מבוא למערכות לומדות- תרגיל בית 3- דו״ח עבודה

מגישות:

[redacted]

[redacted]

Section 1: Linear regression implementation

(Q1)

הפונקציה $L_\delta(w,b,x_i,y_i)=\begin{cases}\frac{1}{2}(w^Tx_i+b-y_i)^2, & |w^Tx_i+b-y_i|\le\delta\\ \delta\left(|w^Tx_i+b-y_i|-\frac{1}{2}\delta\right) & ,\text{else}\end{cases}$ היא פונקציית מקרים.

נתייחס לנגזרת שלה לפי b לפי המקרים:

- עבור המקרה $|w^Tx_i+b-y_i|<\delta$, הטעות היא ריבועית ביחס לb והפונקציה היא גזירה ביחס לb. אין נקודות אי גזירות בתחום זה ולכן הגדרת הנגזרת והsubderivative מתלכדות.
- עבור המקרה $|w^Tx_i+b-y_i|>\delta$, הטעות היא מוגדרת באמצעות הזזה והכפלה בסקלר של פונקציית הערך המוחלט- $|w^Tx_i+b-y_i|$. פונקציה זו היא גזירה ביחס לb בכל מקום מלבד באופן פוטנציאלי בנקודה שבה$w^Tx_i+b-y_i=0$. עם זאת, מאחר שאנו בתחום שבו $|w^Tx_i+b-y_i|>\delta$, הפונקציה גזירה באזורים אלו גם כן, ולכן הגדרת הנגזרת והsubderivative מתלכדות.
- עבור המקרים בהם $|w^Tx_i+b-y_i|=\delta$, כלומר כאשר
- $b_1=\delta+y_i-w^Tx_i, b_2=-\delta+y_i-w^Tx_i$ הפונקציה עדיין גזירה ביחס לb בנקודות אלו כיוון שהנגזרות מימין מתלכדות עם הנגזרות משמאל.

סה״כ מאחר והפונקציה גזירה ביחס לb על כל התחום נוכל לחשב את הנגזרות החלקיות בנפרד:

$$\frac{\partial\ell_\delta(w,b;x_i,y_i)}{\partial b}=\begin{cases}\dfrac{\partial\left(\frac{1}{2}(w^Tx_i+b-y_i)^2\right)}{\partial b}, & |w^Tx_i+b-y_i|\le\delta\\ \dfrac{\partial\left(\delta\left(|w^Tx_i+b-y_i|-\frac{1}{2}\delta\right)\right)}{\partial b}, & \text{otherwise}\end{cases}$$

נחשב כל חלק בנפרד:

1. $|w^Tx_i+b-y_i|\le\delta$:

$$\frac{\partial\left(\frac{1}{2}(w^Tx_i+b-y_i)^2\right)}{\partial b}=\frac{1}{2}\cdot 2\cdot(w^Tx_i+b-y_i)\cdot\frac{\partial(w^Tx_i+b-y_i)}{\partial b}=(w^Tx_i+b-y_i)\cdot 1$$

$$\Rightarrow\frac{\partial\left(\frac{1}{2}(w^Tx_i+b-y_i)^2\right)}{\partial b}=(w^Tx_i+b-y_i)$$

2. Otherwise:

$$\frac{\partial\left(\delta\left(|w^Tx_i+b-y_i|-\frac{1}{2}\delta\right)\right)}{\partial b}=\frac{\partial\left(\delta|w^Tx_i+b-y_i|-\frac{1}{2}\delta^2\right)}{\partial b}$$

$$=\begin{cases}\dfrac{\partial\left(-\delta(w^Tx_i+b-y_i)-\frac{1}{2}\delta^2\right)}{\partial b}, & (w^Tx_i+b-y_i)<0\\ \dfrac{\partial\left(\delta(w^Tx_i+b-y_i)-\frac{1}{2}\delta^2\right)}{\partial b}, & (w^Tx_i+b-y_i)\ge 0\end{cases}$$

$$= \begin{cases} -\delta, & (w^T x_i + b - y_i) < 0 \\ \delta, & (w^T x_i + b - y_i) \geq 0 \end{cases} = \delta \cdot \mathrm{sign}(w^T x_i + b - y_i)$$

נקבל :

$$\frac{\partial \ell_\delta(w, b; x_i, y_i)}{\partial b} = \begin{cases} (w^T x_i + b - y_i) & , \quad |w^T x_i + b - y_i| \leq \delta \\ \delta \cdot \mathrm{sign}(w^T x_i + b - y_i) & , \quad \text{otherwise} \end{cases}$$

(Q2)

נניח שיש לנו m זוגות של דוגמאות ותיוגים, כאשר כל דוגמה ממימד d כלומר :

$$\{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m) |\ x_i \in R^d, y_i \in R \text{ for } i \in [m]\}$$

נסמן את האלמנטים של דוגמה מסוימת, x_i באופן הבא: $x_i = \left(x^{(1)}, x^{(2)} \dots, x^{(d)}\right)^T$

אזי נגדיר את המטריצה $X = \begin{pmatrix} x_1^{(1)} & \cdots & x_1^{(d)} \\ \vdots & \ddots & \vdots \\ x_m^{(1)} & \cdots & x_m^{(d)} \end{pmatrix}$ ונגדיר את וקטור העמודה $y = (y_1, y_2, \dots, y_m)^T$.

עבור וקטור $z = \left(z^{(1)}, z^{(2)} \dots, z^{(m)}\right)^T \in R^m$ נגדיר פונקציית אינדיקטור $\mathbb{I}_{\leq\delta}: R^m \to R^m$ באופן הבא

$\mathbb{I}_{\leq\delta}(z)_i = \begin{cases} 1, & |z^{(i)}| \leq \delta. \\ 0, & |z^{(i)}| > \delta. \end{cases}$.

עבור וקטור $z = \left(z^{(1)}, z^{(2)} \dots, z^{(m)}\right)^T \in R^m$ נגדיר פונקציית אינדיקטור $\mathbb{I}_{>\delta}: R^m \to R^m$ באופן הבא

$\mathbb{I}_{>\delta}(z)_i = \begin{cases} 0, & |z^{(i)}| \leq \delta. \\ 1, & |z^{(i)}| > \delta. \end{cases}$, כאשר $z^{(i)}$ הוא האלמנט במיקום הi בוקטור z .

נשים לב כי $1_m - \mathbb{I}_{\leq\delta}(z) = \mathbb{I}_{>\delta}(z)_i$

בנוסף, עבור וקטור $z = \left(z^{(1)}, z^{(2)} \dots, z^{(m)}\right)^T \in R^m$ נגדיר פונקציית אינדיקטור $\mathrm{sign}: R^m \to R^m$ באופן הבא

$\mathrm{sign}(z)_i = \begin{cases} -1, & |z^{(i)}| < 0 \\ 1, & |z^{(i)}| > 0: \\ 0, & |z^{(i)}| = 0 \end{cases}$, כאשר $z^{(i)}$ הוא האלמנט במיקום הi בוקטור z .

