



دانشگاه صنعتی خواجه نصیرالدین طوسی  
دانشکده مهندسی برق - گروه مهندسی کنترل

درس بادگیری ماشین  
گزارش مینی پروژه شماره یک

نام و نام خانوادگی	علیرضا جهانی
شماره دانشجویی	۴۰۲۲۳۰۴۴
استاد درس	دکتر علیاری
تاریخ	بهمن ماه ۱۴۰۲

[https:](https://github.com/Earnoo/ML_course_HW1)

[//github.com/Earnoo/ML\\_course\\_HW1](https://github.com/Earnoo/ML_course_HW1)

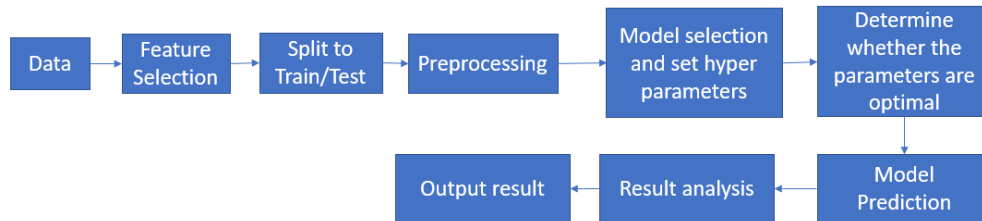
[https://colab.research.google.com/  
drive/1Z07gx\\_dtEUNMBQkHfaY77L\\_  
-SfZmbcFe?usp=sharing](https://colab.research.google.com/drive/1Z07gx_dtEUNMBQkHfaY77L_-SfZmbcFe?usp=sharing)

## فهرست مطالب

۲	۱ سوال اول
۲	۱.۱ بخش اول سوال اول
۳	۲.۱ بخش دوم سوال اول
۴	۳.۱ بخش سوم سوال اول
۵	۴.۱ بخش چهارم سوال اول
۶	۵.۱ بخش پنجم سوال اول
۷	۲ سوال دوم
۷	۱.۲ بخش اول سوال دوم
۸	۱.۱.۲ بخش آ بخش دوم سوال دوم
۸	۲.۱.۲ بخش ب بخش دوم سوال دوم
۸	۳.۱.۲ بخش ج بخش دوم سوال دوم
۸	۴.۱.۲ بخش د بخش دوم سوال دوم
۹	۲.۲ بخش سوم سوال دوم
۱۰	۳.۲ بخش چهارم سوال دوم
۱۱	۴.۲ قسمت پنجم سوال دوم
۱۳	۳ عنوان سوال سوم
۱۳	۱.۳ بخش اول سوال سوم
۱۴	۲.۳ بخش دوم سوال سوم
۱۵	۳.۳ بخش سوم سوال سوم
۱۶	۴.۳ بخش چهارم سوال سوم

## ۱ سوال اول

## ۱.۱ بخش اول سوال اول



شکل ۱: بلوک دیاگرام عملکرد مدل دسته بندی

با توجه به شکل ۱ داریم:

## ۱. Data

در اولین قدم باید دیتاست را جمع آوری کنیم.

## ۲. Selection Feature

در این قسمت باید ویژگی‌ها را انتخاب کنیم که موجب بهبود عملکرد مدل شود. در این قسمت حتی می‌توان *Feature extraction* نیز انجام داد و ویژگی‌هایی را استخراج کنیم.

## ۳. Test Train/ to Split

در این قسمت دیتاها را برای ارزیابی به ۲ و گاهی می‌توان به ۳ دسته نیز تقسیم نمود. غالباً تقسیم ۳ تایی به صورت داده تمرین و تست که به مدل داده میشوند و نتایج را روی داده ارزیابی بررسی و تحلیل میکنند. داده ارزیابی را به هیچ عنوان مدل نباید ببیند.

## ۴. Preprocessing

در این قسمت پیش پردازش‌هایی از جمله نرمالایز کردن دیتاها، پاک کردن دیتا‌هایی با اطلاعات ناقص و... انجام میشود.

## ۵. parameters hyper set and Selection Model

در این قسمت مدل دسته بندی انتخاب میشود که با توجه به نیاز میتوان از مدل‌های درخت تصمیم‌گیری تا مدل‌های *SVM* استفاده نمود. همچنین در این قسمت پارامترهایی از جمله کرنل‌ها و... موجود میباشد که میتوان آنها را نیز تنظیم نمود.

## ۶. Parameters Optimal

با توجه به روش‌های موجود میتوان بهترین پارامترها را با ایجاد حلقه روی دیتاست و پیاده‌سازی تمام حالات بهترین پارامترها با نتایج روی داده‌های *train/test* انتخاب نمود.

## ۷. prediction Model

حال میتوان بهترین مدل را ذخیره نمود و روی داده‌های ارزیابی پیاده‌سازی کرد.

## ۸. Results

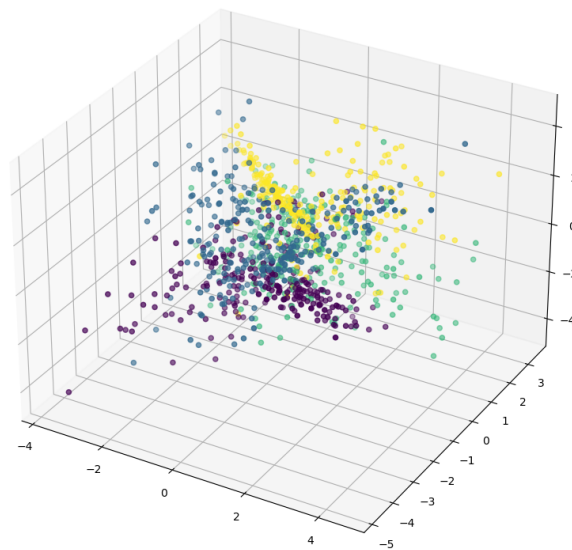
نتایج باید تحلیل شوند و بتوان نقاط ضعف و قدرت مدل را تحلیل نمود.

برای تغییر نوع طبقه بندی از حالت binary به چند کلاسه برای برخی مدل ها باید از قسمت تنظیم پارامتر ها اقدام نمود و برای برخی مدل ها نیازی نیست و با توجه به دیتای label خود مدل چند کلاسه عمل میکند.

