), כשכל קבוצה מורכבת מחמישה שחקנים (ליגות פנטזי לרוב גדולות בהרבה, כ-12 קבוצות בגודל 13 שחקנים מהווה ליגה סטנדרטית). לפני התחלת הדראפט, **מוגדר סדר הבחירות באופן רנדומלי**, נניח כי תוצאת ההגרלה הינה A בוחר ראשון, B בוחר שני ו-C בוחר שלישי. **הדראפט מתנהל בסיבובים** – בכל סיבוב כל קבוצה יכולה לבחור שחקן אחד, כלומר, במקרה המתואר לעיל, יתקיימו חמישה סיבובים לדראפט. הסיבובים מתבצעים בסדר "נחש" – בסיבובים אי-זוגיים סדר הבחירות לפי ההגרלה, ובסיבובים זוגיים בסדר הפוך, כך שבדוגמא שלנו סדר הבחירות יהיה – A, B, C, C, B, A, A, B, C, C...

3. **הבחירות מתבצעות מתוך מאגר השחקנים הפנויים**. בתחילת הדראפט, המאגר מלא בכל שחקני ה-NBA (האמיתיים, החתומים על חוזה בקבוצת NBA כלשהי בליגה באותה עונה). בכל פעם ששחקן נבחר על-ידי קבוצה הוא מוצא מהמאגר, ואף קבוצה אחרת לא יכולה לבחור אותו יותר. המאגר והבחירות חשופות לכל המשתתפים לכל אורך הדראפט. כמו כן, כל בחירה מוגבלת בזמן, לרוב דקה אחת במהלכה יש לבחור, אחרת מתבצעת בחירה אוטומטית על סמך רשימת דירוג מוכנה מראש.

4. בנוסף, **במרבית הליגות ישנן הגבלות על עמדות השחקנים בכל קבוצה** (כלומר, זוהי בעיה עם היבט של סיפוק אילוצים). לכל שחקן NBA יש עמדה (או מספר עמדות) על מגרש הכדורסל שבה הוא משחק, מתוך 5 עמדות המוגדרות למשחק הכדורסל. באתר הפלטפורמה שבה משוחקת ליגת הפנטזי (למשל, Yahoo) מגדירים טרום העונה את העמדות של כל השחקנים, וכך, כל קבוצה בליגת הפנטזי צריכה להתאים את השחקנים שהיא בוחרת לעמדות שהוגדרו בליגה שבה היא משתתפת (זוהי מגבלה גמישה שנקבעת על-ידי משתתפי ליגת הפנטזי).

## דגשים, גישות ואסטרטגיות למשחק הפנטזי

### 1. חשיבות החיזוי:

א. כפי שניתן להבין מהמתואר לעיל אודות משחק הפנטזי, **ישנה תלות מהותית בחיזוי מדויק ככל האפשר של הסטטיסטיקות של שחקני ה-NBA טרום העונה**. לרוב, שחקני הפנטזי נעזרים בחיזויים שמספקות פלטפורמות המשחק השונות כגון Yahoo ו-ESPN או אתרי כדורסל אחרים כגון אתר Basketball Monsters<sup>iii</sup> המוביל בתחום. החיזויים הללו נבנים על ידי מומחי כדורסל ופנטזי, אשר משקללים נתונים רבים כדי להעריך כמה נקודות למשחק ייקלע כל שחקן, כמה אסיסטים ימסור בממוצע למשחק וכו'.

ב. בנוסף לחיזוי הסטטיסטיקות, **ישנו מדד על המכונה "דירוג פנטזי"**, המנסה לדרג את כלל השחקנים בליגת ה-NBA לפי איכותם עבור משחק הפנטזי. מדד זה מנסה לשקלל לתוכו את כלל החיזויים הסטטיסטיים של השחקנים, יחד עם אלמנטים נוספים (המפורטים

בהמשך), כדי לתת מעין דירוג מוחלט. המדד הנ"ל הינו שימושי מאוד עבור שחקני הפנטזי, בעיקר לצורך סינון ראשוני של שחקנים במהלך הדראפט – לרוב כאשר מגיע תור הקבוצה לבחור בדראפט, ותחת מגבלת הזמן, נוח לסנן כ-20-10 שחקנים פנויים (שטרם נבחרו) אשר מדורגים גבוה לפי מדד דירוג הפנטזי, ורק עבורם לנסות ולבצע ניתוח מעמיק יותר של תרומתם האפשרית לקבוצה.

Rank	Player	ADP	FGM	FGA	FG%	FTM	FTA	FT%	3PTM	PTS	REB	AST	ST	BLK	TO
5	Giannis Antetokounmpo Mil – SF,PF	3.6	776	1362	570	551	737	748	78	2181	938	443	96	115	280
6	Nikola Jokic Den – PF,C	6.4	607	1203	505	311	368	845	99	1624	846	591	98	54	242
7	Damian Lillard Por – PG	8.5	682	1525	447	452	500	904	246	2062	360	534	85	30	214
8	Bradley Beal Was – SG	11.5	750	1598	469	373	462	807	230	2103	392	422	107	55	214
9	Kyrie Irving Bkn – PG,SG	12.6	622	1302	478	271	303	894	194	1709	333	450	110	27	184
10	Deandre Ayton Pho – C	27.8	631	1050	601	189	247	765	6	1457	861	173	85	114	165
11	Paul George LAC – SF,PF	13.5	532	1233	431	342	399	857	229	1635	484	274	137	27	165
12	Kawhi Leonard LAC – SG,SF	9.9	577	1164	496	380	451	843	120	1654	465	193	121	36	128
13	Andre Drummond Det – PF,C	20.7	592	1091	543	265	435	609	5	1454	1301	132	123	146	193
14	Mitchell Robinson NY – C	27.8	338	489	691	129	209	617	0	805	641	56	79	236	53
15	Kemba Walker Bos – PG	16.6	615	1400	439	354	413	857	225	1809	331	444	94	23	159
16	Jimmy Butler Mia – SG,SF	16.3	511	1105	462	409	477	857	110	1541	395	336	128	30	136
17	Joel Embiid Phi – PF,C	7.9	578	1187	487	527	642	821	88	1771	820	210	52	138	224
18	Nikola Vucevic Ori – PF,C	24.3	614	1271	483	155	201	771	92	1475	859	272	77	84	144
19	Myles Turner Ind – PF,C	23.5	422	861	490	184	241	763	84	1112	575	125	65	220	108
20	Jrue Holiday NO – PG,SG	19.1	582	1218	478	247	317	779	142	1553	385	519	118	58	208

איור 3: חיזוי תרומת העונה המסופק על ידי אתר Yahoo

## 2. הערכת טיב הקבוצה:

א. מכיוון שהדראפט מתנהל לפני תחילת העונה, וכולו נסמך על חיזוי והערכות ביצועיהם של שחקני ה-NBA בעונה הבאה, ישנו קושי להעריך את טיב הקבוצה שמרכיבים במהלך ולאחר הדראפט. לא קיים מדד מוחלט אשר מגדיר מהי קבוצה מנצחת.

ב. מדד מוביל (אך אינו בהכרח מדויק), אשר מסופק לשחקני הפנטזי במהלך הדראפט בפלטפורמת המשחק של Yahoo הינו המיקום היחסי של הקבוצה, על פי השחקנים שנבחרו עד כה, בכל אחת מ-9 הקטגוריות, אל מול שאר קבוצות הליגה. באמצעות מדד זה ניתן להעריך לדוגמא תוך כדי הדראפט האם לקבוצה שלי חסר שחקן שקולע הרבה נקודות, חוטף הרבה כדורים או מוסר הרבה אסיסטים, ולבחור בהתאם.

Projected Stats		This table shows cumulative stats for all teams based on players they have drafted.									
Rank	Team	FG%	FT%	3PTM	PTS	REB	AST	ST	BLK	TO	Total
1	Ofirg6	.521	.824	272.0	5424.0	2117.0	803.0	261.0	327.0	579.0	82
2	Team 2	.459	.852	389.0	3549.0	1067.0	751.0	212.0	270.0	484.0	65
3.5	Team 7	.443	.883	471.0	3871.0	691.0	978.0	179.0	53.0	373.0	61
3.5	Team 3	.525	.816	358.0	2875.0	1020.0	494.0	178.0	252.0	272.0	61
5	林成宇	.499	.711	235.0	3557.0	1693.0	554.0	230.0	201.0	407.0	60
6	JY	.490	.792	374.0	3991.0	1669.0	790.0	170.0	165.0	503.0	59.5
7	Team 5	.528	.753	170.0	3656.0	1797.0	715.0	173.0	199.0	424.0	58
8	Team 11	.486	.863	314.0	3363.0	798.0	643.0	231.0	63.0	312.0	56.5
9	Team 6	.465	.847	325.0	3432.0	1150.0	1266.0	183.0	69.0	561.0	55
10	Team 10	.486	.766	264.0	3428.0	972.0	1000.0	225.0	73.0	432.0	52
11	Carlo	.508	.743	229.0	2840.0	1465.0	430.0	201.0	213.0	291.0	50
12	Team 8	.482	.807	230.0	3324.0	1205.0	729.0	170.0	196.0	432.0	42

איור 4: מסך דירוג קבוצות הליגה בכל אחת מקטגוריות המשחק, מתוך פלטפורמת משחק הדראפט של Yahoo

### 3. אסטרטגיית Punt (פאנט):

א. מרבית שחקני ה-NBA מתמחים באספקטים ספציפיים במשחק הכדורסל. מעטים השחקנים שמספקים שורה סטטיסטית עם נתונים גבוהים בכל קטגוריות המשחק. לרוב ישנו Trade-off בין קטגוריות שונות, למשל, שחקן שקולע הרבה שלשות (3P) לרוב יהיה שחקן נמוך, ולכן ייקח פחות כדורים חוזרים (TRB) או יחסום פחות זריקות (BLK). עובדה זו מקשה על הרכבת קבוצה שתצליח להוביל בכל הקטגוריות הסטטיסטיות.

ב. כתוצאה מה-Trade-off שצויין להלן, עולה אסטרטגיה, אשר מובילה במשחק הפנטזי בפורמט Head-2-Head שתואר לעיל, והנחתה אותנו רבות בעת פיתוח מערכת השחקן האוטונומי – **אסטרטגיה זו מכונה Punt**. כאמור, על פי הגדרת ניצחון ב-matchu, בתום שבוע של משחקים יש להוביל על הקבוצה היריבה בלפחות 5 מתוך 9 קטגוריות כדי להבטיח ניצחון. לפיכך, **אסטרטגיה אפשרית לבניית קבוצה הינה ויתור על מספר קטגוריות כדי להבטיח עליונות ברוב של לפחות 5 קטגוריות**.

ג. לדוגמא, נניח שבסיבובים הראשונים בדראפט נבחרו ע"י הקבוצה 1-2 שחקנים אשר קולעים הרבה שלשות ומוסרים הרבה אסיסטים. כפי שהוסבר בסעיפים הקודמים, קיים Trade-off, ולכן סביר להניח כי שחקנים אלו לוקחים פחות כדורים חוזרים וקולעים באחוזים פחות טובים מהשדה (FG% נמוך). בשלב כזה במהלך הדראפט ניתן להחליט על אסטרטגיית "פאנט כדורים חוזרים", כלומר, להתעלם מהעובדה שהקטגוריה הזו קיימת, ולבצע את הבחירות הבאות ללא התחשבות בכמות הכדורים החוזרים שהשחקנים הנבחרים לוקחים. בשיטה זו (ועקב ה-Trade-off הקיים), סביר להניח שנבחר שחקנים שיחזקו את הקבוצה עוד יותר בקטגוריות שכבר חזקות (שלשות ואסיסטים), ובכך יבטיחו עליונות בקטגוריות אלו על פני מרבית הקבוצות בליגה.

ד. חשוב לציין כי הדוגמא הנ"ל משטחת את האסטרטגיה בהסתכלות על Trade-off יחיד, וכפי שהוסבר, חשוב להשיג עליונות ברוב של קטגוריות מתוך 9 קטגוריות המשחק. **מניסיונינו, ניסיון ליצור קבוצה מאוזנת ש"טובה בהכל", כלומר, בכל שלב בדראפט להישאר באזור הממוצע בדירוג בכל קטגוריה, לרוב נכשל במבחן התוצאה.**

#### 4. אלמנטים נוספים בעלי השפעה על המשחק:

א. ישנם אלמנטים נוספים הקיימים בליגת ה-NBA ומשפיעים ישירות על תוצאות משחק הפנטזי, על איכות השחקנים עבור המשחק וטיב הקבוצה. אלמנטים אלו אינם בהכרח סטטיסטיקות של השחקן וקשה הרבה יותר לאמוד ולחזות אותם.

ב. האלמנט המוביל הינו **פציעות** – שחקני NBA הינם ספורטאים מקצוענים המשחקים ברמה גבוהה מאוד, ועל כן לעיתים נפצעים ומחמיצים משחקים ואף חלקים נכבדים מהעונה הסדירה. במצב שבו שחקן נפצע במציאות ולא ישתתף במשחקי קבוצת ה-NBA, קבוצת הפנטזי המחזיקה את השחקן תהיה בחיסרון משמעותי, כיוון שתפסיד שחקן שאמור לתרום לקטגוריות הסטטיסטיות הנצברות. שחקני פנטזי לרוב עוקבים באדיקות אחר ליגת ה-NBA, מכירים את השחקנים ויודעים מי יותר מועד לפציעות, ולוקחים זאת בחשבון בעת הרכבת הקבוצה.

ג. אלמנט נוסף אשר קשה במיוחד להעריך באמצעות מודל ממוחשב הינו **הרכב הקבוצה (האמיתית) שבה שחקן משחק** – לדוגמא, נניח כי שחקן המתמחה בקליעת שלשות עבר קבוצה והגיע לקבוצה שבה יש עוד שחקן, המשחק בדיוק באותה עמדה ומתמחה גם כן בקליעת שלשות. סביר להניח במצב כזה כי התפוקה של שני השחקנים בקטגוריית הקליעות לשלוש תרד, נתון שישפיע משמעותית על ערכם של השחקנים הללו במשחק הפנטזי. ריבוי שחקנים באותה עמדה ובעלי יכולות דומות באותה הקבוצה, כמו גם סגנון המשחק של הקבוצה שבה נמצא השחקן (משחק מהיר, איטי, אופי הגנתי וכד') הינם אלמנטים משמעותיים עבור חיזוי ביצועיו של שחקן.

ד. **ישנו משקל שונה עבור משחק הפנטזי לכל קטגוריה סטטיסטית**, דבר הנובע מהממוצע הכללי בליגה בכל קטגוריה. למשל, ישנם מעט מאוד שחקנים, יחסית למספר השחקנים בליגה, שחוסמים הרבה פעמים במשחק. לעומת זאת, שחקנים שקולעים הרבה קליעות ל-3 נקודות יש בשפע. על כן, לרוב נעריך יותר שחקנים שמתמחים בחסימות לעומת שחקנים שמתמחים בשלשות. הנתון הנ"ל חשוב מאוד במהלך הדראפט, שכן בשלבים המוקדמים לרוב כדאי לבחור שחקנים המתמחים בקטגוריות "הפחות נפוצות" (כגון חסימות), כיוון שלא יהיה ניתן להשלים את הקבוצה עם בחירה של שחקנים כאלו בשלבים מאוחרים יותר.

ה. כמו כן, למשחק הפנטזי אלמנטים נוספים אשר לא נלקחו בחשבון במסגרת הפרויקט. הפרויקט התמקד בפיתוח שחקן עבור שלב הדראפט, והערכת השחקן לפי ביצועיו בדראפט בלבד. כפי שצויין, במהלך העונה ישנם מהלכים נוספים שקבוצה בליגת הפנטזי יכולה לבצע כגון טריידים (לשינוי אסטרטגיית הקבוצה, חיזוק ה-Punt וכו'), החלפת שחקנים חופשיים ועוד.

## **הגדרת הבעיה עבור הפרויקט**

כפי שניתן ללמוד מההסבר אודות משחק הפנטזי, מדובר במשחק רחב ומגוון בעל מספר גרסאות גדול ומספר רב של אלמנטים המשפיעים על תוצאותיו. **בפרויקט שלנו החלטנו להתמקד במספר חלקים נבחרים מתוך המשחק, המתאימים למסגרת הזמנים ולהגדרות הפרויקט שניתנו במסגרת הקורס.** אנו מתמקדים כאמור במשחק הפנטזי בגירסת Head-2-Head עם ניצחונות שבועיים על פני 9 הקטגוריות הנבחרות שצויינו לעיל.

בחרנו בגירסה זו של המשחק ממספר סיבות. ראשית, **זו אחת הגרסאות הפופולריות יותר של המשחק.** בנוסף, יש לחלקינו ניסיון במשחק בעיקר בגירסה זו, דבר אשר איפשר העלאה וניתוח של רעיונות מגוונים מתוך הניסיון האישי להתמודדות עם הבעיה. כמו כן, **המידע שנדרש לספק עבור כל שחקן NBA בשביל לשחק בגירסה הזו הינו פשוט יותר מבגרסאות אחרות וההנחה היא שקל יותר לאמוד אותו על סמך נתוני עבר ומשתנים נוספים** – יש לספק סטטיסטיקה "פר משחק" של כל שחקן NBA – נתון בעל שונות לא גדולה בין עונות עבור כל שחקן. מעבר לכך, **נתון זה הוא בעל ערך גם לגרסאות נוספות של המשחק ומהווה מעין אבן בסיס, הנדרשת עבור כל גירסה של משחק הפנטזי.**

חשוב לציין כי לצורך התאמת הבעיה למסגרת הזמנים והיקף העבודה של הפרויקט **התבצעו מספר הנחות מקלות וכן הושמטו מספר אלמנטים אשר בגירסה המלאה של המשחק עשויים להיות קריטיים,** כגון התחשבות בפציעות שחקנים, התייחסות לשחקנים בעונתם הראשונה בליגה (עבורם לא קיים בסיס מידע מעונות קודמות), הגבלת הזמן על תור בחירה במהלך הדראפט, גודל הליגה וכד'.

נציג להלן את **מטרות העל של הפרויקט, מרכיבי הבעיה איתם נאלצנו להתמודד במהלך העבודה על הפרויקט והשיקולים בהגדרת הבעיה באופן הזה, אל מול ההגדרה הכוללת של משחק הפנטזי.** בפרקים הבאים בדו"ח מובא פירוט נרחב על אופן ההתמודדות עם הבעיות, תהליך הפיתוח, הניסויים והתוצאות הסופיות.

## מטרת העל

הפרויקט מורכב משתי מטרות מרכזיות, כאשר עבור כל מטרה מומשה מערכת נפרדת:

1. **מערכת החיזוי** – חיזוי ההישגים של שחקני ה-NBA בעונה נתונה ב-9 קטגוריות המשחק ובמדד דירוג הפנטזי הכללי.
2. **שחקן דראפט אוטונומי** – הרכבת קבוצת פנטזי במסגרת משחק הדראפט תחת הגבלות נתונות – מספר משתתפים, גודל קבוצה, הגבלת עמדות ומיקום בחירה רנדומלי.

## פירוט מרכיבי הבעיה

### 1. שלב החיזוי

- א. בשלב זה אנו נדרשים לספק חיזוי, זהה לזה שניתן ע"י פלטפורמות משחק הפנטזי, עבור ההישגים בקטגוריות הסטטיסטיות של כל אחד משחקני ה-NBA, לקראת עונה נתונה.
- ב. לצורך כך אנו נדרשים לאפיין כיצד מוגדר שחקן NBA (אילו תכונות מגדירות שחקן), לרכז את המידע הנדרש ולהשתמש בו למטרת אימון מערכת לומדת אשר תספק את החיזוי לכל אחת מהקטגוריות בנפרד.
- ג. כמו כן, ישנו צורך מרכזי בהגדרת מדד אמין ומוחלט ככל הניתן לטיב החיזוי בכל קטגוריה.

### 2. שלב הדראפט

- א. בשלב זה אנו נדרשים להרכיב קבוצת פנטזי, תחת מגבלות משחק הדראפט בליגה נתונה, מול מתחרים נוספים שאסטרטגיית המשחק שלהם לא ידועה ומשאבי המשחק (השחקנים שניתן לבחור) הינם מוגבלים.
- ב. לצורך כך אנו נדרשים למצוא אלגוריתם מתאים (סוכן) אשר בכל תור יבצע בחירה של שחקן תחת המגבלות הקיימות, כך שלבסוף, הקבוצה שלו תהיה טובה ככל שניתן.
- ג. כאמור, הגדרת "ניצחון" בדראפט אינה מוחלטת, ועל כן, נדרש להגדיר פונקציית יעילות (Utility) אשר באמצעותה נאמוד את הצלחת האלגוריתם אל מול היריבים.
- ד. כמו כן, לצורך ביצוע הבחירות, יש צורך בחיזוי המדדים הסטטיסטיים הצפויים של כל שחקן NBA, כפי שקיים במשחק הפנטזי האמיתי. בשלב זה השתמשנו בחיזוי המסופק מהמערכת בשלב 1 של הפרויקט (שלב החיזוי).

## מוטיבציה

משחק הפנטזי הינו משחק מעניין, מאתגר ומורכב, המכיל אלמנטים רבים בהם יש להתחשב בכדי להגיע לתוצאות טובות, יחד עם מימד תמידי של אי-וודאות. המשחק כולל צורך בביצוע החלטות בזמן אמת, מעקב וניתוח יום-יומי של מאגרי נתונים שונים וניהול אינטרקציה מול אנשים אחרים. **כל אלו מגדירים בעיה שאינה טריוויאלית לפיתרון, וניתן למצוא מספר רב של גישות ואסטרטגיות לפיתרון, כאשר אף אחת מהן אינה מבטיחה ניצחון מוחלט, וכל שינוי בגישה עשוי לשפר או לפגוע במידת ההצלחה בצורה ניכרת.**

מערכת אשר משלבת את שני החלקים הנחוצים למשחק – החיזוי והרכבת הקבוצה בדראפט, **תוכל לשמש שחקני פנטזי בעולם האמיתי, ולתרום בבנייה ותכנון של אסטרטגיות לקראת עונת ה-NBA.** כיום, לדוגמא, קיימת פלטפורמה באתרי המשחק המאפשרת, טרום העונה, להתנסות במשחק הדראפט בתחרות מול שחקנים אמיתיים או שחקנים מלאכותיים, המשתמשים באסטרטגיה שטחית בלבד, תוך שימוש בנתוני החיזוי של מומחי האתרים. המערכת המפותחת בפרויקט זה תאפשר התנסות במשחק הדראפט מול שחקנים בעלי אסטרטגיה מתקדמת, ובנוסף תספק חיזויים מבוססי למידה, **ובכך עשויה לתרום לשחקני הפנטזי להיערך בצורה מיטבית לעונת המשחק.**

**השילוב בין הצורך בחיזוי ההתנהגות וההישגים של שחקני ה-NBA, לבין ניהול משחק רב משתתפים, מגדירים בעיה מעניינת, המתאימה במיוחד לפרויקט המשלב בינה מלאכותית ומערכות לומדות.** הכלים הקיימים בתחומים אלו מאפשרים התמודדות עם הבעיה במגוון דרכים, ולהערכתנו ניתן להגיע איתם להישגים טובים (הן בחיזוי והן בניהול דראפט), אשר ניתן יהיה מאוחר יותר אף ליישם, בתור שחקני פנטזי בעולם האמיתי.

הסיבה המרכזית שהובילה לבחירת הפרויקט הנ"ל הינה הפלטפורמה שהוא מהווה להתנסות בפועל בכלים השונים שלמדנו בקורס המבוא. **החלוקה לשתי מערכות המבוססות על עקרונות שונים שנלמדו בקורס איפשרה לנו להעמיק את הידע במגוון נושאים.** מימוש הסוכן למשחק הדראפט איפשר לנו להתנסות בפיתוח, בחינה ושיפור של אלגוריתמים רבים שנלמדו במהלך הקורס ומימוש המערכת הלומדת הצריך העמקה בשלל שיטות הלמידה שאת הבסיס של חלקן הכרנו בקורס. החיבור בין שתי המערכות הינו מאתגר, ומהווה הזדמנות להתנסות בפיתוח פיתרון לבעיה "מקצה לקצה", תוך שילוב מגוון רחב של טכנולוגיות מתחומים שונים בעולם הלמידה והבינה המלאכותית.

## תיאור מאגרי הנתונים ומקורות המידע

בכדי לחזות את ביצועיו הסטטיסטיים של שחקן בעונה נתונה, עלה הצורך בהשגת מגוון רחב של תכונות לאפיון כל שחקן. על כן, **הסתמכנו על נתונים ממספר מקורות מידע**. כפי שיתואר בהמשך בפרק על מערכת החיזוי, **אובייקט השחקן הוגדר על פי אוסף של תכונות, כאשר ביצענו חלוקה של כלל השחקנים עבורם נאסף מידע לפי נתוניהם ב-10 העונות האחרונות בליגת ה-NBA**. בחרנו להסתמך על 10 העונות האחרונות בלבד ממספר סיבות.

ראשית, במסגרת תקופת הזמן הזו הצלחנו להרכיב בסיס נתונים רחב עם מספר גדול של אובייקטים אשר נראה מספק עבור ביצוע הלמידה והחיזוי. בנוסף, מהיכרותנו עם ליגת ה-NBA, הערכנו כי **הסתמכות על מידע מוקדם יותר מעונת 2010 עשוי לפגוע בדיוק המערכת, היות וישנם שינויים בשיטת המשחק המודרנית בליגת ה-NBA לעומת זו שהתקיימה לפני עשור ויותר**. כיום קיימים "טרנדים" אשר משפיעים על הישגיהם הסטטיסטיים של השחקנים, כגון התמקדות בקליעת זריקות לשלוש, וקצב משחק מהיר המבוסס על יכולת אישית גבוהה, אל מול סגנון המשחק הישן אשר התבסס יותר על ניצול אלמנטים פיזיים כגון גובה ואתלטיות, זריקה ממרחק קרוב לסל וניצול שחקן גובה וחוזק באזורים הקרובים לסל. בעקבות השינוי בטרנדים בין השנים, בחרנו ב-10 העונות האחרונות, **על מנת לבסס את הנתונים המתקבלים על סגנון המשחק כיום ובכך להקטין את ההטיה של הדוגמאות ולייעל את שיטת הלמידה של המודלים שלנו**.

## פירוט הפרמטרים לאפיון שחקן

עבור כל שחקן אשר שיחק בעונות שצינו, אנו מבצעים חלוקה של התכונות המאפיינות אותו ל**נתונים יבשים (סטנדרטיים ומתקדמים)**, **נתונים כלליים ונתונים ממשחק הפנטזי המשוערכים ע"י מומחים**.

### 1. נתונים יבשים:

#### א. נתונים סטנדרטיים:

i. **נתונים אלו מכילים מידע וסטטיסטיקות כלליים** לפיהם ניתן לאמוד את טיב השחקן על פני מספר קטגוריות שונות, בפרט הקטגוריות אותן מודדים ב-matchup של משחק פנטזי. בנוסף, קטגוריה זו כוללת נתונים כמותיים שונים לפיהם ניתן לנסות לאמוד את חשיבותו של השחקן עבור קבוצתו, למשל: זמן משחק (MP – Minutes Played) ומספר זריקות לסל (FGA – Field Goal Attempts). עבור דוגמאות אלו ניתן לשער כי שחקן ששיחק יותר דקות במשחק וזרק מספר זריקות גבוה (מעל הממוצע) הינו שחקן מרכזי בקבוצה ולפיכך נעדיף אותו על פני שחקן פחות דומיננטי.

ii. את כל אחד מהנתונים המוצגים בסעיף הקודם ניתן לפלח באופן שונה על פי החלוקה הבאה:



- **ממוצע למשחק** – ממוצע של השחקן עבור קטגוריה לפי מספר המשחקים ששיחק באותה עונה (למשל שחקן שקולע 20 נק' למשחק).
- **כלל עונתי** – סך התוצאות של השחקן עבור קטגוריה על פני עונה שלמה (למשל, שחקן שבסך כל משחקיו יחד בעונה בודדת קלע 1,000 נק').
- **לפי 36 דקות משחק** – חישוב תוצאות השחקן עבור קטגוריה אילו היה משחק 36 דקות. נתון זה הינו חשוב שכן הוא יכול להעיד על יעילות (או אי יעילות) של שחקן אשר יכול להיות "מוחבא" לפי הפילוחים הקודמים. לדוגמא, עבור שני שחקנים שונים אשר שניהם קלעו 10 נקודות במשחק כאשר השחקן הראשון שיחק 5 דקות והשחקן השני שיחק 10 דקות. מהסתכלות על מספר נקודות למשחק בלבד, שני השחקנים יקבלו ערך שווה, אך סביר מאוד שנעדיף את השחקן הראשון שכן הוא היה יעיל יותר, וסביר להניח שאם יקבל יותר דקות משחק יגיע להישגים גבוהים יותר מהשחקן הראשון. לפי מדד 36 דקות השחקן הראשון יקבל ערך גבוה מאשר השחקן השני.

חשוב לציין כי פילוח הנתונים על פי הקטגוריות המצויינות לעיל הינו **מידע נפוץ ושימושי, אשר משמש מומחים בתחום הכדורסל למדידת ביצועים באופן שוטף**. הסתכלות על אותם נתונים בפילוחים השונים **איפשר לנו להרחיב את מאגר התכונות** כשלב ראשון, בכדי לאפשר בהמשך בחירת תכונות מגוונות יותר ובעלות השפעה ישירה יותר על ביצועיו של השחקן בפועל.

## 2. נתונים מתקדמים:

**נתונים אלו הינם נתונים נפוצים המשמשים לניתוח הישגים במשחק הכדורסל בעולם האמיתי**, והם מורכבים מחישובים מסובכים המודדים את יעילות השחקן במגוון קטגוריות המשוקללות יחד לכדי מדד כולל. הנתונים יכולים להעיד למשל על מידת תרומתו הכוללת של שחקן לקבוצה, נתון אשר לא ניתן לאמוד מבחינה של נתונים סטטיסטיים יבשים בלבד. להלן כמה מנתונים מתקדמים אלו:

א. **PER (Player Efficiency Rating)** – זהו אחד המדדים הבולטים מתוך הנתונים המתקדמים. מדד זה פותח מתוך ניסיון לתת ערך מספרי יחיד לתרומה הכוללת של שחקן עבור קבוצתו בהתחשב בסגנון המשחק של קבוצה, ובחלקו של השחקן בתוצאות הכוללות של הקבוצה. מדד זה מחושב באופן לא טריוויאלי כלל ומתבסס על מספר רב של קטגוריות, כגון נקודות של השחקן והקבוצה, אסיסטים של השחקן והקבוצה, שלשות, איבודים ועוד.

ב. **TS% (True Shooting Percentage)** – מדד זה בוחן את יעילות הקליעות של השחקן עבור כלל הזריקות שזרק. המדד מתחשב בכמות הנקודות שקלע, קליעות שדה, שלשות וזריקות עונשין ומהווה מעין הערכה כוללת ליכולת הקליעה של השחקן.

ג. **WS (Win Shares)** – מדד זה, אשר נלקח במקור מסטטיסטיקות מעולם הבייסבול, מספק שערך של מספר הניצחונות אשר תרם השחקן לקבוצתו. גם מדד זה מסובך לחישוב ומסתמך על מגוון רחב של קטגוריות המחולקות לתרומות התקפיות והגנתיות של השחקן.

### 3. נתונים כלליים:

בנוסף לנתונים סטטיסטיים המציגים תוצאות של שחקן במשחק עצמו, אספנו **מידע כללי שלהערכתנו יכול לתרום בסיווג שחקן ויכולותיו**. נתונים כלליים אלו כוללים – גיל, משקל, גובה, שנות ניסיון בליגה, עמדות (ממשחק הכדורסל) בהן השחקן משחק, דירוג הקבוצה בה הוא משחק והשכר שהשחקן מרוויח. במבט ראשון נתונים אלו עשויים להיראות בעלי השפעה פחותה על הישגי השחקן אל מול תוצאות קונקרטיות של הישגיו, אך הקשר ביניהם יכול להעיד על איכויות השחקן ולרמוז על מגמה בתוצאותיו לקראת העונה הבאה, לדוגמא –

א. שילוב של גיל ושנות ניסיון של שחקן יכולים להצביע על "השלב" של השחקן בקריירה. ההערכה הרווחת היא כי שחקן NBA מגיע לשיא הפוטנציאל שלו סביב הגילאים 28-32, לכן לפי גיל השחקן ניתן לשערך האם לצפות לעלייה בביצועיו אל מול העונה הקודמת או שמא השחקן בשלהי הקריירה ולצפות לירידה בביצועים. נוסף על כך, שנות הותק של שחקן משפיעות מאוד על ביצועיו ("אין חכם כבעל הניסיון"). מצופה משחקנים חדשים בעלי מעט שנות ניסיון בליגה ובפרט Rookies (שחקנים הנמצאים בשנה הראשונה שלהם) להציג תוצאות פחות טובות ואף לקבל מספר דקות משחק מועט אל מול שחקן בעל ניסיון.

### ב. ישנן שלוש עמדות עיקריות המגדירות את תפקידו של השחקן בקבוצת כדורסל:

i. **Guard** – לרוב שחקנים יותר נמוכים ומהירים בעלי יכולות כדרור, מסירה וקליעה מרחוק.

ii. **Forward** – שחקנים בעמדה זו הם לרוב יותר "ורסטיליים", עם סגנון משחק המשלב בין זריקות ממרחק בינוני וחדירות לסל.

iii. **Center** – שחקנים גבוהים וחזקים, מציגים יותר יכולות הגנתיות וקליעת סלים במרחק קצר מאוד מהסל, כמו כן הם מציגים לרוב יכולות קליעה ירודות אל מול שאר העמדות.

**בהתייחס לעמדה של שחקן יחד עם גובה ומשקל ניתן לקבל מושג על אופי המשחק, החוזקות והחולשות שלו.** יחד עם זאת, ניתן לנסות למצוא קשר בין שחקן בעמדה מסוימת אל מול הבדלי גובה ומשקל מהשחקן הממוצע באותה עמדה, כלומר האם שחקן בעמדת Guard אשר יותר גבוה או בעל משקל גבוה יותר מהווה עבורו יתרון או חיסרון. כמו כן, מידע זה הינו נחוץ לשלב השני של הפרויקט – משחק הדראפט, כאשר נרצה ליישם את הגבלת העמדות הקיימת במשחק האמיתי.

ג. לרוב אנו רואים קשר ישיר בין גובה השכר של שחקן לבין היכולות שלו. כמו בכל תחום, אדם בעל יכולות גבוהות אשר מצליח להביא תוצאות טובות יותר מהשאר יהיה מתוגמל יותר, ואף מעלה אצלו מוטיבציה להמשיך ולהשתפר. **מכאן ניתן לשערך כי שחקן עם שכר גבוה הינו שחקן אשר יציג תוצאות טובות יותר.**

#### 4. נתוני פנטזי:

על מנת לחזות את ביצועיו של שחקן ואת תרומתו לקבוצת פנטזי, **סביר כי ההשפעה של מדדי פנטזי לפיהם דורג בעונות קודמות תהיה מועילה**, על כן אספנו מספר מדדים מרכזיים המפורסמים באתרים שונים ע"י מקצוענים בתחום הספורט הסטטיסטי ומשחק הפנטזי:

א. **Fantasy Rank** – ערך זהו הוא הדירוג המשוערך של שחקן אל מול שאר השחקנים בליגה, בעונת פנטזי נתונה. השחקן אשר סיפק את הערך הגבוה ביותר עם התרומה הגדולה ביותר יקבל דירוג 1, השחקן הבא יקבל דירוג 2 וכן הלאה. חשוב לציין כי בנוסף זהו אחד המדדים שברצוננו לחזות, כיוון שמהווה מעין שקלול כולל של ערך השחקן למשחק הפנטזי אל מול כל השחקנים האחרים.

ב. **Fantasy Value** – מדד זה מאפשר גם הוא שערך תרומה של שחקן בליגת פנטזי. בשונה מהמדד הקודם, מדד זה מציג השוואה יותר מדויקת בין השחקנים כיוון שהערכים הינם ממשיים ונעים בין 1.5 לבין -1.5, כאשר השחקן הטוב ביותר באותה עונה יקבל את הערך החיובי הגבוה ביותר.

### אופן איסוף המידע

1. את מרבית הנתונים היבשים, הסטנדרטיים והמתקדמים, אספנו מהאתר basketball-references.com<sup>iv</sup> המכיל גישה נוחה לסטטיסטיקות המפורטות, מחולקות לכל שחקן לפי עונה והפילוחים השונים שצוינו לעיל (סטטיסטיקות למשחק, לעונה ולפי 36 דקות משחק). שמרנו את המידע בקבצי CSV נפרדים כך שכל קובץ מכיל מידע על השחקנים עבור עונה מסוימת ועבור אחד הפילוחים. בנוסף, עבור עשר העונות הנבחרות, שלפנו לכל קבוצה את המיקום שלה בטבלת הליגה,

על מנת שנוכל למדוד עבור שחקן מסוים האם הוא משחק עבור קבוצה חזקה או חלשה (מהסיבות שצוינו בחלק פירוט הפרמטרים).

2. את נתוני הפנטזי אספנו מהאתר [basketballmonster.com](http://basketballmonster.com)<sup>iii</sup>. אתר זה הינו המוביל ברשת באספקת מידע, הערכות וחיזויים עבור משחק הפנטזי, ומרכז תחתיו שלל ניתוחים סטטיסטיים, כתבות והמלצות. כמעריצי ליגת ה-NBA ושחקני משחק הפנטזי, הזדמן לנו להשתמש בנתונים מאתר זה כדי לבסס את החלטותינו במשחק ולסייע בבחירת השחקן המתאים לקבוצתנו לפי הדירוגים המסופקים באתר. גם במקרה זה חילקנו את המידע לקבצים נפרדים לפי עונות.

3. את הנתונים הכלליים הוצאנו ממאגר שפורסם באתר [Kaggle.com](http://Kaggle.com)<sup>v</sup> המכיל לכל שחקן החל משנת 1950 את גובהו, משקלו, שנות תחילה וסיום הקריירה שלו (אם קיים) בליגה, העמדות בהם הוא משחק ושנת לידה.

4. **חיזוי מומחים** – בנוסף לנתונים הנ"ל שאספנו עבור המערכת הלומדת, השתמשנו גם במידע מאתר [vihashtagbasketball.com](http://vihashtagbasketball.com) לצורך איסוף חיזוי מומחים בתחום הפנטזי. חיזויים אלו כוללים הערכה להישגיהם הסטטיסטיים של השחקנים בכלל הקטגוריות הנמדדות במשחק הפנטזי, בדומה לתוצאה שאנו מעוניינים לספק עם המערכת הלומדת. נתונים אלו נאספו על מנת שנוכל להשוות את תוצאות החיזוי שלנו אל מול תוצאות החיזוי של המומחים, ובכך להעריך בצורה מוסמכת את ביצועי המערכת, כפי שיפורט בהרחבה בפרק על מערכת החיזוי.

## **הרכבת מאגר המידע**

הדוגמאות בהן השתמשנו לצורך אימון המערכת הלומדת הינן שחקנים, המתוארים על פי אוסף התכונות שפירטנו לעיל, עבור עונה ספציפית. לכן, האופן לפיו חילקנו את המידע כך שכל קובץ מידע יתאים לעונה מסוימת, עזר לנו מאוד בבניית דוגמא יחידה של שחקן עם הנתונים שלו באותה עונה.

על מנת לקבל מאגר מלא בו כל רשומה מייצגת שחקן בעונה מסוימת עם הנתונים והסטטיסטיקות השונות הנאספו מאותה עונה, התבצעה הצלבה בין כל טבלאות המידע מהקבצים השונים שצוינו לעיל, על פי המפתח "שם השחקן".

## **אתגרים ופעולות נוספות בתהליך הרכבת מאגר המידע**

1. **הקושי העיקרי שעלה במהלך בניית מאגר הנתונים, הינו התמודדות עם שוני בשמות השחקנים בין הקבצים השונים.** היות והמידע שאספנו נלקח ממקורות שונים, מספר רב של שחקנים הופיעו כאשר שמם נכתב באופן שונה, למשל שחקנים אשר שמם הכיל את המילה JR (Junior) הופיע במספר וריאציות שונות כמו JR., JR, Junior. דוגמא נוספת היו שחקנים אשר שם המשפחה

שלהם מורכב מיותר משם אחד הופרד בחלק מהמקרים במקף ("'-") בין השמות ובחלק המקרים הופרד ברווח כמו השחקן Willie Cauley-Stein. התגברנו על הבדלים אלו ע"י בחירת קובץ ממקור אחד, מעבר ידני על אופן כתיבת השמות במקרים הבעייתיים ושינוי השמות בשאר הקבצים לקבלת קו אחיד בשמות.

