卒業論文

深層ニューラルネットワークにおける学習ダイナミクスの初等的解析

XXXXXXXX　宇都宮 幸大

指導教員　XXXXXXXX

2025 年 2 月

XXXXXXXX

概要

膨大な数の学習可能パラメータを持つ深層ニューラルネットワークは，幅広いタスクにおいて驚異的な性能を発揮している．こうした背景の下，中間層のニューロン数が極めて大きいニューラルネットワークの情報処理様式に関する研究が盛んに行われている．特に，ニューラルネットワークの線形モデルとしての振る舞いを記述する *Neural Tangent Kernel* (NTK) の理論 [1, 2] や，外界の情報の内在的な特徴表現を学習を通して獲得するというニューラルネットワーク特有の側面に焦点を当てた *Maximal Update Parametrization* (μP) [3] が代表的である．これらの研究は，すべての中間層のニューロン数が同じオーダーで無限大に漸近する場合の学習ダイナミクスについての知見を提供している．では，中間層のニューロン数が層によって異なるオーダーで増加する場合には何が起きるだろうか．本稿ではこの問いについて考える．まず，各層におけるパラメータ更新がニューロン数に依存せずに損失の減少に寄与し，出力のダイナミクスがニューロン数に依存して消失や発散しない安定な学習を実現する条件を調べた結果，各中間表現の更新量はニューロン数に関して同じオーダーである必要があり，中間層における最小のニューロン数 n_{min} がその上限として現れることがわかった．これらの条件は，各中間層におけるパラメータの更新能力がニューロン数に依存しない対称性が高い状況では学習が不安定であり，対称性が n_{min} の方向に崩れた状況で安定することを示唆する [表 1.1]．最後に，上記の知見を踏まえてパラメータの更新量を n_{min} で制約することで，訓練性や汎化性能の向上が期待できることを実験的に確認した．

キーワード： 深層学習，NTK，μP，過剰パラメータ，特徴学習，ボトルネック構造

目次

第 1 章　はじめに 1

1.1　研究背景 2

1.1.1　過剰パラメータ系を調べる必要性 —汎化誤差曲線の観点から— . . . 2

1.1.2　学習レジームの概念 4

1.2　主要な問いと研究結果 8

1.3　記法 11

第 2 章　学習が安定に進む条件の導出 12

2.1　深層ニューラルネットワークの定義と学習の枠組み 12

2.2　初期化時のオーダー評価 14

2.3　勾配降下法によるパラメータの更新 15

2.3.1　σ_ℓ と g_ℓ が満たすべき条件 15

2.3.2　勾配とモデル出力の発散を防ぐための基準 18

2.4　最小のニューロン数に基づくパラメータ設定 20

2.5　学習率を層ごとに調整するパラメータ設定との関係 21

第 3 章　計算機実験による理論検証 23

3.1　Spectral Parametrization による最適化の性質 23

3.2　Dynamic Parametrization による効果的な最適化 25

3.3　性能への影響 27

第 4 章　おわりに 30

謝辞 31

参考文献 32

第 1 章

はじめに

深層ニューラルネットワーク[*1]は，学習可能なパラメータ数が膨大な過剰パラメータ系として，画像認識や自然言語処理をはじめとする多様なタスクで驚異的な性能を発揮している．こうした背景の下，中間層のニューロン数が無限大の極限におけるニューラルネットワークの学習ダイナミクスを決定論的に記述する *Neural Tangent Kernel* (NTK) [1,2] や，パラメータの最適化時の更新量を保証する *Maximal Update Parametrization* (μP) [3] などの理論が発展してきた．一方，これらの研究はすべての中間層のニューロン数が同じオーダーで増加する状況を想定しており，現実のモデルで頻繁に用いられるボトルネック構造（特定の中間層のニューロン数が相対的に著しく少ないアーキテクチャ）における学習能力については未解明の点が多い．

