

باسمه تعالی



دانشگاه صنعتی شریف

دانشکده مهندسی برق

## درس یادگیری ماشین

گزارش تمرین شماره دو

علی محرابیان 96102331

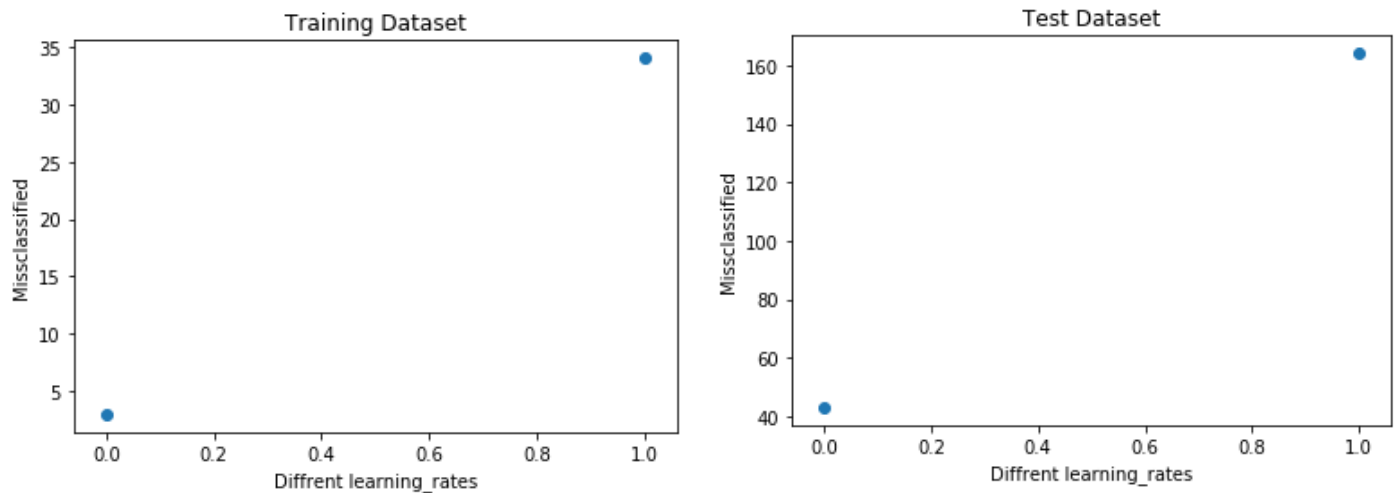
استاد: دکتر صالح کلیبر

پاییز 1398



سوال 1:

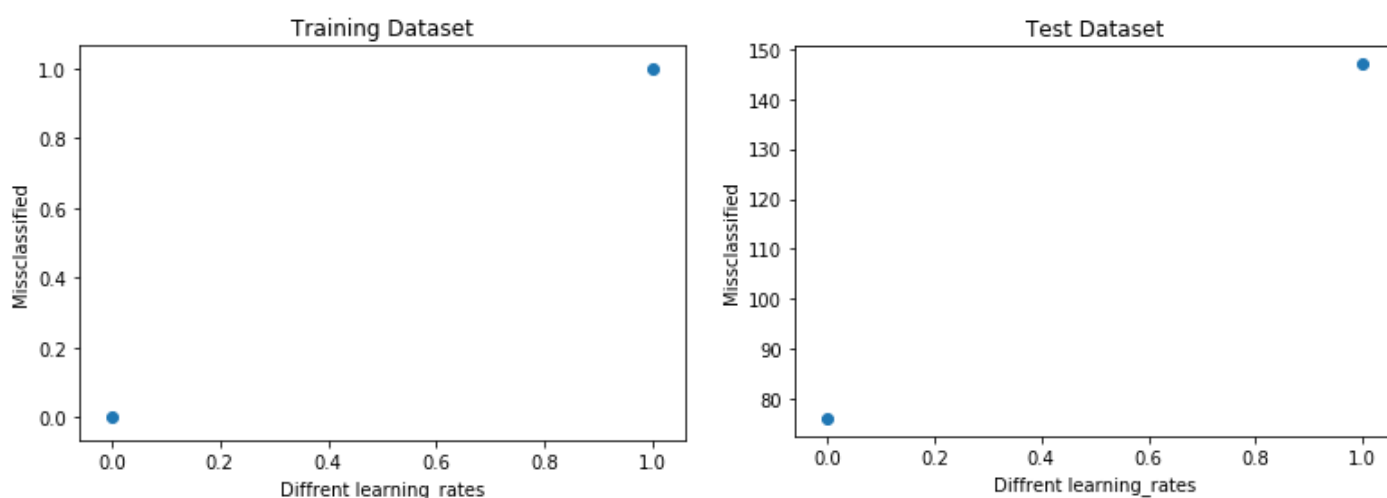
در ابتدا روش batch mode را پیاده سازی کرده و پس از استفاده از تمام داده ها برای آپدیت وزن هایی که در ابتدا به صورت رندوم انتخاب کرده ایم، تعداد خروجی هایی که برچسب اشتباه می خورند را رسم می کنیم. الگوریتم زمانی به انتها می رسد که تعداد خروجی هایی که برچسب اشتباه می خورند، دیگر تغییر نکند.



