

دانشگاه صنعتي امیرکبیر  
(پلی تکنیک تهران)

دانشكده علوم کامپیوتر و ریاضی

پروژه اول یادگیری ماشین

نگارش

سید حسین محمدی

مدرس

دکتر اکبری

آبان 1401

چكيده

بخش اول پاسخ به 4 سوال تشریحی در رابطه با توابع هزینه، بهینه سازی یک مساله طبقه بندی و تشخیص تاثیر واریانس و بایاس در مدل با کمک شکل میباشد.

بخش دوم توضیحات در رابطه با یادگیری فعال (انتخاب داده های یا همان سطر ها آموزشی) در مدل خطی به همراه جزئیات دیکر میباشد.

بخش سه توضیحات مربوط به تمرین پیاده سازی برداری مدل رگرسیون خطی به همراه کد یافتن بهترین پارامتر برای یک داده آموزشی و تستی میباشد. در پایان همین روش را با کمک از تابع های کتابخانه scikitlearn انجام میدهیم.

واژه‌های کلیدی:

Python, Matrix, Active Learning, Linear Regression, SciKitLearn

|  |  |
| --- | --- |
| فهرست مطالب | صفحه |

[بخش اول: سؤالات تشریحی 4](#_Toc117960957)

[1-1- فرمول به روزرسانی وزن ها و bias 4](#_Toc117960958)

[1-2- آیا گرادیان به یک جواب یکسان و بهینه همگرا میرسد 4](#_Toc117960959)

[2-1- مقدار بهینه Θ با فرض m تا مقدار هدف 5](#_Toc117960960)

[3-1- موارد خواسته بر نمودار 5](#_Toc117960961)

[بخش دوم: تمرین مطالعه (Active Linear Regression) 6](#_Toc117960962)

[2-1 مقدمه 6](#_Toc117960963)

[2-1 تعریف مسئله 6](#_Toc117960964)

[2-1 عملیات مورد نظر 6](#_Toc117960965)

[2-1 بهبود الگوریتم 7](#_Toc117960966)

[2-1 الگوریتم sparce 8](#_Toc117960967)

[2-1 مقایسه نتیجه 8](#_Toc117960968)

[بخش سوم: پیادهسازی 9](#_Toc117960969)

[3-1- تمرین تکمیل فایل linreg.py (به صورت برداری) 9](#_Toc117960970)

[3-2- خروجی آزمایش ها 10](#_Toc117960971)

[3-3- محاسبه خطای و gridsearch برای مدل در فایل نوتبوک 11](#_Toc117960972)

[3-4- محاسبه به کمک کتابخانه SKlearn 12](#_Toc117960973)

# بخش اول: سؤالات تشریحی

## فرمول به روزرسانی وزن ها و bias

باید دقت کرد که این توابع به صورت برداری نوشته شداند.

