Analysis of Gradient Descent on the Triplet Margin Loss
EECS227C Project

Lucas Jaffe
PhD Student in Berkeley EECS Department
Document date: May 9th, 2020

1 Introduction

The goal of this project is to study gradient descent on the triplet margin loss function, a non-convex objective function used in the metric learning domain. The GitHub repository for the project is linked here[1].

We show that the triplet margin loss function can be reformulated as a difference of convex functions, and that a reasonable smooth approximation can be formulated with the LogSumExp function. We show that this function is smooth, and give an upper bound for the smoothness which is dependent on the iterate. Then, we show that a descent condition holds for the gradient descent algorithm, given the right choice of stepsize. Finally, we attempt a gradient descent convergence analysis for the iterate, the squared gradient norm, and the function value, and discuss insights from these efforts.

1.1 Background

Given some data $X \in \mathbb{R}^{n \times m}$ and corresponding labels $Y \in \mathbb{R}^n$, in the supervised metric learning problem, the goal is to find a mapping $\phi : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^{n \times d}$, such that samples with the same label are close together in some metric space, and samples with different labels are further apart. Classically, this type of problem formulation was studied as a dimensionality reduction method, as in Hadsell et al. [2006], for k-nearest neighbors classification, as in Weinberger and Saul [2009], or for the retrieval problem, as in Chechik et al. [2009]. More recently, metric learning has been combined with deep learning into a highly accurate vehicle for solving a range problems in computer vision and natural language processing.

1.2 Problem setup

The *triplet margin loss* function is a common metric learning objective function, which has been demonstrated as empirically effective in the deep learning literature, esp. for facial recognition Schroff et al. [2015]. This is the primary objective function used in the popular OpenFace library (Amos et al. [2016]). Further, this objective has longevity in the literature, having been used in Chechik et al. [2009], defended in Hermans et al. [2017], and recently generalized in Qian et al. [2019].

A triplet is defined as a set of three samples, two of which share the same label, and one of which has a different label. Given some data $X \in \mathbb{R}^{n \times m}$ and corresponding labels $Y \in \mathbb{R}^n$, we define $X' \in \mathbb{R}^{n' \times m}$ as the set of all possible triplets from X, Y. For a given triplet x_i, we denote the two samples sharing a label as x_i^a, x_i^p (anchor and positive), and the third sample x_i^n (negative). In addition, we define a margin parameter $\alpha \in \mathbb{R}^+$. Then the triplet margin loss can be stated as follows:

[1] `https://github.com/LukeJaffe/eecs227c_project`

$$\frac{1}{n'}\sum_{i=1}^{n'}\max(0, \|\phi(x_i^a)-\phi(x_i^p)\|_2^2 - \|\phi(x_i^a)-\phi(x_i^n)\|_2^2+\alpha) \tag{1}$$

We note that the triplet margin formulation described here closely follows from Schroff et al. [2015].

Observing ϕ as a function of some parameter set θ, parameterized by the triplet set X', we can use our objective for empirical risk minimization by solving the following optimization problem:

$$\min_\theta \frac{1}{n'}\sum_{i=1}^{n'}\max(0, \|\phi(\theta;x_i^a)-\phi(\theta;x_i^p)\|_2^2 - \|\phi(\theta;x_i^a)-\phi(\theta;x_i^n)\|_2^2+\alpha) \tag{2}$$

Conceptually, the goal of this problem is to learn θ such that samples with the same label are closer together in Euclidean space than samples with different labels. Observing each class cluster as a hypersphere in $\mathbb{R}^d$, achieving a value of 0 for this objective corresponds to pushing each class hypersphere apart by the max squared diameter of any class hypersphere plus the margin parameter α. In practice, this objective has been found to have nice generalization properties for out-of-sample data.

While this objective has seen widespread usage for learning image and word embeddings, there has been no corresponding optimization convergence analysis to date, even for a simple case. The goal of this work is to present an analysis of gradient descent for a simplified smooth version of (2), with ϕ: $\phi(w;x) = \langle w, x\rangle$, $w \in \mathbb{R}^m$, with $n = 1, d = 1$. The reasoning for this choice will be presented in the following sections.

1.3 DC reformulation

It is clear that (2) is not convex or differentiable as posed, but we will show that for affine choice of ϕ, the objective can be reformulated as a difference of convex functions (DC function). Let $A \in \mathbb{R}^{d\times m}$, and let $\phi(A;x) = Ax$. Then we have

$$\begin{aligned} f(A;X') &= \frac{1}{n'}\sum_{i=1}^{n'}\max(0, \|Ax_i^a - Ax_i^p\|_2^2 - \|Ax_i^a - Ax_i^n\|_2^2 + \alpha) \\ &= \frac{1}{n'}\sum_{i=1}^{n'}\max(0, (x_i^a - x_i^p)A^\top A(x_i^a - x_i^p) - (x_i^a - x_i^n)A^\top A(x_i^a - x_i^n) + \alpha) \end{aligned}$$

