**ניתוח אלגוריתם Gaussian Process Regression עם Ridge ו-Nystroem Kernel Approximation**

**מבוא**

בפרויקט זה נעשה שימוש ברגרסיה גאוסיאנית משולבת עם Ridge Regression, תוך שימוש בשיטת Nystroem Kernel Approximation. המטרה הייתה לבצע חיזוי מחירי טיסות בהתבסס על מאפיינים שונים ולבחון אילו פרמטרים מניבים את התוצאות הטובות ביותר.

**למה לא Gaussian Process רגיל?**

רגרסיה גאוסיאנית רגילה היא אלגוריתם רב עוצמה, אך כאשר כמות הנתונים גדלה, מורכבות החישוב גדלה לרמת O(n3)O(n^3), מה שהופך את השיטה ללא ישימה עבור מאגרי נתונים גדולים כמו זה שלנו (500,000 דגימות). כדי להתגבר על בעיה זו, השתמשנו ב-**Nystroem Kernel Approximation**, שמאפשר גישה חישובית יעילה יותר באמצעות דגימת תת-קבוצה של הנקודות כדי לבצע קירוב לפונקציית הקרנל.

**רגרסיית Ridge**

רגרסיית Ridge היא הרחבה של רגרסיה לינארית רגילה עם רגולריזציה L2L2. היא מוסיפה עונש על גודל המשקלים במודל כדי למנוע אוברפיטינג. בפרויקט השתמשנו ב-α=0.05\alpha=0.05, שהוא פרמטר הרגולריזציה שנבחר לאחר Grid Search.

**שימוש ב-Nystroem Kernel Approximation**

כדי להתמודד עם החישובים הכבדים של קרנלים, השתמשנו בשיטת Nystroem, שמאפשרת לקרב את מטריצת הקרנל ע"י דגימה של מספר מופחת של רכיבים.

**כיצד מתקבלת המשוואה של המודל ומה קורה לנתונים בפועל?**

כאשר אנחנו משתמשים ברגרסיית Ridge עם Nystroem Kernel Approximation, אנחנו בעצם יוצרים טרנספורמציה לא-לינארית של הנתונים, ומריצים עליה מודל לינארי. בואו נראה מה קורה מתמטית לכל שלב.

**פרמטרים שנבדקו:**

* **n\_components:** מספר הרכיבים בהם נעשה שימוש לקירוב הקרנל. לאחר בדיקות, נמצא כי ncomponents=7000n\_{components}=7000 מניב ביצועים מיטביים.
* **קרנלים שנבדקו:**
  + **RBF (Radial Basis Function)** – קרנל נפוץ המתאים לבעיות לא לינאריות.
  + **Polynomial** – קרנל פולינומי המתאים ליחסים מסובכים יותר בין הנתונים.
  + **Sigmoid** – קרנל המדמה רשתות נוירונים אך נמצא כפחות מתאים.
  + **Laplacian** – קרנל דומה ל-RBF אך נותן משקל חזק יותר לערכים קרובים מאוד, ונמצא כמניב תוצאות טובות ביותר.

בסופו של דבר, הקרנל **Laplacian** נמצא כמתאים ביותר, ולכן נבחר.

**הערכת Overfitting / Underfitting**

* התוצאות מראות כי המודל מספק ביצועים טובים הן על סט האימון והן על סט הבדיקה, ללא פער משמעותי, מה שמצביע על כך שאין אוברפיטינג משמעותי.
* הקרנל **Laplacian** והרגולריזציה ב-Ridge עוזרים להימנע מאוברפיטינג ע"י שליטה במורכבות המודל.
* אם היה אוברפיטינג, ניתן היה להפחית את **n\_components** או להגדיל את **alpha** ברגרסיית Ridge כדי להפחית מורכבות המודל.

**מסקנות**

באמצעות שילוב של Nystroem Kernel Approximation ורגרסיית Ridge הצלחנו להתמודד עם גודל הנתונים הגדול ולשמור על ביצועים טובים. הבחירה בקרנל Laplacian נמצאה כמתאימה ביותר, והרגולריזציה α=0.05\alpha=0.05 ברגרסיית Ridge נתנה תוצאות מאוזנות ומדויקות.

ניתן לשפר את המודל בעתיד על ידי ניסוי קרנלים נוספים, שינוי מספר הרכיבים ב-Nystroem, או שימוש בשיטות Feature Selection מתקדמות יותר.

