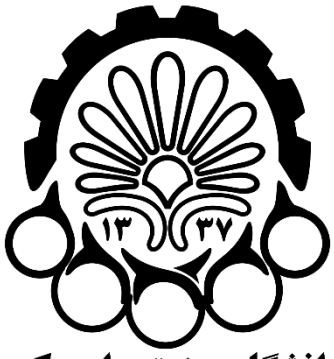
به نام خدا



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

تمرین درس یادگیری ماشین

فردین آیار

شماره دانشجویی: ۹۹۱۳۱۰۴۰

استاد: دکتر ناظرفرد

دانشکده کامپیوتر- پاییز ۹۹

۱)

الف) در یادگیری نظارتی یک مجموعه ورودی و خروجی صحیح(برچسب) در دسترس است و سیستم باید سعی کند رابطه ای بین ورودی ها و خروجی ها یادبگیرد.

ب) دراین نوع یادگیری، هم از داده های برچسب خورده و هم از داده های بدون برچسب، برای پیدا کردن رابطه بین ورودی و خروجی استفاده می شود.

پ) در این روش، داده ها هیچگونه برچسبی به عنوان خروجی ندارند و هدف از آن، برخلاف روش های قبلی،نه پیدا کردن خروجی ؛ بلکه دسته بندی داده های ورودی است.

ت) یادگیری تقویتی تا حدی شبیه به یادگیری مبتنی بر سعی و خطای انسان هاست. در این روش سیستم بعد از هر خروجی، یک بازخورد از محیط دریافت می کند که نشان دهنده‌ی مطلوبیت آن خروجی است. به بیانی دیگر سیستم سعی میکند پاداش(بازخورد مطلوب) را بیشینه کند و از این طریق خروجی های مناسب را میابد.

ث) یادگیری عمیق سعی دارد ساختار مغز انسان را شبیه سازی کند. در یادگیری عمیق با استفاده از یک گراف عمیق که شامل چند لایه پردازشی است سعی می شود مانند انسان، یادگیری از طریق مثال ها صورت گیرد. در این روش نیازی به تعیین ویژگی ها توسط کاربر نیست و الگوریتم مستقیما از روی داده ها یاد می گیرد.

ج) این روش زیر مجموعه یادگیری نظارتی است. در رگرسیون، مجموعه ای از داده ها و خروجی آنها که یک متغیر پیوسته است را داریم و سیستم باید یک رابطه ریاضی بین داده ها و خروجی بیابد. در اینجا داده ها دارای چند ویژگی عددی هستند که به آنها متغییر های مستقل می گویند. هدف یافتن تابع خروجی(متغیر وابسته) است.

چ) در یادگیری بر خط، تمام داده های آموزشی از ابتدا در دسترس نیستند و جریانی از داده ها بتدریج در اختیار الگوریتم قرار می گیرد. در نتیجه الگوریتم باید توانایی یادگیری به صورت لحظه ای را داشته باشد.

ح) در یادگیری فعال،مجموعه بزرگی از داده های بدون برچسب داریم و سیستم می تواند برچسب داده های مورد نظرش را از ناظر بپرسد. مسئله اصلی در چنین سیستمی این است که یادگیری با حداقل تعداد پرسش ممکن صورت بگیرد؛ بنابراین سیستم باید بهترین داده ها را برای پرسش انتخاب کند.

خ)مسئله ای که در آن نتایج خروجی به صورت گسسته باشد، دسته بندی نام دارد. در این‌گونه مسائل، رابطه ای بین ویژگی های داده ها(متغیرهای مستقل) و برچسب داده ها(متغییر وابسته گسسته) یافت می شود. در چنین مسائلی معمولا کل داده های آموزشی یا مقداری از آنها برچسب دار هستند.

د) در خوشه بندی، مانند دسته بندی، داده ها به چند دسته تقسیم می شوند؛ با این تفاوت که در اینجا دسته ها برچسب ندارند. داده ها در مسئله خوشه بندی بدون برچسب هستند و صرفا بر اساس ویژگی‌های آن‌ها، در چند دسته قرار می‌گیرند؛ از این رو این مسئله مربوط به یادگیری بدون نظارت است.

ذ) هنگامی که مدل(سیستم) به خوبی روی مجموعه داده های آموزشی کار کند(خطا روی داده های آموزشی بسیار کم باشد) اما قابل تعمیم به داده های خارج از این مجموعه نباشد،(خطا خارج از داده های آموزشی زیاد باشد) بیش برازش اتفاق افتاده است. این اتفاق معمولا زمانی می‌افتد که داده های آموزش نویزی می باشند یا سیستم سعی کند با در نظر گرفتن ویژگی های متعدد، کاملا بر روی داده های آموزشی منطبق شود. در این حالت اصطلاحا گفته می شود مدل داده ها را حفظ کرده است. در مقابل اگر داده ها به اندازه کافی بر روی داده های آموزشی منطبق نشود، یادگیری کامل صورت نمی گیرد و بدیهی است که خطا خارج از داده‌های آموزشی نیز، زیاد خواهد بود.