Let $u_i = x_i^a - x_i^p$ and $v_i = x_i^a - x_i^n$. Using the identity $\max(0, a-b) = \max(a,b) - b$, we have

$$\begin{aligned} f(A;X') &= \frac{1}{n'}\sum_{i=1}^{n'}\max(u_iA^\top Au_i + \alpha,\ v_iA^\top Av_i) - v_iA^\top Av_i \\ &= \frac{1}{n'}\sum_{i=1}^{n'}\max(u_iA^\top Au_i + \alpha,\ v_iA^\top Av_i) - \frac{1}{n'}\sum_{i=1}^{n'}v_iA^\top Av_i \end{aligned}$$

Since $A^\top A \succeq 0$, $\ u_iA^\top Au_i \geq 0$ is convex. In addition, we know that a max of convex functions is convex, and a sum of convex functions is convex. Thus, we have a difference of convex functions

$$f(A;X') = g(A;X') - h(A;X') \tag{3}$$

with

$$g(A;X') = \frac{1}{n'}\sum_{i=1}^{n'}\max(u_iA^\top Au_i + \alpha,\ v_iA^\top Av_i) \tag{4}$$

$$h(A; X') = \frac{1}{n'} \sum_{i=1}^{n'} v_i A^\top A v_i \tag{5}$$

We note that there is a large body of literature focused on optimizing functions of the DC form, including An and Tao [2005], Lipp and Boyd [2016], and Khamaru and Wainwright [2018].

To simplify the analysis, we focus on the case where $n = 1, d = 1$, and replace the matrix A with the vector $w \in \mathbb{R}^m$. Using this substitution, we have

$$f(w; X') = \max((w^\top u)^2 + \alpha,\ (w^\top v)^2) - (w^\top v)^2 \tag{6}$$

$$g(w; X') = \max((w^\top u)^2 + \alpha,\ (w^\top v)^2) \tag{7}$$

$$h(w; X') = (w^\top v)^2 \tag{8}$$

1.4 Smooth approximation

In addition, the objective can be smoothed using the LogSumExp function with parameter $\mu > 0$, which approximates the max function. Defining the LogSumExp function LSE(x), $x \in \mathbb{R}^k$ as

$$\text{LSE}(x; \mu) = \mu \log \sum_{j=1}^{k} \exp(\frac{1}{\mu} x_j) \tag{9}$$

This function approximates the max function arbitrarily closely as μ approaches 0

$$\lim_{\mu \to 0^+} \text{LSE}(x; \mu) = \max_j x_j \tag{10}$$

Using this approximation, we can rewrite the first term of our DC reformulation as

$$g(w; X') = \mu \log(\exp(\frac{1}{\mu}((w^\top u)^2 + \alpha) + \exp(\frac{1}{\mu}(w^\top v)^2)) \tag{11}$$

For clarity, we write out the full function in this form as well

$$f(w; X') = \mu \log(\exp(\frac{1}{\mu}((w^\top u)^2 + \alpha) + \exp(\frac{1}{\mu}(w^\top v)^2)) - (w^\top v)^2 \tag{12}$$

2 Results

For our initial analysis, we observe the case where $\mu = 1$ and $\alpha = 0$.

$$f(w; X') = \log(\exp((w^\top u)^2) + \exp((w^\top v)^2)) - (w^\top v)^2 \tag{13}$$

$$g(w; X') = \log(\exp((w^\top u)^2) + \exp((w^\top v)^2)) \tag{14}$$

2.1 Notation

In this section, we clarify notation which will be utilized in following sections.

We use the notation $f, f(w), f(w; X'), f(w; u, v)$ interchangeably, with $u = u_i = x_i^a - x_i^p$ and $v = v_i = x_i^a - x_i^n$. The same goes for functions g and h.

We use the substitution $B = uu^\top - vv^\top$ to simplify algebra.

The notation $xx^\top$ is used frequently, referring to the dyadic matrix which is an outer product of some vector $x \in \mathbb{R}^m$ with itself. This matrix is rank one and positive semi-definite, with its only nonzero eigenvalue $\lambda_{\max}(xx^\top) = x^\top x = \|x\|_2^2$.

When we say a function is M-smooth, we mean that it has M-Lipschitz gradients.

2.2 Smoothness of composite functions

To help interpret later results, we analyze the smoothness of the quadratic function $w^\top Bw$ and the LogSumExp function.

We have that

$$\nabla_w^2 w^\top Bw = B \tag{15}$$

Meaning that $w^\top Bw$ is M-smooth, with $M = \lambda_{\max}(B)$.

Next, we analyze the LogSumExp function. Using the substitution $\tilde{w} = [\frac{1}{\mu}\exp(w_1), ..., \frac{1}{\mu}\exp(w_m)]$, we can write the softmax function as

$$s(w) = \frac{\tilde{w}}{\sum_{j=1}^m \tilde{w}_j} \tag{16}$$

Then we have that the Hessian of the LogSumExp function is

$$\nabla^2 \text{LSE}(w; \mu) = \frac{1}{\mu}(\text{diag}(s(w)) - s(w)s(w)^\top) \tag{17}$$

and we have

$$0 \preceq \nabla^2 \text{LSE}(w; \mu) \preceq \frac{1}{\mu} I_m \tag{18}$$

meaning that LSE$(w; \mu)$ is convex and $\frac{1}{\mu}$-smooth. A proof of these facts can be found in Gao and Pavel [2018].

