신경망의 손실 환경 시각화하기

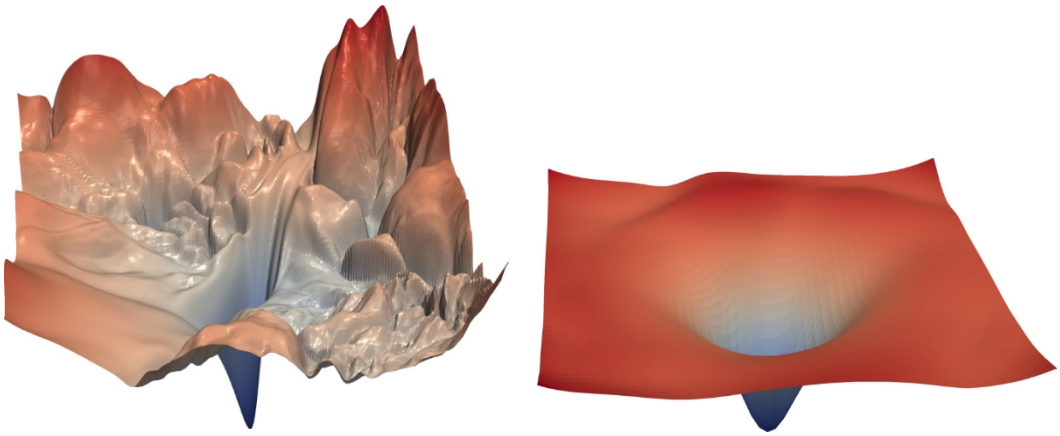
하오 리[1], **정** 쉬[1], **개빈** 테일러[2], **크리스토프** 스터더[3], **톰** 골드스타인[1]
[1]메릴랜드 대학교 칼리지 파크[2] 미국 해군사관학교[3] 코넬 대학교
{haoli,xuzh,tomg}@cs.umd.edu, taylor@usna.edu, studer@cornell.edu

초록

신경망 훈련은 매우 비볼록한 손실 함수의 "좋은" 최소화자를 찾는 능력에 의존합니다. 특정 네트워크 아키텍처 설계(예: 연결 건너뛰기)는 더 쉽게 학습하는 손실 함수를 생성하고, 잘 선택된 학습 매개변수(배치 크기, 학습 속도, 최적화 도구)는 더 잘 일반화하는 최소화 함수를 생성한다는 사실은 잘 알려져 있습니다. 그러나 이러한 차이의 이유와 기본 손실 환경에 미치는 영향은 잘 알려져 있지 않습니다. 이 백서에서는 다양한 시각화 방법을 사용하여 신경 손실 함수의 구조와 일반화에 대한 손실 풍경의 영향을 살펴봅니다. 먼저 손실 함수의 곡률을 시각화하고 손실 함수 간에 의미 있는 나란히 비교하는 데 도움이 되는 간단한 '필터 정규화' 방법을 소개합니다. 그런 다음 다양한 시각화를 사용하여 네트워크 아키텍처가 손실 환경에 어떤 영향을 미치는지, 훈련 매개변수가 최소화기의 모양에 어떤 영향을 미치는지 살펴봅니다.

1 소개

신경망을 훈련하려면 고차원 비볼록 손실 함수를 최소화해야 하는데, 이는 이론적으로는 어렵지만 실제로는 쉬운 작업이기도 합니다. 일반적인 신경 손실 함수를 훈련하는 것은 NP-경도임에도 불구하고[2], 단순 기울기 방법은 훈련 전에 데이터와 레이블이 무작위화된 경우에도 전역 최소자(훈련 손실이 0이거나 거의 0에 가까운 매개변수 구성)를 찾는 경우가 많습니다[42]. 그러나 신경망의 훈련 가능성은 네트워크 아키텍처 설계 선택, 최적화기 선택, 변수 초기화 및 기타 다양한 고려 사항에 따라 크게 달라집니다. 안타깝게도 이러한 각 선택이 기본 손실 표면의 구조에 미치는 영향은 불분명합니다. 손실 함수 평가에 드는 엄청난 비용(훈련 세트의 모든 데이터 포인트를 반복해야 함)으로 인해 이 분야의 연구는 주로 이론적인 수준에 머물러 있습니다.



(a) 스킵 연결이 없는 경우 (b) 스킵 연결이 있는 경우

그림 1: 스킵 연결이 있는 경우와 없는 경우의 ResNet-56 손실 표면. 제안된 필터 정규화 방식은 두 그림 간의 선명도/평탄도를 비교하는 데 사용됩니다.

제32회 신경 정보 처리 시스템 컨퍼런스(NIPS 2018), 캐나다 몬트리올.

시각화는 신경망이 작동하는 이유에 대한 몇 가지 중요한 질문에 답하는 데 도움이 될 수 있습니다. 특히, 매우 비볼록한 신경 손실 함수를 최소화할 수 있는 이유는 무엇일까요? 그리고 그 결과로 나온 최소값이 일반화되는 이유는 무엇일까요? 이러한 질문을 명확히 하기 위해 고해상도 시각화를 사용하여 신경 손실 함수의 경험적 특성을 분석하고, 다양한 네트워크 아키텍처 선택이 손실 환경에 어떤 영향을 미치는지 살펴봅니다. 또한 신경 손실 함수의 비볼록 구조가 훈련 가능성과 어떤 관련이 있는지, 신경 최소화의 기하학적 구조(예: 선명도/평탄도 및 주변 환경)가 일반화 속성에 어떤 영향을 미치는지 살펴봅니다.

이를 의미 있는 방식으로 수행하기 위해, 훈련 중에 발견된 다양한 최소값을 나란히 비교할 수 있는 간단한 "필터 정규화" 방식을 제안합니다. 그런 다음 시각화를 사용하여 다양한 방법으로 찾은 최소값의 선명도/평탄도뿐만 아니라 네트워크 아키텍처 선택(스킵 연결 사용, 필터 수, 네트워크 깊이)이 손실 지형에 미치는 영향을 탐색합니다. 우리의 목표는 손실 함수 기하학이 신경망의 일반화에 어떤 영향을 미치는지 이해하는 것입니다.

1.1 기여

의미 있는 손실 함수 시각화를 생성하는 방법을 연구합니다. 그런 다음 이러한 시각화 방법을 사용하여 손실 풍경 기하학이 일반화 오류와 훈련 가능성에 어떤 영향을 미치는지 살펴봅니다. 보다 구체적으로 다음과 같은 문제를 다룹니다:

- 손실 함수에 대한 여러 시각화 방법의 결함을 밝히고, 단순한 시각화 전략으로는 손실 함수 최소화기의 국부적 형상(선명도 또는 평탄도)을 정확하게 포착하지 못한다는 사실을 보여줍니다.
- "필터 정규화"를 기반으로 하는 간단한 시각화 방법을 소개합니다. 이 정규화를 사용하면 서로 다른 네트워크 아키텍처와 훈련 방법을 비교하더라도 최소화 기법의 선명도는 일반화 오류와 밀접한 상관관계가 있습니다. 이를 통해 서로 다른 최소화 기법을 나란히 비교할 수 있습니다.[1].
- 네트워크가 충분히 깊어지면 신경 손실 환경이 거의 볼록한 형태에서 매우 혼란스러운 형태로 빠르게 전환되는 것을 관찰할 수 있습니다. 이러한 볼록한 행동에서 혼란스러운 행동으로의 전환은 일반화 오류가 급격히 감소하고 궁극적으로 학습 가능성이 부족해지는 것과 일치합니다.
- 스킵 연결이 플랫 미니마이저를 촉진하고 혼란스러운 행동으로의 전환을 방지하는 것으로 관찰되었으며, 이는 스킵 연결이 매우 심층적인 네트워크를 훈련하는 데 필요한 이유를 설명하는 데 도움이 됩니다.
- 국부 최소값을 중심으로 헤시안 고유값의 가장 작은(가장 음수인) 값을 계산하고 그 결과를 히트 맵으로 시각화하여 비볼록성을 정량적으로 측정합니다.
- 우리는 SGD 최적화 궤적의 시각화에 대해 연구합니다. 이러한 궤적을 시각화할 때 발생

하는 어려움을 설명하고 최적화 궤적이 극히 낮은 차원의 공간에 있음을 보여줍니다. 이러한 낮은 차원은 2차원 시각화에서 관찰되는 것과 같이 손실 지형에 거의 볼록한 큰 영역이 존재하기 때문에 설명할 수 있습니다.

2 이론적 배경

신경 손실 함수를 최적화하는 능력에 대한 수많은 이론적 연구가 수행되었습니다 [5, 4]. 이론적 결과는 일반적으로 샘플 분포, 아키텍처의 비선형성 또는 손실 함수에 대해 제한적인 가정을 합니다 [16, 31, 40, 36, 9, 39]. 단일 숨겨진 계층이 있는 네트워크와 같이 제한된 네트워크 클래스의 경우 일반적인 최적화 방법을 통해 전역적으로 최적 또는 최적에 가까운 솔루션을 찾을 수 있습니다[35, 26, 38]. 특정 구조를 가진 네트워크의 경우, 초기화에서 전역 최소값까지 단조롭게 감소하는 경로가 존재할 가능성이 높습니다[32, 15]. Swirszcz et al.
[37]은 장난감 문제에 대해 "나쁜" 국소 최소값을 달성하는 반대 사례를 보여줍니다.

여러 연구에서 국소 최소값의 선명도/평탄도와 일반화 능력 사이의 관계를 다루었습니다. Hochreiter와 슈미드후버[18]는 "평탄도"를 훈련 손실이 낮게 유지되는 최소값 주변의 연결된 영역의 크기로 정의했습니다. Keskar 등[24]은 다음과 같이 특징짓습니다.

[1]코드와 플롯은 `https://github.com/tomgoldstein/loss-landscape` 에서 확인할 수 있습니다.

의 고유값을 사용하여 평탄도를 측정하고, 최소값의 인근에서 최대 손실을 살펴보는 근사치로 ϵ-선명도를 제안합니다. Dinh 등[7], Neyshabur 등[30]은 이러한 선명도의 정량적 측정값이 네트워크의 대칭성에 불변하지 않으므로 일반화 능력을 결정하기에 충분하지 않다는 것을 보여줍니다. Chaudhari 등[3]은 선명도의 척도로 국부 엔트로피를 사용했는데, 이는 [7]의 단순 변환에 불변하지만 정확하게 계산하기 어렵습니다. Dziugaite와 Roy [8]는 일반화를 위해 선명도를 PAC-Bayes 경계에 연결합니다.

3 손실 함수 시각화의 기초

신경망은 특징 벡터(예: 이미지) $\{x_i\}$의 말뭉치와 그에 수반되는 레이블 $\{y_i\}$의 손실을 최소화하여 $L(\vartheta) = \frac{1}{m}\sum_{i=1}^{m} l(x_i, y_i; \vartheta)$, 여기서 θ는 신경망의 매개변수(가중치)인 $l(x_i, y_i; \vartheta)$ 함수는 매개변수 ϑ를 가진 신경망이 데이터 샘플의 레이블을 얼마나 잘 예측하는지를 측정하며, m은 데이터 샘플의 수입니다. 신경망은 많은 매개변수를 포함하므로 손실 함수는 매우 고차원적인 공간에 존재합니다. 안타깝게도 시각화는 저차원의 1차원(선) 또는 2차원(표면) 플롯을 사용해야만 가능합니다. 이러한 차원 차이를 좁히기 위한 몇 가지 방법이 존재합니다.

