פרויקט בקורס לימוד מכונה

חלק ב'

ינואר 2022

קבוצה 6

מגישים:

ניתאי בן ברוך - 312473242

מירית זליצ'ונוק - 318381852

תוכן עניינים

[תיקונים עבור חלק א - (ראו נספחים) 2](#_Toc93410598)

[Decision Trees 2](#_Toc93410599)

[Artificial Neural Networks 3](#_Toc93410600)

[Unsupervised Learning - Clustering 6](#_Toc93410601)

[השוואה בין המודלים - Evaluation 7](#_Toc93410602)

[שיפור המודל הנבחר – Improvements 7](#_Toc93410603)

[הגשת חיזויים סופיים 8](#_Toc93410604)

[נספחים 8](#_Toc93410605)

# תיקונים עבור חלק א - (ראו נספחים)

**חלוקת הנתונים נעשתה לפי שיטת וולידציה שבחרנו בחלק א:** בחרנו בשיטת הוולידציה K-fold cross-validation - נחלק את סט האימון שלנו ל-10 foldים שווים.

# **Decision Trees**

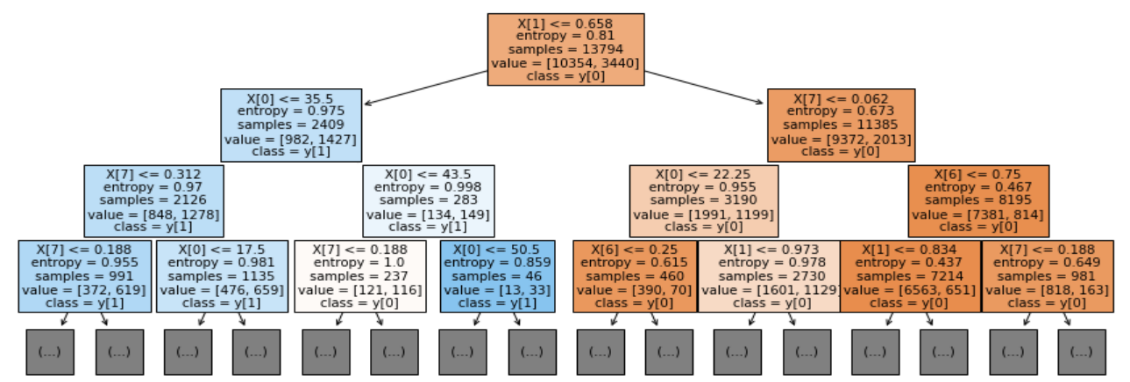
הפרמטרים אשר בחרנו לכוון הינם:

|  |  |  |
| --- | --- | --- |
| פרמטר | טווח ערכים | משמעות והסבר |
| max\_depth | 1-30 | עץ גדול מדיי עלול לגרום ל-over-fitting, ועץ קטן מדיי עלול לתת חיזוי לא מספיק מדויק. בחרנו בטווח בין 1 ל-30 מכיוון שבבדיקה ידנית על כל סט הנתונים ראינו כי ב- max-depth = 21 אחוז הדיוק עמד על 0.96, ומ-22 עד 30 אחוז הדיוק עלה ועמד על 0.97. |
| splitter | best, random | מספר הפיצ'רים אשר יבדקו בכל פיצול. בשיטת best, המודל יבדוק מתוך כל הפיצ'רים ויבחר בטוב ביותר. בשיטת random, המודל יבחן באופן רנדומלי מספר פיצ'רים ויבחר מתוכם את הפיצ'ר הטוב ביותר. |
| criterion | gini, entropy | פונקציה שבוחנת כיצד לבצע את הפיצולים. |

סה"כ 161 קומבינציות אפשריות.