2.3 Analysis of function minima

The function $f(w; u, v)$ has two possible minima, which are dependent on the data u, v. Let $B = uu^\top - vv^\top$. We can rewrite the function in the softplus form by pushing all terms from the max into an exponential. Then we have

$$f(w; B) = \log(\exp(w^\top Bw) + 1) \tag{19}$$

Since $w^\top Bw$ has the form of a standard quadratic in w, we have

$$\min_w w^\top Bw = \begin{cases} 0 & \text{if } B \succeq 0 \\ -\infty & \text{otherwise} \end{cases} \tag{20}$$

Extending this to $f(w; B)$, we have

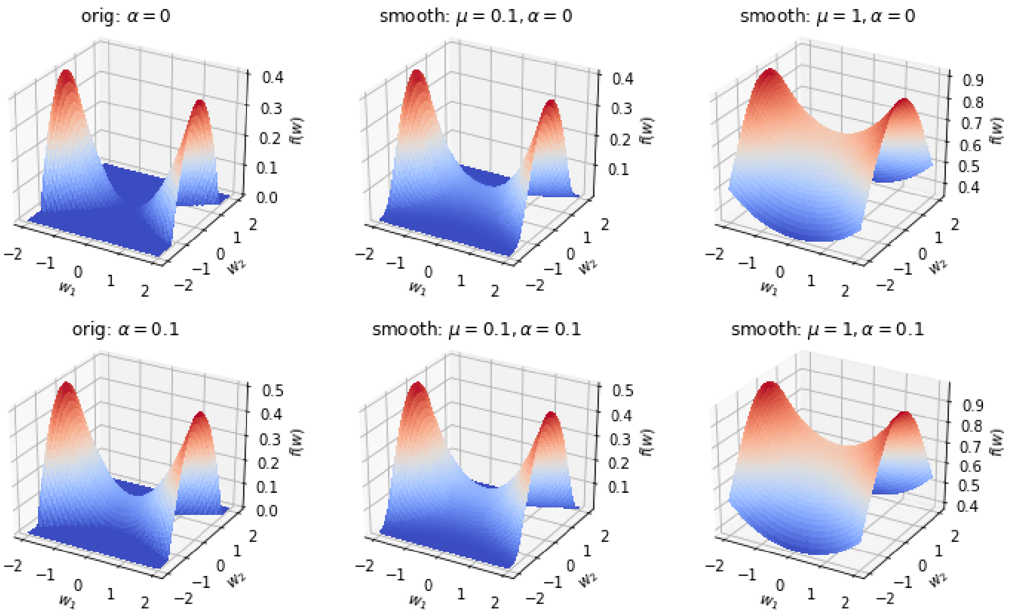

Figure 1: This figure shows the triplet margin loss surface for a randomly generated 2-d triplet sample, with $\alpha = 0$ in the first row and $\alpha = 0.1$ in the second row. The leftmost plots show the original objective with no smoothing. The center plots show the smooth objective with $\mu = 0.1$. The rightmost plots show the smooth objective with $\mu = 1$.

$$\inf_w f(w; B) = \begin{cases} \log(2) & \text{if } B \succeq 0 \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

We note that $B \succeq 0$ will occur if u, v are linearly dependent and B is positive.

In Figure 1, we show the triplet margin loss surface for a randomly generated 2-d triplet sample, as posed in (6) and (12), with $\alpha = 0$ and $\alpha = 0.1$. We see that the loss surface has a classical saddle shape, intersecting with a plane for the original formulation. We note that the randomly generated sample gives $\inf_w f = 0$ for both cases.

2.4 Smoothness analysis of $g(w)$

We start by analyzing the formulation $f(w) = g(w) - h(w)$ using the methodology from Khamaru and Wainwright [2018]. To use the theorem proven in that work, we need g to be continuously differentiable and M_g-smooth, and we need h to be continuous and convex. These properties are satisfied for the smooth reformulation from (13) and (14). Here, we will derive the value of M_g.

$$\nabla g(w) = \frac{2\{\exp((w^\top u)^2) uu^\top + \exp((w^\top v)^2) vv^\top\} w}{\exp((w^\top u)^2) + \exp((w^\top v)^2)} \tag{22}$$

To find the Hessian, we break the gradient into two parts for use of the chain rule:

$$\begin{aligned} \nabla(\nabla g(w) \text{ top}) &= 4\{\exp((w^\top u)^2) uu^\top ww^\top uu^\top + \exp((w^\top v)^2) vv^\top ww^\top vv^\top\} \\ &= 4\{\exp((w^\top u)^2)(w^\top u)^2 uu^\top + \exp((w^\top v)^2)(w^\top v)^2 vv^\top\} \end{aligned}$$

$$\nabla(\nabla g(w) \text{ bot}) = \frac{-2\{\exp((w^\top u)^2) uu^\top + \exp((w^\top v)^2) vv^\top\} w}{\{\exp((w^\top u)^2) + \exp((w^\top v)^2)\}^2}$$