1차원 선형 보간 손실 함수를 그리는 간단하고 가벼운 방법 중 하나는 두 개의 매개변수 θ와 ϑ'를 선택하고 이 두 점을 연결하는 선을 따라 손실 함수의 값을 그리는 것입니다. 스칼라 매개변수 α를 선택하고 가중 평균 $\vartheta(\alpha) = (1-\alpha)\vartheta + \alpha\vartheta'$를 정의하여 이 선을 매개변수화할 수 있습니다. 마지막으로 함수 $f(\alpha) = L(\vartheta(\alpha))$를 플롯합니다. 이 전략은 굿펠로우 등[13]이 취한 것으로, 이들은 다음 사이의 선을 따라 손실 표면을 연구했습니다.

무작위 초기 추측과 확률적 기울기 하강을 통해 얻은 근사 최소값을 사용합니다. 이 방법은 다양한 최소값의 "선명도"와 "평탄도", 배치 크기에 대한 선명도의 의존성을 연구하는 데 널리 사용되어 왔습니다[24, 7]. 스미스와 토핀[34]은 동일한 기법을 사용하여 서로 다른 최소값과 그 사이의 "피크"를 보여주며, 임 등[21]은 서로 다른 최적화기를 통해 얻은 최소값 사이의 선을 플롯합니다.

1D 선형 보간 방법에는 몇 가지 약점이 있습니다. 첫째, 1D 플롯을 사용하여 비볼록성을 시각화하기 어렵다는 점입니다. 실제로 굿펠로우 등[13]은 손실 함수가 최소화 궤적을 따라 국소 최소값이 없는 것처럼 보인다는 사실을 발견했습니다. 나중에 2D 방법을 사용하여 일부 손실 함수가 극단적인 비볼록성을 가지고 있으며 이러한 비볼록성이 서로 다른 네트워크 아키텍처 간의 일반화 차이와 상관관계가 있음을 살펴볼 것입니다. 둘째, 이 방법은 네트워크에서 일괄 정규화[22] 또는 불변 대칭을 고려하지 않습니다. 이러한 이유로 1D 보간 플롯에서 생성된 시각적 선명도 비교는 오해의 소지가 있을 수 있으며, 이 문제는 섹션 5에서 자세히 살펴볼 것입니다.

등고선 플롯 및 임의 방향 이 방법을 사용하려면 그래프에서 중심점 ϑ^* 을 선택하고 두 방향 벡터인 δ 및 η 을 선택합니다. 그런 다음 1D(선)의 경우 $f(\alpha) = L(\vartheta^* + \alpha\delta)$ 형식의 함수를 그리거나

$$f(\alpha, \beta) = L(\vartheta^* + \alpha\delta + \beta\eta) \tag{1}$$

2D(표면)의 경우[2]. 이 접근법은 [13]에서 다양한 최소화 방법의 궤적을 탐색하는 데 사용되었습니

다. 또한 [21]에서는 다른 최적화 알고리즘이 2D 투영 공간 내에서 서로 다른 국소 최소값을 찾는다는 것을 보여주기 위해 사용되었습니다. 2D 플롯의 계산 부담으로 인해 이러한 방법은 일반적으로 손실 표면의 복잡한 비볼록성을 포착하지 못한 작은 영역의 저해상도 플롯을 생성합니다. 아래에서는 큰 가중치 공간에 대한 고해상도 시각화를 사용하여 네트워크 설계가 비볼록 구조에 어떤 영향을 미치는지 시각화합니다.

4 제안된 시각화: 필터 기반 정규화

이 연구에서는 적절한 스케일링이 적용된 무작위 가우스 분포에서 각각 샘플링된 무작위 방향 벡터 δ 및 η를 사용하여 생성된 (1)과 같은 형태의 플롯에 크게 의존합니다(아래 설명 참조). 플롯에 대한 "무작위 방향" 접근 방식은 간단하지만 손실 표면의 고유한 기하학적 구조를 포착하지 못하며, 두 개의 서로 다른 최소화기 또는 두 개의 서로 다른 최소화기의 기하학적 구조를 비교하는 데 사용할 수 없습니다.

[2]이 백서에서 2D 플롯을 만들 때 배치 정규화 매개변수는 일정하게 유지됩니다. 즉, 배치 정규화 매개변수에 임의의 방향이 적용되지 않습니다.

네트워크. 이는 네트워크 가중치의 *스케일 불변성 때문입니다*. ReLU 비선형성을 사용하면, 예를 들어 네트워크의 한 계층에 가중치를 10으로 곱하고 다음 계층을 10으로 나누면 네트워크는 변하지 않습니다. 이러한 불변성은 일괄 정규화를 사용할 때 더욱 두드러집니다. 이 경우 일괄 정규화 중에 각 계층의 출력 크기가 다시 조정되기 때문에 필터의 크기(즉, 규범)는 관련이 없습니다. 따라서 가중치를 다시 조정해도 네트워크의 동작은 변경되지 않습니다. 이 스케일 불변성은 정류된 네트워크에만 적용됩니다.

스케일 불변성은 특별한 예방 조치를 취하지 않는 한 플롯 간에 의미 있는 비교를 할 수 없게 합니다. 가중치가 큰 신경망은 손실 함수가 부드럽고 천천히 변화하는 것처럼 보일 수 있으며, 가중치가 1보다 훨씬 큰 스케일일 경우 가중치를 1 단위로 교란해도 네트워크 성능에 미치는 영향은 거의 없습니다. 그러나 가중치가 1보다 훨씬 작으면 동일한 단위의 교 란 이 치명적인 영향을 미칠 수 있으므로 손실 함수가 가중치 교란에 매우 민감하게 나타납니다. 신경망은 규모 불변이므로 이 예에서 작은 매개변수 네트워크와 큰 매개변수 네트워크가 동일하다면(하나는 단순히 다른 하나의 스케일을 재조정하는 것이므로), 손실 함수의 명백한 차이는 규모 불변의 인공물에 불과합니다. 이 스케일 불변성은 Dinh 등[7]이 겉보기 선명도가 다른 동등한 네트워크 쌍을 구축하는 데 이용되었습니다.

이러한 스케일링 효과를 제거하기 위해 필터별로 정규화된 방향을 사용하여 손실 함수를 플롯합니다. 파라미터 θ가 있는 네트워크에 대해 이러한 방향을 얻으려면 먼저 ϑ*와* 호환되는 차원의 무작위 가우스 방향 벡터 d*를* 생성합니다. 그런 다음 각 필터를 d*로* 정규화하여 동일한 값을 갖도록 합니다
.

θ에서 해당 필터의 노름을 대체합니다. 즉, 대체 $d_{i,j} \leftarrow \frac{d_{i,j}}{\|d_{i,j}\|} \|\vartheta_{i,j}\|$,

여기서 $d_{i,j}$ *는* d*의* i*th* 레이어의 j*번째* 필터(j*번째* 가중치가 아님)를 나타내고$\|\cdot\|$ *는* 프로베니우스 노름을 나타냅니다. 필터별 정규화는 개별 필터의 규범을 고려하지 않고 방향만 정규화하는 [21]의 정규화 방식과 다르다는 점에 유의하세요. 필터 노멀라이제이션은 컨볼루션(Conv) 레이어에만 국한되지 않고 완전 연결(FC) 레이어에도 적용된다는 점에 유의하세요. FC 레이어는 1 × 1 출력 피처 맵을 가진 Conv 레이어와 동일하며 필터는 하나의 뉴런을 생성하는 가중치에 해당합니다.

방향 δ와 η*가* 필터 정규화된 경우 (1)과 같은 형태의 등고선 플롯이 손실 표면의 자연스러운 거리 척도를 포착할 수 있을까요? 섹션 5에서는 필터 정규화된 플롯의 선명도가 일반화 오류와 잘 상관관계가 있는 반면, 필터 정규화가 없는 플롯은 매우 오해의 소지가 있음을 보여줌으로써 이 질문에 긍정적으로 대답합니다. 부록 A.2에서는 필터별 정규화와 레이어별 정규화(및 정규화 없음)를 비교하여 필터 정규화가 선명도와 일반화 오류 간에 우수한 상관관계를 생성한다는 것을 보여줍니다.

5 날카로운 대 평평한 딜레마

섹션 4에서는 필터 정규화의 개념을 소개하고 그 사용에 대한 직관적인 근거를 제시합니다. 이 섹션에서는 선명한 최소화기가 평평한 최소화기보다 일반화가 더 잘 되는지에 대한 문제를 다룹니다. 이를 통해 필터 정규화를 사용할 때 최소화기의 선명도가 일반화 오류와 밀접한 상관관계가 있음

을 알 수 있습니다. 이를 통해 플롯을 나란히 비교할 수 있습니다. 반대로 정규화되지 않은 플롯의 선명도는 왜곡되고 예측할 수 없는 것처럼 보일 수 있습니다.

소량의 SGD는 일반화가 잘 되는 "평평한" 최소값을 생성하는 반면, 대규모 배치는 일반화가 잘 되지 않는 "날카로운" 최소값을 생성한다고 널리 알려져 있습니다[3, 24, 18]. 하지만 Dinh 등[7], Kawaguchi 등[23]은 일반화가 손실 표면의 곡률과 직접적인 관련이 없다고 주장하고, 일부 저자는 배치 크기가 클수록 좋은 성능을 달성하는 특화된 훈련 방법을 제안하는 등 이 주장에 대한 논쟁이 있습니다[19, 14, 6]. 여기에서는 날카로운 최소화기와 평평한 최소화기의 차이점을 살펴봅니다. 먼저 이러한 시각화를 수행할 때 발생하는 어려움과 적절한 정규화를 통해 이러한 플롯이 왜곡된 결과를 생성하는 것을 방지할 수 있는 방법에 대해 논의합니다.

고정된 수의 에포크에 대해 배치 정규화를 사용하는 9층 VGG 네트워크[33]를 사용하여 CIFAR-10 분류기를 훈련합니다. 두 가지 배치 크기를 사용합니다. 큰 배치 크기인 8192(CIFAR-10 훈련 데이터의 16.4%)와 작은 배치 크기인 128입니다. ϑ^s 와 ϑ^l 는 각각 작은 배치 크기와 큰 배치 크기를 사용하여 SGD를 실행하여 얻은 해를 나타냅니다.[3]. 선형 보간 접근법 사용

[3]이 섹션에서는 "주행 평균"과 "주행 분산"을 훈련 가능한 파라미터로 간주하여 θ에 포함시킵니다. Goodfellow 등[13]의 원래 연구에서는 배치 정규화를 고려하지 않았습니다. 이러한 파라미터는 두 최소화기 사이에서 보간할 때만 필요하므로 향후 섹션에서는 θ에 포함되지 않습니다.

