למידה חישובית ממידע: הרצאה 5 – Bayes Variations and EM

Density Estimation, Gaussian Mixture Models, EM

- **Density Estimation** – בהינתן סט דגימות xi נמצא את פונקציית הצפיפות PDF (Probability Density Function) שמסבירה באופן הטוב ביותר את הדאטה.
- נדבר על מודלים המתאימים מצב בו יש שתי שכבות השולטות בהתפלגות. במקרה כזה ראשית יש להחליט לאיזה "ענגה" אני שייך, ומתוכו להחליט על סיווג סופי.

אלגוריתם EM = Expectation Maximization

C = (X, Z)

- C: complete data ("augmented data")
- X: observable data ("incomplete" data)
- Z: hidden data ("missing" data)

- D: the actual observed X values (from a sample)

- שיטה איטרטיבית עבור הערכת פרמטרים כאשר שכבות של דאטה חסרים מה-observation. אלגוריתם זה כולל שני שלבים: (E) Expectation ו-(M) Maximization.
- נגדיר את המשתנים באלגוריתם EM: D הוא קבוצה של data points (ה-observed data), טטה היא וקטור הפרמטרים, EM הוא אלגוריתם איטרטיבי עבור מציאת טטה ML (maximum likelihood).
- אנחנו נניח שיש שתי רמות של דאטה. יש את הדאטה השלם "augmented data", אנחנו לא נראה את הדאטה המלא. x הוא הדאטה שאנחנו רואים "observable data" ו-z הוא ה-hidden data הדאטה הנסתר מעיני (לא ממש חסר). התצפיות שלנו יהיו בתוך D.

בחירה רנדומלית בין 2 מטבעות

יש לנו 2 מטבעות עם הסתברויות Pa, Pb (אינם משלימים! יכולים להיות 0.45 ו-0.65).
אחד המטבעות יבחר עם ההסתברויות Wa, Wb (משלימים אחד את השני ל-1: Wa = 1 – Wb). לאחר מכן מטילים את המטבע הנבחר 10 פעמים. נתבונן בתוצאות של הניסוי ונערוך אותו מספר פעמים. אם ידוע לנו איזה מטבע נזרק בכל סט, אזי נוכל לבצע MLE ולקבל את ה-P-ים ואת ה-W-ים. אבל אנחנו לא יודעים זאת. לכן EM יכול לעזור כאן.
אנחנו רואים את התוצאה של 10 הטלות 8 פעמים אבל אנחנו לא יודעים בכל פעם מהפעמים הללו איזה מטבע הוטל. נרצה להסיק מהניסויים האלה את המטבע הבא ו -10 ההטלות שהוא יניב.
האיור עם הכתומים-ירוקים אינו ידוע לנו, אנחנו נמצאים במצב בו כל המטבעות כחולים – hidden עבורנו.

אלגוריתם EM – התהליך הרעיוני

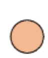
- נאתחל בסט של הפרמטרים ההתחלתיים של המודל כולל הפרמטרים של z – בדוגמה שלנו יש 3 פרמטרים Pa, Pb, Wa (Wb יהיה המשלים ל-1 של a) . כלומר "ננחש" ערכים התחלתיים עבור הפרמטרים של המודל.
- נשתמש בפרמטרים אלו כדי "לשערך" את הערכים של הדאטה החבוי z, לפי התצפית שלנו (ה-observed data point) - עבור כל אחת מנקודות הדאטה נשאל האם הניחוש שנעשה מתאים?
- נשתמש בדאטה ה"שלם" כדי לעדכן את כל הפרמטרים (גם של z וגם של x | z). נמשיך בתהליך זה עד שנגיע להתכנסות.

