**מטלה 2**

חלק א' – Data exploration

המטרה:

ביצוע חיזוי (forecasting) של צריכת חשמל (Global Active Power) עבור קבוע הזמן הקרוב העתיד לבוא (t), בהינתן המידע הקודם לו.

תיאור כללי של בסיס הנתונים:

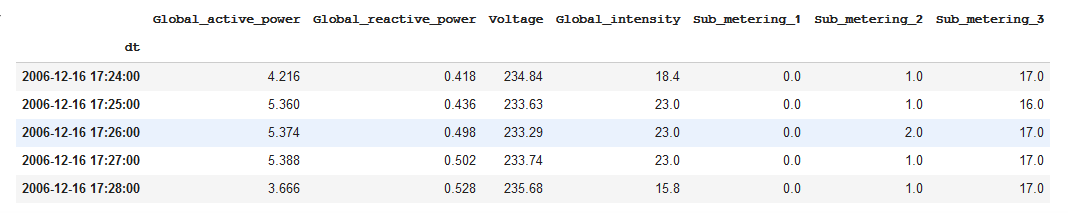
בחרנו ללכת על הData Set של individual household electric power consumption, שמכילה כ-2,075,259 מדידות של צריכת חשמל עבור בית הממוקם בסו (פרבר שנמצא 7 קמ מחוץ לפריז):



כאשר כל מדידה מייצגת ממוצע צריכה של דקת שימוש. המדדים נלקחו בתאריכים: 12/2006-10/2010 – סה"כ 47 חודשים.

תיאור הנתונים בצורה טבלאית:

בוא נצפה בנתוני בצורה טבלאית ונסביר על כל עמודה:



קצת קשה לראות, אז אפרט:

1. **Date:** תאריך בפורמט dd/mm/yyyy
2. **Time:** זמן בפורמט: hh/mm/ss
3. **Global Active Power:** כמות חשמל ממוצעת בדקה של המדידה (בקילו וואט).
4. **Global Reactive Power:** כמות החשמל שהתבזזה ולא נוצלה בדקה של מדידה (בקילו וואט)
5. **Voltage:** רמת המתח (בוולט)
6. **Global Intensity:** עוצמת הזרם הממוצעת שעברה בדקה של מדידה (אמפר).
7. **Sub\_Metering\_1:** האנרגיה שנוצלה עבור תת קבוצה 1 (שמוגדרת להיות המטבח בבית - כולל את המדיח כלים, תנור, מיקרוגל וכו'). נמדד בוואט לשעה של אנרגיה פעילה.
8. **Sub\_Metering\_2:** האנרגיה שנוצלה עבור תת קבוצה 2 (שמוגדרת להיות חדר הכביסות בבית - כולל את מכונת הכביסה, מייבש ומקרר). נמדד בוואט לשעה של אנרגיה פעילה.
9. **Sub\_Metering\_3:** האנרגיה שנוצלה עבור תת קבוצה 3 (שמוגדרת להיות אביזרי החשמל - בוילר (מחמם המים), מיזוג אוויר). נמדד בוואט לשעה של אנרגיה פעילה.

בזמן היבוא ראיתי כי ישנם ערכים חסרים שמיוצגים ע"י "?", לכן הפכתי אותם לnan (אובייקט של numpy), על מנת לטפל בהם בהמשך.

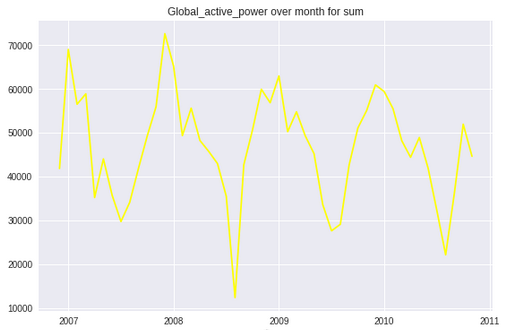
מבנה הנתונים:

 כמו שציינתי: 2,075,259 רשומות כאשר כל רשומה מכילה פרטים על 7 Features (האינדקס לא נחשב).

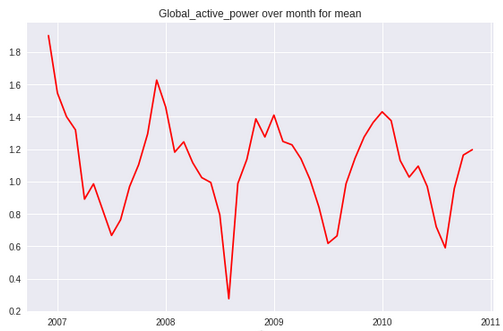
הצגת הנתונים בצורות ויזואלית:

כיוון שהאינדקס שלנו הוא בפורמט תאריך נוכל להשתמש בפונקציות resample ולהציג נתונים מקובצים לפי יום / חודש וכו'.

