به نام خدا

1928
K. N. Toosi University of Technology

دانشگاه صنعتی خواجه نصیرالدین طوسی

دانشکده برق

مبانی هوشند سازی

گزارش تمرین شماره 1

سجادرجبی باغستان

40005393

استاد : جناب دکتر مهدی علیاری

بهمن 1403

فهرست مطالب

سوالات

سوال اول

1.

- این مجموعه داده شامل اطلاعات 10,000 مشتری است که از خدمات کارت اعتباری یک بانک استفاده می‌کنند. داده‌ها شامل 18 ویژگی هستند که مشخصات مختلفی از مشتریان مانند **سن، حقوق، وضعیت تأهل، سقف اعتبار کارت اعتباری، نوع کارت اعتباری و ...**را ارائه می‌دهند. هدف اصلی این داده‌ها پیش‌بینی مشتریانی است که احتمال دارد خدمات کارت اعتباری خود را ترک کنند .یکی از چالش‌های اصلی این مجموعه داده، **عدم توازن در برچسب‌ها** است؛ تنها **16.07**٪ از مشتریان در گذشته خدمات کارت اعتباری خود را ترک کرده‌اند. این موضوع باعث می‌شود مدل‌های یادگیری ماشین برای تشخیص مشتریان در معرض ترک، نیاز به دقت بیشتری داشته باشند. این داده برای تحلیل رفتار مشتریان و اتخاذ استراتژی‌های مناسب جهت کاهش نرخ ترک مشتریان استفاده می‌شود.
- این مجموعه دیتا دارای 23 ویژگی است که عبارت است از:

i. CLIENTNUM: شماره مشتری
ii. Attrition_Flag: پرچم ترک خدمت (وضعیت ترک مشتری)
iii. Customer_Age: سن مشتری
iv. Gender: جنسیت
v. Dependent_count: تعداد وابستگان
vi. Education_Level: سطح تحصیلات
vii. Marital_Status: وضعیت تأهل
viii. Income_Category: دسته‌بندی درآمد
ix. Card_Category: دسته‌بندی کارت
x. Months_on_book: تعداد ماه‌های فعال در سیستم
xi. Total_Relationship_Count: تعداد روابط کل (مشتری با بانک)
xii. Months_Inactive_12_mon: تعداد ماه‌های غیرفعال در ۱۲ ماه گذشته
xiii. Contacts_Count_12_mon: تعداد تماس‌ها در ۱۲ ماه گذشته
xiv. Credit_Limit: سقف اعتبار
xv. Total_Revolving_Bal: مجموع مانده موجودی چرخشی
xvi. Avg_Open_To_Buy: میانگین اعتبار باز به خرید
xvii. Total_Amt_Chng_Q4_Q1: تغییرات کل مبلغ در چهارم و اول فصل
xviii. Total_Trans_Amt: مجموع مبلغ تراکنش‌ها
xix. Total_Trans_Ct: مجموع تعداد تراکنش‌ها

xx. Total_Ct_Chng_Q4_Q1: تغییرات تعداد تراکنش‌ها در چهارم و اول فصل

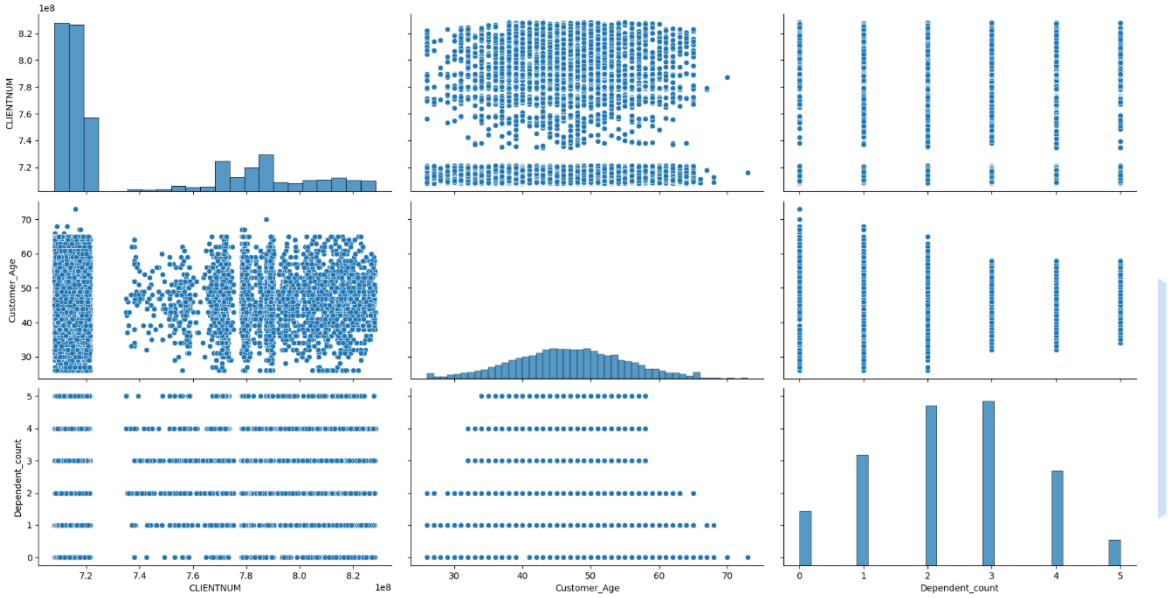
xxi. Avg_Utilization_Ratio: نسبت استفاده متوسط

xxii. Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_1: طبقه‌بندی بیز نادرست بر اساس پرچم ترک خدمت، دسته‌بندی کارت، تعداد تماس‌ها در ۱۲ ماه گذشته، تعداد وابستگان، سطح تحصیلات، تعداد ماه‌های غیرفعال در ۱۲ ماه گذشته (مدل 1)

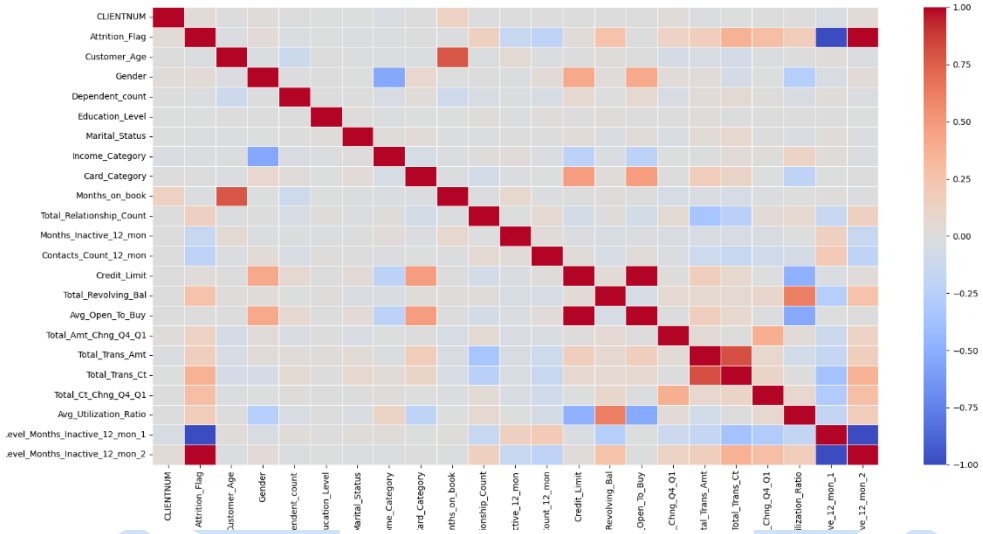
xxiii. Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_2: طبقه‌بندی بیز نادرست بر اساس پرچم ترک خدمت، دسته‌بندی کارت، تعداد تماس‌ها در ۱۲ ماه گذشته، تعداد وابستگان، سطح تحصیلات، تعداد ماه‌های غیرفعال در ۱۲ ماه گذشته (مدل 2

- این مجموعه دیتا دارای 127,10 سطر و 23 ستون است که مجموعا تعداد آن برابر میشود با 232,921.

2. Pair plot



Heat map .3



با توجه به اسم دو ستون آخر (مشخص است داده مصنوعی و ساخته شده است) و نمودار heat map و همبستگی این فیچر ها و تارگت، باید این فیچر ها حذف شوند.(این نکته که قبل از هر عملی این فیچر ها از دیتا حذف شود در توضیحات دیتا آمده است)

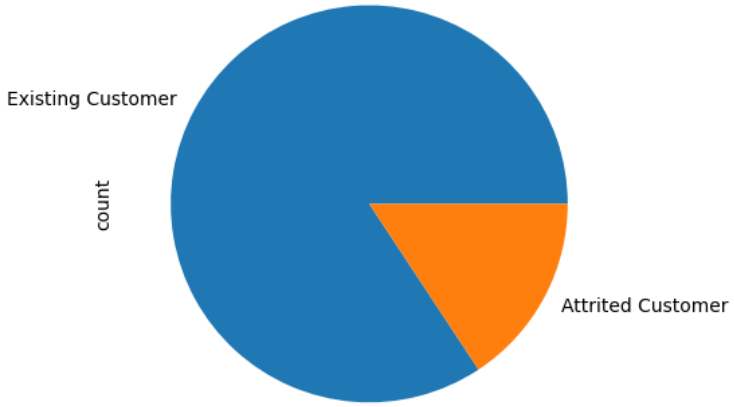
.4

بله دیتا دارای دیتا Nan (‘Unknown’) است.

.5

- این فیچر دارای دو کلاس مختلف است {"Attrited Customer"، "Existing Customer"}

- Pie plot



- بله دیتا نا متوازن است.

کلاس‌های نامتوازن مشکلی شناخته‌شده است که به‌ویژه در مسائل طبقه‌بندی (classification) رخ می‌دهد، زمانی که نسبت داده‌های هر کلاس در مجموعه داده نابرابر باشد.
آموزش مدل در این شرایط دشوارتر می‌شود، چون دقت معمولی دیگر معیار قابل‌اعتمادی برای اندازه‌گیری عملکرد مدل نیست. اگر تعداد داده‌های مربوط به کلاس اقلیت بسیار کم باشد، ممکن است در طول آموزش کاملاً نادیده گرفته شوند.

توزیع نامتعادل کلاس‌ها مشکل جدی ایجاد می‌کنند، به صورتی که حتی بهترین الگوریتم‌های یادگیری ماشین زمانی عملکرد درستی دارند که تعداد نمونه‌ها در هر کلاس تقریباً برابر باشد.
در شرایط نامتعادلی دیتا ها، مدل ممکن است به‌ظاهر دقت بالایی نشان دهد، اما اکثرا این دقت اشتباه هست چون بیشتر پیش‌بینی‌ها به نفع کلاس با تعداد بیشتر داده انجام می شود.

از تکنیک‌های رایج برای مقابله با مجموعه‌داده‌های نامتوازن، **under-sampling** و**over-sampling** است.
(over-sampling)، نمونه‌های مصنوعی با توجه به ویژگی های کلاس اقلیت ساخته و اضافه میشوند.
(under-sampling)، نمونه‌هایی از کلاس اکثریت حذف می شنود تا تعداد دیتا در کلاس ها برابر شوند.

در این سوال به دلیل ماهیت باینری بودن کلاس ها ساخت دیتا مصنوعی دشوار یا ناممکن است پس از روش under-sampling استفاده میکنیم.

با توجه به برابر شدن دیتا ها قطعا بر آموزش درست مدل تاثیر دارد به صورتی که مدل هر دو کلاس را به خوبی یاد خواهد گرفت.

- در مورد این موضوع که چه زمانی برای انجام این کار مناسب است میتوان گفت

زمانی که بخواهیم در دیتا ها تغییراتی ایجاد کنیم مثل نرمال سازی یا استفاده از رمزنگار ها برای دیتا های غیر عددی و یا متوازن سازی داده ها و ... تمامی این کار ها باید قبل از قسمت کردن دیتا ها به سه بخش train, test, val انجام شود تا مطمئن باشیم که داده ها به صورت عادلانه بین این یه بحش تقسیم شده است و هر سه بخش دارای همه نوع دیتا هست.