۲) همبستگی نشان دهنده ارتباط بین ویژگی(متغیر)های مختلف داده هاست. اگر همبستگی مثبت باشد، یعنی با افزایش یک متغیر، مقدار متغیر دیگر نیز افزایش می یابد.(یا با کاهش یکی،دیگری نیز کاهش می یابد) همبستگی منفی یعنی با افزایش یک متغیر، متغیر دیگر کاهش می یابد و همبستگی صفر نشان می دهد هیچ رابطه معنا داری بین دو متغیر وجود ندارد. وجود همبستگی به سادگی از روی ماتریس کوواریانس قابل تشخیص است. به این صورت که اگر عنصر ij ماتریس صفر نباشد، یعنی دو متغیر همبستگی دارند. برای بدست آوردن مقدار همبستگی(که نشان دهنده شدت همبستگی است)، باید عنصر مربوط به دو متغیر در ماتریس کوواریانس را بر حاصل‌ضرب انحراف معیار آن‌ها، تقسیم کرد.

۳) در معیار MSE، میانگین مربعات خطاها به عنوان ارزیابی در نظر گرفته می شود.(منظور از خطا تفاضل مقدار واقعی و مقدار پیش بینی شده است) این روش، به دلیل وجود توان دو، به خطاهای بزرگتر، وزن بیشتری می دهد؛ بنابراین وقتی داده های پرت وجود داشته باشد، ممکن است خطا بیش از حد نشان داده شود و یک مدل خوب به عنوان مدل بد تشخیص داده شود.

معیار MAE از میانگین قدرمطلق ها به عنوان ارزیابی استفاده می شود. به همین دلیل بر خلاف MSE نسبت به داده های پرت و نویزی حساسیت کمتری دارد. ویژگی دیگر این معیار این است که مقیاس آن با مقیاس مقادیر خروجی یکسان است؛ بنابراین به سادگی می توان از آن برای تخمین درصد خطا استفاده کرد.

معیار RMSE برابر است با ریشه دوم MSE. این معیار مانند MSE نسبت به داده های پرت حساسیت دارد، اما مقیاس آن با داده های خروجی یکسان است. اگرچه مقدار این شاخص، از شاخص MAE بیشتر خواهد بود.

به طور خلاصه می توان گفت معیار MAE به دلایلی که شرح داده شد، در صورت داشتن داده های نویزی بهتر عمل خواهدکرد.

٤)

گرادیان نزولی	معادله نرمال
نیاز به تعیین ضریب یادگیری توسط کاربر دارد. بنابراین برای یافتن مقدار مناسب الگوریتم چندبار باید اجرا شود.	نیاز به تعیین ضریب یادگیری ندارد.
نیاز به تکرار دارد. و به خصوص در روش batch gradient descent هزینه هر تکرار با افزایش تعداد داده ها، افزایش می یابد.	نیاز به تکرار ندارد.
برای تعداد بالای feature به خوبی عمل می کند.	با افزایش تعداد feature ها سرعت آن کاهش می یابد.

۵) در رگرسیون خطی، تعدادی داده با چند ویژگی وجود دارد و سعی می شود با نسبت دادن ضریب مناسب به هر یک از ویژگی ها، تابع هدف یافته شود. به طور دقیق تر هدف در رگرسیون خطی، هدف یافتن پارامترهای θ_j با کمینه کردن تابع زیر است.

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h(x_i) - y_i)^2$$

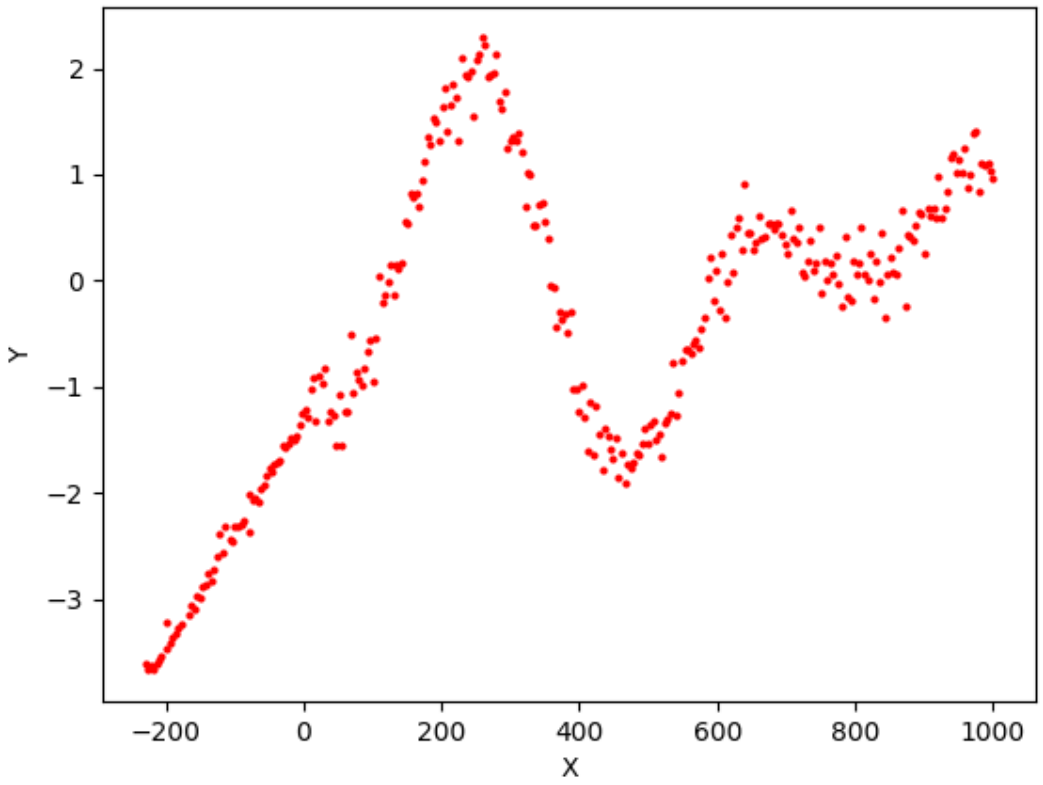
در روش lasso و نیز سایر روش های رگولاریزیشن با افزودن یک عبارت پنالتی(مجازات) سعی می شود مقدار ضرایب θ را کاهش دهند تا مشکلات بیش برازش حل شود. به طور خاص عبارت پنالتی در روش lasso به شکل زیر است:

