گزارش کار تمرین شماره یک درس یادگیری ماشین

محمد لشکری ۴۰۰۱۱۲۰۸۷

۲۶ آبان ۱۴۰۰

۱ نتایج رگرسیون خطی یک متغیره

در فایل linreg.py کد ها به صورت ماترسی و بدون حلقه با استفاده از تابع np.dot پیاده سازی شده است تا محاسبات سریع تر انجام شود. مقادیر مشاهده شده برای تابع هزینه با مقادیر متفاوت هایپرپارامتر ها به صورت زیر است:

$$\theta_0 = [۱۵, ۱۵]^T,\ \alpha = ۰/۱,\ n = ۱۵۰۰ \Rightarrow J(\theta) \simeq Nan \quad (۱.۱)$$

$$\theta_0 = [۱۰, ۱۰]^T,\ \alpha = ۰/۰۱,\ n = ۱۵۰۰ \Rightarrow J(\theta) \simeq ۴/۴۸ \quad (۲.۱)$$

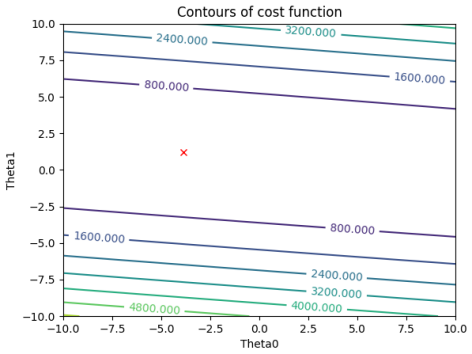
$$\theta_0 = [۵, ۵]^T,\ \alpha = ۰/۰۰۱,\ n = ۱۵۰۰ \Rightarrow J(\theta) \simeq ۶/۶۷ \quad (۳.۱)$$

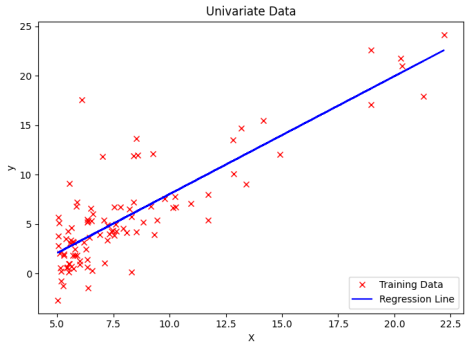
$$\theta_0 = [۱۵, ۱۵]^T,\ \alpha = ۰/۱,\ n = ۲۰۰۰ \Rightarrow J(\theta) \simeq Nan \quad (۴.۱)$$

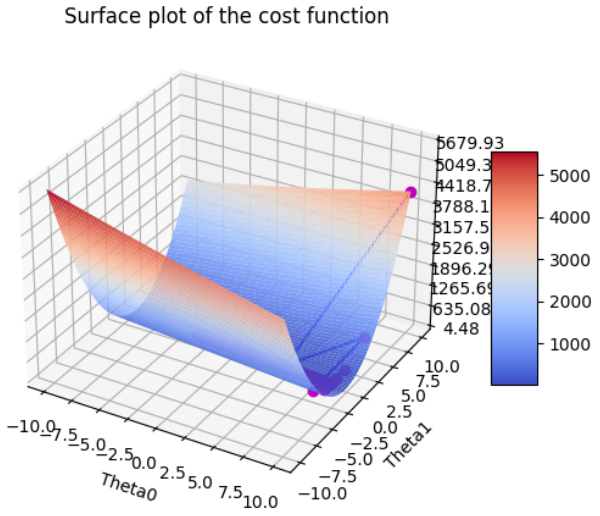
$$\theta_0 = [۱۰, ۱۰]^T,\ \alpha = ۰/۰۱,\ n = ۲۰۰۰ \Rightarrow J(\theta) \simeq ۴/۴۷۷ \quad (۵.۱)$$

$$\theta_0 = [۵, ۵]^T,\ \alpha = ۰/۰۰۱,\ n = ۲۰۰۰ \Rightarrow J(\theta) \simeq ۶/۰۱ \quad (۶.۱)$$