.6

- نتایج به دست آمده بدون متوازن سازی دیتا

```
Accuracy on Test Set: 96.52%
Classification Report:
              precision    recall  f1-score   support

           0       0.93      0.85      0.89       169
           1       0.97      0.99      0.98       894

    accuracy                           0.97      1063
   macro avg       0.95      0.92      0.93      1063
weighted avg       0.96      0.97      0.96      1063

Confusion Matrix:
                    predict class 0  predict class 1
ground truth class 0              143               26
ground truth class 1               11              883
```

- **دقت کلی (Accuracy)** مدل شده **96.52%** است که خیلی خوب به نظر می‌رسه، ولی چون داده‌ها نامتوازن است، این عدد احتمالا درست نیست.

- **گزارش طبقه‌بندی** :(Classification Report)

1. برای کلاس **0**(کلاس اقلیت) مدل **93%دقت (Precision)** داره، یعنی وقتی می‌گه "کلاس 0" هست، تا حد زیادی درست می‌گه. ولی **85% Recall**داریم، یعنی از کل مواردی که واقعاً کلاس 0 بودن، فقط 85٪ رو درست شناسایی کرده. پس یه مقدار از نمونه‌های کلاس 0 رو از دست داده.
2. برای کلاس **1**(کلاس اکثریت) دقت (Precision) و Recall هر دو نزدیک **97-99%** هستن، یعنی تقریباً همه نمونه‌های کلاس اکثریت رو درست شناسایی کرده و به درستی این کلاس را یادگرفته است.

- **ماتریس درهم‌ریختگی:(Confusion Matrix)**

1. کلاس 0**:**

  - 143تا رو درست تشخیص داده.(True Negative)
  - 26تا به اشتباه نسبت داده شده به کلاس 1.(False Negative)

2. کلاس 1**:**

  - 883تا درست تشخیص داده.(True Positive)
  - 11تا به اشتباه نسبت داده شده به کلاس 0 .(False Positive)

مدل داره بیشتر روی کلاس اکثریت (کلاس 1) تمرکز می‌کنه و خیلی خوب هم توی شناسایی اون عمل می‌کنه. ولی برای کلاس اقلیت (کلاس 0)، هنوز ضعف‌هایی داره .

- نتایج به دست آمده بعد از متوازن سازی دیتا با روش آندرسمپلینگ

```
Accuracy on Test Set: 95.51%
Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.94      0.96       177
           1       0.94      0.97      0.95       157

    accuracy                           0.96       334
   macro avg       0.95      0.96      0.95       334
weighted avg       0.96      0.96      0.96       334

Confusion Matrix:
                    predict class 0  predict class 1
ground truth class 0              167               10
ground truth class 1                5              152
```

- **دقت کلی :(Accuracy)** اینجا شده **95.51%** که همچنان عدد خوبی است. اما مهم‌تر از دقت کلی، نحوه عملکرد روی هر کلاس هست.
- در **گزارش طبقه‌بندی**:(Classification Report)

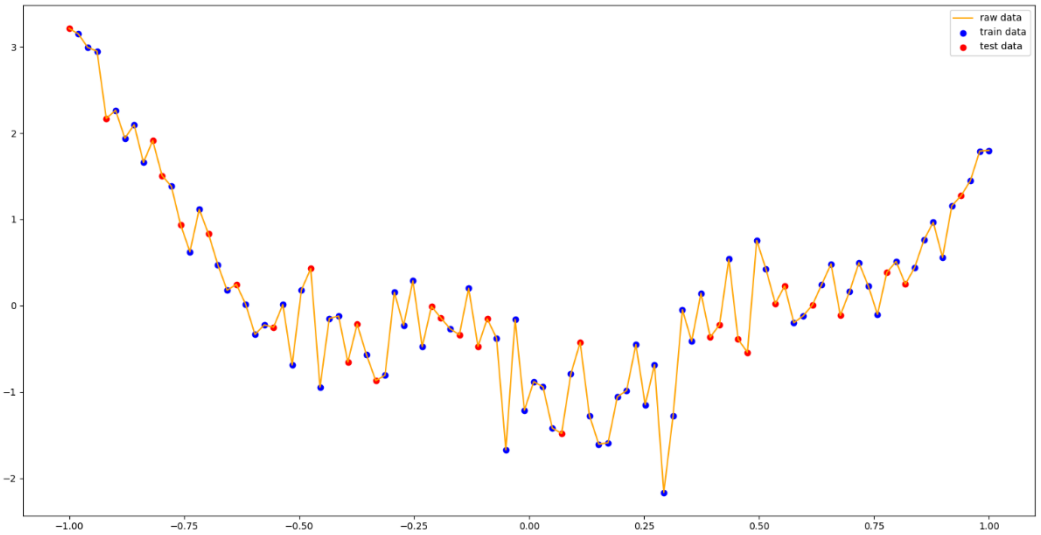
- **کلاس 0** :دقت (Precision) شده **97%**، یعنی دقت مدل در انتخاب کلاس صفر٪97 است و همچنین Recall هم **94%**هست، یعنی از کل نمونه‌های کلاس 0، ٪94 رو درست شناسایی کرده.
- **کلاس 1** Precision :شده **94%**، یعنی توی تشخیص "کلاس 1" هم عملکردش خوبه و همچنین Recall **97%**، که یعنی تقریباً همه نمونه‌های کلاس 1 رو درست شناسایی کرده.
- امتیازات F1 برای هر دو کلاس نزدیکه، یعنی مدل تعادل خوبی بین Precision و Recallبرقرار کرده.

- **ماتریس درهم‌ریختگی:(Confusion Matrix)**
  - **کلاس 0:**
    - 167تا درست شناسایی شده.(True Negative)
    - 10تا به اشتباه نسبت داده شده به کلاس 1 .(False Negative)
  - کلاس 1**:**
    - 152تا درست شناسایی شده.(True Positive)
    - 5تا به اشتباه نسبت داده شده به کلاس0 .(False Positive)

بعد از آندرسمپلینگ، مدل خیلی بهتر روی هر دو کلاس کار می‌کنه. دقت و Recall برای هر دو کلاس بالاست و مدل دیگه مثل قبل به سمت کلاس اکثریت (کلاس 1) متمایل نیست.

سوال دوم

1. نمایش داده تقسیم شده



.2

- میانگین مربعات خطا(Mean Squared Error - MSE)

فرمول:

$$MSE = \sum_{i=1}^{n} \frac{1}{n} (\hat{y}_i - y_i)^2$$

- **توضیح:**
  این معیار میانگین مجذور اختلاف بین مقادیر واقعی (y) و مقادیر پیش‌بینی‌شده (y^) را محاسبه می‌کند.
- **مزیت :**حساسیت بالایی به خطاهای بزرگ دارد، بنابراین اگر مدل روی برخی نقاط عملکرد ضعیفی داشته باشد، این معیار به وضوح آن را نشان می‌دهد.
- **معایب :**به دلیل مجذور کردن خطا، خطاهای بزرگ بیش از حد تاثیر می‌گذارند.

- میانگین قدرمطلق خطا(Mean Absolute Error - MAE)

فرمول:

$$MAE = \sum_{i=1}^{n} \frac{1}{n} |\hat{y}_i - y_i|$$

- **توضیح:**
  این معیار میانگین مقدار مطلق اختلاف بین مقادیر واقعی و پیش‌بینی‌شده را محاسبه می‌کند.
- **مزیت :**کمتر تحت تاثیر خطاهای بزرگ قرار می‌گیرد و به طور مستقیم میانگین خطا را به صورت قابل تفسیر ارائه می‌دهد.
- **معایب :**به دلیل خطی بودن، ممکن است تغییرات ظریف در عملکرد مدل را به خوبی نشان ندهد.

- ضریب تعیین

فرمول:

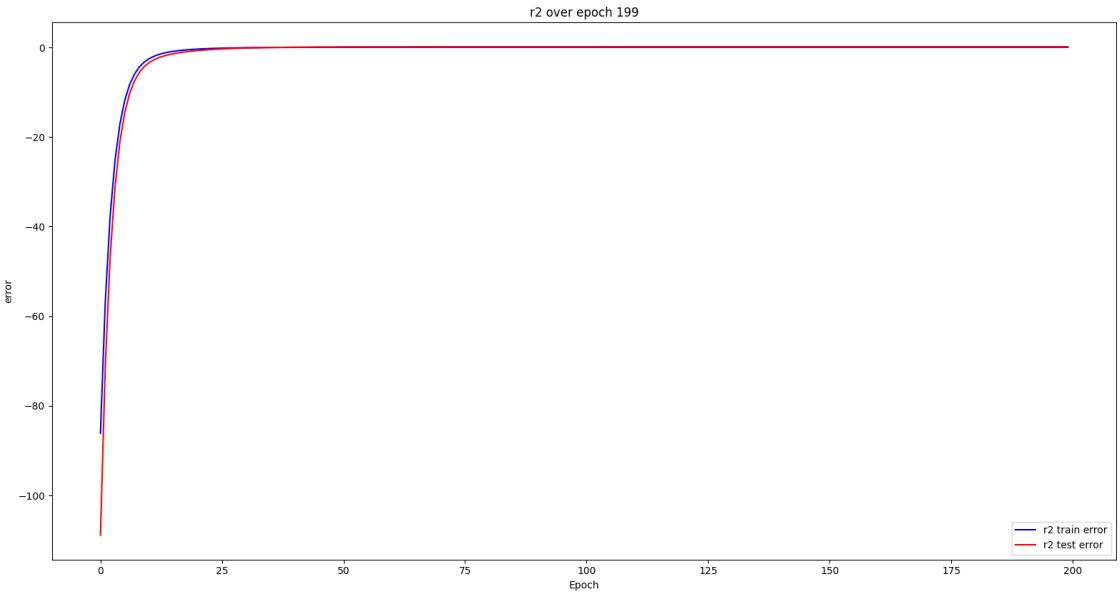
$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y - \hat{y})^2}{\sum_{i=1}^{n}(y - \bar{y})^2}$$

- **توضیح:**

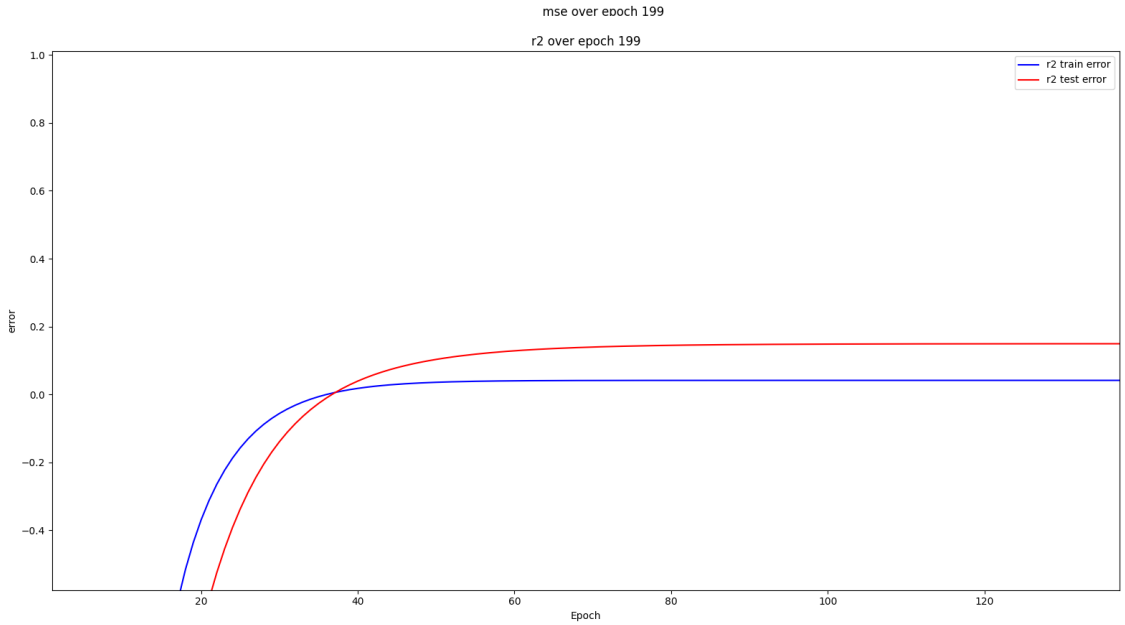
این معیار نسبت واریانس توضیح‌داده‌شده توسط مدل به واریانس کل داده‌ها را می‌سنجد.
R^2بین 0 و 1 قرار دارد؛ عدد بالاتر نشان‌دهنده مدل بهتر است.