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h(x_i) - y_i)^2 + \gamma\sum_{j=1}^{n}|\theta_j|$$

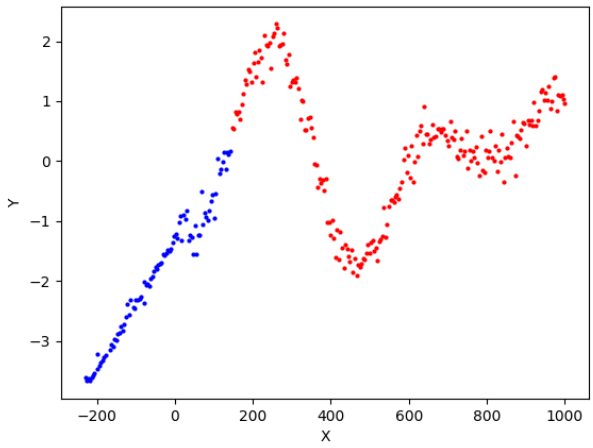
تفاوت اصلی این روش با سایر روش ها این است که lasso تمایل بیشتری به صفر کردن ضرایب دارد. درصورتی که در روش عادی معمولا همه ضرایب در تابع نهایی ظاهر می شوند. صفر شدن ضرایب باعث ساده تر شدن مدل می شود.

سوالات پیاده سازی: بخش اول

۱)



۲) هنگامی که داده ها دارای نظم خاصی باشند، جداکردن مجموعه آموزش و آزمون از آنها باعث می شود این دو مجموعه پراکندگی لازم را نداشته باشند. به این ترتیب امکان بیش برازش روی داده‌های آموزش وجود دارد و الگوریتم نمی تواند همه جنبه‌های لازم را یاد بگیرد. برای حل این مشکل داده ها باید شافل شوند. دیتاست۱ مربوط به این سوال شامل داده هایی است که بر حسب X مرتب شده اند؛ بنابراین به وضوح نیاز به شافل کردن دارند. شکل زیر نشان دهنده حالتی است که داده های آموزش و آزمون بدون شافل کردن استخراج شده اند.



با توجه به شکل بالا مشخص است الگوریتم نمی تواند با استفاده از داده های آموزش(نقاط قرمز)، داده های آزمون(نقاط آبی) را پیش بینی کند. در این پروژه، داده ها با استفاده از فایل shuffling.py شافل شده، و در دیتاست جدیدی به نام s_data ذخیره شده اند. در برنامه اصلی از فایل s_data به عنوان ورودی استفاده شده است.