## ۲.۱ بخش دوم سوال اول

به کمک دستور `make_classification` میتوانیم دیتاست را تولید کنیم. با توجه به دیفالت پارامتر هایی که ما تنظیمشون نکردیم و روی دیفالت هستند نباید دیتاست خیلی چالش بر انگیز باشد. برای متوجه شدن این موضوع باید مدلی را پیاده سازی کرد و نتایج را تحلیل نمود و یا به خوبی دیتا ها را نمایش دهیم و با متریک هایی که وجود دارد تحلیل و بررسی کنیم. شکل ۲ نمایش دیتای ایجاد شده با ۳ ویژگی میباشد. برای تعیین اینکه آیا دیتاست چالش برانگیز است یا نه، باید به توزیع داده ها در هر کلاس و چگونگی تداخل آن ها نگاه کنیم. دیتاست چالش برانگیز تر خواهد بود اگر کلاس ها به هم نزدیک باشند و جداسازی آن ها دشوار باشد.

- کاهش تعداد ویژگی های معلوماتی `informative n` که به طور مستقیم در تعیین کلاس نمونه ها دخالت دارند
- افزایش تعداد ویژگی های اضافی یا تکراری `redundant n` و `class per clusters n` که می توانند در تصمیم گیری اشتباه ایجاد کنند
- کاهش فاصله بین کلاس ها تا تداخل بین آن ها بیشتر شود