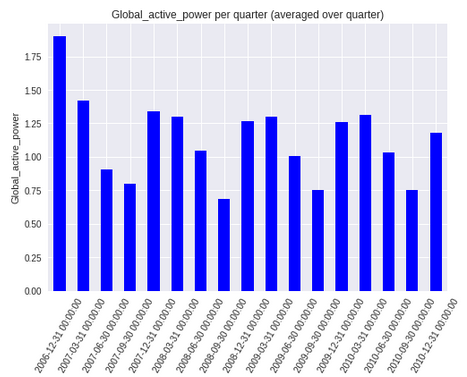
נציג במספר סוגי תצוגה:



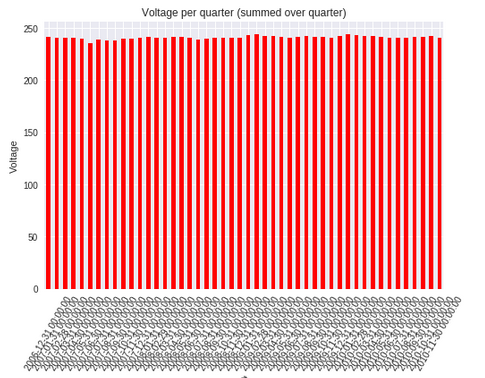
סיכום חודשי של סכימת שימוש החשמל



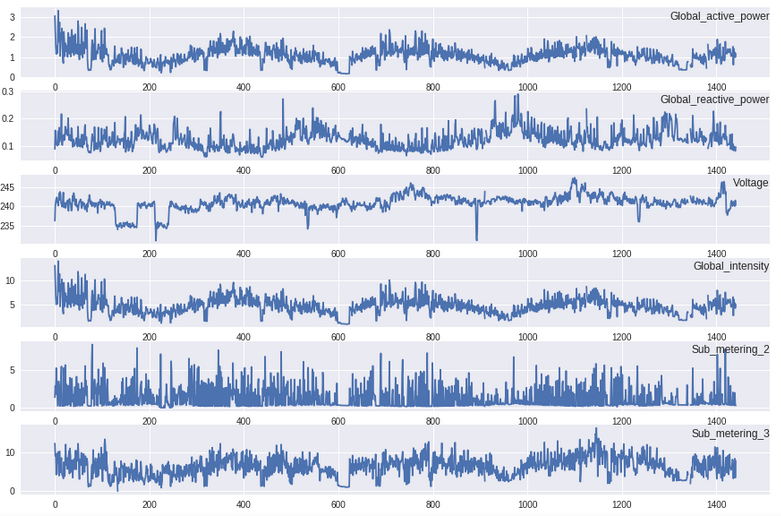
ממוצע שימוש החשמל היומי (ממוצע חודשי עבור שימוש יומי)



ממוצע שימוש החשמל היומי (ממוצע רבעוני עבור שימוש יומי)



ממוצע שימוש המתח היומי (ממוצע רבעוני עבור שימוש יומי)



השוואת ממוצע שימוש יומי עבור Feature

**מסקנות מהצגת הנתונים בצורה גרפית:**

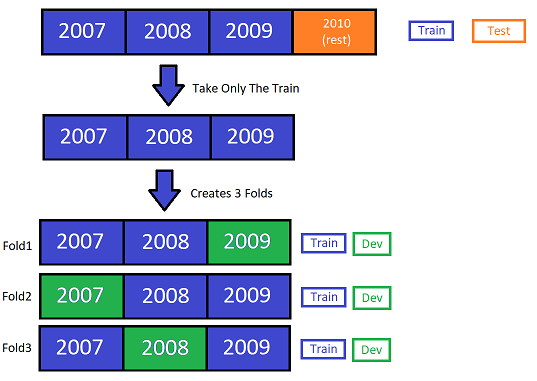
1. ניתן לראות מחוזריות ברורה ברוה הפרמטרים ובעיקר בGlobal Active Power אותו אנחנו מנבאים.
2. החצי השנה הראשונה נראית מאוד רועשת. לכן לאחר מספר בדיקות החלטנו להוריד אותה ולהתחיל את הDataset מאחרי החצי שנה הראשונה.

חלק ב' – The Forecasting Model

1. **בחירת שיטת הValidation:**

תחילה, ביצענו רק חלוקה לTrain-Test רגיל כך ששלושת השנים הראשונות יהוו את האימון והניבוי יתבצע על השנה האחרונה.

