تمرین سری اول یادگیری ماشین

شیوا وفادار ۸۱۰۸۹۹۰۷۴

بخش تئوری در پی دی اف ضمیمه شده و به طور کامل حل شده.

برای بخش پیاده سازی،

۱-الف)

1. بارگذاری و تقسیم داده: داده از مجموعه داده بوستون بارگذاری شده و با استفاده از `train_test_split` به دو قسمت آموزش و آزمون تقسیم می‌شود.

2. نرمال‌سازی: تابع `normalization` برای مقیاس دادها به بازه [0، 1] تعریف شده است. این می‌تواند به گرادیان کاهشی کمک کند تا سریعتر همگرا شود.

3. آموزش مینی-بچ: تابع `mini_batch_train` گرادیان کاهشی مینی-بچ را انجام می‌دهد. این تابع بر روی اپاکها حلقه می‌زند و در هر اپاک، داده را شافل کرده و به مینی-بچ‌ها تقسیم می‌کند. پیش‌بینی‌ها را محاسبه می‌کند، خطاها را محاسبه می‌کند، گرادیان‌ها را محاسبه می‌کند و وزن‌ها (θ) را با استفاده از گرادیان‌های مینی-بچ به‌روزرسانی می‌کند.

4. پیش‌بینی: تابع `predict` برای پیش‌بینی بر روی داده‌های جدید استفاده می‌شود. ویژگی‌های ورودی را نرمال‌سازی می‌کند، یک عبارت انحراف (bias) اضافه می‌کند، پیش‌بینی‌ها را با استفاده از θ یادگرفته شده محاسبه می‌کند و هزینه (خطای میانگین مربعات) برای پیش‌بینی‌ها را محاسبه می‌کند.

5. چاپ: در طول آموزش، کد هر ۱۰ اپاک هزینه را چاپ می‌کند تا پیشرفت آموزش را نظارت کند. پس از آموزش، مقادیر واقعی و پیش‌بینی شده برای مجموعه آزمون چاپ می‌شوند.

۱-ب)

این کد یک پیاده‌سازی ابتدایی از رگرسیون خطی با گرادیان کاهشی مینی-بچ و استفاده از تنظیم (regularization) برای جلوگیری از بیش‌برازش است. دو تغییر اصلی در این پیاده‌سازی اعمال شده است:

1. تابع آموزش با تنظیم: تابع `mini_batch_with_regularization_train` شبیه به `mini_batch_train` است، با این تفاوت که در هر مرحله از آموزش، به گرادیان محاسبه شده یک جمله جریمه (regularization term) اضافه می‌شود. این جمله جریمه به صورت ضریبی از تمامی پارامترهای مدل (به جز پارامتر انحراف) محاسبه می‌شود، تا از افزایش ناخواسته پارامترها و بیش‌برازش جلوگیری شود.

2. هزینه با تنظیم: در هر اپاک، هزینه محاسبه می‌شود با اضافه کردن جمله جریمه به هزینه معمولی بدون تنظیم. این جمله جریمه شامل جمع مربعات تمامی پارامترهای مدل (به جز پارامتر انحراف) است، که به کمک یک ضریب کوچک (lambda_val) تقسیم می‌شود. این کار به جلوگیری از بیش‌برازش کمک می‌کند.

۱-ج)
این تغییرات باعث بهبود عملکرد و عملکرد مدل می‌شوند، به ویژه زمانی که داده‌های آموزش اندازه کمی دارند یا احتمال بیش‌برازش بالا است.

۲-الف)

1. بارگذاری و پیش‌پردازش داده: مجموعه داده سرطان پستان بارگذاری شده و ویژگی‌های ورودی با استفاده از تابع `normalization` نرمال‌سازی شده است.

2. آموزش مدل: تابع `fit` مدل را با استفاده از گرادیان کاهشی مینی-بچ آموزش می‌دهد. این تابع گرادیان تابع هزینه نسبت به پارامترهای تتا را محاسبه کرده و تتا را به‌روزرسانی می‌کند. هزینه هر ۱۰ اپاک چاپ می‌شود تا فرآیند آموزش را نظارت کند.