אזי נוכל לבטא את $\nabla_w \mathcal{L}_H(w, b)$ באופן הבא:

$$\nabla_w \mathcal{L}_H(w, b) = \nabla_w \left(\frac{1}{m} \sum_{i=1}^{m} \ell_\delta(w, b; x_i, y_i)\right) = \frac{1}{m} \left(\sum_{i=1}^{m} \nabla \ell_\delta(w, b; x_i, y_i)\right)$$

$$= \frac{1}{m} \left(\sum_{i \text{ s.t } |w^T x_i + b - y_i| \leq \delta} (w^T x_i + b - y_i) x_i + \sum_{i \text{ s.t } |w^T x_i + b - y_i| > \delta} \delta \cdot \mathrm{sign}(w^T x_i + b - y_i) x_i \right)$$

$$= \frac{1}{m} X^T((Xw + b1_m - y) \odot \mathbb{I}_{\le\delta}(Xw + b1_m - y)) +$$

$$\frac{1}{m}\delta \cdot X^T(\mathrm{sign}(Xw + b1_m - y) \odot \mathbb{I}_{>\delta}(Xw + b1_m - y)) =$$

$$= \frac{1}{m} X^T((Xw + b1_m - y) \odot \mathbb{I}_{\le\delta}(Xw + b1_m - y)) +$$

$$\frac{1}{m}\delta \cdot X^T(\mathrm{sign}(Xw + b1_m - y) \odot (1_m - \mathbb{I}_{\le\delta}(Xw + b1_m - y)))$$

*כאשר הסימן $\odot$ הוא element-wise.

נבדוק שהנוסחה חוקית מבחינת גדלים:)

- $X_{mxd} * w_{dx1} = v_{mx1}$ היא מכפלה חוקית ונותנת לנו וקטור עמודה בעל m מימדים, לכן גם הסכום חוקי $(X_{mxd} *$ $w_{dx1} + b1_m - y)$ כיוון שכל הוקטורים שייכים לR^m.
- הקלט ל $\mathbb{I}_{\le\delta}$ הוא $z \in R^m$ ואכן $(X_{mxd} * w_{dx1} + b1_m - y) \in R^m$ והפלט הוא גם כן וקטור $u \in R^m$
- מכפלת הדמרד (איבר – איבר) בין $(X_{mxd} * w_{dx1} + b1_m - y) \in R^m$ ובין $\mathbb{I}_{\le\delta}(Xw + b1_m - y) \in R^m$ ולידית כי שניהם מאותו מימד והפלט הוא גם כן בR^m
- המטריצה $X^T \in R^{d*m}$ ולכן חוקי להכפיל בין המטריצה לווקטור $(Xw + b1_m - y) \odot \mathbb{I}_{\le\delta}(Xw + b1_m - y) \in R^m$ ונקבל פלט של וקטור $v \in R^d$ בגודל המצופה לוקטור גרדיאנט כיוון שאנו גוזרים לפי $w \in R^d$.
- סה"כ הביטוי $\frac{1}{m}X^T((Xw + b1_m - y) \odot \mathbb{I}_{\le\delta}(Xw + b1_m - y))$ מחזיר לנו $v \in R^d$ כפי שאנחנו רוצים.
- הביטוי $(1_m - \mathbb{I}_{\le\delta}(Xw + b1_m - y))$ הוא ולידי כיוון שראינו כי $\mathbb{I}_{\le\delta}(Xw + b1_m - y) \in R^m$ ולכן חישוב ההפרש בין 2 בוקטורים הוא לידי ומתקבל וקטור ששייך ל R^m.
- לפי איך שהגדרנו את sign לוקטורים ממימד m, אזי הביטוי $\mathrm{sign}(Xw + b1_m - y)$ הוא ולידי ומחזיר וקטור ששייך ל R^m.
- $(\mathrm{sign}(Xw + b1_m - y) \odot (1_m - \mathbb{I}_{\le\delta}(Xw + b1_m - y))$ ולידי כיוון שאנו מבצעים מכפלה איבר-איבר בין שני וקטורים ממימד m.

  כמו קודם, גם המכפלה $X^T(\mathrm{sign}(Xw + b1_m - y) \odot (1_m - \mathbb{I}_{\le\delta}(Xw + b1_m - y)))$ חוקית כיוון שכופלים $X^T \in R^{d*m}$ בוקטור מימד m. נקבל פלט של וקטור $v \in R^d$ בגודל המצופה לוקטור גרדיאנט כיוון שאנו גוזרים לפי $w \in R^d$.
- סה"כ נקבל מהסכום של 2 הביטויים, 2 וקטורים ממימד d, לכן פעולת החיבור הזו היא אכן ולידית ואנחנו מקבלים וקטור גרדיאנט ממימד d כמצופה.)

כעת נפתח נוסחה עבור

$$\frac{\partial L_H(w,b)}{\partial b} = \frac{\partial}{\partial b}\left(\frac{1}{m}\sum_{i=1}^{m}\ell_\delta(w,b;x_i,y_i)\right) = \frac{1}{m}\left(\sum_{i=1}^{m}\frac{\partial}{\partial b}(w,b;x_i,y_i)\right)$$

$$= \frac{1}{m}\left(\sum_{i\ s.t\ |w^Tx_i+b-y_i|\le\delta}(w^Tx_i + b - y_i) + \sum_{i\ s.t\ |w^Tx_i+b-y_i|>\delta}\delta \cdot \mathrm{sign}(w^Tx_i + b - y_i)\right) =$$

$$\frac{1}{m}(Xw + b1_m - y)^T\mathbb{I}_{\le\delta}(Xw + b1_m - y) +$$

$$\frac{1}{m}\delta \cdot \left(\mathrm{sign}(Xw + b1_m - y)\right)^T(1_m - \mathbb{I}_{\le\delta}(Xw + b1_m - y))$$

(נבדוק שהנוסחה חוקית מבחינת גדלים:

- $X_{mxd} * w_{dx1} = v_{mx1}$ היא מכפלה חוקית ונותנת לנו וקטור עמודה בעל m מימדים, לכן גם הסכום ($X_{mxd} * w_{dx1} + b1_m - y$) כיוון שכל הוקטורים שייכים לR^m.
- הקלט ל $\mathbb{I}_{\le\delta}$ הוא $z \in R^m$ ואכן $(X_{mxd} * w_{dx1} + b1_m - y) \in R^m$ והפלט הוא גם כן וקטור $u \in R^m$
- $(Xw + b1_m - y)^T \mathbb{I}_{\le\delta}(Xw + b1_m - y)$ ולידי מאחר ומדובר במכפלה פנימית בין 2 וקטורים מימד m ובסופו של דבר הפלט מהמכפלה הזו הוא מספר כפי שהיינו מצפים.
- גם הביטוי $\frac{1}{m}(Xw + b1_m - y)^T \mathbb{I}_{\le\delta}(Xw + b1_m - y)$ הוא מספר , כמצופה.
- הביטוי $(1_m - \mathbb{I}_{\le\delta}(Xw + b1_m - y))$ הוא ולידי כיוון שראינו כי $\mathbb{I}_{\le\delta}(Xw + b1_m - y) \in R^m$ ולכן חישוב ההפרש בין 2 בוקטורים הוא לידי ומתקבל וקטור ששייך ל R^m.
- לפי איך שהגדרנו את sign לוקטורים ממימד m, אזי הביטוי $\mathrm{sign}(Xw + b1_m - y)$ הוא ולידי ומחזיר וקטור ששייך ל R^m.
- $\left(\left(\mathrm{sign}(Xw + b1_m - y)\right)^T(1_m - \mathbb{I}_{\le\delta}(Xw + b1_m - y)\right)$ ולידי מאחר ומדובר במכפלה פנימית בין 2 וקטורים מימד m ובסופו של דבר הפלט מהמכפלה הזו הוא מספר כפי שהיינו מצפים.
- סה״כ הביטוי $\frac{1}{m}\delta \cdot \left(\mathrm{sign}(Xw + b1_m - y)\right)^T(1_m - \mathbb{I}_{\le\delta}(Xw + b1_m - y)$ מחזיר מספר ממשי מאחר ומדובר במכפלה פנימית.
- לכן הסכום בין שני הביטויים של הנגזרות לפי b חוקי ומחזיר לנו מספר ממשי , כמצופה.)

(Q3)

נשים לב כי הפרמטר דלתא בפונקציית הhuber loss הוא גודל המרחק (בערך המוחלט) בין החיזוי שלנו לתיוג האמיתי כלומר הresiduals. הפרמטר דלתא קובע איך פונקציית הloss תתנהג- כאשר הresidual קטן מספיק , הפונקציה מתנהגת כמו שגיאת MSE רגילה. עבור ערכים גדולים יותר, פונקציית הloss מתנהגת כמו שגיאת הערך המוחלט . מכאן שדלתא הוא סף כלשהו שמגדיר לנו האם בסבירות גבוהה מדידה מסויימת היא חריגה (outlier) ובהתאם נרצה לתת לה משקל נמוך יותר בחישוב הloss .

לצורך הגדרת דלתא נרצה לבצע הערכה אילו נקודות עשויות להיחשב "outliers". בהינתן שהמידע היחיד שיש לנו על הבעיה הן הדגימות בדאטה סט , נרצה לקבל הערכה מהדאטהסט על מה גודל הresiduals האפשריים שיכולים להיות לנו. לצורך כך נרצה לאמן רגרסור לינארי (עם שגיאה ריבועית רגילה) ולהבין אילו ערכי residuals הם יחסית חריגים ביחס לדגימות האחרות.

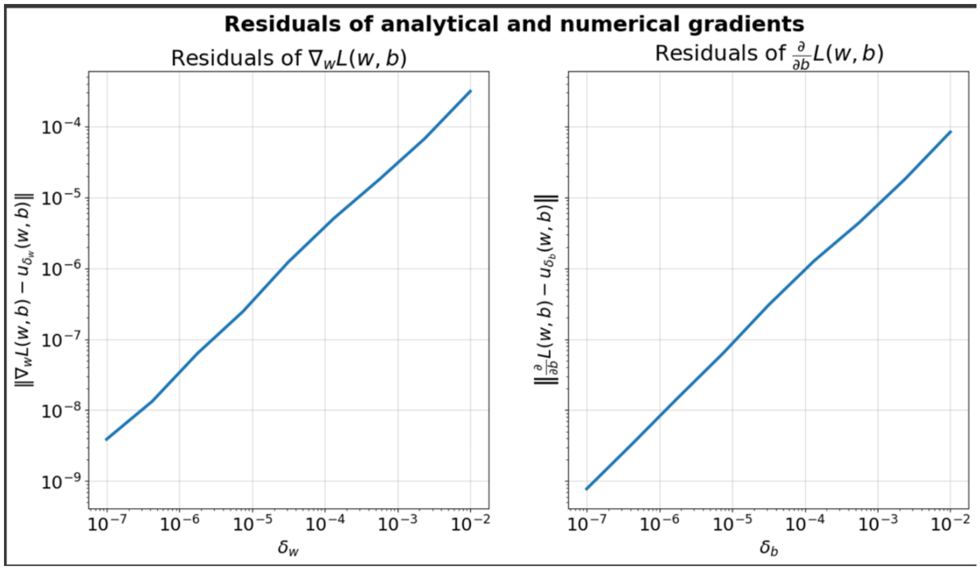
אנחנו בחרנו להשתמש במדד הטווח הבין רבעוני (IQR) שראינו בתרגול 1 כדי לקבוע מהו outlier. ע״פ המקובל ערכים קיצוניים הן תצפיות שנופלות מתחת ל-Q1-1.5(IQR) או מעל Q3+1.5(IQR). מכיוון שבמקרה זה אנו מסתכלים על הערך המוחלט של הresiduals, מספיק להסתכל רק על Q3+1.5(IQR) כדי לקבוע את דלתא.

נציין שכמובן שהבחירה שלנו במדד הIQR לצורך קביעת outliers אינה יחידה וישנן אפשרויות נוספות

<u>פסודו קוד:</u>

1. אמן את הדאטה עם רגרסור לינארי ופונקציית loss של MSE
2. חשב את הresiduals בערך מוחלט
3. א. חשב את האחוזון ה25 של הדאטה (Q1)
   ב. חשב את האחוזון של ה75 של הדאטה (Q3)
   ג. חשב את הטווח הבין רבעוני (IQR) : IQR=Q3-Q1
4. הגדר $\delta = Q3 + 1.5 * QR$
5. החזר את δ

(Q4)



(Q5)

Training and Validation Losses Across Different Learning Rates for Huber Regression with huber_delta=3.67

<u>השתמשנו בhuber_delta=3.67 (חושב רק על החלק של הtrain מתוך הtrain-validation) ע״פ האלגוריתם שייצרנו בסעיף 3.</u>

ניתן להצדיק שרואים בגרפים כך:

*עבור קצבי למידה נמוכים (1e-09-1e-5), ניתן לראות שבקושי יש ירידה בloss הן בtrain והן בvalidation מאחר וככל הנראה גודל הצעד בgradient קטן מדי ולכן ההתכנסות איטית מאוד ולא קורית ב1500 צעדים.

* כאשר קצב הלמידה עולה לlr=0.0001 אנו רואים כי כן יש ירידה קטנה בloss, הן בvalidation והן בtrain, אך עדיין הירידה הזו קטנה ביחס לloss שמגיעים אליו בקצבי למידה גבוהים יותר. עבור קצב למידה הזה, אנו אכן מתקרבים לכיוון המינימום אך כמות הצעדים לא מספיקה עדיין להתכנסות.