משום שאנחנו צריכים לכוון שלושה פרמטרים - נשתמש ב-Hyperparameter tuning מסוג Grid Search. המודל הטוב ביותר אשר נמצא הינו: max\_depth=5, criterion=entropy, splitter=best. אחוז הדיוק המתקבל על סט האימון הוא 0.798 עם סטיית תקן של 0.002, ואחוז הדיוק המתקבל על סט הוולידציה הוא 0.788 עם סטיית תקן 0.013. ניתן להסיק מתוצאות אלה שמספיק להגיע ל-max\_depth=5 על מנת להגיע לאחוז הדיוק הגבוה ביותר.

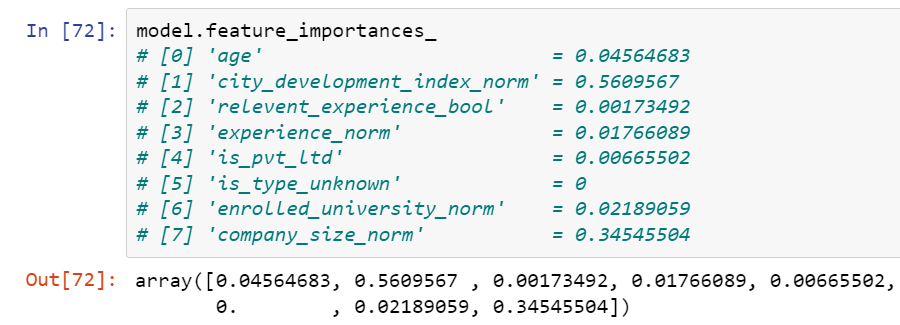
היתרון הבולט של מודל עץ החלטה הוא יכולת ההסבר שלו. הכוונה היא שכאשר מתבוננים על התרשים ניתן לראות כי כל צומת מייצג תנאי פשוט להבנה על פיצ'ר מסוים ומה הפלט. יכולת זו תסייע לנו להבין לפי איזו לוגיקה המודל בחר לסווג האם עובד ישאר או יעזוב.

הגרף שהתקבל:

ניתן לראות ממבנה העץ כי ה-dataset שלנו אינו מאוזן (10,354 עבור y=0 ו-3,440 עבור y=1). בנוסף, ניתן לראות כי הפיצ'ר הראשון בו העץ משתמש הינו city\_development\_index\_norm (כלומר [1]X), לכן אנחנו מבינים כי לפיצ'ר זה משמעות גדולה עבור משימת הלמידה. כלומר, נסיק כי דירוג עיר גבוה (החל מ-0.658) מצביע על כוונת העובד להישאר בחברה.

לאחר מכן, ניתן לראות כי הפיצ'ר שחוזר על עצמו הכי פעמים בעץ (עד השכבה השלישית לפחות) הינו company\_size\_norm (כלומר [7]X) - משמע, גם לו יש משמעות גדולה.

בנוסף, ניתן לראות כי הפיצ'רים age (שהינו [0]X) ו-enrolled\_university\_norm (שהינו [6]X) הם גם משמעותיים עבור המשימה, אך פחות מהשניים הקודמים.

השימוש בתכונת ה-feature\_importances מחזק את המסקנות שלנו ממקודם - בהחלט ניתן לראות כי הפיצ'רים בעלי המשמעות הגדולה ביותר לסיווג האם עובד ישאר בחברה או יעזוב הם דירוג העיר ממנה מגיע וגודל החברה בה עובד.