Combining the parts:

$$\nabla^2 g(w) = \frac{4\{\exp((w^\top u)^2)(w^\top u)^2 uu^\top + \exp((w^\top v)^2)(w^\top v)^2 vv^\top\}}{\exp((w^\top u)^2) + \exp((w^\top v)^2)}$$
$$- \frac{4\{\exp((w^\top u)^2)uu^\top + \exp((w^\top v)^2)vv^\top\}ww^\top\{\exp((w^\top u)^2)uu^\top + \exp((w^\top v)^2)vv^\top\}}{\{\exp((w^\top u)^2) + \exp((w^\top v)^2)\}^2}$$

Re-arranging, we have:

$$\begin{aligned}\nabla^2 g(w) = 4&\left\{\frac{\exp((w^\top u)^2)(w^\top u)^2 uu^\top}{\exp((w^\top u)^2) + \exp((w^\top v)^2)} + \frac{\exp((w^\top v)^2)(w^\top v)^2 vv^\top}{\exp((w^\top u)^2) + \exp((w^\top v)^2)}\right\} \\ - 4&\left\{\frac{\exp((w^\top u)^2)uu^\top}{\exp((w^\top u)^2) + \exp((w^\top v)^2)} + \frac{\exp((w^\top v)^2)vv^\top}{\exp((w^\top u)^2) + \exp((w^\top v)^2)}\right\} ww^\top \\ &\left\{\frac{\exp((w^\top u)^2)uu^\top}{\exp((w^\top u)^2) + \exp((w^\top v)^2)} + \frac{\exp((w^\top v)^2)vv^\top}{\exp((w^\top u)^2) + \exp((w^\top v)^2)}\right\}\end{aligned}$$

We can simplify further by using the substitution:

$$p = \frac{\exp((w^\top u)^2)}{\exp((w^\top u)^2) + \exp((w^\top v)^2)}$$

Note that $0 < p < 1$, and:

$$1 - p = \frac{\exp((w^\top v)^2)}{\exp((w^\top u)^2) + \exp((w^\top v)^2)}$$

Plugging in to the Hessian expression:

$$\begin{aligned}\nabla^2 g(w) &= 4\{p(w^\top u)^2 uu^\top + (1-p)(w^\top v)^2 vv^\top\} \\ &\quad - 4\{puu^\top + (1-p)vv^\top\}ww^\top\{puu^\top + (1-p)vv^\top\} \\ &= 4p(w^\top u)^2 uu^\top + 4(1-p)(w^\top v)^2 vv^\top \\ &\quad - 4p^2(w^\top u)^2 uu^\top - 4(1-p)^2(w^\top v)^2 vv^\top \\ &\quad - 8p(1-p)(w^\top u)(w^\top v)(uv^\top + vu^\top) \\ &= 4p(1-p)(w^\top u)^2 uu^\top + 4p(1-p)(w^\top v)^2 vv^\top \\ &\quad - 8p(1-p)(w^\top u)(w^\top v)(uv^\top + vu^\top)\end{aligned}$$

Using $0 < p(1-p) \leq \frac{1}{4}$, we have, $\forall z \in \mathbb{R}^m$:

$$\begin{aligned}z^\top \nabla^2 g(w) z &\leq z^\top \left\{(w^\top u)^2 uu^\top + (w^\top v)^2 vv^\top - 2(w^\top u)(w^\top v)(uv^\top + vu^\top)\right\} z \\ &\leq \|z\|_2^2 \left\{(w^\top u)^2\|u\|_2^2 + (w^\top v)^2\|v\|_2^2 + 4\left|(w^\top u)(w^\top v)(u^\top v)\right|\right\} \\ &\leq \|z\|_2^2\|w\|_2^2 \left\{\|u\|_2^4 + \|v\|_2^4 + 4\|u\|_2^2\|v\|_2^2\right\}\end{aligned}$$

This gives us:

$$\nabla^2 g(w) \preceq \|w\|_2^2 \left\{\|u\|_2^4 + \|v\|_2^4 + 4\|u\|_2^2\|v\|_2^2\right\} I_m \tag{23}$$

Meaning that $g(w)$ is M_g-smooth, with an upper bound being $M_g = \|w\|_2^2 \left\{\|u\|_2^4 + \|v\|_2^4 + 4\|u\|_2^2\|v\|_2^2\right\}$.

2.5 Smoothness analysis of $f(w)$

In this section, we analyze the full function, instead of just $g(w)$, to see how the smoothness bound compares. Recall that $B = uu^\top - vv^\top$. We analyze the function in the softplus form

$$f(w) = \log(\exp(w^\top Bw) + 1) \tag{24}$$

$$\nabla f(w) = \frac{2\exp(w^\top Bw)Bw}{\exp(w^\top Bw) + 1} \tag{25}$$

We can simplify the notation using the substitution

$$q = \frac{\exp(w^\top Bw)}{\exp(w^\top Bw) + 1} \tag{26}$$

Applying this substitution, we have

$$\nabla f(w) = 2qBw \tag{27}$$

To find the Hessian, we break the gradient into two parts for use of the chain rule:

$$\nabla(\nabla f(w) \text{ top}) = 4\exp(w^\top Bw)Bww^\top B + 2\exp(w^\top Bw)B$$

$$\nabla(\nabla f(w) \text{ bot}) = \frac{-2\exp(w^\top Bw)B}{\exp(w^\top Bw) + 1}$$