本研究では，中間層のニューロン数が層によって異なるオーダーで増加する状況に焦点を当て，ボトルネック構造が強調されるときに生じる情報処理の一端をニューロン数に関するオーダー評価の枠組みで捉えた．特に，各層におけるパラメータ更新がニューロン数に依存せずに損失の減少に寄与し，出力のダイナミクスがニューロン数に依存して消失したり発散したりしないような学習を保証するパラメータ設定を提案する．このような学習は安定であるという．まず，初等的な理論解析に基づき，安定な学習を実現するためには各層の更新量のオーダーが中間層における最小のニューロン数 $n_{\min}$ によって制約される必要性があることを確認した．この結果は，ボトルネック構造の下では非対称なパラメータ設定で安定することを示唆する．さらに，標準的なデータセット CIFAR-10 [4] を対象とした画像分類タスクにおいて，μP の拡張である Spectral Parametrization [5] と比較し，提案手法が訓練性と汎化性能の両面で優位性があることを実験的に検証した．

本稿の構成は以下の通りである．第 1 章では，過剰パラメータ系の特性と，特徴学習と呼ばれる概念を整理した後，研究課題を明確化する．第 2 章では，安定な学習を実現するパラメータ設定の導出を行う．第 3 章では，提案手法の有効性を計算機実験により検証し，第 4 章で総括と今後の展望を述べる．

[*1] 深層ニューラルネットワークは，信号の affine 変換と非線形変換を行う層を多段に積み重ねることで複雑な関数を構成するモデルである．具体的な定式化や説明は 2.1 節で行っているため，必要に応じて参照されたい．

1.1 研究背景

本稿では，中間層のニューロン数が極めて大きい深層ニューラルネットワークの学習ダイナミクスについて考える．このようなネットワークは可変なパラメータを膨大に持ち，それらの値によって入出力関係が決まる．本題の前に，このようなネットワークを研究対象とする背景や意義について触れておきたい．パラメータ数が学習に用いるデータ数をはるかに超える領域でなされる情報処理には興味深い性質がある．これについて，学習システムとしての性能の観点から 1.1.1 項で紹介した後，中間層のニューロン数が無限大の極限を考えることによって学習ダイナミクスの普遍則を記述する近年の試みを 1.1.2 項で概観する．

1.2 節では，上記の背景を踏まえた自然な問いを提示するとともに，その問いに対する本研究の結果をまとめる．

1.1.1 過剰パラメータ系を調べる必要性 —汎化誤差曲線の観点から—

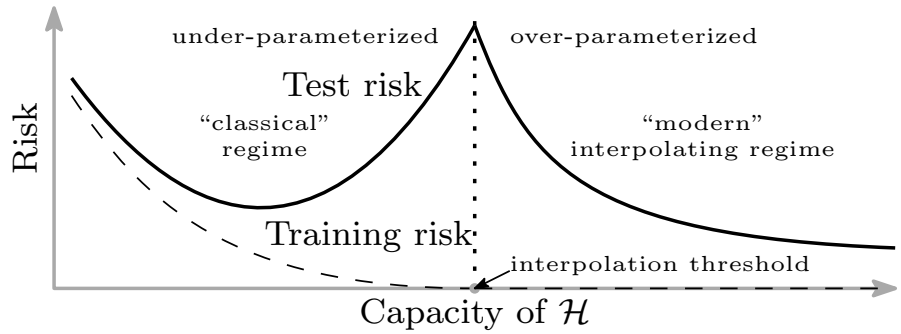

図 1.1: 汎化誤差曲線の二重降下．実線は訓練誤差，破線は汎化誤差に相当する．横軸はモデル f の複雑度（主に学習可能パラメータ数 p で測られる）を表す．図は Belkin *et al.* (2019) より引用 [14].

深層ニューラルネットワークは，与えられた入力 $\boldsymbol{x}$ に応じて何かしらの出力を返す関数 f として捉えられる．f は p 個の可変なパラメータ $\boldsymbol{\theta} \coloneqq (\theta_1, \theta_2, \ldots, \theta_p)^\top$ を持っており，これらの値によって f の入出力関係が決まるとしよう．この意味を込めて f の出力を $f(\boldsymbol{x}; \boldsymbol{\theta})$ とも書く．ここで，外界の情報として独立同分布に従う入出力関係 $\mathcal{D} \coloneqq \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{|\mathcal{D}|}$ を知ったとき，入力 $\boldsymbol{x}_i$ に対して y_i に近い値を返すようなパラメータ $\boldsymbol{\theta}^*$ を何かしらの学習アルゴリズム[*2]を用いて求め，$\mathcal{D}$ の入出力関係を再現する関数 f^* を構成することが考えられる．これが，$\mathcal{D}$ を **訓練データ**，$\boldsymbol{\theta}$ を **学習可能パラメータ** とする教師あり学習の枠組みである．近年，機械学習モデルにおけるパラメータ数 p の増加が著しい．例えば，深層ニューラルネットワークを用いた言語モデルである GPT (Generative Pretrained Transformer) シリーズでは，第 1 世代の GPT のパラメータ数は約 1.17 億，GPT-2 は約 15.4 億，GPT-3 は約 1750 億に上るとい