3. پیش‌بینی: تابع `predict` برچسب‌های مجموعه آزمون را پیش‌بینی می‌کند. این تابع از تتاهای آموزش‌دیده استفاده می‌کند و احتمال‌ها را با استفاده از تابع لجستیک محاسبه می‌کند. اگر احتمال بیشتر از 0.5 باشد، برچسب پیش‌بینی شده 1 است؛ در غیر این صورت 0 است.

4. ارزیابی: تابع `f1_score` امتیاز F1 را محاسبه می‌کند که یک اندازه از دقت مدل استفاده می‌شود. این تابع همزمان بازخوانی و دقت را در نظر می‌گیرد و از آن برای ارزیابی عملکرد مدل در مجموعه آزمون استفاده می‌شود.

به طور کلی، این کد یک پیاده‌سازی ابتدایی از رگرسیون لجستیک برای دسته‌بندی دوتایی فراهم می‌کند و نحوه آموزش مدل، انجام پیش‌بینی و ارزیابی عملکرد آن با استفاده از امتیاز F1 را نشان می‌دهد.

۲-ب)

این کد یک پیاده‌سازی ابتدایی از رگرسیون لجستیک برای دسته‌بندی دوتایی با استفاده از مجموعه داده سرطان پستان است. تغییرات اعمال شده در این پیاده‌سازی عبارتند از:

1. افزودن تابع هزینه با تنظیم: تابع `cost_with_regularization` همانند تابع هزینه اصلی است، با این تفاوت که یک جمله تنظیم به آن اضافه شده است. این جمله تنظیم شامل مجموع مربعات همه پارامترهای تتا (به جز پارامتر انحراف) است، که به تعداد نمونه‌ها (۲ * m) تقسیم می‌شود.

2. آموزش با تنظیم: تابع `fit_with_regularization` مدل را با استفاده از گرادیان کاهشی مینی-بچ و تنظیم آموزش می‌دهد. گرادیان محاسبه شده شامل یک جمله تنظیم است که به گرادیان معمولی اضافه شده است.

3. پیش‌بینی با تنظیم: تابع `predict_with_regularization` پیش‌بینی برچسب‌های مجموعه آزمون را با استفاده از مدل آموزش‌دیده و تنظیم انجام می‌دهد.

۲-ج)

این تغییرات باعث بهبود عملکرد و عملکرد مدل می‌شوند، به ویژه زمانی که داده‌های آموزش اندازه کمی دارند یا احتمال بیش‌برازش بالا است.

-۳

این کد نشان می‌دهد چگونه از رگرسیون خطی و رگرسیون ریج بر روی مجموعه داده مسکن بوستون استفاده کنیم. در زیر توضیحی درباره هر قسمت آمده است:

1. بارگذاری و پیش‌پردازش داده: مجموعه داده مسکن بوستون بارگذاری شده و به ویژگی‌ها (داده) و هدف (target) تقسیم می‌شود. سپس داده‌ها ترکیب می‌شوند و مقیاس‌بندی MinMax بر روی ویژگی‌ها اعمال می‌شود تا نرمال‌سازی صورت گیرد.

2. رگرسیون خطی:
- ویژگی‌های مقیاس‌داده شده به مجموعه‌های آموزش و آزمون تقسیم می‌شوند.
- یک مدل رگرسیون خطی ایجاد و بر روی داده‌های آموزش آموزش داده می‌شود.
- پیش‌بینی‌ها بر روی مجموعه‌های آموزش و آزمون انجام می‌شود.
- خطای میانگین مربعات (MSE) برای پیش‌بینی‌های آموزش و آزمون محاسبه و چاپ می‌شود.

