

פרויקט בייסיאנים:

חלק א:

לפניך dataset שמתאר כמות לקוחות שנכנסות לרשת מסעדות טבעוניות הנקראת cooper branch בקנדה. נרצה לעשות תחזית כמות הצרכנים שייכנסו לחנות בעיר מסוימת ביום לפי שני משתנים מסבירים: כמות טבעונים שגרים בעיר ודירוג סוציו-אקונומי של העיר.

לפניך dataset עם העמודות הבאות:

מצב סוציו-אקונומי: socio_economic_profile.

מספר טבעונים: num_vegans.

עיר: city.

מספר לקוחות שנכנסו ביום: tot_customers.

האם הנתון שייך לאימון: is_train. אם זה true זה נתון שאיתו בונים מודלים בייסיאנים אם לא זה נתון שאיתו בודקים דיוק אחרי בניית המודל.

עליכם לטעון את datasetn על ידי השורה:

```
df = pickle.load(open('./data/group_'+str(id)+'regression.pkl', 'rb'))
```

משימות:

1. נא לבחור את העיר הראשונה שמופיעה לכם בdataset ולבנות איתה מודל רגרסיה לינארית שחזה את מספר הלקוחות שנכנסות ליום בעיר הזאת לפי מצב סוציו-אקונומי ומספר טבעונים. יש לבנות מודל רגרסיה עם אינטראקציה כלומר:

$$Y = b_0 + b_1 \cdot \text{num_vegan} + b_2 \cdot \text{socio_economic_profile} + b_3 \cdot \text{socio_economic_profile} \cdot \text{num_vegan}$$

נא להציג את משוואת הרגרסיה לפי הממוצעים של posteriors.

2. נא לעשות ניתוח שבודק האם mcmc התכנס להתפלגות האמיתית של posterior עבור אחד המשתנים לבחירתכם.

3. נא לעשות ניתוח posterior predictive על מנת לבדוק אם ההתפלגויות שבחרתם likelihood ולprior מתאימים כאן.

4. נא לחשב את mse עבור test set של העיר שבחרתם מול התחזית שלכם. יש להציג משוואה בגרף תלת מימדי את משוואת הרגרסיה והנתונים האמיתיים. כאשר ציר אחד זה מספר טבעונים, ציר שני זה מצב סוציואקונומי - והציר האופקי הוא מספר לקוחות ביום.

5. נא לבנות רגרסיה היררכית כך שלכל עיר יש את אותה משוואת רגרסיה מסעיף 1 אבל יש לכל הערים priors משותפים.

6. נא לחשב mse לכל עיר בנפרד על סמך test set.

7. יש לחזור על הניתוח של סעיף 1 עבור כל הערים ולחשב את mse. לאחר מכן יש להשוות mse בכל עיר בין המודל Unpooled והמודל ההיררכי. מה פרופוזיציות הערים שקיבלתם דיוק גדול יותר מודל ההיררכי.

8. למה מודל היררכי עשוי לעזור במקרה של הבעיה הזאת?

9. נא לבחור 2 ערים ולחשב את ה-cohen's d של הממוצע של מספר הלקוחות שנכנסים ביום לחנות. מה הסיכוי שהפרש תוחלות גדול מ-0? מה הסיכוי שהוא גדול מ-1?

חלק ב:

חוות שרתים של Nvidia.

כעת אתם אחראים על מערכת חוות שרתים של Nvidia.

דרישות ל-GPUs מגיעות בקצב דרישות משתנה במשך השבוע. המרכז חישובים שלהם יכול לספק 15 דרישות בשעה בממוצע. קצב ההגעה הוא לפי יום ושעה בשבוע. מאחר ויש 24 שעות ושבעה ימים סה"כ יכולים להיות לנו 168 קצבי הגעה שונים.

ברצוננו לנתח את מספר הדרישות ל-GPUs הצפוי בחוות שרתים כפונקציה של הזמן. ברשותנו יומן אירועים של דרישות המגיעים לחוות שרתים. הנתונים נאספו לאורך 1000 ימים.

הנתונים מאורגנים כמילון של 1000 ימים שונים, הממוספרים מ-0 עד 999. כל ערך במילון הוא יומן אירועים של יום שונה.

עמודות הנתונים הן:

1. Demand_id – מזהה של הדרישה
2. Time-stamp – הזמן שבו מתרחש האירוע. יחידות הזמן הן בשעות. השעה ה-i היא מספר השעות שחלפו מאז 00:00 של אותו יום.
3. Type – סוג האירוע: ביקוש (Demand) או עזיבה (Release)
4. day – ערך מספרי של היום: {0: יום ראשון, 1: יום שני, 2: יום שלישי, 3: יום רביעי, 4: יום חמישי, 5: יום שישי, 6: שבת}
5. hour – השעה ביום, מספר שלם בין 0 ל-23. אם הערך הוא 0, הכוונה לשעה שבין חצות ל-01:00.
6. day_name – שם היום בשבוע.