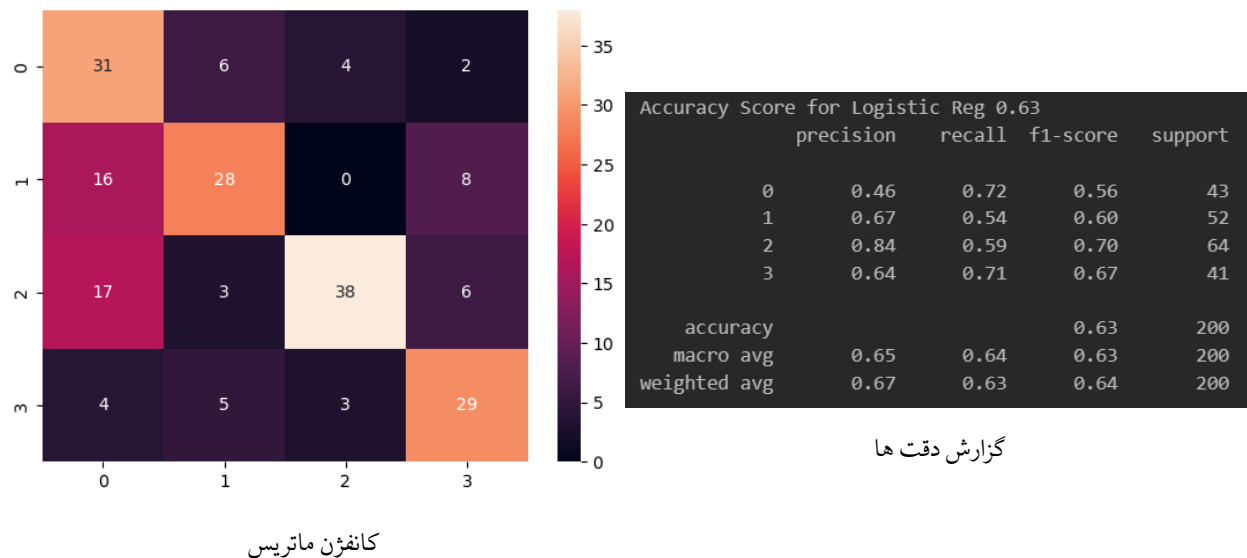


شکل ۲: نمایش سه بعدی دیتاها بر حسب ویژگی های هر رنگ نماد یک کلاس

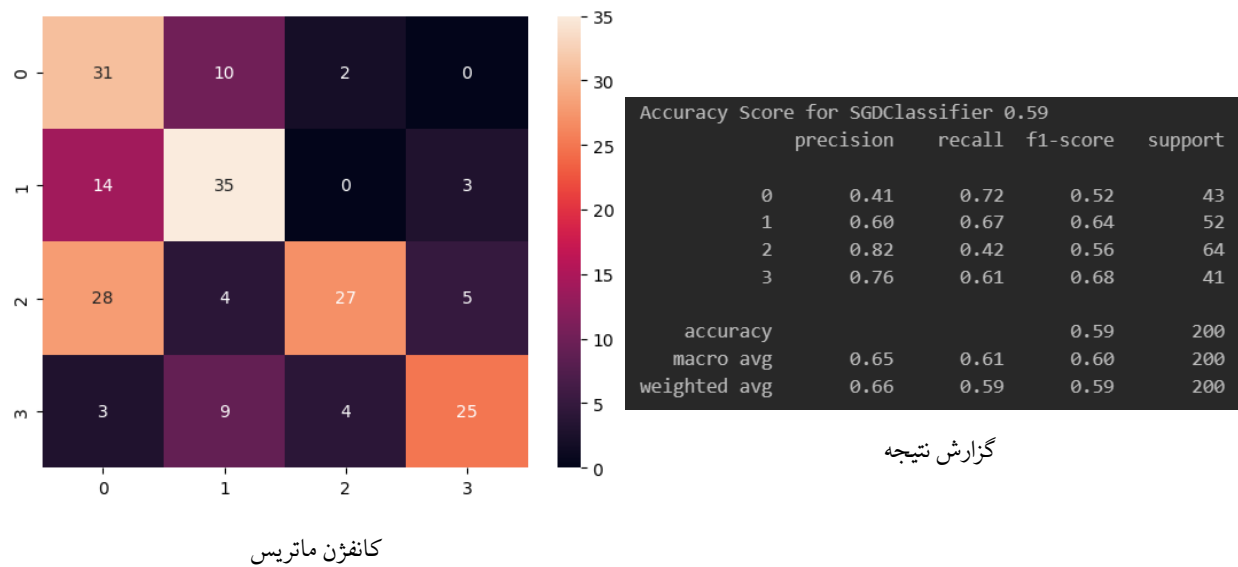
## ۳.۱ بخش سوم سوال اول

برای دسته بندی از دو روش Regression Logistic و SGDClassifier استفاده شده است و نتایج به شرح زیر میباشد. به منظور بهتر شدن نتایج میتوان نوع solver یا تعداد تکرار روند را تغییر داد. تعداد تکرار روند را ۱۰۰۰ در نظر گرفتیم که هرچه بیشتر باشد مدل بیشتر از نقاط ایتیمایز محلی خارج میشود و به نقطه ایتیمایز گلوبال نزدیک میشود. تعیین مقدار لرنینگ ریت نیز به همین صورت میباشد. کوچک در نظر گرفتن آن موجب گیر کردن مدل در نقطه های ایتیمایز محلی میشود و زیاد از حد بزرگ در نظر گرفتن آن نیز موجب گراهی مدل میشود. به غیر از تنظیم هایپر پارامتر ها برای دقت بهتر نیز از Scale کردن داده های ورودی نیز استفاده شده است. به منظور بهتری کردن نتایج مدل نیز میتوان از روش های selection feature استفاده نمود. در واقع با حذف feature هایی که Correlation بالایی دارند میتوان از پیچیدگی مدل جلوگیری نمود. همچنین با ترکیب فیچر ها نیز میتوان به بهبود عملکرد مدل نیز کمک کرد.

با توجه به ۳ نتایج در کل مدل regression Logistic بهتر عمل نموده است. ولی مدل SGD برای شناختن کلاس ۲ از مدل reg Logistic بهتر عمل کرده است.



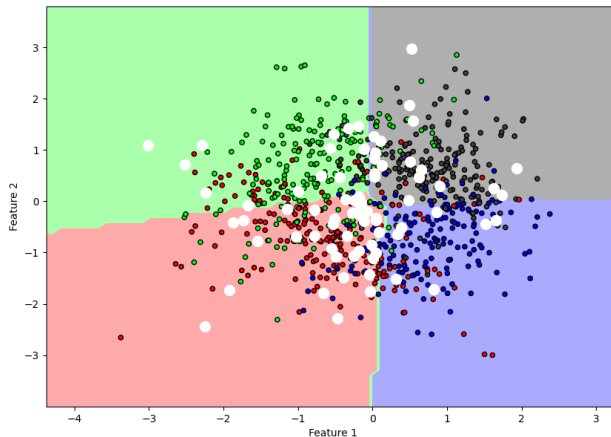
شکل ۳: نتایج بدست آمده برای Regression Logistic



شکل ۴: نتایج بدست آمده از روش SGDClassifier

#### ۴.۱ بخش چهارم سوال اول

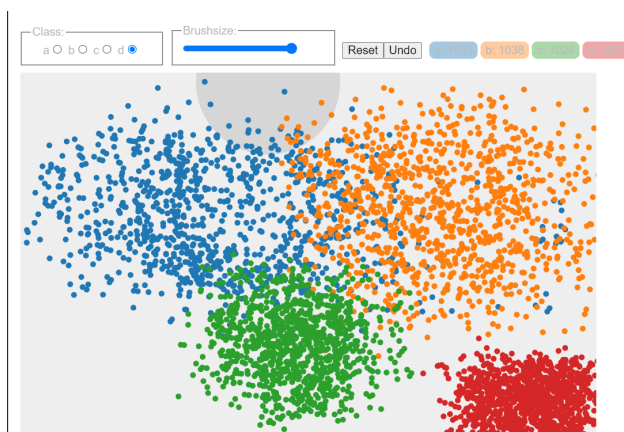
با توجه به شکل ۵ دایره های سفید تو خالی اشتباه پیشبینی شده اند و هر رنگ نماد یک کلاس می باشد. به منظور نمایش بهتر ۲ ویژگی در نظر گرفته شده است.



شکل ۵: نمودار تصمیم گیری

## ۵.۱ بخش پنجم سوال اول

به کمک ابزار drawdata داریم:



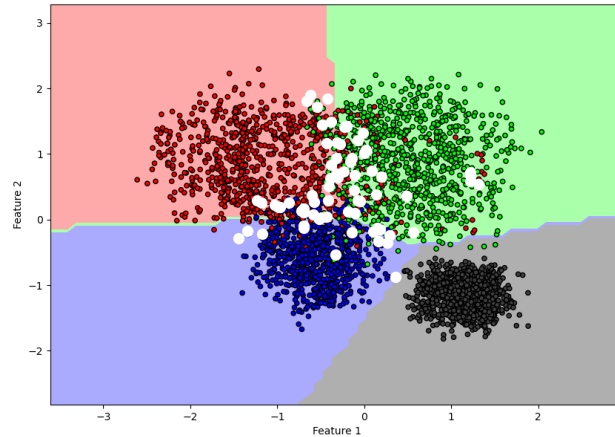
شکل ۶: نمایش دیتاست ایجاد شده در ۴ کلاس

تعداد کلاس ها به ترتیب ۱۰۴۷، ۱۰۳۸، ۱۰۳۳ و ۱۰۲۴ میباشد. مدل لاجستیک بهتر عمل کرده است و توانسته است با دقت بهتری پیشبینی کند.

Accuracy Score for Logistic Reg 0.9143546441495778				
	precision	recall	f1-score	support
0	0.84	0.84	0.84	189
1	0.90	0.89	0.89	223
2	0.91	0.93	0.92	207
3	1.00	1.00	1.00	210
accuracy			0.91	829
macro avg	0.91	0.91	0.91	829
weighted avg	0.91	0.91	0.91	829
Accuracy Score for SGDClassifier 0.9083232810615199				
	precision	recall	f1-score	support
0	0.81	0.81	0.81	189
1	0.87	0.93	0.90	223
2	0.94	0.89	0.92	207
3	1.00	1.00	1.00	210
accuracy			0.91	829
macro avg	0.91	0.91	0.91	829
weighted avg	0.91	0.91	0.91	829

شکل ۷: نتایج ارزیابی دو روش پیاده سازی شده





شکل ۸: نمودار تصمیم گیری

همانطور که از شکل ۸ مشهود است مدل به خوبی توانسته است تشخیص دهد. و تعداد اشتباهات بسیار اندک است.