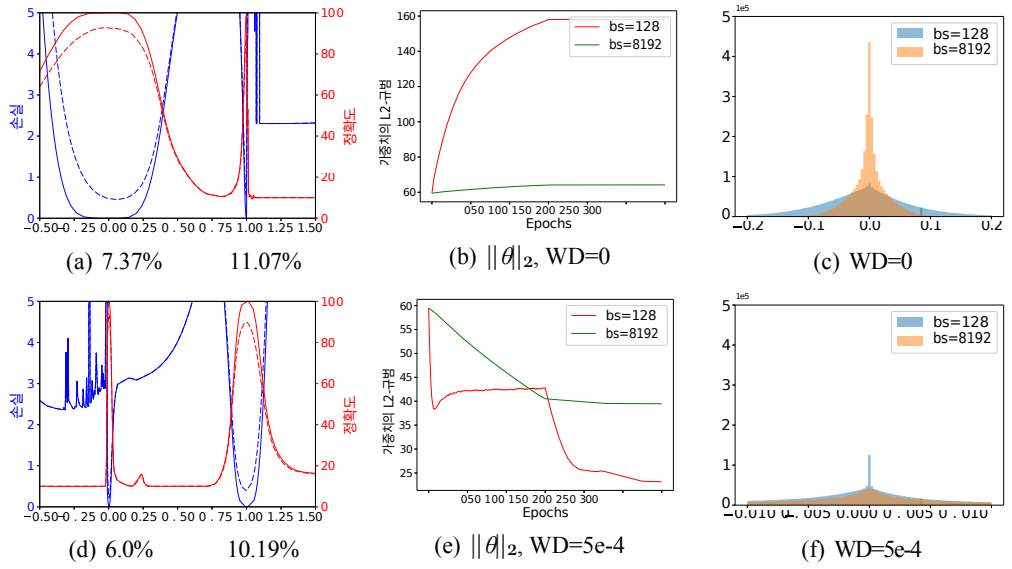


그림 2: (a)와 (d)는 소규모 배치 및 대규모 배치 훈련 방법으로 얻은 VGG-9 솔루션의 1D 선형 보간 결과입니다. 파란색 선은 손실 값이고 빨간색 선은 정확도입니다. 실선은 훈련 곡선이고 점선은 테스트용 곡선입니다. 소규모 배치가 0, 대규모 배치가 1입니다. 해당 테스트 오류는 아래와 같습니다. (b) 및 (e)는 훈련 중 가중치 표준$\|\vartheta\|_2$ 의 변화를 보여줍니다. 가중치 감쇠가 비활성화되면 가중치 규범은 제약 없이 훈련 중에 꾸준히 증가합니다. (c)와 (f)는 가중치 히스토그램으로, 소규모 배치 방식이 가중치 감쇠가 0인 큰 가중치와 0이 아닌 작은 가중치를 더 많이 생성한다는 것을 확인할 수 있습니다.

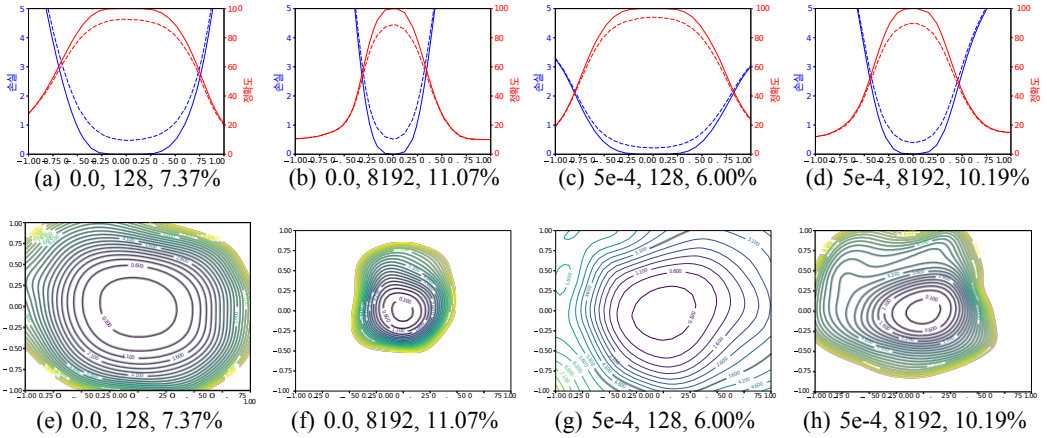
[13]에서 두 솔루션을 포함하는 방향, 즉 $f(\alpha) = L(\vartheta^s + \alpha(\theta^l - \vartheta^s))$을 따라 CIFAR-10의 훈련 및 테스트 데이터 세트의 손실값을 플롯합니다.

그림 2(a)는 x축 위치 0에서 ϑ^s , 위치 1에서 ϑ^l 의 선형 보간 플롯을 보여줍니다. 24]에서 관찰한 것처럼, 소량 배치 솔루션은 상당히 넓고 대량 배치 솔루션은 선명하다는 것을 분명히 알 수 있습니다 . 그러나 이러한 선명도 균형은 가중치 감쇠를 켜는 것만으로 간단히 뒤집을 수 있습니다 [25]. 그림 2(d)는 동일한 실험의 결과를 보여주지만 이번에는 0이 아닌 가중치 감쇠 매개변수를 사용했습니다. 이번에는 큰 배치 최소화기가 날카로운 작은 배치 최소화기보다 상당히 평평합니다. 그러나 모든 실험에서 작은 배치가 더 잘 일반화되는 것을 볼 수 있으며 선명도와 일반화 사이에는 뚜렷한 상관관계가 없습니다. 이러한 선명도 비교는 극도로 오해의 소지가 있으며 최소값의 내생적 속성을 포착하지 못한다는 것을 알 수 있습니다.

선명도의 명백한 차이는 각 최소화기의 가중치를 조사하여 설명할 수 있습니다. 네트워크 가중치의 히스토그램은 그림 2(c)와 (f)에 각 실험에 대해 나와 있습니다. 가중치 붕괴가 없는 큰 배치를 사용할 경우, 결과 가중치가 작은 배치의 경우보다 작아지는 경향이 있음을 알 수 있습니다. 가중치 감쇠를 추가하면 이 효과가 반전되는데, 이 경우 큰 배치 최소화기는 작은 배치 최소화기보다 훨씬 더 큰 가중치를 갖습니다. 이러한 규모의 차이는 간단한 이유 때문에 발생합니다: 배치 크기가 작을수록 배치 크기가 클 때보다 에포크당 가중치 업데이트가 더 많이 발생하므로 가중치 감쇠(가중치 규

범에 페널티를 부과하는)의 축소 효과가 더 뚜렷하게 나타납니다. 훈련 중 가중치 규범의 진화는 그림 2(b)와 (e)에 나와 있습니다. 그림 2는 최소화 기법의 내생적 선명도를 시각화한 것이 아니라 (관련 없는) 가중치 스케일링만 보여줍니다. 배치 정규화는 단위 분산을 갖도록 출력의 스케일을 다시 조정하기 때문에 이러한 네트워크에서 가중치의 스케일링은 관련이 없습니다. 그러나 작은 가중치는 여전히 섭동에 더 민감하게 나타나며 더 선명하게 보이는 최소화자를 생성합니다.

필터 정규화 플롯 그림 2의 실험을 반복하지만 이번에는 임의의 필터 정규화 방향을 사용하여 각 최소값 근처의 손실 함수를 개별적으로 플롯합니다. 이렇게 하면 그림 2(c)와 (f)에 표시된 스케일링으로 인한 지오메트리의 명백한 차이를 제거할 수 있습니다. 그림 3에 표시된 결과는 여전히 작은 배치와 큰 배치 최소값 간의 선명도 차이를 보여주지만, 이러한 차이는 정규화되지 않은 플롯에서 나타나는 것보다 훨씬 더 미묘합니다. 비교를 위해 정규화되지 않은 플롯 샘플과 레이어 정규화 플롯 샘플이 섹션 A.2에 나와 있습니다.



(a) 0.0, 128, 7.37% (b) 0.0, 8192, 11.07% (c) 5e-4, 128, 6.00% (d) 5e-4, 8192, 10.19%

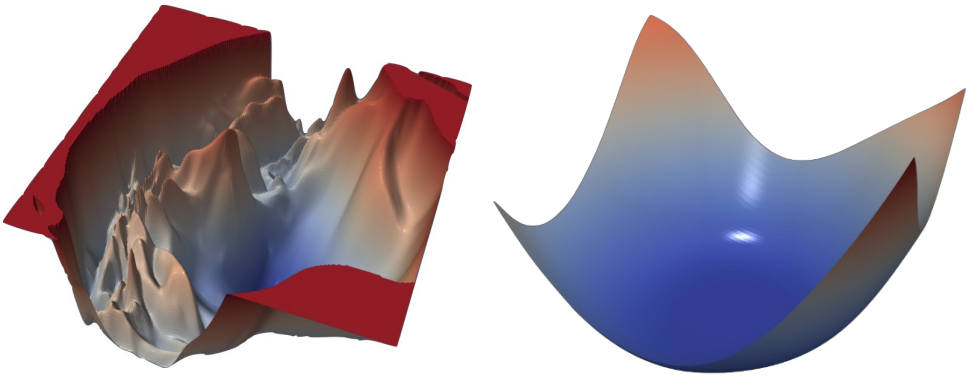
(e) 0.0, 128, 7.37% (f) 0.0, 8192, 11.07% (g) 5e-4, 128, 6.00% (h) 5e-4, 8192, 10.19%

그림 3: 다양한 가중치 감쇠와 배치 크기로 SGD를 사용하여 얻은 솔루션의 1D 및 2D 시각화. 각 하위 그림의 제목에는 가중치 감쇠, 배치 크기 및 테스트 오류가 포함되어 있습니다.

부록을 참조하세요. 또한 두 개의 임의 방향과 등고선 플롯을 사용하여 이러한 결과를 시각화합니다. 작은 배치 크기와 0이 아닌 가중치 감쇠로 얻은 가중치는 더 선명한 대형 배치 최소화기보다 더 넓은 윤곽을 가집니다. ResNet-56의 결과는 부록의 그림 15에 나와 있습니다. 그림 3의 필터 정규화 플롯을 사용하면 최소화기 간에 나란히 비교할 수 있으며, 이제 선명도가 일반화 오류와 잘 상관관계가 있음을 알 수 있습니다. 대규모 배치는 시각적으로 더 선명한 최소값을 생성했지만(극적인 차이는 아니지만) 테스트 오류는 더 높았습니다.

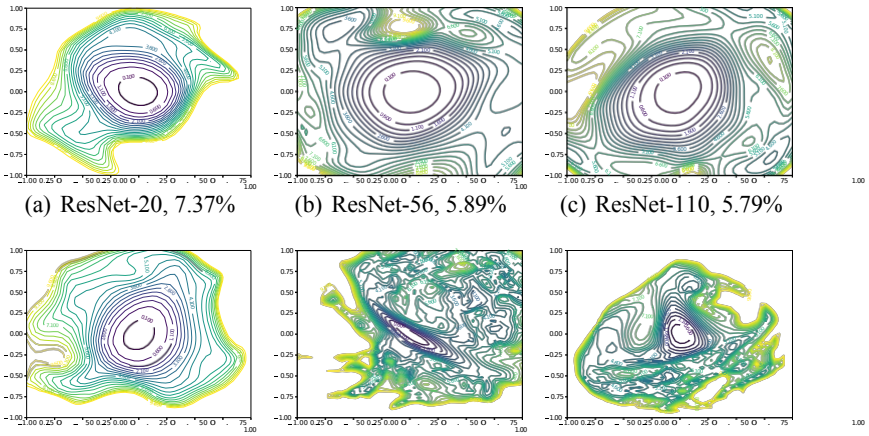
6 무엇이 신경망을 훈련 가능하게 만들까요? 손실 표면의 (비)볼록성 구조에 대한 인사이트

신경 손실 함수에 대한 전역 최소화자를 찾는 능력은 보편적인 것이 아니며, 일부 신경 아키텍처는 다른 아키텍처보다 최소화하기가 더 쉬운 것으로 보입니다. 예를 들어, [17]의 저자들은 스킵 연결을 사용하여 매우 심층적인 아키텍처를 훈련한 반면, 스킵 연결이 없는 유사한 아키텍처는 훈련할 수 없었습니다. 또한, 우리의 훈련 능력은 훈련을 시작하는 초기 매개 변수에 크게 좌우되는 것으로 보입니다. 시각화 방법을 사용하여 신경 아키텍처에 대한 경험적 연구를 수행하여 손실 함수의 비볼록성이 어떤 상황에서는 문제가 되는 것처럼 보이지만 다른 상황에서는 문제가 되지 않는 이유를 탐구합니다. 다음 질문에 대한 인사이트를 제공하는 것을 목표로 합니다: 손실 함수에 심각한 비볼록성이 있는가? 눈에 띄는 비볼록성이 존재한다면 왜 모든 상황에서 문제가 되지 않을까요? 어떤 아키텍처는 훈련하기 쉬운 반면, 어떤 아키텍처는 초기화에 민감하게 반응하는 이유는 무엇인가요? 아키텍처에 따라 이러한 질문에 답할 수 있는 비볼록성 구조에 극명한 차이가 있으며, 이러한 차이가 일반화 오류와 상관관계가 있다는 것을 알게 될 것입니다.