- **مزیت :**یک معیار نسبی است که به شما امکان می‌دهد عملکرد مدل را نسبت به یک مدل پایه (مانند میانگین) بسنجید.
- **معایب :**وقتی تعداد ویژگی‌ها زیاد باشد، ممکن است R^2 به صورت کاذب بالا برود.

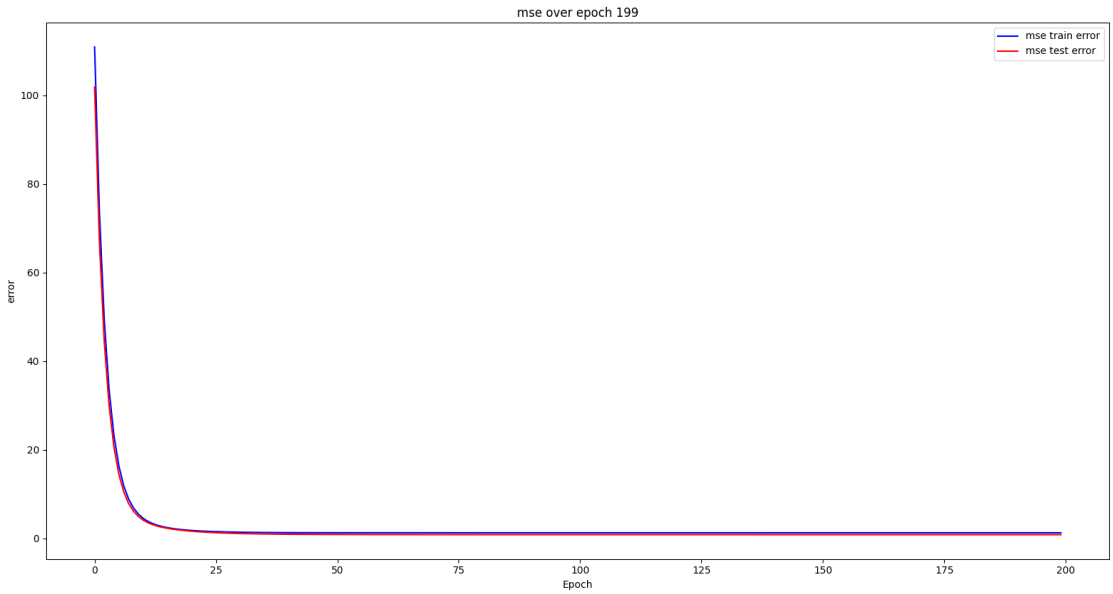
3. نتایج



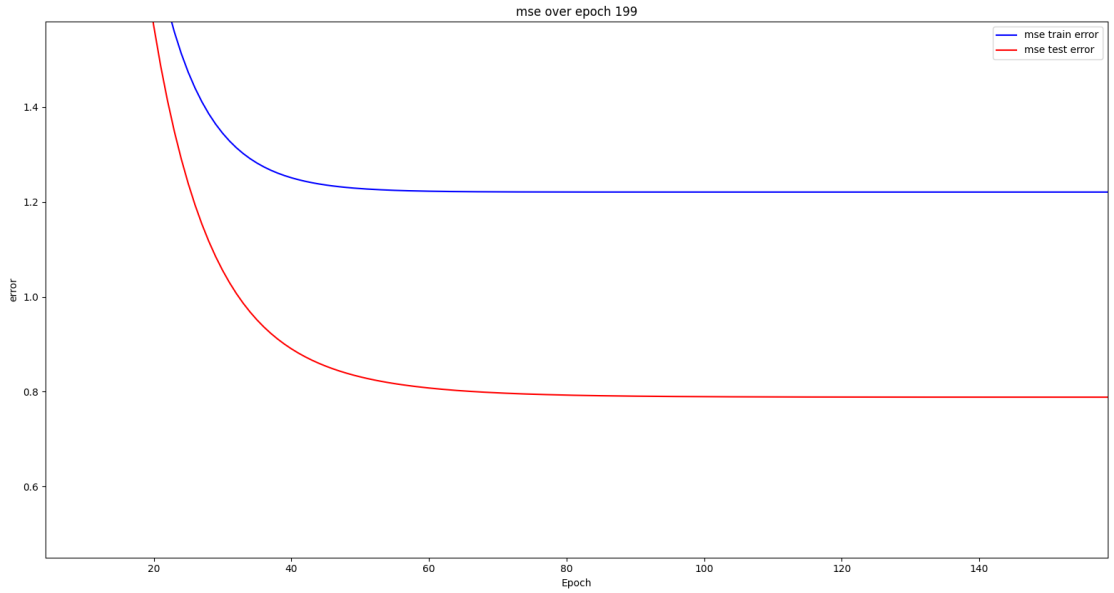
نمودار نمره ضریب تعیین



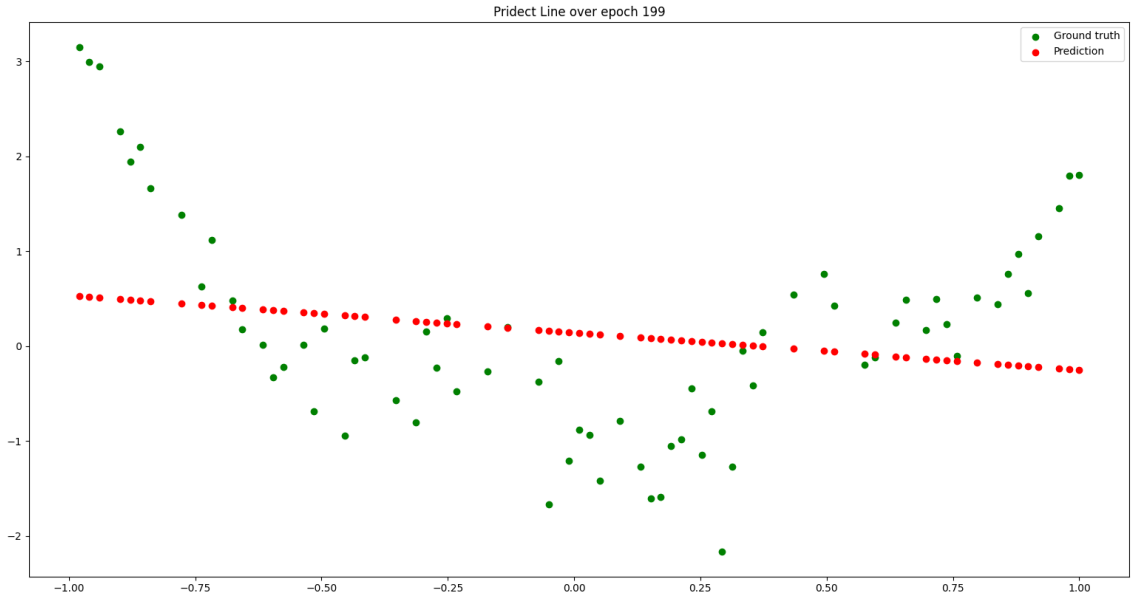
نمودار نمره ضریب تعیین



نمودار میاگین مربع خطا



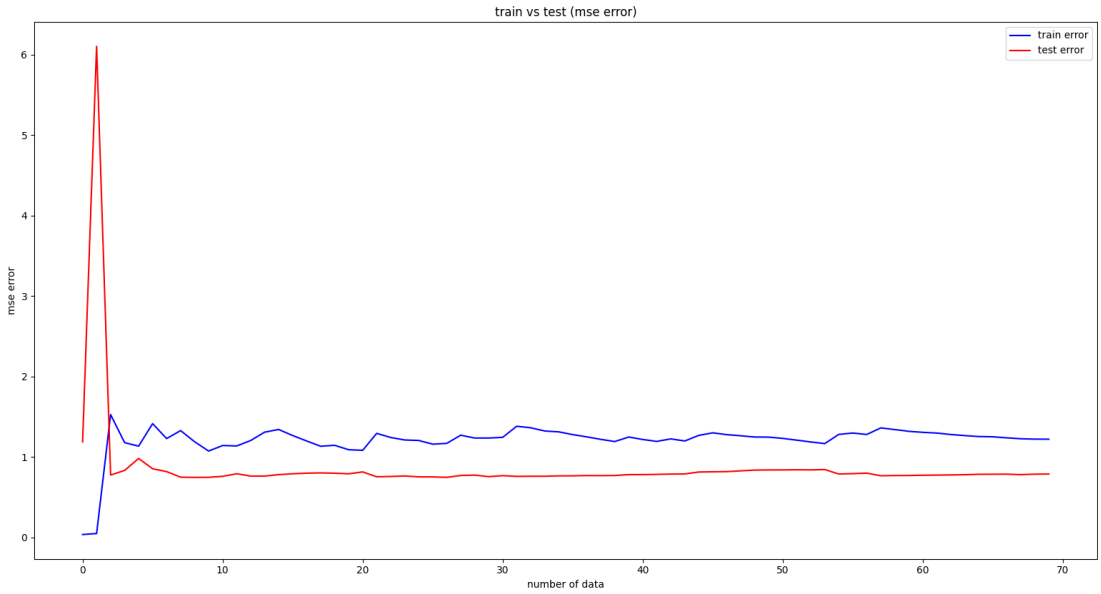
نمودار میاگین مربع خطا



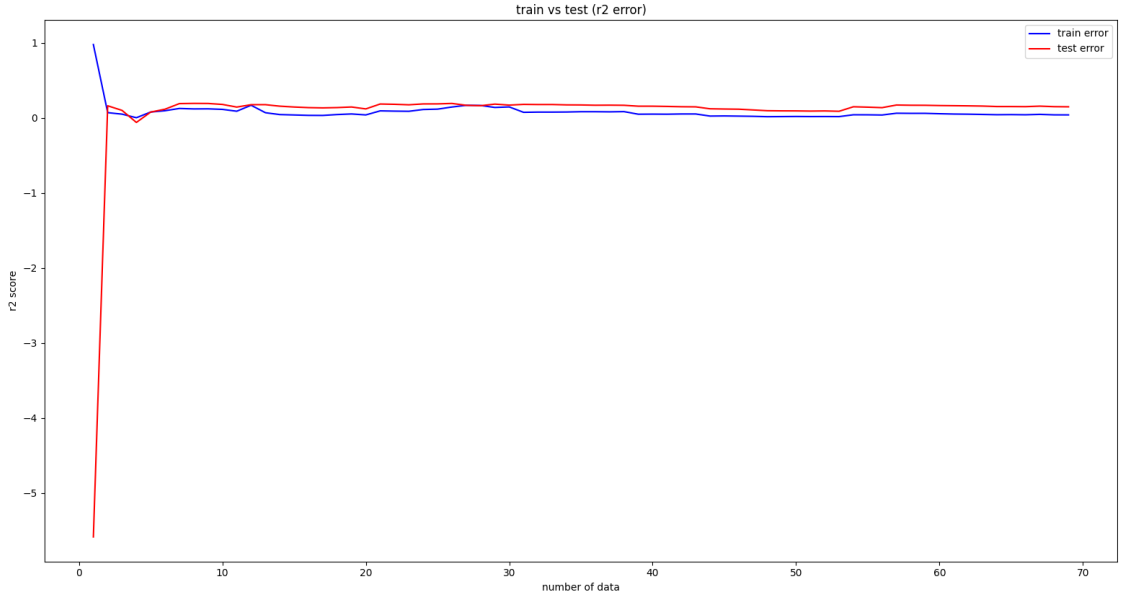
نمودار خط فیت شده روی دیتا

- با توجه به نمودار خطا و ضریب تعیین و با توجه به اینکه رگرسور یک خط ساده است، نتیجه به دست آمده قابل قبول است اما اگرنیاز به تخمین دقیق تری باشد باید درجه رگرسور افزایش پیدا کند تا خطا و نمره بهتری به دست آید.
- در حالت کلی این خط تخمین مناسبی برای تابع هدف و دیتا نیست.