בחזרה הדוגמה שלנו עם 2 המטבעות

1. **נאתחל** Pa = 0.6, Pb = 0.5, Ws = 0.5 (שני ה-w-ים), זהו הניחוש ממנו נתחיל. הניחוש לא מתבסס על כלום.
2. **נחשב את ה-responsibilities** – נחשב עבור כל אינסטנס, כל ניסוי של 10 הטלות, נחשב את הפוסטריור: נציב את הניחוש שלנו בהתפלגות בינומית בהינתן שאנו יודעים איזה מטבע נבחר, לכן יש שני חישובים לכל ניסוי:

$$P_A(x_1) = w_A \binom{10}{9} 0.6^9 0.4^1 = 0.04$$

$$P_B(x_1) = w_B \binom{10}{9} 0.5^9 0.5^1 = 0.01$$

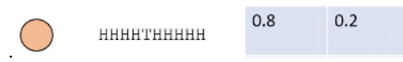
חישוב לדוגמה - עבור השורה -  HHHHTHHHHH

חישוב זה הוא "כמעט הפוסטריור" חסר כאן החלוקה באווידנט, המכנה. המכנים יהיו שונים לכן נחלק במכנים, לומר נחלק את התוצאות בסכום של שתי ההסתברויות שקיבלנו בדוגמה ובכך נקבל את **הפוסטריור האמתי** -

$$r(x_1, A) = \frac{0.04}{0.05} = 0.8$$

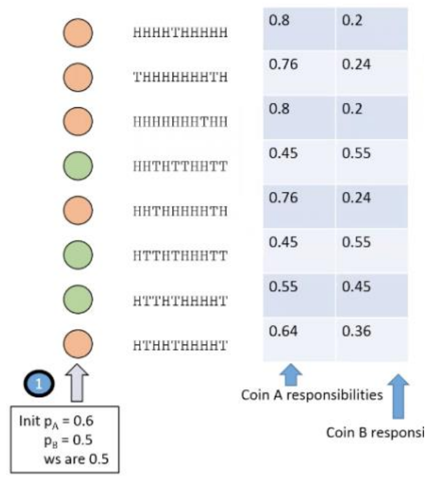
$$r(x_1, B) = \frac{0.01}{0.05} = 0.2$$

3. נציב את התוצאות בטבלת ה-Resposibilities העמודה השמאלית מייצגת את מטבע A, והימינית את מטבע B:



טבלת ה-responsibilities מתארת כמה ההטלה חושבת שהיא הגיעה ממטבע A, וכמה היא חושבת שהיא הגיעה ממטבע B. הדבר הגיוני מפני שהענקנו עבור מטבע A הסתברות גבוהה יותר ל-H מאשר מטבע B. לכן האחריות באותה השורה תמיד תסכום ל-1.

4. נמשיך למלא את הטבלה באותו האופן המתואר בסעיף 2 ונקבל את הטבלה הבאה:



והתקבל וקטור ה-responsibilies של A וקטור ה-responsibilies של B. מהווקטורים שהתקבלו מסתמן שהסבירות לשני המטבעות איננו חצי-חצי כפי שניחשנו את ה-W-ים ויש נטייה יותר ל-A.

5. **<u>לכן, בשלב הבא יהיה עלינו לעדכן את ה-W-ים. העדכון מתבצע כך:</u>**

ניקח את הממוצע המשוקלל של וקטורי האחריות של כל מטבע ונהפוך את התוצאה ל-W החדש של כל מטבע: מימין הנוסחה הכללית ומשמאל החישוב המתאים לדוגמה שלנו.

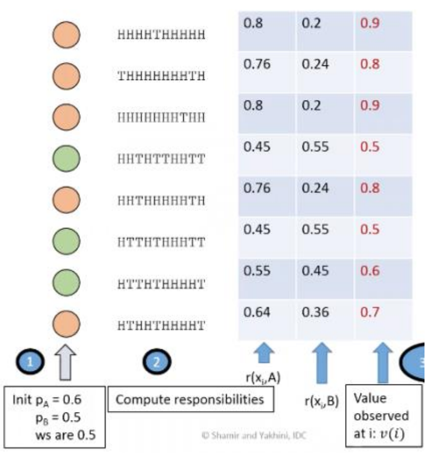
$$New\ w_A = \frac{1}{N}\sum_{i=1}^{N} r(x_i, A)$$