(a) ResNet-110, 스킵 연결 없음 (b) DenseNet, 121 레이어

그림 4: CIFAR-10용 ResNet-110-noshort 및 DenseNet의 손실 표면.



(a) ResNet-20, 7.37% (b) ResNet-56, 5.89% (c) ResNet-110, 5.79%

(d) ResNet-20-NS, 8.18% (e) ResNet-56-NS, 13.31% (f) ResNet-110-NS, 16.44%

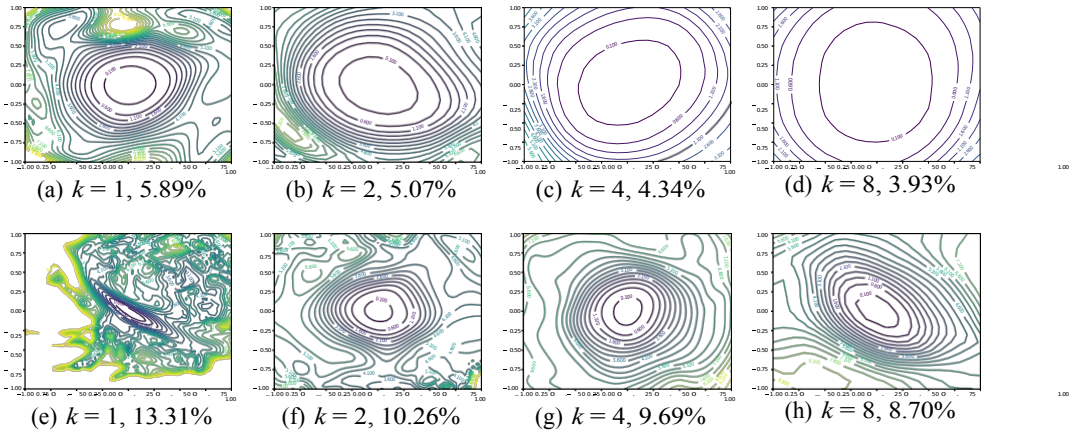
그림 5: 깊이가 다른 ResNet과 ResNet-noshort의 손실 표면 2D 시각화.

실험 설정 비볼록성에 대한 네트워크 아키텍처의 효과를 이해하기 위해 여러 네트워크를 훈련하고 섹션 4에서 설명한 필터 정규화된 무작위 방향 방법을 사용하여 얻은 최소값을 중심으로 풍경을 플롯했습니다. 세 가지 종류의 신경망을 고려했습니다: 1) CIFAR-10의 성능에 최적화된 ResNet [17]. 여기서는 ResNet-20/56/110을 고려하며, 각 이름에는 레이어 수에 따라 레이블이 지정됩니다. 2) 바로가기/스킵 연결을 포함하지 않는 "VGG와 유사한" 네트워크. 이러한 네트워크는 ResNet에서 바로 가기 연결을 제거하여 만들었습니다. 이러한 네트워크를 ResNet-20/56/110-noshort라고 부릅니다. 3) "와이드" ResNet은 CIFAR-10에 최적화된 네트워크보다 레이어당 필터 수가 더 많습니다. 모든 모델은 네스테로프 모멘텀, 배치 크기 128, 가중치 감쇠 0.0005의 SGD를 사용하여 300개의 에포크에 대해 CIFAR-10 데이터 세트에서 훈련되었습니다. 학습 속도는 0.1로 초기화되었으며 150, 225, 275 에포크에서 10배씩 감소했습니다. 더 심층적인 실험용 VGG 유사 네트워크(예: 아래에 설명된 ResNet-56-noshort)는 초기 학습률이 0.01로 더 낮아야 했습니다. 다양한 신경망에 대한 최소화기의 고해상도 2D 플롯은 그림 5와 그림 6에 나와 있습니다. 결과가 표면 플롯이 아닌 윤곽 플롯으로 표시된 이유는 볼록하지 않은 구조를 보고 선명도를 평가하기가 매우 쉽기 때문입니다. ResNet-56의 표면 플롯은 그림 1을 참조하십시오. 각 플롯의 중심은 최소값에 해당하며, 두 축은 (1)에서와 같이 필터별 정규화를 통해 두 개의 임의 방향을 매개변수화한다는 점에 유의하세요. 아키텍처가 손실 환경에 미치는 영향에 대해 아래에서 몇 가지 관찰 결과를 살펴봅니다.

네트워크 깊이의 효과 그림 5에서 네트워크 깊이는 스킵 연결을 사용하지 않을 때 신경망의 손실 표면에 극적인 영향을 미친다는 것을 알 수 있습니다. ResNet-20- noshort 네트워크는 중앙에 볼록한 윤곽을 가진 영역이 지배적이며 극적인 비볼록성이 없는 상당히 양호한 환경을 가지고 있습니다. 이는 그다지 놀라운 일이 아닙니다. ImageNet의 원래 VGG 네트워크는 19개의 레이어를 가지고 있었고 효과적으로 훈련할 수 있었습니다[33]. 그러나 네트워크 깊이가 증가함에 따라 VGG와 같은 네트워크의 손실 표면은 (거의) 볼록에서 혼돈으로 자연적으로 전환됩니다. ResNet-56-noshort는 극적인 비볼록성과 그래디언트 방향(플롯에 표시된 윤곽선에 대해 정상임)이 중앙의 최소자를 가리키지 않는 넓은 영역을 가지고 있습니다. 또한 일부 방향으로 이동하면 손실 함수가 매우 커집

니다. ResNet-110-noshort는 훨씬 더 극적인 비볼록성을 표시하며 플롯에 표시된 모든 방향으로 이동함에 따라 극도로 가파르게 됩니다. 또한 깊은 VGG와 같은 그물망의 중앙에 있는 최소값이 상당히 날카로운 것을 알 수 있습니다. ResNet-56-noshort의 경우, 최소화기 근처의 윤곽이 상당한 편심을 가지고 있기 때문에 최소화기도 상당히 상태가 좋지 않습니다.

구조에 대한 바로 가기 연결 바로 가기 연결은 손실 함수의 지오메트리에 따라 극적인 효과를 발휘합니다. 그림 5에서는 잔여 연결이 깊이가 증가함에 따라 혼돈스러운 동작으로 전환되는 것을 방지하는 것을 볼 수 있습니다. 실제로 0.1레벨 윤곽의 폭과 모양은 20층 네트워크와 110층 네트워크에서 거의 동일합니다. 흥미롭게도 스킵 연결의 효과는 깊은 네트워크에서 가장 중요한 것으로 보입니다. 보다 얕은 네트워크(ResNet-20 및 ResNet-20-noshort)의 경우 건너뛰기 연결의 효과는 상당히 눈에 띄지 않습니다. 그러나 잔여 연결은 네트워크가 깊어질 때 발생하는 비볼록성의 폭발적인 증가를 방지합니다. 이 효과는 다른 종류에도 적용되는 것으로 보입니다.



(a) $k = 1$, 5.89% (b) $k = 2$, 5.07% (c) $k = 4$, 4.34% (d) $k = 8$, 3.93%

(e) $k = 1$, 13.31% (f) $k = 2$, 10.26% (g) $k = 4$, 9.69% (h) $k = 8$, 8.70%

그림 6: 바로가기 연결이 있는 경우(상단)와 없는 경우(하단) 모두 CIFAR-10의 Wide-ResNet-56. 라벨 $k = 2$는 레이어당 필터 수가 두 배임을 의미합니다. 각 그림 아래에 테스트 오류가 보고되어 있습니다.

그림 4는 눈에 띄는 비볼록성을 보이지 않는 DenseNet [20]의 손실 경관을 보여줍니다.

와이드 모델과 씬 모델 레이어당 컨볼루션 필터 수의 효과를 확인하기 위해 레이어당 필터 수에 $k = 2, 4, 8$을 곱하여 좁은 CIFAR 최적화 ResNet(ResNet-56)을 와이드-ResNet[41]과 비교합니다. 그림 6에서 더 넓은 모델에서 눈에 띄는 혼란스러운 동작이 없는 손실 풍경을 볼 수 있습니다. 네트워크 폭이 증가하면 최소값이 평평해지고 볼록한 영역이 넓어집니다. 폭을 늘리면 혼란스러운 동작을 방지할 수 있으며, 건너뛰기 연결은 최소값을 크게 넓히는 것을 알 수 있습니다. 마지막으로 선명도는 테스트 오류와 매우 밀접한 상관관계가 있다는 점에 유의하십시오.

네트워크 초기화에 대한 시사점 그림 5에서 볼 수 있는 한 가지 흥미로운 특성은 고려된 모든 네트워크의 손실 풍경이 손실 값이 낮고 볼록한 윤곽으로 잘 정의된 영역과 높은 손실 값과 비볼록한 윤곽으로 잘 정의된 영역으로 분할되어 있는 것처럼 보인다는 것입니다. 이러한 혼돈 영역과 볼록 영역의 분할은 좋은 초기화 전략의 중요성과 "좋은" 아키텍처의 쉬운 훈련 동작을 설명할 수 있습니다. 글로롯과 벤지오[11]가 제안한 것과 같은 정규화된 무작위 초기화 전략을 사용할 경우, 일반적인 신경망은 초기 손실 값이 2.5 미만에 도달합니다. 그림 5의 잘 작동하는 손실 풍경(ResNet 및 얕은 VGG 유사 네트워크)은 4 이상의 손실 값으로 상승하는 크고 평평하며 거의 볼록한 끌어당김이 지배적입니다. 이러한 랜드스케이프의 경우 무작위 초기화는 "잘 작동하는" 손실 영역에 위치할 가능성이 높으며, 최적화 알고리즘은 높은 손실의 카오스 고원에서 발생하는 병적인 비볼록성을 "보지" 못할 수도 있습니다. 카오틱 손실 지형(ResNet-56/110- 노쇼트)은 더 낮은 손실 값으로 상승하는 더 얕은 볼록 영역을 갖습니다. 인력이 충분히 얕고 충분히 깊은 네트워크의 경우 초기 반복은 그라데이션이 정보가 없는 카오스 영역에 위치할 가능성이 높습니다. 이 경우 기울기가 "산산조각"[1]나면서 훈련이 불가능해집니다. SGD는 매우 낮은 학습 속도에서도 스킵 연결 없이 156개의 레이어 네트워크를 훈련할 수 없었으며, 이는 이 가설에 무게를 더합니다.

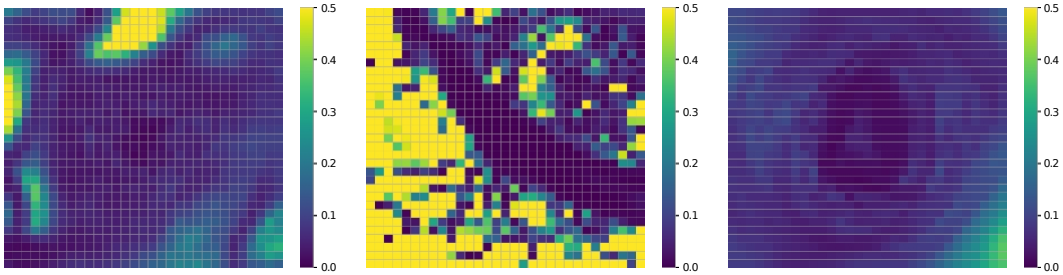
일반화에 영향을 미치는 랜드스케이프 지오메트리 그림 5와 6은 랜드스케이프 지오메트리가 일반화에 극적인 영향을 미친다는 것을 보여줍니다. 첫째, 시각적으로 더 평평한 최소값이 일관되게 테스트 오류를 낮추는 것으로 나타나 필터 정규화가 손실 함수 기하구조를 시각화하는 자연스러운 방법이라는 주장을 더욱 강화합니다. 둘째, 혼란스러운 랜드스케이프(스킵 연결이 없는 딥 네트워크)는 훈련 및 테스트 오류를 악화시키는 반면, 볼록한 랜드스케이프는 오류 값이 더 낮다는 것을 알 수 있습니다. 실제로 가장 볼록한 랜드스케이프(그림 6의 맨 윗줄에 있는 와이드 레스넷)는 가장 잘 일반화되며 눈에 띄는 혼란스러운 동작을 보이지 않습니다.