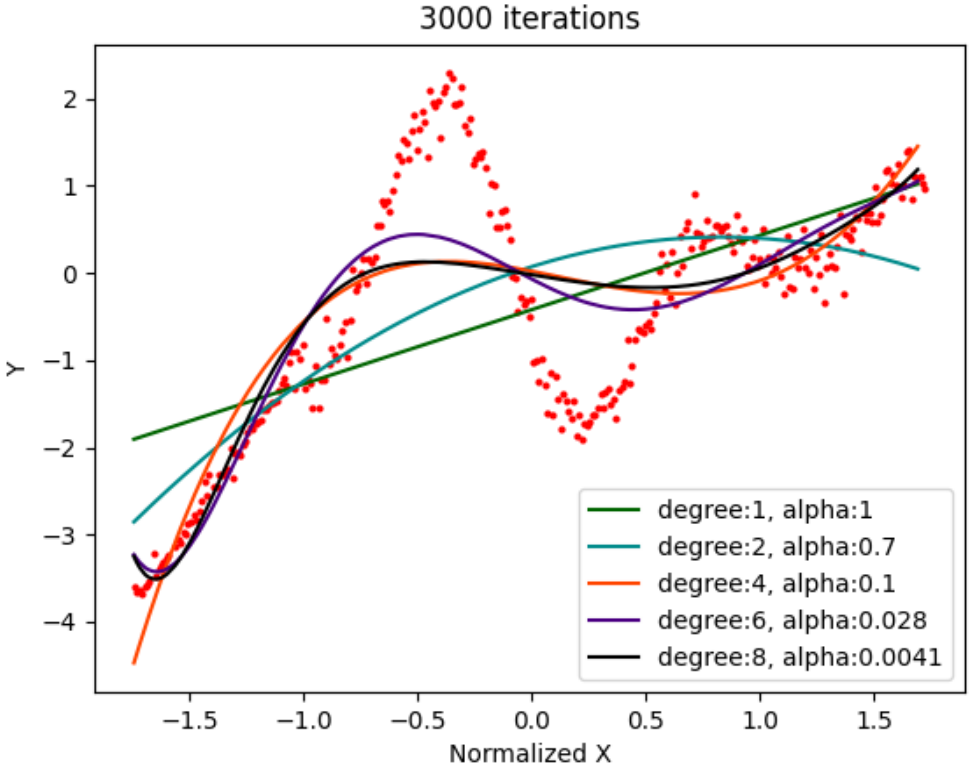
نرمال سازی داده ها به منظور افزایش سرعت همگرا شدن در الگوریتم گرادیان نزولی انجام می شود. داده های دیتاست۱ با توجه به پراکندگی مقادیر X نیاز به نرمال سازی دارند. نرمال سازی در این پروژه با استفاده از فرمول زیر صورت گرفته است:

$$x = \frac{X - mean}{standard\ deviation}$$

نرمال سازی فوق بعد از آزمایش چندین نرمال سازی مختلف، به علت سرعت همگرایی بیشتر آن، انتخاب شد. همچنین داده های آزمون باید با توجه به پارامترهای داده های آموزش، نرمال سازی شوند که در این پروژه، این مورد رعایت شده است.

۳) برنامه مربوط به این بخش در فایل Q3.py قرار دارد. درجات مقایسه شده و ضریب یادگیری متناظر با آنها در نمودار ذکر شده است. سی درصد از داده ها به عنوان داده های آزمون و هفتاد درصد به عنوان داده های آموزش استفاده شده اند.

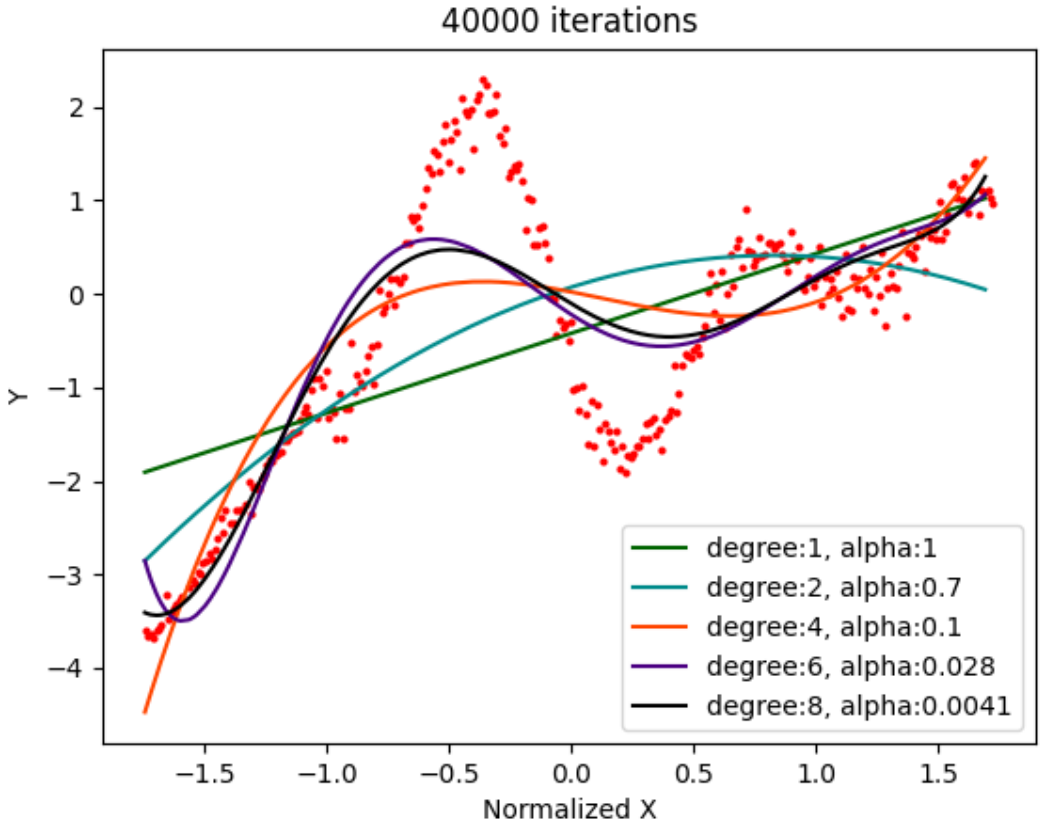
تکرار ۳۰۰۰



degree	Error Train	Error Test
1	1.16001	1.73840
2	0.97109	1.45099
4	0.60233	0.91327
6	0.45370	0.67237
8	0.56106	0.88639