אולם, קיבלנו תוצאות שנראו לנו מוזרות. קיבלנו כי הLoss של הTest יותר נמוך מהLoss של הTrain. כדי לוודא שהחלוקה שלנו באמת מוצלחת, עברנו להשתמש בK Fold cross Validation בצורה הבאה:



K Fold Cross Validation

לבדיקת טיב המודל נשתמש בRMSE לחישוב טעויות:

RMSE. הדגש במשימה ניתן לRMSE (יתכן ויוצגו תמונות עם MAE, אך הם נועדו לשימוש עצמי שלנו), באופן כללי ניתן להגיד כי נכון להשתמש בRMSE כיוון שהוא "מעניש" תוצאות עם טעות גבוהה, לעומת MAE שיותר מתייחס לממוצע. אנחנו רוצים שכל תוצאה יחסית גבוהה, ושבאמת לא תהיה פתאום תוצאה עם טעות קיצונית כיוון שניבוי בטעות קיצוני יכול לא לשרת אותנו כראוי במטלה זו.

**ביצוע Pre Processing:**

1. **טיפול במשתנים שאינם מספריים:** יש לנו 2 משתנים שהם אינם Float, איתם הרשת לא תדע להתמודד – מדובר בDate ובTime. אם נרצה לחשוב מה משפיע יכול להשפיע על הניבוי שלנו הם 2 דברים: בהקשר של התאריך: לדעתי הגורם שהכי יכול להשפיע על הצריכה ביום זה איזה חודש אנחנו – אנו חושבים זאת כיוון שבחורף למשל יש הרבה יותר השפעה על צריכת החשמל, האופציה של לבחור בשנה לא נראה ריאלי וגם לא היום בחודש, לכן הפכנו כל תאריך למספר החודש בשנה.

לגבי השעה: הפכנו את השעה ביום למה השעה עצמה בלי הדקות והשניות. האם בזבוז החשמל הנוכחי הוא ב17:25 או 17:26 לא נראה כגורם משפיע, אך האם החשמל כרגע מבוזבז ב03 (AM) או 03 (PM) נראה מאוד משמעותי. אלו השורות:



1. **טיפול ב"?":**  היה לנו בבסיס הנתונים המקורים ערכים חסרים, ההשלמה הכי מתאימה לנו הייתה נראית כמו השלמה לפי הערך שהיה בדקה הקודמת (למשל עדיפה על פני ממוצע העמודה במקרה זה).
2. **נרמול הנתונים:** המספרים לא נעים באותם טווחים, לכן נרצה לנרמל את הנתונים. נעשה זאת ע"י החסרה בממוצע וחילוק בסטיית תקן (כדי לקבל ממוצע 1 וסטיית תקן 0).

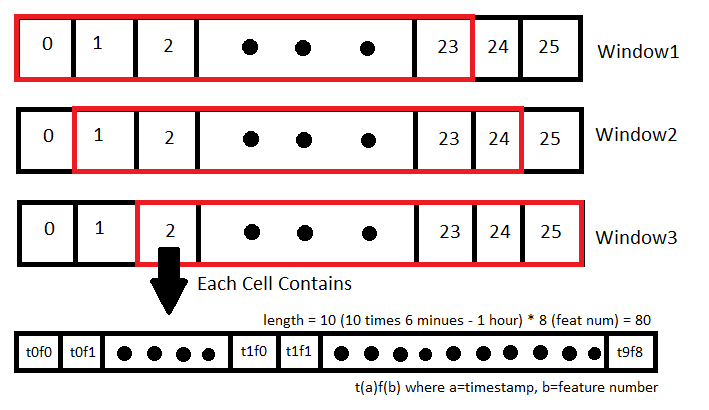
את העמודה המשמעותית של צריכת החשמל לאחר מספר ניסיונות השארנו בעריכה המקוריים כיוון שהיא מהווה את מירב המשקל לצורך הניבוי וגם כיוון שתוצאות הניבוי שלנו יצאו נמוכות מאוד, וכאשר הערך האמיתי הוא גבוה, הרשת לא מנבאת ערך שקרוב אליו.

1. **הזזת הנתונים שלב אחד אחורה כדי ליצור את קבועי הזמן:**

תחילה, כדי גם להקטין את כמות המידע הגדול שלנו (בעיית חומרה וזמן הרצה), וגם כיוון שאין כמעט שונות במשך 6 דקות בצריכת החשמל, החלטנו לקבץ את הנתונים לכל 6 דקות על פי ממוצע.

רצינו ליצור מצב שתהליך הניבוי שלנו פועל בצורה כזו:

הקלט: 24 שעות אחורה (מוחלק ל24 תאים שכל אחד כולל וקטור ארוך של שעה–מוסבר בציור למטה):



כלומר:

**באדום:** חלון תצפיות שמכיל את השעה האחרונה, מחולקת ל24 יחידות של שעה כאשר בכל יחידת שעה יש לנו 10 יחידות של 6 דקות עם 8 הפיצ'רים המלווים לרשומה)

הפלט: מה תהיה צריכת החשמל הממוצעת בעוד כשעה מעכשיו.

כלומר אם כרגע יום ב' בשעה 1300. אני אקבל כקלט את הנתונים מיום א' בשעה 1300 עד הרגע, ואצטרך לחזות מה תהיה צריכת החשמל הממוצעת בשעות 1400-1500. (פלט יחיד של הממוצע השעתי).