## آیا گرادیان به یک جواب یکسان و بهینه همگرا میرسد

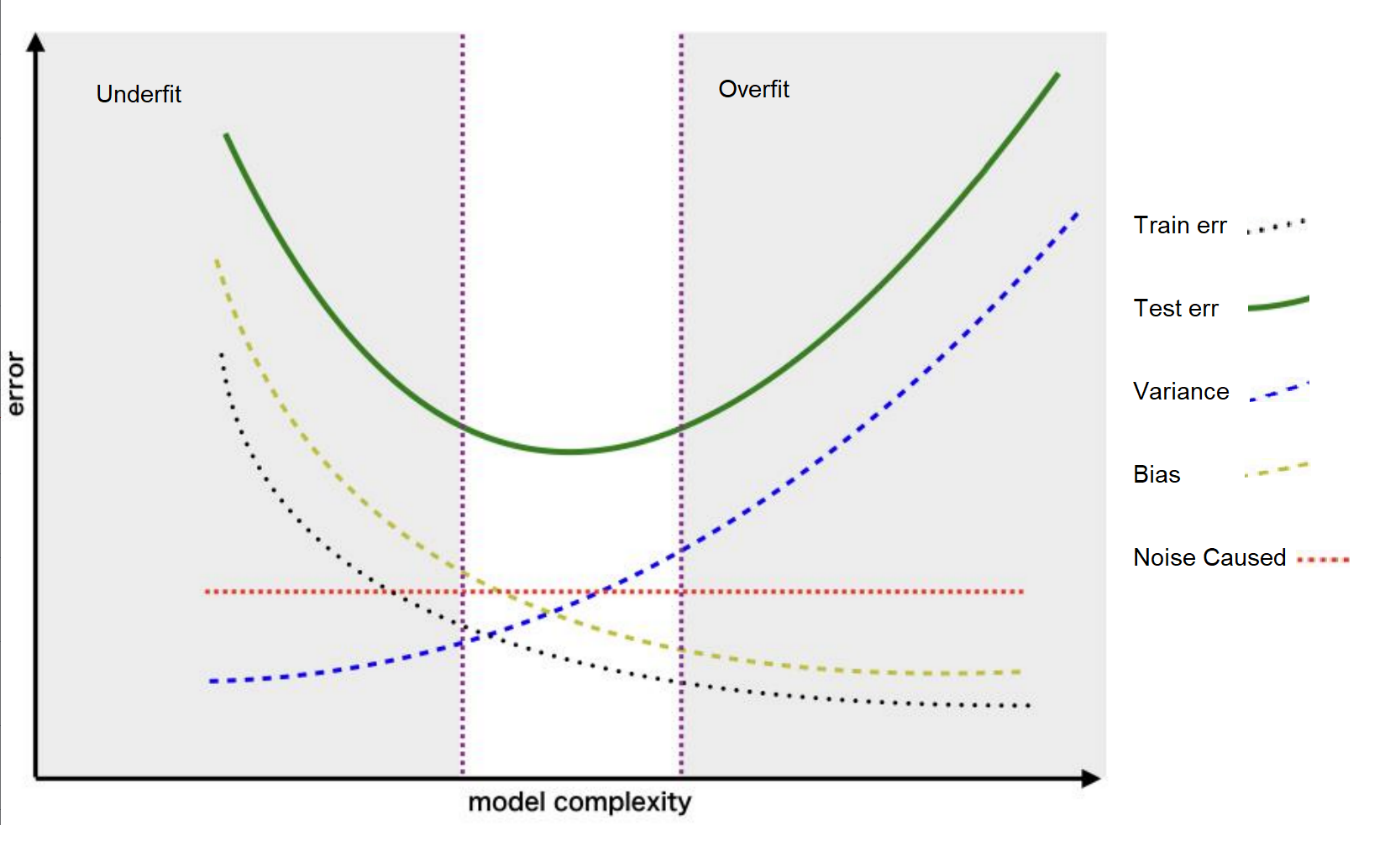
همانطور که در بالا میبینید، تابع گرادیانت، مجموعی از توابع تناوبی هست و مشتق درجه اول و دو معادله هزینه لزوما یک محدب نمیسازد. با اینکه این توابع دارای دور تناوب هستند، لزوما در این فاصله دور تناوب به یک جواب نخواهیم رسید زیرا که فضای آن پر هست از مینیموم های محلی. این مسئله هم به این معنا هست که لزوما به جواب بهینه نیز همگرا نمیشود (زیرا برخی از این نقاط مینیمم بالاتر از بهینه هستند) و حتی ممکن هست در نقاط تعدی گیر کند.

## مقدار بهینه Θ با فرض m تا مقدار هدف

باید تلاش کنیم مقدار زیر را بدست آوریم.