$$New\ w_B = \frac{1}{N}\sum_{i=1}^{N} r(x_i, B)$$

$$New\ w_A = \frac{1}{8}\sum_{i=1}^{8} r(x_i, A) = 0.65$$

$$New\ w_B = \frac{1}{8}\sum_{i=1}^{8} r(x_i, B) = 0.35$$

6. **<u>לאחר עדכון ה-W-ים, עלינו לעדכן את Pa ו-Pb. העדכון מתבצע באופן הבא:</u>**



אנחנו עושים סוג של voting, כל אחד מהניסויים קיבל "ציון אחריות", כמה הוא חושב שהוא הגיע ממטבע A וכמה הוא חושב שהוא הגיע ממטבע B, לכן נבצע **voting ממושקל**;
נחשב את ה-MLE = הפוסטריור לכל ניסוי, אם ניסוי 1 הניב 9 heads הוא יחשוב ש-P של המטבע שנבחר הינו 0.9 (9 לחלק ל-10). ונעניק לתוצאה הזו משקול על פי כמה ניסוי 1 חושב שהוא הגיע ממטבע A, כלומר 0.8, ובחישוב השני משקול 0.2.
החישוב המלא של ה-MLE מופיע בוקטור האדום המסומן **ב-v(i).**

על כן החישוב הכללי יהיה מימין, והחישוב עבור הדוגמה שלנו מופיע משמאל:

$$p_A = \frac{1}{(New\ w_A)N}\sum_{i=1}^{N} r(x_i, A)v(i)$$

$$p_B = \frac{1}{(New\ w_B)N}\sum_{i=1}^{N} r(x_i, B)v(i)$$

$$p_A = \frac{1}{5.2}\sum_{i=1}^{8} r(x_i, A)v(i) = 0.745$$

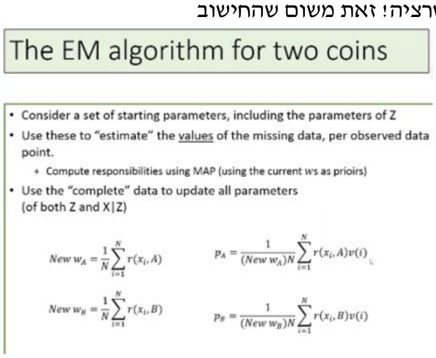
$$p_B = \frac{1}{3.8}\sum_{i=1}^{8} r(x_i, B)v(i) = 0.48$$

7. <u>כעת, יש לנו Ws חדשים, Pa ו-Pb:</u>

$$w_A = 0.65$$
$$w_B = 0.35$$
$$p_A = 0.745$$
$$p_B = 0.48$$

על כן נחזור על כל התהליך החל מסעיף 2 עד התכנסות (לא הגדרנו באופן רשמי תנאי עצירה).

באיטרציה הבאה הוקטור האדום לא ישתנה, הוא למעשה לא ישתנה באף איטרציה! זאת משום שהחישוב מבוסס על הדגימות שלנו בלבד, שאינן משתנות. שאר העמודות לעומת זאת אכן ישתנו, וככל שהן ימשיכו להשתנות נמשיך "לעבוד".



Gaussian Mixture Models

$$f(x) = \sum_{i=1}^{k} w_i f_i(x)$$

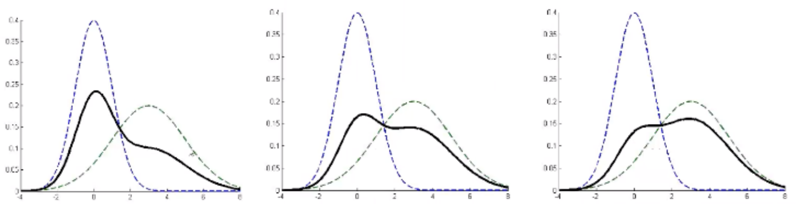
נגדיר: X הוא משתנה מקרי Gaussian Mixture (תערובת גאוסיאנית) אם פונקציית הצפיפות של ההתפלגות של X הינה:

k מוגדר מראש – למשל תערובת גאוסיאנית עם k=5 , wi הם משקלים עבור fi פונקציות, כך שסכום ה-wi-ים סוכמים ל-1.

$$f_i(x) = \frac{1}{\sigma_i\sqrt{2\pi}} e^{-(x-\mu_i)^2/2\sigma^2}$$

כאשר לכל fi פונקציות הצפיפות היא פונקציית הצפיפות הגאוסיאנית (לכן כל אחד מהם הוא גאוסיאן):

להלן דוגמאות של השפעת המשקלים wi על פונקציית התערובת הגאוסיאנית (בשחור) בהינתן שיש בידינו שני גאוסיאנים k=2 (ירוק וכחול)



EM for GMMs

- **שלב 1: Expectation (E-step)**

נשערך את ה"responsibilities" עבור כל נקודת דאטה Xi, לכל גאוסיאן (k גאוסיאנים), על ידי שימוש בפרמטריים הנוכחיים (מה הפוסטריור של כל הגאוסיאנים בהינתן נקודת הדאטה הזו).