برای این تعداد تکرار، همانطور که مشاهده می شود. نمودار درجه ۶ کمترین خطا را دارد. نمودار درجه ۸ به دلیل نیاز به تکرار بالا در این قسمت هنوز همگرا نشده و بنابراین خطای آن از درجه ۶ نیز بیشتر است. سایر درجات به نظر میرسد به همگرایی رسیده اند و خطای آنها مطابق انتظار است. همچنین مطابق انتظار خطای داده های آموزشی از داده های آزمون کمتر است.

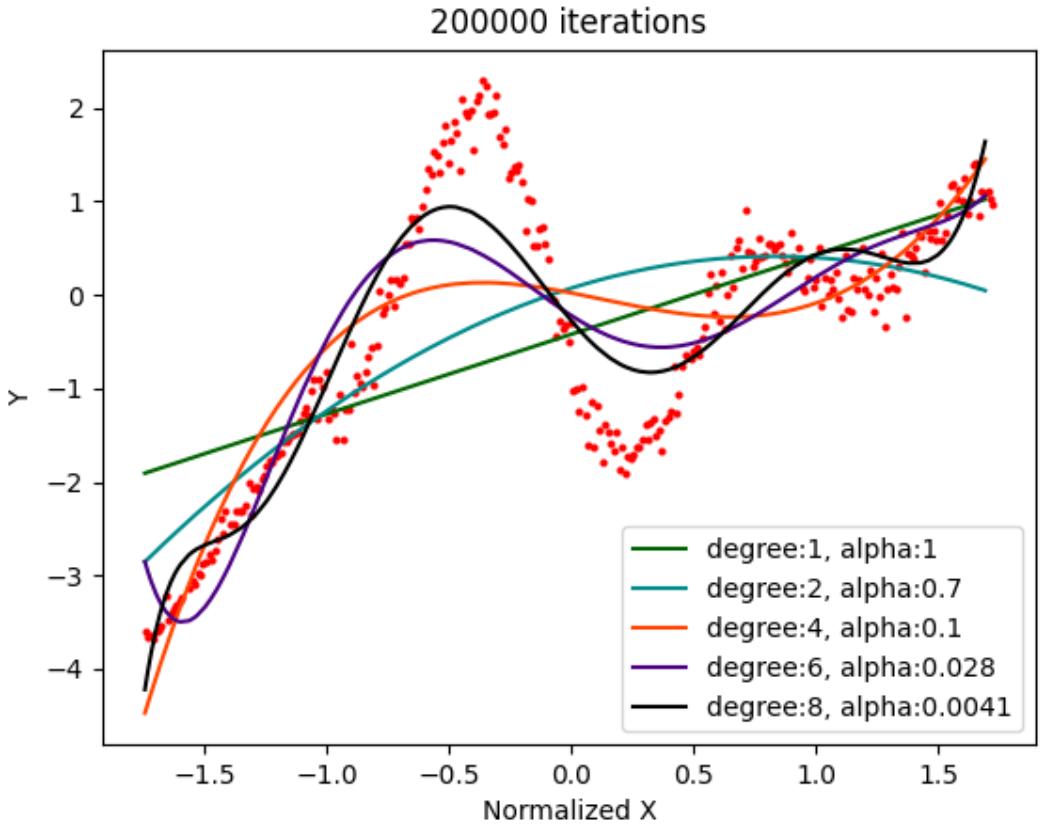
تکرار ٤٠٠٠٠



degree	Error Train	Error Test
1	1.16001	1.73840
2	0.97109	1.45099
4	0.60233	0.91327
6	0.43757	0.62041
8	0.42154	0.63301

برای این تعداد تکرار همانطور که انتظار می‌رفت تغییری در نتایج درجات ۱ ، ۲ و ۴ مشاهده نمی شود؛ زیرا در مرحله قبل کاملا همگرا شده بودند. نتایج درجه ۶ اندکی بهبود یافته اند و خطای درجه ۸ به مقدار زیادی کاهش یافته است؛ اما همچنان برتری خاصی نسبت به درجه ۶ ندارد. بنابراین تعداد تکرار را مجدداً افزایش می دهیم.