כל יום מתחיל עם 0 דרישות בשעה 00:00.

מטרת המשימה היא לאמוד את קצב הדרישות עבור כל שעה ביום.

מתוך הנתונים ניתן להשתמש בשיטות סטטיסטיות קלאסיות על מנת לאמוד את קצב ההגעה עבור כל שעה בשבוע. הבעיה היא שקיימים נתונים חסרים, אין לנו יומן אירועים עבור כל השעות.

הרעיון המרכזי הוא לאמוד את קצב הדרישה עבור אזורי זמן שבהם קיימים נתונים, ולאחר מכן להשתמש ברגרסיית תהליך גאוזי (GP) עבור אזורי הזמן שבהם הנתונים חסרים.

חלק זה מחולק לשלושה חלקים.

חלק 1:

משימות:

1. בצעו ניתוח בייסיאני למספר הדרישות הממוצע המגיע ביום שבת בין השעות 00:00–01:00.
2. באיזו פונקציית הסתברות (Likelihood) השתמשתם ומדוע?
3. בצעו ניתוח ניבוי פוסטריורי (Posterior Predictive Analysis) כדי לוודא שהתפלגות ה-Likelihood שנבחרה אכן מתאימה.
4. על פי תוצאות הניתוח, עד כמה אתם בטוחים בהערכת ממוצע קצב הדרישות?
5. מהו קצב הדרישות המוערך? הציגו ערך יחיד המבוסס על ההתפלגות הפוסטריורית.

חלק 2:

יש לנו 24 שעות ביום ו-7 ימים בשבוע. בסך הכול מתקבל גריד של 168 ערכים.

ברצוננו למפות את קצב ההגעה הממוצע בכל תא זמן.

במאגר הנתונים קיימים חלקים חסרים. חלק מהתאים בגריד אינם כוללים נתונים. תפקידכם הוא להשלים את הגריד.

1. חלצו את קצב הדרישה לכל תא שבו קיימים נתונים (חישוב ממוצע פשוט מספיק, אין צורך בניתוח בייסיאני נוסף).
2. השתמשו ברגרסיית תהליך גאוזי (GP) על מנת להשלים את קצב ההגעה הממוצע עבור יתר התאים בגריד.
3. הציגו גרף תלת-ממדי הכולל:
 - a. גרף פיזור (scatter plot) של הנתונים שאומדו ישירות מהdataframe.
 - b. גרף פיזור (scatter plot) של הנתונים שאומדו באמצעות רגרסיית GP.
 - c. גרף משטח של רגרסיית ה-GP. כלומר surface של GP regression.
4. מהי ההסתברות שקצב ההגעה בין השעות 10:00–11:00, ביום שבת, יהיה בין 9 ל-10 דרישות לשעה?
5. אנא הציגו את המפרט המלא של ההתפלגות הגאוזיית של קצב ההגעה ביום שבת בין השעות 10:00–11:00 וביום שני בין השעות 11:00–12:00.

6. מהו המתאם (קורלציה) בין קצב ההגעה ביום ראשון בין השעות 07:00–08:00 לבין קצב ההגעה ביום ראשון בין השעות 08:00–09:00?
מהו המתאם בין קצב ההגעה ביום ראשון בין השעות 07:00–08:00 לבין קצב ההגעה ביום ראשון בין השעות 19:00–20:00?
איזה ערך מתאם גבוה יותר? האם תוצאה זו הגיונית?

חישוב תוצאות של החלק התחרותי

בחלק זה נשתמש בתחזית שעשינו בחלק הקודם.

כעת ברצוננו להשתמש בהם לצורך חיזוי מספר הדרישות הפעילות הצפוי במערכת כפונקציה של הזמן.

כל שעליכם לעשות הוא ליצור DataFrame עם תוצאות בפורמט הבא.

העלו את `n_true` ואת `df_res`.

`n_true` הוא מספר הדרישות הצפוי האמיתי במערכת.

`df_res` הוא DataFrame הכולל שלוש עמודות: `hour`, `day` ו-`rate`. נכון לעכשיו עמודת `rate` מכילה אפסים; עליכם למלא בה את קצבי ההגעה שחזיתם באמצעות רגרסיית GP, ולאחר מכן להריץ את הפונקציה `plot_results`.

הפונקציה תמחיש את מספר הדרישות הצפוי שהערכתם כפונקציה של הזמן, בהשוואה לערכים האמיתיים (מתוך `n_true`).

בנוסף, הפונקציה תדפיס את ערך ה-MSE, והקבוצה עם ערך ה-MSE הנמוך ביותר תנצח.

נקודות הבונס הן:

מקום ראשון: 3 נקודות בונס.

מקום שני: 2 נקודות בונס.

מקום שלישי: נקודת בונס אחת.

שהקבוצה הטובה ביותר תנצח!!

בהצלחה!