اگر نرخ یادگیری زیاد باشد، میزان نوسانات بالا می‌رود و ممکن است الگوریتم گرادیان کاهشی همگرا نشود که این مطلب، عبارات (۱.۱) و (۴.۱) را توجیه می‌کند. از طرفی اگر نرخ یادگیری خیلی کم باشد سرعت همگرایی پایین می‌آید و باید تعداد تکرار را افزایش دهیم تا مقدار تابع هزینه کاهش یابد که این، گواهی بر نتایج (۳.۱) و (۶.۱) است. اما بهترین نتیجه با نرخ یادگیری مناسب و تعداد تکرار کافی برای $\alpha = ۰/۰۱, n = ۲۰۰۰$ رخ داده است که مقادیر ضرایب رگرسیون خطی برای این دو مقدار برابر $\theta \simeq [-۳/۸۹, ۱/۱۹]^T$ است که $\theta_0 = [۱۰, ۱۰]^T$. همچنین نمودار های مربوط به این مدل را می‌توانید در ادامه مشاهده کنید:







۲ نتایج رگرسیون خطی چند متغیره

نتایج رگرسیون خطی چند متغیره به ازای مقادیر مختلف α و n به شرح زیر است:

$$\alpha = ۰/۱, n = ۲۰۰۰ \Rightarrow J(\theta) \simeq ۲۰۴۳۲۸۰۰۵۰/۶$$

$$\alpha = ۰/۰۱, n = ۲۰۰۰ \Rightarrow J(\theta) \simeq ۲۰۴۳۲۸۰۰۵۰/۶$$

$$\alpha = ۰/۱, n = ۳۰۰۰ \Rightarrow J(\theta) \simeq ۲۰۴۳۲۸۰۰۵۰/۶$$

$$\alpha = ۰/۰۱, n = ۳۰۰۰ \Rightarrow J(\theta) \simeq ۲۰۴۳۲۸۰۰۵۰/۶$$

می‌دانیم تابع هزینه رگرسیون محدب است و همانطور که مشاهده می‌شود بعد از چند تکرار، تابع هزینه ثابت می‌ماند پس مینیمم تابع پیدا شده است اما مقدار آن زیاد است که این امر می‌تواند دو دلیل داشته باشد: ۱.کم بودن تعداد داده ها ۲.عدم برقراری فرض خطی بودن وابستگی. همچنین بردار ضرایب که در هر ۴ بار اجرا تقریباً یکسان بود، به صورت زیر است:

$$\theta \simeq [۳۴۰۴۱۲/۶۶, ۱۰۹۴۴۷/۸, -۶۵۷۸/۳۵]^T$$

۳ خطای مدل روی داده های فایل holdout.npz

با فرض اینکه متغیر های مستقل مجموعه تست را با X_{test} نمایش دهیم، این مجموعه قبل از پیش بینی به صورت زیر استاندار شده است:

$$X_{test} = \frac{X_{test-\mu}}{\sigma}$$

که در آن $\mu = \bar{X}$ و $\sigma = S_X^۲$. در جدول ۱، خطای تست به ازای مقادی مختلف برای هایپرپارامتر ها قابل مشاهده است:

نرخ یادگیری	تعداد تکرار	خطای تست
۰/۱	۲۰۰۰	۱۱۵۳۰۰۳۸۶۷/۸۷
۰/۰۱	۲۰۰۰	۱۱۵۳۰۰۳۸۷۱/۶۳
۰/۱	۳۰۰۰	۱۱۵۳۰۰۳۸۶۷/۸۷
۰/۰۱	۳۰۰۰	۱۱۵۳۰۰۳۸۶۷/۸۷

جدول ۱: خطای داده های تست

خطای تست در هر ۴ بار اجرا تقریباٌ عددی یکسان و بزرگ است که با توجه به بالا بودن مقدار تابع هزینه در انتهای آموزش نتیجه دور از ذهنی نیست. خطای داده های آموزش و تست از هم فاصله دارند و پیچیدگی مدل کم است پس underfit رخ داده است. بنابراین این مدل مطلوب نیست و برای بهبود آن باید تعداد داده ها زیاد شود و پیچیدگی مدل تغییر یابد.