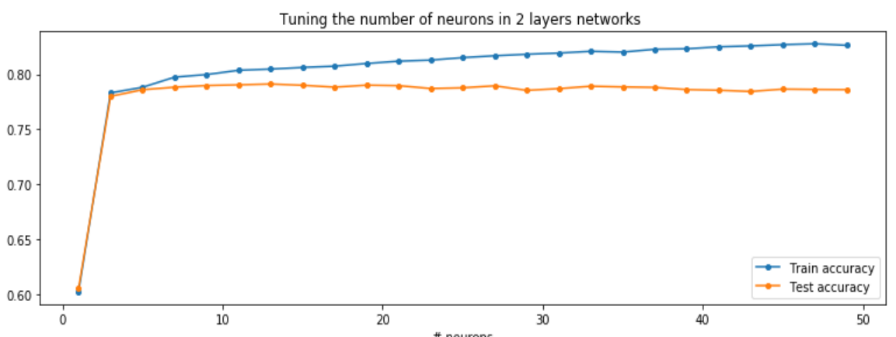
# **Artificial Neural Networks**

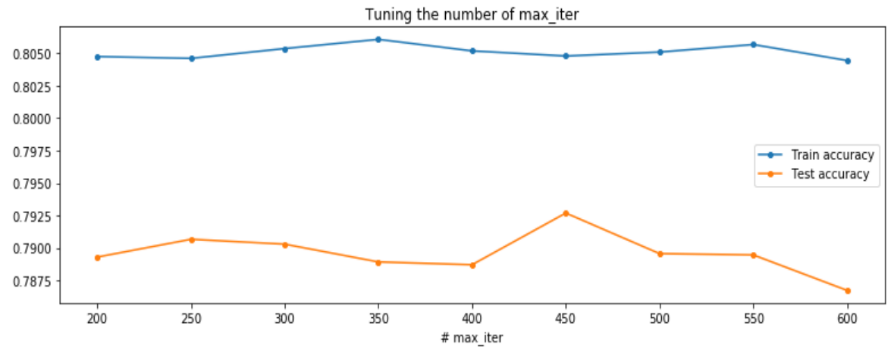
הקונפיגורציה שנלמדה ע"י המודל בערכי ברירת המחדל:

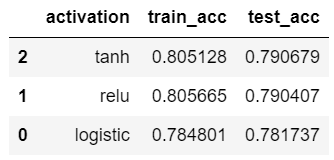
* קיימים 9 נוירונים בשכבת הכניסה (ככמות הפיצ'רים במודל פלוס 1).
* ישנה שכבה חבויה אחת ובה יש 100 נוירונים.

אחוז הדיוק המתקבל על סט האימון הוא 0.808 עם סטיית תקן של 0.002, ואחוז הדיוק המתקבל על סט הוולידציה הוא 0.79 עם סטיית תקן 0.014. ניתן להסיק מתוצאות אלה כי הערכים הדיפולטיביים עבור מודל זה נותנים ביצועים יחסית טובים. בהמשך נבדוק האם אנחנו יכולים לשפר אותם ואת אחוזי הדיוק שלנו.

|  |  |  |
| --- | --- | --- |
| פרמטר | טווח ערכים | מוטיבציה ומשמעות |
| activation | logistic, tanh, relu | פונקציית האקטיבציה עבור השכבה החבויה - כלומר כל פונקציה תוציא פלט אחר. נרצה לבדוק פונקציות אקטיבציה שונות על מנת למצוא את הפונקציה האופטימלית עבור משימת הלמידה שלנו. |
| max\_iter | 200-600  בקפיצות של 50 | כמות הפעמים שהרשת תעדכן את המשקולות שלה. ככל שנעדכן יותר - כך נמזער פונקציית הloss יותר. נרצה לבדוק אם ככל שנגדיל את מספר האיטרציות, פונקציית הloss אכן תקטן (כעת ערך ברירת המחדל הוא 200 איטרציות). |
| hidden\_layer\_size | 2 layers בעלי אותו גודל נוירונים (בדקנו בין 1 ל-50) | מספר הנוירונים בכל שכבה ומספר השכבות משפיעות על מספר הקשרים בין השכבות. הפרמטר הדיפולטיבי הוא שכבה אחת המכילה 100 נוירונים , רצינו לבדוק אם שכבות נוספות בגדלים שונים יוכלו להעמיק את הקשרים במודל ולשפרו. |