مشاهده می شود که برای  $\text{learning rate} = 0.1$ ، یادگیری بهتری حاصل می شود که البته انتظار هم میرفت، چرا که آپدیت شدن وزن ها خیلی بهتر انجام می شود در حالی که با  $\text{learning\_rate} = 10^{-7}$ ، آپدیت شدن وزن ها به کندی انجام می شود. برای مقدار 0.1، با خطای 14% مواجه هستیم در حالی که برای مقدار  $10^{-7}$  با خطای 54% مواجه می شویم.



در روش online mode، با بررسی هر نمونه وزن ها آپدیت می شوند. تعداد خروجی هایی که برچسب اشتباه می خورند به صورت زیر است.

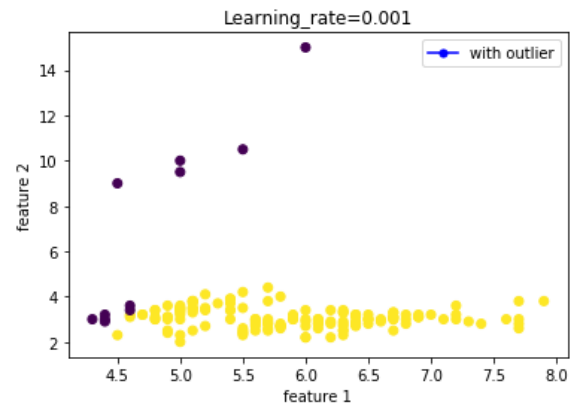
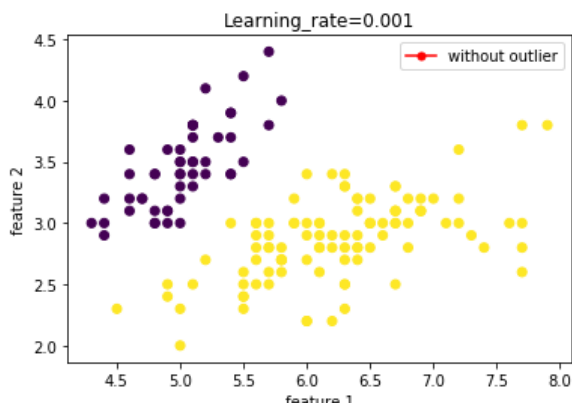
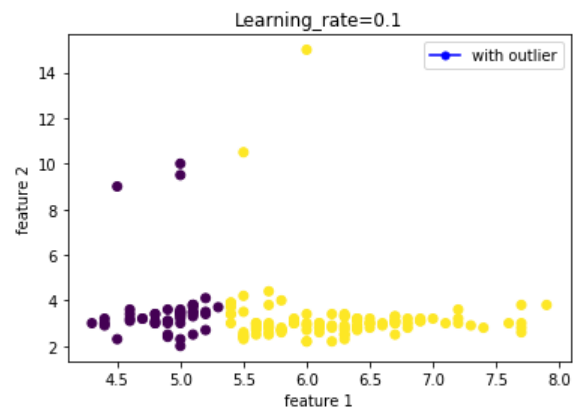
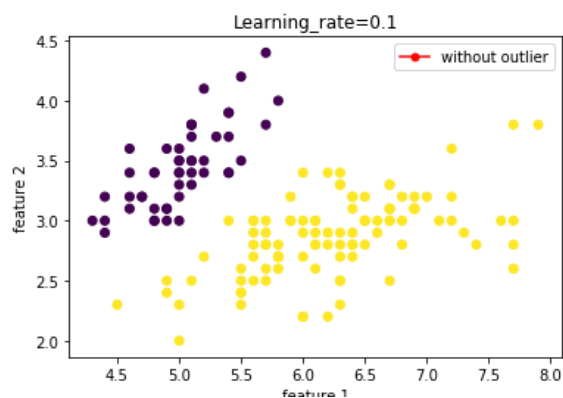