2. **קושי נוסף היה אופן ייצוג שמות קבוצות ה-NBA.** בקבצים שונים קבוצות יוצגו לעיתים בשם המלא, לעיתים בראשי תיבות של הקבוצה, וגם במקרה של ראשי תיבות נתקלנו בשוני בין הקבצים. כמו כן, חוסר התאמה בין שמות הקבצים נבע משינויים בשמות קבוצות במהלך השנים בליגת ה-NBA, שינויים כגון שם של קבוצה, העיר אליה היא מיוחסת ואף קבוצות חדשות שנוספו. דוגמאות לקבוצות כנ"ל הן Seattle Supersonics ששינתה גם את שמה וגם את העיר אליה היא מיוחסת אל Oklahoma City Thunder או ה-New Jersey Nets שעברו לעיר Brooklyn. גם במקרה זה בחרנו בייצוג אחיד (מאחד הקבצים) ויצרנו מיפוי של כל ייצוג בעייתי של הקבוצות מהקבצים השונים אל הייצוג האחיד.

3. **קושי מהותי נוסף עלה כתוצאה משחקנים אשר עברו קבוצות במהלך העונה. בליגת ה-NBA ניתן לבצע החלפות שחקנים בין קבוצות במהלך העונה.** כתוצאה מכך, בקבצי הנתונים היבשים לשחקנים אלו ישנה שורה עבור כל קבוצה בהם שיחקו באותה עונה (עם הנתונים והסטטיסטיקות שהשיגו באותה קבוצה). בנוסף, טבלאות אלו הכילו שורה שריכזה וסכמה את כל הנתונים על פני כלל הקבוצות, כאשר שם הקבוצה עבור שורה זו הייתה TOT (Total). עלתה התלבטות לגבי אופן הייצוג של שחקנים אלו במאגר שלנו, האם להשאיר את כל השורות של כל הקבוצות כדוגמאות (ובעצם יהיו מספר שורות במאגר עבור אותו שחקן) או שמא להשאיר רק את שורת ה-TOT של השחקן. בחרנו באפשרות השנייה שהינה פשוטה יותר, ובנוסף אנו מעריכים שמספקת ייצוג מדויק יותר לביצועיו וליכולותיו של השחקן בעונה הנתונה.

4. **על מנת לקבל את התכונות של גיל ושנות ניסיון בליגה נעזרנו בנתונים של תאריך לידה ושנת כניסה לליגה עבור שחקן בעונה מסוימת.** עבור כל עונה חישבנו את ההפרש משנת העונה לבין שנות הלידה והכניסה לליגה כדי לקבל את הערך המדויק לאותה עונה.

5. **הרכבת התכונה של דירוג הקבוצה התבצע בעזרת הצלבה פשוטה בין הקבוצה עבורה השחקן שיחק מאותה עונה לבין הקבצים המכילים לכל עונה את דירוגי הקבוצות.**

# חלק 1: חיזוי סטטיסטיקה לשחקני NBA

## תיאור פתרון הבעיה

בפרק זה נתאר את הפתרון לבעיית חיזוי הסטטיסטיקה של שחקני ה-NBA, כפי שהוגדרה בפרק הקודם. נזכיר כי אנו מעוניינים לחזות תשעה נתונים סטטיסטיים לכל אחד משחקני ה-NBA המשחקים בעונה הבאה (העתידית), המשמשים לצורך קביעת תוצאות משחק הפנטזי שאותו הגדרנו. בנוסף, אחת ממטרות הפרויקט הינה לחזות את דירוג הפנטזי הכולל של כל אחד מהשחקנים בעונה העוקבת, חיזוי זה ישמש, בין היתר, גם את סוכן משחק הדראפט בחלק השני של הפרויקט.

**החיזוי בכל אחת מהקטגוריות מתבצע באמצעות מערכת לומדת**, שפיתוחה מבוסס אימון על מאגר דוגמאות. מטרת חלק זה של הפרויקט הינו מימוש מספר מערכות לומדות (לכל אחת מקטגוריות החיזוי), תוך שימוש בכלים שונים של עיבוד מידע ואופטימיזציה התכונות והמערכת הלומדת לצורך שיפור תוצאות החיזוי.

## סידור והכנת המידע

בפרק העוסק במאגרי הנתונים בהם השתמשנו, תיארונו את המידע הגולמי אותו כרינו מהאתרים השונים, ואת התכונות השונות שאספנו על מגוון סוגיהן. כמו כן הסברנו מה החשיבות של התכונות ומדוע לדעתנו מידע על תכונות מיוחדות יכול לעזור לנו לחזות ביתר דיוק את הנתונים שאנו מעוניינים לחזות. בפרק זה נסביר אילו פעולות עיבוד ביצענו על המידע הגולמי, על מנת להתאים אותו לאלגוריתמי הרגרסיה בהם השתמשנו לצורך ביצוע הלמידה.

## הגדרת אובייקט השחקן

לאחר שלב כריית המידע ושליפת הנתונים הרלוונטיים, עלה הצורך לאחד את המידע מהקבצים השונים, בכדי שנוכל להגדיר כיצד ייראה אובייקט השחקן שאת נתוניו נרצה לחזות.

בשלב הראשון **נבנתה טבלה אחידה עבור כל עונת NBA בין השנים 2010-2019**, שתכיל עבור כל שחקן ששיחק בליגה באותה עונה את כלל המידע עבור שחקן נתון ששיחק באותה עונה. התקבלו עשר טבלאות המכילות בין 316 ל-458 שורות (בהתאם למספר השחקנים הפעילים בעונה). עבור כל שחקן מופיעות בטבלה בדיוק 101 עמודות המתארות את התכונות של השחקנים (כאשר העמודה הראשונה היא שם השחקן, אשר מהווה מזהה ייחודי, ולא תשמש כקלט לתהליך הלמידה).

לאחר מכן, **נדרשנו להגדיר את אופן הייצוג של אובייקט השחקן** – לצורך למידה וחיזוי התוצאות של העונה הבאה יש להגדיר באופן מדויק מהן התכונות המאפיינות שחקן NBA עבור המערכת. זוהי בעיה מורכבת, כיוון שנתונים מסוימים על שחקן NBA משמשים אותנו גם לצורך למידה, בתור ערך אמת של עונת עבר, אך גם בתור תכונות המשמשות לאפיון אובייקט השחקן עבור העונה הנוכחית. לדוגמא, אם

נסתכל על שחקן ששיחק בעונות 2018 ו-2019, כמות הנקודות למשחק שקלע השחקן בעונת 2019 הינה תכונה של אובייקט השחקן כחלק ממאגר המידע של עונת 2019, אך גם תוצאת השחקן בקטגוריה זו עבור אובייקט השחקן ב-2018. נוסף לכך, ישנה מורכבות הנובעת מהעובדה שנתונים שונים נדגמו בנקודות זמן שונות (עונות שונות), אולם משחק הכדורסל הוא דינמי ועובר שינויים רבים בסגנון ובאופי המשחק לאורך השנים.

ישנן מספר אפשרויות להגדרת האובייקט, אשר הכרעה ביניהן הינה קריטית בהשפעתה על תהליך הלמידה והחיזוי שיגיעו בהמשך. להלן פירוט דרכי הפעולה האפשריות ופירוט הדרך שנבחרה:

### 1. אובייקט יורכב מנתוני שחקן NBA בעונה אחת בלבד, והלמידה תתבצע על סמך העונה הקודמת

#### **בלבד לעונה אותה אנו מעוניינים לחזות:**

באפשרות זו אנחנו מניחים שיש הבדל גדול בין עונות NBA שונות, ולכן בשביל לחזות את ממוצע הנקודות למשחק של עונת 2019, אנחנו יכולים ללמוד רק לפי המידע הכי עדכני, כלומר לפי נתוני עונת 2017 ותוצאות האמת שהיו בעונת 2018.

יש לשים לב כי בשיטה זו, עבור חיזוי לעונת 2019, התעלמנו לחלוטין מהמידע על עונת 2016. מספר דוגמאות הלמידה שלנו הוא כמספר השחקנים ששיחקו גם בעונת 2017 וגם בעונת 2018 (inner join) המפתחות בכל אחת מטבלאות השחקנים), מכיוון שאנו מאמנים את המערכת לפי תכונות מעונה ספציפית ומתאימים את קטגוריית החיזוי לפי הערך שהתקבל בעונה העוקבת. כלומר, אם היה קיים שחקן ששיחק בשנת 2017 בלבד, לא היינו יכולים לעשות דבר עם הנתונים שלו כיוון שאין לנו תיוג (תוצאת אמת) שלו מעונת 2018 להזין למערכת בתהליך הלמידה.

### 2. אובייקט יורכב מנתוני שחקן NBA בעונה אחת בלבד, אך הלמידה תסתמך על אובייקטים

#### **ממספר כלשהו של עונות רצופות הקודמות לעונה אותה אנו מעוניינים לחזות:**

באפשרות זו אנו מניחים שלא קיים הבדל גדול בין עונות שונות של ליגת ה-NBA, ולכן על מנת לחזות את ממוצע הנקודות למשחק של עונת 2019, נאמן את המערכת על אובייקטים מכלל העונות במאגר (2010-2019) כאשר בכל עונה נגדיר את השחקנים אשר שיחקו באותה עונה, ותוצאת "החיזוי" (עבור אימון המערכת) תהיה ההישג האמיתי שלהם מהעונה העוקבת.

יש לשים לב כי הקלט למערכת החיזוי הסופית שלנו לא השתנה לעומת דרך הפעולה הראשונה. עם זאת, **הגדלנו משמעותית את מספר הדוגמאות**, כאשר לצורך דוגמא יש צורך בשחקן ששיחק בשתי עונות עוקבות כלשהן (הראשונה לצורך איסוף התכונות והשנייה לצורך איסוף ערך הקטגוריה הנחזית), הרלוונטיות לדעתנו לצורך יצירת מודל ששיג תוצאות טובות בחיזוי הנתונים של 2019.

### 3. אובייקט יורכב מנתוני שחקן NBA על פני יותר מעונה אחת (מספר שרירותי כלשהו של עונות

#### **עוקבות), כאשר לכל תכונה של השחקן יתווסף תיוג שיציין לאיזו עונה התכונה שייכת:**

באפשרות זו מנצלים יכולת של חלק ממערכות הלמידה העובדות בשיטת רגרסיה לזהות קשרים בין תכונות זהות בשנים שונות, וניתן משקל רב יותר להתמדה ולביצועים המתמשכים של שחקן על פני מספר עונות וכן למגמת השינוי שלו, ולא רק לביצועים העדכניים ביותר שלו. מספר הדוגמאות שיתקבל

הינו כמספר השחקנים ששיחקו בכל העונות מהן מורכב אובייקט השחקן, כלומר, אם אובייקט שחקן מורכב מנתוני שלוש עונות רצופות, בכדי לחזות את ערכו של שחקן בעונת 2019, נצטרך את נתוניו בעונות 2016-2018.

**בדרך פעולה זו ניתן להרחיב משמעותית את אוסף התכונות המאפיין אובייקט**, כך שיתקבלו מספר סדרות של נתונים מאותו סוג לגביו עבור מספר קודמות לעונה שממנה מוציאים את הקטגוריה שנרצה לחזות (למשל, נקודות למשחק מעונת 2016, 2017 ו-2018, כל נתון כנ"ל ייחשב כתכונה נפרדת). דרך זו נותנת חשיבות רבה יותר לתהליך ששחקן NBA עובר על פני מספר עונות, וכן יכולה להסיק מסקנות ביניים בזמן הלמידה על מגמות השינוי שלו במשך הזמן. **מתודולוגיה זו נפוצה בפתרון בעיות בהן מנסים לבצע אקסטרפולציה של ערכים עתידיים מתוך נתונים כרונולוגיים.**

החיסרון הבולט של שיטה זו היא בצמצום של כמות הדוגמאות ככל שמנסים לקחת יותר עונות לייצוג אובייקט. חסרונות נוספים שהינם עמוקים יותר וקשורים לדינמיות של ליגת ה-NBA וכן למורכבות איסוף נתוני ספורטאים בעולם האמיתי, הם שישנם מקרים בהם שחקנים מקצועיים משחקים בליגת ה-NBA למשך עונה בלבד, או אפילו משחקים למשך עונה, עוברים לשחק בחו"ל בעונה העוקבת, ואז חוזרים ל-NBA בעונה שלאחר מכן. בנוסף לכך, ספורטאים מקצועיים נוטים להיפצע, ובעקבות כך להחמיץ לעיתים עונות שלמות או את מרבית המשחקים בעונה מסוימת, עובדה שעלולה ליצור "חורים" משמעותיים בטבלת הנתונים הנוצרת כתוצאה מנקיטה בדרך פעולה זו, או אפילו נתונים חריגים או מעוותים על שחקן כלשהו בעונה מסוימת בה הוא ישב פצוע על הספסל ברוב המשחקים.

לבסוף, **הוחלט לבחור בדרך הפעולה השנייה**. להלן הפירוט היתרונות והחסרונות הקיימים בשיטה זו לעומת שאר דרכי הפעולה שהצענו:

#### 1. מספר הדוגמאות ללמידה

לדרך הפעולה הנבחרת ישנו יתרון בולט על פני הדרכים האחרות. בשיטה זו מתקבל מספר רב יותר של דוגמאות עבור שלב האימון בתהליך הלמידה.

#### 2. הבדלים משמעותיים בין עונות שונות

דרך הפעולה הראשונה ממזערת את ההטיה הנובעת מהבדלים בין עונות ומשינוי באופן המשחק לאורך השנים, אך עם זאת מתעלמת מנתונים רבים מעונות קודמות. להערכתנו זהו מהלך קיצוני מידי ביחס לקצב השינוי באופי המשחק. דרך הפעולה השלישית מוטה יותר מדרך הפעולה שבחרנו מכיוון שעבורה המערכת החוזה עלולה לקלוט מגמות היסטוריות מעונות עבר בקלות יתרה מידי, ולנסות להשליך אותן על עונות מאוחרות יותר. בתור "סוכנים אנושיים" במשחק הפנטזי בעלי ניסיון והיכרות מוקדמים עם עולם ה-NBA, אנו משערים שניסיון כזה יפגע ביכולות החיזוי, משום שלמגמות שהיו רלוונטיות למשל לפני עשור, כבר אין משמעות במשחק בימינו.

#### 3. עיקרון התער של אוקאם והשאיפה להשתמש במספר תכונות מצומצם

בשלב זה יש לקחת בחשבון את השלכות בחירת שיטת הפעולה על המשך הפרויקט. לפי עיקרון התער של אוקאם (שנלמד בקורס המבוא), כאשר ישנם מספר הסברים שקולים לבעיה יש להעדיף את ההסבר



הפשוט יותר. תוצאות אמפיריות רבות תומכות בעיקרון זה. במקרה שלנו כדי ליישם את העיקרון, נשאף ליצור מערכת לומדת המגיעה לביצועים טובים תוך שימוש במספר תכונות קטן. לכן שיטת הפעולה הנבחרת עדיפה על השיטה השלישית, הדורשת מספר תכונות רב.

#### 4. ניצול מירבי של המידע

לפי דרך הפעולה הראשונה, המידע הרלוונטי לחיזוי הישגי שחקן בעונה מסויימת מגיע רק משתי העונות שקדמו לה, ולכן אפילו אם נאסף מידע על עשר עונות, היא מנצלת נתונים רק על שתיים מהן. באופן ישיר נגרמת הקטנה של מספר הדוגמאות הנלמדות. בדרך הפעולה השלישית יש להתעלם משחקנים שלא שיחקו את מספר עונות הנבחר באופן רצוף, כלומר נתעלם מכל הנתונים על שחקן ששיחק שתי עונות שלמות אך לא ברצף. החסרון במקרים אלו הוא לא רק שמספר הדוגמאות קטן יותר, אלא גם שאין ניצול מלא של מידע שיכול להיות שימושי. מבחינה זו, דרך הפעולה השניה מאפשרת לנצל באופן מיטבי את המידע שנאסף.

**כאמור, לבסוף החלטנו שייצוג אובייקט יורכב מנתוני שחקן NBA בעונה אחת בלבד.** עם זאת, בחרנו לאמן את המערכת על אובייקטים מכלל העונות שאספנו, ללא יצירת הבדל ביניהם עבור אלגוריתם הלמידה, בהתאם לשיקולים שפירטנו לעיל. בהתאם לכך התקבלה טבלה אחת גדולה המכילה כ-3,600 שורות, כאשר כל שורה מייצגת שחקן NBA בעונה ספציפית, ולכל שחקן יש 100 עמודות המייצגות תכונות המאפיינות אותו ועמודה נוספת עם שם השחקן.

לצורך המחשה, ישנם שחקנים ששם מופיע רק באחת השורות מפני שהם שיחקו בשתי עונות עוקבות בלבד (לדוגמא Amare Stoudemire ששיחק בעונות 2010-2011). אחד השחקנים ששמו מופיע במספר המרבי של שורות הוא למשל LeBron James, המופיע ב-10 שורות שונות בטבלה עבור כל עונה רלוונטית בה הוא שיחק.

#### חלוקת המידע לסטים של Train/Validation/Test

כפי שלמדנו בקורס המבוא, **על מנת לקבל תוצאה אמינה של יכולת החיזוי של המערכת שלנו, לא ניתן לבחון את ביצועי המערכת על אותו המידע עליו אומנה**, מכיוון שהמידע האמיתי עליו נרצה להפעיל את המערכת בעתיד יהיה שונה ויכלול דוגמאות אחרות מאלו שבעזרתן התבצע האימון. כדי שיהיה ניתן לבחון את יכולת ההכללה (Generalization Error)<sup>1</sup> של המערכת, **נשמור תת קבוצה של קבוצת הדוגמאות, שתשמש כסט מבחן לבדיקת הביצועים לאחר ביצוע תהליך הלמידה.** בנוסף לכך, נרצה קבוצה קטנה של דוגמאות שתשמש כסט של וולידציה. סט זה ישמש אותנו בעת כוונן היפר-פרמטרים של אלגוריתמי הלמידה, כאשר לאחר בחירת היפר-פרמטר אופטימלי (באופן שיתואר בהמשך), נוודא כי הוא באמת נותן את הביצועים המצופים גם על סט הוולידציה.

<sup>1</sup> הכללה (Generalization Error) – מתאר את מידת הדיוק של האלגוריתם בחיזוי ערכים על data לא מוכר.

יש לציין כבר בנקודה זו שסט המבחן שימש אותנו לבסוף רק לצורך וידוא סופי של המודל, אך כל הנתונים שיופיעו בהמשך בתיאור הניסויים שביצענו בחלק החיזוי של הפרויקט אינם הביצועים על סט המבחן, כי אם על כלל הדוגמאות, כאשר כל הניסויים נערכו באמצעות מודל הערכת ביצועים של Cross-Validation. פירוט על השימוש שלנו במודל זה מופיע בפרק מתודולוגיית הניסויים.

בבעיה הנתונה, עבור בחירה של העונות שנתוניהן משמשים לצורך הלמידה, מתקבלת קבוצה סופית בעלת גודל קבוע של דוגמאות בהן ניתן להשתמש. יצירת דוגמאות נוספות אפשרית רק על ידי הרחבת מספר העונות שנלקחות בחשבון.

**חלוקת הדוגמאות לסטים השונים התבצעה באמצעות הגרלה רנדומלית כאשר בממוצע תהליך זה מבטיח כי ההתפלגות בכל אחד מהסטים תהיה זהה.** לרוב בבעיות סיווג (כולל בעיות multi-class<sup>2</sup>), נהוג לבצע את חלוקת הדוגמאות לסטים השונים בצורה הנקראת Stratification<sup>3</sup>, כך שיחס הדוגמאות מכל תיוג בכל אחד מהסטים השונים יהיה זהה, אך מכיוון שהבעיה איתה אנו מתמודדים היא בעיית רגרסיה, כבר בשלב הראשון ישנו קושי בביצוע חלוקה של המידע לקבוצות עבורן ההתפלגות הדוגמאות תהיה זהה. על מנת לא ליצור הטיה בין היכולת שלנו לחזות את הקטגוריות השונות, אנו מעוניינים להשתמש בדיוק באותן דוגמאות אימון עבור כל תכונה לצורך למידה, ובאותן דוגמאות מבחן לצורך בדיקת ביצועים, ולכן שיטה זו אינה מתאימה.

ככל שאחוז הדוגמאות הנשמרות בצד עבור סט המבחן גדול יותר – כך מקבלים מדידה אמינה ומדויקת יותר של ביצועי המערכת לאחר שלב הלמידה (לפי מדדי השגיאה), אך עם זאת מאבדים דוגמאות אימון, כלומר המודל הסופי שמתקבל אינו מאומן בצורה המיטבית שניתן להגיע אליה. בכיוון השני, כאשר מקטינים את אחוז הדוגמאות של סט המבחן – מקבלים מודל מאומן בעל שונות (Variance) נמוכה יותר, אך מאבדים מדיוק מדידת הביצועים הסופית.

**הוחלט לתת דגש רב יותר לאימון המיטבי של המודל שלנו, ופחות לקבלת דיוק מרבי של מדדי השגיאה, ולכן חולקו הדוגמאות בשני שלבים באופן הבא:**

1. בשלב הראשון מגרילים 10% מתוך כלל הדוגמאות ושומרים אותן בצד. דוגמאות אלו יהיו סט המבחן.
2. הדוגמאות שנותרו לאחר הפרשת סט המבחן הן 90% מכלל הדוגמאות. מתוך קבוצה זו מגרילים שוב 10% שישמשו בתור סט וולידציה. כלומר, סט הוולידציה מורכב מ-9% מכלל הדוגמאות שבידינו, וסט האימון מורכב מ-81% מהדוגמאות.

### **טרנספורמציה של תכונות קטגוריאליות**

מתוך כלל התכונות (Features) של אובייקט השחקן, קיימות שתי תכונות בלבד שאינן מבוטאות בערכים מספריים. תכונות אלה הן שם הקבוצה של השחקן (Team), והעמדה בה השחקן משחק (Position), או בקיצור

---

<sup>2</sup> Multi-Class – בעיות סיווג לא בינאריות – בעלות יותר משתי מחלקות סיווג.

<sup>3</sup> Stratification – שיטה לדגימת האובייקטים כך שתתקבל חלוקה המכילה יחס של אובייקטים מכל סיווג בכל קבוצה, הזהה ליחס המופיע בכלל המדגם.

**(Pos).** אלגוריתמי הלמידה בהם השתמשנו לצורך קבלת החיזוי דורשים תכונות מערכים נומריים בלבד, ולכן בשלב זה נדרשנו לבצע המרה של התכונות הקטגוריאליות הללו לתכונות מספריות.

ישנן שתי דרכים מקובלות להתמודד עם תכונות קטגוריאליות. הדרך הראשונה הינה **טרנספורמציה של כל קטגוריה לתכונה בפני עצמה בשיטת One-Hot**. כלומר, כל ערך קטגוריאל של התכונה המקורית הופך לתכונה בינארית שיכולה לקבל 0 או 1 בהתאם לשייכות של האובייקט לקטגוריה מסוימת. חסרון של המרה לתכונות One-Hot הינו שתכונה אחת במקור שמכילה הרבה קטגוריות יוצרת מספר תכונות רב (כמספר הקטגוריות), ויש לקחת זאת בחשבון.

הדרך השנייה היא פשוטה יותר ובמסגרתה **ממספרים את הקטגוריות השונות ונותנים להן ערכים עוקבים** (למשל 0,1,2,3...). החיסרון של שיטה זו הוא שהיא מייצרת סדר מדומה בין הקטגוריות השונות. אם נשליך את השיטה על תכונת שם הקבוצה, כל שם קבוצה ימופה אל "גודל" כלשהו, כאשר אין בהכרח הגיון בכך שקבוצה B תהיה "קטנה" מקבוצה C אך "גדולה" מקבוצה A.

כפי שהוסבר בהקדמה, בליגת ה-NBA משתתפות 30 קבוצות שונות, ולכן לתכונה Team יש 30 ערכים אפשריים שונים. מכיוון שמדובר במספר רב של ערכים אפשריים לא רצינו למפות אותה לתכונות One-Hot. מצד שני, לא רצינו ליצור סדר אקראי בין הקבוצות השונות. משום כך, **החלטנו להחליף את השם של כל קבוצה בדירוג היחסי של הקבוצה באותה עונה שאליו שייך האובייקט** (לפי הדירוג הסופי של עונת ה-NBA), **וכך נפטרנו מהתכונה הקטגוריאלית תוך יצירת סדר בעל משמעות אמיתית בין הקבוצות השונות.**

עבור קטגוריות עמדות המשחק (Pos), בפועל קיימות 5 עמדות משחק בלבד (PG, SG, SF, PF, C). אף על פי כן, במידע שאספנו חלק מהשחקנים שויכו ליותר מעמדה אחת, כך שקיבלנו בסופו של דבר 14 ערכים אפשריים (הכוללים את העמדות המקוריות וחלק מהשילובים ביניהן, כגון SF-PF). יש לציין כי מספר זוגות השילובים ההגיוניים של עמדות ששחקן יכול לשחק בהן הוא קטן יחסית, למשל לא יתכן שחקן שיכול לשחק בעמדה C וגם בעמדה SG. **מכיוון שמדובר במספר יחסית מועט של תכונות נוספות, וכיוון שלא היינו מעוניינים ליצור סדר בין העמדות, החלטנו להפוך את התכונה Pos ל-14 תכונות מסוג One-Hot**, כאשר עבור שחקן נתון רק אחת מהן בדיוק תהיה בעלת הערך 1, וכל השאר יהיו בעלות הערך 0. לאחר שלב זה כל התכונות בטבלת האובייקטים שלנו מכילות ערכים נומריים בלבד.

### טיפול בדוגמאות חריגות וערכים חריגים (Outliers)

ערכים חריגים הינם ערכים אשר רחוקים מהערך הממוצע או הסביר עבור קטגוריה ספציפית, או שאינם נמצאים בטווח הגיוני (למשל ערכים שליליים עבור קטגוריה חיובית). ערכים אלו עשויים לייצר רעש בתהליך הלמידה וכן להקשות על ביצוע עיבוד מידע כנדרש, כגון בשלב ה-scaling.

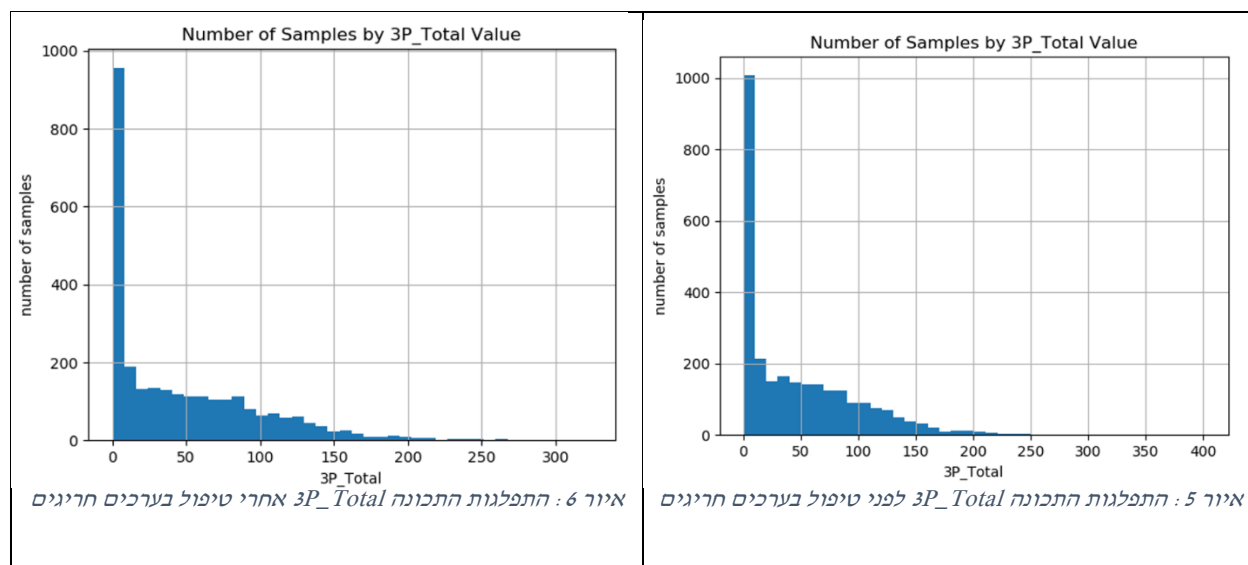
ישנן מגוון דרכי טיפול בערכים חריגים וביניהם שינוי ערכי החריגות או הסרה מוחלטת של אובייקטים שמכילים ערכים שמוגדרים כחריגים. **להלן פירוט השינויים שערכנו בטבלת האובייקטים עקב דוגמאות חריגות או ערכים ספציפיים חריגים:**

1. **השמטת שחקנים ששיחקו פחות מעשרה משחקים בעונה – כל שחקן ששיחק עשרה משחקים או פחות בעונה הנתונה הוסר מהטבלה.** הסיבה לכך היא ששחקנים אלו יוצרים עיוותים סטטיסטיים שעלולים לבלבל את המערכת הלומדת שלנו, ועם זאת, אינם מהווים דוגמא קריטית שנרצה שהמערכת תדע לסווג (שחקנים ששיחקו מספר מועט של משחקים לרוב אינם שחקנים משמעותיים ולא ייבחרו לקבוצת הפנטזי). לדוגמא, ייתכן שחקן גרוע באופן יחסי ששיחק רק במשחק אחד בעונה, והצליח (באופן נדיר) לחסום 5 זריקות של הקבוצה היריבה. במצב כזה, נתון ה-BLK שלו, מספר החסימות למשחק, יהיה 5 בממוצע למשחק, וזה גבוה מאוד. מצד שני, ייתכן שכל שאר נתוניו נמוכים במיוחד, וזה מאוד לא אופייני ועלול לגרום להטיה של המערכת (לצורך העניין, ידוע כי ישנה קורלציה חיובית כלשהי בין נתון החסימות BLK לבין נתון הכדורים החוזרים (TRB)).  
בהתאם למשפט הגבול המרכזי (הסתברות), השונות הקיימת בביצועים הממוצעים של שחקן על פני מספר משחקים יורדת ככל שמספר המשחקים גבוה יותר, ולכן היא אינה מהווה בעיה כאשר שחקן שיחק למשל 80 משחקים. מסיבה זו החלטנו לסמן כל שחקן ששיחק עשרה משחקים או פחות בתור דוגמא חריגה (Outlier) ולהסיר אותו מהטבלה.  
ניתן לטעון כי לאחר הלמידה, כאשר המערכת שלנו תתבקש לבצע הערכה של שחקנים עבור עונה עתידית, ייתכן כי נרצה להעריך שחקן ששיחק רק עשרה משחקים בעונה הקודמת. אולם, **ברוב המקרים יהיה זה שחקן פחות איכותי (ולכן קיבל פחות זמן משחק, אלא אם מדובר בשחקן שנפצע) שבאופן ישיר תרומתו למשחק הפנטזי נמוכה.** בהמשך כאשר נפרט את שיטות הערכת הביצועים שלנו, נסביר כי זהו שחקן שמראש מיוחסת חשיבות פחותה לדיוק הסיווג שיוחזר לו על ידי המערכת.  
הורדת הדוגמאות החריגות הנ"ל תורמת לשיפור היכולת להעריך את נתוני השחקנים המשמעותיים יותר (שחקנים שיוצאים בדירוג גבוה במדד הפנטזי – לרוב משתתפים במספר משחקים גבוה במהלך העונה).
2. **הסרה חלקית של אובייקטים עם ערכים חסרים –** בטבלה הסופית המכילה את כלל אובייקטי השחקנים שהרכבנו, **עלו כ-30 שחקנים עבורם חסרים עשרה ערכים או יותר המופיעים כ-NaN** (עקב מגבלות זמינות נתונים מסוימים בשלב איסוף המידע). למרות שאנו משתמשים בשיטה יעילה יחסית להשלמת ערכים חסרים (כפי שיפורט בהמשך), היעילות שלה גדולה יותר ככל שמספר הערכים החסרים באובייקט קטן יותר. בנוסף, ראינו כי יש חפיפה גבוהה בין 30 השחקנים הנ"ל לבין שחקנים ששיחקו עשרה משחקים או פחות. לכן, הוחלט לסמן גם את האובייקטים בעלי עשרה ערכים חסרים (או יותר) כחריגים ולהשמיטם. לאחר השמטת השחקנים עם מספר משחקים נמוך מהסף שנקבע או מספר ערכים חסרים גבוה מהסף, מספר הדוגמאות ירד בכ-150 דוגמאות.
3. **טיפול בערכים חריגים –** נוסף לדוגמאות שלמות שהוגדרו כחריגות, המידע כלל ערכים ספציפיים בקטגוריות מסוימות הנחשבים לחריגים, אשר דרשו תיקון בהתאם. בשלב זה התבצע מעבר ידני על ההיסטוגרמה של כל אחת מהתכונות, במטרה לזהות האם ישנם ערכים חריגים של התכונה שעלולים להפריע לתהליך הנרמול (Scaling) ועל ידי כך לפגוע בטיב הלמידה שיבצע המודל שלנו. הרעיון הוא שאם רוב הערכים מקובצים בתחום קטן, אך יש ערכים קיצוניים בודדים שרחוקים מהמסה העיקרית,

אז לאחר נרמול, הערכים הסבירים יהיו בעלי הבדל זניח זה מזה, אך בהפרש גדול מהערכים הקיצוניים. דבר זה מוריד את השונות של התכונה, מה שמקשה על ההפרדה ביניהם לפיה וגורם להפחתה מחשיבותה.

**שלב זה בוצע באמצעות ברירת הערכים הקיצוניים עבור תכונות בהן ראינו שיש להם השפעה גדולה.** בחרנו אחוז כלשהו או מספר קבוע של הדוגמאות בעלות הערכים הקיצוניים (למשל - ה-0.5% הכי טובים, לרוב מדובר במספר בודד של דוגמאות) וקבענו את ערך התכונה שלהן להיות הערך של הדוגמא הכי קיצונית שבאה לפנייהם (במקרה של בחירת 0.5% הכי קיצוניים, ביצענו השמה של ערך האחוזון ה-99.5%). באופן זה השארנו את האובייקטים הקיצוניים להיות עדיין "הכי טובים" (או "הכי גרועים", במקרים של קיצון לכיוון השלילי), אך יתאפשר לבצע נרמול תוך קבלת שונות גבוהה יותר בין הערכים המנורמלים.

**להמחשה נתבונן בדוגמא הבאה של טיפול בחריגות עבור התכונה 3P\_Total (כמות השלשות הכוללת לעונה):**



בשני הגרפים מוצגות היסטוגרמות של מספר האובייקטים לפי ערך התכונה 3P\_Total עם 40 תאים (bins). בגרף הימני (עם הערכים החריגים), בצד הערכים הגבוהים, ישנן דוגמאות קיצוניות בודדות (למעשה בלתי נראות בגרף). ערכים אלו קובעים את הסקאלה של ציר ה-X להיות בין 0 ל-400 (בקירוב). לאחר קירוב הערכים החריגים ביותר אל המרכז, ע"י השמת הערכים הבאים שלא הוחשבו כחריגים, ניתן לראות בגרף השמאלי כי הסקאלה של ציר ה-X נעה בין 0 ל-300 (בקירוב). יש להתעלם מהשינוי בציר Y שקרה עקב שימוש במספר קבוע של תאים בגודל אחיד (שחלקם היו ריקים לפני הטיפול). השינוי אמנם מזערי, אך כאשר נבצע נרמול לטווחי הערכים בכל התכונות, פעולה זאת תגדיל משמעותית את השונות (ולכן את החשיבות לתהליך הלמידה) של תכונה זו.

## ביצוע נרמול (Scaling)

על מנת לא ליצור סטייה בחשיבות של תכונות שונות עבור אלגוריתמי הלמידה (בעיקר אלו שמבצעים הפרדות ליניאריות, אבל גם עבור KNN למשל), וכן על מנת ליצור אחידות במשקל התכונות עבור השלמת ערכים חסרים, עלה הצורך בנירמול הערכים של כל התכונות לטווח אחיד.

בדרך כלל כאשר מבצעים נרמול ישנן שתי אפשרויות עיקריות (שתיהן מהוות טרנספורמציות ליניאריות):

1. **נרמול Min-Max** – קביעת ערך מקסימום ומינימום חדשים עבור התכונה. הסטה של כל ערך בהתאם אל מיקומו בטווח החדש.

2. **נרמול Z-Score** – ביצוע טרנספורמציה לכל הערכים כך שהתוחלת (הממוצע) והשונות יהיו 0 ו-1 בהתאמה, כלומר תואמות להתפלגות נורמלית.

ניתן לבחור לכל תכונה שיטת נרמול שונה, אך אז עלול להיווצר הבדל בטווחים של התכונות השונות. אנו **בחרנו לבצע לכל התכונות נרמול מסוג Min-Max**, כאשר טווח הערכים החדש הוא  $[0,1]$ . כלומר כל הערכים עברו את הטרנספורמציה הבאה:

$$x_{new} = \frac{x - \min X}{\max X - \min X}$$

אמנם הבחנו במספר תכונות של אובייקט המתפלגות נורמלית, אך בחרנו לא לנרמל אותן על פי Z-score, תוך העדפה לקבוע טווח אחיד לכל הערכים על מנת ליצור איזון בין התכונות השונות (בנרמול Z-score הטווח באופן תיאורטי הוא ממינוס אינסוף עד אינסוף). הסיבה לכך היא **שבהינתן שלא קיים ידע מוקדם לגבי אילו תכונות יהיו חשובות לחיזוי כל אחד מהנתונים אותם אנו מעוניינים לחזות, אנו רוצים לשמור על אחידות במשקלים של כל תכונה ולא ליצור הטיה מובנית לאלגוריתמי הלמידה**. נביא דוגמא בה נתקלנו בתרגילי הבית בקורס המבוא. אם יש לנו שתי תכונות שהטווח של אחת מהן בין 0 ל-1, והשניה בין 0 ל-100, אז אלגוריתם כמו KNN שמסווג בעזרת מרחק בין אובייקטים, יושפע בעיקר מההפרשים של התכונה השניה, וכמעט לא יושפע מהראשונה.

## השלמת ערכים חסרים

התבצע חישוב גורף של כמות השורות בעלות ערכים חסרים (NaN) בכל תכונה. **נמצאו כ-14 תכונות בהן הופיעו ערכים חסרים**. המספר הגבוה ביותר של ערכים חסרים התקבל עבור התכונה 3P% (הסיבה לכך היא שישנם שחקנים אשר לא זרקו כלל זריקות מ-3 נקודות, ולכן לא הופיע עבורם נתון זה). תכונות נוספות בהן עלו חוסרים עקב זמינות המידע במאגרים בהם השתמשנו הינן למשל משכורת (Salary) ושם הקבוצה (Team Name). קיימת חפיפה בין קיום של ערך NaN בתכונות שונות באותה שורה, כלומר, שחקן כלשהו שבמספר מאגרי מידע לא נמצאו הנתונים עבורו. למשל גובה ומשקל של השחקן (Height ו-Weight בהתאמה) הגיעו מאותו המאגר, והם חסרים בדיוק באותן דוגמאות.

בשלב הראשון, הושמטו כלל השורות עם חוסרים בלפחות אחת מהתכונות הבאות - 'Fantasy Value Total', 'Fantasy Rank Total', 'Fantasy Value Per Game', 'Fantasy Rank Per Game', זאת כיוון שתכונות אלו הינן בעלות השפעה גבוהה על תוצאת החיזוי (על פי הערכה מוקדמת), וכן כיוון שאת חלקן אנחנו רוצים לחזות, ולכן הן נדרשות לצורך אימון המודל.

לאחר מכן נשקלו אלגוריתמים שונים להשלמת מידע חסר. אפשרות אחת הינה השמת הערך הממוצע או החציוני של התכונה עבור דוגמאות בהן הוא חסר, אך פעולה זו פוגעת בהתפלגות המקורית של המידע (למשל בהשלמה לפי הממוצע, התוחלת לא משתנה אך השונות קטנה). בנוסף, קיים אלגוריתם הנקרא **Closest Fit** לפיו, הערכים החסרים מושלמים ע"י השוואת הערך לזה של האובייקט הקרוב ביותר. אלגוריתם זה פועל באופן הבא – בהינתן דוגמא עם ערכים חסרים, נמצא את הדוגמא הכי קרובה אליה לפי מרחק אוקלידי של יתר התכונות, ונבצע השמה של הערכים החסרים על סמך ערכי האובייקט הכי קרוב שמצאנו. חשוב מאוד לציין כי אלגוריתם זה, בדומה לאלגוריתם KNN, מושפע מהטווחים של התכונות השונות, ולכן הקפדנו להשתמש בו לאחר ביצוע **Scaling**, כך הבטחנו שהחשיבות של כל התכונות תהיה אחידה בעת מציאת השכן הקרוב ביותר. מבחינה אינטואיטיבית, זהו האלגוריתם המשלים את הערכים לפי מידת המרחק או הדמיון בין הדוגמאות, ובעולם ה-NBA נפוץ מאוד מצב שבו עבור שני שחקנים בעלי 95% דמיון בנתונים, גם 5% הפרמטרים הנוותרים יהיו דומים, ולכן בחרנו להשתמש באלגוריתם זה באופן נרחב.