주의 사항입니다: 정말 볼록성을 보고 있나요? 우리는 극적으로 차원이 감소된 손실 표면을 보고 있으며, 이러한 플롯을 해석하는 방법에 주의해야 합니다. 손실 함수의 볼록성 수준을 측정하는 한 가지 방법은 헤시안 고유값인 *윤곡률을* 계산하는 것입니다. 진정한 볼록 함수는 음수가 아닌 곡률(양수인 경우

반정확 헤시안), 비볼록 함수는 음의 곡률을 갖습니다. 차원 축소 플롯(랜덤 가우스 방향 포함)의 기본 곡률은 전체 차원 표면의 기본 곡률의 가중 평균(가중치는 카이제곱 무작위 변수)임을 알 수 있습니다.

이는 몇 가지 결과를 가져옵니다. 우선, 차원이 축소된 플롯에 볼록하지 않은 것이 존재한다면 전체 차원 표면에도 볼록하지 않은 것이 존재해야 합니다. 그러나 저차원 서페이스에서 볼록성이 보인다고 해서 고차원 함수가 실제로 볼록하다는 의미는 아닙니다. 오히려 양의 곡률이 우세하다는 것을 의미합니다(보다 공식적으로는 *평균* 곡률 또는 평균 고유값이 양수입니다).

이 분석은 안심할 수 있지만, 이러한 시각화에서 포착하지 못한 중요한 '숨겨진' 비볼록성이 있는지 궁금할 수 있습니다. 이 질문에 답하기 위해 헤시안, λ_{min} 및 λ_{max} *의 최소* 및 *최대* 고유값을 계산합니다.[4] 그림 7은 위에서 연구한 손실 표면의 비율 $|\lambda_{min}/\lambda_{max}|$을 매핑한 것입니다(동일한 최소값과 동일한 임의 방향 사용). 파란색은 더 볼록한 영역(양의 고유값에 비해 음의 고유값이 거의 0에 가까움)을 나타내고, 노란색은 상당한 수준의 음의 곡률을 나타냅니다. 서피스 플롯에서 볼록하게 보이는 영역은 실제로 음의 고유값이 미미한 영역에 해당하며(즉, 플롯에서 놓친 주요 비볼록 특징이 없음), 혼란스러운 영역은 큰 음의 곡률을 포함하고 있음을 알 수 있습니다. DenseNet과 같이 볼록하게 보이는 표면의 경우 음의 고유값은 플롯의 넓은 영역에서 극히 작게 유지됩니다(양의 곡률 크기의 1% 미만).



(a) Resnet-56 (b) Resnet-56-noshort (c) DenseNet-121

그림 7: 필터 정규화된 표면 플롯의 각 점에 대해 헤시안 최대값과 최소 고유값을 계산하고 이 두 값의 비율을 매핑합니다.

7 최적화 경로 시각화

마지막으로 다양한 옵티마이저의 궤적을 시각화하는 방법을 살펴봅니다. 이 응용 프로그램에서는 임의의 방향은 효과적이지 않습니다. 무작위 방향이 실패하는 이유에 대한 이론적 설명을 제공하고 손실 함수 윤곽선 위에 궤적을 효과적으로 플로팅하는 방법을 살펴봅니다.

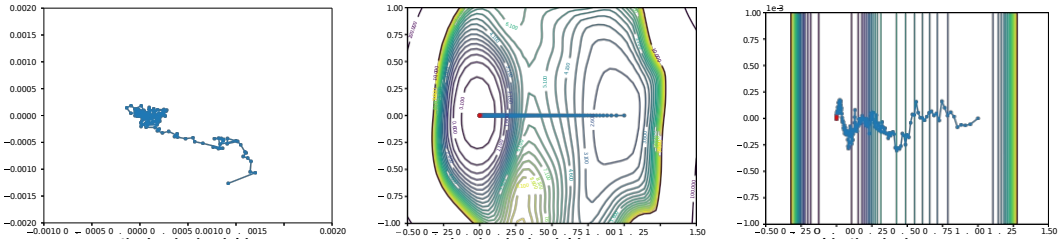
여러 저자들은 무작위 방향이 최적화 궤적의 변화를 포착하지 못한다는 사실을 관찰했습니다[10, 29, 28, 27]. 실패한 시각화의 예는 그림 8에 나와 있습니다. 그림 8(a)에서는 두 개의 임의 방향에 의해 정의된 평면에 투영된 SGD의 반복을 볼 수 있습니다. 거의 모든 동작이 캡처되지 않습니다(슈퍼 줌인된 축과 무작위로 걷는 것처럼 보이는 것을 볼 수 있습니다). 이 문제를 발견한 [13]은 초

기화에서 해를 가리키는 한 방향과 임의의 방향 하나를 사용하여 궤적을 시각화했습니다. 이 접근 방식은 그림 8(b)에 나와 있습니다. 그림 8(c)에서 볼 수 있듯이 무작위 축은 변화를 거의 포착하지 못하여 (오해의 소지가 있는) 직선 경로처럼 보이게 됩니다.

7.1 무작위 방향이 실패하는 이유: 저차원 최적화 궤적

고차원 공간에서 두 개의 랜덤 벡터는 높은 확률로 거의 직교할 것이라는 것은 잘 알려져 있습니다. 실제로 n차원에서 가우스 랜덤 벡터 간의 예상 코사인 유사성은 대략 $2/(\pi n)$ 입니다([12], 정리 5).

[4]자동 미분을 사용하여 직접 계산되는 헤시안 벡터 곱만 필요하고 헤시안 또는 그 인수 분해의 명시적 표현이 필요하지 않은 암시적으로 재시작된 Lanczos 방법을 사용하여 계산합니다.



(a) 두 개의 임의 방향 (b) Y축의 임의 방향 (c) 확대 버전(b)

그림 8: 최적화 궤적의 비효율적인 시각화. 이러한 시각화는 고차원에서 임의 방향의 직교성 문제로 인해 어려움을 겪습니다.

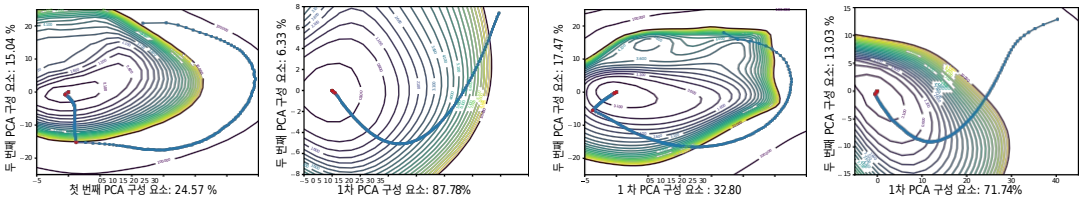
이는 최적화 궤적이 극히 낮은 차원 공간에 있을 때 문제가 됩니다. 이 경우 무작위로 선택된 벡터는 최적화 경로가 포함된 저차원 공간에 직교하게 되며, 임의의 방향으로 투영하면 거의 변화가 포착되지 않습니다. 그림 8(b)는 무작위 방향이 최적화 경로를 따라 가리키는 벡터보다 훨씬 적은 변동을 포착하기 때문에 최적화 궤적이 저차원임을 시사합니다. 아래에서는 PCA 방향을 사용하여 이러한 저차원성을 직접 검증하고 효과적인 시각화를 생성합니다.

7.2 PCA 방향을 사용한 효과적인 궤적 그리기

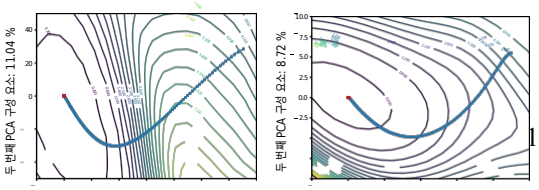
궤적의 변화를 포착하려면 무작위가 아닌 (신중하게 선택한) 방향을 사용해야 합니다. 여기에서는 PCA를 기반으로 한 접근 방식을 제안하여 캡처한 변동의 양을 측정하고 손실 표면의 윤곽을 따라 이러한 궤적을 플롯으로 제공합니다.

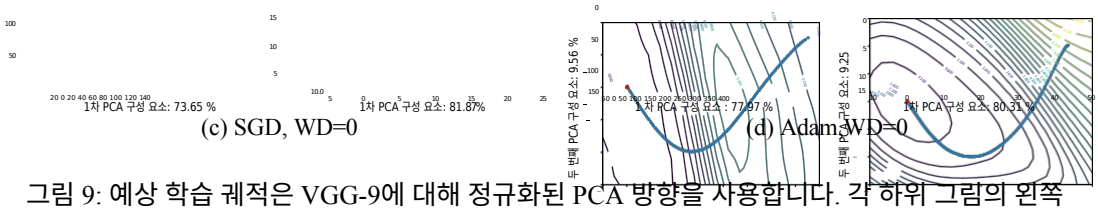
ϑ_i 는 에포크 i 의 모델 파라미터를 나타내고, n *개의* 에포크 훈련 후의 최종 파라미터는 ϑ_n 로 표시됩니다. n*개의* 훈련 에포크가 주어지면 행렬 $M = [\vartheta_0 - \vartheta_n ; \cdots ; \vartheta_{n-1} - \vartheta_n]$에 PCA를 적용한 다음 가장 설명력이 높은 두 가지 방향을 선택할 수 있습니다. 최적화 궤적(파란색 점) 및 손실 서페이스는 그림 9에 나와 있습니다. 학습률이 감소한 시기는 빨간색 점으로 표시되어 있습니다. 각 축에서 해당 PCA 방향에 의해 캡처된 하강 경로의 변화량을 측정합니다.

훈련 초기 단계에서는 경로가 손실 표면의 윤곽에 수직으로, 즉 비확률적 경사 하강에서 예상할 수 있는 경사 방향을 따라 이동하는 경향이 있습니다. 훈련의 후반 단계에서 여러 플롯에서 확률성이 상당히 뚜렷해집니다. 이는 특히 가중치 감쇠와 작은 배치를 사용하는 플롯(더 많은 경사도



(a) SGD, WD=5e-4 (b) Adam, WD=5e-4





(c) SGD, WD=0 (d) Adam, WD=0

그림 9: 예상 학습 궤적은 VGG-9에 대해 정규화된 PCA 방향을 사용합니다. 각 하위 그림의 왼쪽 플롯은 배치 크기 128을 사용하고 오른쪽 플롯은 배치 크기 8192를 사용합니다.

노이즈, 결정론적 기울기 방향에서 더 급진적으로 벗어남). 가중치 감쇠와 작은 배치를 사용하면 경로가 윤곽선과 거의 평행하게 변하고 스텝 크기가 클 때 솔루션이 '궤도'를 도는 것을 볼 수 있습니다. 스텝 크기가 떨어지면(빨간색 점) 시스템의 유효 노이즈가 감소하고 궤적이 가장 가까운 국소 최소값으로 떨어지면서 경로가 꼬이는 것을 볼 수 있습니다.