برای حداکثر کردن آن، مقدار مشتق را برابر با صفر میکنیم. در نهایت Θ بهینه برابر است با:

## موارد خواسته بر نمودار

**

# بخش دوم: تمرین مطالعه (Active Linear Regression)

## مقدمه

در آموزش مدل خیلی وقت ها نیازمند قدرت پردازشی , بالا هستیم. دلیل این مساله وابسته به نوع مدل و حجم داده دارد. در پی کاهش این هزینه، بسیاری روی به استفاده از مدل های ساده تر ولی قدرتمند همچون رگرسیون خطی آوردند (حتی برای داده های غیر خطی). در کنار آن تلاش زیادی برای کاهش تعداد ویژگی ها ورودی و حذف داده های نامربوط نیز صورت میپزیرد. در این گزارش، یک روش برای کمک به مساله هزینه در رگرسیون خطی را شرح میدهیم.

کلمه Active اشاره به انتخاب هوشمند داده از کل داده دارد به گونه ای که خطای مدل را کم کند. منطق آن این هست که به دلیل وجود مقدار زیاد اطلاعات تکراری در ویژگی ها (که باعث نویز در مدل خطی میشود) و هزینه پردازشی آن، برخی از داده ها را ابتدا برای تمرین انتخاب میکنیم.

یک دلیل دیگر مسائلی مانند نبود امکان جمعآوری داده به صورت راحت و یکجاست. برای مثال یک شرکت تلاش برای بهبود تبلیغ محصول خود و دریافت بازخورد آن تبلیغ هست. هزینه تبلیغ و پیگیری آن برای هر فرد بالاست اما میتوان از داده های افرادی که قرار هست تبلیغ را ببینند، فهمید که آیا بازخورد آنها به بهبود مدل کمک میکند یا خیر.

## تعریف مسئله

هدف انتخاب k داده از X به گونه ای است که بدست آمده از این داده ها، بهینه باشد و خطا را به نسبت انتخاب های دیگر به حداقل برساند. برای k هایی که از تعداد ویژگی بیشتر هستند، باید مقدار زیر را بدست آورد ( پارامتر بهینه برای همه داده هاو Σ ماتریس covariance هست):

میتوان از Gauss-Markov Theorem استفاده کرد و مقدار بهینه OLS را برای انتخاب کرد.

## عملیات مورد نظر

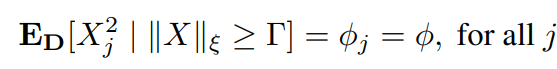
فرض میکنیم داده میانگین صفر دارد. ما تلاش داریم همبندی درون داده را کم کنیم که هم معنی با ماکزیمم کردن کوچکترین مقدار ویژه ماتریس ههمبندی هست. یک راه فکر کردن به این مسئله از دیدگاه PCA/SVD هست و اینکه اگر همبندی وجود نداشته باشد، آنگاه مقادیر ویژه اعداد مشابه هستند و بزرگترین حالت هستند. سپس مقادیر با نورم بزرگ را انتخاب میکنیم.

در پی این هدف، فرض کنید X’ داده های با همبستگی هستند. در اینصورت با انجام عمل زیر، این همبستگی را کم میکنیم:

%

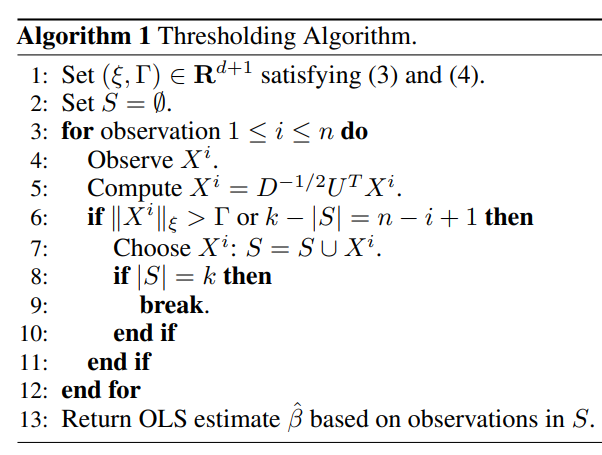
حال Г و ɛ را به گونه ای مییابیم که با شروط زیر همخوانی داشته باشد، سپس داده هایی که نورم وزن دار ɛ آن ها بالاتر از Г را انتخاب میکنیم.