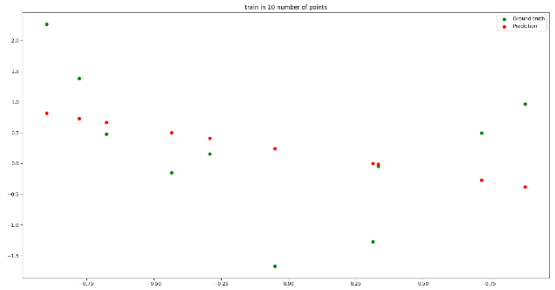
4. نتایج



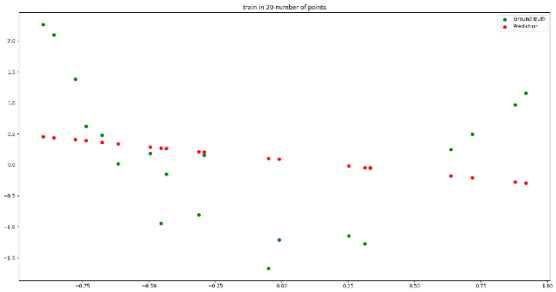
نمودار میانگین مربعات خطا بر حسب تعداد دیتا



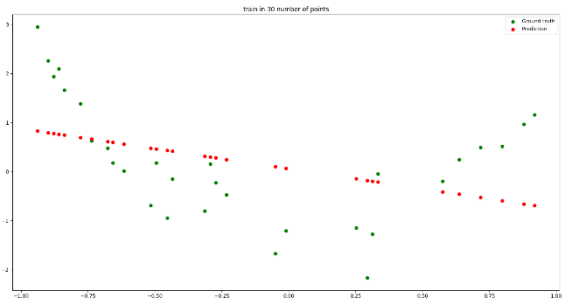
نمودار ضریب تعیین بر حسب تعداد دیتا

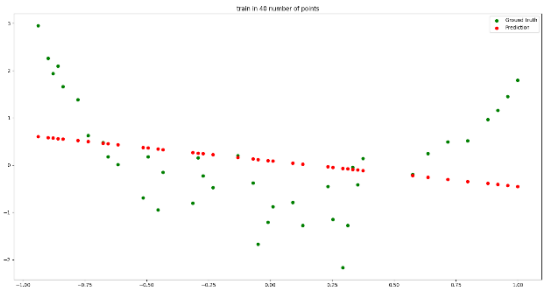


1

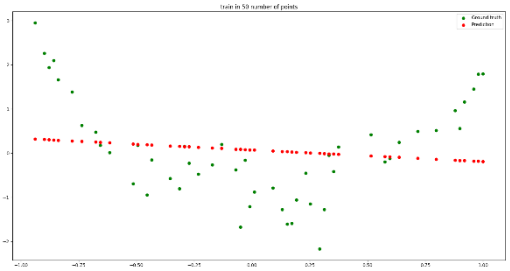


2

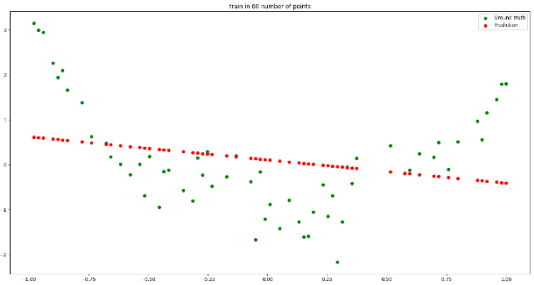




3

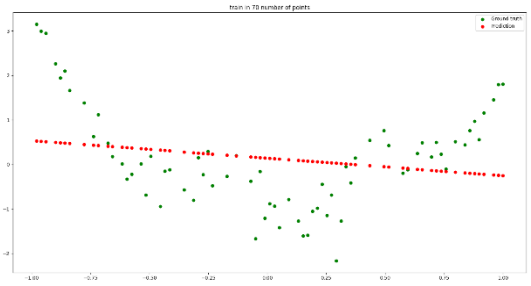


4



5

6



7

- با توجه به نتایج به دست امده، رسیدن مدل به تابع هدف و به نقطه ایده آل وابستگی زیادی نسبت به مقدار دیتا موجود از هدف دارد. در این بین امکان خطا برای ناظر درصورتی که مقدار کمی دیتا وجود دارد زیاد است به طوری که اگر از یک مجموعه بزرگ دیتا تنها تعداد کمی از این دیتا برای آموزش مدل مورد استفاده قرار بگیرد این مدل به احتمال زیاد نتایج خوبی روی همان مجموعه کوچک ارائه میدهد(خطلای پایین و ضریب تعیین بالا) اما در واقعیت مدل آموزش داده شده اصلا برای مورد استفاده قرار گرفتن در مجموعه دیتا بزرگ و واقعی مناسب نیست و خطای زیادی خواهد داشت.

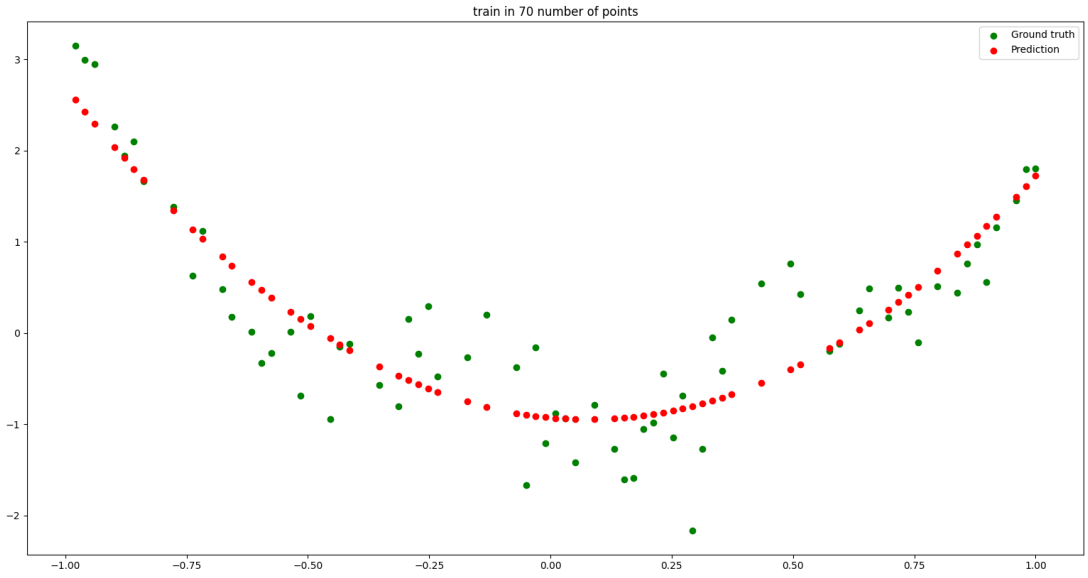
5.

- با توجه به نتایج آزمایش قبل با وجو دیتا بالا خطای مدل قطعا کاهش پیدا میکند اما ین موضوع که آیا قطعا از عملکرد انسان بهتر میشود را نمیتوان به صورت قطعی اعلام کرد و البته دقت مدل تنها به وجود دیتا زیاد بستگی ندارد.

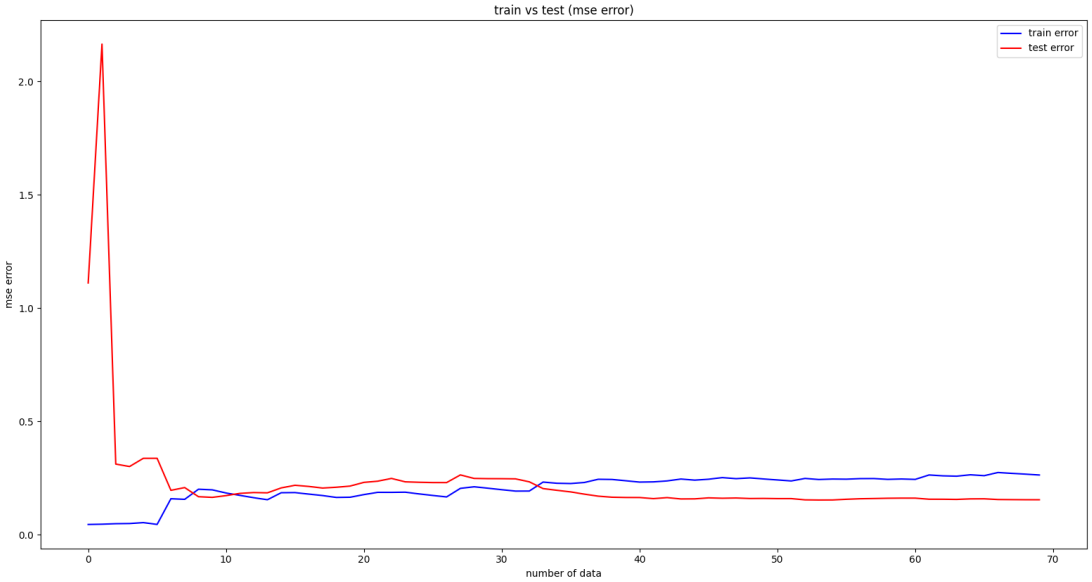
در روند آموزش یک مدل انتخاب دیتا هایی که واقعا بر روی نتیجه اثر گذار هستند، انتخاب ظرفیت مناسب برای مدل(درجه جملات)، انتخاب تابع مناسبی برای محاسبه loss و در کل انجام عملیات های پری پراسس و پست پراسس مناسب خیلی مهم است و از خطرات اشباع و اور فیت و ... جلوگیری میکند که خود باعث کاهش خطا میشود.

6.