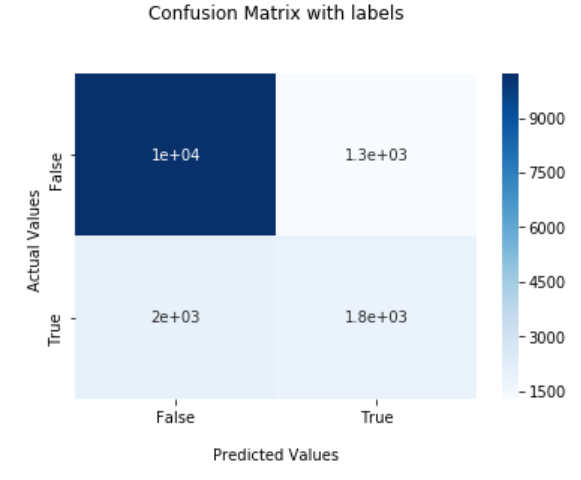
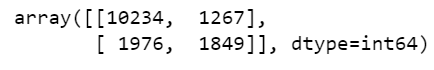
****עבור hidden\_layer\_sizes - נראה כי 2 שכבות חבויות בגודל 13 נוירונים כל אחת מביאה את אחוז הדיוק הטוב ביותר על סט האימון (שהינו 0.805) ועל סט הוולידציה (שהינו 0.781).

עבור max\_iter - מתוך הטווח שבדקנו נראה כי מספר האיטרציות האופטימלי עבור המודל שלנו הינו 450, כאשר אחוז הדיוק הטוב ביותר על סט האימון הינו 0.805 ועל סט הוולידציה הינו 0.793.

עבור activation - ניתן לראות כי פונקציית האקטיבציה בעלת אחוז הדיוק הטוב ביותר היא פונקציית tanh, אבל פוקנציית relu נותנת אחוזי דיוק כמעט זהים גם היא. אחוז הדיוק הטוב ביותר על סט האימון הינו 0.805 ועל סט הוולידציה הינו 0.79.

לפי צירוף הפרמטרים הטוב ביותר שמצאנו מהניתוח שלעיל: activation=tanh, max\_iter=450, hidden\_layer\_sizes=(13,13) - אחוז הדיוק על סט האימון הינו 0.805 עם סטיית תקן של 0.003 ועל סט הוולידציה הינו 0.789 עם סטיית תקן של 0.016.

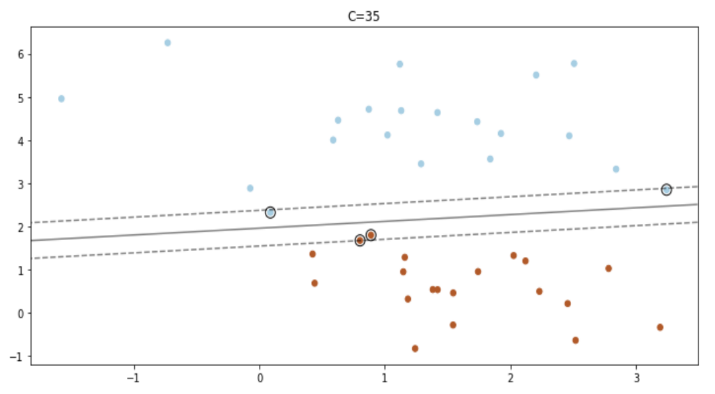
לא קיימים הבדלים משמעותיים בין תוצאות סעיף 1 לבין הפרמטרים שבחנו. כלומר נראה כי מודל עם שכבה חבויה אחת בעלת 100 נוירונים, פונקציית אקטיבציה relu ומקס' איטרציות 200 נותנת אחוז דיוק כמעט זהה למודל עם 2 שכבות חבויות בעלות 13 נוירונים, פונקציית אקטיבציה tanh ומקס' איטרציות 450. יתכן כי קיימים פרמטרים נוספים שאם נכוון אותם, נוכל להגיע לאחוז דיוק גבוה יותר.