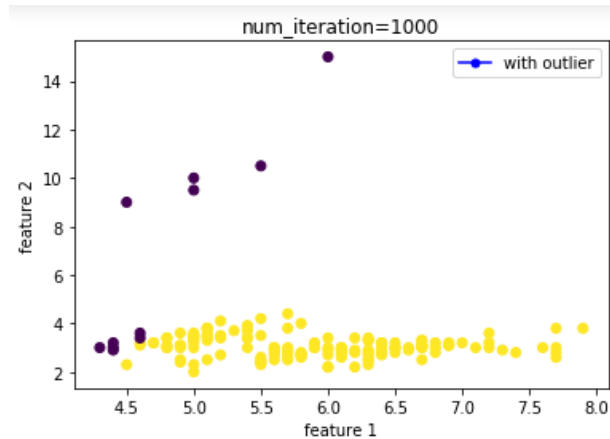
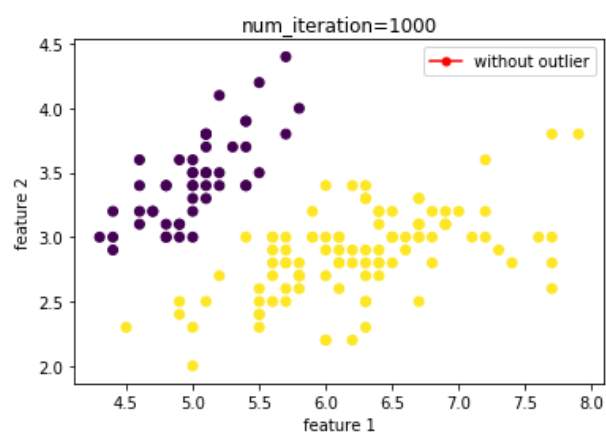
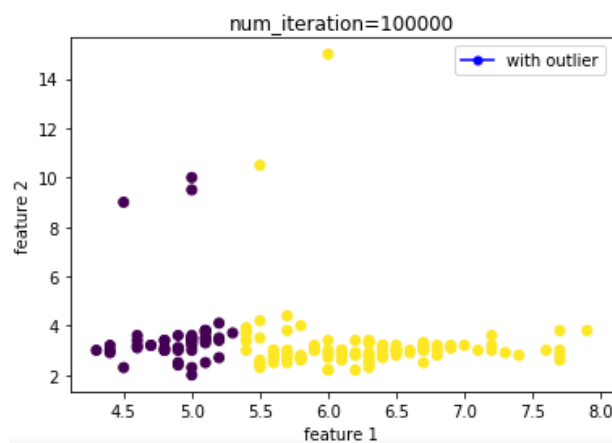
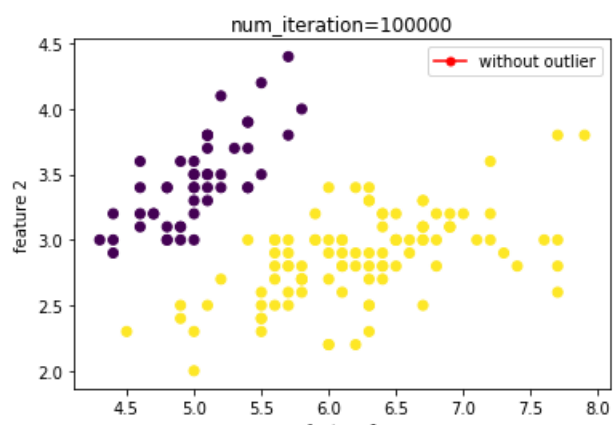
در این حالت نیز برتری  $0.1$  learning\_rate نسبت به  $10^{-7}$  قابل مشاهده است. برای مقدار  $0.1$  خطای 25% و برای  $10^{-7}$  خطای 50% داریم. پس می توان گفت که عملکرد روش batch نسبت به online بهتر است. به طور کلی برای این که خروجی بهتری داشته باشیم، در ابتدای کار باید learning\_rate مقدار بیشتری داشته باشد تا وزن ها با نرخ خوبی آپدیت شوند و هرچه گذشت باید این مقدار را کاهش دهیم تا آپدیت وزن ها به طور کنترل شده انجام شود. اگر learning\_rate مقدار زیادی داشته باشد، ممکن است به جواب بهینه نرسیم و اگر مقدار کمی داشته باشد، برای رسیدن به خروجی مطلوب به گام های زمانی زیادی نیاز خواهیم داشت و روند پیشرفت به کندی انجام می شود.