마지막으로, 하강 경로가 매우 저차원적이라는 것을 직접 관찰할 수 있습니다. 하강 경로의 변화 중 40%에서 90%가 2차원 공간에 불과합니다. 그림 9의 최적화 궤적은 근처 인력 방향으로의 움직임에 의해 지배되는 것처럼 보입니다. 이러한 낮은 차원은 섹션 6의 관찰 결과와 일치하는데, 여기서 우리는 무질서하지 않은 풍경이 넓고 거의 볼록한 최소화기에 의해 지배되는 것을 관찰했습니다.

8 결론

네트워크 아키텍처, 최적화 도구 선택, 배치 크기 등 신경망 실무자가 직면한 다양한 선택의 결과에 대한 인사이트를 제공하는 시각화 기법을 소개했습니다. 신경망은 복잡한 가정을 바탕으로 한 이론적 결과와 일화적 지식에 힘입어 최근 몇 년간 비약적으로 발전해 왔습니다. 계속 발전하기 위해서는 신경망의 구조에 대한 보다 일반적인 이해가 필요합니다. 효과적인 시각화가 이론의 지속적인 발전과 결합되면 더 빠른 학습, 더 단순한 모델, 더 나은 일반화로 이어질 수 있을 것으로 기대합니다.

감사

리, 쉬, 골드스타인은 미 해군 연구실(N00014-17-1-2078), DARPA 평생 학습 기계(FA8650-18-2-7833), DARPA YFA 프로그램(D18AP00055), 슬론 재단(Sloan Foundation)의 지원으로 연구를 수행했습니다. 테일러는 ONR(N0001418WX01582) 및 국방부 HPC 현대화 프로그램의 지원을 받았습니다. Studer는 자일링스(Xilinx, Inc.)와 미국 국립과학재단(NSF)의 보조금 ECCS-1408006, CCF-1535897, CCF-1652065, CNS-1717559 및 ECCS-1824379의 일부 지원을 받았습니다.

참조

[1] 데이비드 발두치, 마커스 프랜, 레녹스 리어리, JP 루이스, 커트 완-두오 마, 브라이언 맥윌리엄스. 깨진 그라데이션 문제: 리셋이 해답이라면 문제는 무엇일까요? *ICML*, 2017.

[2] 아브림 블룸과 로널드 L 리베스트. 3노드 신경망 훈련은 np-완결성입니다. 1989년 *NIPS에서.*

[3] 프라틱 쇼다리, 안나 초로만스카, 스테파노 소아토, 얀 르쿤. 엔트로피-sgd: 넓은 계곡으로의 편향된 경사 하강. *ICLR*, 2017.

[4] 안나 초로만스카, 미카엘 헤나프, 마이클 마티유, 제라르 벤 아루스, 얀 르쿤. 다층 네트워크의 손실 표면. In *AISTATS*, 2015.

[5] 얀 도핀, 라즈반 파스카누, 카글라 굴체레, 조경현, 수리야 강굴리, 요슈아 벤지오. 고차원 비볼록 최적화에서 새들 포인트 문제 식별 및 공격. In *NIPS*, 2014.

[6] 소함 드, 아베이 야다브, 데이비드 제이콥스, 톰 골드스타인. 적응형 배치를 통한 자동화된 추론. *AISTATS*, 2017.

[7] 로랑 딘, 라즈반 파스카누, 사미 벤지오, 요슈아 벤지오. 샤프 최소값은 딥넷에 일반화할 수 있습니다. *ICML*, 2017.

[8] 긴타레 카롤리나 디주가이트와 다니엘 M 로이. 훈련 데이터보다 더 많은 매개 변수가 있는 심층(확률론적) 신경망에 대한 비공백 일반화 경계 계산. *UAI*, 2017.

[9] C 다니엘 프리먼과 조안 브루나. 반 정류 네트워크 최적화의 토폴로지 및 기하학. In *ICLR*, 2017.

[10] 마커스 갤러거와 톰 다운스. 주성분 분석을 이용한 다층 퍼셉트론 네트워크의 학습 시각화. *IEEE 시스템, 인간 및 사이버네틱스 트랜잭션, 파트 B(사이버네틱스)*, 33(1):28-34, 2003.

[11] 자비에 글로롯과 요슈아 벤지오. 심층 피드포워드 신경망 훈련의 어려움에 대한 이해. In *AISTATS*, 2010.

[12] 톰 골드스타인과 크리스토프 스터더. Phasemax: 기저 추적을 통한 볼록 위상 검색. *arXiv 프리프린트 arXiv:1610.07531*, 2016.

[13] 이안 J 굿펠로우, 오리올 빈얄스, 앤드류 M 색스. 신경망 최적화 문제의 질적 특성화. In *ICLR*, 2015.

[14] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia 및 Kaiming He. 정확한 대규모 미니배치 sgd: 1시간 만에 이미지넷 훈련. *arXiv 사전 인쇄물 arXiv:1706.02677*, 2017.

[15] 벤자민 D 해펠레와 르네 비달. 신경망 훈련의 글로벌 최적성. In *CVPR*, 2017.

[16] 모리츠 하르트와 텐유 마. 딥러닝에서 신원 확인의 중요성. *ICLR*, 2017.

[17] 허 카이밍, 장 샹위, 렌 샤오칭, 지안 선. 이미지 인식을 위한 심층 잔여 학습. In *CVPR*, 2016.

[18] 셉 호크라이터와 위르겐 슈미트후버. 플랫 미니멈. *신경 계산*, 9(1):1-42, 1997.

[19] 엘라드 호퍼, 이타이 후바라, 다니엘 수드리. 더 오래 훈련하고 더 잘 일반화하기: 신경망의 대규모 배치 훈련에서 일반화 격차 줄이기. *NIPS*, 2017.

[20] 가오 황, 좡 리우, 킬리안 큐 와인버거, 로렌스 반 데어 마텐. 조밀하게 연결된 컨볼루션 네트워크. *CVPR*, 2017.

[21] 임지웅, 마이클 타오, 크리스틴 브랜슨. 심층 네트워크 손실 표면의 경험적 분석. *arXiv 사전 인쇄물 arXiv:1612.04010*, 2016.

[22] 세르게이 이오페와 크리스티안 세게디. 일괄 정규화: 내부 공변량 이동을 줄임으로써 딥 네트워크 훈련 가속화. In *ICML*, 2015.

[23] 카와구치 켄지, 레슬리 팩 카엘블링, 요슈아 벤지오. 딥 러닝의 일반화. *arXiv 사전 인쇄물 arXiv:1710.05468*, 2017.

[24] 니티쉬 쉬리쉬 케스카, 데바사 무디게레, 호르헤 노세달, 미하일 스멜리얀스키, 핑탁 피터 탕. 딥러닝을 위한 대규모 배치 학습에 대해: 일반화 갭과 날카로운 최소값. In *ICLR*, 2017.

[25] 앤더스 크로흐와 존 A 헤르츠. 단순한 체중 감소는 일반화를 향상시킬 수 있습니다. In *NIPS*, 1992.

[26] 유안지 리와 양 위안. 릴루 활성화가 있는 2계층 신경망의 융합 분석. *arXiv 사전 인쇄물 arXiv:1705.09886*, 2017.

[27] 첸리 리아오와 토마소 포지오. 딥 러닝 이론 II: 딥 러닝의 경험적 위험의 풍경. *arXiv preprint arXiv:1703.09833*, 2017.

[28] 재커리 C 립튼. 중량 공간에서의 모험. *ICLR 워크숍*, 2016.

[29] 엘리아나 로치. pca로 딥 네트워크 훈련 궤적 시각화하기. *딥 러닝을 위한 시각화에 관한 ICML 워크샵*, 2016.

[30] 베남 네이샤부르, 스리나드 보자나팔리, 데이비드 맥알레스터, 나티 스레브로. 딥러닝의 일반화 탐구. *NIPS*, 2017.

[31] 퀴 응우옌과 마티아스 하인. 깊고 넓은 신경망의 손실 표면. In *ICML*, 2017.

[32] 이타이 사프란과 오하드 샤미르. 과도하게 지정된 신경망에서 초기 분지의 품질에 대해. In *ICML*, 2016.

[33] 카렌 시모니안과 앤드류 지서먼. 대규모 이미지 인식을 위한 매우 심층적인 컨볼루션 네트워크. *ICLR*, 2015.

[34] 레슬리 N 스미스 및 니콜라이 토핀. 주기적 학습률로 손실 함수 토폴로지 탐색. *arXiv 사전 인쇄물 arXiv:1702.04283*, 2017.

[35] 마흐디 솔타놀코타비, 아델 자바마드, 제이슨 디 리. 과도하게 매개변수화된 얕은 신경망의 최적화 환경에 대한 이론적 통찰력. *arXiv 사전 인쇄물 arXiv:1707.04926*, 2017.

[36] 다니엘 수드리와 엘라드 호퍼. 다층 신경망에서 기하급수적으로 사라지는 서브-최적 국부 최소값. *arXiv preprint arXiv:1702.05777*, 2017.

[37] 그르제고르 스비르슈치, 보이치에흐 마리안 차르네츠키, 라즈반 파스카누. 딥 네트워크 훈련의 국부 최소값. *arXiv 사전 인쇄물 arXiv:1611.06310*, 2016.

[38] 위안동 티안. 2계층 릴루 네트워크에 대한 인구 기울기 분석 공식과 수렴 및 임계점 분석에 대한 응용. In *ICML*, 2017.

[39] 보 시에, 잉위 량, 레 송. 진정한 목표 함수를 학습하는 다양한 신경망. In *AISTATS*, 2017.

[40] 윤철희, 수브릿 스라, 알리 자드바바이. 심층 신경망의 글로벌 최적 조건. In *ICLR*, 2017.

[41] 세르게이 자고루이코와 니코스 코모다키스. 광범위한 잔여 네트워크. *BMVC*, 2016.

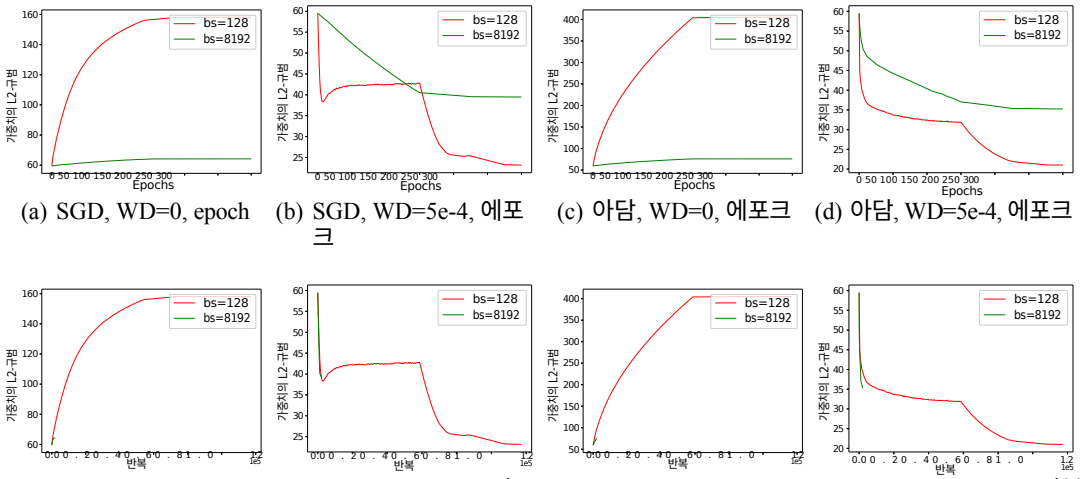
[42] 치위안 장, 새미 벤지오, 모리츠 하르트, 벤자민 레흐트, 오리올 빈얄스. 딥러닝을 이해하려면 일반화에 대해 다시 생각해야 합니다. *ICLR*, 2017.

신경망의 손실 환경 시각화하기

A 손실 표면 비교

A.1 훈련 중 웨이트 규범의 변화

그림 10은 훈련 중 가중치 표준의 변화를 에포크와 반복 측면에서 보여줍니다.