* עבור קצבי הלמידה 0.001,0.01 אנו רואים כבר שחלה ירידה משמעותית בlossב במהלך האימון והם מגיעים לערכי loss יחסית דומים בסוף האימון עבור שני הקצבים. עם זאת, הירידה עבור הקצב 0.001 היא מתונה יותר, בהשוואה ל0.01 , מאחר שגודל הצעד קטן יותר ולכן אנחנו מתקדמים לעבר המינימום בצורה איטית יותר.

* עבור קצב הלמידה 0.1 , ההתכנסות היא מהירה מאד אבל יש הרבה יותר תנודות בערך הlossה בתהליך האימון וגם בקבוצת ולידיה. התנהגות זו מצביעה על כך שקצב למידה זה גדול מדי ולכן האלגוריתם מפספס מעט את המינימום בכל איטרציה.

* עוד נשים לב, שערך הlossה של קבוצת האימון ושל קבוצת הוולידציה דומה במהלך כל תהליך האימון של המסווג, ושווקטור w כלשהו שאנו מוצאים במהלך תהליך האימון שמקטין את הlossה על הtrainה, מקטין את הlossה על הvalidationה. התנהגות זו היא התנהגות טובה, שמראה לנו שהמסווג שלנו אכן לומד במהלך תהליך האימון ויודע לבצע הכללה על דאטה שלא ראה (validation). לא נראה שהגענו עדיין לנק׳ overfit במהלך תהליך האימון כיוון שלא ראינו אינדיקציה לכך שהtrainה ממשיך לרדת אבל הlossה של הvalidationה כבר מתחיל לעלות. כמו כן, ברור שעבור קצב הלמידה 0.01 הייתה התכנסות מהירה יחסית למינימום כלשהו לאחר מס׳ נמוך של צעדים, ואימון נוסף בקושי מקטין את הlossה על הtrainה והvalidationה.

לדעתנו , קצב הלמידה האופטימלי הוא lr=0.01 כיוון שהוא מתכנס למינימום, ללא תנודות כמעט ובצורה מהירה יותר מאשר הקצב 0.001. בנוסף, הוא מקטין את הlossה גם על הtrainה וגם על הvalidationה והlossה שקיבלנו על קבוצת הוולידציה היה הנמוך ביותר עבור קבוצה זו.

הגדלה של מספר הצעדים עבור lr זה אינה הכרחית כיון שניתן לראות שהאלגוריתם הגיע כבר לנק׳ מינימום כבר לאחר מס׳ מועט של צעדים, ואימון נוסף כבר בקושי הקטין את הlossה.

(Q6)

הrobustnessה של פונקציית הhuber lossה מנוצלת בצורה טובה היותר כאשר הדאטה מכיל outliers , כאשר הדאטה מגיע מהתפלגות עם זנב ארוך (יש הסתברות נמוכה לקבל ערכים מאוד רחוקים מהערכים שבדר״כ מקבלים) או כאשר ההנחה שלנו שהרעש מתפלג נורמלית נשברת. במקרים כאלו, רגרסיה מסוג OLS (בעיית הריבועים הפחותים הרגילה) עובדת פחות טוב כיוון שהיא רגישה מאוד למדידות חריגות. מהגדרת בעיית הריבועים הפחותים באמצעות שגיאה ריבועית, ערכי מדידות חריגים שיש להם שגיאה גדולה יותר מהשגיאה הממוצעת, משפיעים הרבה יותר על גודל הlossה מאשר המדידות האחרות ובכך גורמים לקו הרגרסיה להתאים את עצמו הרבה יותר לרעש. המאפיינים של פונקציית הhuberה מסייעים לה להתמודד עם בעיית הרעש- פונקציית loss זו מתנהגת כמו שגיאה ריבועית כאשר הresidualsה קטנים ומתנהגת כמו שגיאת הערך המוחלט כאשר הresidualsה גדולים מסף מסויים.

פונקציית הhuberה מאפשרת לנצל את היתרונות של שתי השיטות- השגיאה האבסולוטית אינה גזירה ב0 וקשה לאופטימיזציה, בעוד שפונק׳ הhuberה מתנהגת סביב 0 כמו שגיאה ריבועית ולכן גזירה. עם זאת, עבור ערכי שגיאה גדולים, פונק׳ הhuberה פחות רגישה לרעש כיוון שנותנת פחות חשיבות לרעש ואנומליות (שם השגיאה לינארית).

Section 2: Evaluation and Baseline

(Q7) עבור הדאמי יש לחשב שגיאה ריבועית על האימון

Model	**Section**	**Train Huber Loss** (cross validated) Mean (std)	**Train MSE** (cross validated) Mean (std)	**Valid MSE** (cross validated) Mean (std)
Dummy	2	2.29(+/- 0.1)	5.27 (+/- 0.26)	5.29 (+/- 1.051)

(Q8)

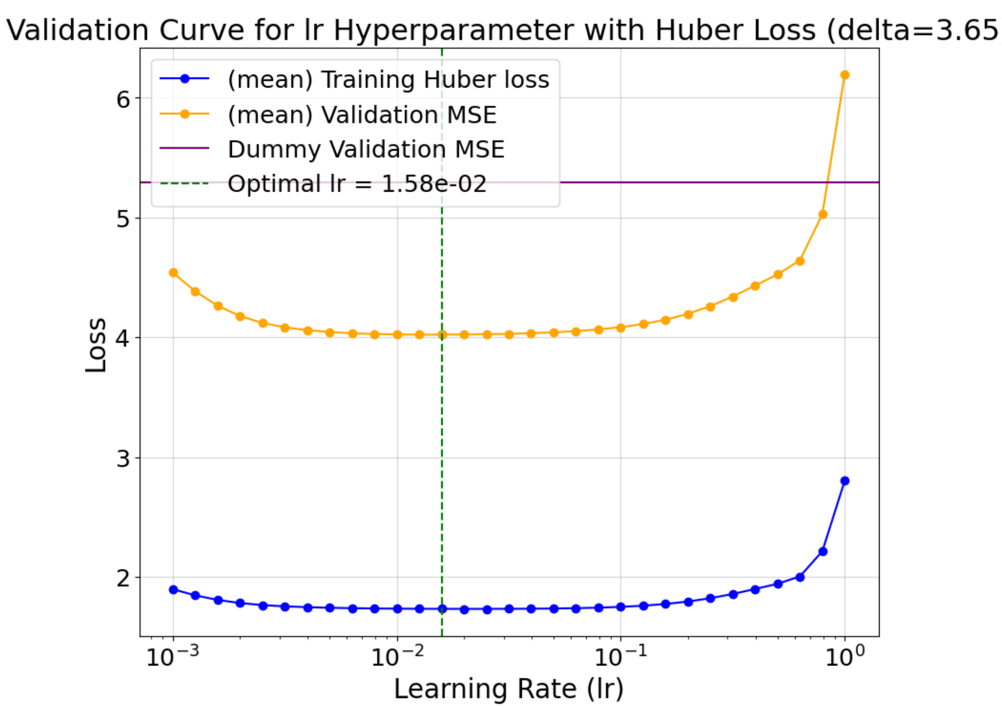
הערכים שבחרנו לבדוק עבור הlearning rate הם 31 ערכים ברווחי log בטווח $10^{-3} - 10^{0}$) כך יצא שההפרש בין חזקה לחזקה הוא 0.1)

בחרנו בטווח זה כיוון שראינו בשאלה 5 שהאלגוריתם שלנו מתכנס עבור learning rates בין אזור $10^{-2}, 10^{-3}$ ושבאזור 10^{-1} יש קצת רעש אך עדיין הייתה התכנסות. לכן החלטנו להרחיב את טווח החיפוש על טווח הlearning rates הזה כדי לאפטם את הlearning rate כך שנקבל שגיאת ולידציה הכי קטנה.