نورم وزن دار: 

شرط: 

برای هر j، مربع داده هایی که مقدار نورم وزندار بزرگتر از Г را دارند را میانگین میگیریم. اگر این مقادیر با هم (به زبان دیگر با عددی مثل ∅ باشد)، آنگاه این Г و ɛ صادق هستند.

\*توجه کنید که داده ها به گونه ای انتخاب شود که تعداد آن به k (تعدادی که میخواهیم انتخاب کنیم) برسد.



## بهبود الگوریتم

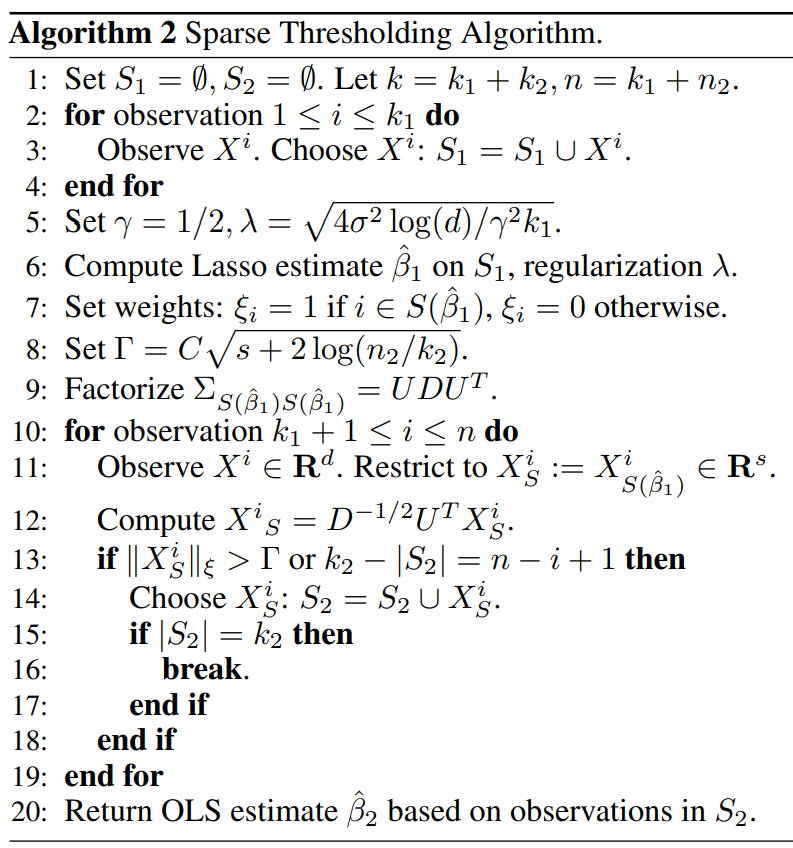
برای بهبود عملکرد پردازشی این روش، میتوان Г به صورت داینامیک حساب کرد. برای اینکار مقادیر Г و ɛ را در هر مرحله، برای هر سطر داده پیموده شده، حساب میکنیم.

به دلیل پیمایش قدم به قدم این روش، میتوان آن را به صورت آنلاین استفاده کرد که به این معناست که کل داده نیاز نیست و میتوان داده را طی زمان در بازه متفاوت پردازش کرد. (که به معنای online بودن این روش نیز هست)