### מתודולוגיה לביצוע שלבי עיבוד המידע

יש לציין כי מבחינה מתודולוגית, הקפדנו לבצע את כל החישובים, ולקבוע את כל הפרמטרים של שלבי עיבוד המידע על סמך **סט האימון בלבד**, ולאחר מכן השלכנו את התוצאות שהתקבלו על קבוצות הוולידציה והמבחן. עובדה זו חשובה כדי לדמות את התהליך העתידי שנבצע כאשר נקבל מידע חדש לצורך מתן חיזוי. כלומר, כאשר חישבנו את הספים המותרים של תכונות בעלות ערכים חריגים, קבענו אותם רק על סמך דוגמאות האימון, ולאחר מכן סיננו את הערכים שחרגו מהספים גם בקבוצת האימון, אך גם בקבוצות הוולידציה והמבחן. באותו אופן, כאשר ביצענו נרמול, מציאת הערכי המקסימום והמינימום של כל תכונה התבצעה רק על דוגמאות האימון, ולאחר בניית הטרנספורמציה, ביצענו אותה על כל קבוצות הדוגמאות (ז"א באופן עקרוני שייתכן כי ערכים בקבוצת המבחן לא יהיו בין 0 ל-1 אלא יחרגו מתחום זה). כמוכן כאשר ביצענו השלמת ערכים חסרים בעזרת **Closest Fit**, גם כאן הדוגמאות שלקחנו בחשבון למציאת השכן הקרוב ביותר היו מקבוצת האימון בלבד.

### סינון תכונות (Feature Selection)

כידוע, מציאת תת-קבוצה אופטימלית של תכונות עבור למידה של מושג מטרה שרירותי היא בעיה NP-קשה. זוהי בעיקרה בעיית אופטימיזציה במרחב חיפוש של כל תתי הקבוצות. כדי לפתור אותה בזמן סביר, יש לקבוע מטריקה שבאמצעותה נוכל להעריך תת-קבוצה ספציפית, וכן מומלץ להשתמש בהיוריסטיקה או אלגוריתם שיעזרו לנו לגזור ענפים של מרחב החיפוש באופן יעיל, כדי להימנע מביצוע חיפוש ממצה (Brute Force).

### קיימת חלוקה של אלגוריתמים או שיטות לבחירת תכונות ל-3 הקטגוריות הבאות:

1. **שיטות Filter** – בשיטות אלו יוצרים דירוג של התכונות באופן בלתי תלוי באלגוריתם הלמידה. כלומר, התכונות שנבחרו אינן בהכרח אופטימליות עבור אלגוריתם ספציפי, אלא קבוצת התכונות מכילה תכונות רצויות על פי מטריקה מוגדרת מראש.

דוגמא לשיטה זו היא אלגוריתם Select K Best. האלגוריתם מדרג את כל התכונות על פי פונקציה נתונה, ובחר את k התכונות בעלות הדירוג הגבוה ביותר. פונקציות הניקוד הנפוצות ביותר הן mutual information<sup>4</sup>, שמסוגלת להעיד על קורלציה שאינה בהכרח ליניארית בין תכונות שונות, ופונקציית ANOVA F-score, שהיא בעצם היחס בין שונות הממוצעים של התכונות לבין השונות של הדגימות עצמן. ניתן גם להשתמש בפונקציות פשוטות יותר כגון מקדם הקורלציה של פירסון, אך הוא מסוגל לזהות רק תלויות ליניאריות בין תכונות.

2. **שיטות Wrapper** – בשיטה זו בוחרים אלגוריתם למידה מסוים ומוצאים את תת הקבוצה של תכונות האופטימלית עבור האלגוריתם. לרוב מתייחסים לאלגוריתם בתור קופסה שחורה שיש אפשרות להזין אותה בקלט ללמידה או לביצוע חיזוי. פונקציות הניקוד הנדרשות בשיטה זו הן למשל F-accuracy, score וכו'. כלומר, בדרך כלל בכל שלב, דרוש לבצע למידה של האלגוריתם על תתי הקבוצות האפשריים של התכונות המותרים בצעד הנוכחי, ולאחר מכן להפעיל את האלגוריתם על סט מבחן כלשהו, בשביל לקבל את הניקוד של הצעד. בהתאם לתוצאות המתקבלות בוחרים את הצעד הבא.

דוגמאות נפוצות לשיטה זו הן Sequential Forward Selection (SFS) ו-Sequential Backward Selection (SBS). למען האמת, אלו הם פשוט מימושים אד-הוק לצורך בחירת תכונות עבור אלגוריתמים לפתרון בעיות אופטימיזציה שנתקלנו בהם בקורס המבוא, כגון Gradient Descent או Steepest Hill Climbing. אלגוריתמים אלה מתחילים עם תת קבוצה של תכונות שהיא המצב ההתחלתי, ובאופן איטרטיבי בוחרים את הצעד שמשיג את התוצאה האופטימלית בכל איטרציה. ב-SFS המצב ההתחלתי הוא קבוצת תכונות ריקה, ובכל שלב בוחרים תכונה אחת נוספת שהוספתה לקבוצה המתוחזקת ממקסמת את פונקציית הניקוד. את האיטרציות ניתן להפסיק כאשר מגיעים למספר תכונות קבוע מראש, או לסף של פונקציית הניקוד, או פשוט כאשר הגענו למקסימום מקומי ואף תכונה נוספת אינה משפרת את הניקוד הנוכחי. אלגוריתם SBS פועל בצורה דומה אך מתחיל מקבוצת כל התכונות, ובכל צעד נדרש להסיר תכונה אחת כך שהניקוד של קבוצת התכונות שנותרה הוא מקסימלי.

לשיטות wrapper קיימות דוגמאות רבות נוספות, הכוללות צעדים סטוכסטיים, אלגוריתמי בחירה גנטיים, ועוד מיני שילובים מורכבים של השיטות הנ"ל.

3. **שיטות Embedded** – שיטות אלו מסתמכות על תהליך בחירת תכונות פנימי של אלגוריתם הלמידה. זאת אומרת שבמהלך הלמידה האלגוריתם בוחר את התכונות המתאימות לו.

---

<sup>4</sup> **Mutual Information** – מדד תיאורטי לחישוב התלות בין משתנים מקריים.



הדוגמא הקלאסית לשיטה זו היא באמצעות עץ החלטה, אשר בכל צומת בעץ בוחר לפצל על פי התכונה שממקסמת פונקציית מטרה כגון מדד האנטרופיה<sup>5</sup> או מדד gini<sup>6</sup>. ניתן לדרג את התכונות על פי התרומה שלהן להורדת האנטרופיה (או הגדלת מדד הטוהר של הבנים של הצומת), וניתן פשוט לדרג אותן לפי הסדר בו הן נבחרו לפיצול במהלך בניית העץ, כך שככל שתכונה נבחרה מוקדם יותר היא חשובה יותר. ישנן בנוסף שיטות כגון ביצוע רגרסיה עם רגולריזציה כך שקבוצת התכונות הנבחרת היא זו של כל התכונות עם מקדם שונה מאפס (זו סיבה נפוצה להשתמש ברגולריזציות Lasso או L1 – נפרט על כך בהמשך).

במהלך הניסויים בחנו לפחות דרך אחת מכל שיטת סינון, בכדי למצוא תת קבוצה אופטימלית של תכונות עבור הקטגוריה שאנו מעוניינים לחזות. יש להזכיר כי ישנם בסך הכל עשרה נתונים שאנו רוצים לחזות באמצעות המערכת הלומדת שלנו, וסביר להניח שעבור כל אחד מהם תתאים קבוצת תכונות אחרת, לכן לכל אחד מהם נבחנו לפחות דרך אחת מכל שיטת סינון.

#### השיטות שנבחנו הינן:

1. SelectKBest עם פונקציית ניקוד mutual information.
2. Recursive Feature Elimination - זהו למעשה שם נוסף לשיטת SBS שתוארה לעיל.
3. דירוג תכונות של עץ החלטה.

בפרק הניסויים בהמשך חלק זה של הפרויקט מתוארים בהרחבה ניסויים שביצענו לצורך בחירת תכונות.

## שיטות ומטריקות להערכת ביצועים

### גישות הערכה

ישנה חשיבות גבוהה להחלטה מקדימה של אופן הערכת איכות מודל הלמידה, זאת על מנת שנוכל לקבוע האם התוצאות של המודל שפותח הן טובות (לעיתים ניתן לקבוע סף המוגדר "מספיק טוב" עבור המשימה), לבצע אופטימיזציה של הפרמטרים של המודל כדי לשפר את התוצאות, וכמו כן בכדי להשוות בין מודלים שונים ולבחור מתוכם את המודל שביצעו טובים ביותר.

חשוב לציין כי **חישוב שגיאת החיזוי התבצע אל מול נתוני האמת**, כלומר, עבור חיזוי של ההישגים הסטטיסטיים של עונת 2019 (על סמך אובייקטים המורכבים מנתוני השחקנים בעונה הקודמת – 2018), חושב המרחק בין פלט מערכת החיזוי לבין ההישגים הסטטיסטיים האמיתיים של אותם שחקנים בעונת 2019.

<sup>5</sup> מדד אנטרופיה – מדד לקצב הממוצע של הגידול במידע המופק ממקור הסתברותי.

<sup>6</sup> מדד gini – מדד המייצג את הסבירות של שגיאת סיווג של מופע חדש של משתנה מקרי כלשהו, כאשר שהמשתנה החדש מסווג באופן אקראי בהתאם להתפלגות דוגמאות הלמידה.

## המטריקות עבור הערכת השגיאה חושבו בשתי שיטות:

1. **הערכת השגיאה הכוללת** – בשיטה זו, נלקחו התוצאות עבור כלל השחקנים עבורם התבצע חיזוי, וחושבה השגיאה הכוללת, על פי המטריקה הנבחרת.

2. **הערכת השגיאה לפי עשירונים** – בשיטה זו, חולקו כלל השחקנים לעשירונים (Buckets) על פי תוצאות האמת שלהם בקטגוריה הנחזית. במשחק הפנטזי ישנה חשיבות משתנה לגודל השגיאה בחיזוי שחקנים "טובים מאוד" לעומת השגיאה בחיזוי שחקנים "גרועים" (טעות בחיזוי עבור שחקן שקולע 30 נק' למשחק הינה קריטית, לעומת טעות בחיזוי עבור שחקן שקולע 5 נק'). שיטה זו נועדה לאפשר בחינה של ביצועי מערכת החיזוי בצורה אמינה ומדויקת יותר, על פי חשיבות השחקנים למשחק הפנטזי.

ישנן מספר שיטות נפוצות להערכת אלגוריתמי למידה. גישה אחת היא להעריך את ביצועיו של אלגוריתם על ידי פיתוח חסמים תיאורטיים על מדדים שונים כתלות בפרמטרים של הבעיה, כגון חישוב חסם תחתון על מספר הדוגמאות הדרוש לקבלת שגיאה מסוימת (Sample Complexity). אולם, **אנו נבחן את המערכת שלנו על סמך תוצאות אמפיריות בלבד ולפי המדדים שתוארו לעיל.**

## להלן שיטות הערכה אמפיריות שנשקלו:

1. **שיטת קבוצת המבחן (Test Set)** – בשיטה זו בוחרים קבוצה של אובייקטים מתוך קבוצת הדוגמאות (לרוב כ-10% עד 20%) ושומרים אותן בצד. מאמנים את המודל על הדוגמאות האחרות (שנקראות דוגמאות האימון). לאחר מכן מפעילים את החיזוי של המודל על דוגמאות המבחן ומודדים מתוך התוצאות את פונקציית הניקוד (המטריקה) שנבחרה לצורך הערכה. הערך שהתקבל יהיה הציון של המודל.

2. **שיטת Cross Validation** – בשיטה זו מחלקים את כל הדוגמאות ל- $k$  קבוצות בגודל שווה עבור  $k$  שנבחר מראש. מבצעים  $k$  איטרציות כשבכל אחת מהן מוציאים קבוצה אחרת מתוך  $k$  הקבוצות להיות קבוצת המבחן, ומאמנים את המודל על  $k-1$  הקבוצות הנותרות. כעת מפעילים את החיזוי על קבוצת המבחן של האיטרציה, ומודדים את פונקציית הניקוד שנבחרה על התוצאות. הציון הסופי של המודל יהיה הממוצע של הניקוד שלו על פני כל האיטרציות.

3. **שיטת Leave-One-Out (LOO)** – זהו למעשה מקרה פרטי של Cross Validation שבו  $k$  נבחר להיות מספר הדוגמאות. כלומר מבצעים מספר איטרציות כמספר הדוגמאות, כאשר בכל שלב מאמנים את המודל על כל הדוגמאות חוץ מדוגמה בודדת שמהווה את קבוצת המבחן. מבצעים חיזוי על דוגמת המבחן ומחשבים את פונקציית הניקוד שהתקבלה. בסוף ציון המודל הוא שוב ממוצע הניקוד על פני כל האיטרציות.

אף אחת מהשיטות אינה אופטימלית בכל המקרים, לכל אחת מהן ניתן למצוא דוגמאות בהן היא פחות מדויקת מהאחרות, כתלות במודל הלמידה ובאלמנטים נוספים של הבעיה. **בחרנו להשתמש בשיטת קבוצת המבחן וכן בשיטת Cross Validation לצורך הערכת ביצועים**, כאשר שיטת קבוצת המבחן היחידה שימשה בשלבים הראשונים של תהליך הפיתוח לצורך בחינת הביצועים הראשוניים של המערכת, ושיטת ה-Cross Validation היא זו ששימשה לבסוף להערכת הביצועים בשלב הניסויים. פסלנו את שיטת LOO מכיוון שהסיבוכיות של זמן החישוב שלה גדולה מדי, והיא לא פרקטית לביצוע כאשר יש מספר גבוה של ניסויים שמתוכנן להתבצע. בסופו של דבר, החלטנו לדווח בתוצאות הניסויים רק על תוצאות בשיטת Cross Validation, מכיוון שבשיטה זו התקבלו התוצאות הכי יציבות ואמינות. על סמך הניקוד שקיבלנו בשיטה זו בחרנו אילו מודלים עובדים טוב יותר ואילו פחות.

### מטריקות להערכת ביצועי המודל

**הבעיה שעומדת בפנינו בשלב החיזוי (לכל הקטגוריות החיזוי) הינה בעיית רגרסיה של ערך נומרי.** קיימות מספר מטריקות סטנדרטיות להערכת דיוק של מודל רגרסיה. להלן פירוט המטריקות בהן השתמשנו כדי להעריך את הפלט של המערכת.

1. **Mean Square Error (MSE)** – זהו ממוצע ריבועי הסטיות של ערכי החיזוי מערכי האמת. באופן פורמלי, אם מספר דוגמאות המבחן שלנו הוא  $n$ , ווקטור החיזוי של המודל הוא  $\hat{y}$  כאשר וקטור הערכים האמיתיים הוא  $y$ , אז:

$$MSE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

2. **Mean Absolute Error (MAE)** – ממוצע הערכים המוחלטים של הסטיות של ערכי החיזוי מערכי האמת. באופן פורמלי:

$$MAE(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

3. **Median Absolute Error** – זהו ערכו של חציון הערכים המוחלטים של סטיות ערכי החיזוי מערכי האמת.

מטריקת MSE היא הנפוצה ביותר בניתוחים סטטיסטיים של רגרסיה, אך חסרונה הוא בכך שיחידות השגיאה אינן באותו סדר גודל של יחידות הנתון עליו מבוצעת רגרסיה, אלא בריבוע. לכן ברוב הניסויים שלנו התייחסנו יותר לשגיאות האבסולוטיות, על מנת לאפשר הערכה אינטואיטיבית ופשוטה של מרחק החיזוי שלנו מערכי האמת.

## הערכה אל מול חיזוי המומחים

בעיה בולטת של הערכת תוצאות רגרסיה הינה **הקושי בקביעת איכות התוצאות**, כלומר, האם תוצאות של ניסוי הן טובות או מספקות. בהנחה סבירה שלא נצליח להגיע לשגיאת חיזוי אפס בקטגוריות השונות, **עלינו להכין מראש מטרת יעד שהשגה שלה תוגדר כהצלחה של המערכת הלומדת**.

לצורך כך, אספנו מראש תוצאות חיזוי של מומחים מאתר [vi.hashtagbasketball.com](https://vi.hashtagbasketball.com), עבור הקטגוריות אותן אנו מעוניינים לחזות בעצמנו. **אלו נתונים שנחזו על ידי מומחים בעלי ניסיון רב בתחום ה-NBA ומשחק הפנטזי**, המנסים לחזות את הישגי השחקנים בעונות העתידיות, ומסופקים בפלטפורמות משחק הפנטזי השונות. נציין כי לא ידוע לנו על האופן על פיו מתבצעים החיזויים, או בעזרת אילו מודלים המומחים מבצעים את החיזוי שלהם, אך הנתונים נחשבים לאמינים ונמצאים בשימוש נפוץ אצל כל שחקני ליגות הפנטזי, כאשר למעשה חיזויים אלו הם הנתונים העיקריים עליהם מסתמך סוכן במשחק הפנטזי על מנת להרכיב קבוצה בדראפט. נתונים אלו מתפרסמים בכל שנה בדרך כלל מעט לפני תחילת העונה.

הסתמכנו על נתוני החיזוי לעונת 2019 והשתמשנו בהם בכדי לחשב את השגיאה הממוצעת של המומחים (על פי אותן המטריקות שפורטו לעיל). **את הנתונים הללו השוונו עם תוצאות מערכות החיזוי הנבחרות (בכל קטגוריה) כדי לאמוד האם מרחק השגיאה שהשיגה המערכת הינו מרחק "סביר", כלומר, באותו סדר גודל של שגיאת המומחים**.

## אלגוריתמי למידה לשלב החיזוי

**בתת-פרק זה נפרט בקצרה את המודלים אותם בחנו לצורך חיזוי הקטגוריות**. בחלק מהמודלים נתקלנו כבר בקורס המבוא, אך את חלקם למדנו באופן ממוקד למטרת הפרויקט. המימוש בו נעזרנו לכל אלגוריתמי הלמידה הוא של ספריית `sklearn`.

**המודלים הנבחרים נבחרו על פי מספר שיקולים** – התאמה לחיזוי ערכים רציפים (בעיית רגרסיה), התאמה למספר דוגמאות לא רב, גיוון בשיטות הלמידה ופשטות יחסית באופן הפעולה. ככלל, לא ניתן היה לאפיין על סמך ידע מוקדם מודל ספציפי אשר יתאים בצורה מיטבית לבעיה הנ"ל, ועל כן נדרש לבחון מגוון יחסית רחב של מודלים.

### 1. רגרסיה ליניארית

מודל זה הוא מודל סטטיסטי פשוט אשר מניח קיום תלות ליניארית בין וקטור המשתנים לבין ערך התוצאה, אך כולל גורם של רעש אקראי שמוביל לכך שלא ניתן לשערך את התוצאה באופן מדויק על סמך הקלט. המודל שואף להקטין את השגיאה הריבועית הממוצעת (MSE). התיאור שלו באופן פורמלי הוא:

$$\min_w \|Xw - y\|_2^2$$

כאשר  $X$  היא מטריצת הערכים של הדוגמאות,  $y$  הוא וקטור ערכי התוצאה האמיתיים, ו- $w$  הוא וקטור המכיל את המשקל של כל תכונה. לפתרון של  $w$  לבעיית מזעור MSE ניתן להגיע באופן סגור או באופן איטרטיבי באמצעות אלגוריתם Widrow-Hoff (או LMS).

## 2. רגרסיה ליניארית עם רגולריזציה

למעשה תחת סעיף זה נכנסים שלושה מודלים שונים שבחנו. כל המודלים מניחים תלות ליניארית בדומה למודל רגרסיה ליניארית רגיל, אך הם נבדלים זה מזה באילוצים שהם מכניסים לבעיית מזעור ה-MSE:

### א. Lasso Regression –

במודל זה רוצים למזער את סכום השגיאה הריבועית וגורם נוסף שהוא הנורמה  $L1$  של וקטור המשקלים  $w$ , עם מקדם שנקבע מראש. התיאור הפורמלי הוא:

$$\min_w \|Xw - y\|_2^2 + \lambda \cdot \|w\|_1$$

שימוש נפוץ במודל זה הוא כאשר מנסים למצוא פתרון דליל (sparse) לבעיה, כלומר וקטור משקלים שכמה שיותר מהמשקלים בו הם 0. ניתן להשתמש במודל זה בתור שיטה לסינון תכונות מסוג Embedded, כאשר לאחר תהליך הלמידה של המודל מקבלים שהתכונות שנבחרו לצורך מתן רגרסיה לבעיה הן אלו שהמקדמים שלהן בוקטור  $w$  שונים מאפס.

### ב. Ridge Regression –

מודל זה ממזער את סכום השגיאה הריבועית עם גורם נוסף שהוא נורמה  $L2$  בריבוע של וקטור המשקלים  $w$ , עם מקדם שנקבע מראש. התיאור הפורמלי הוא:

$$\min_w \|Xw - y\|_2^2 + \lambda \cdot \|w\|_2^2$$

הרעיון במודל זה הוא לא רק להקטין את השגיאה הריבועית אלא גם למצוא וקטור משקלים קטן. באופן אינטואיטיבי הפתרון לבעיה מתקשר לעיקרון התער של אוקאם, מכיוון שככל שהמשקלים של  $w$  קטנים יותר, כך מתקבל פתרון פשוט יותר. הקביעה של המקדם  $\lambda$  משפיעה באופן הבא – ככל שהמקדם גדול יותר מרכיב הגודל של  $w$  בבעיה הינו בעל חשיבות רבה יותר והדגש הוא על הקטנת המשקלים, כאשר באופן קיצוני אם המקדם שואף לאינסוף, הפתרון יהיה וקטור האפס ללא קשר לדגימות  $X$  או לוקטור התוצאה  $y$ . מן הצד השני, ככל שהמקדם קטן יותר, כך הדגש הוא יותר על הקטנת רכיב השגיאה הריבועית. בבעיות רבות יש למצוא איזון בין שני גורמים אלו על מנת להשיג יכולת חיזוי ורגרסיה מדויקים יותר על ערכים חדשים.

### ג. Elastic Net –

מודל זה מנסה לשלב את היתרונות של שני המודלים האחרונים. הוא מכניס שני גורמי רגולריזציה שהם נורמה  $L1$  של  $w$  וכן נורמה  $L2$  של  $w$ , עם שני מקדמים בהתאמה שסכומם הוא 1. באופן פורמלי:

$$\min_w \|Xw - y\|_2^2 + \alpha \|w\|_1 + (1 - \alpha) \cdot \|w\|_2^2$$

למעשה ניתן לראות כי Lasso וגם Ridge הם מקרים פרטיים של Elastic Net עם ערכי אלפא 1 ו-0. מודל זה שואף למצוא פתרון ביניים שהוא גם דליל באופן יחסי אך גם קטן מבחינת נורמת L2 שלו.

כל מודלי הרגרסיה הליניארית הנ"ל מתאפיינים בפשטות שלהם, ובמהירות למידה וחיזוי גבוהים. הם מוצלחים במיוחד עבור בעיות המקיימות תלות ליניארית או תלות שקרובה להיות כזו. כדי להגיע לביצועים טובים במודלים הנ"ל מומלץ להקפיד על נרמול של התכונות. החסרון העיקרי שלהם הוא שהם אינם מתמודדים היטב עם רעשים במדידות או דוגמאות חריגות, כאשר בתרחישים אלו לרוב השימוש ברגולריזציה מוביל לתוצאות עדיפות.

### 3. מודלי SVR ו-LinearSVR

מודל זה מבוסס על SVM (Support Vector Machine), מודל אשר מוצא היפר-מישור<sup>7</sup> (כלומר מפריד ליניארי) במרחב ממימד גבוה יותר ממרחב התכונות הנתונות, זאת באמצעות פונקציות Kernel אשר משמשות לחישוב מכפלה פנימית בין דוגמאות במרחב המימד הגבוה ללא ביצוע המיפוי באופן ישיר. בבעיית סיווג, ה-SVM שואף למצוא הפרדה שממקסמת את ה-margin<sup>8</sup> (השוליים) בין מחלקות הדוגמאות השונות, תוך הקטנת ההפסד שנובע מדוגמאות שנמצאות בתוך ה-margin. המושג וקטורי תמיכה (Support Vector) מתאר את הדוגמאות הכי קרובות להיפר-מישור המפריד, והן אלו שקובעות איפה המישור יימצא, כאשר כל הדוגמאות שאינן מהוות וקטורי תמיכה, אינן משפיעות כלל על המודל הסופי שהאלגוריתם לומד.

מודל ה-SVR הוא וואריאציה על מודל ה-SVM המשמשת לחיזוי ערכים רציפים ולא לצורך סיווג. מודל ה-SVR מנסה למצוא היפר-מישור שכל הדוגמאות (במימד הגבוה יותר) נמצאות במרחק של עד אפסילון נתון ממנו, כאשר אפסילון זה מגדיר את השוליים בהם מתירים קיום של דוגמאות. ה-SVR נותן את חיזוי הערך הרציף בהתאם למיקום של דוגמת הקלט ביחס להיפר-מישור שנלמד. לשתי גרסאות האלגוריתם יש פרמטר C שקובע את מידת "הענישה" (הקנס) שיש לשלם על דוגמאות שחורגות מהשוליים. בנוסף, ניתן לבחור את סוג פונקציית ה-Kernel, כלומר את המימדים הנוספים שהאלגוריתם יוצר בזמן הלמידה, המבוססים על קשרים בין התכונות המקוריות. ה-Kernel הפשוט ביותר הינו ה-Kernel הליניארי, אשר מאפשר הוספת מימדים המהווים פונקציות ליניאריות של התכונות המקוריות. סוגי Kernel נוספים יוצרים פונקציות פולינומיות, פונקציות המבוססות על פונקציית הסיגמואיד, וגם Kernel מסוג rbf המייצר פונקציות על סמך מרחק בין הדוגמאות (ניתן לחשוב על הפונקציות של Kernel זה בתור כדורים במימד גבוה). מודל LinearSVR הוא פשוט מימוש מהיר יותר של מודל SVR עם גרעין ליניארי מובנה.

<sup>7</sup> היפר מישור – תת מרחב וקטורי ממימד  $n-1$  של מרחב וקטורי ממימד  $n$ .

<sup>8</sup> Margin – המרחק בין המפריד הליניארי המתקבל ע"י SVM לבין וקטורי התמיכה.

#### 4. מודל Decision Tree

מודל זה נלמד בהרחבה בקורס המבוא. מודל זה אינו נבחן בפני עצמו כמערכת לומדת, אלא שימש כבסיס למודלים אחרים ולשלב ה-Feature Selection. אלגוריתם זה בונה מודל הכולל צמתי החלטה בינאריים היוצרים מבנה של עץ. הבניה מתבצעת באופן רקורסיבי כאשר בכל שלב נבחרת תכונה אחת וערך אחד שלה שישמשו לפיצול הצומת הנוכחי, זאת על סמך קריטריון קבוע מראש (מדד אנטרופיה, מדד gini). כאשר מתקבלת דוגמא שיש לסווג (במקרה הנ"ל - לבצע לה רגרסיה), מתחילים מצומת השורש ובכל צומת מפעילים את פונקציית הבחירה הבינארית ששויכה אליו, ממשיכים במסלול אל הבן המתאים, עד שמגיעים לצומת עלה. בבעיית רגרסיה מחזירים את ערך הקטגוריה הממוצע על פני הדוגמאות ששייכות לעלה.

#### 5. מודל Random Forest

מודל זה יוצר ועדה (Ensemble) המורכבת ממספר מודלים בסיסיים מסוג עץ החלטה. המודלים שהצגנו עד פה הם מודלים המייצרים היפותזה אחת ויחידה עבור הבעיה. כאן מדובר לראשונה על מודל המייצר ועדה של היפותזות שונות זו מזו, ומשלב את החיזוי של כולן על מנת לתת חיזוי אחד סופי. המוטיבציה העיקרית לשימוש ב-Random Forest על פני שימוש בעץ החלטה בודד היא שיפור ביצועי המערכת, בהיבט של דיוק התוצאות. העיקרון המרכזי באלגוריתם זה הינו שעצי ההחלטה המשותפים לוועדה יהיו בלתי תלויים זה בזה, וזאת על מנת ליצור גיוון בהיפותזות שהם מייצגים. ברור כי יצירת ועדה של עצים שבה כל העצים זהים אינה אפקטיבית, מכיוון שאז ההיפותזה המשותפת שלהם שקולה להיפותזה של עץ בודד מתוך הוועדה.

ישנן מספר דרכים אפשריות להשיג אי תלות בין עצי הוועדה – ניתן להגריל לכל עץ קבוצת אימון שונה של דוגמאות מתוך כלל הדוגמאות (כאשר פעולה זו מבוצעת עם החזרה, זה נקרא Bootstrapping), ניתן לאמן כל עץ על אותן דוגמאות אך עם תת קבוצה רנדומלית של תכונות מתוך כלל מאגר התכונות, וניתן לאמן כל עץ עם היפר-פרמטרים שונים.

מודל זה הינו ניסיון לממש את עיקרון "חוכמת ההמונים". חשוב לזכור שעל מנת שתהיה תועלת בוועדה (או ב"המונים") חובה שיתקיימו נקודות אי הסכמה בין חברי הוועדה, כלומר הבדלים בין ההיפותזות שלהם. נציין כי ישנן דרכים שונות לשלב את ההצבעות של חברי הוועדה. לדוגמא, ניתן ליצור וועדה שאינה מורכבת מעצי החלטה, אלא מאלגוריתמים המסוגלים לתת הערכה סטטיסטית לנכונות החיזוי שלהם, ולהשתמש בנתון זה בתור משקל ההצבעה. אנו בחרנו לייחס משקל שווה להחלטה של כל עץ.

#### 6. מודל AdaBoost

זהו מודל נוסף שמטרתו ליצור ועדה של אלגוריתמי למידה בסיסיים, אך שונה במהותו מאלגוריתם Random Forest. ההבדל העיקרי במודל זה, הינו שחברי הוועדה השונים תלויים זה בזה. אלגוריתם זה פותח במקור על מנת לענות על השאלה – "האם ניתן לשפר אלגוריתם למידה חלש (לבצע Boosting)?" באופן מופשט, אלגוריתם למידה חלש מוגדר כאלגוריתם כללי אשר עבור כל בעיה שניתן לו, הביצועים שלו יהיו טובים קצת יותר מאשר ניחוש אקראי. באופן פרקטי, בניית אלגוריתם כזה אינה טריוויאלית בכלל, כיוון שהדרישה על הבטחת הביצועים שלו צריכה להיות תקפה לכל התפלגות

(קבועה אך לא ידועה). משמעות השאלה הנ"ל היא האם קיים תהליך שבעזרתו ניתן לקחת אלגוריתם למידה חלש, ולהפוך אותו לאלגוריתם למידה חזק (כזה שבאופן תיאורטי מסוגל להיות כמה מדויק שנרצה, כתלות במספר הדוגמאות הזמינות ללמידה).

אלגוריתם AdaBoost הצליח להוכיח באופן תיאורטי וגם דרך תוצאות אמפיריות כי התשובה לשאלה היא חיובית. האלגוריתם פועל באופן איטרטיבי בצורה הבאה – מאתחלים וקטור משקלים עבור דוגמאות האימון עם משקל אחיד לכל דוגמא. בכל איטרציה מאמנים מסווג חלש על כל הדוגמאות כאשר החשיבות של כל דוגמא נקבעת לפי וקטור המשקל, ואז בוחנים את הביצועים שלו על כל הדוגמאות. בסוף האיטרציה, מעדכנים את וקטור המשקל כך שמשקל הדוגמאות בהן טעינו גדול יותר באיטרציה הבאה, ומשקל הדוגמאות בהן צדקנו קטן יותר (וקטור המשקל מנורמל). בצורה זו מתקבל כי כל מסווג בוועדה תלוי במסווגים שנבנו לפניו. בסוף התהליך האיטרטיבי, הסיווג הסופי נקבע על פי הצבעה ממושקלת של כל המסווגים החלשים שנבנו, וזאת בהתאם לטעויות של כל אחד מהם.

## 7. מודל Multi-Layer Perceptron

מודל זה נחשב לרשת הנוירונית הבסיסית ביותר. מכיוון שזהו מודל מוכר בתחום הלמידה, לא נרחיב בפרוטרוט לגבי האלגוריתם שעומד מאחוריו, אך נזכיר את עיקר הרעיון. רשתות נוירונים נבדלות מאלגוריתמי למידה קלאסיים ביכולת לעבד את המידע ולאסוף ממנו תכונות מורכבות בצורה היררכית, כחלק מובנה מתהליך הלמידה. הדרך העיקרית בה תהליך זה מתבצע היא על ידי סדרה של (אחת או יותר) טרנספורמציות לא ליניאריות. רשתות אלה מורכבות משכבות, כל שכבה היא פונקציה שמגדירה מה הפלט של השכבה בהינתן הקלט, וכמו כן את הנגזרות החלקיות ביחס לקלט ולפרמטרים. ה-MLP מבצע פעולה הנקראת Backpropagation, כלומר מידע לא זורם רק קדימה ברשת, אלא גם אחורה, זאת על מנת לעדכן את מרכיבי הרשת השונים בזמן הלמידה.

ה-MLP בנוי מערימה של שכבות אקטיבציה (הנקראות שכבות נסתרות, או Hidden Layers) אחת על גבי השנייה, כאשר המטרה היא לייצר סיווג. בסוף תהליך הלמידה המודל יוצר טרנספורמציה שמורכבת משרשור פעולת השכבות השונות, כך שבמרחב הסופי המתקבל, קיימת הפרדה ליניארית בין מחלקות שונות של הדוגמאות (הפרדה אותה מסוגל לזהות פרספטרון רגיל).

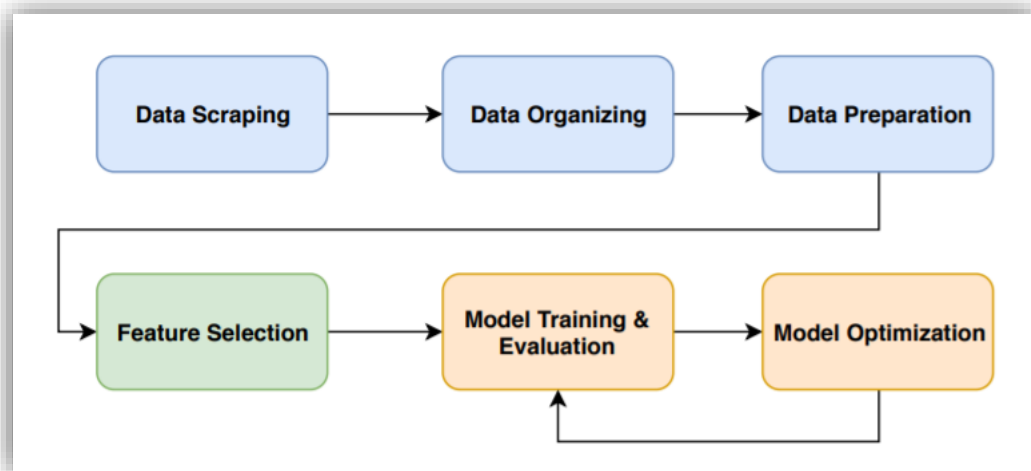
לאלגוריתם זה ישנם היפר-פרמטרים רבים, וביניהם ניתן לקבוע את מספר השכבות ברשת, את פונקציית האקטיבציה הלא ליניארית (למשל פונקציית הסיגמואיד, פונקציית ReLU, וכו'), את מקדם הרגולריזציה ועוד. לאלגוריתם זה יתרון בולט כאשר צריך ללמוד מושג מטרה מורכב ובפרט כאשר הקשר בין הקלט לפלט אכן אינו ליניארי. מצד שני זמן הלמידה שלו עלול להיות ארוך מאוד ביחס לאלגוריתמים אחרים, הפתרון שהוא מספק הוא מורכב וקשה לשלוף ממנו תובנות עמוקות (שיהיו בעלות משמעות עבור בני אדם) לגבי אופי הבעיה, בניגוד לעץ החלטה למשל.



## תיאור מבנה מערכת החיזוי

המערכת לפתרון בעיית החיזוי של הפרויקט מורכבת ממספר סקריפטים (בשפת Python ועם שימוש נרחב בסיפרייה Pandas), אשר כל אחד מהם עומד בפני עצמו ומבצע חלק מסוים מהתהליך הכולל, החל משלב איסוף המידע, חלוקה ועיבוד וכלה בשלב מתן החיזוי ובחינת המודלים השונים. כל אחד מן החלקים הלוגיים של המערכת בנוי מסקריפט אחד או יותר כנ"ל. המטרה באופן מימוש זה היא בניית סט פעולות אשר יאפשר שחזור התהליכים בנוחות בפעמים הבאות, כאשר נרצה לייצר אובייקטים לחיזוי בכל עונת NBA.

נתייחס לכל חלק לוגי ונתאר מהן הפעולות עליהן הוא אחראי, לכל שלב לוגי של מימוש הפתרון נתייחס בתור "מודול" (אין קשר ישיר בין המונח הזה לבין המונח module של שפת Python).



איור 7: תרשים מבנה מערכת החיזוי

### המודולים של המערכת

#### 1. כריית המידע (data scraping):

מודול זה כולל הורדת מידע נגיש לכלל הציבור מהאתרים עליהם פירטנו בפרק מאגרי המידע, ששימשו לנו כמקורות מידע עבור בניית האובייקטים עליהן נסמך תהליך הלמידה. המידע הנ"ל כולל נתונים כלליים על שחקני NBA, סטטיסטיקות וחיזויי עבר של מומחים המשמשים אותנו לצורך הערכת המודל הסופי. יש לציין כי המימוש של המודול התבצע באופן ידני לחלוטין, כולל גישה לאתרים השונים ברשת האינטרנט, והורדה או העתקה של הנתונים עבור כל עונה וסידורם בקבצי csv. חלק מהאתרים הנ"ל מאפשרים הורדה של קבצי csv עם סטטיסטיקות עבור עונה נתונה, אך חלקם מציגים את הנתונים על גבי דף אינטרנט ולא מאפשרים הורדה, והם דרשו עבודה עדינה של סידור המידע הרלוונטי לכל עונה ולפי פילוח מסויים (פר משחק, פר 36 דקות וכו') ולאחר מכן העתקה שלו אל קובץ csv.

בסיום השלב, הפלט שלו הוא מספר קבצים המכילים מידע לא-מעובד (Raw Data), כאשר פרטי מידע שונים על כל שחקן, כל עונה ולפי פילוחים שונים מפורזים על פני מספר קבצים. כל המידע המשמש בשלבים המאוחרים יותר של התהליך כבר קיים בסוף שלב זה.

## 2. סידור המידע ואיסוף התכונות (data organization):

מודול זה מקבל מספר רב של קבצי csv בעלי מבנה ומאפיינים שונים כקלט. הוא אחראי על איחוד בין טבלאות מידע שונות, סידור העמודות וביצוע התאמה בין פרטי מידע בקבצים שונים המתייחסים לאותו שחקן NBA. בשלב זה בוצעו כמעט כל התיקונים המפורטים בפרק מאגרי המידע, כגון התאמה בין אופנים שונים לכתיבת שם של שחקן ובין סוגי כתיב שונים של קבוצות ה-NBA, סיכום נתוני שחקנים שעברו קבוצות במהלך העונה, וחישובים של תכונות ייחודיות שהחלטנו להוסיף בעצמנו על סמך המידע שנאסף (כגון מספר שנות הניסיון של שחקן בליגה, שחושב על סמך שנת הכניסה שלו לליגה והשנה של אותה העונה). לאחר מכן, מודול זה כולל גם את היצירה של טבלה מאוחדת שלראשונה מכילה אובייקטים עליהם יבוצעו הלמידה והחיזוי, בהתאם לאופן בו החלטנו לייצג אובייקט של שחקן בעונה ספציפית. המימוש של מודול זה ברובו התבצע באופן אוטומטי על ידי סקריפטים שונים, אשר נועדו לצורך ביצוע חוזר של הפעולות במידת הצורך. חלק מהמימוש עדיין בוצע באופן ידני. התהליך המתבצע בשלב לוגי זה הינו קריטי, מכיוון שהחלטות שהתקבלו במימושו משפיעות על יכולת החיזוי הסופית שהשגנו. **בסוף שלב זה, הפלט הינו טבלה אחת גדולה המכילה דוגמאות המוכנות ללמידה.** טבלה זו מיוצרת במספר וואריאציות שונות כמספר הקטגוריות שאנו מעוניינים לחזות, כך שכל וואריאציה מכילה עמודה אחת בלבד שהיא קטגוריית החיזוי, וכל שאר העמודות הן תכונות. נזכיר כי כל שורה בטבלה זו מייצגת שחקן NBA אחד בעונה ספציפית.