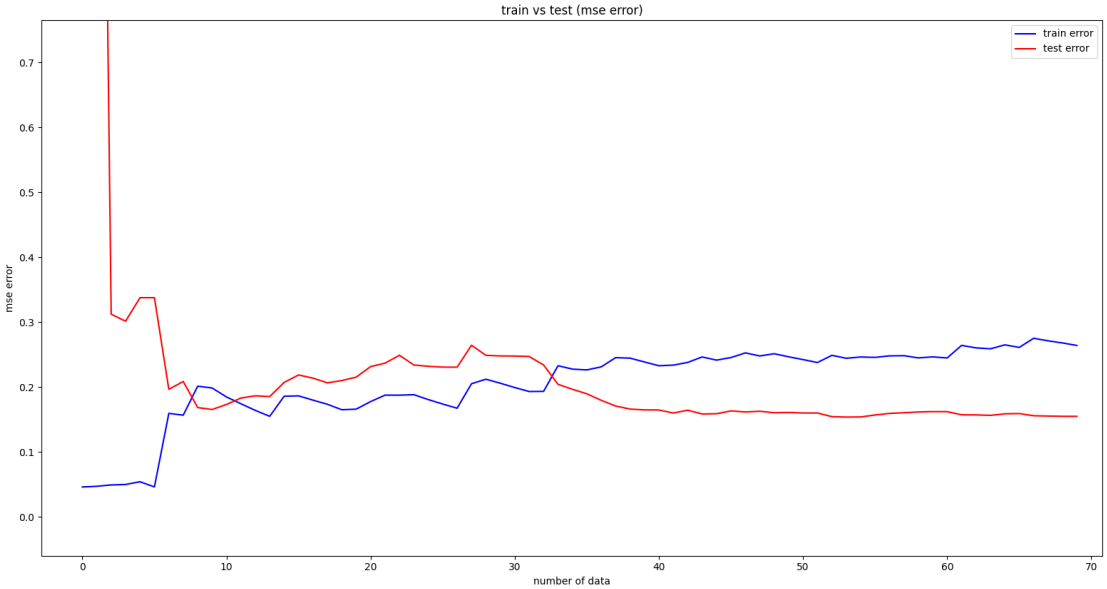
- نتایج درجه 2



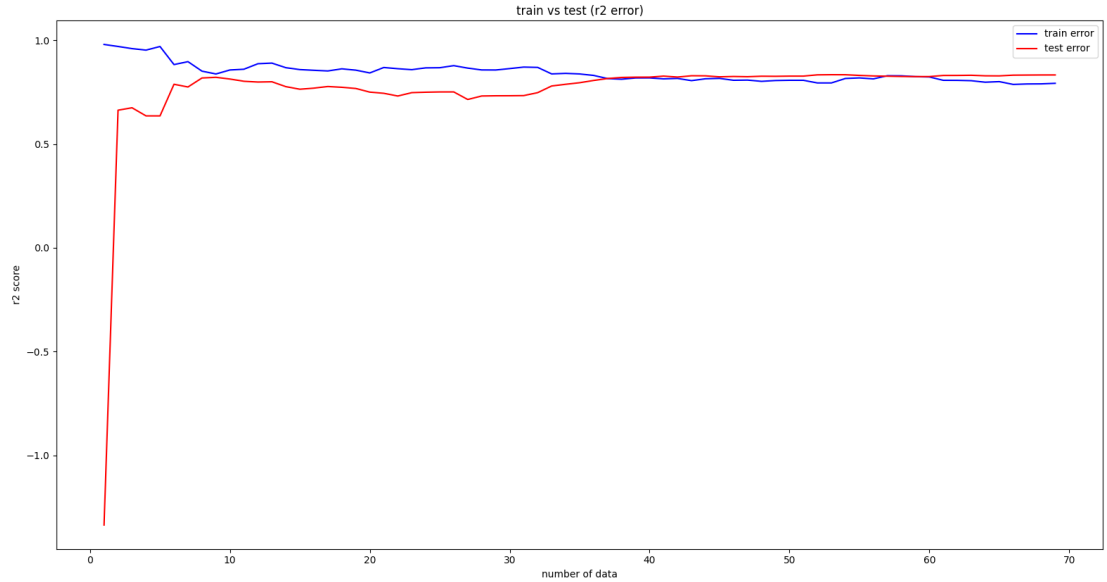
خط فیت شده بر روی دیتا



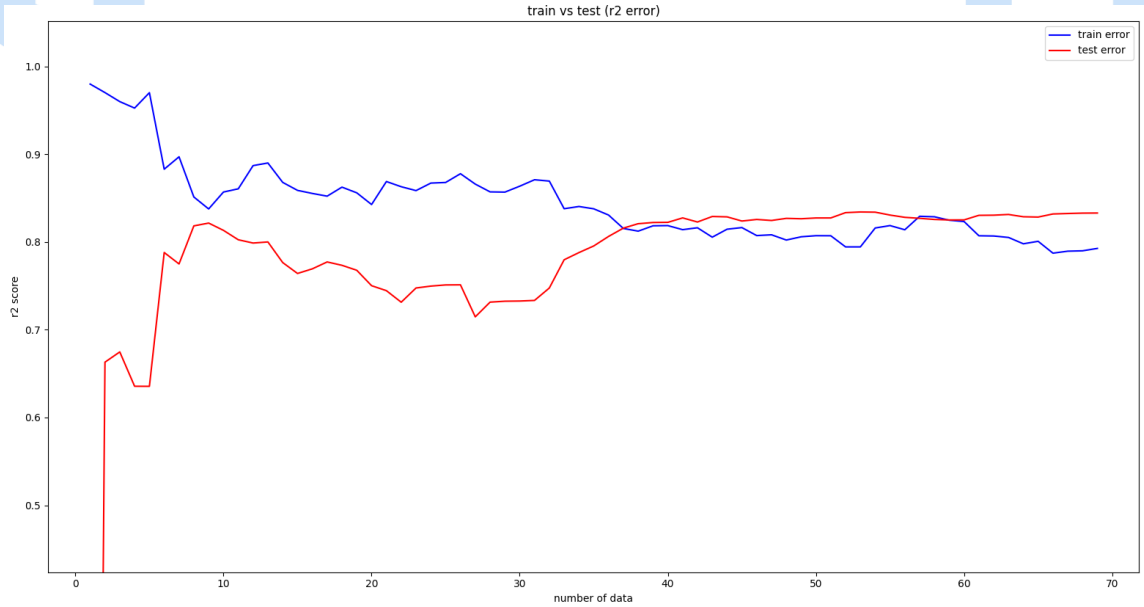
نمودار میانگین مربعات خطا



نمودار میانگین مربعات خطا

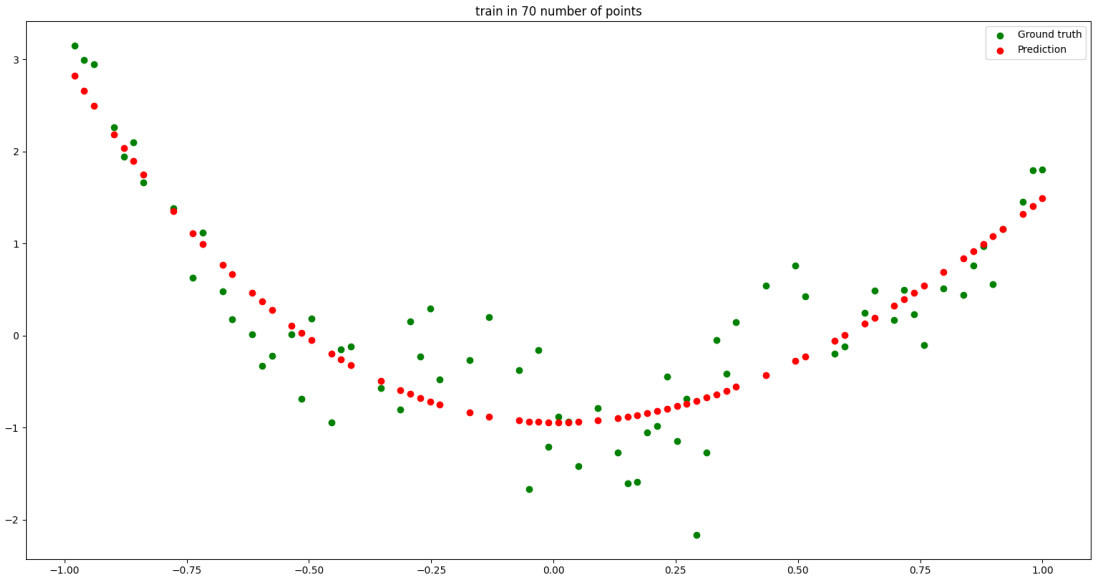


نمودار ضریب تعیین

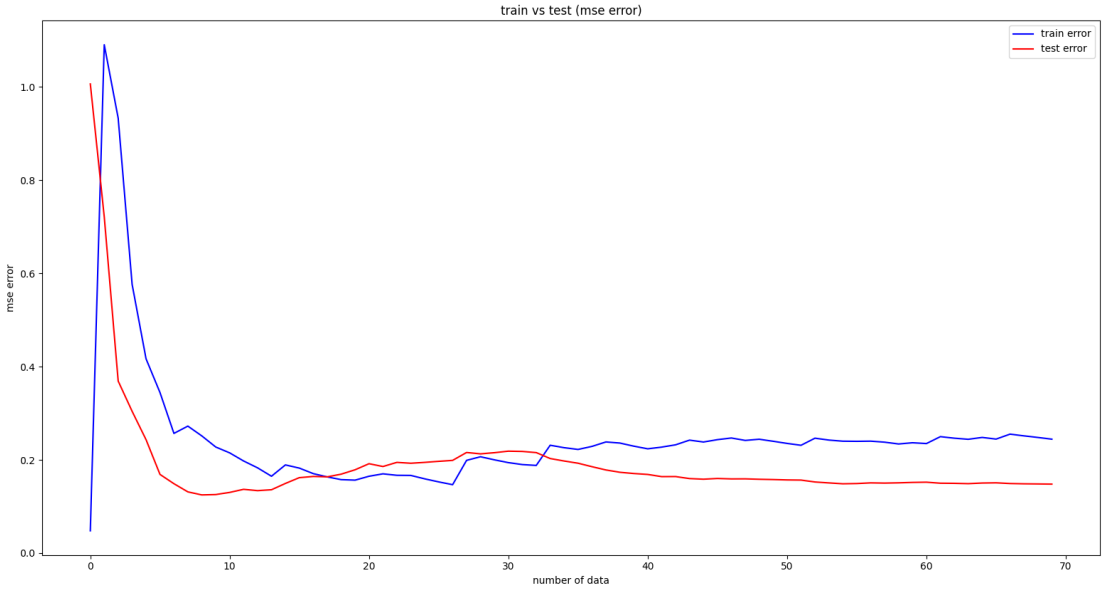


نمودار ضریب تعیین

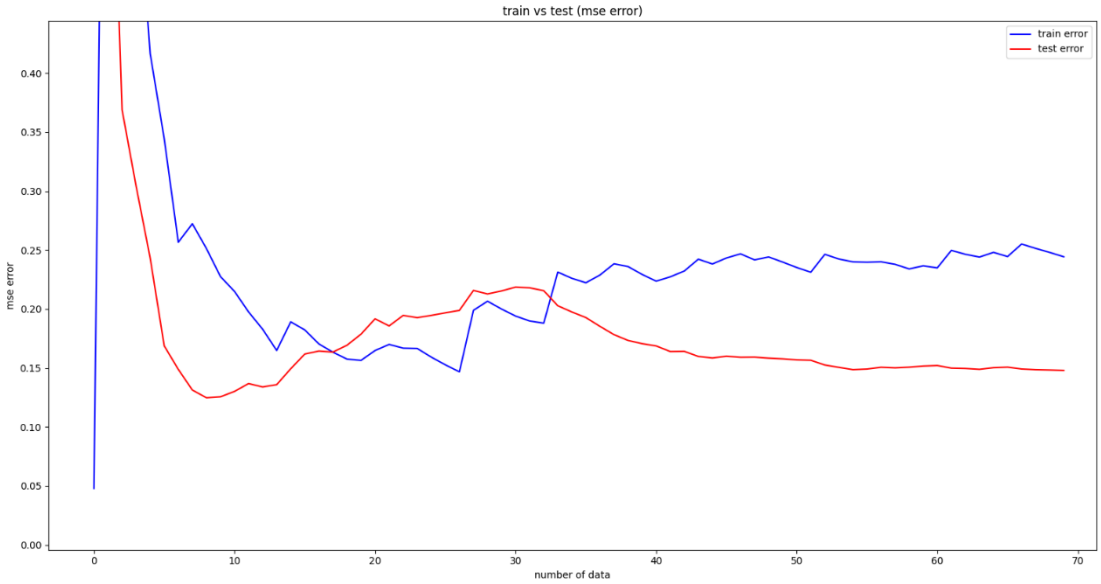
- درجه 3



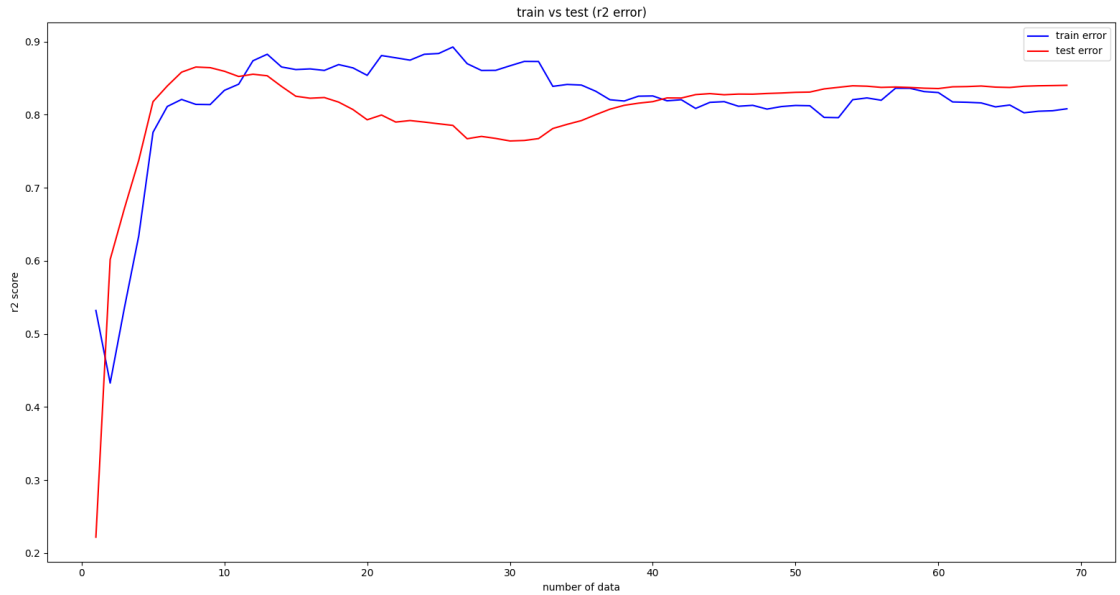
خط فیت شده بر روی دیتا



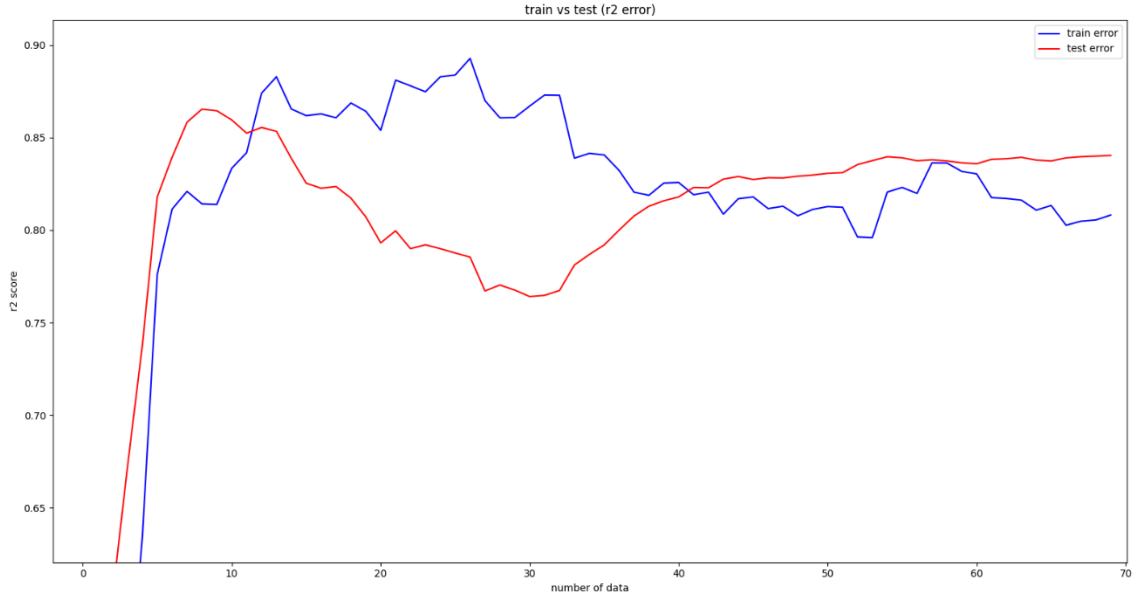
نمودار میانگین مربعات خطا