سوال 2:

همان طور که می دانیم، استفاده از MSE مناسب نمی باشد، چرا که چون با تابع sigmoid سروکار داریم، تابع loss ما تابع محدبی نمی باشد. دلیل دیگر این است که MSE، داده های misclassified را به اندازه کافی مجازات نمی کند در حالی که انتخاب بسیار خوبی برای مسائل regression است. بنابراین می توان گفت که چون در مسائل classification مرز تصمیم گیری بزرگتری داریم، استفاده از cross-entropy مناسب تر است. برای تشخیص outlier، در ابتدا داده ها را به صورت صوری رسم کردیم و نقاط پرت را از مجموعه داده ها حذف کردیم. نمودار داده های طبقه بندی شده برای نرخ های یادگیری و گام های زمانی مختلف به صورت زیر است.





در هر دو حالت به وضوح اثر حذف outlier دیده می شود. می توان دید که حتی برای نرخ های یادگیری و گام های زمانی کم نیز با حذف outlier، خروجی به خوبی طبقه بندی می شود.

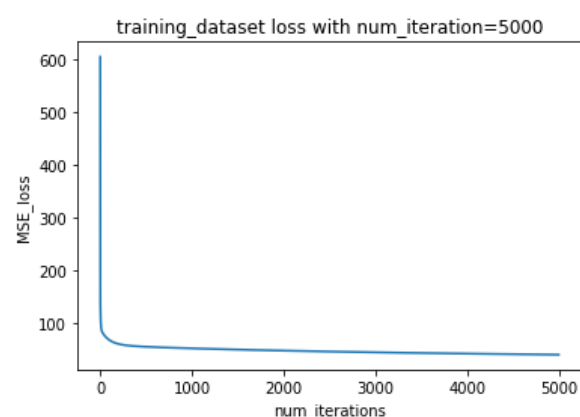
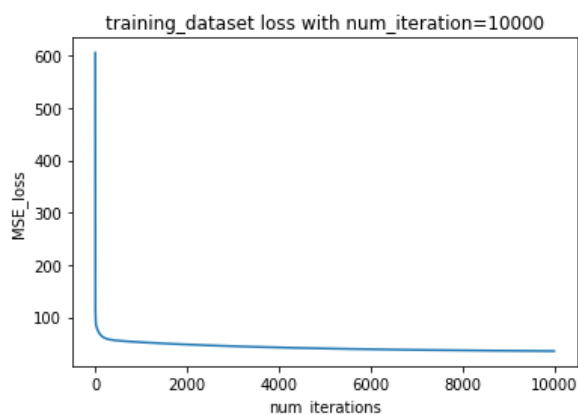
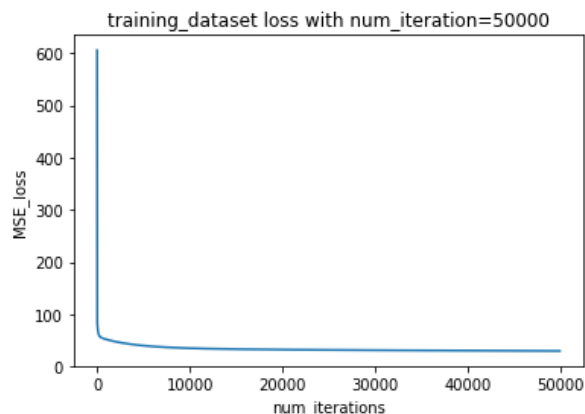
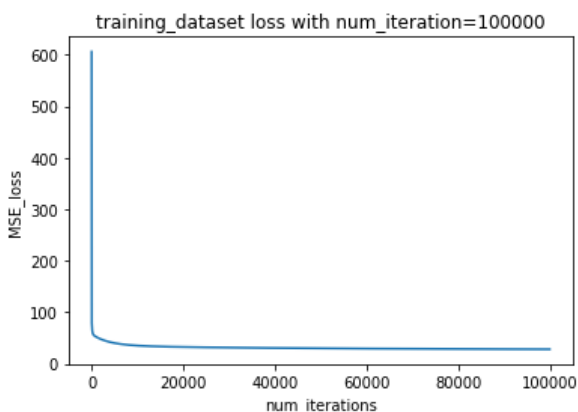