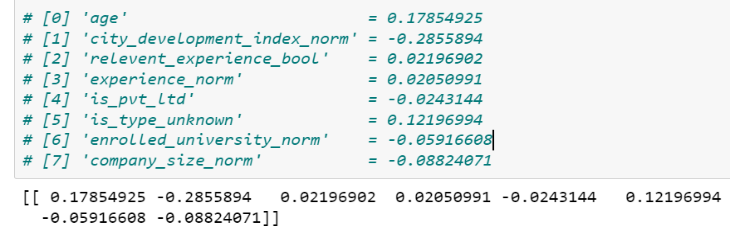
מטריצת מבוכה - ניתן להסיק כי המודל שלנו מזהה יחסית טוב כאלו שמעוניינים להשאר במקום העבודה (10,234 מתוך 11,501 - 88.98%), ומזהה פחות טוב כאלו שרוצים לעזוב (1976 מתוך 3825 - 51.66%). זאת, משום מהדאטה סט שלנו אינו מאוזן.

**SVM**

בדקנו באמצעות grid\_search את הפרמטרים הבאים:

|  |  |
| --- | --- |
| פרמטר | טווח ערכים |
| C | בין 1-50 |
| loss | hinge, squared\_hinge |

מצאנו כי הקונפגורציה המיטבית הינה כאשר C=35 ו-loss=squared\_hinge. במודל הנבחר, אחוז הדיוק המתקבל על סט האימון הוא 0.748 עם סטיית תקן של 0.017, ואחוז הדיוק המתקבל על סט הוולידציה הוא 0.746 עם סטיית תקן 0.029.

משוואת הקו הישר המפריד - עבור כל פיצ'ר הוצאנו את השיפוע שלו במשוואת הישר, כך שככל שהשיפוע (בערך מוחלט) גדול יותר, כך יש לו חשיבות גדולה יותר. ניתן לראות שלדירוג העיר השיפוע המשמעותי ביותר ותואם את המסקנות הקודמות שלנו. עם זאת, נראה כי היחס בין הפיצ'רים האחרים השתנה (כמו למשל, ניתן לראות כי השיפוע של compnay\_size\_norm קטן יותר משל age, לעומת הפעמים הקודמות).

# **Unsupervised Learning - Clustering**

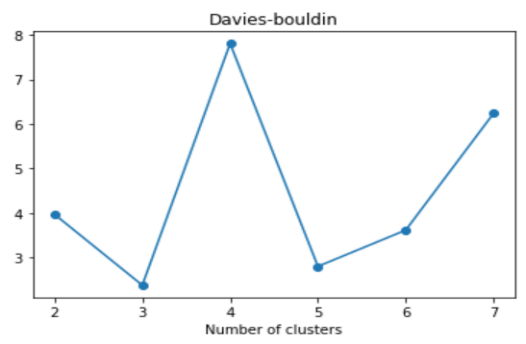
את האלגוריתם נריץ על סט הנתונים שלנו ללא הלייבלים. לעומת המודלים הקודמים, בהן משימת הלימוד הייתה סיווג, כאן משימת הלימוד היא clustering, כלומר למצוא קבוצות שונות בסט הנתונים שלנו שאין בו לייבלים.

הרצת האלגוריתם K-medoids - סט הפיצ'רים אשר בחרנו להשתמש הינם הפיצ'רים אשר היו בעלי המשמעות הגדולה ביותר לפי המודלים הקודמים: city\_development\_index\_norm, age, compnay\_size\_norm, enrolled\_university\_norm. בחרנו בסט פיצ'רים זה על מנת לבדוק באמצעות הclustering האם פיצ'רים אלו מספיק טובים להבחנה בין שני הקלאסים בדאטה סט המקורי שלנו. המטריקה בה נשתמש למדידת המרחק הינה gower מכיוון שאנו רוצים לחשב מרחק בין ישויות שיש להם תכונות מספריות וקטגוריאליות.

היינו רוצים לראות שני קלאסטרים עם מרחק יחסית גדול, כאשר אחד ייצג את העובדים אשר מעוניינים לעזוב והאחר את אלו המעוניינים להשאר.