## الگوریتم sparce

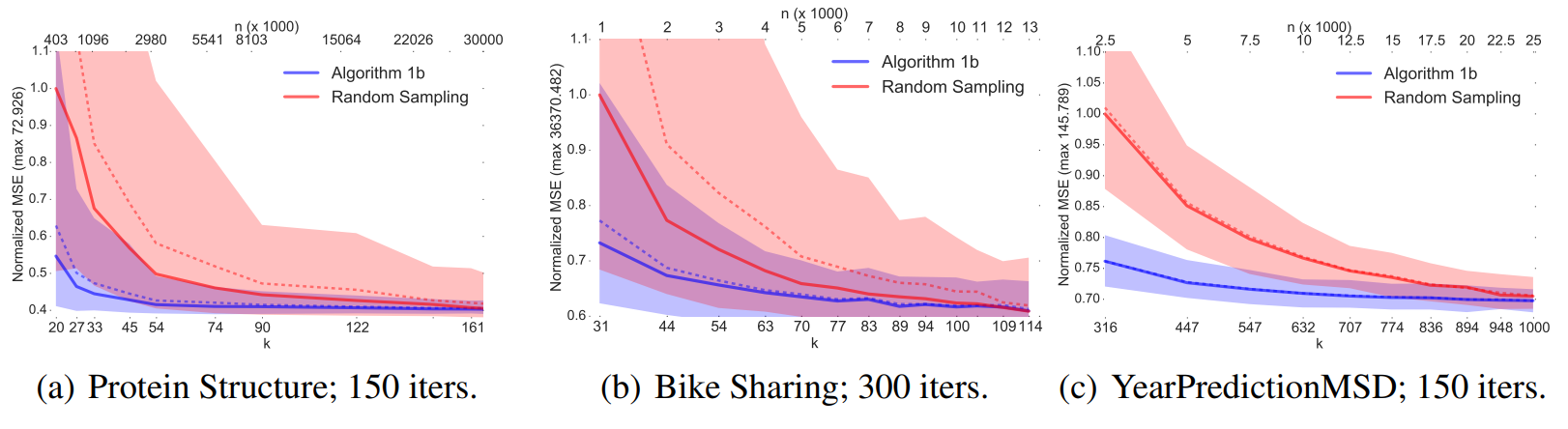
یک مسئله قابل توجه این هست که با افزایش تعداد ویژگی ها، توان پیشبینی آن کاهش میابد. این مشکل به صورت کلی به دلیل ایجاد داده هایی است که مقادیر عادی دارند اما از نظر ترکیب ویژگی ها یک outlier محسوب میشود. این مشکل با افزایش