**איך האלגוריתם שלנו עובד ואיך זה קשור לרגרסיה גאוסיאנית?**

אנחנו משתמשים **ברגרסיית Ridge** יחד עם **Nystroem Kernel Approximation** כדי לחזות מחירי טיסות. החלק הקריטי כאן הוא השימוש בקרנלים, ובעיקר בקשר שלהם לרגרסיה גאוסיאנית (Gaussian Regression).

**🚀 איך זה עובד בפשטות?**

1️⃣ **המודל הבסיסי** – רגרסיה רגילה מנסה למצוא משוואה מהצורה:

מחיר=משקל1×משתנה1+משקל2×משתנה2+...+הטיהמחיר = משקל\_1 \times משתנה\_1 + משקל\_2 \times משתנה\_2 + ... + הטיה

אבל זה עובד טוב רק אם הקשרים בין המשתנים הם **לינאריים** – כלומר, שכל שינוי קטן במחיר תלוי באופן ישיר ומשתנה בקצב קבוע לפי גורמים כמו יום יציאה, זמן הזמנה וכו'.

2️⃣ **הבעיה** – הקשרים בעולם האמיתי הם **לא לינאריים**. מחירי טיסות לא משתנים תמיד בקצב קבוע – למשל, יום מסוים יכול להיות זול מאוד אבל יום אחר, קרוב אליו, יקר בטירוף.

3️⃣ **הפתרון - קרנלים** 🧠  
כאן נכנסת הגישה של **קרנלים גאוסיאניים**: במקום לנסות לחפש קשר ישיר, אנחנו מסתכלים על **קשרים מרחביים** בין הנתונים.

**🎯 איך זה קשור לרגרסיה גאוסיאנית?**

**מה זה רגרסיה גאוסיאנית?**

במודלים מבוססי **Gaussian Processes**, במקום למצוא משוואה אחת מדויקת, אנחנו בעצם מניחים שהנתונים שלנו **מתנהגים כמו התפלגות גאוסיאנית** (כלומר, כמו עקומת פעמון) ומנסים להבין **איך כל נקודת מחיר מושפעת משכנותיה**.

**איך זה בא לידי ביטוי אצלנו?**

🔹 השתמשנו ב**Laplacian Kernel**, שהוא דומה לקרנל גאוסיאני – הוא מאפשר לנו **למצוא קשרים בין נתונים בצורה חכמה יותר**, גם אם הם לא נראים ישרים במבט ראשון.  
🔹 **Nystroem Kernel Approximation** הוא טריק מתמטי שעוזר לנו להשתמש בגישה הזאת **בצורה הרבה יותר מהירה** (כי רגרסיה גאוסיאנית מלאה היא מאוד כבדה חישובית).  
🔹 רגרסיית **Ridge** מאפשרת לנו **לאזן בין דיוק לבין הימנעות מ-Overfitting** (ללמוד יותר מדי פרטים אקראיים בנתונים).

**🔎 אז איך זה משפיע על החיזוי שלנו?**

✅ במקום מודל שמנסה לקבוע משוואה קשיחה, אנחנו נותנים למודל גמישות **למצוא קשרים מורכבים במחירי הטיסות**.  
✅ שימוש בקרנלים מאפשר למודל **ללמוד דפוסים לא ישירים** – למשל, שהבדל של יום אחד בטיסה יכול להשפיע על המחיר בצורה שונה לגמרי במקרים שונים.  
✅ בסופו של דבר, קיבלנו **דיוק גבוה מאוד**, בגלל שהצלחנו לנצל את הכוח של גישות גאוסיאניות בלי לשלם את המחיר הכבד של חישובי Gaussian Process מלאים.

**📌 סיכום - למה זה עובד טוב?**

💡 אנחנו **לא מנסים למצוא משוואה לינארית פשוטה**, אלא משתמשים בקרנלים שמכניסים חוכמה גאוסיאנית למודל.  
💡 **Nystroem** מאפשר לנו לעשות את זה **מהר ויעיל**.  
💡 **Ridge Regression** שומר על המודל מפני למידה עודפת ונותן איזון טוב בין **גמישות לדיוק**.

🔹 **בסופו של דבר, המודל שלנו לא נותן "משוואה אחת קבועה", אלא חיזוי שמתחשב במורכבות של המחירים בעולם האמיתי!** 🔹