3. رگرسیون ریج:
- مراحل مشابهی به رگرسیون خطی انجام می‌شود، اما از یک مدل رگرسیون ریج استفاده می‌شود.
- رگرسیون ریج از تنظیم با استفاده از پارامتر alpha برای مجازات ضرایب بزرگ استفاده می‌کند.
- MSE برای پیش‌بینی‌های آموزش و آزمون محاسبه و چاپ می‌شود.

خروجی شامل MSE برای مدل‌های رگرسیون خطی و رگرسیون ریج بر روی مجموعه‌های آموزش و آزمون است، که اجازه مقایسه عملکرد آن‌ها را می‌دهد.

-۴

این کد یک مثال از استفاده از رگرسیون لجستیک بر روی مجموعه داده سرطان پستان را نشان می‌دهد. در زیر توضیحی درباره هر قسمت آمده است:

1. بارگذاری و پیش‌پردازش داده: مجموعه داده سرطان پستان بارگذاری شده و به ویژگی‌ها (X) و برچسب‌ها (y) تقسیم می‌شود. سپس داده‌ها توسط مقیاس‌بندی MinMax نرمال‌سازی می‌شوند.

2. تقسیم داده: داده‌ها به مجموعه‌های آموزش و آزمون تقسیم می‌شوند، که 80 درصد از داده‌ها به عنوان داده‌های آموزش و 20 درصد به عنوان داده‌های آزمون استفاده می‌شود.

3. رگرسیون لجستیک بدون تنظیم: یک مدل رگرسیون لجستیک بدون تنظیم ایجاد و بر روی داده‌های آموزش آموزش داده می‌شود. سپس پیش‌بینی‌ها بر روی داده‌های آزمون انجام شده و نتایج چاپ می‌شود.

4. رگرسیون لجستیک با تنظیم L2: مراحل مشابهی به رگرسیون لجستیک بدون تنظیم انجام می‌شود، اما این‌بار از تنظیم L2 استفاده می‌شود. تنظیم L2 از یک پنالتی برای ضرایب بزرگ استفاده می‌کند، که می‌تواند به جلوگیری از بیش‌برازش کمک کند.

5. نمره‌دهی مدل: نمره‌های آموزش و آزمون برای هر دو مدل محاسبه و چاپ می‌شود. این نمره‌ها نشان‌دهنده دقت مدل در پیش‌بینی برچسب‌ها است.

این کد از مفاهیم پایه رگرسیون لجستیک و تاثیر تنظیم در کیفیت مدل استفاده می‌کند و نشان می‌دهد چگونه می‌توان از این روش‌ها برای حل یک مسئله دسته‌بندی استفاده کرد.

-۵

این کد یک تابع برای محاسبه پارامترهای یک مدل رگرسیون خطی با استفاده از معادله نرمال با تنظیم ارائه می‌دهد. در زیر توضیحی درباره هر بخش کد آمده است:

1. تابع normal_equation_with_regularization: این تابع ماتریس ویژگی‌ها `X`، بردار هدف `y` و پارامتر تنظیم `lambda_reg` را می‌گیرد. این تابع پارامترهای `theta` مدل رگرسیون خطی را با استفاده از معادله نرمال با تنظیم محاسبه می‌کند.

2. تولید داده: یک مجموعه داده مصنوعی `X` با ۱۰۰ نمونه و ۳ ویژگی تولید می‌شود. بردار هدف `y` به عنوان یک ترکیب خطی از ویژگی‌ها با اضافه کردن نویز محاسبه می‌شود.

3. پارامتر تنظیم: پارامتر تنظیم `lambda_reg` به ۰.۳ تنظیم شده است.

4. محاسبه پارامترها: تابع `normal_equation_with_regularization` با استفاده از ماتریس ویژگی‌های تکمیل شده `X_b` (که شامل یک ستون اضافی با مقدار یک برای عبارت انحراف)، بردار هدف `y` و `lambda_reg` فراخوانی می‌شود. سپس پارامترهای محاسبه شده `theta` چاپ می‌شود.