## 3. עיבוד המידע (data preparation):

מודול זה מקבל כקלט את טבלת הדוגמאות המסודרת שהיא הפלט של השלב הקודם. **טבלה זו עדיין מכילה תכונות שאינן מספריות, דוגמאות בעלות ערכים חריגים, תאים עם ערכים חסרים וטווחים שונים לכל אחת מן התכונות.** במהלך שלב זה מופעלות מספר בדיקות, סינונים וטרנספורמציות של המידע, כך שבסופו מתקבלת טבלת דוגמאות אחידה, בעלת ערכים נומריים בלבד וללא ערכים חסרים, המוכנה לשמש כקלט עבור אלגוריתמי הלמידה.

בנוסף, מתבצעת חלוקה – מודול עיבוד המידע הוא המודול הראשון שלאחר שנכתבו הסקריפטים המבצעים אותו, ממומש כולו באופן אוטומטי וניתן להריץ אותו בעזרת command line יחיד. הסקריפט העיקרי בחלק זה מומש באופן כזה שניתן להשתמש בו על מנת להוציא טבלת דוגמאות מוכנה עם עמודה אחת ספציפית של קטגוריה לחיזוי. הסקריפט רץ בתחילת כל ניסוי שביצענו בהמשך הפיתוח, על מנת להכין מידע עבור קטגוריית מטרה ספציפית לחיזוי (למשל – נקודות למשחק), כלומר, אחד מהפרמטרים שניתן לקבוע הינו איזו קטגוריה רוצים לחזות, לאחר מכן המידע מעובד כך שהפלט תואם לקטגוריה המבוקשת.

#### 4. **בחירת תכונות (feature selection):**

מודול זה כולל מעטפת של אלגוריתמים שונים מ-3 השיטות לביצוע Feature Selection אותן הזכרנו. במהלכו מייצרים את האלגוריתם הנדרש (תוך שימוש בסיפרייה sklearn) ומפעילים אותו על קבוצת דוגמאות מוכנה המתקבלת כקלט. לבסוף מחזירים את סט התכונות הנבחר על ידי האלגוריתם. הקלט הנוסף למודול הינו מספר התכונות הרצוי, כאשר כל אחד מן האלגוריתמים בהם השתמשנו לצורך בחירת התכונות מקבל את מספר תכונות היעד ומחזיר תמיד קבוצת תכונות בגודל זה בדיוק. זהו אחד המודולים המרכזיים ששימשו אותנו בעת עריכת הניסויים.

נזכיר כי האופציות בהן מודול זה תומך לבחירת תכונות הן – שיטת filter עם אלגוריתם Select K Best, שיטת Wrapper המקבלת מודל למידה ומבצעת Recursive Feature Elimination עד הגעה למספר תכונות היעד (כאשר תמיד נגיע אליו בדיוק גם אם במהלך הסרת התכונות חלה ירידה בביצועים במקום עליה), ובשיטת Embedded באמצעות למידת עץ החלטה על דוגמאות האימון ולאחר מכן דירוג התכונות על פי המודל וחיתוך שלהן לפי הכמות המבוקשת.

#### 5. **הפעלת מודל למידה והערכת ביצועים:**

במודול זה מפעילים מודל נתון (המשתמש במימוש נתון ע"י ספריית sklearn) על קבוצת דוגמאות האימון לאחר הטלה של המידע על התכונות שנבחרו בלבד. מתבצע אימון של המודל (fit), ולאחר מכן חיזוי קטגוריית הרגרסיה הנבחרת על קבוצת המבחן והוצאת תוצאות החיזוי אל קובץ. כמובן את הפעלת אלגוריתם הלמידה בשלב זה ניתן לבצע גם לפי שיטה של Cross Validation. בסוף ניתן לקבל את מדדי הביצועים המבוקשים וכך להעריך את השגיאה של מודל הלמידה.

נזכיר כי הערכת המודל מתבצעת על פי מספר מטריקות, כאשר כל אחת מהן מחושבת גם על כלל הדוגמאות לקבלת מדד אחד גס, ובנוסף לפי Buckets, כלומר ערך המטריקה על כל עשירון של שחקנים לפי דירוג הפנטזי שלהם. על מנת לקבל הבנה עמוקה בנוגע לביצועים של כל מודל ישנו צורך בבחינת התוצאות באופן ידני והתחשבות ביקורתית בביצועי המודל על כל Bucket. השוואה זו תלויה בתוצאות הייחודיות של המודלים הבולטים במשימת חיזוי כלשהי, ועקב המורכבות שלה החלטנו מראש לא לנסות ליצור פונקציית בחירה אוטומטית המקבלת תוצאות של שני אלגוריתמים ומכריעה מי טוב יותר באופן מוחלט. ההשוואה הסופית כוללת שיקולים המותאמים לעיתים באופן ספציפי לקטגוריה הנחזית ואף אלמנטים סובייקטיביים שתלויים במטרת השימוש בתוצאות החיזוי.

#### 6. **אופטימיזציית מודל למידה:**

מודול זה מקבל כקלט קובץ csv שבו נרשמו התכונות שנבחרו עבור אלגוריתם למידה כלשהו על ידי מודול בחירת התכונות, וכן מודל למידה והגדרה של מרחב חיפוש המתאר את כל צירופי ההיפר-פרמטרים אותם אנו מעוניינים לבחון. המודול מבצע חיפוש ממצה (באמצעות כלי ייעודי של ספריית sklearn) במרחב כל הצירופים של ההיפר פרמטרים שהוגדרו, ומוציא את הניקוד של כל צירוף אל קובץ csv. כחלק מהתהליך של שלב זה, התבצע מעבר על הפלט ונבחר באופן ידני צירוף ההיפר פרמטרים האופטימלי. יש להדגיש כי בשלב בחירת ההיפר פרמטרים של המודל התחשבנו אך ורק במטריקות שחושבו על כלל דוגמאות הקלט (ללא Buckets). נוצר הכרח לאפשר בשלב זה השוואה אוטומטית בין ביצועי המודל עם

פרמטרים שונים, ולא היתה אפשרות להשוות את הביצועים בעזרת שיטת ה-Buckets, עקב המספר הרב של הפלטים המתקבלים בשלב זה, והעובדה שההבדלים בין תוצאות המודלים הינם עדינים מאוד (עד כדי עשירות הנקודה). הפתרון לאתגר זה היה שימוש בפונקציית השוואה מעט גסה יותר אך שתבצע באופן אוטומטי. היבט נוסף שתומך בביצוע ההשוואה האוטומטית ולא על סמך התאים היא שלהערכתנו למרות שהביצועים הכוללים של מודל מסויים משתנים באופן משמעותי עם פרמטרים שונים, ההתפלגות של ביצועי המודל על פני ה-Buckets היא פונקציה של האלגוריתם והיא משתנה באופן זניח עבור הפעלה של אותו המודל עם פרמטרים שונים.

**שלב זה בוצע על מודלים ספציפיים שהוכיחו יכולות טובות בשלב הערכת הביצועים, אך התוצאות והפלט שלו מפעפעלים בחזרה לשלב הקודם.** זאת על מנת לבחון שוב את המודל עם סט התכונות המקורי אך עם הפרמטרים האופטימליים שמצאנו, ולהוציא את הפלט של שלב הערכת הביצועים הכולל את המטריקות לפי ה-Buckets, כאשר בסוף הניסויים השתמשנו גם בנתונים אלה לטובת בחירת המודל האידיאלי לכל קטגוריה (באמצעות השוואה ידנית ולצורך הכרעה סופית בין מודלים שהשיגו תוצאות יחסית דומות).

## ניסויים למערכת החיזוי

### מתודולוגיה ניסויית

מטרת חלק זה של הפרויקט הינה פיתוח מערכת לחיזוי נתונים סטטיסטיים של שחקני NBA, לצורך שימוש במשחק הפנטזי. על מנת לבחור מודל אידיאלי לכל קטגוריית חיזוי, ביצענו סדרת ניסויים למודלים השונים שתוארו בפרק פיתרון בעיית החיזוי. ניסויים אלה נועדו לענות על השאלות הבאות:

- **איזה מודל חיזוי הינו האידיאלי** עבור חיזוי הערכים בכל קטגוריה (מבין המודלים שנבחנו).
- **מציאת קונפיגורציה מתאימה** ביותר של פרמטרים עבור כל מודל חיזוי וקטגוריה נבחרים.
- עבור כל מודל וקטגוריה – **תת קבוצה של תכונות נבחרות** מתוך כלל התכונות שהוגדרו לאובייקט השחקן.

על מנת להשיג תוצאות אלו, הניסויים חולקו לשני חלקים, אשר בכל אחד מהן נמדד מרחק שגיאת החיזוי באמצעות מטריקות שונות. שלבי הניסוי מחולקים באופן הבא:

1. **שלב ראשון** – בחינה ראשונית של מודלי החיזוי לצורך סינון ראשוני של מודלים מובילים.
2. **שלב שני** – אופטימיזציה והיזון חוזר של מודלי חיזוי נבחרים.

### מדדים ושיטות לבחינת תוצאות החיזוי

לצורך בחינת תוצאות החיזוי נעשה שימוש בשלוש מטריקות מרכזיות למדידת מרחק השיגאה, כפי שתואר בתת הפרק "שיטות ומטריקות להערכת ביצועים" בפרק החיזוי – **Mean Square Error**

(MSE), Mean Absolute Error (MAE), Median Absolute Error. ההתייחסות הן לשגיאה הממוצעת והן לשגיאה החציונית הינה חשובה להערכת הביצועים באופן איכותי, כיוון שלרוב במשחק הפנטזי, שחקנים בקצה הדירוג (השחקנים המדורגים בדירוג הכללי בין 150-350) הינם בעלי משמעות פחותה ביותר עבור המשחק (כיוון שלרוב אינם נבחרים כלל בדראפט), ועל כן, שגיאה גדולה בחיזוי הנתונים עבורם אינה קריטית לבחינת הביצועים. **עם זאת, שגיאה כזו תפגע בצורה ניכרת בשגיאה הממוצעת, אך פחות תבוא לידי ביטוי בשגיאה החציונית.**

**כלל המודלים נבחנו בשתי שיטות – Test Set ו-Cross Validation**, במטרה להפיק תוצאות אמינות שאינן תלויות בתת קבוצת אימון ספציפית שנבחרת באקראי, כאשר **עיקר ההשוואה בין המודלים התבצעה על סמך תוצאות ניסויי ה-Cross Validation**, כיוון שהינם אמינים יותר.

הערכת השגיאה (על פי המטריקות שצויינו לעיל) התבצעה בשני אופנים –

1. **הערכת שגיאה הכוללת** – שיטת הערכה זו נותנת תמונה חלקית ביותר, מספקת "מבט על" בלבד על רמת הביצועים של המודל, לבחינה וסינון ראשוניים אל מול המודלים האחרים.

2. **הערכת השגיאה לפי עשירונים (Buckets)** – כפי שתואר, ישנו הבדל רב בחשיבות בין מרחק השגיאה בחיזוי שחקנים הנמצאים גבוה בדירוג הכללי לבין אלו הנמצאים במיקום נמוך בדירוג. ההשפעה של שגיאה בחיזוי הסטטיסטיקה של שחקנים "טובים" הינה קריטית, ועל כן, מטרתנו הינה לפתח מערכת שמשיגה שגיאה נמוכה ככל הניתן עבור העשירונים הגבוהים. **החלוקה ל-Buckets התבצעה באופן הבא** – לכל קטגוריית חיזוי, מוינו השחקנים על פי הישגיהם האמיתיים בקטגוריה בעונה המשמשת לאפיון אובייקט השחקן, וחולקו ל-Buckets בגדלים שווים (ככל הניתן) על פי מיון זה. לאחר השגת תוצאות החיזוי, חולקה קבוצת המבחן לפי שייכות כל שחקן ל-Bucket המתאים, וכך חושבו מטריקות השגיאה עבור כל Bucket. חשוב לציין כי בתהליך ה-Cross Validation התבצעה החלוקה המתוארת לעיל בכל איטרציה, ובסופה חושבו תוצאות כל Bucket. בסיום התהליך, חושב ממוצע המטריקה בכל Bucket על פני כלל האיטרציות.

בכדי לאמוד באופן מוחלט את איכות תוצאות מודלי החיזוי הנבחרים, **התבצעה השוואה בין מרחק השגיאה שהתקבל מהמודלים הנבחרים, אל מול מרחק השגיאה של חיזוי מומחי ליגת ה-NBA ומשחק הדראפט**. גם שגיאת המומחים חושבה על פי חלוקה ל-Buckets באותו אופן שתואר עבור תוצאות מערכות החיזוי. השוואה זו מספקת מדד אמין לאיכות התוצאות, שכן, חיזויי המומחים שנבחנו משמשים באופן נרחב את משתתפי משחק הפנטזי בעולם האמיתי. למעשה, מטרתנו הייתה לפתח מערכת אשר תספק חלופה לחיזוי המומחים, באמצעות מודלי הלמידה.

## שלב ראשון – סינון ראשוני של מודלי חיזוי מובילים

בשלב זה ביצענו מספר רב של ניסויים באופן שיטתי, לבחינת כל אחד מהמודלים שפורטו. כל מודל חיזוי הוגדר עם קונפיגורציית פרמטרים קבועה עבור שלב זה. את פירוט המודלים שנבחנו בשלב זה ניתן לראות בפרק "אלגוריתמי למידה לשלב החיזוי" תחת פיתרון בעיית החיזוי. עבור מודלים אלו נעשה שימוש בקונפיגורציית הפרמטרים הדיפולטיבית של כל אחד מהם.

**כל ניסוי בחלק זה הוגדר לפי – קטגוריית חיזוי, סוג מודל החיזוי, שיטת בחירת התכונות ומספר התכונות המבוקש.** לדוגמא, ניסוי לחיזוי קטגוריית הנקודות, שבחן את מודל ה-Random Forest, עם 40 תכונות הנבחרות בשיטת Select K Best. באופן זה, הורצו הניסויים על כל הקומבינציות, כאשר כמות התכונות נעה בין 10 ל-100 בקפיצות של 10, מסיבות פרקטיות (זמן ביצוע, הימנעות מביצוע חיפוש ממצה) ומתוך הנחה שלא יתקבל הבדל ניכר בתוצאות עבור שינוי תכונות בודדות בלבד. שיטות בחירת התכונות נלקחו מתוך שלושת האפשרויות שתוארו בתת הפרק המתאר את השיטות לסינון התכונות – Select K Best, Embedded ו-Wrapper.

### הפלט הרצוי אשר התקבל מכל ניסוי הינו –

- **מרחק השגיאה לכל מטריקה** – כללית ולפי עשירונים.
- **גרפי תוצאות** – היסטוגרמת מרחק השגיאה, מפת חום המתארת את תוצאות החיזוי אל מול ערך האמת
- **סט התכונות הנבחרות**

בנוסף, עבור כל קטגוריית חיזוי, התקבלה כפלט טבלת תוצאות כוללת, אשר הכילה את תוצאות השגיאה הכללית (מניסוי ה-Cross Validation) בכל ניסוי, באמצעותה ניתן לבחון במבט על את ביצועי המודלים ולסנן את המודלים העדיפים ביותר, לצורך בחינה מעמיקה.

## שלב שני – אופטימיזציית פרמטרים ובחירת מודל לכל קטגוריית חיזוי

**בשלב זה נבחרו מודלי החיזוי אשר השיגו את התוצאות הטובות ביותר בחלק הראשון. המודלים נבחרו על פי שיקולים שונים** – התייחסות לתוצאות המטריקות השונות (שגיאה נמוכה), בחינה ידנית של היסטוגרמת השגיאה (התפלגות ובחינת השגיאה המקסימלית), התחשבות בגודל סט התכונות ובנוסף, שאיפה לבחור סט מגוון של מודלים, מתוך הנחה שלאחר אופטימיזציית הפרמטרים, ייתכנו שיפורים משמעותיים במודלים מסויימים.

**אופטימיזציית הפרמטרים התבצעה בשיטת Grid Search** באמצעות הכלי "GridSearchCV" המסופק כחלק מחבילת בחירת מודל של סקיפריית sklearn. **הניסויים בחלק זה הוגדרו לפי – מודל חיזוי מסויים, מספר תכונות** (זהה לזה שעבורו השיג המודל את התוצאות לפיהן סונן בחלק הראשון) **וסט של ערכים עבור חלק מהפרמטרים המגדירים את המודל, עליהם יבוצע החיפוש.** לדוגמא, ניסוי לחיזוי קטגוריית הנקודות באמצעות מודל Random Forest עם 20 פרמטרים הנבחרים בשיטת Embedded על פני מרחב

הפרמטרים הבא – "מספר עצים בועדה" – [5, 10, 15, 20] ו"מספר דוגמאות מינימלי בעלה" – [1, 2, 3, 4, 5]. כלל הניסויים שבוצעו מפורטים בנספח ב'.

בשלב ה-Grid Search התייחסנו רק למדדי השגיאה הכוללת (ממוצעת וחציונית), זאת כיוון שלאחר בחינה ראשונית, הסתבר כי התוצאות היחסיות בין ה-Buckets השונים כמעט ואינן משתנות כתלות בפרמטרים הניתנים למודל.

לאחר ביצוע טיוב הפרמטרים עבור המודלים הנבחרים לכל קטגוריית חיזוי, **התבצע היזון חוזר – ע"י הרצה חוזרת של הניסויים מחלק 1 עבור המודלים עם הפרמטרים המטויבים**. בשלב זה התקבלו כפלט אותם תוצרים שתוארו בחלק 1, לפיהם התבצעה השוואה באופן ידני (כיוון שמדובר במספר מצומצם של ניסויים). המודלים הנבחרים לכל קטגוריה נבחרו בעיקר בהסתמך על ההישיגים ב-Buckets החשובים.

## שלב 1 – תוצאות הניסויים – ניתוח ומסקנות

בחלק זה התבצעה הרצה של מספר רב מאוד של ניסויים, על כן, **נציג מדגם של תוצאות מעניינות הממחישות את התהליך ואת אופן השימוש בהן לצורך סינון מודלי החיזוי**. ניתן למצוא את כלל התוצאות של הניסויים משלב זה בתיקייה "AI-Project-NBA-Fantasy\Scripts\Experiments\_Results" תחת הפרויקט שהוגש. את המודלים שנבחרו כתוצאה משלב זה, יחד עם חלק מהתוצאות שהשיגו, ניתן למצוא תחת נספח א'.

### תוצאות ניסויים לדוגמא – חיזוי קטגוריית האסיסטים (AST)

#### 1. טבלת התוצאות הכוללת עבור קטגוריית אסיסטים –

א. נציג מספר תוצאות מובילות (20) מתוך הטבלה, על פי מדדי השגיאה הממוצעת והשגיאה החציונית<sup>9</sup> –

מיון לפי שגיאה ממוצעת					מיון לפי שגיאה חציונית				
Model Name	Selector	K	Avg Error	Med Error	Model Name	Selector	K	Avg Error	Med Error
BayesianRidge	Wrapper	60	0.564	0.390	RandomForestRegressor	Embed	30	0.576	0.378
MLPRegressor	Embed	90	0.564	0.392	SVR	Embed	40	0.599	0.380
BayesianRidge	Wrapper	70	0.565	0.392	SVR	Embed	30	0.591	0.383
BayesianRidge	Wrapper	40	0.565	0.392	MLPRegressor	SelectKBest	80	0.578	0.383
BayesianRidge	Wrapper	90	0.565	0.391	RandomForestRegressor	Wrapper	20	0.581	0.384
BayesianRidge	Wrapper	80	0.565	0.392	RandomForestRegressor	Embed	40	0.583	0.384
MLPRegressor	SelectKBest	100	0.565	0.390	RandomForestRegressor	Wrapper	30	0.583	0.384
BayesianRidge	Wrapper	50	0.566	0.391	MLPRegressor	SelectKBest	90	0.567	0.385

<sup>9</sup> המודלים המסומנים בצהוב הינם המודלים עבורם מוצגת בחינת התוצרים המעמיקה בסעיף הבא.

BayesianRidge	Embed	80	0.566	0.392	RandomForestRegressor	SelectKBest	80	0.587	0.385
BayesianRidge	Wrapper	20	0.566	0.391	MLPRegressor	SelectKBest	30	0.578	0.385
BayesianRidge	Wrapper	100	0.566	0.390	MLPRegressor	SelectKBest	50	0.574	0.385
MLPRegressor	Embed	100	0.566	0.390	LinearRegression	SelectKBest	30	0.575	0.386
BayesianRidge	Embed	90	0.566	0.394	RandomForestRegressor	Wrapper	100	0.578	0.386
BayesianRidge	Embed	100	0.566	0.391	MLPRegressor	SelectKBest	60	0.577	0.386
BayesianRidge	Wrapper	30	0.566	0.394	RandomForestRegressor	Embed	90	0.580	0.387
BayesianRidge	Embed	60	0.567	0.393	BayesianRidge	SelectKBest	50	0.575	0.387
MLPRegressor	SelectKBest	90	0.567	0.385	BayesianRidge	SelectKBest	60	0.574	0.387
BayesianRidge	Embed	70	0.567	0.395	RandomForestRegressor	Wrapper	40	0.578	0.388
LinearRegression	Embed	30	0.567	0.393	RandomForestRegressor	SelectKBest	90	0.587	0.388

ב. ניתן לראות בטבלאות אלו את הגיוון בסוגי המודלים ומספר התכונות המשמש לחיזוי. כמו כן, ניתן להבחין כי מתקבלים מודלים מובילים שונים על פי כל אחת מהמטריקות, פרט אשר היווה מורכבות מסויימת בסינון ובחירת המודלים העדיפים.

ג. בנוסף, ניתן להבחין בבירור כי לא קיים הבדל רב בין מרחקי השגיאות בכל אחת מהמטריקות בין המודלים השונים (הבדלים בסדר גודל של אלפית יחידת השגיאה), כלומר, לא התקבל מודל חיזוי אשר תוצאותיו בלטו באופן ניכר על פני האחרים. חשוב לציין כי מרחק השגיאה הינו ביחס ליחידות הקטגוריה הנחזית, ולכן עבור כל קטגוריה יתקבל סדר גודל שגיאה שונה. למשל בדוגמא הנ"ל (אסיסטים), טווח הערכים של הקטגוריה נע בין 0 לכ-12, ובהתאם מרחק השגיאה הממוצע הינו בסדר הגודל המוצג, בעוד בקטגוריית הנקודות (שערך המקסימום בה הוא כ-30), השגיאה הממוצעת תהיה גבוהה יותר.

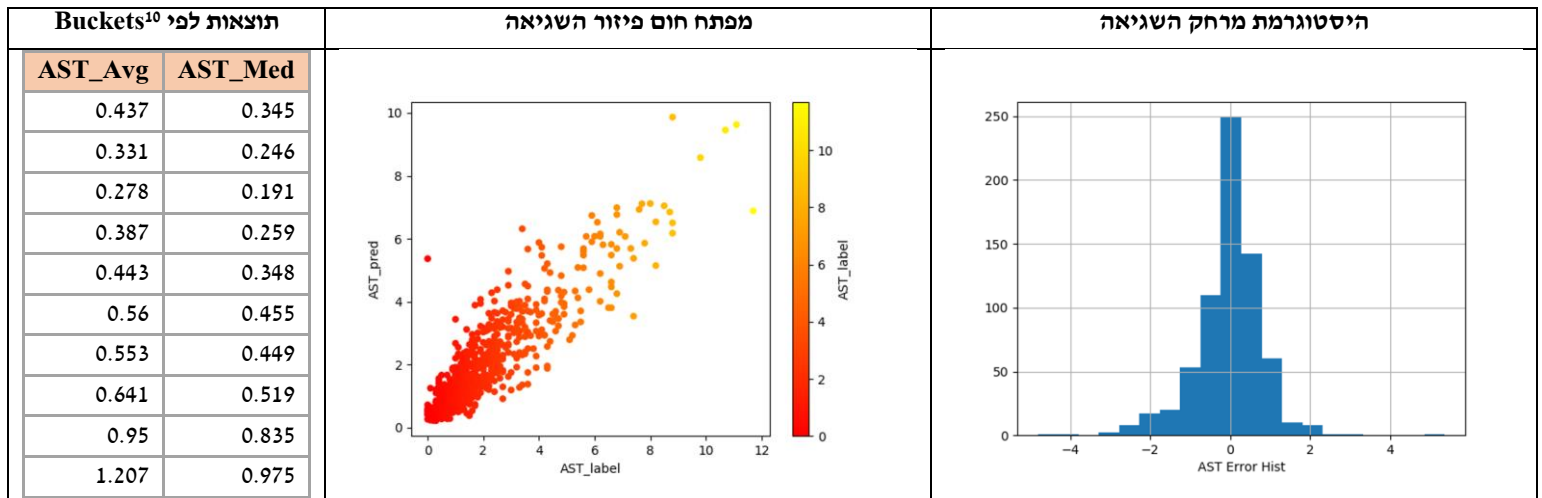
## 2. תוצרים נוספים לבחינה מעמיקה של התוצאות –

א. כאמור, התוצאות המוצגות בטבלה הכוללת מעלים מגוון רב של מודלים אשר ההבדל בתוצאותיהם אינו ניכר. על כן, נדרשת בחינה מעמיקה של התוצאות של מספר מודלים נבחרים מתוך הטבלה. בחינה זו התבצעה באמצעות הסתכלות על התוצרים הנוספים – מדד שגיאה על פי Buckets, גרף היסטוגרמת מרחק השגיאה ומפת החום של תוצאות החיזוי אל מול תוצאות האמת.

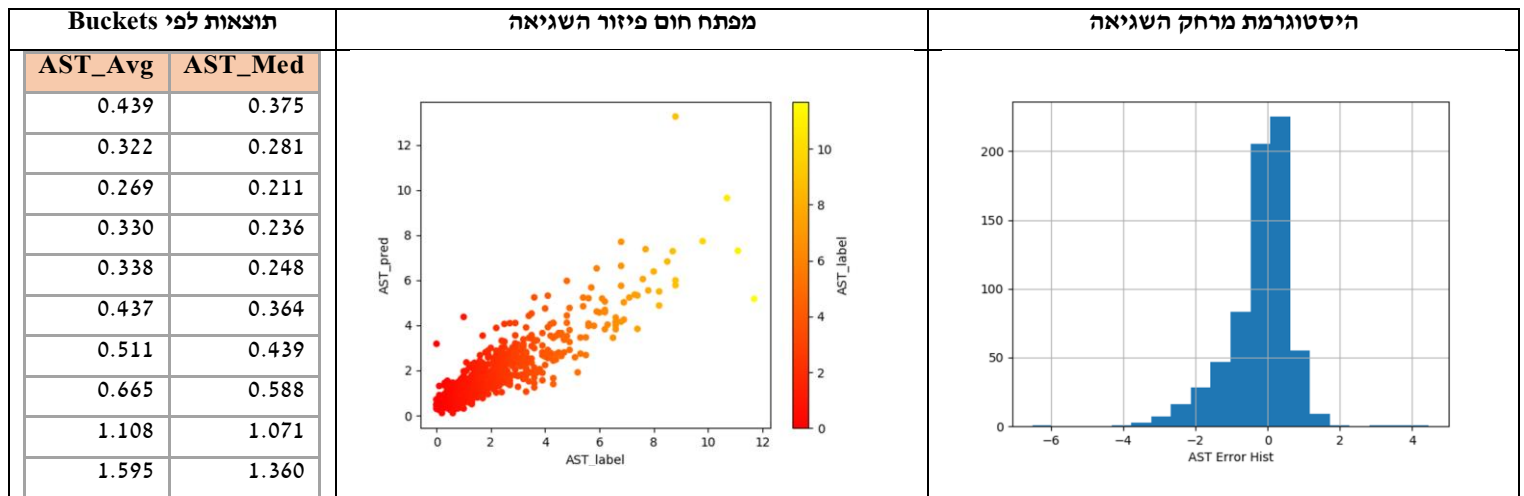


ב. להלן דוגמאות נבחרות המציגות תוצרים אלו –

i. Random Forest, 30 תכונות, בשיטת בחירה Embedded.



ii. SVR, 40 תכונות, בשיטת בחירה Embedded.



ג. ניתן לראות כי בין שני הניסויים ישנו הבדל משמעותי במרחק השגיאה המתקבל בעשירונים

העליונים (השורות התחתונות בטבלה). מודל ה-Random Forest משיג תוצאות טובות

בהרבה עבור עשירונים אלו, נתון משמעותי בהתחשב בהבדלים המזעריים בשגיאה הכוללת בין

שני המודלים. השיוויון היחסי שמתקבל בשגיאה הכוללת נובע מכך שבעשירונים האמצעיים

<sup>10</sup> השורה התחתונה מייצגת את העשירון העליון, כלומר, השחקנים הטובים ביותר בקטגוריית האסיסטים.

מודל ה-SVR "עוקף" את מודל ה-Random Forest, אך לעשירונים אלו משמעות פחותה בהרבה.

ד. במפות החום אנו שואפים להשיג גרף קרוב ככל הניתן לגרף  $f(x) = x$ , כלומר, שמרחק השגיאה יהיה אפסי ככל הניתן לכלל האובייקטים בקבוצת המבחן. גם במקרה זה ישנה חשיבות גבוהה יותר לאובייקטים שתוצאות האמת שלהם נמצאות בעשירונים העליונים (הנקודות הגבוהות בגרף). במקרה זה לא קיים הבדל ניכר בין מפות החום של שני המודלים.

ה. היסטוגרמת מרחק השגיאה מאפשרת לראות בצורה נוחה וברורה את פיזור השגיאה על פני כלל האובייקטים בקבוצת המבחן. כלומר, ניתן להבחין כי עבור מרבית האובייקטים מרחק השגיאה נע סביב הערך 0, בעוד ישנו מספר מצומצם יחסית של אובייקטים אשר שגיאת החיזוי שלהם גבוהה. מבחינת ההיסטוגרמות של שני המודלים, ניתן לראות כי מודל ה-Random Forest יש רוב של אובייקטים עם שגיאה כמעט אפסית (אל מול מודל ה-SVR), וכמו כן, השגיאה המקסימלית במודל ה-SVR גבוהה בהרבה (כ-6-) מזו של מודל ה-Random Forest (כ-4.5). מדד זה הינו משמעותי להבחנה בין המדלים בשלב זה, כיוון ששגיאה גבוהה באובייקטי הקיצון (נקודות רחוקות מהקו הליניארי במפת החום) עשויה להשפיע בצורה קיצונית על בחירה / אי בחירה של שחקן בדראפט.

## ניתוח ומסקנות

בשלב זה סוננו מודלי החיזוי המובילים עבור כל קטגוריה, יחד עם סט תכונות מועדף. את פירוט המודלים הנבחרים ניתן כאמור לראות בנספח ב'. נפרט נקודות מרכזיות ומעניינות שעלו במהלך בחינת תוצאות הניסויים בשלב זה:

1. בחינת ההישגים הכלליים של המודלים השונים – מבחינת תוצאות הניסויים בשלה זה ניתן לראות כי עבור מרבית קטגוריות החיזוי הושגו תוצאות טובות מהמצופה. ניתן לראות זאת מהשוואה בסיסית אל מול תוצאות חיזוי המומחים שהוזכרו בשלב המתודולוגיה הניסויית. ניכר כי אוסף התכונות שנבחרו עבור ייצוג אובייקט מתאים עבור מערכת לומדת המבצעת ריגרסיה, ומאפשר השגת תוצאות עם שגיאה העולה בקנה אחד עם זו של מומחה אנושי.

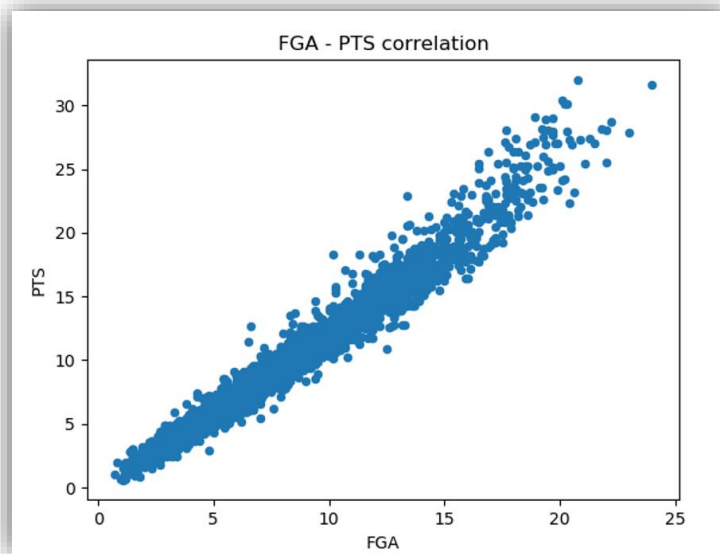
2. הצורך בבחינה מעמיקה של כלל התוצרים – סינון המודלים הנבחרים דרש בחינה מעמיקה של כלל התוצרים על פני כלל המטריקות – התבוננות בטבלת השגיאה הכללית אינה מספקת, שכן ההבדלים בין המודלים המובילים מזערניים יחסית. בנוסף, התייחסות למטריקה בודדת גם היא אינה מספקת ואף עשויה להוביל לבחירת מודל מוטעה. החלוקה לשגיאה על פי עשירונים יחד עם ההיסטוגרמות ומפות החום איפשרו סימון הבדלים קטנים אך משמעותיים בין תוצאות המודלים השונים והיוו נידבך קריטי בבחירת המודלים לשלב הניסויים השני.

3. **סינון מיידי של מודלים ספציפיים** – מתוצאות הטבלה הכללית ניתן להבחין מיידית במודלי חיזוי אשר לא התאימו לבעיה הספציפית (בכלל הקטגוריות), דוגמאות מודל Elastic Net ומודל Ada Boost, אשר השיגו מרחק שגיאה גדול באופן מוחלט, עבור סט תכונות מכל גודל שנבחר. מודלים אלו הוסרו מיידית ולא נלקחו בחשבון עבור שלב הניסויים השני.

#### 4. **ניתוח תהליך ה-Feature Selection**

א. **שימוש בקטגוריית החיזוי בתכונה נבחרת** – ערכי הקטגוריה הנחזית באובייקט הנחזה (השיגיו בעונה הקודמת לעונת החיזוי, מהווים תכונה באובייקט הנחזה) מופיעים תמיד בין סט התכונות הנבחרות וככל הנראה היו בעלי השפעה גבוהה על התוצאות.

ב. **תכונות עם קורלציה גבוהה לקטגוריה הנחזית** – ניכר כי לכל קטגוריית חיזוי, מרבית התכונות שנבחרו בתהליך ה-Feature Selection הינן בעלות קורלציה לקטגוריה הנחזית. לדוגמא, עבור קטגוריית הנקודות, בניסויים שהשיגו את התוצאות הטובות ביותר, נבחרו תכונות כגון FG, FGA, FT% – אשר להן קורלציה חיובית גבוהה עם קטגוריית הנקודות, כפי שניתן לראות בתרשים הבא –



איור 8 : גרף קורלציה בין תכונה FGA לקטגוריה PTS

ג. **תכונות לא סטטיסטיות שתורמו לחיזוי** – עבור חיזוי מרבית הקטגוריות התבצעה בחירה של תכונות לא סטטיסטיות, כגון Age ו-Salary. ניתן ללמוד מכך כי מתכונות אלו אכן ניתן להסיק במידה מסויימת על "איכות" השחקן, כפי שצפינו בעת איסוף המידע והרכבת האובייקט.

ד. **בחירת תכונות באמצעות אלגוריתם Select K Best** – ניתן לראות כי באופן גורף, בניסויים אשר בהם הושגו תוצאות טובות, נעשה שימוש באחת מהשיטות Embedded או Wrapper לבחירת התכונות, בעוד ששיטת Select K Best כשלה בבחירת תכונות אשר יטיבו עם תוצאות החיזוי.

5. **מודל ה-Linear Regression** – במקור, בעת בניית מתודולוגיית הניסויים, שיערנו כי יהיה צורך במודל מורכב בכדי לחזות את רוב הקטגוריות ברמה טובה, אך הכנסנו את מודל ה-Linear Regression בתור מודל ביקורת. אף על פי כן, **ניתן לראות בתוצאות כי מודל זה הגיע לביצועים טובים עבור חיזוי מרבית הקטגוריות**, זאת גם בהשוואה למודלים מורכבים יותר. ניתן להסיק כי **בעיית החיזוי עבור קטגוריות מסויימות מאופיינת ע"י קשרים לינאריים עם תכונות האובייקט**. העובדה שמדובר במודל פשוט יחסית מהווה יתרון, כיוון שמעידה על התאמתו של המודל הנלמד לבעיה הכללית, ועל כן **סביר שתשיג תוצאות טובות חיזוי קבוצת מבחן כללית** (למשל, עבור חיזוי עונות עתידיות).

6. **חיזוי קטגוריית ה-Fantasy Rank** – ניכר כי קטגוריה זו הינה קשה במיוחד לחיזוי ביחס לשאר הקטגוריות. ניתן להסביר זאת בכך שמדד זה הינו מורכב יותר מקטגוריה סטטיסטית פשוטה, אשר מושפע ממספר רב של משתנים, אשר לחלקם אף לא יכולנו להתייחס (כגון פציעות שחקנים). עם זאת, התוצאות שהושגו עבור קטגוריה זו מאפשרים לייצר דירוג בין השחקנים השונים עבור משחק הפנטזי (כלומר, להכריע באופן די מוחלט האם שחקן X עדיף משחק Y).

7. **הבדלים במודלים בין הקטגוריות השונות** – ניתן ללמוד כי כל קטגוריה עומדת בפני עצמה, וכי **לא ניתן לסמן מודל ספציפי או אוסף תכונות ספציפי אשר יתאימו לחיזוי אפקטיבי של כלל הקטגוריות**.

## **שלב 2 – תוצאות הניסויים – ניתוח ומסקנות**

בשלב הניסויים הראשון, נבחנו מודלים עבורם הוגדרה קונפיגורציה קבועה ובסיסית של פרמטרים. **השימוש בקונפיגורציה בסיסית נבע מכך שלא ניתן היה לשער אילו ערכים לפרמטרים מתאימים לבעיה הנתונה וכיצד שינוי כל פרמטר ישפיע על הביצועים**. נוסף לכך, עקב מגבלות זמן ריצה ומספר ניסויים רב בשלב זה, הוחלט להריץ כל מודל עם קונפיגורציה בודדת.

בשלב השני התבצעה אופטימיזציה לפרמטרים עבור מודלים נבחרים משלב 1. **בשלב זה הורצו ניסויים בודדים במסגרתם התבצע חיפוש במרחב הפרמטרים של המודלים**. את תוצאות תהליך ה-Grid Search ניתן למצוא בתיקיה "AI-Project-NBA-Fantasy\Scripts\Experiments\_Results\0\_Grid\_Search" בפרויקט שהוגש. באמצעות תוצאות אלו סומנו הפרמטרים האופטימליים עבור כל אחד מהמודלים הנבחרים (בכל קטגוריה). **תוצרים לדוגמא מתהליך זה מוצגים בסעיף הבא**.

כמו כן, המודלים המטויבים נבחנו שוב במסגרת הניסויים משלב 1 ותוצאותיהם, יחד עם הצגת המודל הנבחר לכל קטגוריה, מובאים בסיכום פרק זה.

## דוגמא לביצוע Grid Search במרחב הפרמטרים

1. להלן דוגמא לתוצר המתקבל מהרצת Grid Search עבור מודל חיזוי Random Forest עם 20 תכונות שנבחרו בשיטת Select K Best, לצורך חיזוי קטגוריית 3P. התוצר המתקבל מתהליך זה הינו טבלה של כל קומבינציות ערכי הפרמטרים (שהועברו כקלט לאלגוריתם), ודירוג ההישג שלהם על פי מטריקה מוגדרת מראש.

2. עבור המודל בדוגמא זו התבצעה אופטימיזציה לפרמטרים עם הערכים הבאים :

פרמטר	מרחב הערכים	תיאור הפרמטר
<b>n_estimators</b>	[5, 10, 15, 20]	מספר העצים בועדה
<b>criterion</b>	['mse', 'mae']	המטריקה לפיה מחושבת השגיאה
<b>min_samples_split</b>	[2-7]	מספר דוגמאות מינימלי לפיצול צומת פנימי
<b>min_samples_leaf</b>	[1-5]	מספר דוגמאות מינימלי בעלה

3. תוצאות התהליך (עשרת הקונפיגורציות המובילות) :

n_estimators	criterion	min_samples_split	min_samples_leaf	mean_test_score	rank_test_score
20	mae	2	4	-0.2629	1
20	mae	5	5	-0.2631	2
15	mae	3	5	-0.2635	3
20	mae	6	4	-0.2635	4
15	mae	2	4	-0.2635	5
20	mae	4	5	-0.2641	6
20	mae	3	5	-0.2643	7
20	mae	2	3	-0.2643	8
20	mae	4	3	-0.2643	9
15	mae	3	4	-0.2644	10

4. לפי תוצאות אלו ניתן לראות בבירור כי למספר גבוה של משתתפים בועדה יש השפעה חיובית על התוצאות. כמו כן, הערכת השגיאה על פי מדד Mean Absolute Error עדיפה גם כן. נבחין כי הערכים העדיפים שונים מערכי הקונפיגורציה הבסיסית שניתנה למודל זה בשלב 1 של הניסויים. **טבלאות דומות התקבלו עבור כלל המודלים בכל הקטגוריות השונות, ולפיהן נבחרו קונדפיגורציות סופיות עבור שלב ההיזון החוזר.**

5. מתוצאות הניסויים לעיל (וממספר ניסויים נוספים בשלב זה) ניתן לראות תימוכין לעיקרון הנלמד בקורס המבוא, לפיו יש להגביל את מורכבות המודל על מנת למנוע Over Fitting. בדוגמא הנ"ל נוכחנו לראות כי מודל Random Forest מגיע לביצועים טובים יותר כאשר עומק העצים בועדה מוגבל – נתון המושפע ע"י הפרמטר min\_sample\_leaf, אשר הינו גבוהה בניסויים שהשיגו את התוצאות הטובות ביותר.