הקריטריונים אשר בחרנו להשוואה הם מדד הם:

מדד Silhouette Coefficient - משקלל לכל סמפל את המרחק הממוצע שלו משאר הסמפלים באותו הקלאסטר, יחד עם המרחק הממוצע שלו משאר הסמפלים בקלאסטרים האחרים. עבור מדד זה נבחר את הערך הגבוה ביותר. על פי מדד Silhouette מספר המחלקות בו נבחר יהיה 2 מחלקות.



מדד Davies-Bouldin Index - משקלל לכל מודל (בעל K אחר) את רמת הדמיון בין הקלאסטרים השונים על פי גודלם של הקלאסטרים והמרחקים ביניהם. עבור מדד זה נבחר את הערך הנמוך ביותר. על פי מדד Davies-bouldin מספר המחלקות בו נבחר יהיה 3 מחלקות.

אם נתחשב בשני המדדים, מספר המחלקות שנבחר יהיה 5, אשר לפי הגרפים הוא השני הגבוה ביותר במדד Silhouette והשני הנמוך ביותר במדד Davies-bouldin.

משום שמספר המחלקות שבחרנו הינו 5, לא נוכל לייחס משמעות לכל קלאסטר - משום שמספר הקלאסטרים גדול כעת ממספר הקלאסים בdataset המקורי. אם היינו בוחרים ב-2 קלאסטרים, היינו יכולים לשער שאחד מייצג את העובדים שרוצים להישאר והשני את העובדים שרוצים לעזוב.

# השוואה בין המודלים - **Evaluation**

מבין שלושת המודלים, המודל אשר נבחר הוא Decision Tree. ניתן לראות כי אחוז הדיוק ב-MLP גבוה ב-0.001, אך סטיית התקן שלו גבוהה יותר ולכן החלטנו לבחור ב-Decision Tree. בנוסף, מודל זה הינו מהיר יותר מ-MLP ובעל יכולת הסברה גבוהה.

|  |  |  |
| --- | --- | --- |
| מודל | אחוז דיוק על ה-**test** | אחוז דיוק על ה-**validation** |
| **Decision Tree** | **0.798 +/- 0.002** | **0.788 +/- 0.013** |
| ANN - MLP | 0.805 +/- 0.003 | 0.789 +/- 0.016 |
| SVM | 0.748 +/- 0.017 | 0.746 +/- 0.029 |

# שיפור המודל הנבחר – **Improvements**

מסקנות עיקריות שעלו לאורך העבודה הנוגעות לאופי הנתונים והמודל שנבחר (DTT):

* ייתכן כי מספר הפיצ'רים אשר בחרנו בחלק ה-feature selection הינו גדול מדיי ואולי אפילו גורם לרעש בחיזוי, משום שניתן היה לראות לאורך העבודה כי קיימים רק שניים-שלושה פיצ'רים משמעותיים למודל. אנחנו חושבים לפתור זאת באמצעות ביצוע חלק זה שוב באמצעות בחירה מחדש רק באותם שלושת הפיצ'רים. אנחנו נצפה לשפר את הדיוק של המודל באמצעות כך.
* מודל הdecision tree אמנם נתן את אחוז הדיוק הגבוה ביותר לעומת המודלים האחרים, אבל אנחנו תוהים אם התנאי הבודד אשר בחר בכל שלב הינו אופטימלי. לאחר קריאה באינטרנט, ננסה להשתמש בrandom forest אשר אינו מסתמך על תנאי אחד בלבד בכל שלב - הוא משתמש במספר עצי החלטה וקובע תנאי מורכב יותר בכל שלב. אנחנו נצפה לראות שינוי בלוגיקה של העץ ולשפר את הדיוק.

ביצוע השיפור:

בחלק של ה-feature selection השארנו את רק את הפיצ'רים:company\_size\_norm, city\_development\_index\_norm, age.