نمودار میانگین مربعات خطا

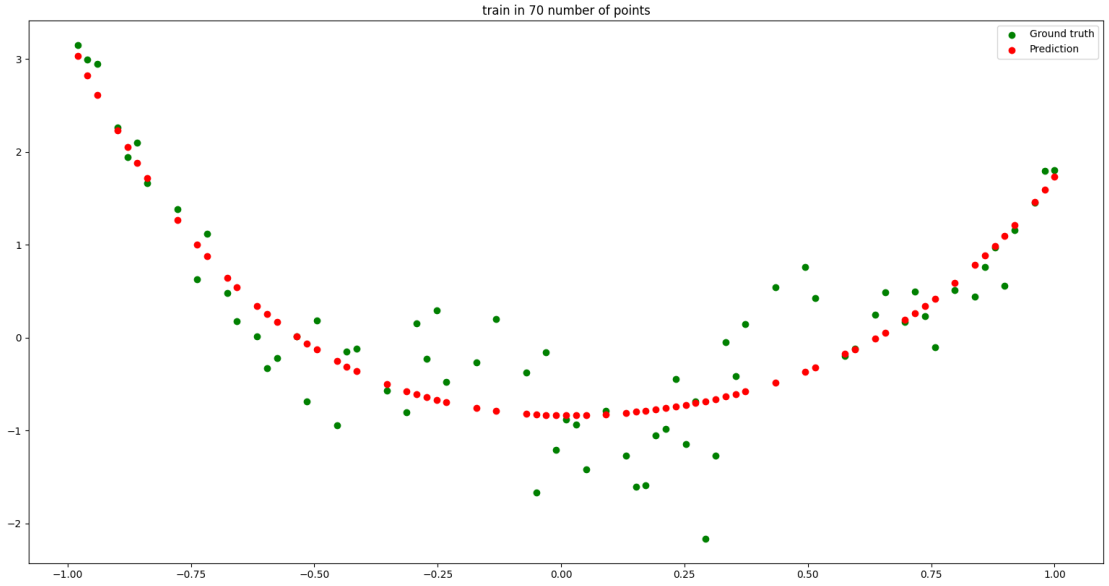


نمودار ضریب تعیین

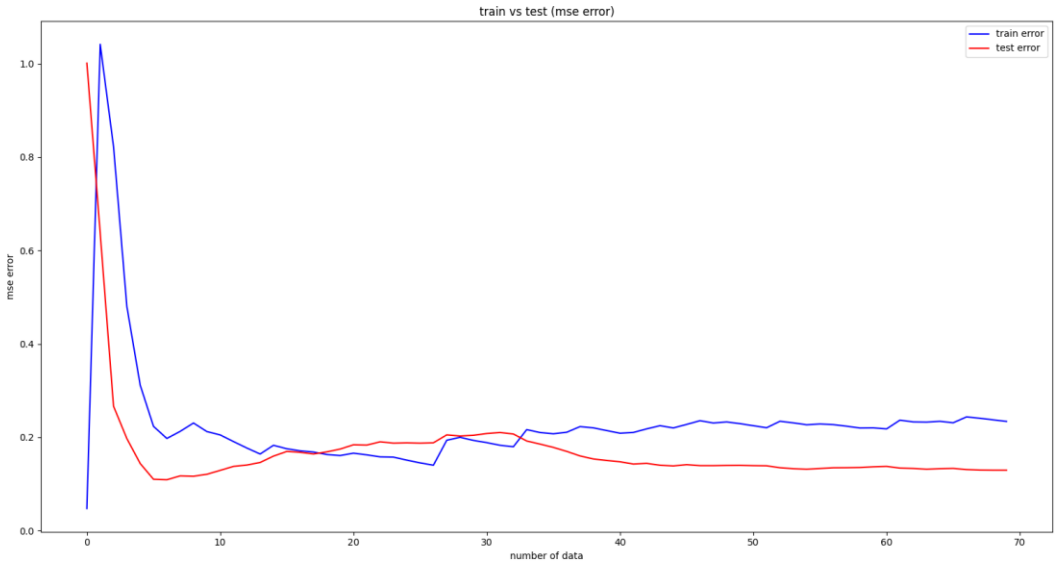


نمودار ضریب تعیین

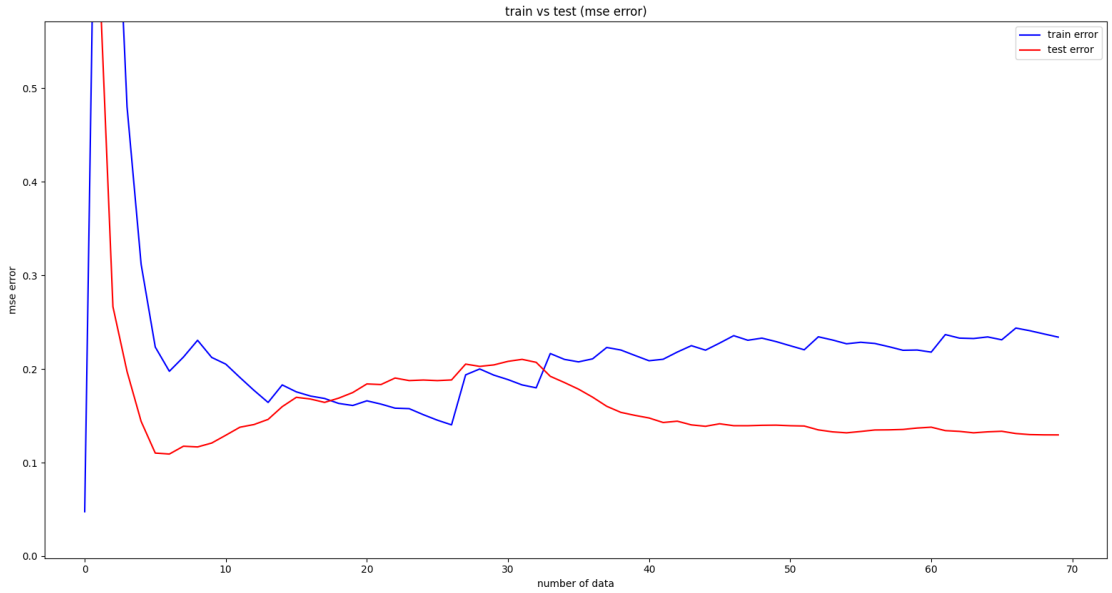
- درجه 4



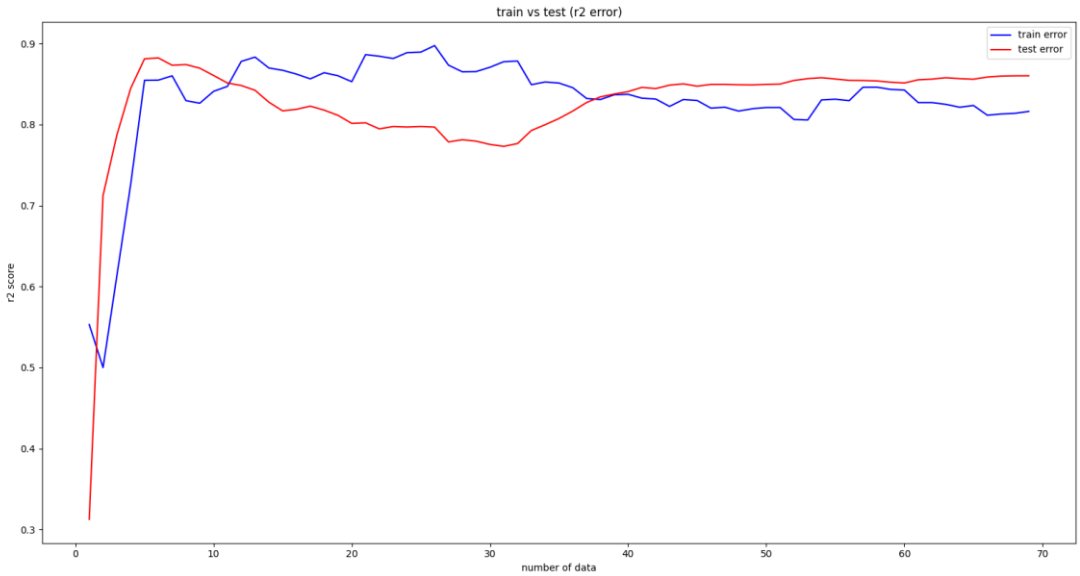
خط فیت شده بر روی دیتا



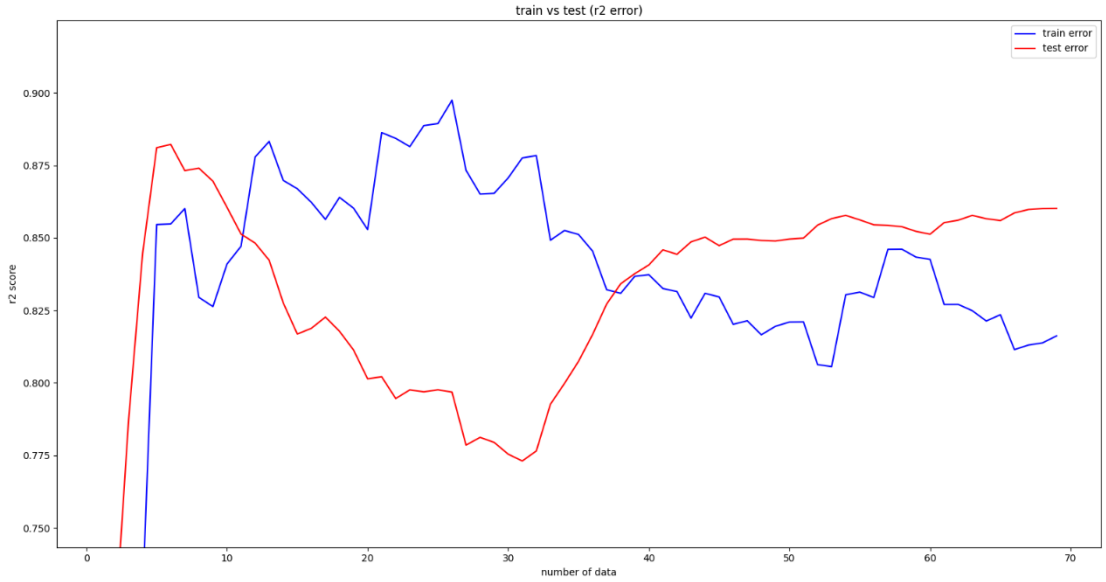
نمودار میانگین مربعات خطا



نمودار میانگین مربعات خطا

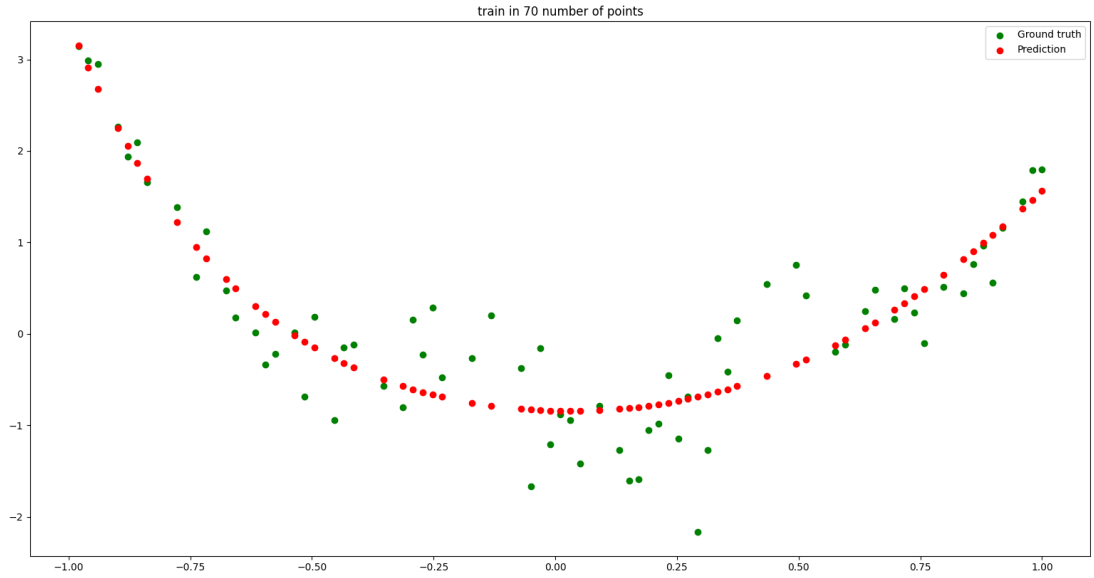


نمودار ضریب تعیین

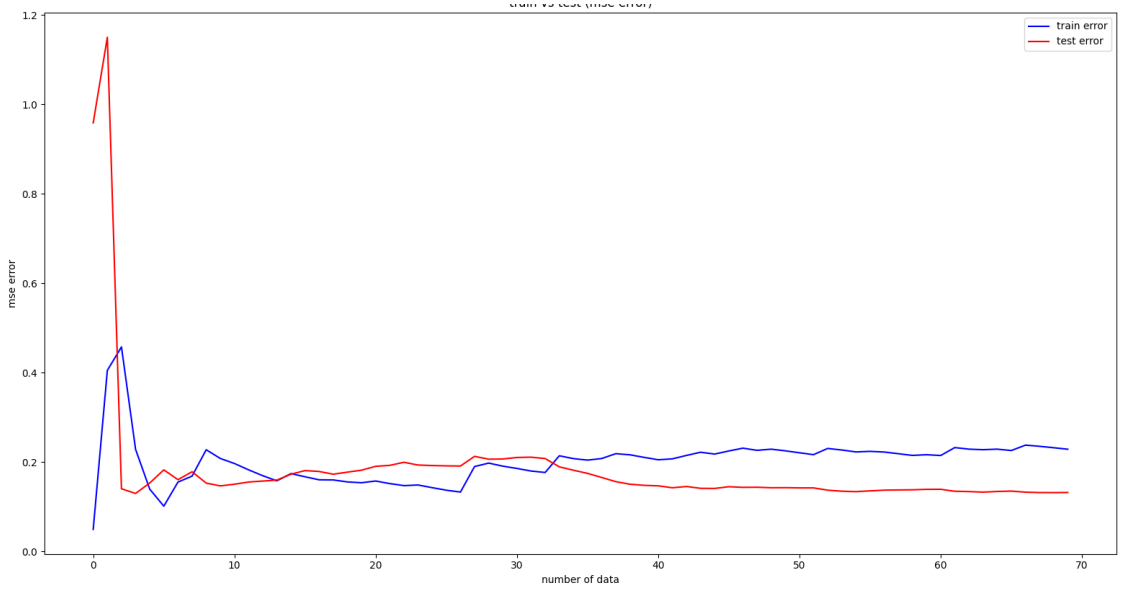


نمودار ضریب تعیین

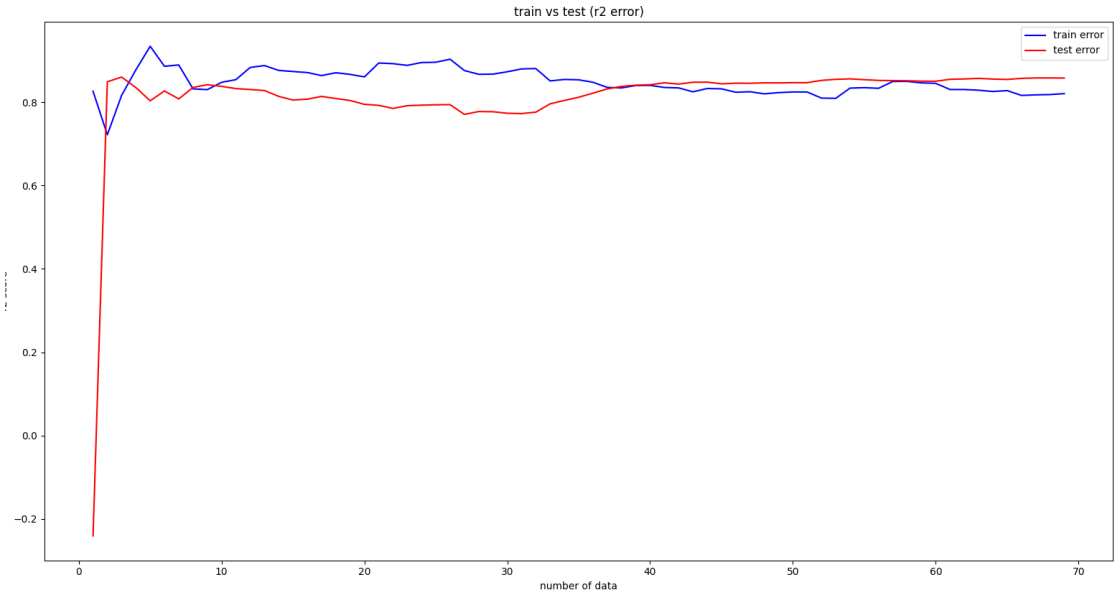
- درجه 5



