

חלק ג' – עיבוד נתוני המעבדה (מתחילים לעבוד)

6. ביצוע רגרסיה לינארית (19 סעיפים לכלי שימושי)

רגרסיה לינארית היא השם של האלגוריתם שמקבל אוסף של נקודות (כלומר, וקטור x ווקטור y) ומחשב את הקו הישר שעובר הכי קרוב אל הנקודות. הרגרסיה מחזירה את השיפוע (slope) ואת האיבר החופשי (intercept) של הקו הישר. כמו כן, היא מחזירה מקדם שנקרא rvalue שמכמת עד כמה הנקודות "רחוקות" מהקו. כש- $rvalue=1$ הנקודות נמצאות ממש על הקו. אם הן נמצאות לידו, אז הערך של rvalue קטן יותר.

רגרסיה לינארית היא כלי מאוד שימושי ואתם תשתמשו בה לעיתים קרובות במעבדה.

1. פתחו cell חדש בשם linear regression
2. הכינו וקטור בשם xs שמכיל 40 ערכים בין 0 ל- 2π (חשבו, באיזו ספרייה יהיה שמור הערך של π ?).
3. הכינו וקטור בשם ys ששווה $3 * xs - 0.2$.
4. בחלון help, חפשו linregress (זו הפונקציה שייבאתם בתחילת הקוד שלכם)
5. מצאו דוגמה לשימוש בפונקציה והעתיקו אותה אל הקוד שלכם. את זאת שנראית לכם הכי פשוטה.
6. חפשו בתיעוד של מה שהפונקציה מחזירה (Returns) וגרמו לה להחזיר גם את השגיאה ב-slope ואת השגיאה ב-intercept.
7. בצעו רגרסיה לינארית על xs,ys.
8. שמרו לתוך משתנים את הערכים של slope, intercept, rvalue, stderr, intercept_stderr
9. האם הרגרסיה מצאה את המקדמים הנכונים של הקו הישר? מה השגיאה שהיא העריכה לכל אחד מהם? מהו ה-rvalue?
10. צרו וקטור חדש ys2 שווה $2 * xs + 1$
11. הפכו את האיבר ה-13 לערך של 30.
12. בצעו רגרסיה לינארית נוספת, על xs,ys2, ושמרו את המקדמים במשתנים חדשים. האם הרגרסיה הצליחה למצוא את המקדמים?

בואו נראה זאת בגרף:

13. צרו figure – fig3 חדש.
14. ציירו את xs,ys בתור "plot 1", כנקודות שחורות.
15. ציירו את xs,ys2 בתור "plot 2", כנקודות כחולות.
16. ציירו את עקום הרגרסיה הראשונה בעזרת:
`plt.plot(xs, slope*xs + intercept, 'r-', label='regression 1')`
17. ציירו את עקום הרגרסיה השנייה באופן דומה, אבל בצבע תכלת cyan
18. אל תשכחו להוסיף מקרא, grid, וכותרות לצירים!
19. התבוננו בגרף. מסקנתכם?

המסקנה שהייתם אמורים להסיק:

רגרסיה היא כלי נהדר, אבל תמיד צריך לראות שאין טעויות בנתונים כשמשתמשים בה – ולכן צריך לצייר אותם בגרף.