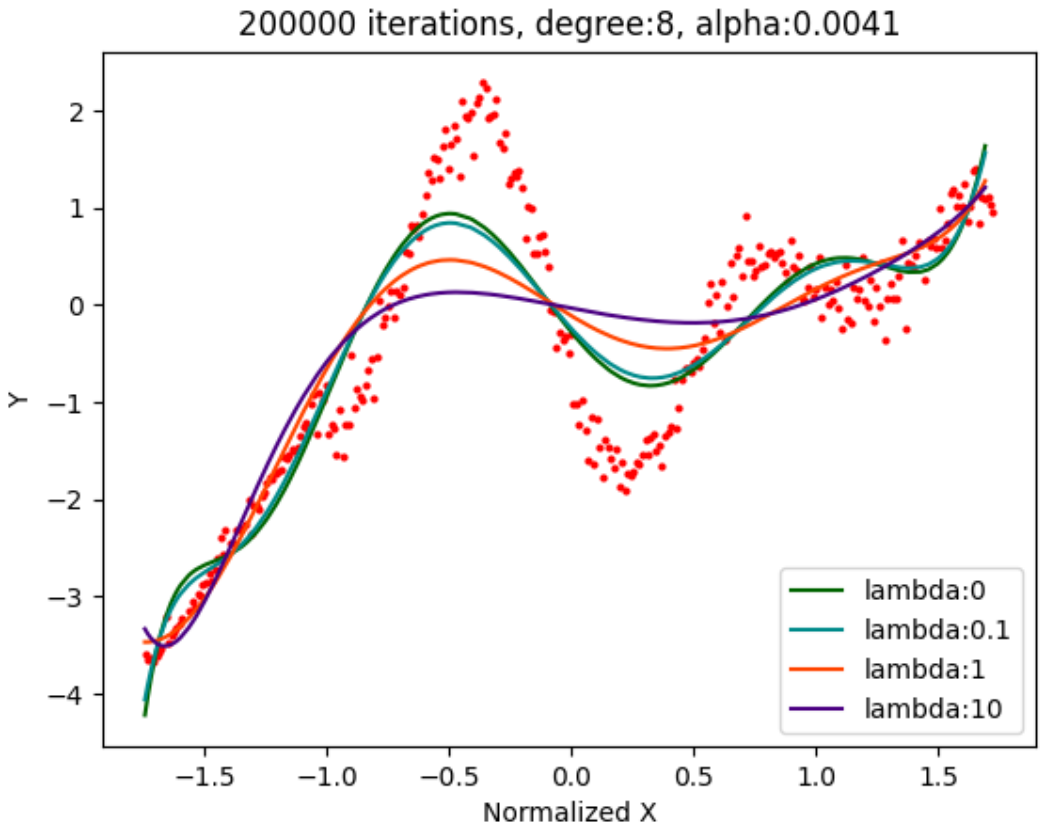
تکرار ۲۰۰۰۰۰



degree	Error Train	Error Test
1	1.16001	1.73840
2	0.97109	1.45099
4	0.60233	0.91327
6	0.43757	0.62042
8	0.28080	0.37803

در تکرار ۲۰۰۰۰۰ ، تابع درجه ۸ نسبتا همگرا شده و خطای آن نسبت به قسمت قبل و همچنین در مقایسه با درجات پایین تر، به شدت کاهش یافته است. در سایر درجات مطابق انتظار تغییری وجود ندارد.

۴) بهترین نتایج بخش قبلی مربوط به درجه ۸ و ۲۰۰۰۰۰ تکرار را برای لامبداهای متفاوت رسم می کنیم. کد مربوط به این بخش در فایل Q4.py قراردارد.



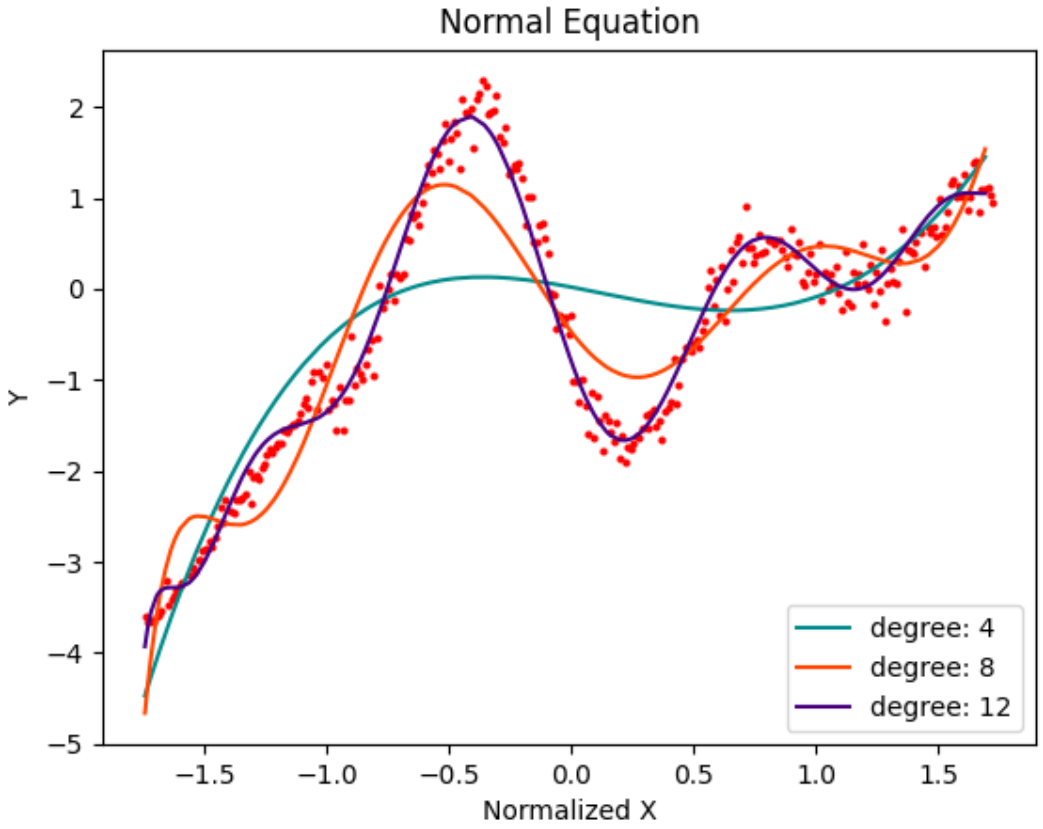
Lambda	Tetha0	Tetha1	Tetha2	Tetha3	Tetha4	Tetha5	Tetha6	Tetha7	Theta8
0	-0.27150	-3.08923	2.54396	6.78791	-4.08453	-3.64898	1.79126	0.64307	-0.24599
0.1	-0.23609	-2.77758	2.17365	5.98240	-3.48616	-3.107	1.48241	0.53769	-0.19624
1	-0.10771	-1.52110	0.82992	2.77602	-1.33273	-0.96442	0.38086	0.12248	-0.02043
10	-0.03238	-0.48224	0.09602	0.60596	-0.25686	0.2989	-0.13032	-0.09871	0.05628