- **שלב 2: Maximization (M-step)**

נשערך מחדש את הפרמטרים (המשקלים Ws, התוחלות mui-ים, וסטיות התקן sigma-ות) בעזרת ה-responsibilities הקיימים.
כלומר – כל נקודת דאטה, x, תורמת לפרמטרים של כל קומפוננט (מרכיב) בגאוסיאן, Gk, יחס לרספונסיביליטי שלו: r(x, Gk).

אם יש בידי 4 גאוסיאנים, יש לנו 11 פרמטרים: 2 של כל גאוסיאן (8 סה"כ), וההסתברות לקבל כל אחד מהם (4, אבל הרביעי משלים את השלושה ל-1, ולכן סה"כ צריך 3 הסתברויות).

1. **לאחר אתחול הפרמטרים (ניחוש) נבצע חישוב ב-responsibilities שהן למעשה הפוסטריור של גאוסיאן k בהינתן x (נקודת דאטה) יערך על ידי הנוסחה הבאה:**

$$r(x,k) = \frac{w_k N(x|\mu_k , \sigma_k)}{\sum_{j=1}^{K} w_j N(x|\mu_j , \sigma_j)}$$

N הוא סימון להסתברות של (במונה) x בהינתן הגאוסיאן k, שהוא בעל הפרמטרים מיוk וסיגמאk. ומכיוון שזהו פוסטריור עלינו לחלק בסכום הממושקל של ההסתברויות N של דגימת הדאטה x בהינתן גאוסיאן j עבור j מ-1 עד k.

2. נעדכן את ה-w-ים:

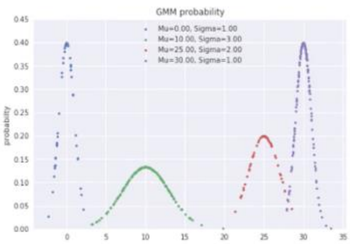
$$New\ w_j = \frac{1}{N}\sum_{i=1}^{N} r(x_i, j)$$

נעדכן את ה-miu-ים של כל גאוסיאן k:

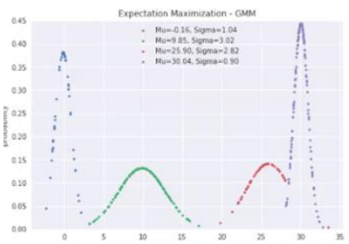
$$New\ \mu_k = \frac{1}{(New\ w_k)N}\sum_{i=1}^{N} r(x_i,k)x_i$$

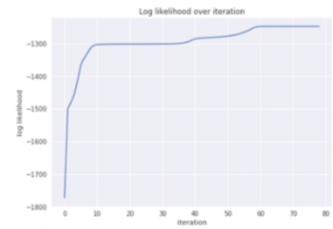
נעדכן את הסיגמות של כל גאוסיאן k:

$$(New\ \sigma_k)^2 = \frac{1}{(New\ w_k)N}\sum_{i=1}^{N} r(x_i,k)(x_i - New\ \mu_k)^2$$



דוגמת הרצה: יצרנו 4 גאוסיאנים עם פריור (w) 0.25 לכל גאוסיאן, ונרצה לנסות לחזות אותם על פי נקודות שהגרלנו מתוכן. נרצה לדעת מאיזה גאוסיאן הגיעו הנקודות הללו (במציאות אנחנו לא רואים את הצבעים אלא רק את הנקודות. למעלה מופיע גרף המתאר את הדאטה שלמדנו, ולמטה זו תוצאה של למידת EM על GMM לאחר 80 איטרציות, ניתן לראות שהגענו לשערוך דומה של ערכי הסיגמות וה-miu-ים של הגאוסיאנים שיצרנו מלכתחילה (הגענו למספרים דומים מאוד).