## ניתוח ומסקנות

1. שיפור התוצאות כתלות בפרמטרים – ככלל, רוב המודלים (בכל הקטגוריות) לא הציגו שיפור משמעותי בתוצאות כאשר נבחנו עם קונפיגורציות פרמטרים שהתקבלו מתהליך ה-Grid Search. לעיתים קרובות השונות בין הקונפיגורציה שהשיגה את התוצאות הטובות ביותר, לבין הגרועה ביותר, הייתה מינורית. נתונים אלו תומכים במתודולוגיית הניסויים שהוגדרה, ובהחלטה לפיה אין צורך לבצע חיפוש ממצה לצורך טיוב הפרמטרים כבר בשלב הסינון הראשוני. עם זאת, מספר מודלים כן השיגו שיפור שאינו זניח במטריקות השגיאה, הן הכוללות ובפרט בחלוקה לעשירונים, ועל כן, תועדפו אל מול המודלים עם הקונפיגורציות הבסיסיות.

2. השמטת מודלים כתוצאה מזמן ריצה ארוך – מודלים בודדים שנבחרו עבור השלב השני לא הורצו במסגרת שלב טיוב הפרמטרים, כיוון שהרצתם ארכה זמן רב מאוד, וניכר כי השיפור בתוצאות תחת קונפיגורציות שונות הינו זניח.

## תוצאות ומסקנות סופיות למערכת החיזוי

1. להלן המודלים הסופיים שנבחרו עבור חיזוי כל קטגוריה. מודלים אלו, יחד עם קונפיגורציות הפרמטרים והרכב התכונות הנבחר מהווים את המערכת מערכת החיזוי הסופית, שהינה התוצר של חלק 1 של הפרויקט. לפירוט התוצרים המלא (גרפים, מדדי שגיאה לפי Buckets) לכל אחד מהמודלים הנבחרים, ראו נספח ג'.

Category	Selected Model	Parameters	Number of Features	Selector	Avg Error	Med Error	Experts Avg Error	Experts Med Error
PTS	Ridge	alpha=1	40	Wrapper	2.225	1.76	2.429	1.99
AST	Linear Regression	None	30	Embedded	0.567	0.393	0.617	0.53
TRB	Ridge	alpha=1	40	Wrapper	0.891	0.644	0.930	0.715
STL	Ridge	alpha = 0.25	40	Wrapper	0.189	0.15	0.211	0.178
BLK	Random Forest	n_estimators = 20 Criterion = 'mae' Min_sample_split = 5 Min_samples_leaf = 2	80	Wrapper	0.152	0.101	0.184	0.15
3P	Random Forest	n_estimators=20 criterion='mae' min_samples_split=2 min_samples_leaf=4	30	Embedded	0.263	0.179	0.358	0.306
FG%	Linear Regression	None	20	Embedded	0.041	0.0271	0.039	0.03
FT%	Random Forest	n_estimators=20 criterion='mae' min_samples_split=2 min_samples_leaf=5	40	Embedded	0.0672	0.0451	0.076	0.05
TOV	SVR	Kernel = 'rbf' Degree = 1 C = 10	60	Embedded	0.305	0.243	0.315	0.27
Fantasy Rank	Ridge	alpha=0.25	30	Wrapper	66.04	54.16	70.575	56

2. **התייחסות כללית לתוצאות הסופיות** – מהטבלה המוצגת לעיל ומפירוט התוצאות הסופיות המופיע בנספח ד', ניתן לראות כי המודלים הנבחרים משיגים תוצאות טובות, גם בהשוואה אובייקטיבית אל מול שגיאת המומחים. במרבית הקטגוריות אף התקבלה שגיאה כוללת (ממוצעת וחציונית) טובה מזו של מומחי הפנטזי. עם זאת, חשוב לציין כי בהשוואה מעמיקה של מדדי השגיאות על פי חלוקה לעשירונים, אותה ניתן למצוא בתיקיה –

"AI-Project-NBA-Fantasy\Data\Experts\experts\_error.csv" תחת הפרויקט, עולה כי עבור מספר קטגוריות, השגיאה שהשיגו המודלים במערכת שלנו הינה גבוהה יותר דווקא בעשירונים העליונים, לעומת שגיאת המומחים.

אף על פי כן, התוצאות מעידות כי למערכת החיזוי שפיתחנו בחלק זה ישנה היכולת לספק חיזוי אמין ואפקטיבי עבור משחק הפנטזי.

## חלק 2: סוכן למשחק דראפט הפנטזי

במשחק הדראפט מתחרות מספר קבוצות (סוכנים) ביניהן במטרה להרכיב קבוצה אופטימלית (ככל הניתן) מתוך מאגר שחקנים (שחקנים פעילים בליגת ה-NBA) ותחת אילוצים שונים. בחלק זה נעשה שימוש בשיטות ואלגוריתמים שונים מתחום החיפוש במרחב מצבים במשחק רב סוכנים, שנלמדו מקורס המבוא.

**מטרת העל הינה בניית סוכן בעל אסטרטגיה להרכבת הקבוצה האופטימלית בהתחשב באילוצים הקיימים בכל בחירה בשלב המשחק.** נתאר את דרכי הפיתרון והשיקולים שנלקחו במהלך פיתוח הסוכן, וכמו כן, את מבנה המערכת, והתוצאות לפיהן ניתן לאמוד את ביצועי כל אחד מהסוכנים שפותחו, ולהתאימם לגירסאות שונות של משחק הדראפט.

## תיאור פתרון הבעיה

### חוקי משחק הדראפט

**במשחק הדראפט על כל סוכן (מנהל קבוצת פנטזי) להרכיב קבוצת שחקנים (שחקני NBA) בגודל נתון מראש.** בחירות השחקנים נערכות על פי סדר בחירות המוגרל ונתון מראש, המתנהל בסיבובים ובצורת "נחש". על כל סוכן המשתתף בליגת הפנטזי לבחור מספר זהה של שחקנים ובעמדות המתאימות מתוך מאגר השחקנים הקיים (כפי שהוגדר בחלק "חוקי משחק הדראפט").

לאחר הדראפט, הקבוצות ייתחרו ביניהן במהלך עונת ה-NBA ב-matchups שבועיים ב-9 קטגוריות ניקוד, כאשר בכל שבוע, הקבוצה שהובילה במספר הקטגוריות בהן צברה ניקוד גבוה יותר (על פי ביצועי שחקני ה-NBA באותו שבוע), תוכרע במנצחת ב-matchup. הגדרות אלו בעלות השפעה ישירה על שיטות החיפוש שהוגדרו לסוכים השונים ועל האופן בו נמדדת הצלחתם במשחק.

כמו כן, חשוב להזכיר כי כיוון שמשחק הפנטזי הוא משחק דינאמי שמתפרש על פני עונה שלמה, וכולל רבדים נוספים מלבד הרכבת הקבוצה בדראפט, **לא קיים מדד מוחלט למדידת טיב הקבוצה ולקביעה האם הקבוצה היא אופטימלית.** מעתה והלאה נתייחס לקבוצה אופטימלית ביחס אל המדדים שקבענו ויפורטו בהמשך.

## הנחות מקלות

כיוון שמשחק הדראפט הינו משחק מורכב המושפע מגורמים רבים, **לצורך התאמתו למסגרת הפרויקט בוצעו מספר הנחות מקלות,** אשר אינן משפיעות בצורה מהותית על אופי וקושי המשחק:

1. **שחקנים פצועים** – פציעה של שחקן NBA עשויה להשפיע על ההחלטתה לבחור אותו בדראפט במקום מסויים. לצורך סינון שחקנים פצועים נדרש מאגר מידע עדכני המעריך את חומרת הפציעה



וזמן ההיעדרות ממשחק. עקב המורכבות הרבה של התאמת מאגר נתונים כזה, הוחלט להתעלם מפציעות שחקנים. כמו כן חשוב לציין כי כיוון שכלל הסוכנים בסימולציות המשחק שערכנו אינם מודעים לנתוני הפציעות, אי הסינון אינו משפיע על תוצאות סוכן כזה או אחר.

2. **שחקנים בשנתם הראשונה (Rookies)** – מאגר השחקנים והחיזוי שניתן עבורם נבנה ע"י התחשבות ביצועיהם בעונה הקודמת לעונת המשחק. מסיבה זו, לא קיימות נתונים על ביצועיהם של שחקנים בשנתם הראשונה ועל כן לא קיים עבורם חיזוי. מכאן שחקנים אלו אינם מופיעים במאגר והסוכן לא יבחר בהם.

3. **שחקנים המשחקים במגוון עמדות** – כחלק מהרכבת הקבוצה, על הסוכן להתחשב באילוצי עמדות. לשחקן אשר יכול לשחק ביותר מעמדה אחת, בחרנו עמדה יחידה שתוגדר עבורו לצורך המשחק. למשל עבור שחקן המשחק בעמדות PG, SG נבחרה למשל עמדת ה-PG, כלומר, אם ייבחר ע"י הסוכן, הוא יתפוס את אילוץ עמדת ה-PG בקבוצתו.

## **שימוש בנתוני החיזוי**

לצורך קבלת החלטות במהלך משחק הדראפט, נדרשת הערכה כלשהי של הישיגהם הסטטיסטיים הצפויים של שחקני ה-NBA לעונת המשחק. **בחלק זה נסתמך על תוצאות מערכת החיזוי שנבנתה בחלק 1 של הפרויקט.** על אף שקיימים חיזויים נוספים המסופקים ע"י מומחים באתרי משחק הפנטזי השונים, הוחלט להשתמש בתוצאות המערכת שלנו, כיוון שתוצאותיה הממוצעות אינן נופלות מתוצאות החיזוי של המומחים (מבחינת מרחק השגיאה), כפי שניתן לראות בפירוט התוצאות בחלק 1. כמו כן, במסגרת הפרויקט רצינו לספק מערכת שלמה מקצה לקצה, שתוכל לחזות את ביצועי השחקן ובעזרת החיזוי להרכיב קבוצה אופטימלית.

נתוני החיזוי שהתקבלו מהמערכת הלומדת שימשו את כלל הסוכנים שנבחנו בשלבי המשחק. עם זאת, בכדי לאמוד באופן אמין ככל שניתן את הצלחת הסוכן בהרכת הקבוצה, יש צורך להשתמש בתוצאות אמת של שחקנים בעונה נתונה ולא להסתמך על החיזוי גם לצורך בחינת התוצאות. על כן, השתמשנו בחיזוי של המערכת הלומדת לעונה 2018/19 (העונה האחרונה, עבורה קיימים נתונים סופיים אמיתיים). באופן זה, איפשרנו לסוכנים לקבל החלטות על סמך הנתונים החזויים, והשתמשנו בתוצאות האמיתיות של השחקנים מאותה עונה על מנת להשוואות בין הקבוצות של הסוכנים ולבחון איזה סוכן הרכיב את הקבוצה הטובה ביותר.

נציין כי ניתן לבצע הרצות של משחק הדראפט גם על עונות נוספות, אך לא עלה צורך לכך, כיוון שניתן להעריך את ביצועי הסוכנים (כלומר, טיב האלגוריתמים השונים לפיתרון הבעיה) על סמך ניסויים עבור עונה אחת בלבד.

## הערכת תוצאות משחק הדראפט

כפי שצוין זה מכבר, לא קיים מדד מוחלט לפיו ניתן לאמוד את טיב קבוצה שהורכבה בדראפט, שכן משחק הפנטזי הינו משחק דינאמי שמתפרש על פני עונה שלמה ואינו מושפע מההישגים הסטטיסטיים של השחקנים בממוצע למשחק. לשם כך, עולה הצורך בפיתוח מדד השוואה לבחינת יעילות (utilities) של קבוצת שחקנים, על מנת שיהיה ניתן לאמוד את הקבוצה שהרכיב כל סוכן ולהשוות אותה אל מול שאר הקבוצות שהורכבו.

השתמשנו בשני מדדים על פיהם ניתן להשוות את תוצאותיהן של הקבוצות, אשר לקוחים מעולם המשחק האמיתי, ואף משמשים שחקני פנטזי בשביל להעריך את טיב קבוצתם. נזכיר כי את שני המדדים חישבנו עבור תוצאות האמת של שחקני ה-NBA מאותה עונה, ולא על פי תוצאות החיזוי.

**להלן הסבר על כל אחד מהמדדים ודוגמא המראה כיצד הם מחושבים:**

נניח שלוש קבוצות שכל אחת מהן מורכבת משני שחקני NBA, ונתייחס לשלושקטגוריות ניקוד בלבד מתוך תשע הקטגוריות –

קבוצה	שחקן	נקודות למשחק	אסיסטים למשחק	ריבאונדים למשחק
Team A	לברון ג'יימס	27.4	8.3	8.5
	אנתוני דיוויס	25.9	3.9	12
Team B	קוואי לנארד	26.6	3.3	7.3
	פול ג'ורג'	28	4.1	8.2
Team C	ראסל ווסטברוק	22.9	10.7	11.1
	ג'יימס הארדן	36.1	7.5	6.6

סך כל הניקוד פר קטגוריה:

קבוצה	נקודות למשחק	אסיסטים למשחק	ריבאונדים למשחק
Team A	53.3	12.2	20.5
Team B	54.6	7.4	15.5
Team C	59	18.2	17.7

### 1. מדד Wins

מדד זה הינו האופציה הטבעית להשוואה, שכן הוא מסתמך על אופן ההכרעה של matchup בין שתי קבוצות. המדד מדרג את הקבוצות לפי מספר הניצחונות הכולל ב-matchup ישיר מול כל אחת מקבוצות האחרות. בשיטה זו אנו בוחנים כל קומבינציה של matchup בין 2 קבוצות שונות, ולפי כללי משחק הפנטזי מכריעים מיהי הקבוצה המנצחת, כאשר הקבוצה המנצחת צוברת נקודה ואילו הקבוצה המפסידה אינה צוברת נקודה. לבסוף, סוכמים את מספר הניצחונות עבור כל קבוצה, ומדרגים את קבוצות הסוכנים לפי מספר ניצחונות כולל, כאשר הסוכן בעל מספר הניצחונות הגדול ביותר מדורג ראשון והסוכן בעל מספר ניצחונות המועט ביותר מדורג אחרון.

עבור הדוגמא הנתונה, ישנן 3 קומבינציות שונות ל-matchup: A vs B, A vs C, B vs C. בהתמודדות הראשונה ניתן לראות כי Team A מובילה ב-2 מתוך 3 הקטגוריות (אסיסטים למשחק וריבאונדים למשחק) ועל כן זוכה ב-matchup וצובר נקודה. בהתמודדות A vs C, Team C מובילה ב-2 קטגוריות (נקודות למשחק וריבאונדים למשחק) וצוברת נקודה. בהתמודדות B vs C, Team C מובילה בכל הקטגוריות ולכן צוברת עוד נקודה. מכאן סך הניקוד הינו: Team A : 2, Team B : 1, Team C : 0 וזהו גם הדירוג המתקבל על פי מדד זה.

## 2. מדד Ranks

במדד זה מטרתנו הינה לנסות ולמצוא שיטת השוואה שתיתן תמונה רחבה יותר על יחס הכוחות בין הקבוצות בכלל הקטגוריות ותיבחן את חוזק הקבוצה עבור הקטגוריות השונות, לעומת "השוואה בינארית" (ניצחון או הפסד) כפי שמתקיים בשיטה הקודמת. באופן זה ניתן לתת משקל ליתרון משמעותי של סוכן על פני קטגוריות שונות למרות חולשה במספר קטגוריות בודדות. גם במשחק הפנטזי האמיתי ישנה אפשרות במהלך כל שלב של הדראפט (וכן בסופו ובמהלך העונה) לראות את דירוג הקבוצה אל מול שאר הקבוצות בקטגוריה נבחרת (ראו איור 4: מסך דירוג קבוצות הליגה בכל אחת מקטגוריות המשחק, מתוך פלטפורמת משחק הדראפט של Yahoo). מדד זה משפיע רבות על שחקני הפנטזי בעת בחירת השחקנים וביצוע מהלכים נוספים במהלך העונה. לפי מדד זה ניתן דירוג לכל סוכן עבור כל קטגוריה, כאשר סוכן עם תוצאה מירבית בקטגוריה מסוימת יקבל דירוג 1 לאותה קטגוריה. לבסוף סוכמים את סך הדירוגים של כל סוכן בכלל הקטגוריות, והסוכן בעל הסכום הנמוך ביותר מדורג ראשון (כיוון שלהימצא במקום הראשון בקטגוריה, כלומר להיות הכי טוב בקטגוריה, מקנה נקודה אחת), ואילו הסוכן בעל הסכום הגבוה ביותר מדורג אחרון.

בדוגמא הנתונה עבור הקטגוריה "נקודות למשחק", הדירוג (מראשון לאחרון) הינו: C (59), B (54.6), A (53.3). מכאן Team C צוברת אפס נקודות (שליליות), Team B צוברת נקודה שלילית אחת ו-Team A צוברת 2 נקודות שליליות. הדירוג עבור הקטגוריה "אסיסטים למשחק": C, A, B. הדירוג עבור הקטגוריה "ריבאונדים למשחק": A, C, B. הניקוד עבור שתי הקטגוריות האחרונות מתבצע באותו אופן כפי שתואר עבור הקטגוריה הראשונה. מכאן אנו מקבלים כי סכום הדירוגים (ניקוד שלילי) על פני שלושת הקטגוריות הנמדדות הינו:

- Team A :  $2+1+0=3$
- Team B :  $1+2+2=5$
- Team C :  $0+0+1=1$

כלומר, קיבלנו כי Team C מדורגת ראשונה במדד, לאחר Team A ולאחריה Team B. הדירוג אמנם יצא זהה לדירוג שהתקבל במדד ה-Wins עבור דוגמא זו, אך ככל שמרחיבים את כמות הקבוצות, השחקנים והקטגוריות הנמדדות, ישנה סבירות גבוהה לקבל דירוג שונה עבור כל אחד מהמדדים.

## אסטרטגיות להתמודדות במשחק דראפט מרובה סוכנים

השיטות המובילות בתחום הבינה המלאכותית למידול שחקן במשחק רב משתתפים **מבוססות חיפוש היוריסטי במרחב מצבי המשחק**. בשיטות אלו, הסוכן מריץ אלגוריתם חיפוש בגרף המצבים של המשחק, במטרה לתכנן את הפעולות שלו בכדי להגיע למצב רצוי, ותוך שימוש בהיוריסטיקה לצורך הערכת התועלת של מצב נתון. הסוכן יישאף להבטיח סדר פעולות אשר יוביל למצב בעל ערך היוריסטי גבוה ככל הניתן, כאשר קיים מימד אי וודאות לגבי אסטרטגיית הפעולה של היריבים.

### תיאור מרחב המצבים

המצבים במרחב החיפוש מוגדרים באופן הבא:

#### 1. מצב –

- א. **מספר הבחירה הנוכחי** – מספר המייצג את מספר הבחירה הנוכחי מתוך סך הבחירות בדראפט (של כלל הסוכנים).
- ב. **מזהה הסוכן** – מספר המייצג את מזהה הסוכן שכעת תורו לבחור שחקן בדראפט.
- ג. **בחירות הסוכנים** – תוצאות המשחק עד כה – מצב הבחירות הנוכחי של כלל הסוכנים.
- ד. **עמדות פנויות לכל קבוצה** – האילוצים שנותר לספק – עבור כל סוכן נשמר את עמדות הנותרות אותן עליו לאייש ע"י בחירת שחקנים.
- ה. **מאגר השחקנים הפנויים** – אוסף השחקנים שניתן לבחור (טרם נבחרו ע"י אף קבוצה).

2. **מצב סופי** – מצב סופי הינו מצב שבו מספר הבחירה הנוכחי הינו גדול מסך הבחירות בדראפט, כלומר כאשר הסוכן שתורו אחרון בחר את שחקנו האחרון.

3. **מצבים עוקבים (successors)** – מצב עוקב הינו המצב המתקבל לאחר שהסוכן ביצע פעולה מתוך אוסף הפעולות החוקיות. במשחק הדראפט פעולה שקולה לבחירת שחקן לקבוצתו בתורו של הסוכן. מצבים עוקבים חוקיים מקיימים:
  - א. השחקן הנבחר הינו שחקן ממאגר השחקנים, כלומר השחקן טרם נבחר ע"י סוכן כלשהו.
  - ב. השחקן הנבחר מקיים את אילוף העמדות הפנויות, ומאייש עמדה כזו עבור אותו סוכן.

4. **גודל מרחב המצבים** – לגודל מרחב המצבים ישנה השפעה גדולה על זמן החיפוש ובעיה שלנו על בחירת השחקן בכל תור של הסוכן. בהינתן מספר הסוכנים, גודל קבוצה ומספר שחקני ה-NBA בליגה, ניתן לחשב את גודל מרחב המצבים במשחק.

5. **גודל מרחב החיפוש** – האלגוריתמים שבחנו הינם מבוססים על חיפוש במרחב המצבים שהוגדר לעיל. מרחב החיפוש מושפע ישירות ע"י עומק ומקדם הסיעוף וגודלו  $O(b^d)$ . בבעיית משחק הפנטזי העומק הינו מספר הבחירות הכולל במשחק, כלומר מספר הסוכנים כפול גודל קבוצה,

ומקדם הסיעוף הינו מספר שחקני ה-NBA (יש לשים לב כי גודל מקדם הסיעוף מצטמצם בכל בחירה כאשר מוסר שחקן מתוך המאגר, כלומר בחירתו אינה פעולה חוקית יותר).

**משיקולי זמן ריצה, בחרנו להגביל את עומק החיפוש ומקדם הסיעוף עבור מרבית הסוכנים.** עומק החיפוש הוגבל באופן משתנה בין סוכן לסוכן. **מקדם הסיעוף הוגבל תוך הסתמכות על חיזוי דירוג הפנטזי של שחקני ה-NBA (Fantasy Rank),** כלומר, הוחזרו x (מוגדר מראש) שחקנים פנויים אשר דירוג הפנטזי שלהם הינו הגבוה ביותר. אסטרטגיה זו מונעת משיקולים אשר נשלקחים בחשבון במשחק הפנטזי האמיתי – שחקני פנטזי לא יכולים לבחון בכל תור את כלל השחקנים הפנויים, ועל כן משתמשים במדד דירוג הפנטזי על מנת לצמצם את מספר האפשרויות, תוך הנחה שהדירוג הנ"ל מציג את השחקנים בעלי הערך הכולל הגבוה ביותר, אשר סביר כי תרומתם לקבוצה תהיה ניכרת על פני יתר השחקנים.

### **אלגוריתמי חיפוש במרחב מצבים**

על מנת לפתור את הבעיה הנתונה, שהינה בעיית משחק מרובה סוכנים, בחנו מספר אלגוריתמים, מרביתם משתמשים בחיפוש היוריסטי על מרחב המצבים שהוגדר לעיל.

נפרט להלן את האלגוריתמים שנבחנו במהלך פיתוח המערכת:

#### **1. Random**

בשיטה זו כל בחירה מתבצעת בצורה רנדומלית מתוך מאגר השחקנים הפנויים, באופן לא מיועד וללא התחשבות בשיקולים כלשהם (מלבד אילוצי העמדות). שיטה זו נבחנה לצורך מתן הערכה ראשונית על אופן העבודה של המערכת, ושימשה כנקודת בוחן לאורך הפיתוח.

#### **2. Maximize Category**

גם שיטה זו אינה מבוססת היוריסטיקה, ומבצעת את הבחירה הבאה באופן חמדני במטרה למקסם קטגוריה מסוימת מתוך קטגוריות המשחק (או דירוג הפנטזי הכללי). ההיגיון בשיטה זו מונע מכך שהשחקנים המובילים בליגה הינם בעלי תוצאות גבוהות בקטגוריות מסוימות כגון נקודות (PTS) ודירוג פנטזי כולל.

אופן בחירה זה הינו פשוט ויעיל, ועם זאת, מאפשר הרכבת קבוצה בעלת פוטנציאל הצלחה סביר. חשוב לציין כי מגבלות שיטה זו באות לידי ביטוי בעיקר עבור ליגות גדולות, כלומר, כאשר מאגר השחקנים מצטמצם (בשלב המאוחרים של משחק הדראפט), ויש צורך לבצע הפרדה מדויקת של השחקנים פנויים על פי היתרון היחסי שלהם, לצורך התאמתם לקבוצה וליתרונותיה. ניתן להניח כי בסיבובים הראשונים של הדראפט, מרבית השחקנים מספקים פחות או יותר את אותה תרומה כוללת, ועל כן בחירת שחקן מוביל תהיה מספקת. עם זאת, בסיבובים המתקדמים, נותרים שחקנים התורמים בקטגוריות מעטות ונדרשת בחינה רחבה יותר של תרומתם האפשרית.

### 3. Random Categories Maximization

בדומה לשיטת מיקסום קטגוריה ספציפית, **בשיטה זו נגריל בכל בחירה את הקטגוריה אותה נרצה למקסם, מתוך רשימת קטגוריות נתונה מראש.** המטרה של שיטה זו נועדה לספק מענה אפשרי לבעייתיות ששהוצגה עבור שיטת ה-Maximize Category, בכך שתאפשר בניית קבוצה מגוונת יותר.

כמו כן, **שיטה זו מאפשרת לכוון את הרכבת הקבוצה על פי Punt מסויים.** כפי שתואר בפרק המבוא, Punt הינה אסטרטגיה מובילה בבניית קבוצת פנטזי בדראפט, לפיה "מוותרים" על קטגוריות מסויימות על מנת לחזק את הקבוצה ברוב של קטגוריות אחרות (לפחות 5 קטגוריות מתוך ה-9). אסטרטגיית Punt תמומש בשיטה זו ע"י בחירה רנדומית מתוך תת קבוצה של קטגוריות הניקוד, בעלות קשר ביניהן (למשל, קטגוריות המותאמות בעיקר לשחקנים נמוכים). החיסרון של שיטה זו הינו ההיבט ההסתברותי שבבחירת הקטגוריה למיקסום, אשר אינה בהכרח מסתמכת על הצורך בפועל ובזמן אמת של הקבוצה.

### 4. Alpha Beta

אלגוריתם זה, שנלמד בקורס המבוא, הינו אלגוריתם מבוסס חיפוש היוריסטי, המבוסס על אסטרטגיית Min-Max עם גיזום ענפים. לפי שיטה זו, **בכל שלב החלטה של הסוכן, נפרש עץ החיפוש במרחב המצבים, כאשר בצמתי בחירת הסוכן (צמתי מקסימום), נבחר הצעד הממקסם ערך היוריסטי מוגדר מראש, בעוד בצמתי היריב (צמתי מינימום) נניח כי התבצע הצעד המרע ביותר עבור הסוכן (מבחינת הערך ההיוריסטי).** כיוון שמרחב המצבים הינו גדול מאוד, החיפוש מתבצע לעומק מוגדר מראש ועבור מקדם סיעוף מוגבל.

**גיזום אלפא-בטא מתבצע באופן הבא –** בצמתי מקסימום, כאשר מובטח צעד אשר יוביל לערך היוריסטי מסוים, נסתכל על צמתי המינימום (הבנים), ואם באחד מהם הושג ערך נמוך יותר מזה שהובטח בצומת המקסימום, נוכל לגזום את תת עץ זה. באותו אופן, נבצע גיזום בצמתי מינימום, כאשר בצמתי הבנים יושג ערך גבוה מזה שהובטח בצומת המינימום. אלגוריתם זה אינו משפיע על התוצאה המתקבלת מאלגוריתם Min-Max, אך זמן הריצה שלו קצר יותר (לכל היותר זהה לזמן הריצה של Min-Max, אם לא מתבצע גיזום כלל), ועל כן בחרנו להשתמש בו.

**שיטה זו בעלת פוטנציאל להשגת תוצאות טובות במשחק הדראפט,** שכן מאפשרת לסוכן "להסתכל קדימה" ולבסס את בחירתו בהתאם לצעדים אפשריים שיבצעו יריביו. שיטה זו דומה במקצת לשיטת הפעולה הנפוצה במשחק הדראפט האמיתי, בו בעת הבחירה, מנסים להעריך אילו שחקנים ייבחרו על ידי היריבים בתורות הבאים, כדי לנסות ולבנות תוכנית פעולה ארוכה טווח – כמה סיבובים קדימה.

עם זאת, בליגות פנטזי מרובות קבוצות, ישנו פער של מספר בחירות רב בין כל שתי בחירות של הסוכן. **ממגבלות זמן הריצה של האלגוריתם (פיתוח עץ חיפוש בכל תור), היתרון שניתן להשיג באמצעות אסטרטגיה זו מצטמצם –** הסוכן למעשה לא בהכרח יוכל "לסמלץ" עץ משחק המכיל את הבחירות הבאות שלו, אלא מספר בחירות עוקבות של שחקנים יריבים בלבד.

כמו כן, בשיטה זו מניחים כי היריב ייבחר בפעולה הרעה ביותר עבור סוכן המקסימום, למרות שאסטרטגיית הפעולה שלו אינה ידועה. לרוב, כל סוכן במשחק ינסה למקסם את תוצאות קבוצתו, ולא דווקא יפעל במטרה לפגוע בסוכן היריב.

## 5. Expectimax

אלגוריתם זה מבוסס גם הוא על חיפוש היוריסטי לעומק. **בשיטה זו בצמתי הסוכן (מקסימום) נבחר הצעד הממקסם את הערך היוריסטי, בעוד בצמתי היריב מניחים התפלגות כלשהי על אפשרויות הבחירה של היריב, והערך ההיוריסטי הניתן למצב הינו תוחלת הערכים המחושבת על כלל הבנים של הצומת.**

מכיוון שאסטרטגיית היריב אינה ידועה, גם בחירת ההתפלגות לפיה מניחים שהיריב פועל הינה בגדר השערה. על כן, נבחנו שתי התפלגויות אפשריות עבור שחקן יריב בדראפט:

1. **התפלגות אחידה בין השחקנים הפנויים עבור מקדם סיעוף מוגדר מראש** – כלומר, סוכן ה-Expectimax מניח כי ישנה הסתברות זהה לבחירת כל אחד מהצעדים האפשריים של היריב, לפי הגבלות מקדם הסיעוף (והמיון לפי ערכי דירוג הפנטזי) כפי שפורט בסעיף תיאור מרחב המצבים.

2. **התפלגות לפי דירוג Fantasy Rank יחסי** – עבור מקדם סיעוף מוגדר מראש  $x$ , נלקחו בחשבון כאפשרויות פעולה של היריב  $x$  השחקנים הפנויים בעלי דירוג הפנטזי הגבוה ביותר. לכל בחירה אפשרית באחד מהשחקנים הללו, ניתנה הסתברות השווה לערך דירוג הפנטזי שלו, חלקי סך כלל הדירוגים של  $x$  השחקנים. בצורה זו, מתקבלת הסתברות חוקית (כל הערכים בין 0 ל-1 וסכומם נותן 1 בדיוק). התפלגות זו מהווה הנחנה הגיונית, מתוך הבנה של משחק הדראפט, על אופן פעולה אפשרי של היריב.

## 6. Maximax

שיטה זו מבוססת על העיקרון **לפיו הסוכן אינו מניח את "המקרה הגרוע ביותר" בעת בחינת תור היריב, אלא את "המקרה הטוב ביותר" עבור היריב**, על פי היוריסטיקה מוגדרת מראש. **מדובר באינטרפטציה אישית שלנו לשיטת פעולה זו**, אשר אינה בהכרח תואמת את ההגדרה הפורמלית של אלגוריתם Maximax.

אלגוריתם זה משתמש בשתי היוריסטיקות לצורך קבלת החלטה – היוריסטיקת הסוכן, להערכת צמתיים בהם תורו של הסוכן לבחור, והיוריסטיקת היריב, להערכת צמתיים בהם בוחר סוכן יריב. באינטרפטציה שהגדרנו, בתור סוכן יריב לא מתבצע חיפוש לעומק, אלא מניחים בחירה על סמך הערך היוריסטי על המצבים העוקבים בשכבה הבאה בלבד.

**באופן זה האלגוריתם מאפשר גמישות רבה יותר ולמעשה מכפר על אחת המגבלות של אלגוריתמי AlphaBeta ו-Expectimax** – ניתן בזמן ריצה סביר להגיע בעומק החיפוש עד לבחירה הבאה של הסוכן (או אף מספר הבחירות הבאות) ובכך לדמות בצורה טובה יותר מעין אסטרטגיה ארוכת

טווח. כמו כן, שיטה זו מדמה באופן מדויק יותר את שיטת הפעולה של היריבים במשחק דראפט אמיתי, כאשר כל סוכן לרוב ישאף למקסם את קבוצתו, ללא התייחסות לפגיעה בקבוצות היריבות.

## 7. Local Search

**אלגוריתמי חיפוש מקומי שואפים בכל תור לבצע את הצעד אשר ממקסם את תועלת התוצאה על פי פונקציית היוריסטיקה נתונה.** בשונה מאלגוריתמי חיפוש לעומק, באלגוריתם זה הבחירה הינה על סמך הערך ההיוריסטי של המצבים העוקבים למצב הנוכחי, ללא בחינה של תוצאה אפשרית שניתן אולי להבטיח בהמשך המשחק וללא התחשבות בפעולת היריב בתורות הבאים.

**שיטה זו אינה נפוצה בפיתרון בעיות מרובות סוכנים, אך אנו מאמינים כי אופי משחק הדראפט עשוי להתאים דווקא לשיטת פעולה כזו.** משחק הדראפט מכיל חוסר ודאות רב בין כל שתי בחירות עוקבות של סוכן וכמו כן, מספר אפשרויות בחירה רבות בכל תור (מקדם הסיעוף), אשר אין ביניהן בהכרח הבדל גדול (לרוב בתור נתון ישנם מספר שחקנים פנויים בעלי נתונים דומים). אלגוריתם זה מאפשר בחינה של אפשרויות פעולה רבות בזמן ריצה קצר יחסית.

אנו השתמשנו בגירסה בסיסית של אלגוריתם Local Search המבוססת על שיטת Steepest Ascent, כלומר, בכל תור הסוכן בוחר לעבור למצב בעל הערך ההיוריסטי הגבוה ביותר, ללא שיקולים נוספים. אנו מעריכים כי שיטה זו הינה מספקת עבור משחק הדראפט, שכן במשחק זה קשה מאוד לבצע צעד שמוריד את ערך הקבוצה ולכן לא סביר "להיתקע" במקסימום מקומי או לנוע על גבי כתף.

## 8. Hybrid Strategy

בשיטה זו אנו משלבים מספר אסטרטגיות לכדי אסטרטגיה אחת, כאשר בכל תור, בהתאם לשלב ולמצב המשחק, נבחרת שיטת פעולה מתאימה. אסטרטגיה זו הינה מבוססת במידה ניכרת על שיטות פעולה נפוצות במשחק הדראפט האמיתי – לרוב, בסיבובים המוקדמים נעדיף לבחור את השחקן "הטוב ביותר על הלוח", כלומר, השחקן בעל התרומה הכוללת המירבית, ללא התחשבות בהתאמתו לקבוצה. בסיבובים מאוחרים נשאף לבצע התאמות ו-"לסתום חורים" בקבוצה או לחזק יותר את הקטגוריות בהן אנו כבר מובילים (בהתאם למשל ל-Punt מסויים שהחלטנו עליו במהלך הדראפט).

## היוריסטיקות על מרחב המצבים

כפי שניתן לראות מתיאור האלגוריתמים בהם השתמשנו, מרביתם מתבססים על היוריסטיקה מוגדרת מראש להערכת המצבים אליהם מגיעים במהלך החיפוש. לשם כך, הוגדרו ההיוריסטיקות הבאות:

1. Rank Per Category – היוריסטיקה זו פועלת באותה שיטה של פונקציית המטרה (utility) של מדד Ranks כמתואר מעלה, ומחזירה עבור מצב נתון את דירוג הסוכן ע"פ מדד זה.



2. **Quorum Wins** – היוריסטיקה זו פועלת באותה שיטה של פונקציית המטרה (utility) של מדד Wins כמתואר מעלה, ומחזירה עבור מצב נתון את דירוג הסוכן ע"פ מדד זה.

3. **Maximize Label** – היוריסטיקה זו מקבלת קטגוריה מוגדרת מראש ומשערכת מצב ע"פ סך התוצאות של שחקני קבוצת הסוכן עבור אותה קטגוריה. שיטה זו הינה מכוונת לחיזוק הקבוצה לפי קטגוריה מסוימת. שיטת שערך זו הינה פשוטה ובעלת זמן ריצה נמוך מאוד המאפשר פיתוח יותר מצבים באותה מסגרת זמן. היוריסטיקה זו טובה עבור אסטרטגיה בה סוכן מעוניין להבטיח דירוג גבוה בקטגוריה ספציפית. בנוסף, קבוצה המורכבת בשיטה זו ומציגה תוצאות טובות יכולה להעיד על תכונה מייצגת עבור שחקנים מובילים בליגה.

4. **Maximizing Multi Labels Weighted** – היוריסטיקה זו מקבלת רשימה של קטגוריות ורשימה של משקלים מתאימים המוגדרים מראש. אופן שערך מצב דומה לאופן השערך עבור ההיוריסטיקה Maximize Label, אך מאפשר הסתכלות על מספר קטגוריות (לעומת לקטגוריה בודדת) ומאפשרת מתן משקל (לפי חשיבות) לכל קטגוריה. שיטת פעולה זו עשויה לאפשר לסוכן לפעול לפי אסטרטגיית Punt כפי שמפורט בתחילת הדו"ח. באופן זה סוכן יכול להרכיב קבוצה בעלת חוזקות מתוכננות מראש, ולבחור באילו קטגוריות הוא מעדיף להתחזק ואילו קטגוריות הוא "מקריב".

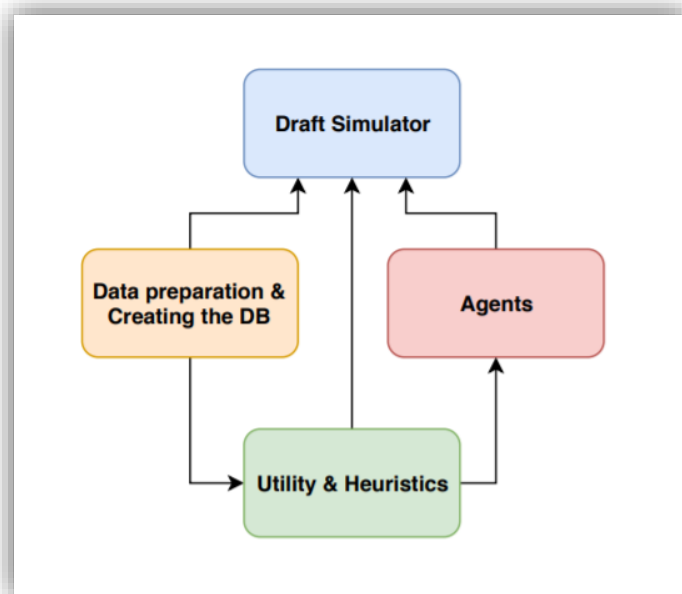
5. **Advantage on Opponent** – היוריסטיקה זו מקבלת רשימת קטגוריות מוגדרות מראש ומשערכת מצב ע"פ **סכום ההפרשים המנורמלים של הסוכן על פני כלל הקטגוריות אל מול כלל הסוכנים**. כלומר, עבור כל קטגוריה שנבחרה, מחשבים את סך תוצאות הסוכן עבור קטגוריה זו כאשר הערך מנורמל ונמצא בטווח  $[-1, 1]$  כך שמתקבל וקטור בגודל מספר הקטגוריות הנבחרות כאשר כל איבר בוקטור מייצג את הערך המנורמל של הסוכן אל מול ערכי שאר הסוכנים בקטגוריה מתאימה. לבסוף, סוכמים את איברי הוקטור לקבלת ערך המייצג את הפרש תוצאות הסוכן אל מול הסוכנים היריב עבור כלל הקטגוריות הנבחרות.