خط فیت شده بر روی دیتا



نمودار میانگین مربعات خطا



نمودار ضریب تعیین

- با توجه به نمودار ها مشخص میشود که بالا بردن درجه نمیتواند همیشه خوب باشد،

در واقع باید درجه مناسب انتخاب شود نه پایین و نه بالا هر کدوم از ای دو حالت میتواند موجب آندرفیت یا اورفیت شود که حالت ایده آلی نیست.

در مورد بلا بردن درجه مدل نتنها موجب اورفیت میشود بلکه سختی محاسبات را افزایش میدهد که خود در کار های RT مشکل است.

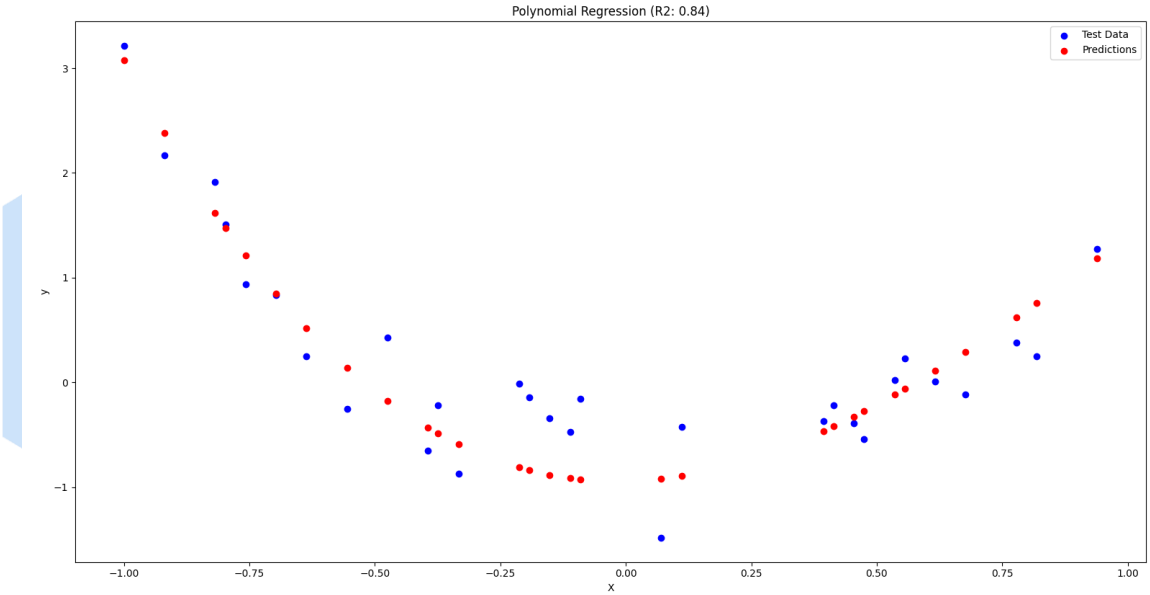
دلیل اینکه در نمودار های نمایش داده شده تابع به دست آمده در درجات بالا اورفیت نشده احتمالا تعداد epoch پایین است.