נשים לב שהפסקנו להריץ לאחר 80 איטרציות מכיוון שניתן לראות, שמבחינת התכנסות של פונקציית ה-log-likelihood כי השינוי **החל להיעשות בלתי ניתן להבחנה** סביב ה-80~ ולהלן גרף המתאר את שינוי ההתנהגות של פונקציית ה-loglikelihood לאורך האיטרציות השונות של האלגוריתם.

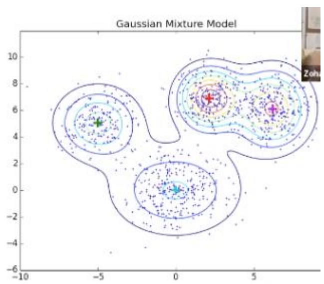
Multidimensional GMM EM

פונקציית הצפיפות של תערובת גאוסיאנית d ממדית היא מהצורה:

$$f(\vec{x}) = \sum_{i=1}^{k} w_i f_i(\vec{x})$$

כאשר כל fi היא פונקציית צפיפות גאוסיאנית d-ממדית.

כמה פרמטרים יהיה עלינו ללמוד במקרה של תערובת גאוסיאנית רב ממדית?



- K משקולות – w (למעשה k-1 מפני שהאחרון משלים לנו ל-1)
- d תוחלות לכל גאוסיאן.
- מספר הערכים שנצטרך ללמוד עבור ערכי מטריצת השונות המשותפת יהיה:

$\binom{d+1}{2}$ matrix entries:
d variances + $\binom{d}{2}$ covariances <u>per Gaussian</u>

<u>הערות על האלגוריתם של EM</u>

- אחד השימושים של EM הוא עבור Clustering (נחזור לנושא זה בהמשך).
- EM לא מחליט על מספר הקומפוננטות של המודל שאנו לומדים, בנוסף לא מבטיח לנו מקסימום גלובלי (כמו כל תהליך איטרטיבי אחר למעשה...), לא מעניק ביטחון שנגיע לאופטימום – למרות שבפרקטיקה כן נגיע קרוב לשם. לא תמיד יהיה נתון לנו הנוסחאות המתמטיות עבור מקרה ונצטרך לפתח נוסחה עבור מקרה ספיציפי בשביל להפעיל עליו EM.

יתרונות של EM

- התכנסות: בכל איטרציה של האלגוריתם, ה-likelihood משתפר מהאיטרציה הקודמת.
- EM מתאים עבור (רוב) משפחות המודלים ועבור כל מספר של פרמטרים.

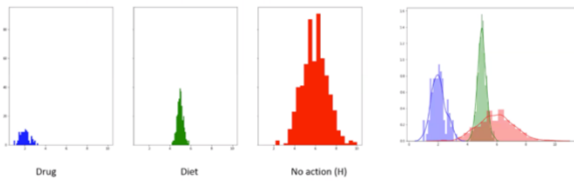
חסרונות של EM

- התכנסות יכולה לעתים להיות איטית מאוד עבור דגימות מסוימות וכן ההתכנסות תלויה מאוד בכמות המידע החסר.
- כמו כל גישת למידה, אנחנו עובדים על training data. ומאוד חשוב להימנע מ-overfitting (למשל מספר הערכים של z).
- שערוך מודלים הם תלויי משתמש, ולא נקבעים על פי עקרון.
- אין הבטחה ל-global optimum (תתכן היתקעות על מקסימום לוקאלי למשל).
- הערכים ההתחלתיים (הניחוש) חשובים ונדרשת הפעלה על מספר "ניחושים" (או קבוצת "ניחושים") כדי להגיע להתכנסות מתאימה.

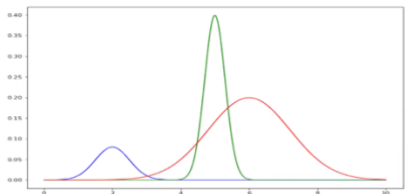
Variations on Bayes Classifiers

<u>פונקציות מחיר כלליות</u>

עד כה למדנו רק על פונקציית loss 0/1. נניח שיש צוות שמפתח בדיקה, עבור סוג של מחלה, המניבה מספר. באוכלוסייה שלנו **יש אנשים בריאים** שלא צריכים התערבות, יש אנשים שאם יעשו **דיאטה** יוכלו להחלים, ויש אנשים שיחלימו על ידי **טיפול תרופתי**. נרצה לפתח מסווג בייסיאני אשר מקבל דגימה (אדם מהאוכלוסייה) וידע להגיד האם הוא בריא או צריך דיאטה או צריך טיפול תרופתי. נייצר training data עם 100 דגימות מתוך התפלגות המייצגת אנשים שזקוקים לטיפול תרופתי, 300 מתוך התפלגות המייצגת אנשים הזקוקים לדיאטה, 600 מתוך התפלגות המייצגת בריאים. <u>מכאן שאנו כבר מסיקים מהו הפריור עבור כל קבוצה</u>: 0.1 עבור תרופתיים, 0.3 עבור דיאטה, 0.6 עבור בריאים.