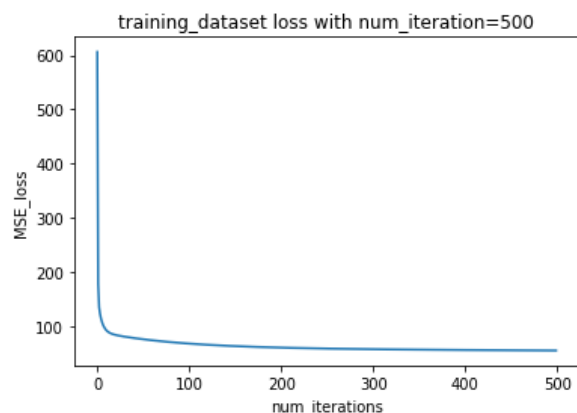
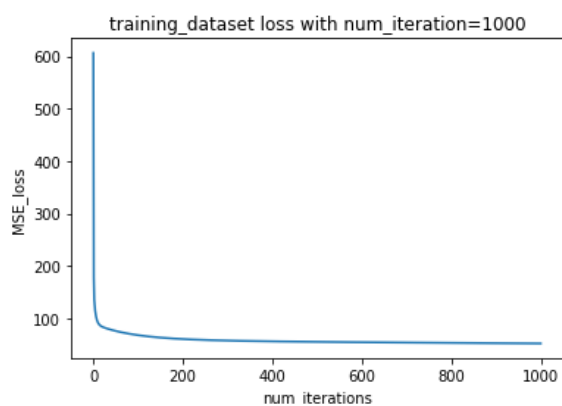
### سوال 3:

MSE مقدار مجازات و خطای بیشتری را به داده هایی که دور تر هستند، اختصاص می دهد. پس از یک منظر می توان گفت که MAE نسبت به داده های outlier، عملکرد بهتری دارد. یکی از معایب MAE، این است که تغییرات گرادیان حتی برای مقدار های کم خطا هم زیاد است که این برای یادگیری خوب نیست. پس می توان نتیجه گرفت که هرکدام در موقعیت های مختلف کاربرد خود را دارند.

الگوریتم Linear regression را با استفاده از MSE loss function پیاده سازی می کنیم.

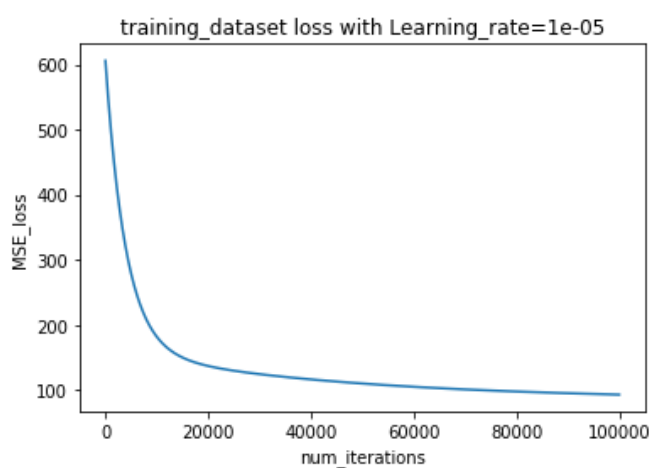
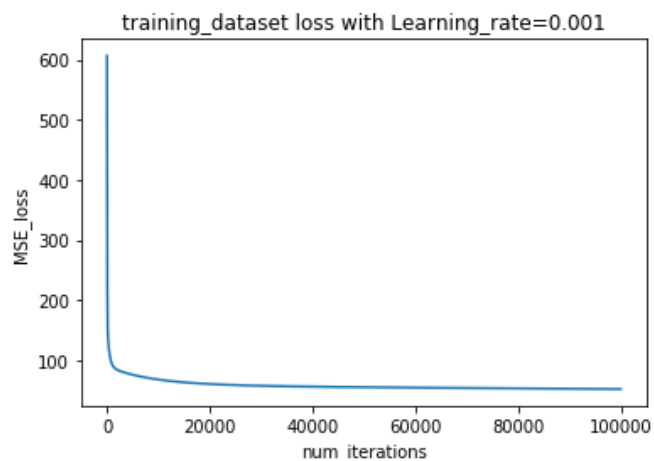
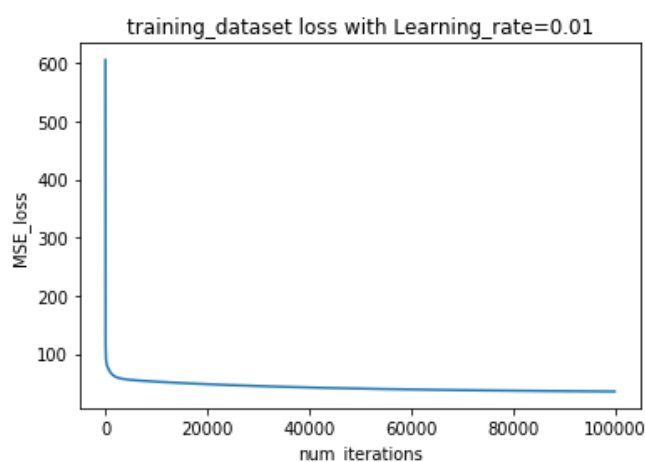
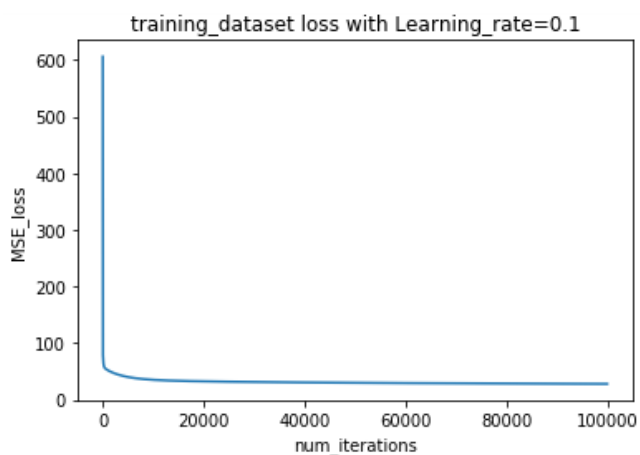
MSE loss function برای گام های زمانی متفاوت به صورت زیر است.