تعداد داده ها و کاهش کوواریانس داخلی آنها، تشدید میشود.

برای حل این مشکل از الگوریتم زیر استفاده میکنیم.



## مقایسه نتیجه

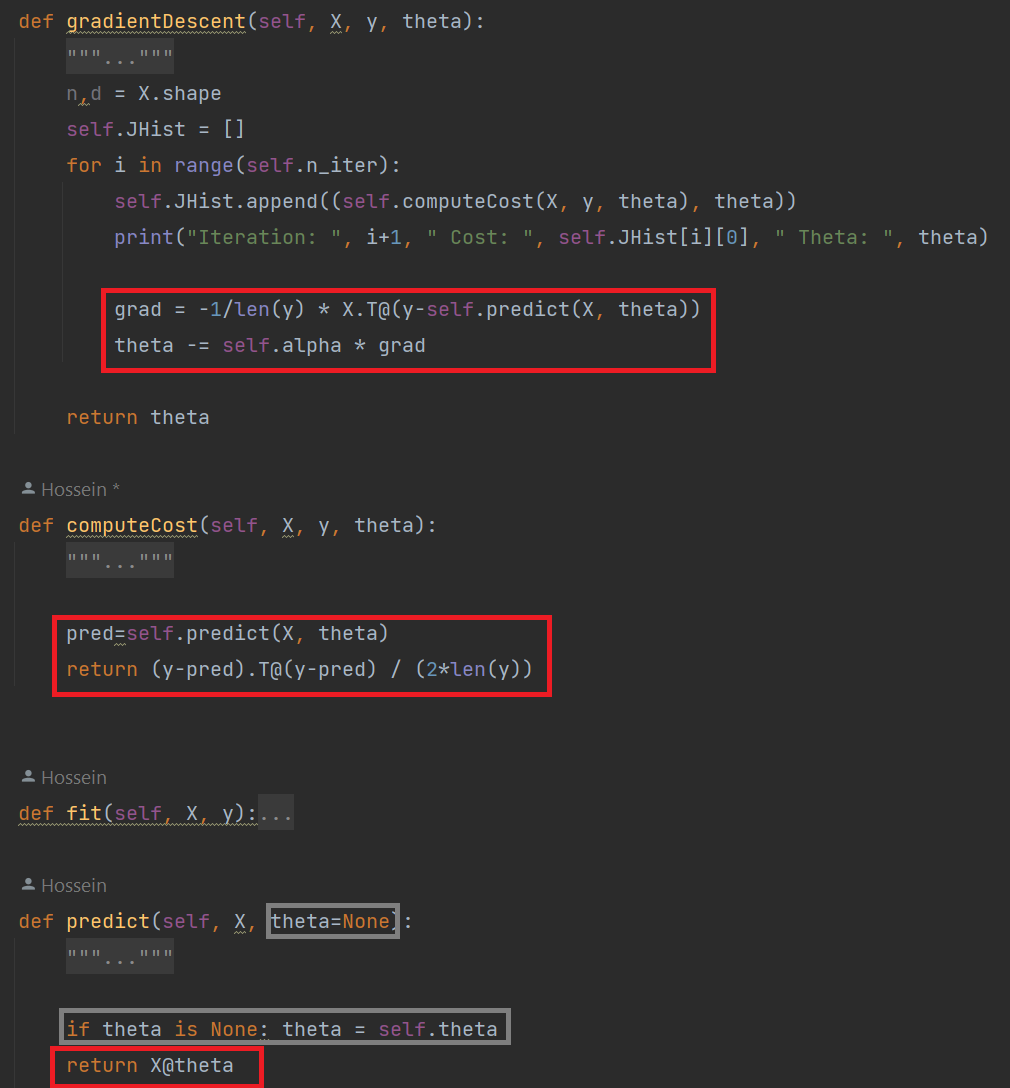
همانطور که در عکس میبینید، این روش انتخاب با انتخاب تصادفی k داده مقایسه شده و برای همه مقادیر، عملکرد بهتری دارد.