## ۲ سوال دوم

### ۱.۲ بخش اول سوال دوم

طبق مقاله Network Neural Deep and Wavelet on Based Clustering Automatic Data Fault Bearing Rolling-Element با  
doi:

<https://doi.org/10.1155/2018/3047830>

دیتاست ۴۸k "Fault Bearing End Drive" Data" مربوط به داده‌های ثبت شده از آزمایشاتی است که بر روی بلبرینگ‌های موتورهای الکتریکی انجام شده است. این دیتاست به طور خاص برای تشخیص خرابی‌های بلبرینگ در انتهای درایو موتورهای الکتریکی طراحی شده است. در ادامه به بررسی اهداف، ویژگی‌ها و حالت‌های مختلف این دیتاست می‌پردازیم.

#### اهداف

تشخیص خرابی: اصلی‌ترین هدف از تولید این دیتاست، کمک به تشخیص خرابی‌های بلبرینگ در ماشین‌آلات چرخان مانند موتورهای الکتریکی است.

#### ویژگی‌ها

کیفیت بالا و دقت اندازه‌گیری: داده‌ها با دقت و وضوح بالا ثبت شده‌اند، اغلب با نرخ نمونه‌برداری ۴۸ کیلوهرتز، که امکان تشخیص دقیق خرابی‌ها را فراهم می‌آورد.

تنوع در خرابی‌ها: شامل داده‌های مربوط به انواع مختلف خرابی‌های بلبرینگ مانند خرابی‌های داخلی، خارجی و توپی بلبرینگ. حالت‌های مختلف

حالت عادی: داده‌هایی که در شرایط عادی و بدون خرابی ثبت شده‌اند.

حالت خرابی: داده‌های ثبت شده در حالت‌های مختلف خرابی، که به منظور آموزش مدل‌های تشخیص خرابی استفاده می‌شوند.

## ۱.۱.۲ بخش آ بخش دوم سوال دوم

به کمک ایجاد یک لوپ for با رنج M و استفاده از دستود sample با طول N و رندوم استیت متفاوت دیتاست را از فیچر  $X_{0.97\_DE\_time}$  ایجاد میکنیم. همینطور برای دیتاست رندوم  $x_{1.05\_DE\_time}$  را تکرار میکنیم و بعد از لیبیل زدن دیتا ها را مرج میکنیم.

## ۲.۱.۲ بخش ب بخش دوم سوال دوم

استخراج ویژگی یکی از مراحل کلیدی در فرآیند یادگیری ماشین است که به بهبود دقت مدل ها کمک می کند. ویژگی های مناسب باعث می شوند که الگوها و روابط معنادار در داده ها بهتر شناسایی شوند.

## • کاهش پیچیدگی محاسباتی

استخراج ویژگی به کاهش پیچیدگی محاسباتی مدل های یادگیری ماشین کمک می کند، زیرا با حذف داده های زائد و کاهش ابعاد داده، فرآیند آموزش مدل ساده تر و سریع تر می شود.

## • جلوگیری از بیش برازش

استخراج ویژگی موثر در جلوگیری از بیش برازش است، زیرا با حذف ویژگی های غیر معنادار و تمرکز بر مشخصه های کلیدی، مدل قادر به تعمیم بهتر بر روی داده های جدید خواهد بود.

## • و ...

در این بخش از ویژگی های زیر استفاده شده است:

*STD, Peak, Crest\_Factor, Cleareance\_Factor, SMR, Mean, Absolute\_mean* □ *Impact\_Factor*.

## ۳.۱.۲ بخش ج بخش دوم سوال دوم

دیتا ها را به صورت ۸۰/۲۰ تقسیم میکنیم شافل کردن داده ها قبل از تقسیم آن ها به دو مجموعه برای آموزش و تست، اهمیت زیادی در یادگیری ماشین دارد. این روند اطمینان می دهد که داده ها به صورت تصادفی تقسیم شوند، که دارای چند مزیت کلیدی است:

• جلوگیری از بایاس آموزشی: شافل کردن مانع از آن می شود که مدل تنها بر روی ترتیب خاصی از داده ها آموزش ببیند و به این ترتیب از بایاس آموزشی جلوگیری می کند.

• توزیع یکنواخت داده ها: با شافل کردن، می توان اطمینان حاصل کرد که هر دو مجموعه آموزش و تست توزیعی یکنواخت از تمام کلاس ها و الگوها دارند.

• بهبود قابلیت تعمیم مدل: این فرآیند به مدل کمک می کند تا بهتر بتواند بر داده های جدید تعمیم یابد، زیرا با تنوع بیشتری از داده ها آموزش دیده است.

## ۴.۱.۲ بخش د بخش دوم سوال دوم

برای نرمال سازی داده ها، دو روش متداول وجود دارد: نرمال سازی به وسیله مقیاس مین-مکس (Min-Max) (Scaling) و استاندارد سازی (Z-score Normalization). نرمال سازی داده ها به ویژه در یادگیری ماشین از اهمیت بالایی برخوردار است، زیرا باعث بهبود عملکرد الگوریتم ها می شود و کمک می کند تا مدل بهتری از داده ها یاد بگیرد.

• نرمال سازی مین-مکس (Min-Max Scaling)

این روش داده‌ها را در محدوده‌ای مشخص (معمولاً ۰ تا ۱) تغییر مقیاس می‌دهد. این کار با کاهش کمترین مقدار داده از هر مقدار و تقسیم آن بر دامنه‌ی داده‌ها انجام می‌شود. فرمول آن به صورت زیر است:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

که در آن  $X_{norm}$  مقدار نرمال‌سازی شده،  $X$  مقدار اصلی،  $X_{min}$  و  $X_{max}$  به ترتیب کمترین و بیشترین مقادیر در داده‌ها هستند.

• استاندارد سازی (Z-score Normalization)

این روش از میانگین و انحراف معیار داده‌ها برای نرمال‌سازی استفاده می‌کند. هدف این است که میانگین داده‌های نرمال‌سازی شده صفر و انحراف معیار آن‌ها یک باشد. فرمول آن به شکل زیر است:

$$Z = \frac{X - \mu}{\sigma}$$

که در آن  $Z$  مقدار نرمال‌سازی شده،  $X$  مقدار اصلی،  $\mu$  میانگین و  $\sigma$  انحراف معیار داده‌ها است.

اهمیت نرمال سازی داده‌ها در یادگیری ماشین شامل بهبود سرعت آموزش و کارایی الگوریتم‌ها، کاهش حساسیت به مقیاس متغیرها و بهبود قابلیت تعمیم دهی مدل است.