שיטת שערך זו דומה לשיטת Rank Per Category, אך **מאפשרת מתן ערך יחיד מייצג של יתרון הסוכן אל מול יריביו על פני מספר קטגוריות (וכן על כלל הקטגוריות) וקבלת הערכה מדויקת יותר על יתרון הסוכן במצב הנתון על פני יריביו**. היוריסטיקה זו נועדה לפתור את המורכבות של שמירת יתרון מירבי ככל הניתן על פני מספר מקסימלי של יריבים. עבור שחקן אשר מציג יכולות טובות בקטגוריה  $x$ , סביר שקיימת קטגוריה  $y$  אשר בה תוצאותיו פחות טובות (למשל שחקן גבוה אשר משיג הרבה ריבאונדים, סביר מאוד שיכולותיו בקליעת שלשות תהיה נמוכה). מכאן עולה הבעיה כי בחירת שחקן במטרה לחזק את הקבוצה בקטגוריה  $x$ , תפגע בקבוצה בקטגוריה  $y$ , ועולה השאלה האם בחירת שחקן תורמת לקבוצה או פוגעת בה, בהתחשב בכלל היריבים בכלל הקטגוריות. לכן, בעזרת היוריסטיקה זו תינתן עדיפות לבחירות שחקן אשר יתרום בצורה המירבית לקבוצה תוך שימור היתרון על פני מספר רב (ככל הניתן) של יריבים.

## תיאור מבנה המערכת

מימוש מערכת השחקן והמשחק כולו הינו בשפת Python עם שימוש בסיפרייה Pandas ומספר ספריות בסיסיות נוספות. המערכת מורכבת ממספר מודולים, אשר יחד מאפשרים להריץ בצורה נוחה ויעילה הדמיה של משחק הדראפט עם קבוצות אשר מבצעות את בחירותיהן על פי שיטות ואלגוריתמים שונים. ניתן לחלק את מבנה המערכת למודולים הבאים (לאו דווקא מדובר בחלוקה מבנית בקוד, אלא על חלוקה רעיונית לפי תפקיד במערכת כולה):

1. **הכנת תשתית המידע עבור המשחק** – כפי שפורט בתיאור משחק הדראפט, לצורך ביצוע הערכות וקבלת החלטות במהלך הדראפט יש צורך בבסיס נתונים המספק הערכות לגבי הישיגיהם הסטטיסטיים של שחקני ה-NBA. בסיס הנתונים שבו השתמשנו הינו תוצאות מערכת החיזוי שמימשנו בחלקו הראשון של הפרויקט, ונדרשנו להתאים את קובץ התוצאות כך שתוכנו יתאים לשימוש במימוש סוכן הדראפט. כמו כן, ישנו צורך בנתונים נוספים המשמשים לניהול המשחק כגון עמדות שחקני ה-NBA, לצורך התאמה להגבלות הדראפט.
2. **מעטפת המשחק** – פיתרון הבעיה בחלק זה מבוסס חיפוש היוריסטי במרחב המצבים של משחק הדראפט (פירוט של רעיון הפיתרון ניתן לראות בפרק "תיאור פיתרון הבעיה" בחלק זה). לצורך כך, נדרשת מעטפת מתאימה אשר תאפשר פיתוח של מצבים במרחב הבעיה. כמו כן, נדרשת מעטפת כוללת אשר תאפשר יצירת מצב התחלתי והרצת משחק הדראפט עצמו.
3. **סוכני משחק הפנטזי** – מטרת המערכת היא לפתח סוכן אשר מבצע בחירות עצמאיות בהאם למצב נתון במשחק הדראפט. לצורך כך, נדרשנו לממש סוכנים שונים הפועלים על פי אלגוריתמים ושיטות פעולה שונות. הסוכנים שפותחו שימשו לשתי מטרות – המטרה הראשית שהינה בניית שחקן אוטונומי עבור הדראפט, ומטרה משנית (אשר שירתה את המטרה הראשונה) – בתור סוכנים "יריבים" בניסויים שנועדו לבחון את ביצועי כל אחת מהשיטות שפיתחנו.
4. **היוריסטיקות ופונקציות יעילות** – כאמור, מרבית הסוכנים שבחנו עבור פתרון הבעיה משתמשים בחיפוש היוריסטי לצורך ביצוע הבחירה שלהם בכל שלב בדראפט. חלק זה כולל מימוש של פונקציות הערכה למצב במרחב המצבים של הבעיה, וכן פונקציות הערכה לפיהן מדדנו את התוצאה הסופית המשחק.



איור 9: תרשים התלויות של מודולי מערכת משחק הדראפט

## הכנת תשתית המידע עבור המשחק

### סידור תוצאות החיזוי

הפלט של מערכת החיזוי כולל, עבור עונה מסויימת, את חיזוי ההישגים הסטטיסטיים של כלל השחקנים באותה עונה בתשעת הקטגוריות שהוגדרו, יחד עם הערכה כוללת לדירוג הפנטזי שלהם. כפי שתואר בחלק של מערכת החיזוי, השתמשנו בשיטת למידה וחיזוי שונות עבור כל קטגוריה סטטיסטית. לפיכך, תוצאות החיזוי סופקו בעשרה קבצים שונים, בהם שם השחקן (key) ותוצאות החיזוי עבור הקטגוריה (value). כמו כן, לצורך ההערכה סופית של תוצאות המשחק, השתמשנו כאמור בנתונים הסטטיסטיים האמיתיים מאותה עונה.

עם כן, בשלב זה מתבצעות שלוש פעולות מרכזיות, אשר ממומשות במספר סקריפטים קצרים –

- הרצת כל אחד ממערכות החיזוי הנבחרות על קבוצת האובייקטים שברצוננו לחזות (בהתאם לעונה שעבורה מתבצע משחק הדראפט).
- איחוד כלל תוצאות החיזוי עבור כל אחת מהקטגוריות לקובץ קלט יחיד עבור העונה המבוקשת. לתוצאה הכללית התווספה עמודת Id – מעתה ההתייחסות לשחקנים במהלך המשחק הינה באמצעות ה-Id (מזהה חח"ע לשחקן) ולא באמצעות שמו (לצורך נוחות המימוש).
- יצירת קובץ עם נתוני האמת מהעונה המתאימה.

יש לציין בשלב זה כי עקב חוסר התאמה מלא בין השחקנים המופיעים בקובץ תוצאות האמת לבין השחקנים המופיעים בקבצי תוצאות מערכת החיזוי, נדרשנו לבצע הצלבה בין המפתחות בכל אחת מהטבלאות, בכדי להבטיח שתהיה הלימה מלאה בין שחקני ה-NBA שהקבוצות המשתתפות בדראפט בוחרות, לבין אלו שניתן להעריך את ערכם בעת חישוב התוצאה הסופית של המשחק. פעולה זו גררה צמצום מועט של כמות השחקנים המהווים את התשתית למשחק הדראפט, אך בעיקר עבור שחקנים משניים שככל הנראה לא היו נבחרים בדראפט גם עבור ליגה עם מספר קבוצות וגודל קבוצות גדול מאוד.

## **בניית DB עבור המשחק**

במהלך הרצת הסימולציה של משחק הדראפט, מוחזק DB הכולל מספר מילונים (טבלאות ערבול) המשמשים לאחזור מהיר ונוח של מידע הדרוש לסינון אפשרויות החבירה בכל שלב ולחישוב הפונקציות ההיוריסטיות. ה-DB הינו אובייקט הנבנה בתחילת הרצת המשחק, וכולל עיבוד נוסף של המידע וחלוקתו למילונים המבוקשים. בנוסף, בשלב אתחול ה-DB ניתן להוסיף מניפולציות נוספות על המידע, בהתאם לדרישות אפשריות שיתווספו למשחק (למשל התחשבות בשחקנים פצועים, הוספת נתונים נוספים מלבד החיזוי הסטטיסטי לשימוש הסוכנים וכד').

### **בעת אתחול ה-DB מתבצעות הפעולות הבאות על המידע:**

1. **נרמול תוצאות דירוג הפנטזי (Fantasy Rank)** – תוצאות קטגוריה זו שהתקבלו מהמסווג אינן בעלות משמעות כאשר בוחנים את ערך התוצאה בפני עצמו. התוצאה הנ"ל נדרשת רק עבור מתן דירוג בין כלל השחקנים – מי בעל ערך גבוה יותר, כאשר ההפרש אינו רלוונטי, זאת בעיקר עקב הסטייה שהתקבלה בתוצאות, אך יש לציין כי גם במשחק האמיתי, לרוב משתמשים בנתון זה כדי לדרג את השחקנים בלבד.

2. **תיקון ערכים שליליים** – עקב השימוש בשיטת רגרסיה בתהליך החיזוי, עבור אובייקטים שדורגו בקצוות בקטגוריות הכוללות ערכים נמוכים (למשל קטגוריית החטיפות – STL), התקבלו ערכים שליליים בודדים. משמעותם של ערכים אלו לנכונות תוצאות משחק הדראפט הינה אפסית, שכן מדובר בשחקנים בעלי ערך פנטזי שולי שככל הנראה לא ייבחרו בדראפט. לצורך נוחות, בחרנו לאפס את כלל הערכים הללו (סדר גודל של 10 אובייקטים שנתוניהם שונו).

3. **הסרת כפילויות וביצוע Re-Indexing לרשימת השחקנים** – כתוצאה מאופי מאגרי הנתונים בהם השתמשנו בשלב החיזוי, נוצרו מעט כפילויות בקובצי תוצאות החיזוי (עקב שחקנים שעברו קבוצה במהלך העונה). בשלב אתחול ה-DB נלקח המופע הראשון של השחקן, לפי מיון של דירוג ה-Fantasy Rank של השחקנים.

#### 4. יצירת ה-DB –

נתוני המשחק (חיזוי, תוצאות אמת ומידע נוסף) נטענים כ-Data Frame למערכת, ולאחר שמבצעים עליהם את המניפולציות שצינו לעיל, מתבצעת חלוקה של הנתונים למספר מילונים לצורך יעול החיפוש ואחזור המידע.

#### DB המשחק מורכב מארבעת המילונים הבאים:

א. **Id → Player Name** – משמש לשחזור בין מזהה השחקן לשם השחקן. מיפוי זה אינו משמש את הסוכנים בשלבי הדראפט ונועד לצורך בחינה עצמאית של התוצאות, וכאפשרות להרחבת המערכת למערכת משחק של ממש, המריצה סימולציית דראפט ומאפשרת צפייה בשחקנים הנבחרים.

ב. **Player Id → {Category → Prediction}** – זהו למעשה מילון של מילונים, המאפשר גישה נוחה ומהירה לתוצאות החיזוי של שחקן בקטגוריה מבוקשת. מיפוי זה משמש בחלקים רבים בשלב החיפוש של הסוכנים ובשלבי חישוב הפונקציות ההיוריסטיות, והינו יעיל יותר מחיפוש על גבי ה-Data Frame המלא המחזיק את תוצאות החיזוי.

ג. **Category → Sorted Prediction List** – במהלך המשחק נדרש לבצע לעיתים תכופות מיון של רשימת השחקנים על פי ערך החיזוי שלהם בקטגוריה נתונה, מילון זה מאפשר לחסוך את זמן המיון החוזר.

ד. **Id → Position** – מיפוי בין מזהה שחקן לעמדה שלו במשחק הכדורסל, לצורך סינון והתאמת מאגר השחקנים למגבלות המשחק בכל שלב.

### מעטפת המשחק

#### מבנה מרחב החיפוש

מצבי מרחב החיפוש במשחק הדראפט מכילים מידע אודות סטטוס המשחק, מצב כל אחת מהקבוצות וזמינות השחקנים שנותרו זמינים לבחירה. **מידע זה נשמר באופן הבא:**

- מבני נתונים הממפים בין מזהה הקבוצה לבין רשימת השחקנים שנבחרו ע"י הקבוצה עד כה, ורשימת העמדות הפנויות לבחירה לכל קבוצה.
- קבוצה (Set) של מאגר השחקנים הפנויים (לפי Id השחקנים). השיקול בשימוש ב-set הוא ביצוע פעולות חיפוש וגישה בזמן מהיר.
- מידע נוסף למעקב אחר השלבים בדראפט – מזהה הסוכן שתורו לבחור ומספר הבחירה (מתוך כלל הבחירות הדראפט). חשוב לציין כי סדר הבחירות הרנדומלי מתקבל כקלט למשחק ונשמר במצב, אך אינו מיוצר ע"י מודולי המערכת.

מצב במרחב מתואר במערכת באמצעות מודול ייעודי אשר מכיל בנוסף למידע שהוזכר לעיל המתאר את מצב המשחק, את הפונקציונאליות הבאה שמטרתה לפשט את הרצת המשחק –

- **יצירת המצב העוקב למצב הנוכחי** – בהתאם לפעולה של הסוכן הנוכחי (בחירת שחקן ממאגר השחקנים הפנויים), מבצעת את החישובים הנדרשים ליצירת המצב העוקב. חישובים אלו כוללים עדכון רשימת השחקנים הנבחרים והעמדות הפנויות של קבוצת הסוכן, עדכון מאגר השחקנים הפנויים.
- **קבלת הפעולות האפשריות עבור הסוכן** – חישוב רשימת מזהיי שחקנים פנויים שהסוכן שזהו תורו יכול לבחור לקבוצתו. המערכת מספקת, בנוסף לאפשרות לקבל את רשימת כלל השחקנים הפנויים, גם אפשרות לקבל רשימה מצומצמת לפי גודל מבוקש (כדי לשלוט בגודל מקדם הסיעוף), וכן אפשרות לקבל רשימה בגודל מצומצם על פי מיון לפי קטגוריה מסויימת, למשל, עשרת השחקנים הפנויים שמדד ה-Fantasy Rank שלהם הוא הגבוה ביותר.
- **זיהוי מצב סופי** – מתבצע באמצעות השוואה בין מספר הבחירה הנוכחי, לבין מספר הבחירות הכולל הצפוי בדראפט (נתון ידוע מראש בתחילת המשחק). בהינתן שמספר הבחירה הנוכחי שווה למספר הבחירות הכולל, ניתן לקבוע כי מדובר במצב סופי (לא ניתן לבצע לאחריו מהלכים חוקיים נוספים).

### **סימולטור להרצת משחק דראפט**

הסימולטור הינו מעטפת המשחק ומבצע את הפעולות הנדרשות לצורך הרצת הדראפט. בשלב הראשון מתבצע אתחול ה-DB שתואר לעיל, ולאחריו, אתחול המצב ההתחלתי של המשחק, אשר מאופיין ברשימות ריקות לבחירות השחקנים ומאגר השחקנים הפנויים המכיל את כל שחקני ה-NBA באותה עונה. כל עוד המשחק אינו מגיע למצב סופי, נבחר הסוכן שתורו לשחק והוא מבצע את תורו. לפי הפעולה שבחר, מתבצע חישוב של המצב העוקב המתאים.

### **מימוש סוכני משחק הדראפט**

לצורך בחינת אסטרטגיות שונות ואלגוריתמי חיפוש שונים להתמודדות עם משחק הדראפט, כפי שתוארו בפרק תיאור פיתרון הבעיה, פותחו במסגרת הפרויקט מספר סוכנים כאשר כל סוכן מהווה למעשה מימוש של אסטרטגיה או אלגוריתם חיפוש. מסוכן במשחק הפנטזי ישנה למעשה דרישה יחידה, והיא מימוש היכולת לבצע בחירה של פעולה (שחקן) מתוך כלל הפעולות האפשריות להתקדמות בחיפוש (מאגר השחקנים הפנויים). מימוש האסטרטגיה של כל סוכן בוצע במסגרת פונקציית הבחירה שלו.

## מימוש הסוכנים השונים – נקודות חשובות ואתגרים

נפרט להלן את סוגי הסוכנים שפותחו ונסביר כיצד הם מביאים לידי ביטוי את האסטרטגיות להתמודדות עם משחק הפנטזי כפי שתוארו בפרק הקודם. כל סוכן מיוצד לפי שם המעיד על האסטרטגיה לפיה הוא פועל. שמות אלו, יחד עם מאפיינים נוספים או שם ההיוריסטיקה בה משתמש בסוכן באלגוריתם החיפוש, משמשים גם בשלב הצגת תוצאות הניסויים בהמשך.

1. **MonkeyAgent** – סוכן זה פועל ע"פ אסטרטגיית Random שתוארה לעיל, ובכל תור בוחר שחקן אקראי מתוך מאגר השחקנים הפנויים. סוכן זה משמש בעיקר בתהליך הפיתוח ובמסגרת מספר ניסויים בסיסיים.

2. **LabelAgent** – סוכן זה מאותחל עם קטגוריה ספציפית מתוך תשע קטגוריות המשחק, ומשתמש באסטרטגיית Maximize Category שתוארה לעיל.

3. **RandomLabelAgent** – סוכן זה מאותחל עם רשימה של מספר קטגוריות מתוך תשע קטגוריות המשחק ומשתמש באסטרטגיית Random Categories Maximization, על ידי הגרלה של קטגוריה מתוך הרשימה, וביצוע בחירה בדומה לסוכן ה-LabelAgent עבור הקטגוריה הנבחרת.

4. **AlphaBetaAgent** – סוכן זה מבצע חיפוש אלפא-בטא במרחב המצבים בכל תור, ומבצע את הבחירה לפי הצעד המאפשר להבטיח את הערך ההיוריסטי הגבוה ביותר. הסוכן מאותחל עם היוריסטיקה (מבין ההיוריסטיקות שפורטו בפרק תיאור הפיתרון) לפיה משוערכים המצבים במהלך החיפוש. כמו כן, הסוכן מאותחל עם עומק חיפוש ומקדם סיעוף עבור פיתוח עץ משחק מוגבל, מפאת מגבלות זמן הריצה.

5. **ExpectimaxAgent** – סוכן זה מבצע חיפוש במרחב המצבים בכל תור על פי אסטרטגיית אקספקטימקס, כלומר, שיערוך צומת במייצג תור של שחקן יריב מתבצע על ידי חישוב תוחלת הערכים ההיוריסטים של כל אחד מהמצבים העוקבים. גם סוכן זה מאותחל עם היוריסטיקה לפיה משוערכים המצבים, עומק חיפוש ומקדם סיעוף. כמו כן, ניתן לבחור האם להשתמש בסוכן המניח התפלגות אחידה על כל אחד מהצעדים של השחקנים היריבים, או התפלגות התלויה בערך דירוג הפנטזי של כל אחד מהשחקנים שהיריב עשוי לבחור.

6. **MaxiMaxAgent** – סוכן זה מבצע חיפוש במרחב המצבים בכל תור, כאשר בכל "סימלוח" של תור שחקן יריב, מבצע את הבחירה על פי הצעד אשר יאפשר הגעה למצב עם הערך ההיוריסטי הגבוה ביותר, כפי שתואר באינטרפטציה לשיטת Maximax שתוארה בפרק הקודם. בדומה לסוכן החיפוש האחרים גם סוכן ה-Maximax מאותחל עם היוריסטיקה להערכת מצבי מקסימום, ובנוסף, מאותחל גם עם היוריסטיקה (אינה בהכרח זהה) עבור שיערוך מצבים בעת סימלוח תור שחקן יריב. מימוש זה מאפשר בחינת הסוכן תחת הנחה של התמודדות עם יריבים הפועלים באסטרטגיות שונות. כמו כן, גם מקדם הסיעוף בצמתי הסוכן ובצמתי היריב אינו חייב להיות זהה

עבור סוכן זה. בנוסף חשוב לציין כי עומק החיפוש איתו מאותחל הסוכן מתייחס **למספר סיבובים במשחק הדראפט**, כלומר, מספר התורות של הסוכן, ועל כן מאפשר בניית אסטרטגיה לטווח ארוך יותר.

7. **LocalSearchAgent** – סוכן זה פועל באסטרטגיית Steepest Ascent, כלומר, מבצע את הבחירה אשר תוביל למצב בעל הערך ההיוריסטי הגבוה ביותר. הסוכן מאותחל עם פונקציית היוריסטיקה לפיה משוערכים מצבים. כמו כן, גם סוכן זה מאותחל עם מקדם סיעוף מוגבל, אך גבוה בהרבה מזה שניתן לתת לסוכנים המשתמשים בחיפוש לעומק.

8. **HybridAgent** – סוכן זה מאפשר מימוש אסטרטגיות היברידיים. הסוכן מאותחל עם רשימה של סוכנים, ובכל תור, בוחר את האסטרטגיה (סוכן) לפיה ירצה לפעול, בהתאם לפונקציית החלטה מוגדרת מראש. ניתן להשתמש בסוכן זה יחד עם פונקציות החלטה פשוטות, כגון התחשבות במספר הבחירה בלבד, אך ניתן גם לשלב פונקציות בחירה מורכבות יותר אשר עשויות להשפיע על הישגיו של הסוכן.

## מימוש ההיוריסטיקות ופונקציות יעילות (Utility)

### פונקציות היעילות

הפונקציות המיועדות לחישוב הערך הכולל של תוצאת משחק מרוכזות מודול ייעודי. כפי שתואר בפרק תיאור הפיתרון לבעית הדראפט, המערכת מאפשרת בחינה של התוצאות אל מול שתי פונקציות יעילות, אשר נלקחו מעולם המשחק ומהאופי בו שחקני הפנטזי אומדים את טיב הקבוצה שלהם – **מדד יעילות Wins**, המציג דירוג יחסי של קבוצות הליגה בהתאם למספר הקבוצות הכולל עליהן הקבוצה "גוברת" ב-matchup הישיר (יתרון בלפחות 5 קטגוריות), ו**מדד יעילות Rank**, המציג דירוג יחסי של קבוצות הליגה בהתאם למיקום היחסי של הקבוצה בכל אחת מתשע קטגוריות המשחק.

לכל אחת מהפונקציות הנ"ל ניתנו שתי גירסאות – גירסא המחזירה דירוג מלא של כלל קבוצות הליגה לפי המדד, וגירסא המחזירה עבור סוכן ספציפי את מיקום קבוצתו על פי הדירוג. הגירסא השניה נחוצה לצורך הערכה מוחלטת של מצב במרחב המצבים של המשחק, בעוד הגירסא הראשונה נועדה בעיקר לצורך בחינת התוצאות באופן ידני.

### פונקציות היוריסטיות

הפונקציות המשמשות את הסוכנים הפועלים באמצעות חיפוש היוריסטי רוכזו גם הן במודול ייעודי, כך שיתאפשר שימוש פשוט של כל אחד מהסוכנים בכל אחת מההיוריסטיקות שרצינו לבחון.



מכיוון שההיוריסטיקות המרכזיות בהן השתמשנו זהות למעשה לפונקציות היעילות, מימוש ההיוריסטיקות מבוסס על מימוש פונקציות היעילות ואף עושה בהן שימוש ישיר. בנוסף, המודול מכיל היוריסטיקה שאינה מבוססת על פונקציות היעילות – **היוריסטיקת Advantage**, המחשבת את דירוג הקבוצה לפי ממוצע משוקלל של המרחק של הקבוצה מיתר הקבוצות בכל אחת מתשע הקטגוריות.

## **אתגרים במימוש פונקציות ההערכה**

- **זמן ריצה ארוך** – במהלך הרצת הניסויים ובחינת ביצועי הסוכנים השונים עם שלל ההיוריסטיקות, **נתקלנו בזמני ריצה ארוכים שהושפעו מאופן המימוש הראשוני של החישובים**. מרבית הפונקציות כללו חישובים אשר דרשו סינון השחקנים שנבחרו ע"י כל אחת מהקבוצות (לפי הרשימות המאוכסנות באובייקט מצב המשחק), סכימה של ההישגים בכלל הקטגוריות ומיון חוזר של רשימת הקבוצות על פי תוצאות הסכימה. פעולות החישוב הללו חוזרות מספר רב של פעמים במהלך ריצת המשחק, בכל תור של סוכן המשתמש בחיפוש היוריסטי, ולכן, ישנה חשיבות גבוהה להשגת ביצועים טובים ושיפור יעילות המימוש. **נוכחנו לגלות כי שימוש במילונים**, אשר הוגדרו בשלב בניית ה-DB, **משפר משמעותית את זמן הריצה בעת חישוב הפונקציות**, למשל, על פני ביצוע פעולות על גבי ה-Data Frame שמכיל את תוצאות כלל השחקנים. שיטה זו הינה מעט מורכבת יותר, אך נבחרה לבסוף עקב השיפור המשמעותי בזמן הריצה.
- **התייחסות לקטגוריות "שליליות"** – קטגוריית האיבודים, שהינה אחת מתשע הקטגוריות הנמדדות לקביעת טיבן של כל אחת מהקבוצות בליגת הפנטזי, הינה למעשה קטגוריה שלילית – ככל שבקבוצה משחקים שחקנים שמאבדים יותר כדורים, כך התוצאה **פחות** טובה. קטגוריה זו שונה משאר הקטגוריות הנמדדות, בהן ככל שההישג גדול יותר כך התוצאה טובה יותר, ולכן דרשה התייחסות מיוחדת בתהליך הפיתוח.
- **חישוב קטגוריות האחוזים** – בשונה מיתר הקטגוריות, **FG% ו-FT% אינן ניתנות לסכימה ומיצוע על פני התוצאה שהשיגו כלל שחקני הקבוצה בקטגוריה**. לפיכך, נדרשת שיטה אחרת לחישוב מדויק של אחוז הקליעה של כל שחקני הקבוצה, בהסתמך על נתונים אחרים. **הוחלט לבצע את החישוב באופן הבא** – סכימת מספר הקליעות של כל שחקן בקבוצה וסכימת מספר הזריקות של כל שחקן וחישוב היחס ביניהן. בצורה זו מתקבל אחוז הקליעה המדויק של שחקני הקבוצה (באופן זהה לאופן בו מחושבת הקטגוריה במשחק הפנטזי האמיתי). חשוב לציין כי לצורך חישוב ההיוריסטיקה עבור קטגוריות אלו, נדרש שימוש במערכת החיזוי (מחלק 1 של הפרויקט) בכדי לחזות גם את הקטגוריות FT, FTA, FG, FGA (קליעות וזריקות משני הטווחים), ולצורך כך השתמשנו באותן מערכות חיזוי שנבחרו עבור הקטגוריות FG% ו-FT% בהתאמה.

- **נרמול התוצאות** – עבור ההיוריסטיקה המחשבת את מיקום הקבוצה לפי המרחק היחסי מכל אחת מהקבוצות האחרות בכל קטגוריה (היוריסטיקת Advantage), **ישנה חשיבות מכרעת בנרמול התוצאות לפני ביצוע הסכימה וחישוב המרחק, עקב ההבדל בערכי המקסימום והמינימום בכל אחת מקטגוריות המשחק.** למשל, כמות הנקודות המקסימלית ששחקן NBA קולע למשחק היא כ-30 נק' למשחק. לפיכך, המרחק בין סך הנקודות שקבוצת שחקנים אחת קולעת למשחק, לבין קבוצה אחרת, עשוי להיות בסדר גודל של כמה עשרות. לעומת זאת, הערך המקסימלי למשחק בקטגוריית החטיפות למשל הינו כ-2.5 חטיפות למשחק, ולפיכך המרחק בקטגוריה זו בין הקבוצות סביר שיהיה מספר בודד של חטיפות. בעת חישוב ההיוריסטיקה הנ"ל, ברצוננו לתת משקל זהה לכל אחת מקטגוריות המשחק, ולכן, נדרשנו לבצע נרמול לערכים טרם ביצוע הסכימה וחישוב המרחק.

## ניסויים למערכת סוכן משחק הדראפט

### מתודולוגיה ניסויית

מטרת העל בחלק זה של הפרויקט הינה פיתוח סוכן אוטונומי עבור הרכבת קבוצת שחקנים במסגרת משחק הפנטזי דראפט. לצורך פיתוח המערכת הוגדרו מספר סוכנים, הפועלים בשיטות ועל פי אלגוריתמים שונים. במסגרת הניסויים נבחנו כלל הסוכנים במגוון תרחישים של משחק הדראפט, אשר נועדו לאפשר ניתוח ובחירה מושכלים של סוכן מועדף עבור משחק דראפט גנרי (מספר קבוצות, גודל קבוצה, אילוצי עמדות ומיקום בחירה כלשהם). בנוסף, הוגדרו מספר מטרות משנה אשר יפורטו להלן.

עבור השגת מיפוי ודירוג כללי של הסוכנים (מטרת העל), חולקו הניסויים לשלושה שלבים –

1. **ניסויים סטנדרטיים** – בחינה ראשונית של הסוכנים השונים לצורך סימון אסטרטגיות מובילות.
2. **ניסויים מתקדמים** – בחינת סוכנים נבחרים בתרחישים מתקדמים ומציאותיים.
3. **ניסוי מסכם** – השוואת סוכנים מובילים זה מול זה בתרחיש משחק פנטזי מלא.

בנוסף, נערכו ניסויים עבור מטרות המשנה הבאות –

1. **מיפוי Agent Per Pick** – בהינתן מיקום הבחירה בדראפט, מענה על השאלה – באיזו אסטרטגיה עדיף לפעול. מיפוי זה התבצע ללא הרצת ניסויים נוספים, ועל סמך תוצאות איטרציות חוזרות של הניסוי המסכם, כפי שיפורט בהמשך.
2. **מיפוי Pick Per Agent** – בהינתן סוכן בעל אסטרטגיה מסויימת, מענה על השאלה – מהו מיקום הבחירה העדיף עבור הסוכן. מיפוי זה התבצע ע"י הרצת ניסויים במסגרתם התחרו סוכנים מאותו סוג במשחק דראפט הכולל 10 קבוצות, למשך מספר איטרציות.

## מדדים ושיטות לבחינת ביצועי הסוכן

תוצאות הסוכנים נמדדו אחד מול השני – לא קיימת אמת מידה חיצונית מוחלטת (כגון חיזוי המומחים החלק 1 של הפרויקט), אשר לפיה ניתן לאמוד את תוצאותיו של סוכן נתון. על כן, בכדי לבחון את ביצועי הסוכנים, הוגדרו מדדי "יעילות" יחסיים עבור תוצאה סופית של סימולציית משחק דראפט – מדד Wins ומדד Ranks (להרחבה ראו פרק "הערכת תוצאות משחק הדראפט").

במסגרת משחק הדראפט קיים אלמנט רנדומלי – סדר הבחירה בין הקבוצות השונות מוגרל בתחילת המשחק. על כן, תוצאות הרצה של סימולציה יחידה עשויות להיות מוטות, כלומר, מושפעות ממיקום הבחירה האקראי שהוגרל לסוכן אותו בוחנים. בכדי לנטרל השפעה זו, כל ניסוי מורכב ממספר איטרציות, השווה למספר הקבוצות המשתתפות במשחק, כאשר כל איטרציה היא משחק דראפט מלא בין הקבוצות.

מדדי Wins ו-Ranks מספקים ערך מספרי הנע בין 0 למספר הקבוצות בדראפט, בהתאם למיקום הסוכן הנבחן על פי המדד. בכדי לקבל תוצאה אחידה בכל הניסויים, אשר אינה תלויה במספר הקבוצות בניסוי הספציפי, התבצע נירמול של תוצאות המדדים (הדירוג) המתקבל, לסקאלה בין 0 ל-1 (למשל, סוכן שסיים חמישי מבין עשר קבוצות לפי מדד מסויים, יקבל ניקוד 0.5 עבור מדד זה).

## שלב הניסויים סטנדרטיים

בשלב זה נערכו 17 ניסויים שונים, כאשר כל ניסוי מורכב מיריבים מאותו טיפוס של סוכן (למשל ניסוי שכל היריבים בו משתמשים באסטרטגיית Alpha Beta). לשלב זה הוגדרו בסה"כ 16 סוכנים, המבוססים על הסוכנים שתוארו בתת הפרק "מימוש סוכני משחק הדראפט". לכלל הסוכנים הוגדרו פרמטרים בסיסיים (כגון, עומק חיפוש ומקדם סיעוף), בעוד שעבור כל סוכן המשתמש בחיפוש היוריסטי, הוגדרו שלוש קונפיגורציות שונות לבחינה, עבור כל אחת מההיוריסטיקות הבאות – Quorum Wins, Rank per Advantage on Opponent-ו-Category. את הסוכנים שנבחנו בשלב זה ניתן למצוא בנספח ד'.

כל ניסוי מוגדר על פי הפרמטרים הבאים – סוג היריבים, מספר הקבוצות, גודל קבוצה, אילוצי עמדות וסוג הסוכן הנבחן. ניתן למצוא את פירוט הניסויים שנערכו בשלב זה בנספח ד'.

## הפלט המתקבל מכל ניסוי בשלב זה כולל את התוצרים הבאים –

- **טבלה מסכמת** – עבור כל סוכן מופיע הניקוד הממוצע שהשיג בניסוי על פני כל האיטרציות, בשני המדדים, וכן ניקוד ממוצע מנורמל, בכדי לנטרל את השפעת מספר הקבוצות בניסוי. כמו כן, בטבלה מוצג זמן הריצה של הניסוי, נתון אשר איפשר לסמן נקודות חולשה במימוש הניסויים, ולשפרם לקראת השלבים הבאים.
- **טבלת תוצאות בחלוקה לאיטרציות** – כל שורה בטבלה מציגה את תוצאות איטרציה בודדת עבור סוכן ספציפי (מיקום במשחק הספציפי על פי שני המדדים). בנוסף, הטבלה כוללת את מיקום הסוכן בסדר הבחירות באיטרציה הספציפית.

## שלב הניסויים המתקדמים

בשלב זה נערכו חמישה ניסויים המשלבים מספר רב של קבוצות ממגוון סוכנים יריבים בעלי אסטרטגיות שסומנו כמתקדמות בשלב הראשון. ניסויים אלו כללו גם כן גודל קבוצה גדול יותר מאשר בניסויים הסטנדרטיים, במטרה לדמות ככל הניתן משחק זראפט אמיתי. חלק משמעותי במשחק הדראפט הינו התמודדות בביצוע בחירות נכונות בשלבים מאוחרים, כאשר היצע השחקנים הזמינים הינו מוגבל, ויש להתאים לאופי הקבוצה שהורכבה עד כה. לשם כך, נדרשו ניסויים רחבי היקף, אשר יחייבו את הסוכנים לבחור שחקנים מדירוג ממוצע ומטה.

במסגרת שלב זה נבחנו 11 סוכנים נבחרים, אשר הציגו ביצועים טובים בשלב הניסויים הסטנדרטיים. בנוסף, התבצע טיוב של הפרמטרים (עומק חיפוש, מקדם סיעוף וכד'), כך שיתקבלו סוכנים "חזקים" יותר, באופן תיאורטי. הפלט המתקבל מניסויים אלו זהה לפלט שהתקבל עבור הניסויים הסטנדרטיים. את תיאור הסוכנים והניסויים המלא ניתן למצוא בנספח ה'.

## שלב הניסוי המסכם

שלב זה כלל ניסוי יחיד אשר הורכב מ-11 קבוצות בגודל 12 שחקנים. כל קבוצה ייצגה סוכן שונה, מבין הסוכנים הזהים לאלו שנבחנו בשלב הניסויים המתקדמים (כאשר לחלקם התבצע טיוב קל של הפרמטרים). באמצעות ניסוי זה ניתן לקבל דירוג סופי ומוחלט של כלל האסטרטגיות המובילות במשחק נתון. תוצאות ניסוי זה כוללות גם כן טבלה מסכמת עבור הניסוי, ובנוסף, טבלה המציגה את תוצאות כל איטרציה בניסוי. הטבלה השניה משמשת בנוסף למיפוי ה-Agent Per Pick, כיוון שניתן באמצעותה למדוד את ההישג הממוצע של כל סוג סוכן במשחק כללי. לצורך בניית המיפוי הנ"ל נדרש מספר רב של איטרציות, ולשם כך הורץ הניסוי המסכם ארבע פעמים.

## שלב הניסויים הסטנדרטיים – תוצאות ומסקנות

בחלק זה נערכו ניסויים בסיסיים רבים. נציג להלן דוגמא לתוצרים שהתקבלו מניסוי ספציפי, ולבסוף, תמונה כוללת של תוצאות הניסויים והמסקנות העולות מהן. התוצרים המלאים לכלל הניסויים שנערכו בשלב זה נמצאים תחת התיקיה "AI-Project-NBA-Fantasy\Draft\Experiments\Standard" בפרויקט שהוגש.

### תוצאות ניסוי לדוגמא

להלן דוגמא לתוצרים שהתקבלו עבור ניסוי מספר 14, בו כל היריבים הינם מסוג Expectimax Agent המשתמש בהיוריסטיקה Advantage on Opponent –

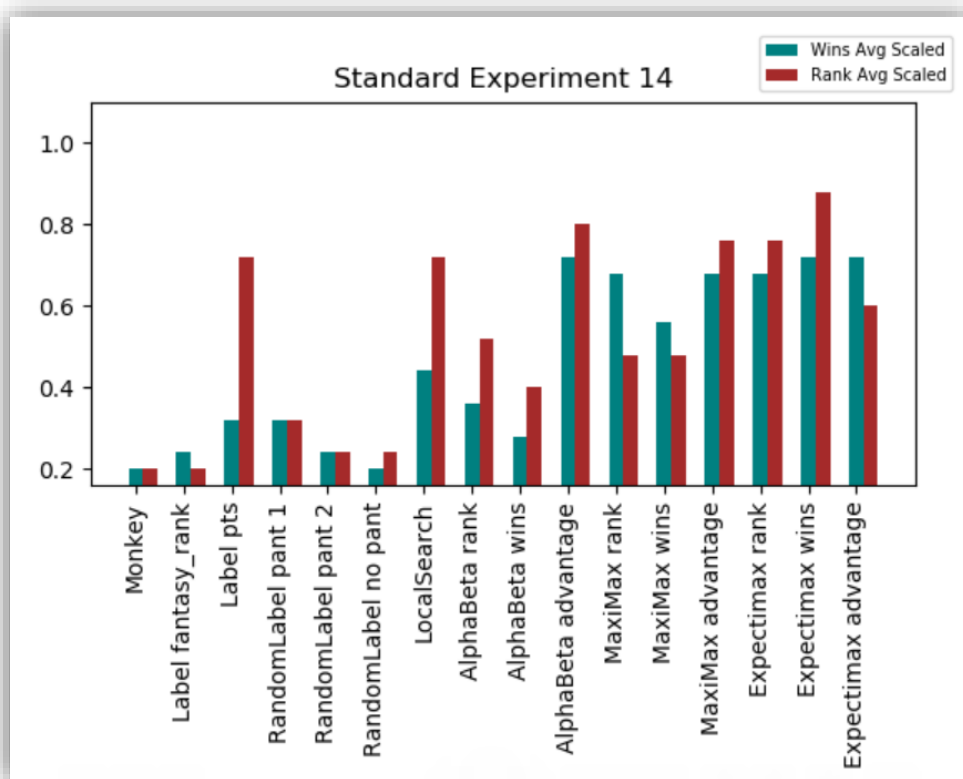
#### 1. הטבלה המסכמת את תוצאות הניסוי –

Agent	Type	Wins Avg Score	Wins Avg Scaled	Rank Avg Score	Rank Avg Scaled	Number of Teams	Run Time
MonkeyAgent		1	0.2	1	0.2	5	00:00:07
LabelAgent	fantasy_rank	1.2	0.24	1	0.2	5	00:00:05
LabelAgent	pts	1.6	0.32	3.6	0.72	5	00:00:05
RandomLabelAgent	punt 1 <sup>11</sup>	1.6	0.32	1.6	0.32	5	00:00:06
RandomLabelAgent	punt 2 <sup>12</sup>	1.2	0.24	1.2	0.24	5	00:00:07
RandomLabelAgent	no punt	1	0.2	1.2	0.24	5	00:00:06
LocalSearchAgent		2.2	0.44	3.6	0.72	5	00:00:06
AlphaBetaAgent	rank	1.8	0.36	2.6	0.52	5	00:00:53
AlphaBetaAgent	wins	1.4	0.28	2	0.4	5	00:00:50
AlphaBetaAgent	advantage	3.6	0.72	4	0.8	5	00:00:08
MaxiMaxAgent	rank	3.4	0.68	2.4	0.48	5	00:01:33
MaxiMaxAgent	wins	2.8	0.56	2.4	0.48	5	00:01:31
MaxiMaxAgent	advantage	3.4	0.68	3.8	0.76	5	00:00:09
ExpectimaxAgent	rank	3.4	0.68	3.8	0.76	5	00:03:42
ExpectimaxAgent	wins	3.6	0.72	4.4	0.88	5	00:03:46
ExpectimaxAgent	advantage	3.6	0.72	3	0.6	5	00:00:07

<sup>11</sup> **Punt 1** – סוכן ה-Random Label הנ"ל הוגדר ע"י סט הקטגוריות הבא שמטרתו לייצר אסטרטגיית punt – PTS, AST, 3P, FT%, STL, Fantasy Rank

<sup>12</sup> **Punt 2** – סוכן ה-Random Label הנ"ל הוגדר ע"י סט הקטגוריות הבא שמטרתו לייצר אסטרטגיית punt – PTS, TRB, FG%, BLK, Fantasy Rank

2. גרף המציג את סיכום תוצאות הניסוי –



איור 10: תוצאות ניסוי סטנדרטי 14 לפי מדדי Wins ו-Ranks

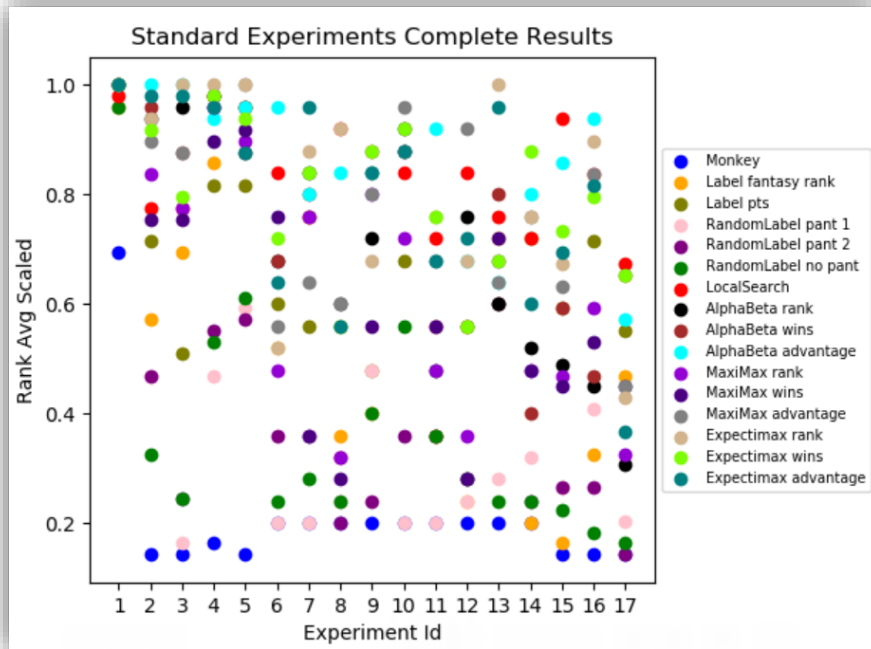
3. מהניסוי הנ"ל, בדומה לשאר הניסויים בשלב זה, ניתן ליצור הבחנה בין הסוכנים השונים, ולסמן בצורה ברורה סוכנים מובילים. בפרט בניסוי זה, ניתן לראות כי הסוכנים המשתמשים באסטרטגיית Expectimax משיגים תוצאות טובות יחסית בשני המדדים, כמו גם הסוכן המשתמש באלגוריתם Alpha Beta עם היוריסטיקת Advantage on Opponent. בנוסף, ניתן לראות בבירור כי הסוכנים שאינם משתמשים בחיפוש היוריסטי משיגים תוצאות נמוכות בהרבה, על אף הניסיון ליצור אסטרטגיית Punt עבור חלק מהסוכנים הללו.