למטרת החיזוי הספציפי הזו הגענו לאחר שניסיתי לחזות למשל את הצריכה לדקה הקרובה על סמך החצי שעה שקדמה לו. אולם במקרה כזה, אין לנו שום יתרון במודל מסוג RNN שמכיל שכבת LSTM שמסוגלת בעצם לזכור דברים לזיכרון טווח קצר-ארוך ולמצוא לנו דפוסים.

1. **המודל הנאיבי לצורך השוואה:**

מבין השיטות הנאיביות בחרנו ללכת על המקרה הכי ממוצע שיש, ולכן המודל המאוד ראשוני ונאיבי שמולו נבצע את ההשוואה יהיה לנבא את הערך הממוצע לאורך כל השנים של העמודה אותו אנו רוצים לנבא (צריכת החשמל). הערך ההממוצע הוא: 1.086 (לפני נרמול).

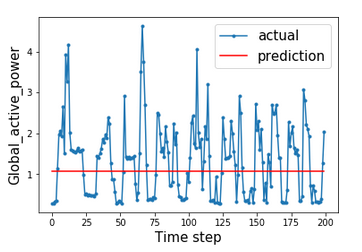
התוצאה:



נוראית

נזכיר כי ככל השגיאה קרובה ל-0 היא טובה, וככל שגדולה יותר – לא טובה.

נציג גם את טבלת הניבויים לעומת הערך האמיתי:



נוראית 2 – שובו של הנוראית

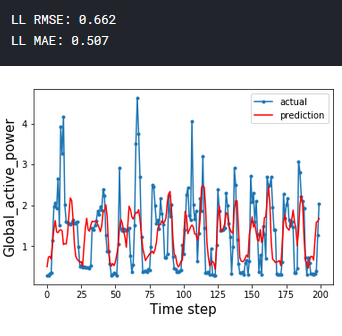
לא ציפינו בשלב זה לקבל תוצאות טובות, וכך באמת יצא כפי שחשבנו.

1. **בחירת מודל ML קלאסי:**

בחרנו לבצע את הניבוי באמצעות מודל ML קלאסי שגם מתאים מאוד למצב הזה:

רגרסיה לינארית.

התוצאות:



התוצאות היו טובות מאוד יחסית לפעם הקודמת. קשה להעריך איך יהיה המודל שנבנה באמצעות רשת הניורונים לכן קשה לקבוע כרגע את טיב התוצאה של המודל הזה, אך כמובן שיש שיפור מאוד משמעותי מהמודל הנאיבי. אנחנו יוצרים לדרך עם Benchmark של RMSE =0.662 לבניית מודל הNN.

1. **בניית מודל הNN הראשוני:**

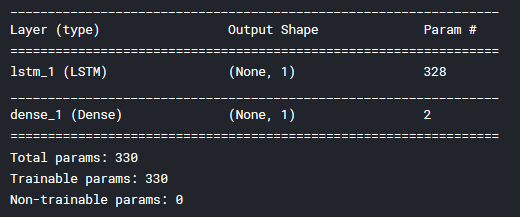
נבנה מודל רשת הבסיסי ביותר שכוללת שכבת LSTM של Many To One (מקבל את השעה אחורה ומנבא את הממוצע של צריכת החשמל בשעה שמיד אחרי השעה הבאה).

Many כיוון שהוא מקבל 24 קלטים שכל אחד בגודל 80 Feature.

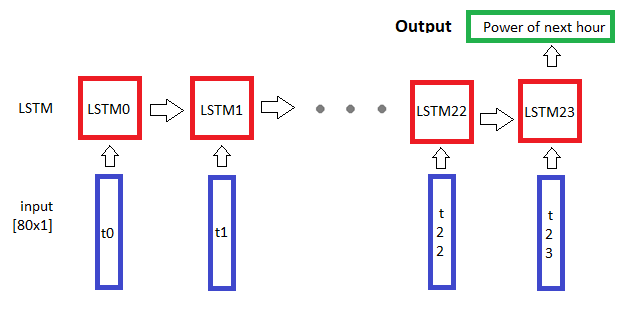
One כיוון שהוא מנבא רק פלט אחד.

כלומר יהיו לנו ברשת שכבת LSTM אחת שמורכבת מ24 "יחידות" LSTM מחוברות בטור.