השתמשנו ב-CV search grid עבור המודל RandomForestClassifier שמוצא את המודל הטוב ביותר, כאשר הפרמטרים שבדקנו עבורו היו max\_depth, criterion, n\_estimators. הרצנו את המודל החדש על הנתונים שלנו - ואחוז הדיוק על ה-validation set נשאר זהה, כלומר לא שיפרנו את המודל (אחוז הדיוק הוא 0.788 עם סטיית תקן של 0.015).

לדעתנו, המודל החדש לא שיפר את אחוז הדיוק משום ש-random forest הינו מודל המשתמש במספר decision trees - לכן יש דימיון בין המודלים. אכן ציפינו לקבל שיפור בדיוק מכיוון ש-random forest מבצע מספר עצי החלטה, אך כנראה שה-grid\_search עשה את תפקידו ובחר את הפרמטרים האופטימלים בשני המודלים כך שהם יצאו שווים כמעט במדויק.

# הגשת חיזויים סופיים

קובץ CSV בשם G6\_ytest מצורף.

# 

# נספחים

**תיקונים עבור חלק א:**

* בחלק ה-pre proccessing - החלטנו להשאיר את השדה city\_development\_index כפי שהוא ולא לחלק אותו לעשירונים.
* בחלק ה-feature extraction - לפי מתודולוגיית החילוץ knowledge based features חילצנו את הפיצ'רים הבאים:
* **is\_first\_job** = if [last\_new\_job == “never”] and [relevant\_experience == “No relevant experience”] then 1 else 0.
* **age** = (if [education\_level == “Primary School”] then 15, elif [education\_level == “High School”] then 18, elif [education\_level == “Graduate”] then 22, elif [education\_level == “Masters”] then 27, else 31) + experience
* בחלק ה-feature representation - שינינו את אופן הייצוג עבור הפיצ'רים שהוצגו בחלק א והוספנו ייצוג עבור משתנים קטגוריאליים:

|  |  |
| --- | --- |
| פיצ'ר | ייצוג |
| city\_development\_index | נרמול ערכים רציפים (חלוקה בערך המקסימלי) |
| relevent\_experience | No relevent experience = 0  Has relevent experience =1 |
| experience | 1> יהפוך ל-0.5 ו-20< יהפוך ל-21 וננרמל את הערכים בחילוק בערך המקסימלי. |
| compnay\_type | המרה למשתני דמה. |
| last\_new\_job | 4< יהפוך ל-5 ו-never ל-0 וננרמל את הערכים בחילוק בערך המקסימלי. |
| enrolled university | no enrollment = 0  part time = 0.5  full time = 1 |
| major\_discipline | המרה למשתני דמה. |
| education\_level | 0 = primary school  0.25 = high school  0.5 = graduate  0.75 = masters  1 = PHD |
| company\_size | Unknown = 0  <10 = 1/8  10-49 = 2/8  50-99 = 3/8  100-500 = 4/8  500-999 = 5/8  1000-4999 = 6/8  5000-9999 = 7/8  10,000+ = 1 |
| last\_new\_job | never = 0  1 = 0.2  2 = 0.4  3 = 0.6  4 = 0.8  >4 = 1 |
| training\_hours | נרמול הערכים, חלוקה בערך המקסימלי |

* בחלק ה-feature selection - החלטנו להשאיר את השדות המפורטים בטבלה (כל השדות אשר הראו קורלציה גבוהה מ-10% עם ה-target:

|  |  |
| --- | --- |
| ציון קורלציה | **feature name** |
| 0.151304 | age |
| 0.342869 | city\_development\_index\_norm |
| 0.127636 | relevent\_experience\_bool |
| 0.176218 | experience\_norm |
| 0.170891 | Is\_pvt\_ltd |
| 0.239649 | is\_type\_unknown |
| 0.149797 | enrolled\_university\_norm |
| 0.178204 | Company\_size\_norm |

* לא נבצע PCA משום שאינו מתאים למשתנים קטגוריאליים. בחלק ב' נמשיך עם הפיצ'רים שבחרנו ב-feature selection.