ע״פ ההוראות, אימנו עבור כל learning rate מודל רגרסיה המאומן על huber_loss (השתמשנו בcross validation עם חלוקה ל5 קבוצות כך שלמעשה לכל learning rate אומנו 5 מסווגים).

עבור כל learning rate חישבנו את ממוצע שגיאת האימון וממוצע שגיאת המבחן (ממוצע על חמשת הסגמנטים מה cross validation).

להלן הגרף שקיבלנו:



כפי שניתן לראות, הן שגיאת האימון והן שגיאת המבחן מתנהגות באופן הבא: ככל שהlr עולה השגיאה הולכת ויורדת ולאחר מכן עבור lr גדולים מדי השגיאה מתחילה לעלות בחזרה. ההתנהגות הזו צפויה, כי כפי שראינו קודם, עבור lr גדולים מדי , אלגוריתם הsgd כבר מתחיל להתבדר ולכן לא מקטין את השגיאה. עבור lr קטנים, ההתכנסות של האלגוריתם איטית יותר ולכן גודל הloss שמגיעים אליו במספר קבוע של צעדים הוא עדין גדול לעומת ה שאפשר להגיע עם lr גדול יותר.

נשים לב כי אנו מודדים את השגיאה באימון והשגיאה במבחן באמצעות מטריקות שונות , וההפרש בין שגיאת האימון לשגיאת המבחן נובע בין השאר גם מההבדל בחישוב (עבור הhuber loss , לא מעלים בריבוע שגיאות גדולות מסף דלתא).

ה learning rate האופטימלי שמצאנו באימון הוא:

```
Optimal learning rate: 0.02
Training Huber loss at optimal lr: 1.73(+/- 0.10)
Validation MSE at optimal lr: 4.02(+/- 1.00)
```

Model	**Section**	**Train Huber Loss** (cross validated)	**Train MSE** (cross validated)	**Valid MSE** (cross validated)
Dummy	2	2.29(+/- 0.1)	5.27 (+/- 0.26)	5.29 (+/- 1.051)
Linear	2	1.73(+/- 0.1)		4.02 (+/- 1)

(Q9)

<u>עבור המודל הdummy</u> שחוזה תמיד ערך אחד והוא הy הממוצע, לנרמול אין שום אפקט. זאת מכיוון שמודל זה בכלל לא מתייחס מה הם הפיצ׳רים של דוגמה ספציפית בעת חיזוי, אלא הוא תמיד חוזה את אותו ערך הy הממוצע (ערך יחיד).

<u>עבור המודל הלינארי עם huber-loss –</u>

ברמה התאורטית-

שיטות הנרמול אליהן נחשפנו בקורס- min-max וstandardization הן טרנספורמציות לינאריות:

$$\min\max scaling: x' = \frac{x}{x_{min} - x_{max}} - \frac{x_{min}}{x_{min} - x_{max}} = cx + b \; where \; c = \frac{1}{x_{min} - x_{max}}, b = \frac{x_{min}}{x_{min} - x_{max}}$$

$$standardization: x' = \frac{x}{\mu} - \frac{\sigma}{\mu} = mx + n \; where \; m = \frac{1}{\mu}, n = \frac{\sigma}{\mu}$$

מאחר ושיטות הנרמול שלנו הן טרנספורמציות לינאריות, ניתן להוכיח כי לכל מסווג w המאופיין ע״י המשוואה $y = w^T \bar{x} + b$ ניתן למצוא מסווג אחר w’ שיפעל על הפיצ׳רים המנורמלים x’ ויחזיר את אותה פרדיקציה כלומר $y' =$ $w'^T \overline{x'} + b'$ כך ש$y = y'$.

(ההוכחה מצורפת למטה).

מההוכחה אנו מקבלים שקיים קשר חד-חד ערכי ועל בין הפרמטרים w,b עבור x לפני נרמול לפרמטרים w’,b’ עבור x לאחר נרמול. הסט של ההיפר-מישורים האפשריים לפני נרמול ואחרי נרמול הוא למעשה אותו סט. מכאן, שהערך של ה training loss (הביצועים), ובמיוחד הערך האופטימלי של הtraining loss יישאר אותו הדבר <u>ברמה התאורטית</u> במהלך תהליך אימון (כלומר תמיד נוכל להגיע לאותו ערך של מינימום, לא לאותו arg min).

בתהליך האימון שלנו אנו מחפשים מינימום עבור פונ׳ מטרה קמורה שהיא הhuber-loss. הloss הוא בסופו של דבר פונקציה של הw,b . כפי שראינו, הנרמול לא אמור להשפיע על ערך הloss שאנו יכולים להשיג כיוון שכל ערך loss שניתן להשיג עם w,b שמוצאים לפני נרמול אפשר למצוא w'b' שיתנו לנו את אותו ערך loss ולהפך.

עם זאת, כפי שראינו בהרצאה, ברמה הפרקטית , הביצועים יכולים להיפגע כאשר לא מנרמלים את הדאטה. עבור דאטהסט מנורמל/לא מנורמל נקבל גרדיאנט שונה שכן במקרה של הhuberloss הגרדיאנט תלוי גם בערך של הפיצ׳רים ולא קבוע. עבור פיצ׳רים שהסקלה שלהם גדולה יחסית, נצטרך learning rate נמוך יותר (כיוון שהנגזרת החלקית לפי המשקולת המתאימה לפיצ׳ר תהיה גדולה יותר) מאשר פיצ׳רים שהסקלה שלהם יחסית קטנה . כתוצאה מכך, נצטרך לבחור lr קטן מספיק לפיצ׳ר עם הסקאלה הכי גדולה. יכול להיווצר מצב שנבחר lr נמוך מדי עבור הפיצ׳רים עם הסקלה הקטנה וההתכנסות תהיה הרבה יותר איטית, כלומר נצטרך יותר איטרציות כדי להתכנס .