מבנה הרשת:



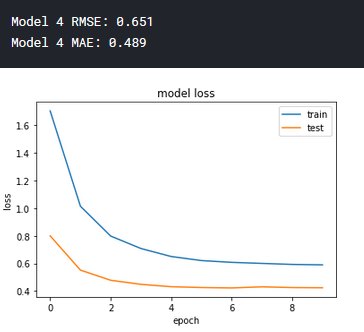
תיאור בKeras



תיאור שלנו

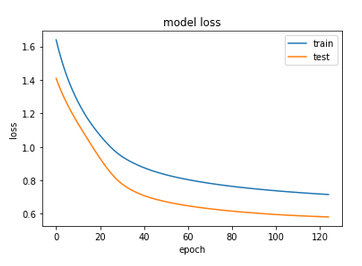
**הגרסא הראשונית (שימוש בTrain Test Split רגיל):**

כמו שתיארתי בשלב בחירת הואלידציה, קיבלנו בהרצה ראשונה את התוצאות הבאות:

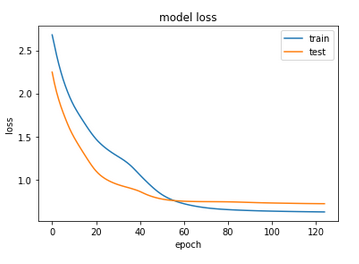


קיבלנו כי הTest Loss נמוך יותר מהTrain Loss. הסבר לכך יכול להיות שהשנה האחרונה קלה מאוד לניבוי (נגיד תמיד בכל דקה בשנה האחרונה תמיד השתמשו ב3 קילו וואט לשעה). לכן בשלב הזה החלטנו לפנות לשיטת הK Fold Cross Validation כדי לבדוק מה יקרה אם ניקח Train וDev שונים. לאחר הרצת המודל על שלושת הFold השונים כפי שמתוארים בציור שהמופיע בחלק של בחירת שיטת הואלידציה קיבלנו את התוצאות הבאות:

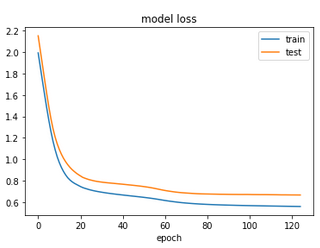
**הגרסא השנייה (שימוש בK Fold Cross Validation):**



Fold1



Fold2

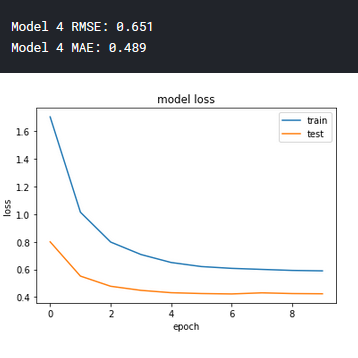


Fold3

נבצע סיכום קצר של התוצאות בטבלה:

|  |  |  |  |
| --- | --- | --- | --- |
| Fold # | RMSE | MAE | Test<Train |
| 1 | 0.67 | 0.525 | V |
| 2 | 0.692 | 0.545 | X |
| 3 | 0.68 | 0.517 | V |
| Total (Avg) | 0.68 | 0.52 | V |

כלומר ניתן לקבוע כי אין בעיה במודל שלנו ויתכן כי באמת בשנה האחרונה (2010) הדגימות היו יחסיות ממוצעות וקלות לניבוי (למשל במשך כל השנה באופן קבוע צריכת החשמל עמדה על 3 קילו וואט לשעה). בוא נצפה בתוצאות המודל על סמך Train-Test המקורי:



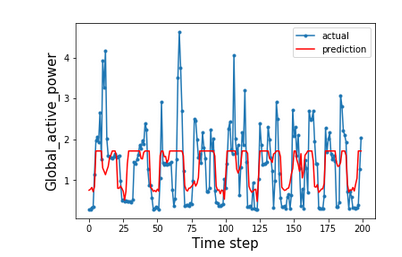
כרגע במצב הנוכחי המודל שלנו טיפה יותר מוצלח בשני הממדים כפי שניתן לראות בטבלה:

|  |  |  |
| --- | --- | --- |
| מודל | RMSE | MAE |
| Mean Prediction | 0.737 | 0.622 |
| ML | 0.662 | 0.507 |
| Naïve NN | 0.651 | 0.489 |

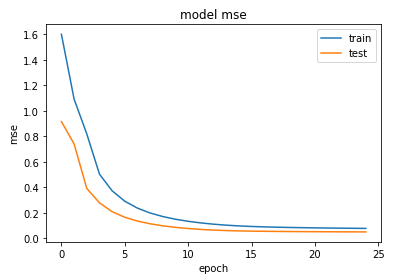
1. **ניתוח תוצאות המודל והבנה כיצד לשפר:**

תחילה נגדיר מה הוא פתרון אופטימלי: בבעיות חיזוי אני חושב שקשה לקבוע מה יהיו תוצאות האלגוריתם האופטימלי כיוון שבשונה מתרגיל קודם בו המשימה הייתה סיווג תמונה, ובמשימה זו ניתן לקבוע כי אדם יוכל לסווג בטעות אפסית עד לא קיימת, פה לא ניתן לקבוע כי ישנו אדם שיוכל לחזות את צריכה החשמל לדקה הקרובה בצורה יותר טובה ממחשב ו/או ניתן לקבוע באיזה דיוק תתבצע התוצאה. לכן נגדיר לנו כי ננסה להשתפר כמה שיותר עד טעות אפסית, אמנם זה הוא לא מצב בהכרח אפשרי.