مطابق انتظار با افزایش لامبدا(ضریب رگولاریزیشن)، شکل نمودار ملایم تر شده؛ و ضرایب تتا در مجموع کاهش یافته اند. اگرچه در بعضی ضرایب با افزایش لامبدا، موارد افزایشی هم دیده می شود که با توجه به مصالحه بین ضرایب، منطقی است.

Lambda	Error Train	Error Test
0	0.28080	0.37803
0.1	0.29665	0.41528
1	0.41731	0.63347
10	0.55306	0.87627

با توجه به شکل و نتایج نمودار درجه هشت، واضح است که مشکل بیش برازش وجود نداشته است. بنابراین افزایش ضریب لامبدا، باعث بهبود خطای داده های آزمون نشده است. این نتیجه در جدول بالا قابل مشاهده است. در مجموع به نظر می رسد؛ برای درجه ۸ در این مسئله، نیازی به استفاده از رگولاریزیشن نیست.

۵)کد مربوط به این بخش در فایل normal.py قرار دارد.برای درجه ۴، ۸ و ۱۲ معادله نرمال را اجرا می کنیم. نتایج در نمودار زیر ارائه شده است.

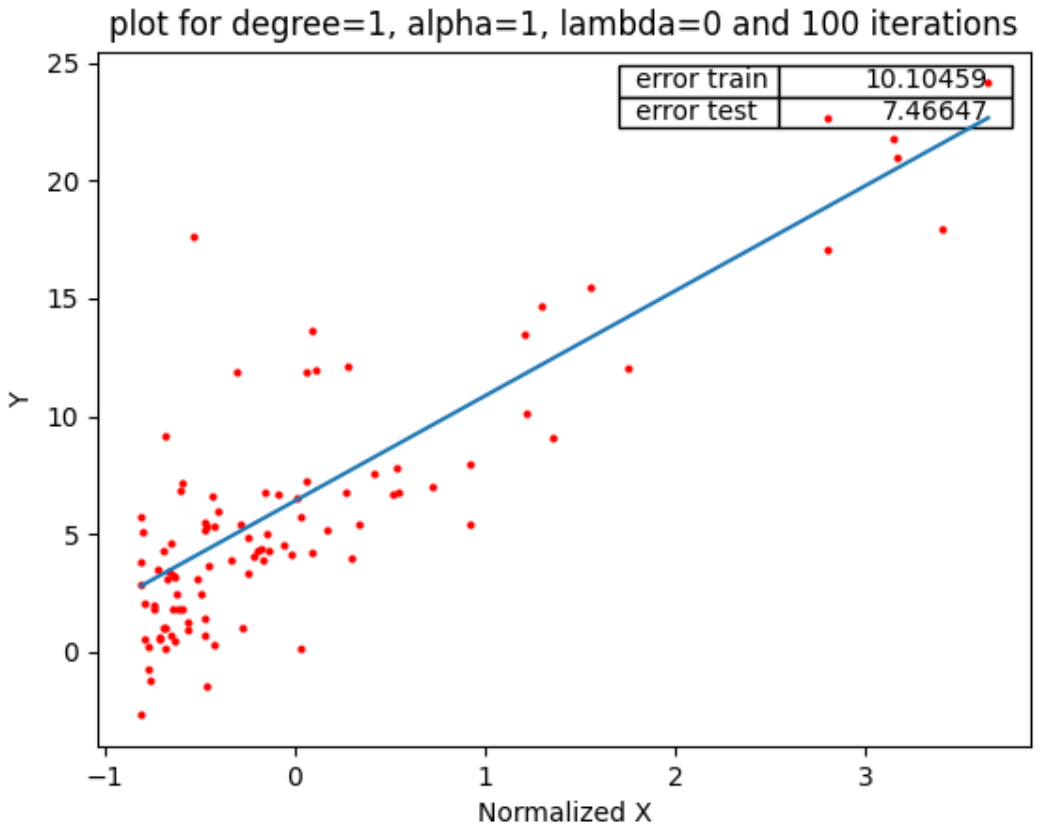


degree	Error Train	Error Test
4	0.60233	0.91327
8	0.26082	0.32514
12	0.04232	0.04005