ما از استاندارد اسکیل استفاده کردیم. خیر چراکه در واقعیت نیز از آینده خبر نداریم و باید با استفاده از دیتای آموزش نرمال‌سازی کنیم و با توجه به معادله نرمال‌لایز کردن دیتای آموزش، دیتای ارزیابی را نیز نرمال کنیم. در واقع روی دیتاست آزمایش fit میکنیم و روی دیتاست ارزیابی transform میکنیم.

## ۲.۲ بخش سوم سوال دوم

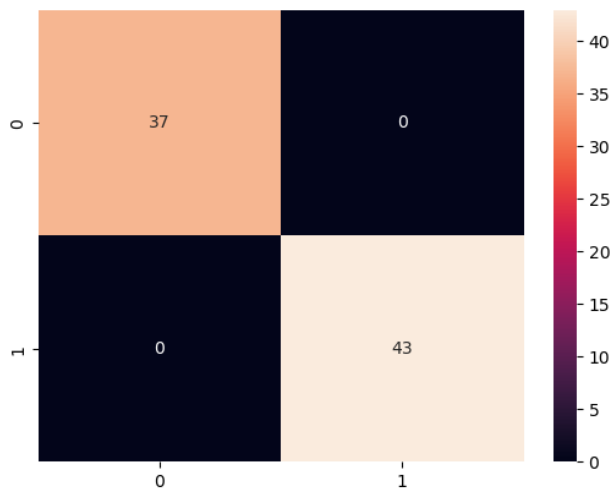
از یک پرسترون با اکتیویشن فانکشن سیگموئید، با loss logistic با ۱۰۰۰ اپاک و ۲ متریک accuracy و F1score مدل را میسازیم از آنجایی مدل باینری میباشد باید آستانه برای پیشبینی در نظر بگیریم که خروجی ۱ یا ۰ باشد یا ۰.۵

نتایج به شرح زیر میباشد:

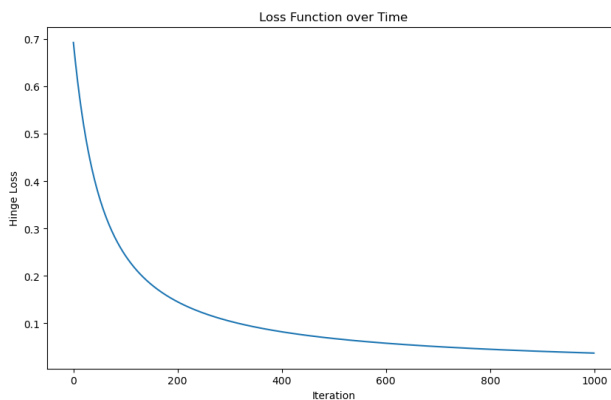
Accuracy و F1Score برابر یک میباشد. یعنی مدل بسیار عالی آموزش دیده است.

همانطور که توقع داشتیم تابع loss در حال کاهش میباشد که نشان میدهد مدل در حال آموزش میباشد.

خیر نمیتوان از روی تابع اتلاف نظر قطعی داد صرفاً میتوان روند آموزش را دید چراکه مدل ممکن است overfit شده باشد و خطای بایاس کم اما خطای واریانس بالا داشته باشد. برای جلوگیری از overfitting باید از روش های موجود مانند stopping early استفاده نمود.



شکل ۹: کانفرن ماتریس خروجی

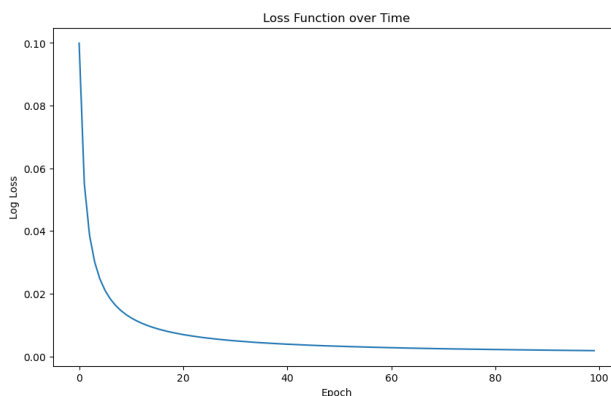


شکل ۱۰: نمودار تابع اتلاف

### ۳.۲ بخش چهارم سوال دوم

به کمک روش SGDClassifier با تابع اتلاف  $\log\_loss$  و با تنظیم باقی پارامترها داریم:  
 $F1Score$  و  $Accuracy$  برابر یک می باشد. یعنی مدل بسیار عالی آموزش دیده است.

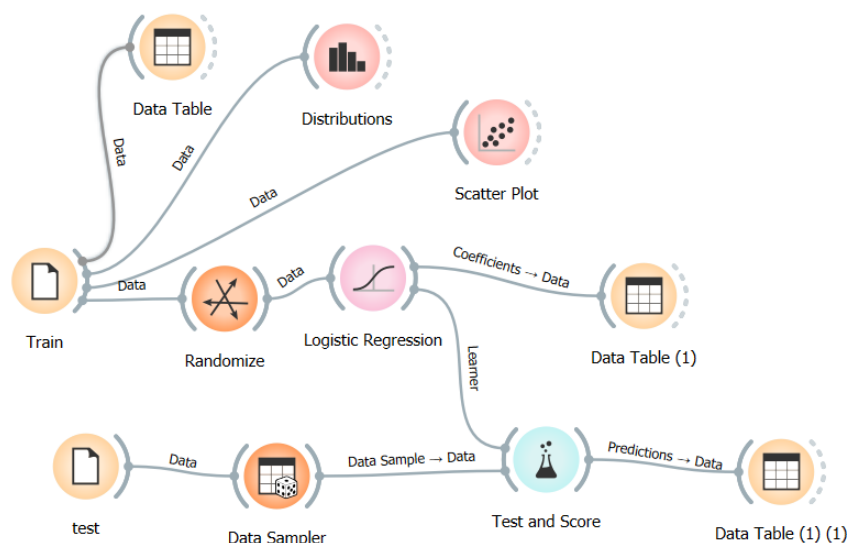
خیر بطور مستقیم نمیتوان تابع اتلاف را رسم نمود و باید حلقه ای با  $\text{partial\_fit}$  نوشت و به کمک آن هر مرحله را ذخیره نمود تا بتوان تابع اتلاف را رسم کرد.