נצפה בתוצאות בצורת גרף של ניבוי אל מול תוצאה שהייתה אמורה לצאת:



לצורך ניתוח עמוק יותר של המצב שלנו בוא נצפה בגרף הmse – epochs:



נזכיר שוב את המושגים מעבודה קודמת לצורך ניתוח הבעיה:

Bias: קובע כמה הטעות שלנו גדולה ביחס לטעות האופטימלית האפשרית. במקרה שלנו הגדרנו מקודם כי לא בהכרח הגיוני לקבוע שהטעות האופטימלית היא 0, אך ננסה לשאוף לטעות נמוכה יותר, לכן ניתן לגבוה כי כרגע הbias שלנו דורש תיקון.

Variance: מה ההפרש בשגיאה בין הTrain לTest. במצב שלנו יש לנו שונות אפסית (ומטה) כלומר הloss של הtest נמוך מהloss של הtrain.

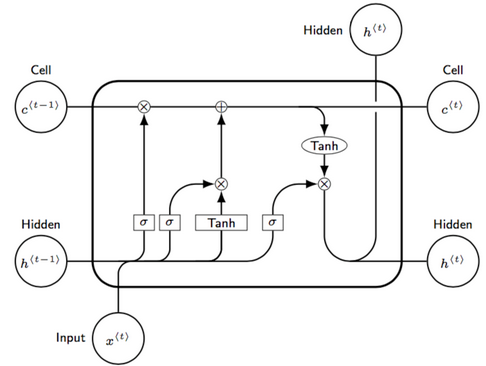
אנחנו נמצאים כרגע במצב של Underfitting (אמנם לא חמור במיוחד אך עדיין). ניגש לטפל בנושא:

פעולות להפחתת הBias:

1. **הרחבת מבנה הרשת:** יתכן ושכבה אחת של LSTM לא מספיקה כדי ליצור ניבויים בצורה מספיק טובה. יתכן ונצטרך להוסיף עוד שכבות.
2. **שינוי סוג השכבה:** יתכן והבחירה בLSTM לא הייתה מתאימה, ואולי GRU על אף שפשוטה יותר תביא לתוצאות טובות יותר.כי במהלך שעה אולי אין קונטקסט של זיכרון ארוך וקצר, אך אנו עובדים בטווח זמן של 24 שעות אחורה ולכן אנו צופים שיתכן ותהיה חוקיות שתוכל לעזור לנו בניוביים.

פעולות לטיפול בבעיית האי ניבוי ערכים הגבוהים מ1.5-2:

1. ביטול הפונקציה הלא לינארית שמתבצעת בתוך שכבת הLSTM. נסביר:



תרשים LSTM

אפשר לראות שהתא הLSTM "משטח" את הנתונים, יכול להיות שזה גורם להגבלת ערכי הפרדיקציה.

תעדוף:

אנו חושבים כי קודם כל עלינו לפתור את בעית הערכים של הפרדיקציה ומשם נוכל להמשיך לבדוק את 2/3.

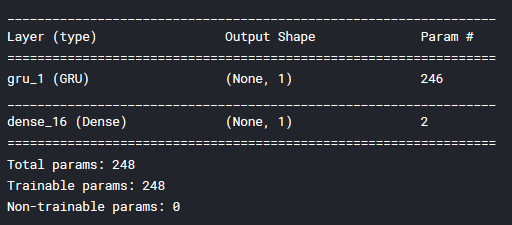
|  |  |  |
| --- | --- | --- |
| מספר פעולה | תיאור | תעדוף |
| 2 | שינוי סוג השכבה | 1 |
| 1 | שינוי מבנה הרשת | 2 |
| 3 | טיפול בערכי הניבוי | 3 |

1. **מימוש התוצאות:**

**שיפור ראשון:**

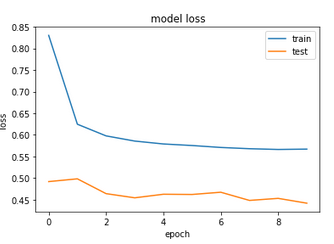
**שינוי סוג השכבה:**

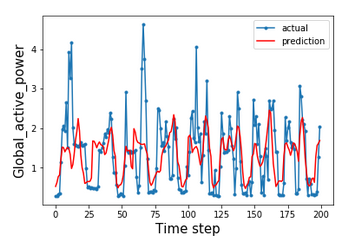
הגיוני קודם להסגר על סוג השכבה. נבצע השוואה לכן בין LSTM לGRU וננסה לראות מי מהם מביא לנו תוצאות יותר טובות. הGRU מודל יותר פשוט, אך אולי יותר מתאים למודל שלנו כיוון שכפי שראינו גם מודל פשוט כמו רגרסיה לינארית הביא תוצאות יפות שלא רחוקות בכלל מהתוצאות של מודל הLSTM.