همانطور که مشاهده می شود معادله درجه ۱۲ تا حد بسیار زیادی خطا را کم کرده است. همچنین نتایج درجه های ۴ و ۸، با خروجی های روش گرادیان نزولی مطابقت دارد. نکته‌ی مهم این روش سرعت بسیار بالا نسبت به روش گرادیان نزولی است. نکته قابل توجه دیگر این است که خطای داده های آزمون از داده های آموزش در درجه ۱۲ کمتر است. البته این اتفاق با توجه به پراکندگی داده ها، کاملا محتمل است.

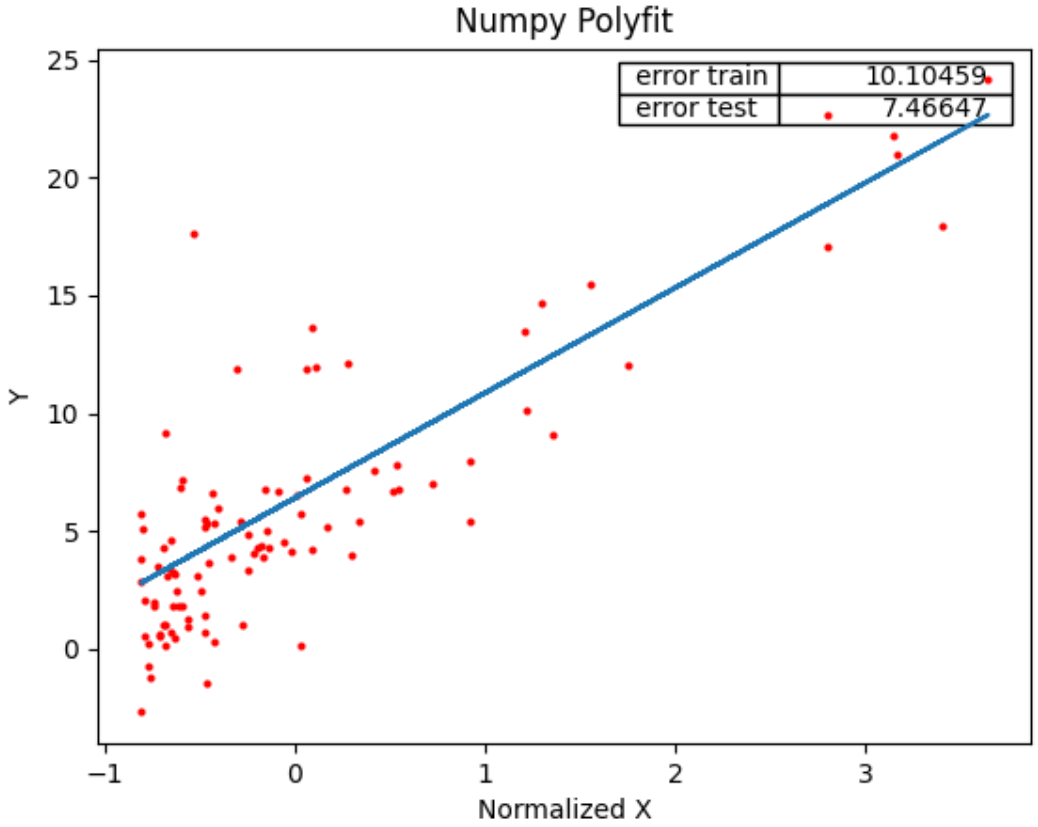
سوالات پیاده سازی: بخش دوم

۱)کد مربوط به این بخش در فایل sec.py قرار دارد. جهت نمایش بهتر خروجی ها اندکی تغییرات در تابع اصلی اعمال شده است. داده ها نرمال شده و خروجی مربوط به آموزش و آزمون روی نمودار مشخص شده است.



در رگرسیون خطی، همگرایی با تکرارهای بسیار کم اتفاق می افتد. نکته قابل توجه، کمتر بودن خطای داده های آزمون نسبت به آموزش است؛ بنابراین به نظر می رسد تعداد داده های نوییزی در مجموعه آزمون کمتر است.

۲) کد مربوط به این بخش در فایل sec2-2.py قرار دارد. از توابع کمکی فایل اصلی برای نرمال سازی و جداسازی داده های آموزش و آزمون استفاده شده است. همچنین برای برازش خطی از دستور polyfit کتابخانه numpy استفاده شده است.



مطابق انتظار نتایج پروژه انجام شده و کتابخانه آماده یکسان می باشد.