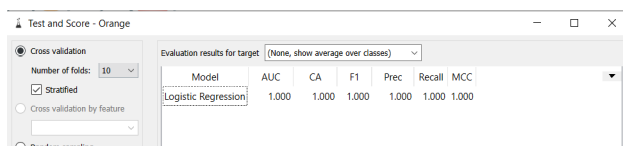
شکل ۱۱: نمودار تابع اتلاف

## ۴.۲ قسمت پنجم سوال دوم

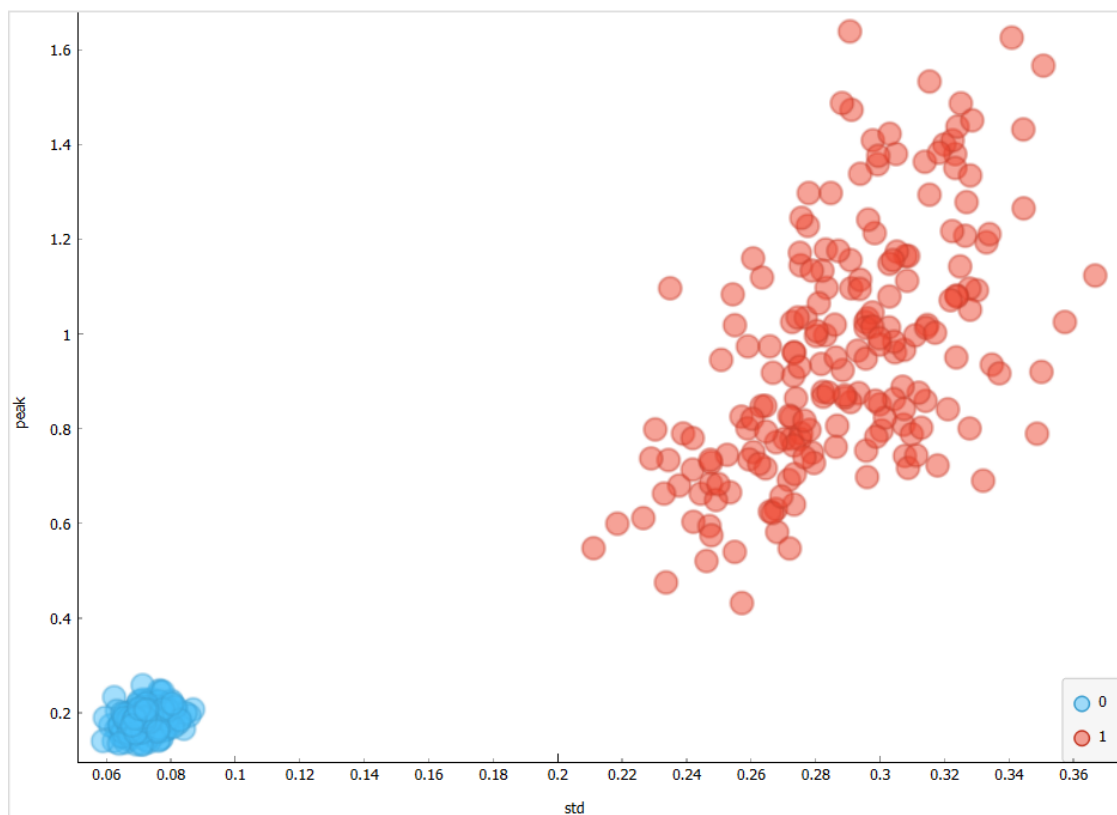
شماتیک مدل ایجاد شده در شکل ۱۲ آورده شده است. همچنین به منظور تحلیل بهتر، `distributions scatter_plot` قبل از ایجاد مدل بررسی شده است. دیتا ها را به دو بخش `Train`، `test` تقسیم میکنیم. دیتاست `Train` را پیش از مدل کردن شافل میکنیم. همچنین به منظور ارزیابی از بلوک `test` `Score and` استفاده میکنیم و نتایج را بدست می آوریم. نتایج ارزیابی `...`، `FI`، `AUC` در شکل ۱۳ آورده شده است. طبق شکل ۱۴ میتوانیم توجیه کنیم که مدل میتواند ۱۰۰ درصد عمل درستی داشته باشد چرا که تکنیک پذیری دیتاست برحسب ویژگی های استخراج شده بسیار خوب است. نمونه پیشینی دیتا ها در شکل ۱۵ آورده شده است.