Combining the parts:

$$\nabla^2 f(w) = \frac{4\exp(w^\top Bw)B(ww^\top B + \frac{1}{2}I_m)}{\exp(w^\top Bw) + 1} - \frac{4\{\exp(w^\top Bw)\}^2 Bww^\top B}{\{\exp(w^\top Bw) + 1\}^2}$$

Using this substitution of q, we have:

$$\nabla^2 f(w) = 4q(1-q)Bww^\top B + 2qB \tag{28}$$

Noting that $0 < q < 1$ and $0 < q^2 \leq \frac{1}{4}$ we have, $\forall z \in \mathbb{R}^m$:

$$\begin{aligned} z^\top \nabla^2 f(w) z &= 4q(1-q)z^\top Bww^\top Bz + 2qz^\top Bz \\ &< z^\top Bww^\top Bz + 2z^\top Bz \\ &\leq \|z\|_2^2 \|w\|_2^2 \lambda_{\max}(B)^2 + 2\|z\|_2^2 \lambda_{\max}(B) \\ &= \|z\|_2^2 (\|w\|_2^2 \lambda_{\max}(B)^2 + 2\lambda_{\max}(B)) \end{aligned}$$

This gives us:

$$\nabla^2 f(w) \prec (\|w\|_2^2 \lambda_{\max}(B)^2 + 2\lambda_{\max}(B)) I_m \tag{29}$$

Meaning that $f(w)$ is M_f-smooth, with an upper bound being $M_f = \|w\|_2^2 \lambda_{\max}(B)^2 + 2\lambda_{\max}(B)$. We note that this smoothness bound is usually better than the one achieved in the previous section from analyzing $g(w)$ alone.

We also note that since we have shown the function is twice continuously differentiable and M_f-smooth, the set of initial points for which gradient descent converges to a strict saddle point has measure zero, as proven in Lee et al. [2016].

2.6 Gradient descent analysis of $f(w)$

We analyze the standard gradient step

$$w^{t+1} = w^t - \eta \nabla f(w^t) \tag{30}$$

We would like to show that the following descent condition holds for some $\eta > 0, c < 0$

$$f(w^{t+1}) \leq f(w^t) + c \tag{31}$$

This property is not guaranteed in general for a difference of convex functions, although it is satisfied by the condition of smoothness. We will derive a different descent condition here.

$$f(w^{t+1}) = f(w^t - \eta \nabla f(w^t)) \tag{32}$$

Using $\nabla f(w)$ from Eq. (27) , we have

$$\begin{aligned} w - \eta \nabla f(w) &= w - 2qBw \\ &= (I_m - 2qB)w \end{aligned}$$

Applying this term to f from (24), we get

$$\begin{aligned} f(w^t - \eta \nabla f(w^t)) &= \log(\exp(w^\top (I_m - 2\eta qB)B(I_d - 2\eta qB)w) + 1) \\ &= \log(\exp(w^\top Bw + w^\top (4\eta^2 q^2 B^3 - 4\eta qB^2)w) + 1) \end{aligned} \tag{33}$$

Let $b_\eta = w^\top (4\eta^2 q^2 B^3 - 4\eta qB^2)w$. Making this substitution, we have

$$\begin{aligned} f(w^t - \eta \nabla f(w^t)) &= \log(\exp(w^\top Bw + b_\eta) + 1) \\ &= \log(\exp(w^\top Bw)\exp(b_\eta) + 1) \end{aligned} \tag{34}$$

Examining the desired descent condition from (31), we want to find some $c < 0$ such that

$$f(w^{t+1}) - f(w^t) \leq c$$

Substituting in, we find that

$$\begin{aligned} f(w^{t+1}) - f(w^t) &= \log(\exp(w^\top Bw)\exp(b_\eta) + 1) - \log(\exp(w^\top Bw) + 1) \\ &= \log\left\{\frac{\exp(w^\top Bw)\exp(b_\eta) + 1}{\exp(w^\top Bw) + 1}\right\} \\ &= \log(q\exp(b_\eta) + (1-q)) \end{aligned}$$

Therefore, we must select $c < \log(q\exp(b_\eta) + (1-q))$ to satisfy the descent condition. In particular, this form shows that the descent condition holds for $b_\eta < 0$.

2.7 Selecting the step size η

Since we have that $f(w^{t+1}) < f(w^t) + c,\ c < 0$ we would like to select the step size η such that c is minimized at each step. Using the direct line search approach, we attempt to solve the following optimization problem:

$$c^* = \min_{\eta} \log(q \exp(b_\eta) + (1-q)) \tag{35}$$

In terms of η, we want to find

$$\begin{aligned} \eta^* &= \arg\min_{\eta} \log(q \exp(b_\eta) + (1-q)) \\ &= \arg\min_{\eta} b_\eta \\ &= \arg\min_{\eta} w^\top (4\eta^2 q^2 B^3 - 4\eta q B^2) w \end{aligned} \tag{36}$$

Although the function is not convex in η, we will attempt to choose η from a first order critical point.