4. סוג הניסויים בשלב זה, המורכבים כולם מיריבים מאותו טיפוס, מבליט מגמות המאפיינות סוכנים ספציפיים. לדוגמה בניסוי המוצג, כלל היריבים משתמשים באלגוריתם Expectimax עם היוריסטיקת Advantage on Opponent. התוצאות של כלל הסוכנים מול יריבים אלו הינן נמוכות יחסית מאשר בניסויים האחרים, ועל כן זוהי עדות נוספת לטיבו של הסוכן הנ"ל.

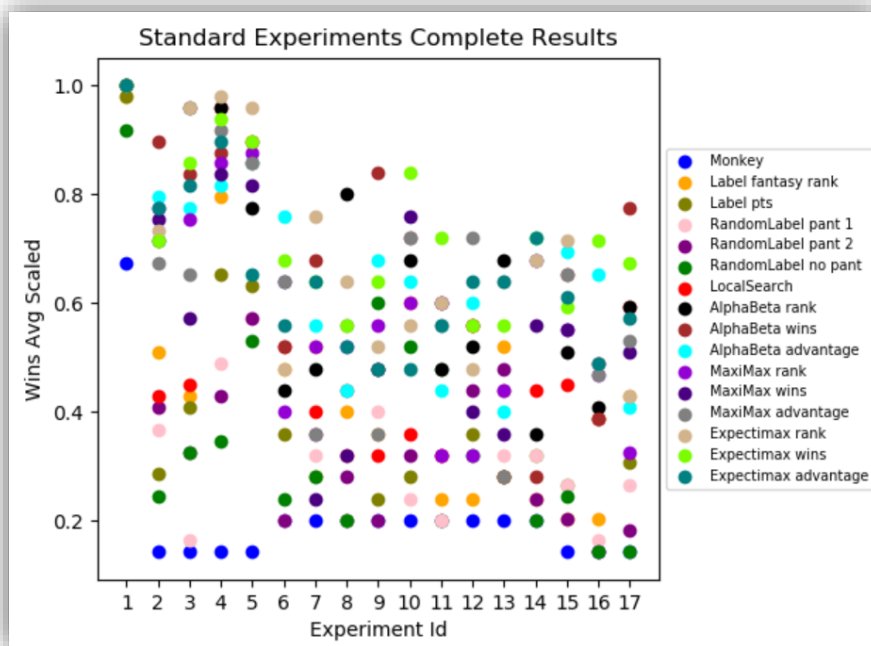
## תוצאות כוללות – ניתוח ומסקנות

1. דירוג כלל הסוכנים בכלל הניסויים הסטנדרטיים –

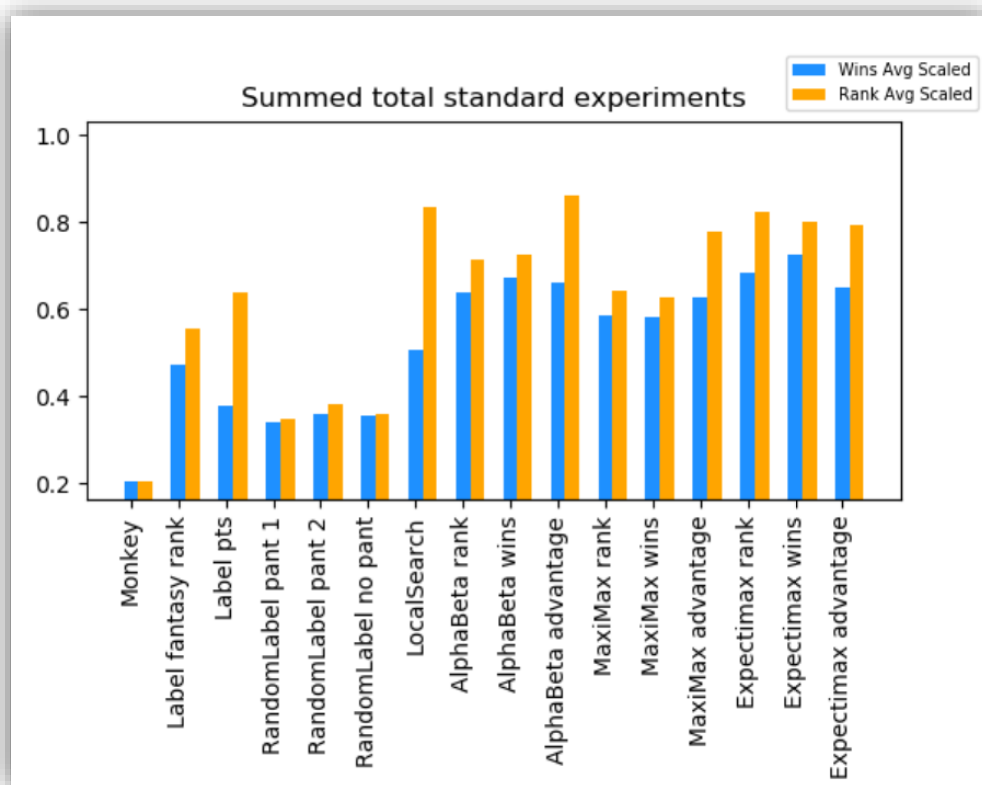
א. לפי מדד Ranks –



ב. לפי מדד Wins –



2. גרף סיכום התוצאה הממוצעת של כל אחד מהסוכנים בכלל הניסויים הסטנדרטיים –



3. נקודות ומסקנות מרכזיות העולות מתוצאות הניסויים בשלב זה –

א. יתרון מובהק לסוכנים המשתמשים באסטרטגיות המבוססות חיפוש היוריסטי – בדומה למסקנה העולה מהדוגמא הפרטית שהוצגה, גם מהתוצאות הכוללות ניתן לראות באופן מובהק כי סוכני ה-Alpha Beta, Maximax, Expectimax וה-Local Search משיגים תוצאות טובות בהרבה משאר הסוכנים, בכלל התרחישים שנבחנו. גם מהסתכלות על הניסויים בהם סוכנים אלו שימשו כיריבים, ניתן להבחין כי התוצאות הכוללות בניסויים אלו נמוכות מיתר הניסויים, נתון המחזק מסקנה זו.

ב. קושיי בביצוע הפרדה בין הסוכנים המובילים – על סמך התוצאות מניסויים אלו לא ניתן להפריד ולסמן אסטרטגיות מובילות מוחלטות אשר השיגו תוצאות גבוהות בכלל הניסויים. על כן, מספר רב של סוכנים נבדקו גם בשלב הניסויים המתקדמים. עם זאת, כן ניתן להבדיל עבור חלק מהסוכנים מכווני היוריסטיקה בין ההיוריסטיקות היעילות יותר, לדוגמא סוכן ה-Maximax עם היוריסטיקת Advantage on Opponent שמשיג תוצאות טובות יותר מאשר עם שימוש בהיוריסטיקות האחרות.



בפרט, ניכר כי ההיוריסטיקה Advantage on Opponent הינה אפקטיבית במיוחד, כפי שניתן היה לצפות.

ג. **אסטרטגיות שאינן מבוססות חיפוש היוריסטי** – ניתן לסנן באופן מוחלט את כלל האסטרטגיות שלא משתמשות בחיפוש היוריסטי, כיוון שהסוכנים הללו השיגו תוצאות נמוכות יחסית בכלל הניסויים. עם זאת, הבחנה מעניינת מתקבלת מהסתכלות על תוצאות הסוכן הפועל בשיטת מיקסום ערך ה-Fantasy Rank – סוכן זה משיג תוצאות סבירות, פרט המחזק את הבחירה שנעשתה בתהליך הפיתוח לפיה הגבלת מקדם הסיעוף של מרבית הסוכנים מתבצע על ידי מיון השחקנים הזמינים על פי ערך דירוג הפנטזי החזוי שלהם.

### שלב הניסויים המתקדמים – תוצאות ומסקנות

שלב זה כלל מספר מצומצם של ניסויים רחבים הכוללים מגוון יריבים (מתוך אלו שסומנו כסוכנים עדיפים) ומספר קבוצות ושחקנים גדול. גם עבור שלב זה נציג תוצאות לדוגמא מתוך אחד הניסויים, ולבסוף תמונה כללית ומסקנות. את התוצרים המלאים ניתן למצוא תחת התיקיה "AI-Project-NBA-FantasyDraftExperiments\Advanced בפרויקט המוגש.

#### תוצאות ניסוי לדוגמא

להלן דוגמא לתוצרים שהתקבלו עבור ניסוי מספר 5, בו היריבים מורכבים מזוגות סוכנים ממגוון אסטרטגיות החיפוש ההיוריסטי – AlphaBeta, Maximax, Expectimax ו-LocalSearch –

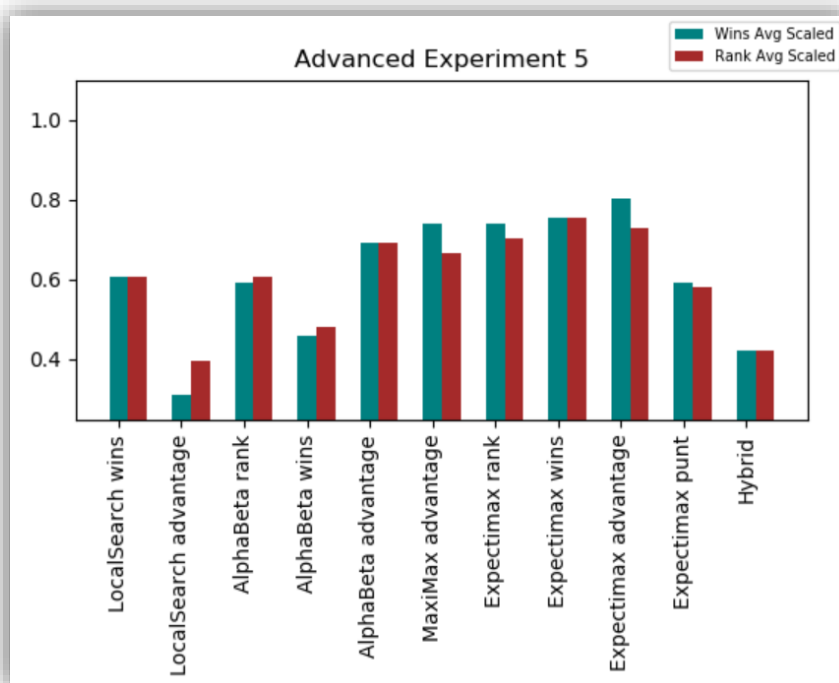
#### 1. הטבלה המסכמת את תוצאות הניסוי –

Agent	Type	Wins Avg Score	Wins Avg Scaled	Rank Avg Score	Rank Avg Scaled	Number of Teams	Run Time
LocalSearchAgent	wins	5.444	0.605	5.444	0.605	9	00:02:01
LocalSearchAgent	advantage	2.778	0.309	3.556	0.395	9	00:01:22
AlphaBetaAgent	rank	5.333	0.593	5.444	0.605	9	00:05:19
AlphaBetaAgent	wins	4.111	0.457	4.333	0.481	9	00:03:10
AlphaBetaAgent	advantage	6.222	0.691	6.222	0.691	9	00:01:58
MaxiMaxAgent	advantage	6.667	0.741	6.000	0.667	9	00:01:39
ExpectimaxAgent	rank	6.667	0.741	6.333	0.704	9	00:24:08
ExpectimaxAgent	wins	6.778	0.753	6.778	0.753	9	00:09:47
ExpectimaxAgent	advantage	7.222	0.802	6.556	0.728	9	00:02:00
ExpectimaxAgent	punt <sup>13</sup>	5.333	0.593	5.222	0.580	9	00:01:47
HybridAgent <sup>14</sup>	fantasy-rank-2, expectimax-4, local-search	3.778	0.420	3.778	0.420	9	00:01:51

<sup>13</sup> עבור סוכן זה התבצע ניסיון להגדיר אסטרטגיית Punt, ע"י העברת סט קטגוריות מסויים לפיו מחושבת ההיוריסטיקה Advantage on Opponent. סט הקטגוריות שנבחר הינו – PTS, TRB, BLK, AST, 3P, STL.

<sup>14</sup> סוכן היברידי אשר פועל באסטרטגיה הבאה – בשני הסיבובים הראשונים בוחר את השחקן הפנוי עם דירוג הפנטזי הגבוה ביותר, בארבעת הסיבובים הבאים פועל באסטרטגיית Expectimax ובהמשך באסטרטגיית Local Search.

2. גרף המציג את סיכום תוצאות הניסוי –



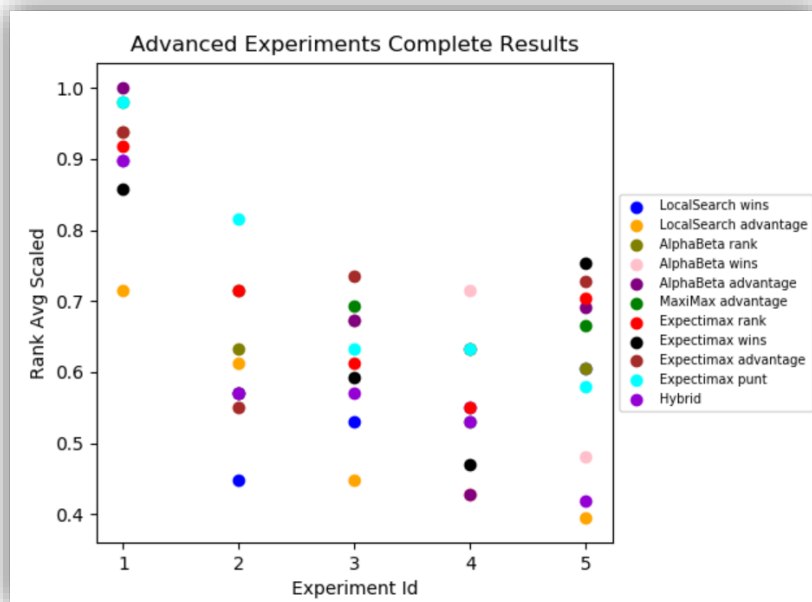
איור 11: תוצאות ניסוי מתקדם 5 לפי מדדי Wins ו-Ranks

3. תוצאות ניסוי זה, בדומה לשאר הניסויים שנערכו בשלב זה, גם כן אינן מציגות מסקנות חד משמעיות. ניתן להבחין כי תוצאות סוכני ה-Expectimax ממשיכים להיות טובות במקצת מיתר הסוכנים, וכי סוכן ה-Local Search, אשר לא מבצע חיפוש לעומק, משיג תוצאות נמוכות יותר.

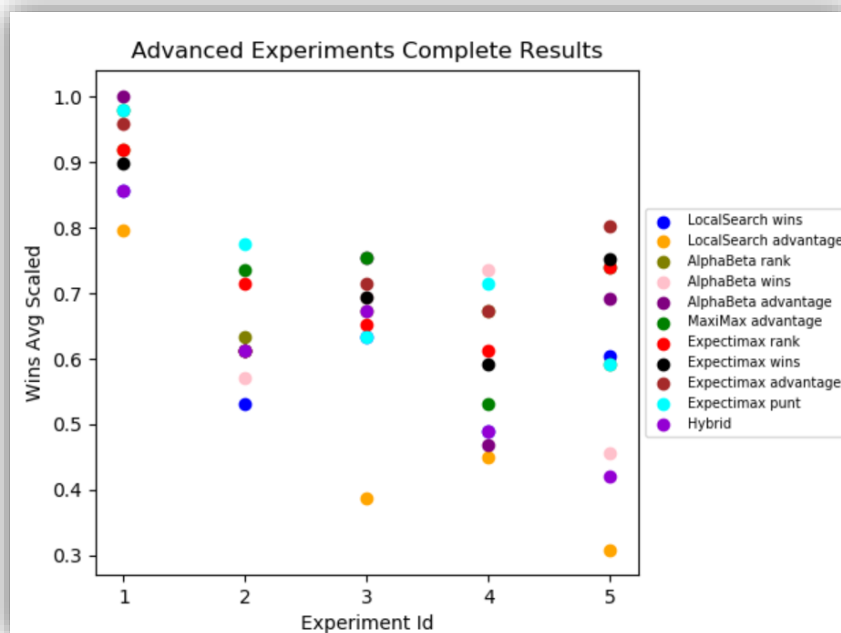
## תוצאות כוללות – ניתוח ומסקנות

1. דירוג כלל הסוכנים בכלל הניסויים המתקדמים –

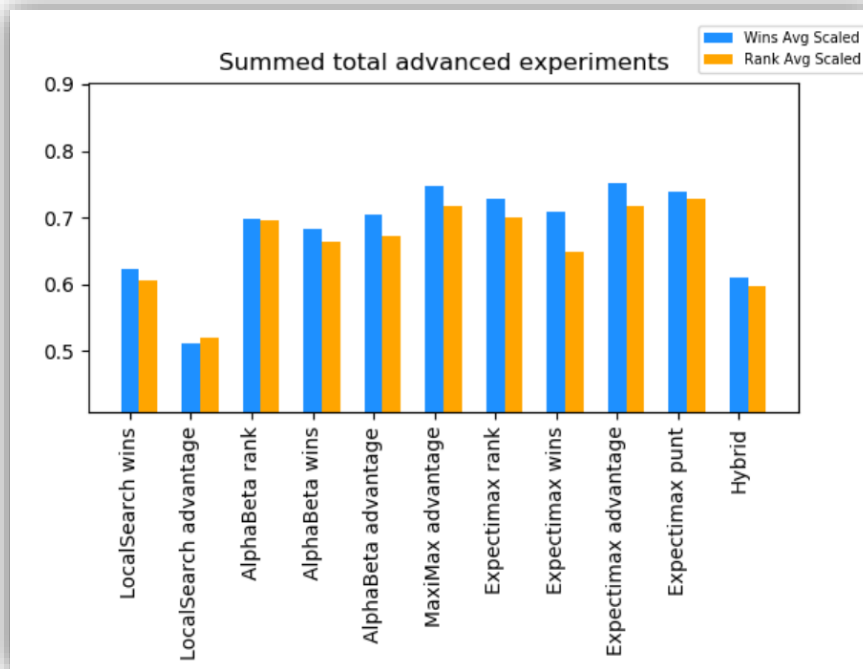
א. לפי מדד Ranks –



ב. לפי מדד Wins –



2. גרף סיכום התוצאה הממוצעת של כל אחד מהסוכנים בכלל הניסויים המתקדמים –



3. נקודות ומסקנות מרכזיות העולות מתוצאות הניסויים בשלב זה –

א. **לא ניתן לקבוע איזה סוכן הוא הטוב ביותר** – גם על פי תוצאות שלב זה לא ניתן לבצע הפרדה מוחלטת ולסמן את האסטרטגיות המובילות ביותר. כלל הסוכנים המשתמשים בחיפוש היוריסטי **לעומק** משיגים תוצאות יחסית זהות בשני המדדים.

ב. **איכות ביצועי הסוכנים** – ניתן לראות כי מרבית הסוכנים משיגים תוצאה ממוצעת גבוהה מ-0.5. פרט זה מעיד על כך **שכללם מצליחים לגבור על יריביהם במגוון התרחישים שנבחנו**.

ג. **סוכן ה-Hybrid** – סוכן זה לא נבחן בשלב הניסויים הסטנדרטיים, עקב השאיפה לסמן אסטרטגיות מובילות בשלב זה ולבנות אותו על פיהן לשלב הניסויים המתקדמים. עם זאת, הסוכן ההיברידי לא השיג את התוצאות הצפויות ותוצאותיו נופלות משל שאר הסוכנים. ייתכן כי הכישלון הנ"ל נובע מבחירת אסטרטגיה לא אפקטיבית. מספר שילובי האסטרטגיות השונים שניתן להגדיר עבור סוכן זה היו גדול מאוד, ולא קיים אמצעי מכווין לבחירת אסטרטגיה יעילה, פרט לידע מקדים על אופי משחק הדראפט.

## שלב הניסוי המסכם – תוצאות ומסקנות

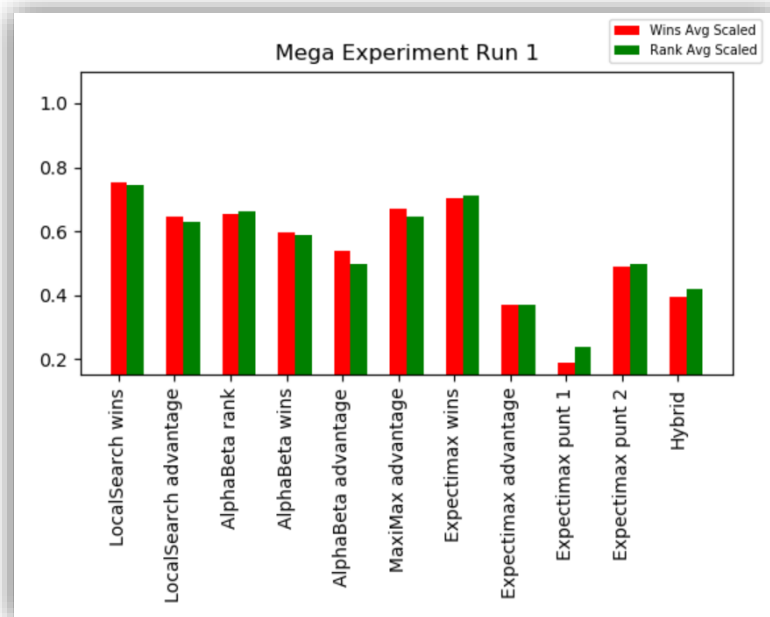
שלב זה כלל ניסוי יחיד אשר הורץ במספר איטרציות, ותוצאותיו מהוות דירוג כללי של הסוכנים המובילים, אשר סומנו בשלבי הניסויים הקודמים. עם זאת, יש לקחת את תוצאות הניסוי המסכם בעירבון מוגבל, כיוון שהוא בוחן את הסוכנים בתרחיש אחד ספציפי, לעומת הניסויים הקודמים אשר בחנו את הסוכנים במגוון תרחישים, וייתכן ומעידים באופן נרחב יותר על טיבו של סוכן כזה או אחר לתרחיש משחק דראפט כללי. המסקנות בחלק זה הינן התייחסות כוללת למסקנות העולות מכלל הניסויים לבחינת הסוכנים השונים.

### להלן תוצאות הניסוי המסכם –

#### 1. הטבלה המסכמת את תוצאות הניסוי –

Agent	Type	Rank Avg Scaled	Rank Avg Score	Wins Avg Scaled	Wins Avg Score
LocalSearchAgent	wins	0.744	8.182	0.752	8.273
LocalSearchAgent	advantage	0.628	6.909	0.645	7.091
AlphaBetaAgent	rank	0.661	7.273	0.653	7.182
AlphaBetaAgent	wins	0.587	6.455	0.595	6.545
AlphaBetaAgent	advantage	0.496	5.455	0.537	5.909
MaxiMaxAgent	advantage	0.645	7.091	0.669	7.364
ExpectimaxAgent	wins	0.711	7.818	0.702	7.727
ExpectimaxAgent	advantage	0.372	4.091	0.372	4.091
ExpectimaxAgent	punt 1	0.240	2.636	0.190	2.091
ExpectimaxAgent	punt 2	0.496	5.455	0.488	5.364
HybridAgent		0.421	4.636	0.397	4.364

#### 2. גרף המציג את סיכום תוצאות הניסוי –



### 3. מסקנות כלליות משלב הניסויים כלליים –

א. **לא התקבלה הכרעה חד משמעית** – מבחינה מעמיקה של כלל הניסויים שנערכו בשלושת השלבים, לא ניתן להגיע להכרעה חד משמעית בדבר אסטרטגיה מובילה למשחק הדראפט. עם זאת, עולות המסקנות הבאות –

i. אסטרטגיות מבוססת חיפוש היוריסטי מתאימות להתמודדות במשחק הדראפט – כלל הסוכנים מבוססי החיפוש ההיוריסטי השיגו תוצאות טובות ויחסית זהות.

ii. ההיוריסטיקות שהוגדרו עבור פיתרון הבעיה מהוות מדד הערכה אפקטיבי למצבי הבעיה בתהליך החיפוש. מעניין לציין כי שתי ההיוריסטיקות המבוססות על מדדי היעילות נמצאו אפקטיביות כמעט באותה מידה בעת בחינת התוצאות תחת שני המדדים השונים.

ייתכן כי מחקר מעמיק בתרחישי משחק ספציפיים יאפשר לבדד יתרונות של סוכנים מסויימים על פני האחרים. נדרשים ניסויים מעמיקים יותר בכדי ליצור הפרדה כזו בין הסוכנים.

ב. **הצלחת סוכן ה-Local Search בניסוי המסכם** – בשונה משלבי הניסויים הקודמים, **בניסוי המסכם השיג סוכן ה-Local Search עם היוריסטיקת Quorum Wins את התוצאות הטובות ביותר** על פי שני המדדים. תוצאה זו מבססת את הטענה כי הניסוי המסכם אינו מייצג באופן מוחלט את דירוג האיכות בין הסוכנים השונים עבור המקרה הכללי.

ייתכן כי יתרונותיו של סוכן זה באו לידי ביטוי בניסוי זה עקב השוני באסטרטגיה שלו אל מול הסוכנים האחרים, הפועלים באסטרטגיות שעלולות להוביל אותם לקבלת החלטות דומות. כלומר, מספר סוכנים "המסתכלים" לעומק רואים את אותו המצב כמצב אידיאלי עבורם, ובעת שהראשון מביניהם מבצע צעד לכיוון מצב זה, הוא פוגע באסטרטגיה של האחרים בצורה בלתי צפויה, בעוד סוכן ה-Local Search שאינו מסתכל לעומק מפתח אסטרטגיה בלתי תלויה שאינה מושפעת ממהלך כזה. זוהי השערה בלבד ואין לכך תימוכין ממשי מתוצאות הניסויים.

ג. **כישלון סוכן ה-Expectimax עם היוריסטיקת Advantage on Opponent בניסוי המסכם** – גם עבור סוכן זה, בשונה מאשר בשלבי הניסוי הקודמים, התקבלו תוצאות מפתיעות. **בשלבי הניסוי הקודמים השיג סוכן זה תוצאות אשר ניכר כי הן הטובות ביותר מבין כלל הסוכנים, בעוד שבניסוי המסכם תוצאותיו נמוכות בהרבה** (התבצעו הרצות נוספות של הניסוי המסכם כדי לוודא סטייה זו, אך בכולן התקבלו תוצאות דומות). נתון זה מחליש גם כן את המשקל של הניסוי המסכם בקביעת הדירוג בין איכות הסוכנים.

ד. **כישלון הניסיונות ליצור אסטרטגיית Punt אפקטיבית** – כפי שפורט בתיאור משחק הפנטזי, אסטרטגיית ה-Punt, במסגרתה זונחים קטגוריות מסויימות על מנת להתחזק ברוב של קטגוריות ספציפיות, הינה אסטרטגיה מובילה להרכבת קבוצה במסגרת הדראפט. **במהלך שלבי הניסוי השונים התבצעו ניסיונות לפתח אסטרטגיות Punt שונות באמצעות סוכן ה-Random Label וההיוריסטיקה Advantage on Opponent** (בשילוב עם סוכן ה-Expectimax). אסטרטגיות אלו מתבססות על קבלת החלטה על סמך תת קבוצה של קטגוריות מתוך קטגוריות אפיון השחקן, ועל כן הנחנו כי הן יוכלו לשמש לפיתוח Punt. עם זאת, כפי שניתן לראות מהתוצאות, **כלל הסוכנים הללו השיגו תוצאות נמוכות משמעותית משאר הסוכנים**. ייתכן כי הניסיון לייצר Punt באמצעות שיטות אלו לא סיפק לסוכן האוטונומי בסיס מספק, המכריח אותו לבחור שחקנים המתאימים במיוחד ל-Punt, **וניכר כי לצורך השגת מטרה זו יש לפתח סוכן בעל אסטרטגיה מורכבת בהרבה**.

ה. **שיוויון במדדי היעילות** – ממרבית התוצאות הכוללות עולה כי ישנה קירבה גבוהה מאוד (עד כדי שיוויון) בהשגיו של כל סוכן בשני המדדים – Wins ו-Ranks. לעיתים קרובות במשחק הפנטזי האמיתי, ישנו קושי רב בהרכבת קבוצה אשר תימצא גבוה בדירוג בשני המדדים. המסקנה העולה מהניסויים הללו, לפיה סוכן אשר משיג תוצאות טובות לפי מדד אחד כנראה ישיג תוצאות טובות גם לפי המדד השני, עשויה להעיד על איכות הביצוע של הסוכנים המובילים ועל התאמתם למשחק הדראפט.

## ניסויים משניים – תוצאות ומסקנות

### Agent Per Pick

המסקנות עבור שלב זה בוססו על ניתוח תוצאות האיטרציות השונות במספר הרצות של הניסוי המסכם, זאת לצורך אפיון הסוכן העדיף לכל מיקום בחירה (מתוך 11 קבוצות, הבחירות ממסופרות מ-0 עד 10). התוצאות מוצגות בתרשים ייעודי עבור כל מיקום בחירה, אותם ניתן למצוא בתיקיה "AI-Project-NBA-Fantasy\Draft\Experiments\Agent\_Per\_Pick בצורה ברורה אילו אסטרטגיות עדיפות עבור בחירה נתונה.

1. להלן דוגמא עבור התוצר שהתקבל עבור הבחירה הראשונה (0), ממנו ניתן לראות כי האסטרטגיה העדיפה עבור בחירה זו היא אסטרטגיית Local Search עם היוריסטיקת Quorum Wins –



2. סיכום האסטרטגיה הנבחרת לכל מיקום בסדר הבחירות –

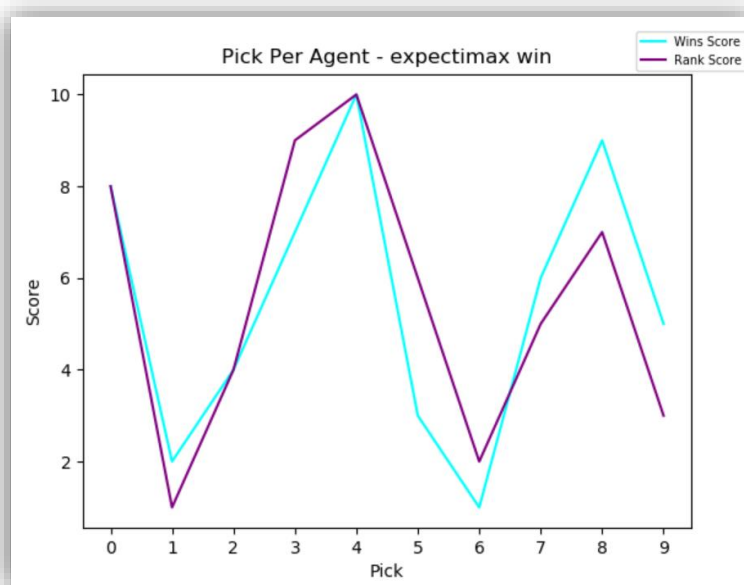
מיקום הבחירה	אסטרטגיה נבחרת
0	Local Search (Wins)
1	Expectimax (Wins)
2	Alpha Beta (Wins or Ranks)
3	Maximax (Advantage on Opponent)
4	Alpha Beta (Wins)
5	Local Search (Wins)
6	Expectimax (Wins)
7	Maximax (Advantage on Opponent)
8	Alpha Beta (Advantage on Opponent)
9	Local Search (Advantage on Opponent)
10	Expectimax (Wins)



## Pick Per agent

המסקנות עבור שלב זה בוססו על הרצת ניסוי יחיד (מספר איטרציות) עבור כל סוכן, כאשר בכל ניסוי השתתפו אך ורק יריבים מאותו הסוג, זאת לצורך אפיון מיקום הבחירה העדיף לכל סוכן (במסגרת דראפט בן 10 קבוצות, מיקומי בחירות 0 עד 9). התוצאות מוצגות בתרשים ייעודי עבור כל סוכן שנבחן, אותם ניתן למצוא בתיקייה "AI-Project-NBA-Fantasy\Draft\Experiments\Pick\_Per\_Agent" בפרויקט שהוגש. מהתשימים ניתן לקבוע בצורה ברורה מהם המיקומים העדיפים בסדר הבחירה עבור כל סוכן.

1. להלן דוגמא עבור התוצר שהתקבל עבור סוכן Expectimax עם היוריסטיקת Quorum Wins, ממנו ניתן לראות כי מיקום הבחירה העדיף עבור סוכן המשחק באסטרטגיה זו הינה הבחירה החמישית (בחירה מספר 4) –



2. סיכום המיקום המועדף לכל אסטרטגיית משחק –

מיקום בחירה מועדף	אסטרטגיה
8	Local Search (Wins)
0	Local Search (Advantage on Opponent)
1	Alpha Beta (Wins)
1	Alpha Beta (Ranks)
0 או 6	Alpha Beta (Advantage on Opponent)
1	Maximax (Advantage on Opponent)
4	Expectimax (Wins)
7	Expectimax (Advantage on Opponent)

## סיכום

הפרויקט שהוצג לעיל עוסק בפיתוח פיתרון מבוסס בינה מלאכותית ולמידת מכונה להתמודדות עם משחק הפנטזי NBA, ובפרט עם שלב חיזוי טרום העונה ומשחק הדראפט. במסגרת הפרויקט פותחו שתי מערכות אשר מטרתן להתמודד עם בעיה זו – **מערכת חיזוי הישגי שחקני ה-NBA בעשר קטגוריות עבור עונה נתונה, ושחקן אוטונומי להתמודד בתחרות דראפט פנטזי רבת משתתפים.**

לצורך פיתוח המערכות הללו התבססנו רבות על הידע והכלים אשר נלמדו במסגרת קורס המבוא, ונדרשנו בנוסף להעמיק את הידע בתחומים אלו, בפרט בנושא אלגוריתמי הלמידה המתאימים לביצוע רגרסיה, עיבוד וניתוח מידע רחב היקף וכו'. **הידע שצברנו במהלך הפרויקט, יחד עם היכרות מוקדמת ועניין רב במשחק הפנטזי, איפשרו לנו לממש בהצלחה את שתי המערכות, המשיגות תוצאות טובות ביחס למדדים שהוגדרו בשלבי הפרויקט השונים.**

**להלן עיקרי המסקנות והנקודות המרכזיות העולות מכל אחד מחלקי הפרויקט, אשר הוצגו בהרחבה במסמך זה:**

1. שני חלקי המערכת מהווים יחדיו מימוש מקצה לקצה של תהליך המבוצע לרוב על ידי סוכן אנושי המשתתף במשחק הפנטזי. כמו הסוכן האנושי, כך גם המערכת שנבנתה במהלך הפרויקט מסוגלת לחזות את הישגי שחקני ה-NBA בעונה הבאה, ולאחר מכן להרכיב קבוצה כחלק ממשחק הדראפט, תוך התמודדות עם סוכנים יריבים, ויכולת להעריך את התרומה של כל שחקן NBA לקבוצה המורכבת.

### 2. מערכות חיזוי סטטיסטיות שחקני ה-NBA –

א. במשחק הפנטזי ובפרט בשלב הדראפט הנערך בתחילת עונת המשחק, נעשה שימוש נרחב בחיזוי סטטיסטיקות שונות ממשחק הכדוסל, אשר אותן צפויים שחקני ה-NBA להשיג במהלך העונה, ועליהם מתבסס משחק הפנטזי. **מטרת הפרויקט בחלק זה הייתה לפתח חלופה ראויה, מבוססת למידת מכונה, אשר תספק את החיזוי הנ"ל, בעשר קטגוריות המשחק הרלוונטיות ביותר.**

ב. **שלב זה של הפרויקט כלל –** השגת המידע המתאים, איסוף תכונות לצורך הגדרת אובייקט מתוך המידע הגולמי, עיבוד המידע, סינון תכונות, הגדרת מטריקות להערכת ביצועים, אימון מודל והערכת ביצועיו ואופטימיזציה של המודלים המובילים.

ג. **התוצר הסופי של חלק זה הינו מערכת המורכבת מעשרה מודלי חיזוי שונים, אחד עבור כל קטגוריית חיזוי.** על פי מדדי מרחק השגיאה הממוצע והחציוני בחיזוי כל אחת מהקטגוריות, ניכר כי תוצאות המערכת משתוות אל תוצאות חיזוי מומחי משחק הפנטזי (ובחלקים מסויימים אף עוקפות אותן).

ד. סוגי המודלים הסופיים שנבחרו על מנת לחזות את קטגוריות המשחק השונות **כוללים אלגוריתמי למידה מגוונים** – Linear Regression, Ridge Regression, Random Forest ו-SVR. **מגוון זה מעיד על הייחודיות של בעיות החיזוי השונות**, הן מבחינת אלגוריתם הלמידה הנדרש על מנת למדל כל אחת מהן והן מבחינת קבוצת התכונות האופטימלית, כפי שאכן ניתן לראות בתוצאות חלק החיזוי.

ה. בניגוד להשערה הראשונית אשר נבעה מהיכרות מוקדמת עם משחק הפנטזי, האלגוריתמים שהתאימו לחיזוי מרבית הקטגוריות הינם אלגוריתמים פשוטים יחסית, המבוססים על קשר ליניארי בין התכונות לבין הקטגוריה הנחזית – Linear Regression ו-Ridge Regression. פשטות זו של חלק הארי מהמודלים הסופיים מהווה יתרון, מכיוון שמודלים אלו פחות נותנים לבצע Overfitting, מאשר מודלים מורכבים יותר, ולהערכתנו עובדה זו תוביל לכך שהמערכת תשיג תוצאות טובות גם בחיזוי נתונים של עונות עתידיות בליגת ה-NBA, תוך הסתמכות על המודלים וקבוצת התכונות שנבחרו, ללא צורך לבחון את המודלים מחדש.

### 3. מערכת סוכן אוטונומי למשחק הדראפט –

א. השלב הראשון בכל עונה במשחק הפנטזי הינו שלב הדראפט, במסגרתו מתחרות כלל הקבוצות בהרכבת קבוצת שחקני NBA מתוך מאגר שחקנים ותחת הגבלות שונות. מטרת הפרויקט בחלק זה הייתה לממש סוכן אוטונומי הפועל באסטרטגיה נבחרת בכדי להרכיב קבוצה טובה ככל הניתן.

ב. התוצר הסופי של חלק זה הינו סדרת סוכנים, רובם מבוססי אלגוריתמי חיפוש היוריסטי, וסימולטור להרצת משחק הדראפט כתחרות מרובת סוכנים. תוצאות הניסויים בחלק זה אינן מצביעות על סוכן אידיאלי לתרחיש הכללי של משחק הדראפט. עם זאת, התוצאות מצביעות על עדיפות מובהקת לסוגי סוכנים המתבססים על חיפוש היוריסטי, ובנוסף, מספקות אינפורמציה יעילה לצורך התאמת סוכן בעל אסטרטגיה ספציפית, בהתאם לתרחיש המשחק – מספר הקבוצות, גודל הקבוצה, אופי היריבים, מיקום הבחירה האקראי וכו'.