درجه مناسب برای این دیتا ها درجه 2 است که هم محاسبات حداقلی را نسبت به درجات بالا تشکیل میدهد هم از اورفیت جلوگیری میکند.

.7

1. رگرسیون چندجمله‌ای(Polynomial Regression)

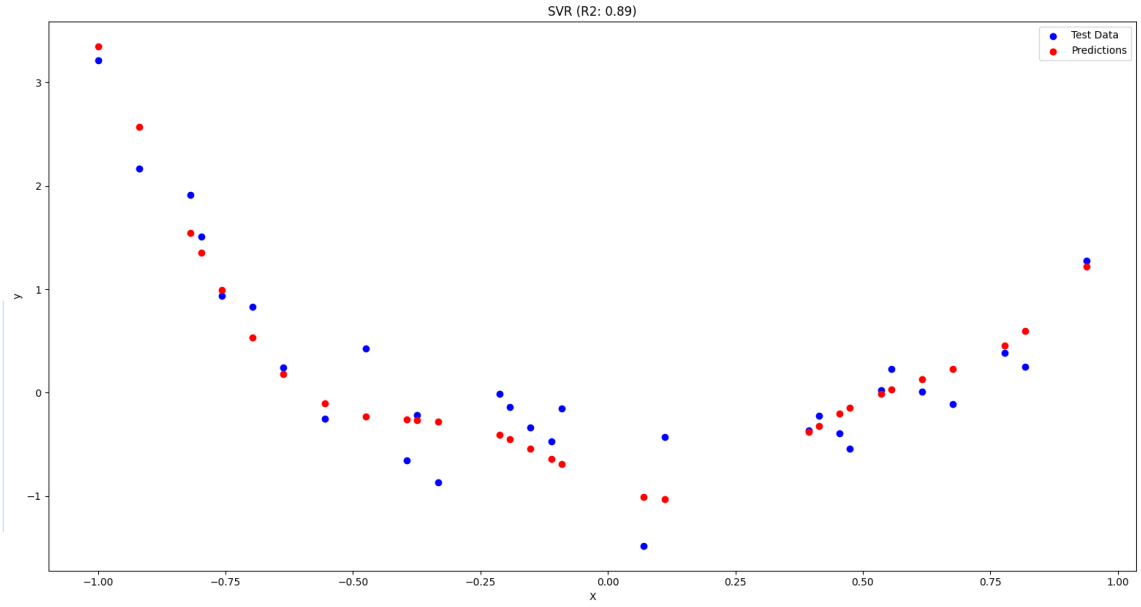
- **توضیح:**
  - این روش توسعه‌ای از رگرسیون خطی است که ویژگی‌های چندجمله‌ای (توان‌های بالاتر از متغیرهای اصلی) را به مدل اضافه می‌کند. این کار اجازه می‌دهد مدل رفتارهای غیرخطی را یاد بگیرد.
  - به عنوان مثال، برای داده‌هایی که روند آن‌ها به شکل منحنی است، این روش عملکرد بهتری نسبت به رگرسیون خطی خواهد داشت.



نتایج رگرسور چند جمله ای

2. SVR (Support Vector Regression)

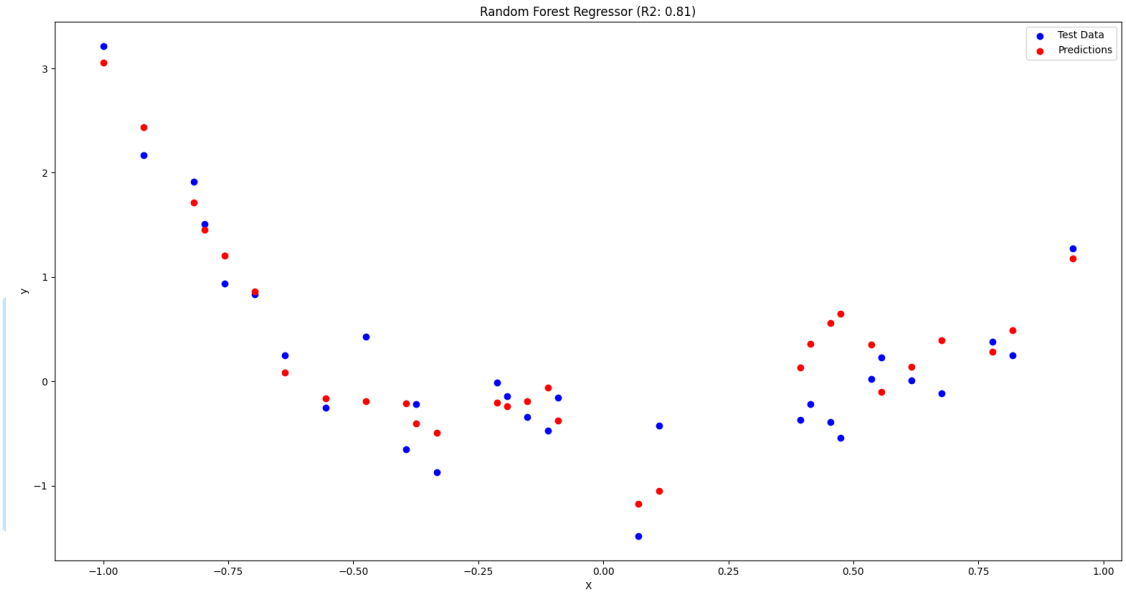
- **توضیح:**
  - یک الگوریتم مبتنی بر ماشین بردار پشتیبان است که برای رگرسیون استفاده می‌شود.
  - این الگوریتم می‌تواند روابط غیرخطی را با استفاده از کرنل‌ها مدل کند. کرنل‌های محبوب شامل (Radial Basis Function) `RBF`هستند که برای داده‌های غیرخطی مناسب است.



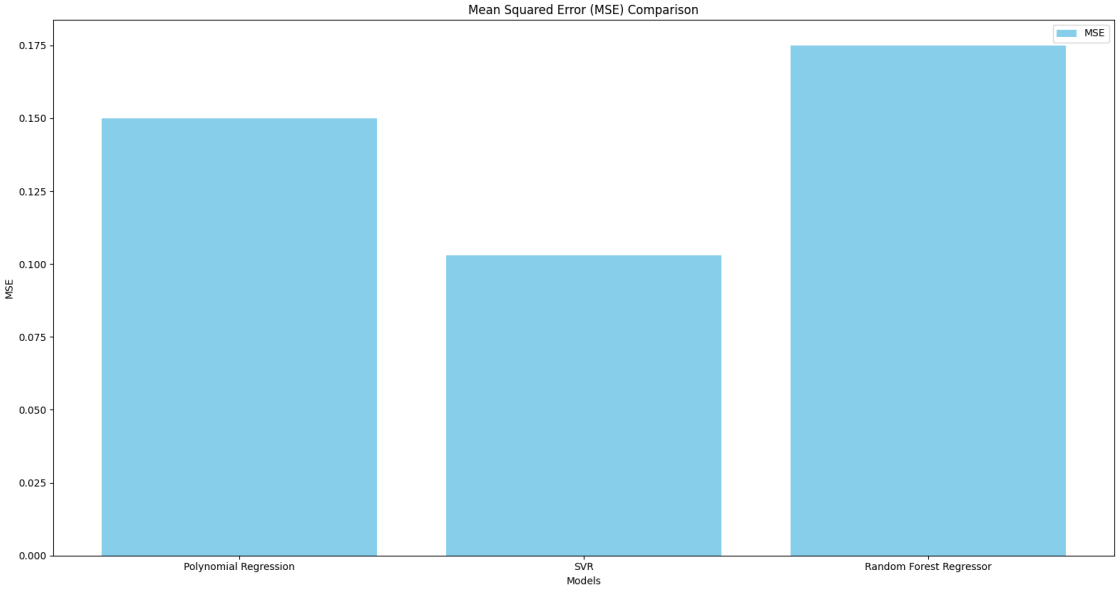
نتایج رگرسور svr

2. Random Forest Regressor

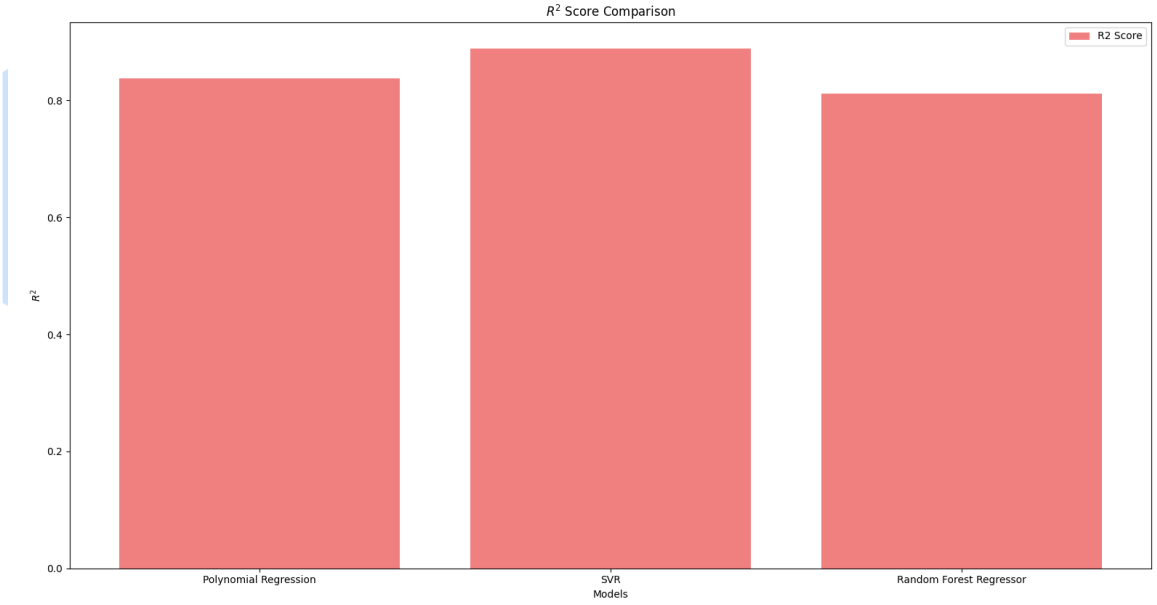
- **توضیح:**
  - یک مدل مبتنی بر مجموعه‌ای از درخت‌های تصمیم است که پیش‌بینی‌های هر درخت را ترکیب می‌کند.
  - این روش برای مدل‌سازی روابط پیچیده و غیرخطی بسیار قدرتمند است و معمولاً در مواجهه با داده‌های noisy عملکرد پایداری دارد.



نتایج رگرسور جنگل تصادفی



نمودار مقایسه میانگین مجموع مربعات خطا رگسور ها



نمودار ضریب تناسب رگرسور ها

- با توجه به نمودار میتوان نتیجه گرفت که خطای svr از دیگر رگرسور ها کمتر است اما باید در نظر داشت گاهی مقدار پایین خطا خود هشداری راجب اورفیت شدن مدل است. به نظر رگرسور چند جمله ای میتواند گزینه خوبی برای انتخاب باشد به این دلیل که هم خطای آن متوسط خطای دیگر رگرسور ها است هم ضریب تناسب آن.

با تشکر.