$$\begin{aligned} \nabla_\eta w^\top (4\eta^2 q^2 B^3 - 4\eta q B^2) w = 8\eta q^2 w^\top B^3 w - 4q w^\top B^2 w = 0 \\ \implies 2\eta q w^\top B^3 w = w^\top B^2 w \\ \implies \eta = \frac{w^\top B^2 w}{2q w^\top B^3 w} \\ \implies \eta \leq \frac{1}{2q\lambda_{\max}(B)} \\ \implies \hat{\eta} = \frac{1}{2q\lambda_{\max}(B)} \end{aligned}$$

Although we can't show $\hat{\eta}$ minimizes b_η, we can show it is a good choice by examining the eigenvalues of $b_{\hat{\eta}}$. Let $B = U^\top \Lambda U$ be the spectral decomposition of B. Then we have

$$\begin{aligned} b_{\hat{\eta}} &= w^\top (4\hat{\eta}^2 q^2 B^3 - 4\hat{\eta} q B^2) w \\ &= w^\top \left(\frac{B^3}{\lambda_{\max}(B)^2} - \frac{2B^2}{\lambda_{\max}(B)} \right) w \\ &= w^\top U^\top \left(\frac{\Lambda^3}{\lambda_{\max}(B)^2} - \frac{2\Lambda^2}{\lambda_{\max}(B)} \right) U w \end{aligned}$$

In this form, we can see that all the eigenvalues in the inner term must be ≤ 0. Although this choice of η guarantees descent, there is not an easy way to isolate terms so we can use the condition for convergence analysis. The strongest claim we can make is that $b_{\hat{\eta}} \leq 0$.

Comparing to the smoothness of $w^\top B w = \lambda_{\max}(B) = M$ from Section 2.2, $\frac{1}{M}$ is very close to the stepsize $\hat{\eta}$, suggesting that the function behavior is dominated by the underlying quadratic.

In Appendix A, we compare this choice of stepsize to ones derived from smoothness in the previous sections, for a simple gradient descent experiment.

2.8 Coarse bound on the iterate

Using the stepsize $\eta = \hat{\eta}$

$$\begin{aligned} w^{t+1} = w^t - \hat{\eta} \nabla f(w^t) &= (I_m - 2q\hat{\eta} B) w^t \\ &= \left(I_m - \frac{B}{\lambda_{\max}(B)} \right) w^t \\ &= \left(I_m - \frac{B}{\lambda_{\max}(B)} \right)^t w^1 \end{aligned}$$

Taking the norm of both sides, we have

$$\begin{aligned}\|w^{t+1}\|_2 &= \left\| \left(I_m - \frac{B}{\lambda_{\max}(B)}\right)^t w^1 \right\|_2 \\ &\leq \left\| \left(I_m - \frac{B}{\lambda_{\max}(B)}\right)^t \right\|_F \|w^1\|_2\end{aligned}$$

Since there is no guarantee on what the norm of some w^* should be, this bound may not be very useful in practice.

2.9 Convergence of the squared gradient norm

Choosing a stepsize of $\eta = \frac{1}{M_f}$, we have the standard bound

$$\sqrt{\frac{1}{T}\sum_{t=1}^{T} \|\nabla f(w^t)\|_2^2} \leq \sqrt{\frac{2M_f(f(w^1) - f(w^*))}{T}} \tag{37}$$

However, since both of our derived smoothness terms depend on the iterate, we cannot assume this bound. So we will instead utilize our descent condition from Section 2.6.

We have

$$\|\nabla f(w)\|_2^2 = 4q^2 w^\top B^2 w \tag{38}$$

From Section 2.6, choosing $\eta = \frac{1}{\lambda_{\max}(B)}$, we have

$$\begin{aligned}b_\eta &= w^\top(4\eta^2 q^2 B^3 - 4\eta q B^2)w \\ &= \frac{\nabla f(w)^\top B \nabla f(w)^\top}{\lambda_{\max}(B)^2} - \frac{\|\nabla f(w)\|_2^2}{q\lambda_{\max}(B)} \\ &\leq \frac{\lambda_{\max}(B)\|\nabla f(w)\|_2^2}{\lambda_{\max}(B)^2} - \frac{\|\nabla f(w)\|_2^2}{q\lambda_{\max}(B)} \\ &= \frac{(q-1)\|\nabla f(w)\|_2^2}{q\lambda_{\max}(B)}\end{aligned}$$

This gives us

$$\begin{aligned}f(w^{t+1}) - f(w^t) &\leq \log(q\exp(b_\eta) + (1-q)) \\ &= \log\left(q\exp\left(\frac{(q-1)\|\nabla f(w^t)\|_2^2}{q\lambda_{\max}(B)}\right) + (1-q)\right) \\ &\implies \log\left(q\exp\left(\frac{(1-q)\|\nabla f(w^t)\|_2^2}{q\lambda_{\max}(B)}\right) + (1-q)\right) \leq f(w^t) - f(w^{t+1})\end{aligned}$$

Finally, we have

$$\begin{aligned}\sum_{t=1}^{T} \log\left(q\exp\left(\frac{(1-q)\|\nabla f(w^t)\|_2^2}{q\lambda_{\max}(B)}\right) + (1-q)\right) &\leq f(w^1) - f(w^T) \\ &\leq f(w^1) - f(w^*)\end{aligned}$$

The term inside the log is always positive, and $q\exp(.)+(1-q) = 1$ if and only if $\exp(.) = 0$, which can occur only if $\|\nabla f(w^t)\|_2^2 = 0$, since $0 < q < 1$. Therefore, we can conclude that $\|\nabla f(w^t)\|_2^2 \to 0$. Unfortunately, it is difficult to compare this bound against the standard smoothness bound, since the norm gradient term can't be trivially freed from the logs and exponentials.