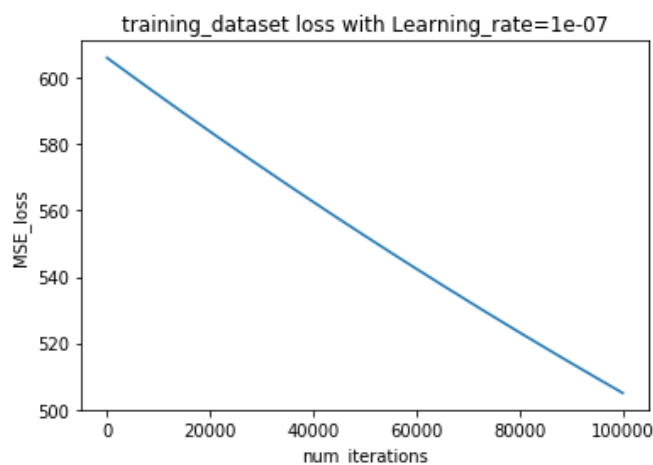
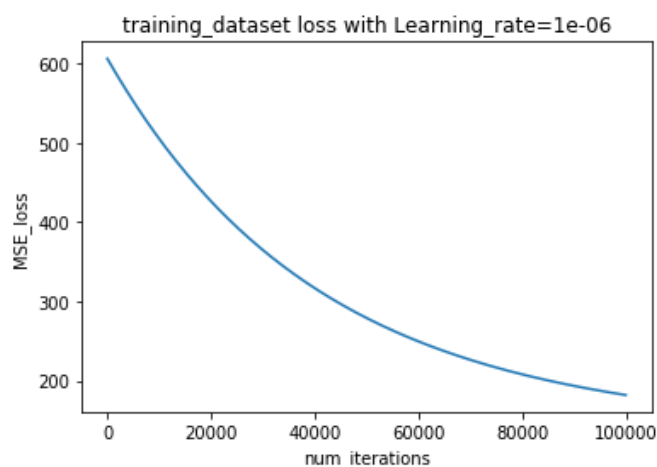




مشاهده می شود که با تغییر گام های زمانی، همچنان خطا با نرخ خوبی کاهش می یابد.

حال MSE loss function را برای Learning\_rate های مختلف رسم می کنیم.





مشاهده می شود که از جایی به بعد، یادگیری بسیار کند شده و عملاً یادگیری حاصل نمی شود. این به این دلیل است که آپدیت شدن وزن ها به کندی انجام می شود. در نهایت خطا برای داده های تست به صورت زیر است.