(a) SGD, WD=0, epoch (b) SGD, WD=5e-4, 에포크 (c) 아담, WD=0, 에포크 (d) 아담, WD=5e-4, 에포크

(e) SGD, WD=0, iter (f) SGD, WD=5e-4, 반복 (g) Adam, WD=0, iter (h) Adam, WD=5e-4, 반복

그림 10: VGG-9 훈련 중 웨이트 규범의 변화. 가중치 감쇠를 비활성화하면 훈련 중에 가중치 규범은 제약 없이 꾸준히 증가합니다. 가중치 감쇠가 0이 아닌 경우, 가중치 규범은 초반에 급격히 감소하고 학습률이 감쇠될 때까지 안정화됩니다. 배치 크기에 따라 고정된 에포크 수를 사용하기 때문에 대규모 배치와 소규모 배치 훈련 간의 가중치 규범 변화의 차이는 주로 소규모 배치를 사용할 때 업데이트 횟수가 많기 때문에 발생합니다. 두 번째 행에서 볼 수 있듯이, 반복 횟수 측면에서 소규모 배치 훈련과 대규모 배치 훈련 모두 가중치 규범의 변화 속도가 동일합니다.

A.2 정규화 방법 비교

여기서는 주어진 임의의 법선 방향 d에 대한 여러 정규화 방법을 비교합니다. θi는 레이어 i의 가중치를 나타내고 $\theta i_{,j}$는 레이어 i의 j번째 필터를 나타냅니다.

- **정규화 없음** 이 경우 방향 d가 처리 없이 가중치에 직접 추가됩니다.
- **필터 정규화** 각 필터의 방향이 θ의 해당 필터와 동일한 규범을 갖도록 방향 d를 정규화합니다,

$$d_{i,j} \leftarrow \frac{d_{i,j}}{\|d_{i,j}\|} \|\theta_{i,j}\|.$$

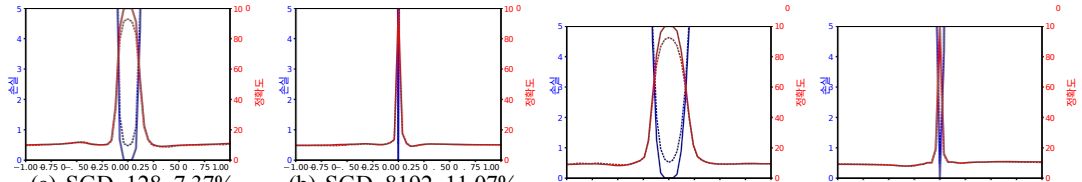
이 문서에서 옹호하는 접근 방식이며 손실 표면을 그리는 데 광범위하게 사용됩니다.

- **레이어 정규화** 각 레이어의 방향이 θ의 해당 레이어와 동일한 규범을 갖도록 레이어 수준에서 방향

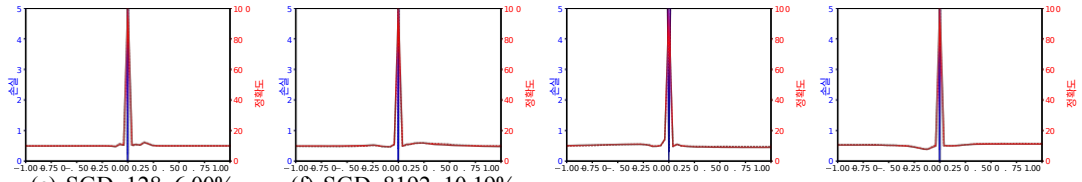
d를 정규화합니다,

$$d_i \leftarrow \frac{d_i}{\|d_i\|}\|\theta_i\|.$$

그림 11은 정규화하지 않은 1D 플롯을 보여줍니다. 정규화되지 않은 플롯의 한 가지 문제점은 x축 범위를 신중하게 선택해야 한다는 것입니다. 그림 12는 $[-0.2, 0.2]$를 x축 범위로 하여 확대한 플롯을 보여줍니다. 정규화를 하지 않으면 플롯에서 평탄도와 일반화 오류 사이의 일관성을 보여주지 못합니다. 여기에서는 필터 정규화와 레이어 정규화를 비교합니다. 필터 정규화가 레이어 정규화보다 더 정확하다는 것을 알 수 있습니다. 레이어 정규화에 실패한 한 가지 사례는 그림 13에 나와 있으며, 그림 13(g)는 그림 13(c)보다 평탄하지만 일반화 오차가 더 심합니다.

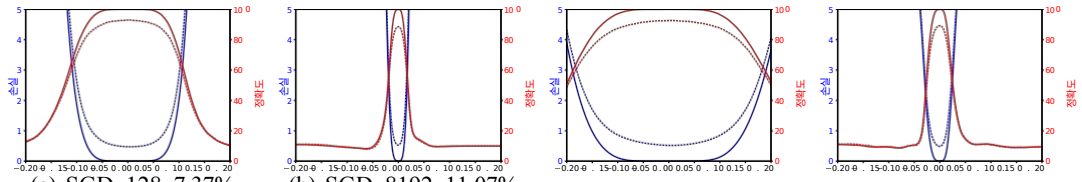


(a) SGD, 128, 7.37% (b) SGD, 8192, 11.07% (c) 아담, 128, 7.44% (d) 아담, 8192, 10.91%

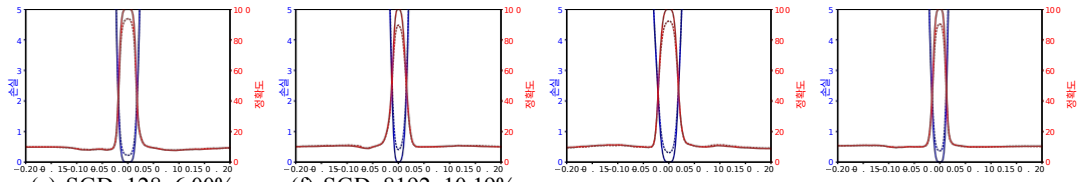


(e) SGD, 128, 6.00% (f) SGD, 8192, 10.19% (g) 아담, 128, 7.80% (h) 아담, 8192, 9.52%

그림 11: 정규화하지 않은 VGG-9의 1D 손실 플롯. 첫 번째 행은 가중치 감쇠가 없고 두 번째 행은 가중치 감쇠 0.0005를 사용합니다.

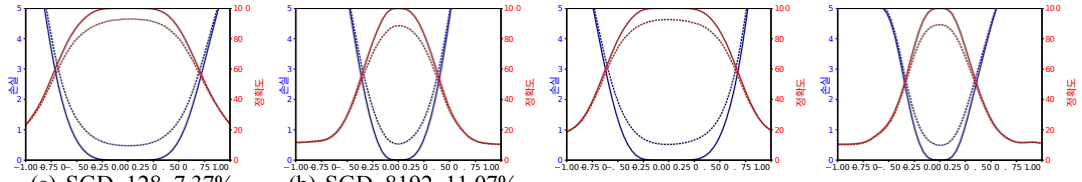


(a) SGD, 128, 7.37% (b) SGD, 8192, 11.07% (c) 아담, 128, 7.44% (d) 아담, 8192, 10.91%

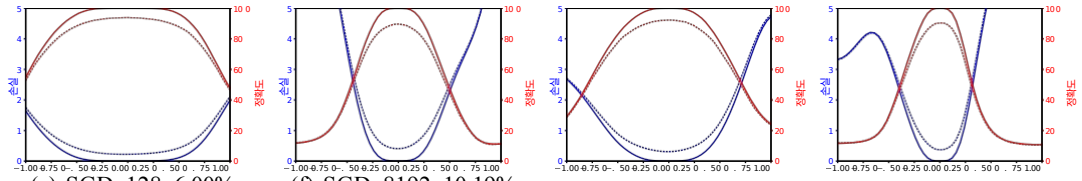


(e) SGD, 128, 6.00% (f) SGD, 8192, 10.19% (g) 아담, 128, 7.80% (h) 아담, 8192, 9.52%

그림 12: 그림 11 확대. *x축의* 범위는 [-1.0, 1.0] 대신 [-0.2, 0.2]입니다. 첫 번째 행은 가중치 감쇠가 없고 두 번째 행은 가중치 감쇠 0.0005를 사용합니다. (a, e) 및 (c, g) 쌍은 최소값의 선명도가 테스트 오류와 잘 상관관계가 없음을 보여줍니다.



(a) SGD, 128, 7.37% (b) SGD, 8192, 11.07% (c) 아담, 128, 7.44% (d) 아담, 8192, 10.91%



(e) SGD, 128, 6.00% (f) SGD, 8192, 10.19% (g) 아담, 128, 7.80% (h) 아담, 8192, 9.52%

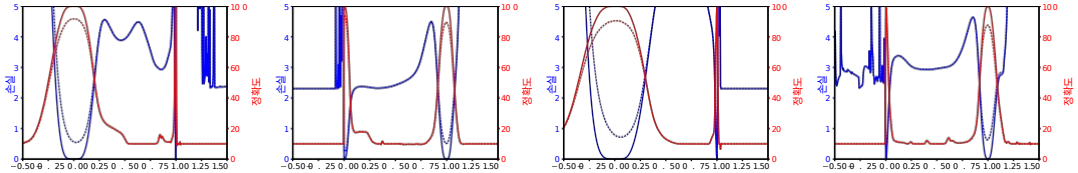
그림 13: 레이어 정규화를 사용한 VGG-9의 1D 손실 플롯. 첫 번째 행은 가중치 감쇠가 없고 두 번째 행은 가중치 감쇠 5e-4를 사용합니다.

A.3 ResNet-56의 소규모 배치와 대규모 배치 비교

섹션 5에서 관찰한 것과 유사하게, 그림 14에 표시된 것처럼 "날카로운 대 평평한 딜레마"는 ResNet-56에도 적용됩니다. 각 솔루션에 대한 일반화 오류는 표 1에 나와 있습니다. 필터 정규화된 방향을 사용한 1D 및 2D 시각화는 그림 15에 나와 있습니다.

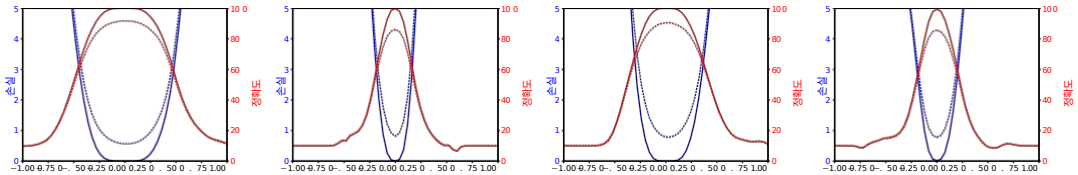
표 1: 최적화 프로그램, 배치 크기, 가중치 감쇠가 다른 ResNet-56에 대한 테스트 오류.

	SGD		Adam	
	bs=128	bs=4096	bs=128	bs=4096
WD =	08.26	13.93	9.55	14.30
WD = 5e-4	**5.89**	**10.59**	**7.67**	**12.36**

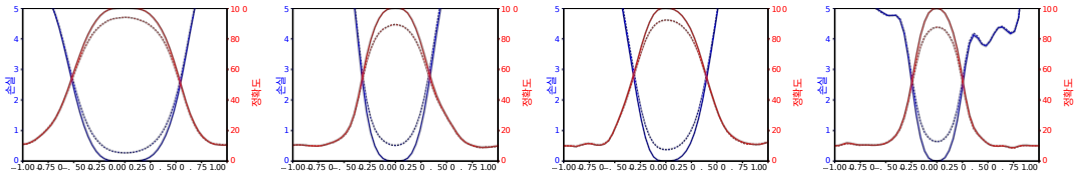


(a) SGD, WD=0 (b) SGD, WD=5e-4 (c) Adam, WD=0 (d) Adam, WD=5e-4

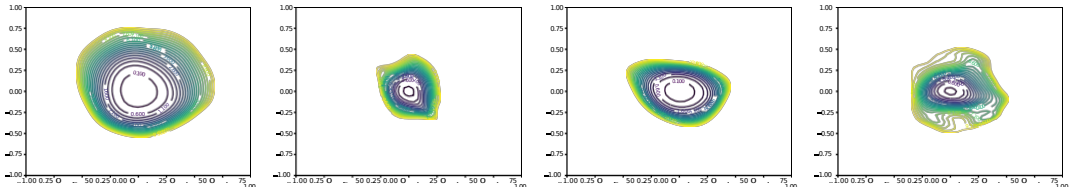
그림 14: ResNet-56에 대해 소규모 배치 및 대규모 배치 방법으로 얻은 솔루션의 1D 선형 보간. 파란색 선은 손실 값이고 빨간색 선은 오차입니다.