شکل ۱۲: شماتیک کلی مدل



شکل ۱۳: نتایج بدست آمده



شکل ۱۴: نمودار کلاس بندی برحسب ۲ ویژگی STD و Peak

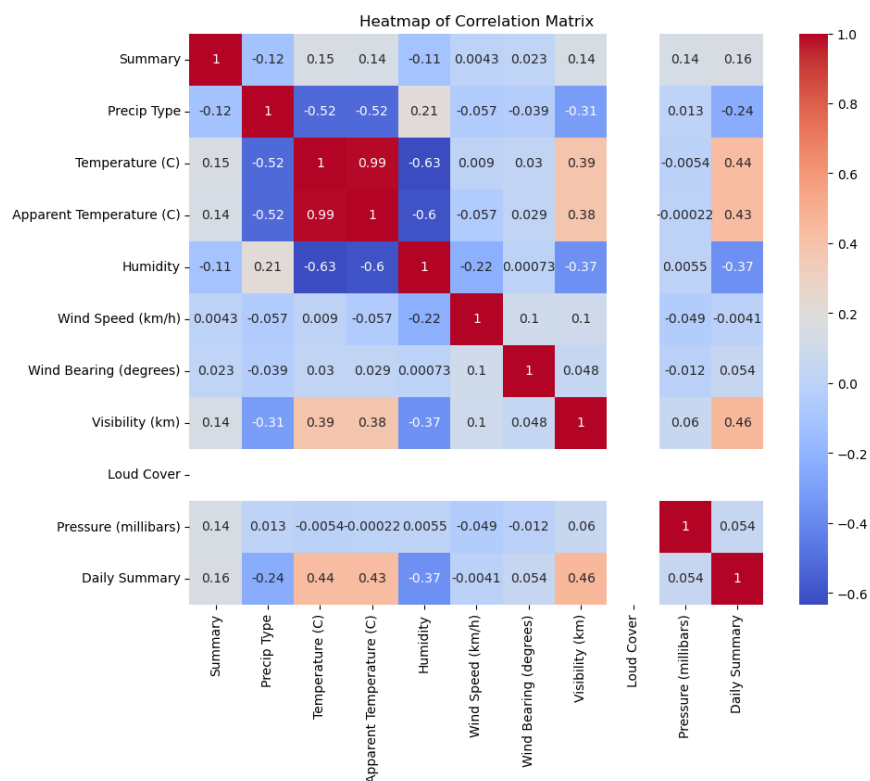
	label	Logistic Regression	Logistic Regression (0)	Logistic Regression (1)	Fold	std	peak	crest	clearance
1	0	0	0.995809	0.004191	1	0.0721786	0.206112	2.84429	32.0
2	0	0	0.995178	0.00482241	1	0.0749314	0.22447	2.95094	17.1
3	1	1	0.0389657	0.961034	1	0.246183	0.520767	2.10207	-18.7
4	1	1	0.000111647	0.999888	1	0.260646	1.15979	4.44085	70.6
5	1	1	0.000380725	0.999619	1	0.273301	0.960641	3.51285	10
6	0	0	0.991292	0.00870782	1	0.0761899	0.141441	1.83099	11.0
7	0	0	0.992459	0.00754133	1	0.0791673	0.196098	2.45578	18.7
8	0	0	0.991355	0.00864456	1	0.0744125	0.154793	1.96504	5.98
9	1	1	0.00635242	0.993648	1	0.230293	0.798531	3.36494	13.9
10	0	0	0.994241	0.00575906	1	0.0701156	0.184416	2.44927	6.72
11	1	1	0.000188398	0.999812	1	0.254338	1.08377	4.07705	14.
12	0	0	0.995924	0.00407617	1	0.0692557	0.184833	2.63154	15.7
13	1	1	0.0004386	0.999561	1	0.254972	1.01863	3.99439	216.
14	1	1	1.89151e-06	0.999998	1	0.313896	1.36364	4.34374	-283.
15	0	0	0.996882	0.00311844	1	0.0656004	0.189631	2.81163	12.1

شکل ۱۵: نمونه خروجی مدل و پیشبینی

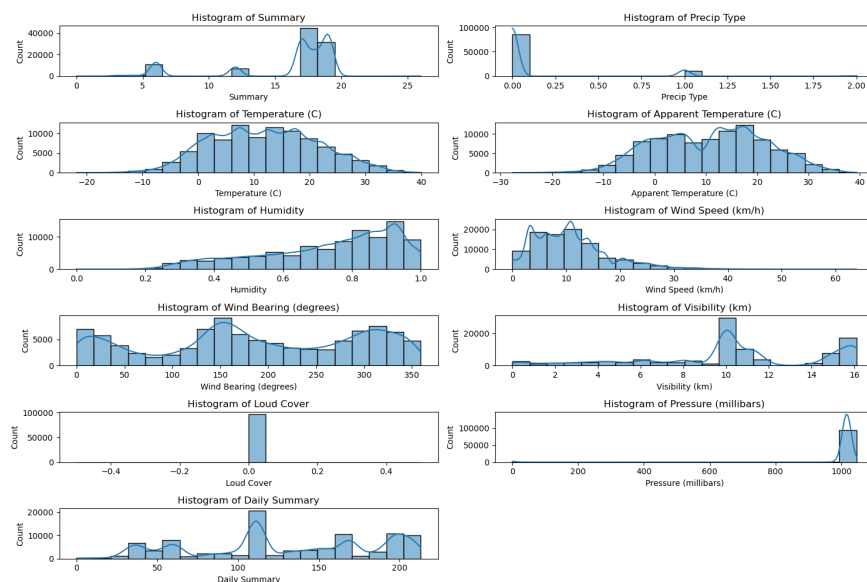
## ۳ عنوان سوال سوم

## ۱.۳ بخش اول سوال سوم

همانطور که از شکل ۱۶ مشهود است دیتای های ، Humidity، Apparent precipType، Temp، temp بیشترین وابستگی را بهم دارند همینطور به کمک هیستوگرام ها میبینیم که این ویژگی ها تقریبا به یک شکل توزیع شده اند. تقریبا زنگوله ای توزیع شده اند.



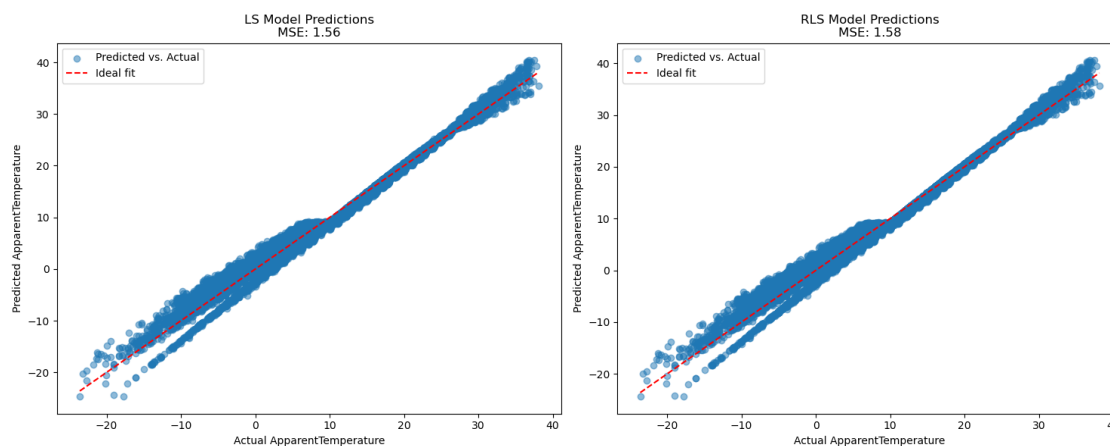
شکل ۱۶: هیت مپ ماتریس همبستگی



شکل ۱۷: هیستوگرام پراکندگی ویژگی‌ها

## ۲.۳ بخش دوم سوال سوم

با توجه به معیاری MSE که برای هر دو مدل برای ارزیابی بدست آورده ایم متوجه می‌شویم مدل بهتر LS از RLS عمل کرده است. نتایج در شکل زیر آورده شده است.