# بخش سوم: پیادهسازی

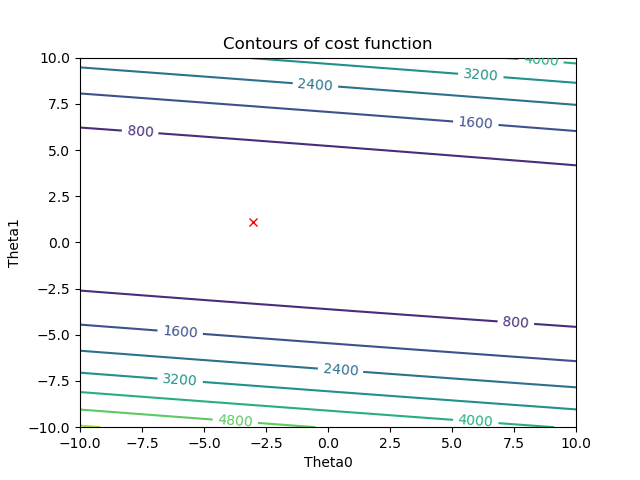
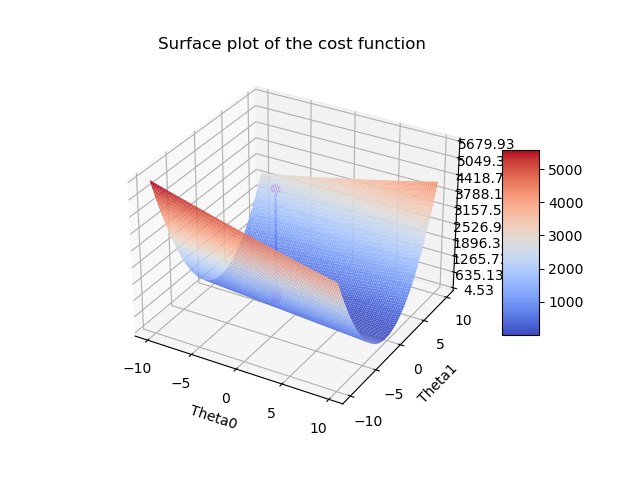
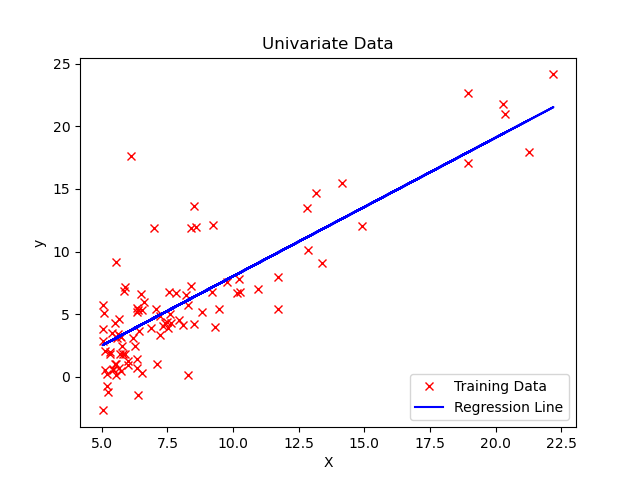
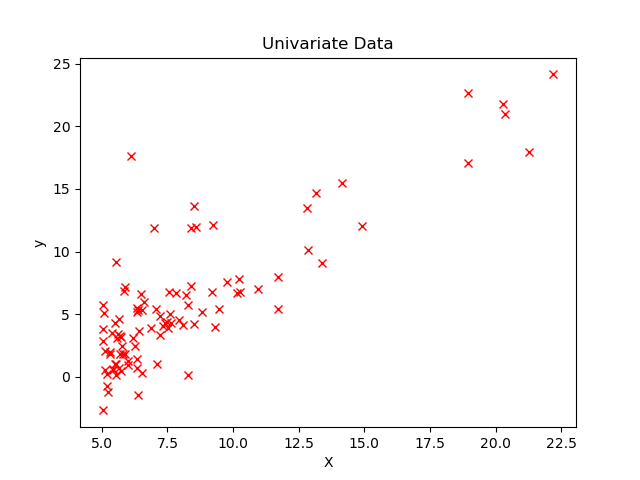
## تمرین تکمیل فایل linreg.py (به صورت برداری)

کد تغییر داده شده در کادر قرمز و طوسی:



یک تغییر اضافه، ایجاد ورودی Θ برای تابع predict میباشد که بتوان از آن در دیگر مکان ها استفاده کرد. علامت @ به معنای ضرب ماتریسی (و برداری) میباشد. A.T ماتریس ترانهاده A میباشد.

## خروجی آزمایش ها



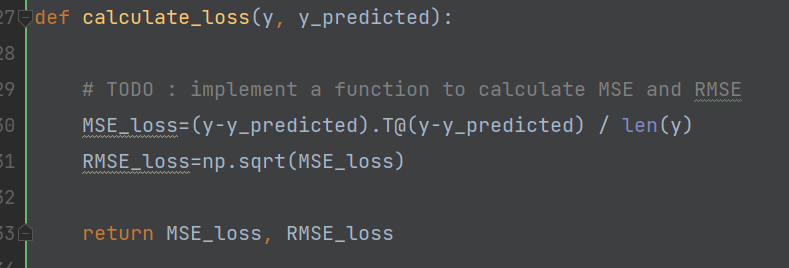
پراکندگی داده به شدت skew شده است و مدل خطی که به نقاط outlier و نویز حساس هست، نتوانسته اکثر داده را به آن شکل که باید در نظر بگیرد و داده های دور تر (x>15) توجه بیشتر از نیاز شده است. لذا ممکن هست داده های مورد توجه ما همان نقاط باشند، پس آن ها را حذف نمیکنیم.

خطا در قدم اول به شدت کاهش میابد و سپس به آرامی به مقدار بهینه میرسد.

برای آزمایش تک متغییر، به خطای 4.5458 میرسیم. تصاویر تولیدی در بالا موجود هستند.

برای آزمایش چند متغییر، به خطای 2,043,280,000 میرسم.