Another approach to this analysis could be to use the smoothness term directly for the standard smoothness descent condition

$$f(w^{t+1}) \leq f(w^t) - \frac{1}{2M_f}\|\nabla f(w^t)\|_2^2 \tag{39}$$

We could attempt to free the iterate term from the smoothness. An easy way to do that would be to restrict $\|w\|_2 \leq 1$.

2.10 Convergence of the function value

Using the gradient update, we have

$$\begin{aligned}
w^{t+1\top}Bw^{t+1} &\leq w^{t\top}Bw^t + b_\eta \\
&\leq w^{1\top}Bw^1 + \sum_{k=1}^{t+1} b_\eta^k \\
&\implies \log(\exp(w^{t+1\top}Bw^{t+1}) + 1) \leq \log(\exp(w^{1\top}Bw^1 + \sum_{k=1}^{t+1} b_\eta^k) + 1) \\
&\implies f(w^{t+1}) \leq \log(\exp(w^{1\top}Bw^1 + \sum_{k=1}^{t+1} b_\eta^k) + 1)
\end{aligned}$$

Now we can say

$$\begin{aligned}
f(w^{t+1}) - f^* &\leq \log(\exp(w^{1\top}Bw^1 + \sum_{k=1}^{t+1} b_\eta^k) + 1) - f^* \\
&= \log(\exp(w^{1\top}Bw^1 + \sum_{k=1}^{t+1} b_\eta^k) + 1) - \log(\exp(w^{*\top}Bw^*) + 1) \\
&= \log\left(\frac{\exp(w^{1\top}Bw^1 + \sum_{k=1}^{t+1} b_\eta^k) + 1}{\exp(w^{*\top}Bw^*) + 1}\right) \\
&\leq \log\left(\frac{\exp(w^{1\top}Bw^1 + \sum_{k=1}^{t+1} b_\eta^k)}{\exp(w^{*\top}Bw^*)} + 1\right) \\
&= \log\left(\exp\left(w^{1\top}Bw^1 - w^{*\top}Bw^* + \sum_{k=1}^{t+1} b_\eta^k\right) + 1\right)
\end{aligned}$$

Simplifying further appears to be nontrivial, and we can't make a claim about the iteration complexity without somehow freeing the iteration count from the summation term.

3 Conclusion

3.1 Discussion

In this work, we show that the triplet margin loss function can be reformulated as a difference of convex functions, and that a reasonable smooth approximation can be formulated with the LogSumExp function. We show that this function is smooth, and give an upper bound for the smoothness which is dependent on the iterate. Then, we show that a descent condition not directly related to the smoothness holds for the gradient descent algorithm, given the right choice of stepsize. Finally, we attempt a gradient descent convergence analysis for the iterate, the squared gradient norm, and the function value. While none of the convergence bounds achieved are very useful, they provide insight into the structure of the function, and lay a nice foundation for future work.

One conclusion from this effort is that the condition of convexity cannot be trivially replaced just by knowing the function structure. Still, by showing that the triplet function is smooth, and that it satisfies a descent condition separate from its smoothness, we have shown that the function is more constrained than the conditions imposed from Khamaru and Wainwright [2018], despite being of the DC form. Further analysis should yield a faster rate than the bound on the norm squared gradient achieved from smoothness.

3.2 Future work

There are several extensions that we would like to perform on this analysis. First, we would like to repeat the analysis for $\alpha > 0$ and $\mu \neq 1$. Second we would like to analyze the multi-sample case ($n > 1$). Third we would like to study the case where $d > 1$. Fourth, we would like to study the formulation with a bias term, i.e. $\phi(w) = \langle w, x \rangle + b$. Finally, we would like to study the stochastic gradient method, so that this work can be applied in practice for empirical risk minimization, given randomly sampled data. These insights can be imported back to the deep learning setting to have a more principled understanding of optimization in that context.

In addition, we would like to conduct a similar analysis for the contrastive loss function posed in Hadsell et al. [2006], which optimizes over pairs of samples instead of triplets. This loss function can be reformulated as a triplet loss by considering two pairs, and is also a DC function that can be smoothed with LogSumExp. We would like to compare convergence bounds, and then conduct an empirical analysis, to determine which objective generalizes better to new data.

We would also like to compare the triplet loss to the soft margin loss and logistic loss for the support vector machine (SVM). It would be interesting to compare the learned hyperplane boundary from w, and see how the models compare in terms of generalization.

References

B. Amos, B. Ludwiczuk, and M. Satyanarayanan. OpenFace: A general-purpose face recognition library with mobile applications. 2016.