הוכחה:

נניח שיש לנו דאטהסט $X \in R^{mxd}, y \in R^m, w \in R^d, b \in R$ **ואנו מבצעים טרנספורמציה לינארית לכל אחד מהפיצ׳רים כך ש**$x_i' = a_i x_i + c_i, a_i \neq 0$ **. כלומר עבור כל** $x_{org} \in X = (x_1, x_2, \dots, x_d)$ **אנו מקבלים** $x_{new} = (x_1', x_2', \dots x_d') = (a_1 x_1 + c_1, a_2 x_2 + c_2, \dots, a_d x_d + c_d)$.

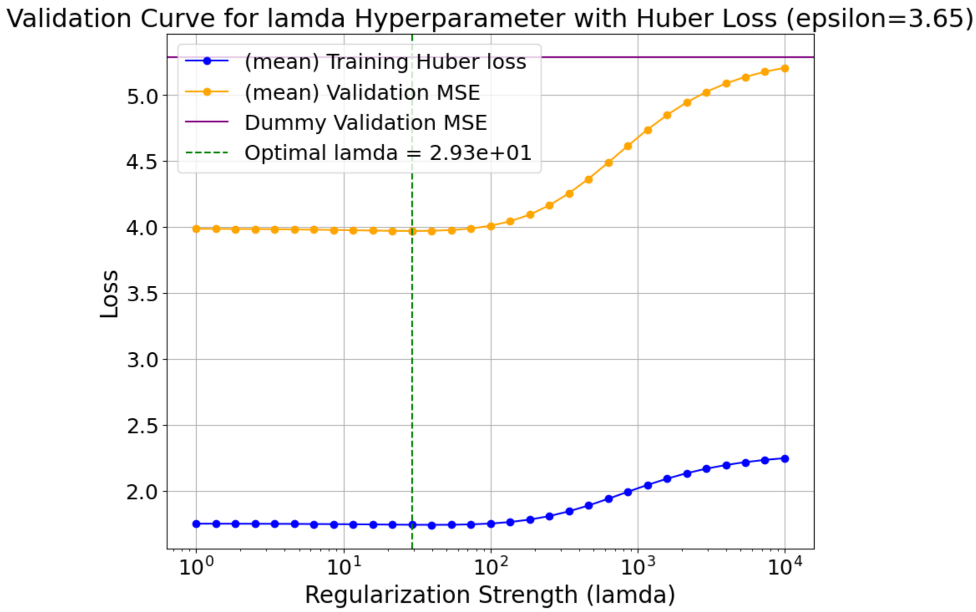
נניח ש$w = (w_1, w_2, \dots, w_d)$ **אזי נגדיר** $w' = (w'_1, w'_2, \dots, w'_d) = \left(\frac{w_1}{a_1}, \frac{w_2}{a_2}, \dots, \frac{w_d}{a_d}\right)$, $b' = b - \sum_{i=1}^{i=d} \frac{w_i a_i}{c_i}$ **ונקבל**:

$$w'^T x_{new} + b' = \sum_{i=1}^{d} w_i' x_i' + b' = \sum_{i=1}^{d} \frac{w_i}{a_i} * (a_i x_i + c_i) + b - \sum_{i=1}^{i=d} \frac{w_i a_i}{c_i} = \sum_{i=1}^{d} w_i x_i + \sum_{i=1}^{d} \frac{w_i a_i}{c_i} + b - \sum_{i=1}^{i=d} \frac{w_i a_i}{c_i}$$
$$= \sum_{i=1}^{d} w_i x_i + b = w^T x_{old} + b$$

Section 3: Ridge linear regression

(Q10)

הערכים שבחרנו לבדוק עבור הregularization parameter הם 31 ערכים ברווחי log בטווח $10^0 - 10^4$ (לאחר שניסינו על טווח גדול יותר בהתחלה). להלן התוצאות:



נשים לב כי עבור ערכי למדא קטנים, אין כמעט השפעה של הרגולריזציה על הביצועים של המודל, כאשר עבור ערכי למדא גדולים רואים כי הloss גדל והמודל מתחיל להכנס כבר לunder fit, הן בtrain והן בtest הביצועים מתחילים לרדת. זו התנהגות צפויה עבור הרגרסיה, שכן ראינו בתרגול כי ככל שאנו מגדילים את הפרמטר למדא, אנחנו מציבים מגבלה על גודל הנורמה האפשרי של argmin כך שהווקטור האופטימלי הולך ומתקרב ל0 ככל שלמדא גדל כי נותנים יותר משקל על הנורמה ולכן יכולת ההכללה מתחילה לרדת בשלב הזה ומקבלים מודלים פשוטים מדי.
הלמדא האופטימלי שמצאנו באימון הוא:

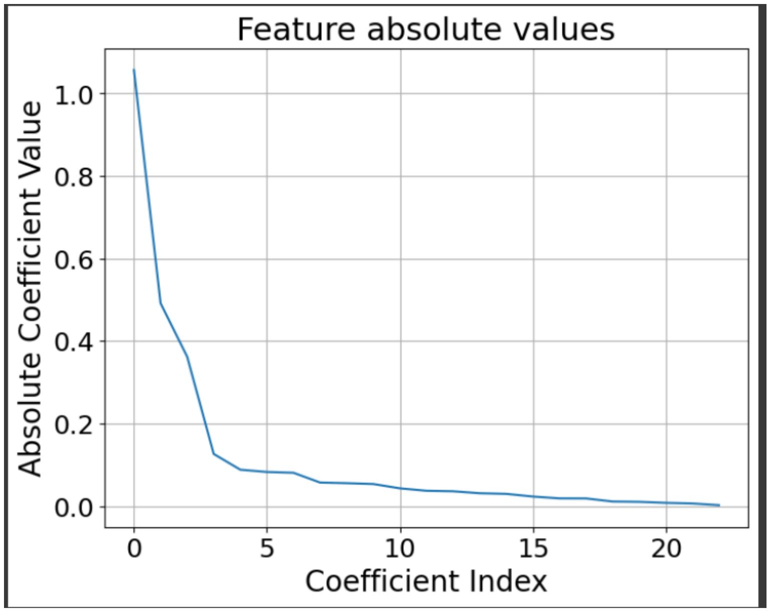
```
Optimal regularization strength (lamda): 2.93e+01
Training Huber loss at optimal lamda: 1.74(+/- 0.10)
Validation MSE at optimal lamda: 3.97(+/- 0.95)
```

(Q11)

Model	**Section**	**Train Huber Loss** (cross validated)	**Train MSE** (cross validated)	**Valid MSE** (cross validated)
Dummy	2	2.29(+/- 0.1)	5.27 (+/- 0.26)	5.29 (+/- 1.051)
Linear	2	1.73(+/- 0.1)		4.02 (+/- 1)
Ridge linear	3	1.74(+/- 0.1)		3.97 (+/- 0.95)

(Q12)

לא רשום מה להוסיף לדוח בשאלה הזאת, אבל אנחנו מניחות שאת זה:




```
sorted_indices: [ 4 10 14 11  1 15 13  7  5  3  9  2  8 17 21 22  6 16  0 12 19 20 18]
```