התוצאות:







נסכם בטבלה:

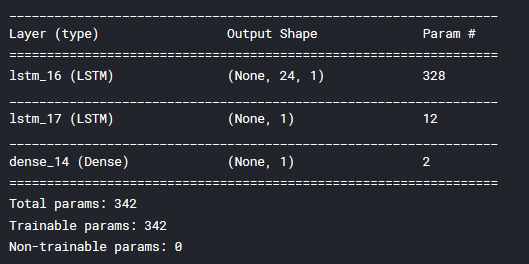
|  |  |  |
| --- | --- | --- |
| מודל | RMSE | MAE |
| Mean Prediction | 0.737 | 0.622 |
| ML | 0.662 | 0.507 |
| Naïve NN (LSTM) | 0.651 | 0.489 |
| Naïve NN (GRU) | 0.665 | 0.512 |

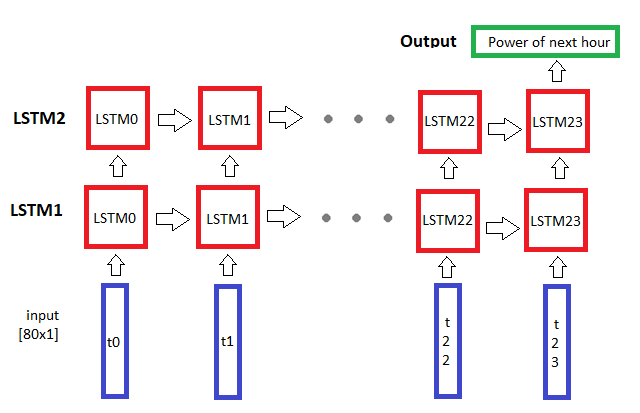
ברור כי **עדיף בהרבה להשתמש בLSTM** אשר טוב ביותר בשני המדדים.

**שיפור שני:**

**שינוי מבנה הרשת:**

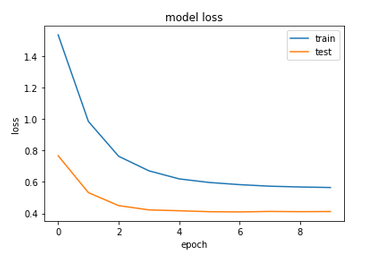
ננסה להכניס עוד עומק לRNN שלנו ולהוסיף עוד שכבה של LSTM. כך זה נראה:

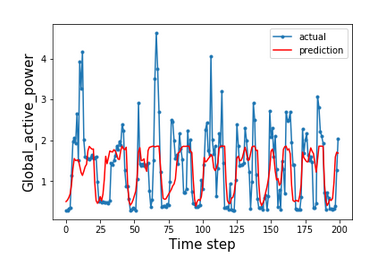




נבחן את התוצאות:







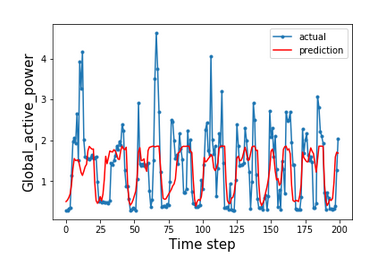
חל שיפור,נבצע השוואה למקרים הקודמים והמשכנו הלאה!

|  |  |  |
| --- | --- | --- |
| מודל | RMSE | MAE |
| Mean Prediction | 0.737 | 0.622 |
| ML | 0.662 | 0.507 |
| Naïve NN (LSTM) | 0.651 | 0.489 |
| 2 LSTM Layers | 0.642 | 0.478 |

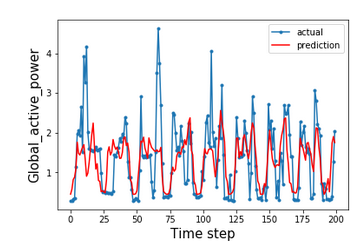
**שיפור שלישי (על אף שלא נתבקשנו, נציג רק את הניסיון):**

**ניסיון טיפול בערכי הניבוי הנמוכים:**

כפי שהסברתי מקודם, ננסה ליישם כל תא בLSTM ללא פונקציות הTanH הדיפולטיבית. מדוע? כי יתכן שפונקציית האקטיבציה הזו "מועכת" את הנתונים. ננסה את השכבה ללא פונקציה אקטיבציה ונבחן מה קורה לתוצאות.



**תוצאות הישנות (עם האקטיבציה)**



**תוצאות החדשות (בלי האקטיבציה)**

אמנם, ניתן לראות כי אכן טווח ערכי הניבוי עלה מעט, אך מצד שני ירדנו בדיוק, לכן נשאר בשיטה הקודמת.