شکل ۱۸: نتایج و نمودار دو روش پیاده شده



## ۳.۳ بخش سوم سوال سوم

تخمین Squares Least Weighted (WLS) یک روش در رگرشن است که زمانی استفاده می‌شود که واریانس خطاهای مدل ناهمگن باشد یا داده‌ها با وزن‌های متفاوتی دارای اهمیت باشند. در WLS، هر نقطه داده با وزنی متناسب با قابل اطمینان بودن یا اهمیت آن نقطه در تحلیل داده‌ها محاسبه می‌شود.

فرمول‌بندی WLS:

فرمول بنیادی برای رگرشن WLS به صورت زیر است:

$$\hat{\beta} = (X^T W X)^{-1} X^T W y$$

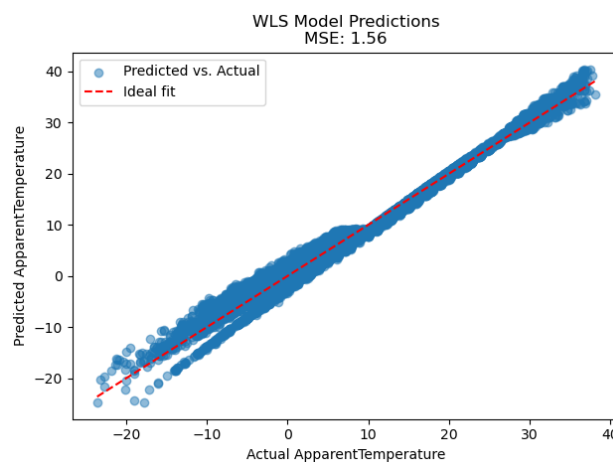
که در آن:

- $\hat{\beta}$  برآوردگرهای مدل رگرشن هستند.
- $X$  ماتریس طراحی است که شامل متغیرهای توضیحی است.
- $y$  بردار پاسخ است.
- $W$  ماتریس وزن‌ها است، که معمولاً یک ماتریس قطری با واریانس‌های معکوس خطاها به عنوان عناصر قطری است.

مزایای WLS:

استفاده از WLS می‌تواند چند مزیت عمده داشته باشد:

۱. بهبود دقت برآوردها در مواجهه با واریانس ناهمگن.
  ۲. توانایی در نظر گرفتن تفاوت‌های میان نقاط داده از نظر قابلیت اطمینان یا اهمیت.
  ۳. بهبود خواص آماری برآوردها، مانند کاهش واریانس.
- روش WLS نیز با MSE ۵۶.۱ خوب عمل کرده است. نمودار عملکرد آن در شکل زیر آورده شده است.



شکل ۱۹: نتایج و نمودار روش پیاده شده

## ۴.۳ بخش چهارم سوال سوم

الگوریتم RLS (کمترین مربعات بازگشتی) یک روش پیشرفته در پردازش سیگنال و فیلترینگ تطبیقی است که برای تخمین پارامترها در مدل‌های زمانی متغیر به کار می‌رود. در مقابل، تجزیه QR به عنوان یک تکنیک محاسباتی در جبر خطی، به کارگیری این الگوریتم را بهینه‌سازی می‌کند. RLS QR-Decomposition-Based با استفاده از تجزیه QR برای به‌روزرسانی پارامترها، پایداری عددی و کارایی محاسباتی بالایی را فراهم می‌آورد و به خصوص در سیستم‌هایی با داده‌های بزرگ و پیچیده مفید است.

در الگوریتم RLS، QR-Decomposition-Based ابتدا ماتریس‌های داده را با استفاده از تجزیه QR به ماتریس‌هایی با ساختار اورتوگونال و مثلثی تقسیم می‌کنند. این فرآیند به تعیین پارامترهای بهینه برای مدل‌های پیش‌بینی کمک می‌کند، و با کاهش اثرات نویز و دیگر مشکلات محاسباتی، دقت برآورد را بهبود می‌بخشد. فرآیند به‌روزرسانی در RLS از طریق یک فیلتر کالمن دینامیکی انجام می‌شود، که به‌طور مداوم خطاهای پیش‌بینی و واریانس‌های تخمین را ارزیابی و تنظیم می‌کند.

استفاده از الگوریتم RLS QR-Decomposition-Based در بسیاری از برنامه‌های کاربردی نظیر پردازش سیگنال، سیستم‌های مخابراتی، کنترل خودکار و تشخیص الگو مطرح است. با توجه به توانایی خود در مقابله با چالش‌های داده‌های زمانی متغیر و نیازهای محاسباتی سنگین، این الگوریتم به یک ابزار قدرتمند در زمینه مهندسی برق و کامپیوتر تبدیل شده است.

## Algorithm: RLS QR-Decomposition-Based of Mathematics

used technique filtering adaptive an is algorithm (RLS) Squares Least Recursive QR-Decomposition-Based The an into matrix data the decompose to is algorithm this of essence The systems. time-varying of estimation online for estimates. parameter updating of process the simplify to  $R$  matrix triangular upper an and  $Q$  matrix orthogonal  $\theta \in R^n$  vector parameter the find to is objective the  $y \in R^m$  vector response a and  $X \in R^{m \times n}$  matrix data a Given

$J(\theta) = \|y - X\theta\|^2$  function cost the minimizes that

by: given is  $X$  of decomposition QR The

$$X = QR \quad (1)$$

matrix. triangular upper an is  $R \in R^{m \times n}$  and matrix. orthogonal an is  $Q \in R^{m \times m}$  where nu- maintain to decomposition QR the utilizing arrives. data new as  $\theta$  of estimate the updates algorithm RLS The follows: as are equations update The stability. merical

$$\theta_{k+1} = \theta_k + K_k(y_k - X_k\theta_k) \quad (2)$$

variance. error posteriori a the minimize to calculated gain, Kalman the is  $K_k$  where dynamics system the where scenarios in beneficial particularly is algorithm RLS QR-Decomposition-Based The RLS standard to compared efficiency computational and stability numerical superior offers It changing. rapidly are algorithms.

فرنس:

Haykin Simon اثر Theory" Filter Adaptive