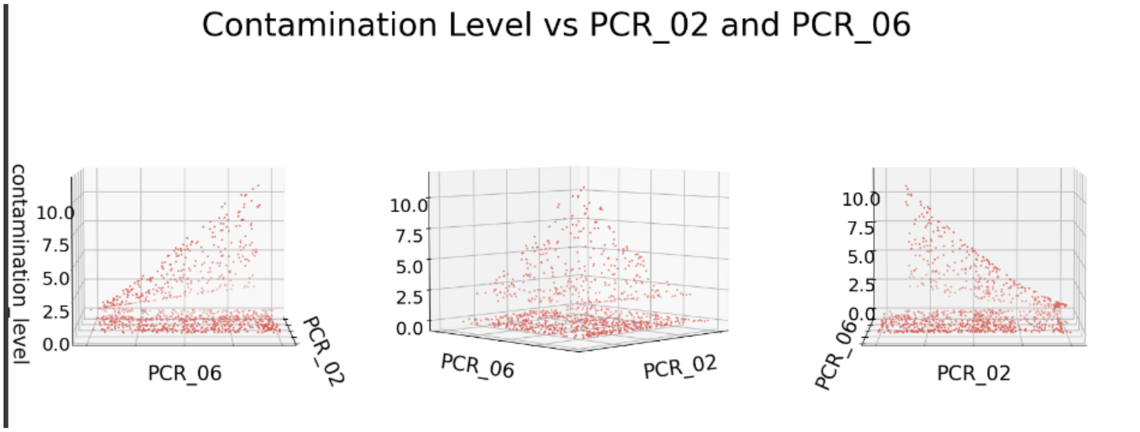
(Q13)

ראינו בתרגול וגם רואים בשאלה הקודמת שככל שמגדילים את הלמדא ברגולריזציה אנחנו מקבלים שהcoefficients הולכים ומתקרבים לאפס(לא מתאפסים) וזה כי אנחנו דורשים נורמה יותר ויותר קטנה על ידי הגדלת מקדם רגולריזציה על הנורמה (למדא) וזה גורם למשקלים לדעוך לכיוון אפס. ייתכן גם שהridge ייתן משקל גבוה יותר לפיצ׳רים חשובים יותר (זה תלוי גם בנרמול הפיצ׳רים) אך אם למדא גדול מאד, יהיה קשה לפרש את החשיבות של כל פיצ׳ר כי כל המשקולות יהיו קטנים מאוד.

לכן אנחנו חושבות שridge פחות מתאים למיקסום הinterpretability .
לעומת זאת, בשימוש ברגולריזציית lasso נוכל למקסם יותר את הinterpretability כיוון שחלק מהמקדמים של הפיצ׳רים מתאפסים. אנחנו עושים בעצם סוג feature selection די בקלות (בשיעורי בית 1 עבדנו מאוד קשה בשביל לעשות זאת). לתכונות עם מקדמים גדולים תהיה השפעה גדולה יותר על החיזוי מה שאומר שזה נותן להם חשיבות גדולה יותר במודל הרגרסייה וזה נותן לנו רמז אילו תכונות טובות לניבוי, לכן נרצה שתכונות שלא טובות לניבוי יתאפסו (עם זאת חשוב לנרמל את הדאטה כדי שזה יתקיים).

Section 4: Feature Mappings (visualization)

(Q14)



אנו מחפשים להבין האם ניתן להשתמש במודל לינארי כדי לחזות את המשתנה הרציף contamination level, לכן נרצה להבין האם קיים איזושהו קשר לינארי בין משתני החיזוי למשתנה הפרדיקציה (אנחנו לא מחפשים הפרדה לינארית כי לא מדובר בסיווג).

ניתן לראות שיש ״2 קבוצות״ שונות של נקודות – הקבוצה האחת הן נקודות שמקבלות contamination level אפס או קרוב מאוד לאפס בלי תלות בערכים בpcr_02 וpcr_06 (כלומר מתנהגות כמו המישור הקבוע z=0) וקבוצה שניה יוצרות ושאר קירוב של היפר מישור עולה. לכן לא נראה שיש רק מישור אחד שיתאים בצורה טובה לחיזוי הדאטה.

אם נאמן מודל רגרסיה על דאטה זה, נקבל מודל שינסה להחזיר את הערך הממוצע של y לכל x בקירוב (כי מדובר בבעיית LS עם שגיאה ריבועית) ולכן המישור שנקבל יהיה מישור ש״יפריד״ בין 2 הקבוצות, כלומר יהיה ממוקם בפוזיציה שנמצאת בין המישור ל z=0 למישור המשופע, והוא לא יהיה קירוב טוב מאד לאף אחת מהקבוצות. (נשים לב שרוב הנקודות נמצאות ב-contamination level אפס ולכן המישור שנקבל יהיה יותר קרוב אליהם ויקבל שיפוע קטן כלפי הנקודות העולות).

לאור מה שכתבנו, אנו חושבות שלמדל את הבעיה עם מודל רגרסיה לינארית לא יחזה מספיק את משתנה המטרה, ולכן אולי כדאי להשתמש בפונקציות קרנל כלשהן שיכולות למפות את הבעיה למימד בו היא מתאימה לחיזוי לינארי.

(Q15)

פה לא כל כך מובן מה להוסיף לדוח, החלטנו להוסיף את הקוד שמבצע את הפעולות הנדרשות.

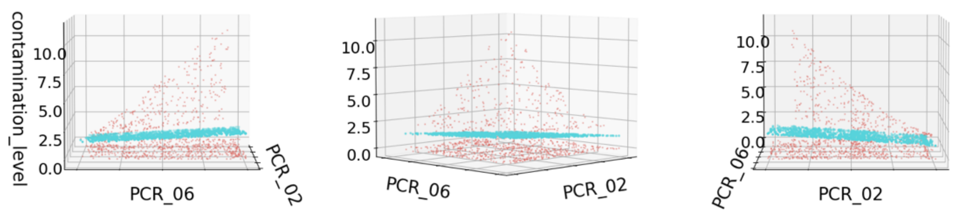
```
delta_0_q_15= calculate_huber_delta(X_train_section4, y_train_section4)
Huber_regressor_model2 = HuberRegressor(alpha=best_lamda, epsilon=delta_0_q_15, fit_intercept=True)
Huber_regressor_model2.fit(X_train_section4, y_train_section4)

y_train_pred = Huber_regressor_model2.predict(X_train_section4)
y_test_pred = Huber_regressor_model2.predict(X_test_section4)
```

והתוצאות שקיבלנו על המודל:

```
delta_0: 3.39
Training Huber Loss: 2.2622476779753633
Training MSE: 5.066776810489589
Test MSE: 4.930536273026795
```

(Q16)