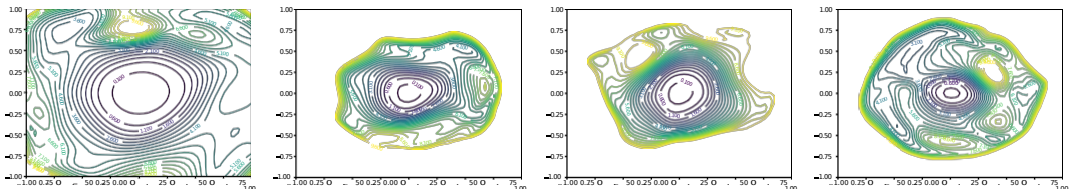
(a) SGD, 128, 8.26% (b) SGD, 4096, 13.93% (c) 아담, 128, 9.55% (d) 아담, 4096, 14.30%



(e) SGD, 128, 5.89% (f) SGD, 4096, 10.59% (g) 아담, 128, 7.67% (h) 아담, 4096, 12.36%



(i) SGD, 128, 8.26% (j) SGD, 4096, 13.93% (k) 아담, 128, 9.55% (l) 아담, 4096, 14.30%

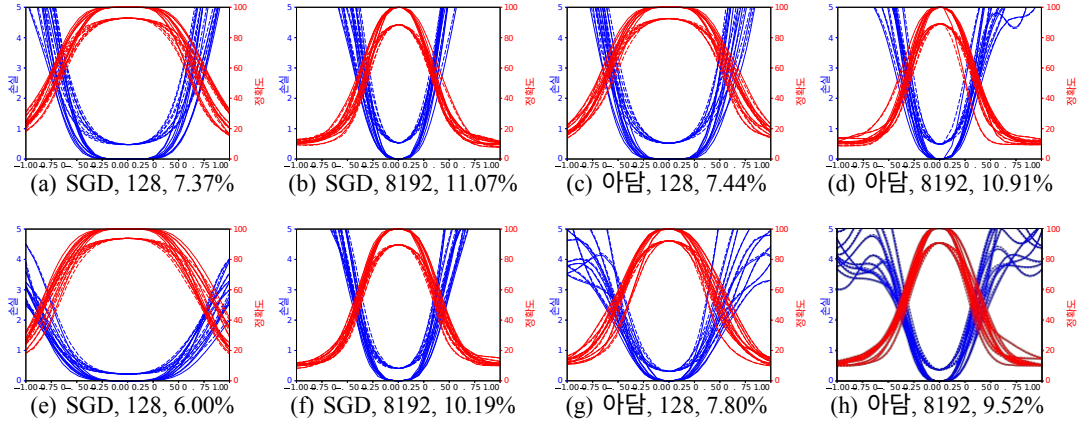


(m) SGD, 128, 5.89% (n) SGD, 4096, 10.59% (o) 아담, 128, 7.67% (p) 아담, 4096, 12.36%

그림 15: 다양한 옵티마이저, 배치 크기, 가중치 감쇠로 훈련된 ResNet-56의 1D 및 2D 시각화. 첫 번째와 세 번째 행은 제로 가중치 감쇠를 사용하고 두 번째와 네 번째 행은 5e-4 가중치 감쇠를 사용합니다.

A.4 손실 표면 시각화의 반복성

무작위 방향이 다르면 극적으로 다른 플롯이 생성되나요? 10개의 임의 필터 정규화 방향을 사용하여 VGG-9의 1D 손실 표면을 플롯합니다. 그림 16에서 볼 수 있듯이 플롯의 모양이 매우 비슷합니다. 또한 일반화 오차가 더 심한 ResNet-56-noshort에 대해서도 2D 손실 표면 플롯을 여러 번 반복합니다. 그림 17에서 볼 수 있듯이 플롯마다 손실 표면의 변화가 뚜렷하게 나타나지만, 질적인 선택적 동작은 플롯 전반에 걸쳐 상당히 일관적입니다.



(a) SGD, 128, 7.37% (b) SGD, 8192, 11.07% (c) 아담, 128, 7.44% (d) 아담, 8192, 10.91%

(e) SGD, 128, 6.00% (f) SGD, 8192, 10.19% (g) 아담, 128, 7.80% (h) 아담, 8192, 9.52%

그림 16: 필터 정규화를 사용한 VGG-9에 대한 표면 플롯의 반복성. 10개의 서로 다른 무작위 필터 정규화 방향을 사용하여 얻은 최소값의 모양입니다.

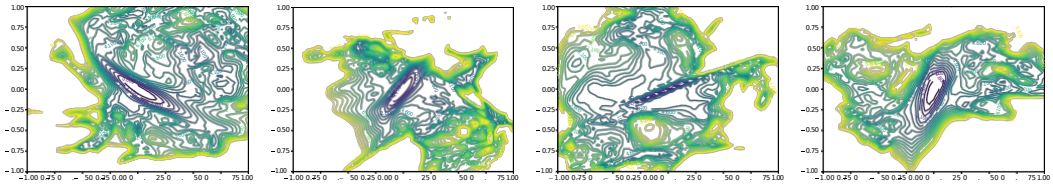


그림 17: ResNet-56-noshort에 대한 2D 표면 플롯의 반복성. 모델은 배치 크기 128, 초기 학습률 0.1, 가중치 감쇠 5e-4로 훈련되었습니다. 최종 훈련 손실은 0.192, 훈련 오류는 6.49, 테스트 오류는 13.31입니다.

A.5 구현 세부 정보

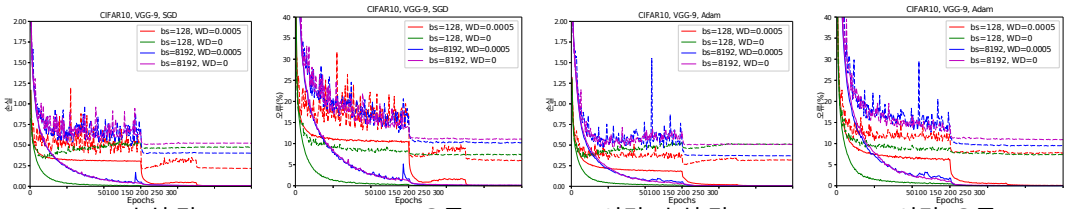
수치 생성을 위한 컴퓨팅 리소스 PyTorch 코드는 다중 GPU 워크스테이션은 물론 `mpi4py`를 사용하는 수백 개의 GPU를 갖춘 HPC에서도 실행할 수 있습니다. 계산 시간은 훈련 세트에 대한 모델의 추론 속도, 플롯의 해상도, GPU 수에 따라 달라집니다. 그림 3의 1D 플롯의 해상도는 *401×401입니다*. 그림 3과 그림 5의 2D 윤곽에 사용된 기본 해상도는 51×51입니다. 더 자세한 내용을 표시하기 위해 그림 1에 사용된 ResNet-56-노쇼트에는 더 높은 해상도(251 × 251)를 사용합니다. 참고로 (상대적으로 낮은) 해상도인 51 × 51의 ResNet-56 2D 등고선 플롯은 4개의 GPU(Titan X Pascal 또는 1080 Ti)가 장착된 워크스테이션에서 약 1시간이 소요됩니다.

배치 정규화 매개변수 1D 선형 보간 방법에서는 "실행 평균" 및 "실행 분산"을 포함한 배치 정규화(BN) 매개변수를 θ의 일부로 고려해야 합니다. 이러한 매개변수를 고려하지 않으면 두 최소화기의 정확한 손실값을 재현할 수 없습니다. 필터 정규화된 시각화에서 임의의 방향은 배치 표준 매개 변수를 제외한 모든 가중치를 교란합니다. 필터 정규화 프로세스는 가중치 스케일링의 효과를 제거하므로 배치 정규화를 무시할 수 있습니다.

아담 VGG-9의 **아키텍처와** 파라미터는 VGG-16의 크롭 버전으로, VGG-16의 첫 7개의 컨버전 레이어와 2개의 FC 레이어를 유지합니다. 각 컨버 레이어와 첫 번째 FC 레이어 뒤에 BN 레이어가 추가됩니다. VGG-9가 CIFAR-10의 VGG-16에 비해 더 나은 성능을 가진 효율적인 네트워크라는 것을 알 수 있습니다. Adam에서 β_1, β_2, g의 기본값은 SGD에서 사용한 것과 동일한 학습 속도 스케줄을 사용합니다.

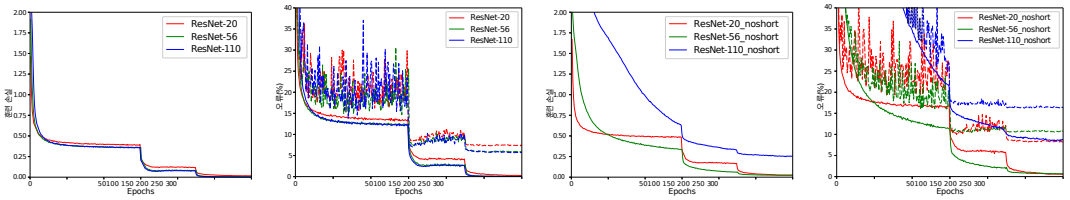
A.6 VGG-9 및 ResNets용 트레이닝 곡선

섹션 5에서 사용된 VGG-9 훈련의 손실 곡선은 그림 18에 나와 있습니다. 그림 19는 섹션 6에서 사용된 아키텍처의 손실 곡선과 오류 곡선을 보여주며, 표 2는 최종 오류 및 손실 값을 보여줍니다. 훈련의 기본 설정은 300개의 에포크에 대해 네스테로프 모멘텀, 배치 크기 128, 0.0005 가중치 감쇠를 가진 SGD를 사용하는 것입니다. 기본 학습 속도는 0.1로 초기화되었으며 150, 225, 275 에포크에서 10배씩 감소했습니다.



(a) SGD, 손실 값 (b) SGD, 오류 (c) 아담, 손실 값 (d) 아담, 오류

그림 18: 다양한 최적화 방법을 사용한 VGG-9의 훈련 손실/오류 곡선. 점선은 테스트용, 실선은 훈련용입니다.



(a) ResNet-CIFAR (b) ResNet-CIFAR (c) ResNet-CIFAR-noshort (d) ResNet-CIFAR-noshort

그림 19: 다양한 아키텍처에 대한 컨버전스 곡선.

표 2: 표 2: CIFAR-10에서 학습된 다양한 아키텍처의 손실 값 및 오류.

	초기화 LR	훈련 손실	교육 오류	테스트 오류
ResNet-20	0.1	0.017	0.286	7.37
ResNet-20-noshort	0.1	0.025	0.560	8.18
ResNet-56	0.1	0.004	0.052	5.89
ResNet-56-noshort	0.1	0.192	6.494	13.31
ResNet-56-noshort	0.01	0.024	0.704	10.83
ResNet-110	0.1	0.002	0.042	5.79
ResNet-110-noshort	0.01	0.258	8.732	16.44