**טבלה מסכמת את התהליך עם אחוז השיפור כל פעם:**

|  |  |  |
| --- | --- | --- |
| מודל | RMSE (% improve) | MAE(% improve) |
| Mean Prediction | 0.737 | 0.622 |
| ML | 0.662 (10.17%) | 0.507 |
| Naïve NN (LSTM) | 0.651 (1.6%) | 0.489 |
| 2 LSTM Layers | 0.642 (1.3%) | 0.478 |
| 2 Layers, No Activation | 0.666 | 0.502 |
| 2 Layers, Relu Activation | 0.639 | 0.474 |

לפני סיום החלטנו לבדוק גם את Relu, כיוון שרק מבטלת את הערכים השלילים, ויתכן כי הדבר יביא לנו אפקט חיובי וישפר את התוצאות, ואכן התוצאות היו קצת יותר טובות ולכן זו הגרסא הסופית של המודל שלנו.

**סיכום החלק הראשון:**

נושא קשה ומורכב יותר להערכתנו לעומת הנושא הקודם של עיבוד תמונה. היה נדרש למידה עמוקה ורצינית של הנושא, ובכל שלב באמת נתקלנו בהרבה בעיות שצצו שנאלצו לפתור. המסקנה שלנו לגבי הDataset הזה היא שבאמת לא ניתן להפיק ממנו מידע לגבי העתיד, וזה לא כיוון שהיו חסרות דוגמאות או שהמודל היה לא טוב, פשוט מאוד עבור Dataset זה לא ניתן להשיג טובות יותר תוך שימוש בשיטות למידה עמוקה אל מול שיטות פשוטות יותר, וזאת בניגוד למטלה הקודמת. יתכן כי באמת ניתן לקבוע כי מודל לינארי הוא המודל המסביר את רוב המידע לחיזוי.

מספר נקודות חשובות נוספות:

* + - 1. אחרי הרבה מיץ שיצא, למדנו שלפעמים מודל פשוט כמו רגרסיה לינארית יכול להביא לתוצאות כמעט זהות לגמרי למודל RNN מורכב עם יותר משכבה אחת. כלומר לא תמיד ללכת בפתרון של למידה עמוקה (שיכול לעלות לי במשאבים וזמן חישוב יקר) הוא הפתרון הטוב והאולטימטיבי.
      2. הגדרת המשימה חשובה מאוד. למה אנחנו אומרים את זה? כי את כל המשימה תחילה ביצענו כאשר הגדרנו לעצמנו את המשימה הבאה: לחזות מה תהיה הצריכה הממוצעת של הדקה הבאה בהינתן החצי שעה האחרונה. זו הייתה הגדרת משימה לא טובה, שגם לא מנצלת את יכולות הלמידה העמוקה, מה שגם שמודל טובה למטרה זו יהיה לנבא מה הייתה צריכת חשמל בדקה שלפני. בגלל הגדרת המשימה השגויה נאלצנו לבצע את כל החלק הזה פעמיים.
      3. אנו שמחים על הלמידה של הנושא החדש, ועל התהליך גם פה שעברנו (בדומה למטלה הקודמת), וראינו איך המודל משתפר. אנו מרגישים כי משימת החיזוי היא משימה קשה ושונה במעט ממשימות אחרות, כי לא תמיד ניתן לקבוע במשימת חיזוי (למשל מהסוג הזה) מה יהיה החיזוי האופטימלי כי לא בהכרח יש אדם מסוים שהיה מסוגל לחזות מה היה תהייה הצריכה בדקה הבאה או אפילו מכונה סופר משוכללת.
      4. אנו מרגישים שדווקא כיוון שבחרנו (בטעות, כך רצה הגורל) Dataset שהיה לנו מאוד מאוד קשה לשפר כל פעם, דווקא בגלל זה באמת יצא לנו לחקור לעומק את הנושא ולנסות למצוא ולקרוא על כל שיטה אפשרית שתוכל לעזור לנו, שבסופו של דבר מטרת התרגיל היא ללמוד ולכן אנו מאוד מרוצים בהקשר הזה על ההתנסות, אך נשמח לשתף באותה נשימה התסכול הרב של ניסיוננו לשפר את המודל ללא ההצלחה, לא משנה מה ניסינו.
      5. נושאים חדשים שלמדנו:
* שימוש בKfold (שהיה חשוב מאוד להליך העבודה שלנו)
* שימוש ומשמעות השימוש בפונקציות אקטיבציה שונות.
* תיקוף הData – לפעמים יש צורך לזרוק Data כיוון שהוא לא סתם לא תורם, אלא מזיק ומוסיף רעש לבן למודל ומשבש את התוצאות.
* שימוש בשיטות נרמול שונות והשפעתן על המודל.

**נמשיך לחלק הבא:**

חלק ג' – Category embeddings