Contamination Level vs PCR_02 and PCR_06 and matched Huber Regressor



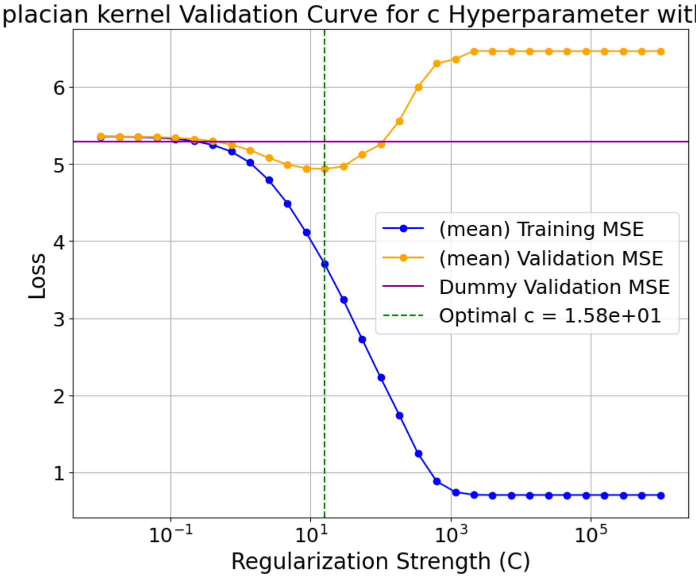
ניתן לראות שקיבלנו מודל שמתאים לציפיות שלנו מסעיף 14.

(Q17)

נדרשנו בסעיף זה לבצע hyperparameter Tuning for C רק על פי שני הפיצרים.

להלן הגרף שקיבלנו:

SVR laplacian kernel Validation Curve for c Hyperparameter with MSE Loss)



הפרמטר האופטימלי שקיבלנו + התוצאות שנתן (רק ע״פ 2 פיצ׳רים):

```
Optimal regularization strength (c): 1.58e+01
Training MSE loss at optimal c: 3.70(+/- 0.26)
Validation MSE loss at optimal c: 4.94(+/- 1.02)
```

Optimal C = 15.8

התוצאות שקיבלנו הן צפויות, ככל שC קטן יותר יש יותר דגש על הקטנת הנורמה (מצב של underfit) וככל שC גדול יותר יש יותר דגש על צמצום השגיאה של מודל הSVR על קבוצת האימון (עד להגעה למצב overfit).

לאחר מכן עשינו cross validation יחיד עם הפרמטר האופטימלי שמצאנו 15.8 = C על כל קבוצת האימון

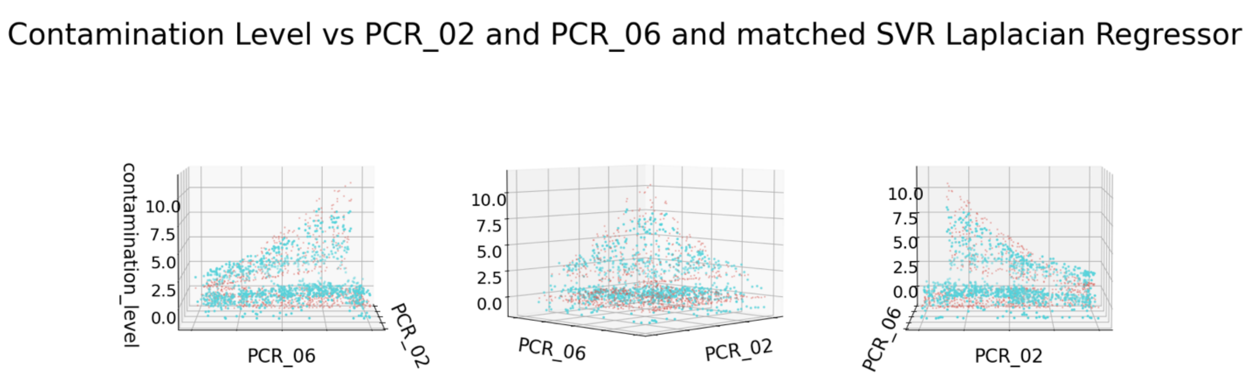
ואלה התוצאות שקיבלנו:

```
Training MSE loss at optimal c: 0.67(+/- 0.26)
Validation MSE loss at optimal c: 2.57(+/- 1.02)
```

אנחנו נשתמש בהם בטבלה, כפי שצוין בהוראות בפיאצה:

Model	**Section**	**Train Huber Loss** (cross validated)	**Train MSE** (cross validated)	**Valid MSE** (cross validated)
Dummy	2	2.29(+/- 0.1)	5.27 (+/- 0.26)	5.29 (+/- 1.051)
Linear	2	1.73(+/- 0.1)		4.02 (+/- 1)
Ridge linear	3	1.74(+/- 0.1)		3.97 (+/- 0.95)
SVR + Laplace	4		0.67(+/- 0.26)	2.57(+/- 1.02)

(Q18)



(Q19)

מודל הHuber regressor המתייחס למטריקה של Huber loss הוא מודל המאמן רגרסור לינארי המפגין עמידות כנגד outliers אך במקרה שלנו ניתן לראות ע״פ הplot ומטריקות השגיאה כי החיזוי שלו על הנקודות יהיה פחות טוב (הסברנו בסעיף 14 שהדאטה פחות מתאים למודל לינארי).

לעומת זאת, מודל הSVR+ מיפוי קרנל מספק התאמה טובה יותר ל training set ולוכד פרטים עדינים יותר . על אף שגם מודל הsvr מחפש חזאי לינארי, המיפוי של הפיצ׳רים באמצעות קרנל הלפלסיאן מאפשר לייצר חזאי שאינו וכך ניתן לקבל ביצועים טובים יותר באיזורים עם דפוסים מורכבים כמו במקרה שלנו. מהמטריקות והplot ניתן לראות שאכן המודל מתאים עצמו בצורה מאד טובה לtrain set (מה שגם יכול לגרום לoverfit) וכי שגיאת הולידציה נמוכה יותר.

אנחנו מסיקות שהשיפור בביצועים הגיע בעיקר מהמיפוי של הפיצ׳רים באמצעות קרנל הלפלסיאן ולא שינוי המודל.

(Q20)

Model	**Section**	**Train Loss** (cross validated)	**Valid MSE** (cross validated)	Test MSE
Dummy	2	5.27 (+/- 0.26) **(MSE)**	5.29 (+/- 1.051)	5.160304602563259
Linear	2	1.73(+/- 0.1) **(Huber)**	4.02 (+/- 1)	3.748605086090574
Ridge linear	3	1.74(+/- 0.1) **(Huber)**	3.97 (+/- 0.95)	3.7388682486726066
SVR + Laplace	4	0.67(+/- 0.26) **(MSE)**	2.57(+/- 1.02)	2.3622709697094715

המודל הטוב ביותר שקיבלנו הוא SVR + Laplace עם שגיאה ריבועית ממוצעת הכי נמוכה על הtest.