להלן הדאטה שהוגרלה וכן ההיסטוגרמה של ה- class-conditional.
<u>בהינתן אדם, נרצה לדעת לאן הוא שייך</u>: לגרף האדום (בריא), לגרף הירוק (דיאטה) או לגרף הכחול (תרופתי). לשם כך נחשב פוסטריור-ים, את הפריור-ים כבר יש לי, מה שחסר לנו הוא ה-likelihoods. נוכל להפעיל **MLE של גאוסיאנים** על מנת למצוא את הפוסטריור-ים. ולאחר מכן נרצה להשתמש במסווג בייסיאני.



Bayes Classification using a loss function

על רקע הדוגמה, כמובן שפונקציית מחיר 0/1 לא ממש מספיקה "להעניש" על שגיאה מפני שיש לנו שלושה קלאסים, ובאופן גם פחות תיאורטי – לא נרצה לסווג מישהו חולה כבריא ולהעניק לו טיפול תרופתי לשווא או שמא לגרום לו לעשות דיאטה למרות שהוא בריא, וכן גם הדבר אינו שווה ערך ללשערך כי אדם חולה הוא אדם בריא! לכן נרצה להשתמש בפונקציית מחיר כללית יותר.

<u>נגדיר פונקציית loss = המחיר עבור החלטה שגויה</u>:

נניח כי יש לנו k קלאסים, כך שלכל Ai, i בין 1 ל-k. על פי התבוננות בדגימה x עלינו להחליט לאן דגימה זו משתייכת מבין הקלאסים Ai (על ידי הפעלת גישת בייס או MAP). החלטה שגויה מובילה להפסד loss! ההפסד תלוי באיזה j סווג באופן שגוי ל-i. נייצג זאת ע"י פונקציית מחיר:

$\lambda_{ij} = \text{Cost}\left(h(x) = A_i \ \wedge \ x \in A_j\right)$ (למשל, ראינו את פונקציית loss 0/1: $\lambda_{ij} = \begin{cases} 0 & i = j \\ 1 & i \neq j \end{cases}$)

נרצה לחשב את הסיכון בפונקציית loss כללית:

$$R(Choose\ A_i|x) = \sum_{j\neq i} \lambda_{ij} P(A_j|x)$$

המסווג הבייסיאני הוא זו שמביא ל**מינימום** את מחיר השגיאה. כלומר, באופן כללי, בהינתן דגימה x נסווג אותה לתוך:

$$C(x) = \underset{i}{\text{argmin}} \sum_{j\neq i} \lambda_{ij} P(A_j|x)$$

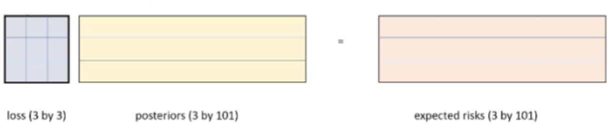
ההבדל הוא למעשה לא באלגוריתם הלמידה זהו אותו האלגוריתם שלמדנו כבר, אלא באלגוריתם המבצע שיבצע ויסווג באופן שונה, מפני שכעת אנחנו מתייחסים לפונקציית מחיר שונה. בגלל שהמחיר שונה – ההחלטה תהיה שונה.