## محاسبه خطای و gridsearch برای مدل در فایل نوتبوک



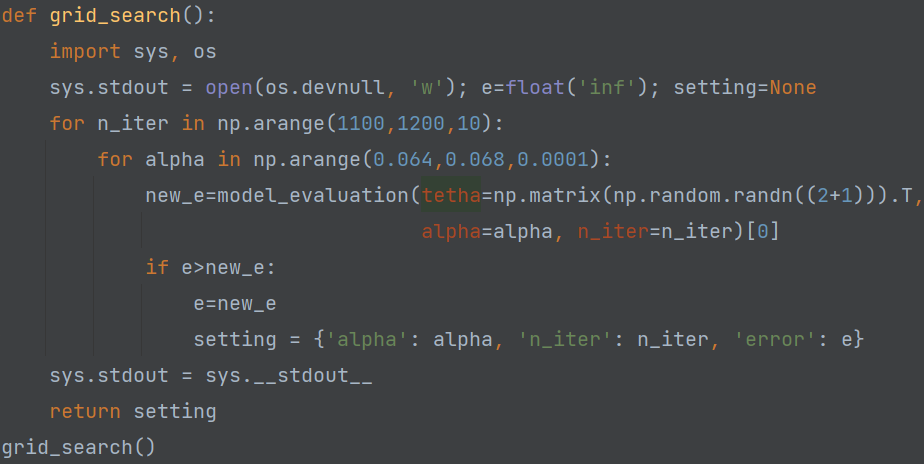
خطا برای مدل با پارامتر و هایپر پارامتر های بهینه شده برابر هست با:

RMSE: 48021, MSE: 2,306,007,735.7

با تکمیل این تابع، میتوان از تابع model\_evaluation برای انجام grdesearch استفاده کرد.

برای راحت تر کردن اینکار، یک تغییر دیگر به تابع model\_evaluation میدهیم که در آن مدل با داده اصلی آموزش و سپس با داده تست آزمایش میشود.

حال یک تابع برای انجام gridsearch مینویسیم. مقادیر پارامتر ها بسیار زیاد بودند و به همین دلیل در طی چند مرحله، بازه و قدم آن ها را کوچکتر میکنیم.

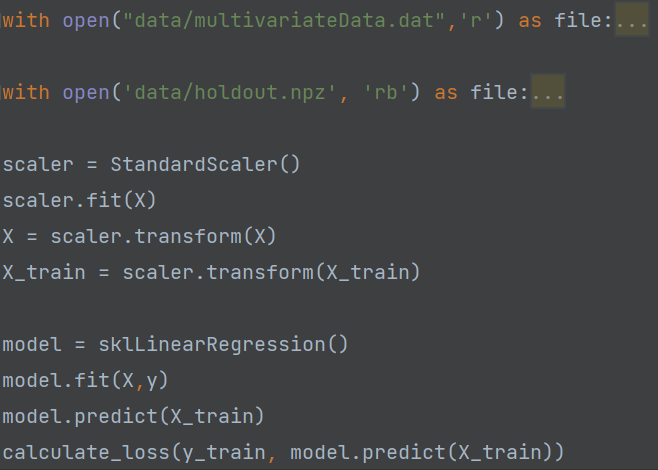


در نهایت، بهترین پارامتر ها به همراه خطا آن برابر میشود با:

alpha: 0.0747, n\_iter: 1026

test\_error: 2,306,007,740

## محاسبه به کمک کتابخانه SKlearn



با کمک از این کتابخانه، دیگر نیاز به پیدا کردن دستی پارامتر های بهینه نیست و کل برنامه به شدت سریع تر هست. با این روش، خطا تست برابر است با 2،306،007،735.7 که فقط 5 واحد با مقدار قبل فاصله دارد (%3.722e-13)



Amirkabir University of Technology  
(Tehran Polytechnic)

Department of Computer Science and Math

Machine Learning Assignment #1

By

Seyed Hossein Mohammadi

Taught by

Dr. Akbari

Oct of 2022