L. T. H. An and P. D. Tao. The DC (Difference of Convex Functions) Programming and DCA Revisited with DC Models of Real World Nonconvex Optimization Problems. *Annals of Operations Research*, 133(1):23–46, Jan. 2005. ISSN 1572-9338. doi: 10.1007/s10479-004-5022-1. URL `https://doi.org/10.1007/s10479-004-5022-1`.

G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large Scale Online Learning of Image Similarity through Ranking. In H. Araujo, A. M. Mendonça, A. J. Pinho, and M. I. Torres, editors, *Pattern Recognition and Image Analysis*, volume 5524, pages 11–14. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-642-02171-8 978-3-642-02172-5. doi: 10.1007/978-3-642-02172-5_2. URL `http://link.springer.com/10.1007/978-3-642-02172-5_2`. Series Title: Lecture Notes in Computer Science.

B. Gao and L. Pavel. On the Properties of the Softmax Function with Application in Game Theory and Reinforcement Learning. *arXiv:1704.00805 [cs, math]*, Aug. 2018. URL `http://arxiv.org/abs/1704.00805`. arXiv: 1704.00805.

R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality Reduction by Learning an Invariant Mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06)*, volume 2, pages 1735–1742, New York, NY, USA, 2006. IEEE. ISBN 978-0-7695-2597-6. doi: 10.1109/CVPR.2006.100. URL `http://ieeexplore.ieee.org/document/1640964/`.

A. Hermans, L. Beyer, and B. Leibe. In Defense of the Triplet Loss for Person Re-Identification. Mar. 2017. URL `https://arxiv.org/abs/1703.07737v4`.

K. Khamaru and M. Wainwright. Convergence guarantees for a class of non-convex and non-smooth optimization problems. In *International Conference on Machine Learning*, pages 2601–2610, July 2018. URL `http://proceedings.mlr.press/v80/khamaru18a.html`. ISSN: 1938-7228 Section: Machine Learning.

J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht. Gradient Descent Only Converges to Minimizers. In *Conference on Learning Theory*, pages 1246–1257, June 2016. URL `http://proceedings.mlr.press/v49/lee16.html`. ISSN: 1938-7228 Section: Machine Learning.

T. Lipp and S. Boyd. Variations and extension of the convex–concave procedure. *Optimization and Engineering*, 17(2):263–287, June 2016. ISSN 1389-4420, 1573-2924. doi: 10.1007/s11081-015-9294-x. URL `http://link.springer.com/10.1007/s11081-015-9294-x`.

Q. Qian, L. Shang, B. Sun, J. Hu, T. Tacoma, H. Li, and R. Jin. SoftTriple Loss: Deep Metric Learning Without Triplet Sampling. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6449–6457, Seoul, Korea (South), Oct. 2019. IEEE. ISBN 978-1-72814-803-8. doi: 10.1109/ICCV.2019.00655. URL `https://ieeexplore.ieee.org/document/9008816/`.

F. Schroff, D. Kalenichenko, and J. Philbin. FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, June 2015. doi: 10.1109/CVPR.2015.7298682. ISSN: 1063-6919.

K. Q. Weinberger and L. K. Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. *Journal of Machine Learning Research*, 10(Feb):207–244, 2009. ISSN ISSN 1533-7928. URL `http://www.jmlr.org/papers/v10/weinberger09a.html`.

Appendix A Gradient Descent Experiment

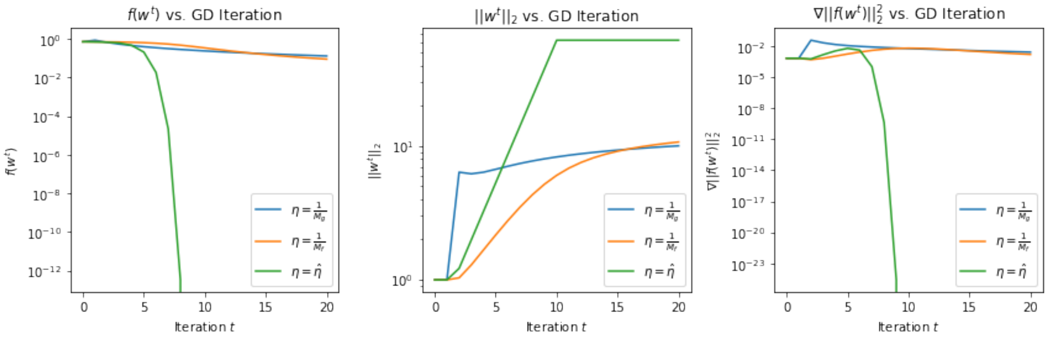

Figure 2: This figure gradient descent on the triplet margin loss for a randomly generated 2-d triplet sample, with $\alpha = 0, \mu = 1$. The three plots correspond to the function value, norm of the iterate, and norm squared of the gradient respectively. Each plot has three stepsizes, chosen based on the bounds for M_g, M_f, and $\hat{\eta}$. Notice that the function value descends monotonically for all choices of η, but descends much faster for $\eta = \hat{\eta}$ (appears to be quadratic in log space). Also, notice that the norm squared of the gradient increases before decreasing, which can be explained by the saddle geometry of the loss surface.