ג. כחלק מתהליך התאמת סוכני החיפוש למשחק הדראפט, הוגדרו מספר היוריסטיקות להערכת מצבי המשחק, ומתוצאות הניסויים ניתן לקבוע כי ההיוריסטיקה Advantage on Opponent, המבוססת על חישוב ערך משוקלל של מיקום הסוכן משאר יריביו בכל אחת מקטגוריות הניקוד, הינה העדיפה ביותר לשימוש בתהליך החיפוש במרחב המצבים.

הפרויקט עסק בגירסה מצומצמת של משחק הפנטזי, תחת הנחות מקלות, לצורך התאמה למסגרת הזמנים ודרישות הפרויקט. **במהלך הפיתוח עלו רעיונות והצעות ייעול ושיפור אשר מימושם אינו נכלל בפרויקט זה.** כיוונים אלו עשויים לשפר את ביצועי המערכת ולאפשר, הן לשלב החיזוי והן לסוכן האוטונומי, להשיג תוצאות טובות יותר, ולהתמודד בסיטואציות משחק מורכבות יותר.

### **להלן פירוט מספר כיווני מחקר להמשך:**

1. **איסוף מידע נוסף וייצוגו באופן שיטתי** – ישנם פרמטרים נוספים המשפיעים על ביצועי שחקן במשחק הכדורסל, כגון פציעות, השפעה של שחקנים נוספים בקבוצה בעלי סגנון משחק דומה וכן פרטים נוספים פחות טריוויאליים. במהלך העבודה על הפרויקט עלה קושי במציאת מידע מסודר עבורן או ביצירת ייצוג הגיוני של פרמטרים אלו. **ייתכן שגישה אל נתונים בעלי השפעה מסוג זה יוכלו לשפר את ביצועי מערכת החיזוי.**

2. **הכנסת אלמנטים נוספים במודל משחק הפנטזי** – בפרויקט זה הנחנו את המודל הפשוט והנפוץ ביותר למשחק הפנטזי, בו מרכיבים קבוצה בתחילת העונה, ועל סמך ניקוד הקטגוריות הכולל בסוף העונה מכריזים על המנצח והדירוג בליגה. ישנם אלמנטים רבים מעניינים אותם לא הכנסנו למודל המשחק, ביניהם חישוב ניקוד על סמך matchup שבועי במקום על פני העונה כולה (התבצע ניסיון להתחשב בשיטת ניקוד זו במסגרת מדד ה-Wins לפיו מדדנו את ביצועי הסוכן), החלפות של שחקנים בין הסוכנים השונים בליגה, וכן האפשרות להחליף שחקן בקבוצה מתוך מאגר של שחקנים פנויים במהלך העונה. **ניתן להניח כי ככל שתהיה הצלחה במציאת אסטרטגיה טובה בוואריאציה מציאותית ומורכבת יותר של המשחק, כך נוכל ליישם את תוצאות הפרויקט בעולם האמיתי בצורה מוצלחת יותר.**

## נספח א' – תוצאות מודלים נבחרים מניסוי החיזוי שלב 1

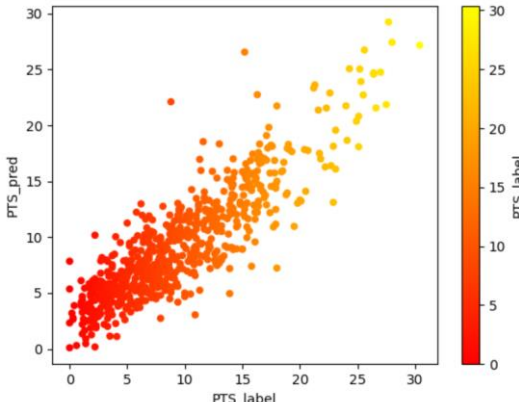
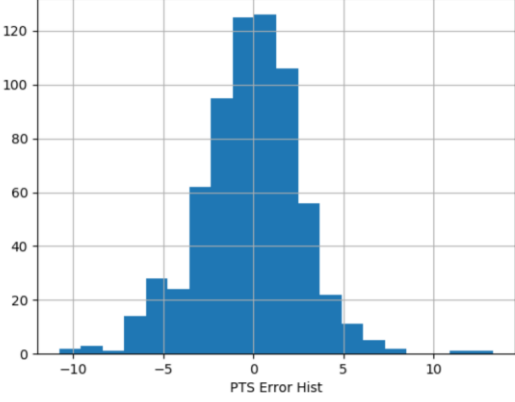
Category	Top Selected Models (Model Type, Num of Features, Selector)	Avg	Med
PTS	BayesianRidge, 20, Wrapper	2.225	1.770
	BayesianRidge, 40, Wrapper	2.225	1.760
	LinearRegression, 30, Embed	2.228	1.790
	MLPRegressor, 10, Embed	2.261	1.750
	RandomForestRegressor, 20, Embed	2.243	1.760
AST	BayesianRidge, 60, Wrapper	0.563	0.389
	SVR, 40, Embed	0.599	0.380
	RandomForestRegressor, 30, Embed	0.575	0.377
	MLPRegressor, 90, Embed	0.564	0.392
	LinearRegression, 30, Embed	0.567	0.393
TRB	BayesianRidge, 40, Wrapper	0.894	0.643
	BayesianRidge, 100, SelectKBest	0.897	0.650
	MLPRegressor, 80, Embed	0.896	0.648
	LinearRegression, 40, Embed	0.895	0.654
STL	BayesianRidge, 40, Wrapper	0.189	0.150
	BayesianRidge, 30, Wrapper	0.190	0.149
	LinearRegression, 20, Embed	0.192	0.150
	LinearRegression, 50, Wrapper	0.191	0.153
BLK	BayesianRidge, 50, Embed	0.191	0.151
	BayesianRidge, 20, Wrapper	0.153	0.104
	RandomForest, 80, Wrapper	0.154	0.101
	RandomForest, 30, Embed	0.154	0.104
	LinearRegression, 30, Embed	0.155	0.105
TOV	BayesianRidge, 40, Embed	0.155	0.104
	BayesianRidge, 30, Wrapper	0.305	0.240
	BayesianRidge, 40, Wrapper	0.306	0.240
	LinearRegression, 50, Wrapper	0.307	0.242
	RandomForset, 20, Wrapper	0.314	0.241
FG%	SVR, 60, Embed	0.310	0.243
	Bayesian Ridge, selectKBest, 30	0.041	0.027
	Random Forest, Embedded, 20	0.041	0.027
	Bayesian Ridge, Wrapper, 60	0.041	0.027
	Linear Regression, Embedded, 20	0.041	0.027
FT%	Bayesian Ridge, Embedded, 10	0.041	0.027
	Random Forest, Embedded, 40	0.068	0.046
	Random Forest, Wrapper, 70	0.068	0.046
	Random Forest, Wrapper, 20	0.068	0.047
	Bayesian Ridge, Wrapper, 50	0.069	0.048
3P	Linear Regression, Wrapper, 40	0.070	0.049
	Random Forest, Embedded, 70	0.268	0.178
	SVR, Embedded, 30	0.278	0.170
	Random Forest, SelectKBest, 20	0.268	0.176
	Bayesian Ridge, Wrapper, 30	0.277	0.190
Fantasy Rank	Linear Regression, Embedded, 10	0.282	0.177
	BayesianRidge, 30, Wrapper	66.070	54.170
	LinearRegression, 40, Embed	66.490	54.270
	RandomForestRegressor, 50, Embed	67.260	53.770
	RandomForestRegressor, 100, Wrapper	67.560	53.680
	MLPRegressor, 100, Embed	67.720	54.130

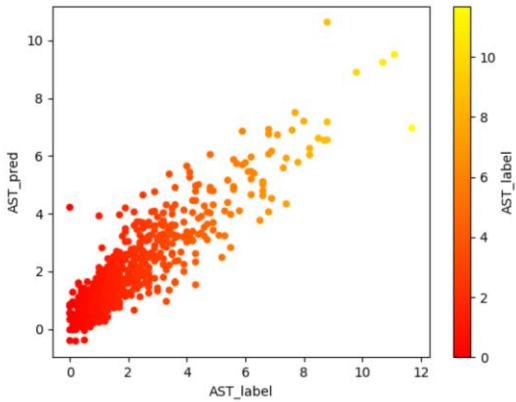
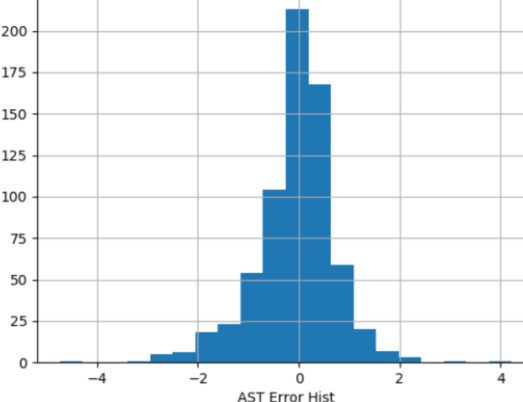
## נספח ב' – ניסוי החיזוי שלב 2 – פרטי המודלים

Category	Model, Num of Features, Selector	Model Params	Params Grid Values
PTS	Ridge, 20, Wrapper	alpha	20 equal parts between 0.1 - 3.0
	LinearRegression, 30, Embed		
	RandomForestRegressor, 20, Embed	n_estimators	[5, 10, 15, 20]
		criterion	['mse', 'mae']
		min_samples_split	[2-7]
		min_samples_leaf	[1-5]
AST	MLPRegressor, 90, Embed	activation	['logistic', 'tanh', 'relu']
		solver	['lbfgs', 'sgd', 'adam']
		alpha	[0.01, 0.001, 0.0001, 0.00001]
	BayesianRidge, 60, Wrapper	alpha_1	[1e-03, 1e-04, 1e-05, 1e-06]
		alpha_2	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_1	[1e-03, 1e-04, 1e-05, 1e-06]
TRB	MLPRegressor, 80, Embed	activation	['logistic', 'tanh', 'relu']
		solver	['lbfgs', 'sgd', 'adam']
		alpha	[0.01, 0.001, 0.0001, 0.00001]
STL	BayesianRidge, 30, Wrapper	alpha_1	[1e-03, 1e-04, 1e-05, 1e-06]
		alpha_2	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_1	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_2	[1e-03, 1e-04, 1e-05, 1e-06]
	BayesianRidge, 40, Wrapper	alpha_1	[1e-03, 1e-04, 1e-05, 1e-06]
		alpha_2	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_1	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_2	[1e-03, 1e-04, 1e-05, 1e-06]
BLK	Ridge, 30, Wrapper	alpha	20 equal parts between 0.1 - 3.0
	Ridge, 40, Wrapper	alpha	20 equal parts between 0.1 - 3.0
	BayesianRidge, 20, Wrapper	alpha_1	[1e-03, 1e-04, 1e-05, 1e-06]
		alpha_2	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_1	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_2	[1e-03, 1e-04, 1e-05, 1e-06]
	RandomForest, 80, Wrapper	n_estimators	[5, 10, 15, 20]
		criterion	['mse', 'mae']
		min_samples_split	[2-7]
		min_samples_leaf	[1-5]
TOV	Ridge, 20, Wrapper	alpha	20 equal parts between 0.1 - 3.0
	BayesianRidge, 30, Wrapper	alpha_1	[1e-03, 1e-04, 1e-05, 1e-06]
		alpha_2	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_1	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_2	[1e-03, 1e-04, 1e-05, 1e-06]
	SVR, 60, Embed	kernel	['rbf', 'linear', 'poly', 'sigmoid']
		degree	[1, 2, 3, 4, 5]
		C	[0.0001, 0.001, 0.01, 0.1, 1, 10, 100]
	Ridge, 30, Wrapper	alpha	20 equal parts between 0.1 - 3.0
FG%	Linear Regression, Embeddd, 20		
	Bayesian Ridge, Embedded, 10	alpha_1	[1e-03, 1e-04, 1e-05, 1e-06]
		alpha_2	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_1	[1e-03, 1e-04, 1e-05, 1e-06]
		lambda_2	[1e-03, 1e-04, 1e-05, 1e-06]
FT%	Ridge, Embed, 20	alpha	20 equal parts between 0.1 - 3.0
	Random Forest, Embedded, 40	n_estimators	[5, 10, 15, 20]

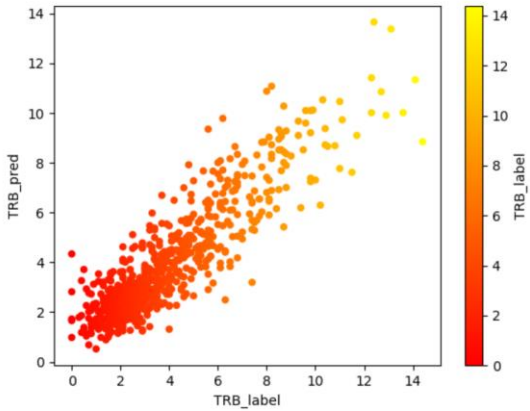
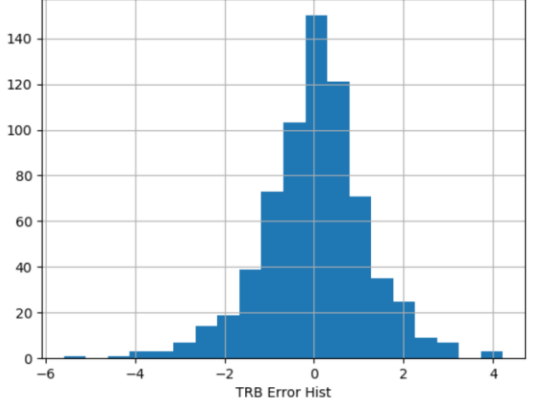
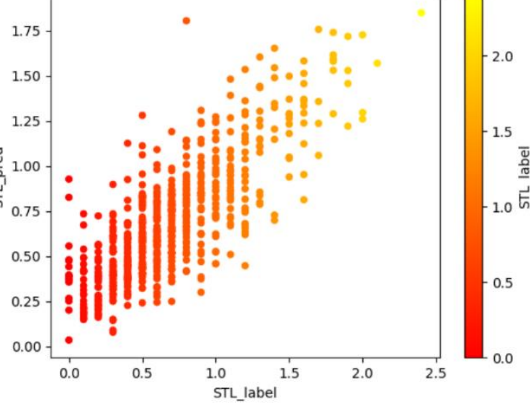
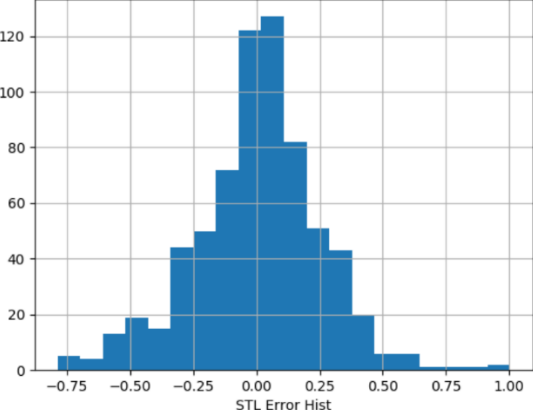
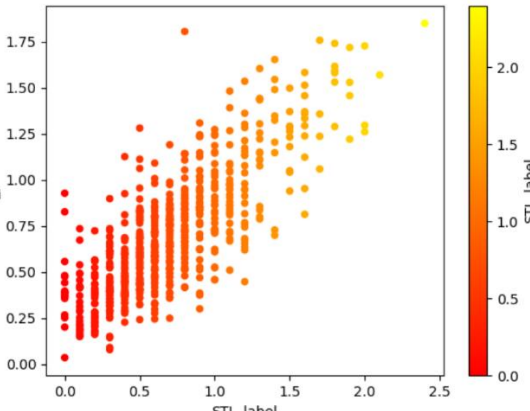
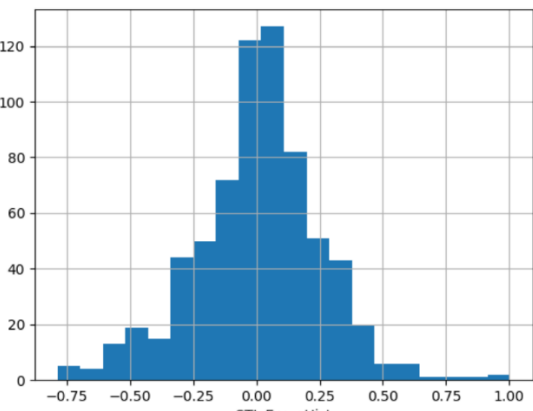
		criterion	['mse', 'mae']	
		min_samples_split	[2-7]	
		min_samples_leaf	[1-5]	
	Bayesian Ridge, Wrapper, 50	alpha_1	[1e-03, 1e-04, 1e-05, 1e-06]	
		alpha_2	[1e-03, 1e-04, 1e-05, 1e-06]	
		lambda_1	[1e-03, 1e-04, 1e-05, 1e-06]	
		lambda_2	[1e-03, 1e-04, 1e-05, 1e-06]	
	Ridge, Wrapper, 50	alpha	20 equal parts between 0.1 - 3.0	
	3P	SVR, Embedded, 30	kernel	['rbf', 'linear', 'poly', 'sigmoid']
			degree	[1, 2, 3, 4, 5]
C			[0.0001, 0.001, 0.01, 0.1, 1, 10, 100]	
Random Forest, SelectKBest, 20		n_estimators	[5, 10, 15, 20]	
		criterion	['mse', 'mae']	
		min_samples_split	[2-7]	
		min_samples_leaf	[1-5]	
Bayesian Ridge, Wrapper, 30		alpha_1	[1e-03, 1e-04, 1e-05, 1e-06]	
		alpha_2	[1e-03, 1e-04, 1e-05, 1e-06]	
		lambda_1	[1e-03, 1e-04, 1e-05, 1e-06]	
	lambda_2	[1e-03, 1e-04, 1e-05, 1e-06]		
Fantasy Rank	MLPRegressor, 100, Embed	activation	['logistic', 'tanh', 'relu']	
		solver	['lbfgs', 'sgd', 'adam']	
		alpha	[0.01, 0.001, 0.0001, 0.00001]	
	RandomForestRegressor, 100, Embed	n_estimators	[5, 10, 15, 20]	
		criterion	['mse', 'mae']	
		min_samples_split	[2-7]	
		min_samples_leaf	[1-5]	

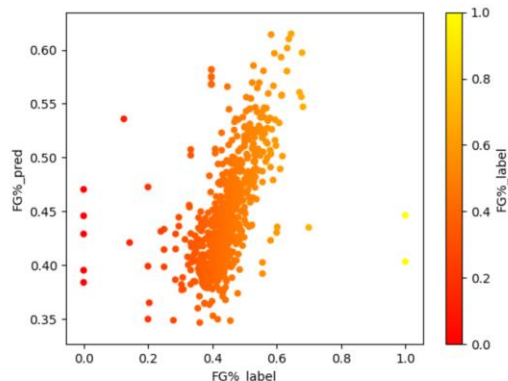
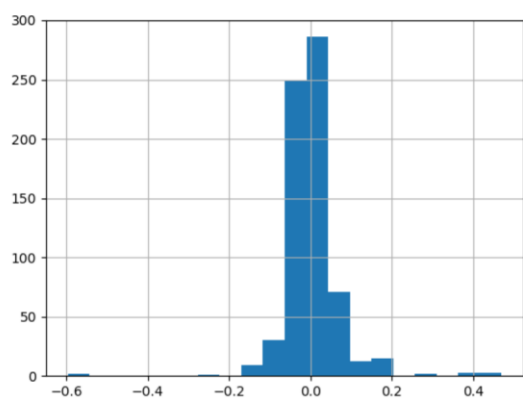
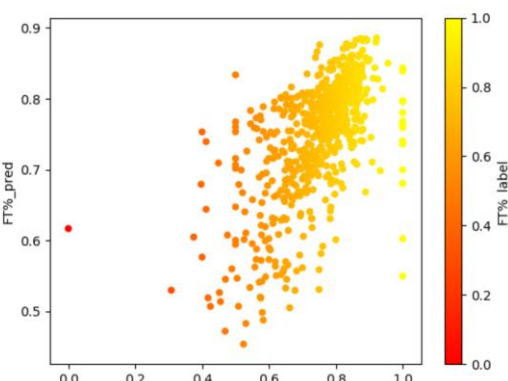
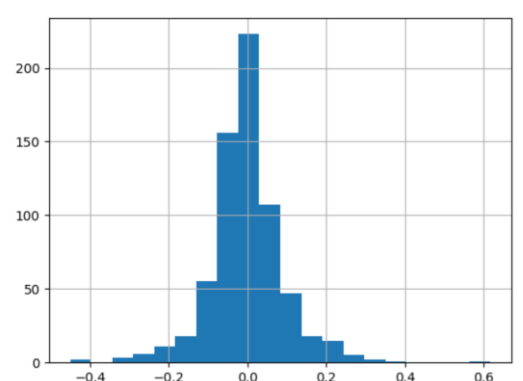
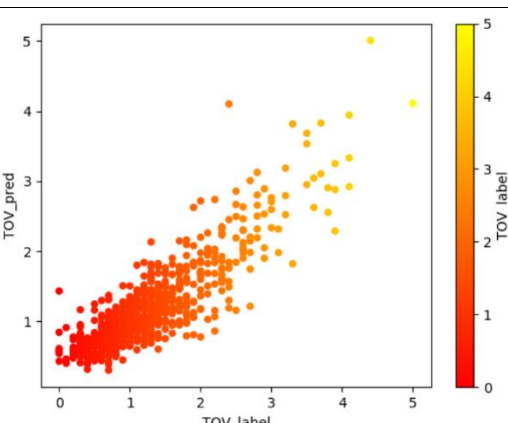
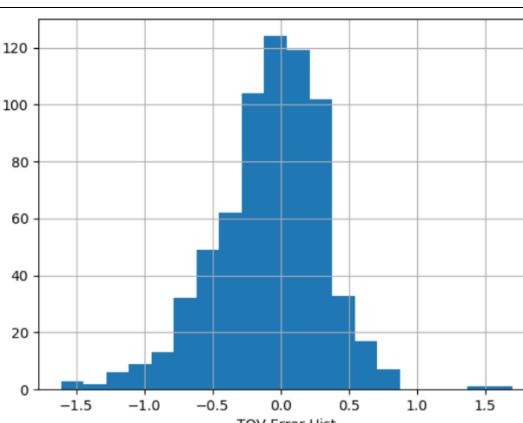
## נספח ג' – מערכת החיזוי – תוצאות סופיות

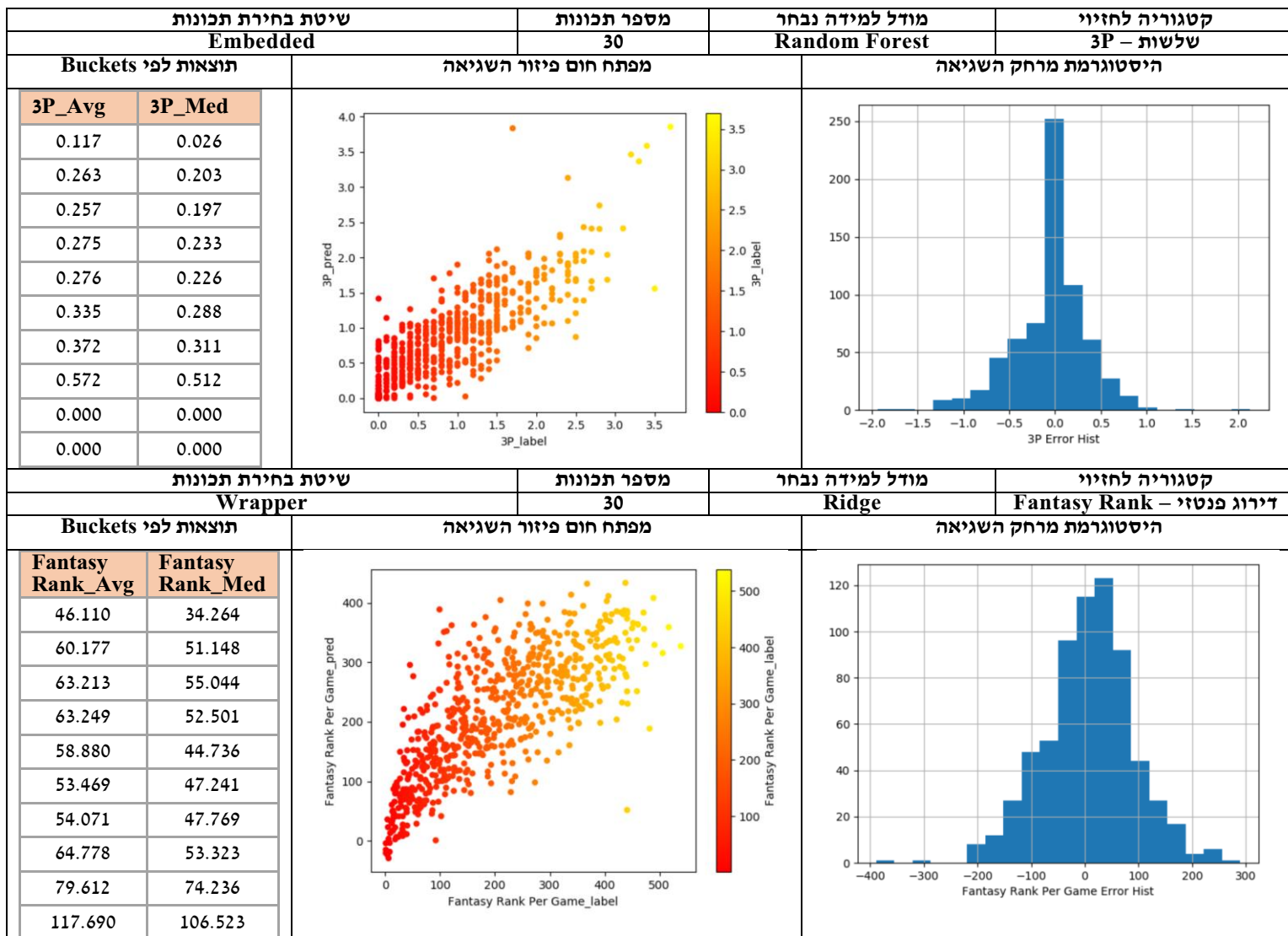
שיטת בחירת תכונות		מספר תכונות	מודל למידה נבחר	קטגוריה לחיזוי
Wrapper		40	Ridge	נקודות - PTS
תוצאות לפי Buckets		מפתח חום פיזור השגיאה		היסטוגרמת מרחק השגיאה
PTS_Avg	PTS_Med			
2.538	2.319			
2.086	1.808			
1.853	1.486			
1.702	1.370			
1.715	1.449			
2.055	1.666			
2.142	1.775			
2.425	1.818			
2.460	2.074			
3.224	2.765			

שיטת בחירת תכונות		מספר תכונות	מודל למידה נבחר	קטגוריה לחיזוי
Embedded		30	Linear Regression	אסיסטים - AST
תוצאות לפי Buckets		מפתח חום פיזור השגיאה		היסטוגרמת מרחק השגיאה
AST_Avg	AST_Med			
0.431	0.311			
0.348	0.280			
0.316	0.261			
0.393	0.284			
0.411	0.316			
0.531	0.436			
0.558	0.443			
0.632	0.507			
0.926	0.809			
1.159	0.889			



שיטת בחירת תכונות		מספר תכונות	מודל למידה נבחר	קטגוריה לחזיון
Wrapper		40	Ridge	ריבאונדים - TRB
תוצאות לפי Buckets		מפתח חום פיזור השגיאה		היסטוגרמת מרחק השגיאה
TRB_Avg	TRB_Med			
1.034	0.898			
0.640	0.491			
0.657	0.468			
0.664	0.497			
0.713	0.538			
0.728	0.607			
0.800	0.652			
0.984	0.726			
1.101	0.952			
1.598	1.347			
שיטת בחירת תכונות		מספר תכונות	מודל למידה נבחר	קטגוריה לחזיון
Wrapper		40	Ridge	חטיפות - STL
תוצאות לפי Buckets		מפתח חום פיזור השגיאה		היסטוגרמת מרחק השגיאה
STL_Avg	STL_Med			
0.228	0.209			
0.152	0.116			
0.126	0.094			
0.119	0.091			
0.147	0.123			
0.151	0.127			
0.186	0.17			
0.214	0.185			
0.262	0.233			
0.26	0.218			
שיטת בחירת תכונות		מספר תכונות	מודל למידה נבחר	קטגוריה לחזיון
Wrapper		80	Random Forest	בלוקים - BLK
תוצאות לפי Buckets		מפתח חום פיזור השגיאה		היסטוגרמת מרחק השגיאה
BLK_Avg	BLK_Med			
0.113	0.077			
0.095	0.065			
0.116	0.086			
0.123	0.095			
0.111	0.092			
0.167	0.134			
0.255	0.220			
0.384	0.331			
0.000	0.000			
0.000	0.000			

שיטת בחירת תכונות Embedded		מספר תכונות 20	מודל למידה נבחר Linear Regression	קטגוריה לחזיון אחוזי שדה – FG%
תוצאות לפי Buckets		מפתח חום פיזור השגיאה		היסטוגרמת מרחק השגיאה
FG%_Avg	FG%_Med			
0.102	0.069			
0.035	0.025			
0.023	0.018			
0.024	0.019			
0.023	0.018			
0.024	0.019			
0.030	0.027			
0.035	0.030			
0.036	0.030			
0.078	0.057			
שיטת בחירת תכונות Embedded		מספר תכונות 40	מודל למידה נבחר Random Forest	קטגוריה לחזיון FT% – עונשין
תוצאות לפי Buckets		מפתח חום פיזור השגיאה		היסטוגרמת מרחק השגיאה
FT%_Avg	FT%_Med			
0.176	0.154			
0.077	0.073			
0.057	0.051			
0.045	0.041			
0.044	0.035			
0.036	0.030			
0.036	0.029			
0.041	0.033			
0.050	0.038			
0.107	0.077			
שיטת בחירת תכונות Embedded		מספר תכונות 60	מודל למידה נבחר SVR	קטגוריה לחזיון איבודים – TOV
תוצאות לפי Buckets		מפתח חום פיזור השגיאה		היסטוגרמת מרחק השגיאה
TOV_Avg	TOV_Med			
0.362	0.322			
0.208	0.171			
0.217	0.168			
0.172	0.130			
0.206	0.174			
0.236	0.210			
0.330	0.282			
0.384	0.353			
0.443	0.392			
0.595	0.519			



## נספח ד' – תיאור הניסויים הסטנדרטים – מערכת סוכן הדראפט

פירוט הסוכנים שנבחנו:

Agent Type	Parameters	Values
<b>Monkey</b>		
<b>Label Fantasy Rank</b>		
<b>Label PTS</b>		
<b>Random Label Punt 1</b>		
<b>Random Label Punt 2</b>		
<b>Random Label No Punt</b>		
<b>Local Search</b>	Heuristic	Advantage on Opponent
	Branch_factor	40
<b>Alpha Beta</b>	Heuristic	Rank per Category
	Depth	3
	Branch_factor	4
<b>Alpha Beta</b>	Heuristic	Quorum Wins
	Depth	3
	Branch_factor	4
<b>Alpha Beta</b>	Heuristic	Advantage on Opponent
	Depth	3
	Branch_factor	4
<b>MaxiMax</b>	Heuristic	Rank per Category
	Opp_Heuristic	Rank per Category
	Depth	2
	Branch_factor	3
	Opp_Branch_factor	2
<b>MaxiMax</b>	Heuristic	Quorum Wins
	Opp_Heuristic	Quorum Wins
	Depth	2
	Branch_factor	3
	Opp_Branch_factor	2
<b>MaxiMax</b>	Heuristic	Advantage on Opponent
	Opp_Heuristic	Advantage on Opponent
	Depth	2
	Branch_factor	3
	Opp_Branch_factor	2
<b>Expectimax</b>	Heuristic	Rank per Category
	Depth	4
	Branch_factor	4
	uniform_opponent_dist	False
<b>Expectimax</b>	Heuristic	Quorum Wins
	Depth	4
	Branch_factor	4
	uniform_opponent_dist	False
<b>Expectimax</b>	Heuristic	Advantage on Opponent
	Depth	4
	Branch_factor	4
	uniform_opponent_dist	False

**פירוט הניסויים:**

Experiment Id	Number of Teams	Number of Picks	Opponents Type	Opponents Params	Params Values	Positions Restriction <sup>15</sup>
1	7	10	Monkey			PG, SG, SF, PF, C * 2
2	7	10	Label - Fantasy Rank			PG, SG, SF, PF, C * 2
3	7	10	Label - PTS			PG, SG, SF, PF, C * 2
4	7	10	Random Label - All labels			PG, SG, SF, PF, C * 2
5	7	10	Random Label (3 of Type Punt 1, 3 of Type Punt 2)			PG, SG, SF, PF, C * 2
6	5	8	Alpha Beta	Heuristic	Rank per Category	PG, SG, SF, PF, C, *, *, *
				Depth	3	
				Branch_factor	3	
7	5	8	Alpha Beta	Heuristic	Quorum Wins	PG, SG, SF, PF, C, *, *, *
				Depth	3	
				Branch_factor	3	
8	5	8	Alpha Beta	Heuristic	Advantage on Opponent	PG, SG, SF, PF, C, *, *, *
				Depth	3	
				Branch_factor	3	
9	5	6	MaxiMax	Heuristic	Rank per Category	PG, SG, SF, PF, C, *
				Opp_Heuristic	Rank per Category	
				Depth	2	
				Branch_factor	2	
				Oponent_Branch_factor	1	
10	5	6	MaxiMax	Heuristic	Quorum Wins	PG, SG, SF, PF, C, *
				Opp_Heuristic	Quorum Wins	
				Depth	2	
				Branch_factor	2	
				Oponent_Branch_factor	1	
11	5	6	MaxiMax	Heuristic	Advantage on Opponent	PG, SG, SF, PF, C, *
				Opp_Heuristic	Advantage on Opponent	
				Depth	2	
				Branch_factor	2	
				Oponent_Branch_factor	1	
12	5	6	Expectimax	Heuristic	Rank per Category	PG, SG, SF, PF, C, *
				Depth	3	
				Branch_factor	3	

<sup>15</sup> עמדות המסומנות כ-\*\* משמעותן "עמדות ללא הגבלה" – ניתן לבחור עבור עמדות אלו שחקן המתאים לכל אחת מעמדות המשחק.

				uniform_opponent_dist	False	
<b>13</b>	5	6	Expectimax	Heuristic	Quorum Wins	PG, SG, SF, PF, C, *
				Depth	3	
				Branch_factor	3	
				uniform_opponent_dist	False	
<b>14</b>	5	6	Expectimax	Heuristic	Advantage on Opponent	PG, SG, SF, PF, C, *
				Depth	3	
				Branch_factor	3	
				uniform_opponent_dist	False	
<b>15</b>	7	10	LocalSearch	Heuristic	Rank per Category	PG, SG, SF, PF, C *2
				Branch_factor	30	
<b>16</b>	7	10	LocalSearch	Heuristic	Quorum Wins	PG, SG, SF, PF, C *2
				Branch_factor	30	
<b>17</b>	7	10	LocalSearch	Heuristic	Advantage on Opponent	PG, SG, SF, PF, C *2
				Branch_factor	30	

## נספח ה' – תיאור הניסויים המתקדמים – מערכת סוכן הדראפט

פירוט הסוכנים שנבחנו:

Agent Type	Parameters	Parameters Values
Local Search	Heuristic	Quorum Wins
	Branch_factor	30
Local Search	Heuristic	Advantage on Opponent
	Branch_factor	30
Alpha Beta	Heuristic	Rank per Category
	Depth	3
	Branch_factor	5
Alpha Beta	Heuristic	Quorum Wins
	Depth	3
	Branch_factor	5
Alpha Beta	Heuristic	Advantage on Opponent
	Depth	4
	Branch_factor	5
MaxiMax	Heuristic	Advantage on Opponent
	Opp_Heuristic	Advantage on Opponent
	Depth	2
	Branch_factor	4
	Opp_Branch_factor	2
Expectimax	Heuristic	Rank per Category
	Depth	4
	Branch_factor	5
	uniform_opponent_dist	False
Expectimax	Heuristic	Quorum Wins
	Depth	4
	Branch_factor	5
	uniform_opponent_dist	False
Expectimax	Heuristic	Advantage on Opponent
	Depth	5
	Branch_factor	5
	uniform_opponent_dist	False
Expectimax Punt	Heuristic	Advantage on Opponent
	labels_list	['PTS','TRB','BLK','AST','3P','STL']
	Depth	5
	Branch_factor	5
	uniform_opponent_dist	False
Hybrid Agent	Agents - 2 rounds - Fantasy rank, 4	
	Expectimax - advantage, Other - Local	
	Search - advantage	

פירוט הניסויים:

Experiment Id	Number of Teams	Number of Picks	Opponents Type	Opponent Params	Params Values	Count	Positions Restriction				
1	7	12	Label - Fantasy Rank			1	(PG, SG, SF, PF, C) x 2 *, *				
			Label - PTS								
			Label - AST								
			Label - TRB								
			Label - BLK								
			Label - STL								
2	7	10	Alpha Beta	Heuristic	Rank per Category	2	(PG, SG, SF, PF, C) x 2				
				Depth	3						
				Branch Factor	3						
			Alpha Beta	Heuristic	Quorum Wins	2					
				Depth	3						
				Branch Factor	3						
			Alpha Beta	Heuristic	Advantage on Opponent	2					
				Depth	3						
				Branch Factor	3						
			3	7	10	Maximax		Heuristic	Rank per Category	2	(PG, SG, SF, PF, C) x 2
								Depth	1		
								Branch Factor	4		
Opp Branch Factor	2										
Maximax	Heuristic	Quorum Wins				2					
	Depth	1									
	Branch Factor	4									
	Opp Branch Factor	2									
Maximax	Heuristic	Advantage on Opponent				2					
	Depth	1									
	Branch Factor	4									
	Opp Branch Factor	2									
4	7	10				Expectimax	Heuristic	Rank per Category	2	(PG, SG, SF, PF, C) x 2	
							Depth	3			
							Branch Factor	4			
						Expectimax	Heuristic	Quorum Wins	2		
			Depth	3							
			Branch Factor	4							
			Expectimax	Heuristic	Advantage on Opponent	2					
				Depth	3						
				Branch Factor	4						



5	9	10	Alpha Beta	Heuristic	Advantage on Opponent	2	(PG, SG, SF, PF, C) x 2
				Depth	4		
				Branch Factor	5		
			Expectimax	Heuristic	Advantage on Opponent	2	
				Depth	4		
				Branch Factor	5		
			Maximax	Heuristic	Advantage on Opponent	2	
				Depth	2		
				Branch Factor	5		
				Opp Branch Factor	Advantage on Opponent		
			Local Search	Heuristic	Advantage on Opponent	2	
				Branch Factor	20		

## מקורות מידע

- i. **אתר הפנטזי של Yahoo** - <https://basketball.fantasysports.yahoo.com/> - פלטפורמת משחק הפנטזי המרכזית ברשת האינטרנט, מספקת כלים לפתיחה וניהול של ליגות פנטזי, ערכית משחקי דראפט, צפייה בחיזויים וכו'.
- ii. **אתר הפנטזי של ESPN** - <https://www.espn.com/fantasy/basketball/> - פלטפורמת משחק הפנטזי מרכזית נוספת במסגרת האתר של רשת הספורט המובילה ESPN. מספקת כלים לפתיחה וניהול של ליגות פנטזי, ערכית משחקי דראפט, צפייה בחיזויים וכו'.
- iii. **אתר Basketball Monsters** - <https://basketballmonster.com/> - אתר המומחים המוביל בתחום משחק הפנטזי NBA. באתר ניתן למצוא סטטיסטיקות, חיזויים והמלצות למשחק הפנטזי, המתעדכנים בתחילת ובמהלך העונה.
- iv. **אתר Basketball References** - <https://www.basketball-reference.com/> - האתר המקיף ביותר ברשת לסטטיסטיקות ונתונים על ליגת ה-NBA. מכיל מגוון רחב מאוד של סטטיסטיקות, תוצאות משחקים ועוד על כל השחקנים בהיסטוריה של הליגה, ומחולק בצורה נוחה לפי עונות משחק, קבוצות, שמות שחקנים וכו'.
- v. **מאגרי נתונים מאתר Kaggle** - <https://www.kaggle.com/> - אתר המכיל מגוון רחב של Datasets לשימוש חופשי במגוון נושאים. מרבית הנתונים בהם השתמשנו נלקחו מה-Datasets - <https://www.kaggle.com/drgilermo/nba-players-stats/version/2>
- vi. **אתר Hashtag Basketball** - <https://hashtagbasketball.com/> - אתר מומחים המספק תחזיות, דירוגים והמלצות למשחק הפנטזי.