כדי להשיג את ה-expected risk, לכל נקודה בדאטה (101 נקודות דאטה חדשות במקרה של הקוד כאן) יהיו 3 ערכים ל-3 הקלאסים, הקלאס שיקבל את הערך סיכון הנמוך ביותר הוא הקלאס אליו נסווג את הדגימה. על ידי הכפלת המטריצות הנ"ל נקבל את מטריצת ה-risk.

of the columns of the risks matrix

```
data = np.linspace(0, 10, 101)
def pred_with_risk(data, loss_matrix):
    posteriors = np.array([drug.pdf(data) * 0.1, diet.pdf(data) *0.3, healthy.pdf(data) * 0.6])
    return np.argmin(np.dot(loss_matrix, posteriors), axis=0)
```

expected risks

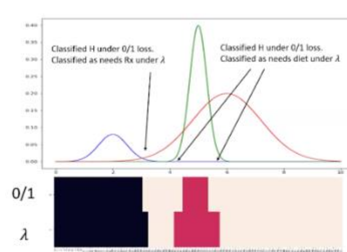


A more appropriate loss function

$$\lambda = \begin{array}{c} \\ \\ \\ \end{array}\begin{array}{c} \text{TRUE} \\ \begin{array}{ccc} Rx & Diet & H \end{array} \\ \begin{pmatrix} 0 & 100 & 300 \\ 800 & 0 & 100 \\ 1000 & 800 & 0 \end{pmatrix} \end{array} \begin{array}{l} \\ \\ Rx \\ Diet \\ H \end{array} \ \text{Predicted}$$

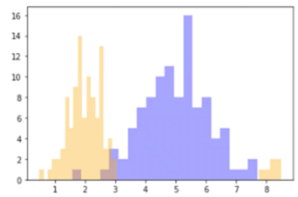
<u>דוגמה לחישוב הסיכון</u>: להלן מחיר "האמת" מול הפרדיקציה, כך שערכי האלכסון מייצגים סיווג נכון (אמת = פרדיקציה) ועל כן המחיר הוא 0. שורה 0 ועמודה 1 מתארת את המחיר שיעלה לסווג אדם שזקוק ב"אמת" לדיאטה, כאדם שזקוק לטיפול תרופתי = המחיר הוא 100. אנחנו אלה שקובעים את המטריצה הזו ואיתה עושים את הפרדיקציה.

להלן התוצאה של האלגוריתם סיווג מצד אחד תחת loss 0/1 לעומת התוצאה של האלגוריתם תחת למדא. ניתן לראות כי מהיות שהמחיר לסווג לא נכון אדם שמיועד לטיפול תרופתי כאדם בריא הוא מאוד גבוהה (1000) אזי לעומת הסיווג עם 0/1, הסיווג "עדין יותר/מצומצם יותר" מאשר ב-loss 0/1 (ניתן לראות זאת בבירור בין הסיווג האדום לשחור).



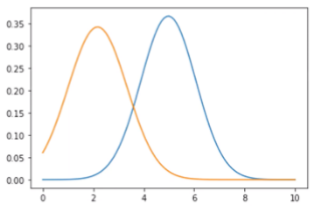
GMM Bayes

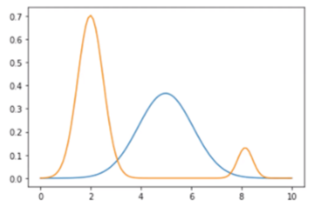
נניח כי יש לנו שני קלאסים A ו-B ולהלן ה-training data עבורם עם prior חצי-חצי. ונרצה לבנות מסווג בייסיאני על הדאטה הזה, לכן הדבר הראשון שנעשה הוא Gauss MLE על הדאטה במטרה לקבל את ה-posteriors של הדאטה ומכיוון שהפריורים שווים ניתן לסווג על ידי ה-likelihood.

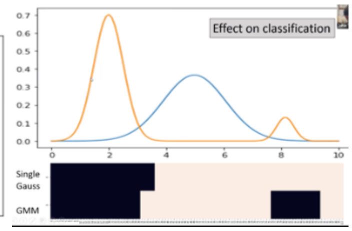


אבל אם במקום גאוס MLE נלמד מודל תערובת גאוסיאנים על ידי EM נקבל את הסיווג הבא (ה-class-conditionals) שכמובן יסב סיווג טוב יותר.

כפי שניתן לראות בהשוואה המסווגת הבאה (סיווג שחור מול כתום)







התנגשות זו עשוייה להתרחש גם בממדים שאינם יחידניים (גם בדו ממד וכדומה)

<u>הערה כללית עבור למידה כזו במטוסים : (למידה חישובית ומסווגים מסוג זה בעולם האמתי)</u>