[*2] $(\boldsymbol{x}_i, y_i)$ に対する f の当てはまりの誤差を測る損失関数の値を最小化するように，勾配降下法などの 1 次最適化手法を用いて $\boldsymbol{\theta}$ をランダムな初期値から少しずつ調整していくことが多い．具体的な定式化は 2.1 節で行う．

う [6–9].

パラメータ数 p を増やして関数 f の内部構造を細かく調整できるようにすれば，複雑な入出力関係を実現することが期待される．実際，十分なパラメータ数 p を持つ深層ニューラルネットワークは，可積分性などの適切な条件の下で任意の関数を任意の精度で近似できることが知られており[*3]，有限個の訓練データ $\mathcal{D}$ を内挿（誤差ゼロで再現）する関数 f^* を構成する能力がある．一方，実用的には，$\mathcal{D}$ を用いた学習を通じてデータの内在的な規則性を捉え，$\mathcal{D}$ と同じ分布に従う未知のデータ $\mathcal{D}'$ に対しても f^* が良く当てはまることが望ましい．これが**汎化性**の問題である．特に，$\mathcal{D}$ に大きな観測ノイズが含まれている場合，$\mathcal{D}$ に対する過剰な適合は汎化の妨げになり得ることが容易に想像できるだろう．このように，モデルが複雑であるほど，学習に用いる訓練データに依存して f^* の入出力関係が大きく変動し，汎化性能が悪化する傾向があるため，パラメータ数 p を必要以上に増やすことは避けるべきという考え方が主流であった [12][*4].

しかし，近年の機械学習モデルはこの常識に反し，パラメータ数 p が巨大であるにもかかわらず高い汎化性能を示し得ることが知られている [13]．このギャップに関して，Belkin *et al.* (2019) は**二重降下** (*double descent*) という概念を提唱した [14]．これは，f の複雑度（主にパラメータ数 p で測られる）を横軸，汎化誤差を縦軸に取った汎化誤差曲線が，図 1.1 のような挙動を示すというものである．まず，パラメータ数 p が訓練データ数 $|\mathcal{D}|$ よりも小さい過小パラメータ領域では，従来の学習理論に合致する U 字型の曲線を描く．すなわち，関数近似能力が非常に乏しい状態から p を増やしていくと，はじめは汎化誤差が訓練誤差とともに減少していくが，あるところで有限個の訓練データ $\mathcal{D}$ に対する過剰な適合によって汎化誤差が上昇に転じる．p をさらに増やすと汎化性能が悪化し続けるように思われるかもしれないが，驚くべきことに，近年の機械学習モデルはそうではないらしい．訓練データを内挿できる臨界点を超えた過剰パラメータ領域 $p \gg |\mathcal{D}|$ において，汎化誤差が再び減少に転じるのである[*5]．この汎化誤差曲線の二重降下（あるいはこれを繰り返す多重降下）は，多くの深層ニューラルネットワークモデルで実験的に検証され[*6]，ランダム行列理論や自由確率論などに基づく理論的解析も進んでいる [15–18].

深層ニューラルネットワークは，従来の学習理論における前提とは異なる領域 —過剰パラメータ領域— で成功を収めている．“モデルは大きければ大きいほど良い”という言説さえ正

[*3] あくまでも理想的なパラメータ $\boldsymbol{\theta}^*$ の存在性のみを主張している点に注意．中間層を 1 つだけ持つ 2 層ニューラルネットワークの関数近似能力に関する文献として，Cybenko (1989) による連続関数の一様近似定理や，Sonoda and Murata (2017) による ReLU 関数の下での普遍近似性の証明などが代表的である [10,11].

[*4] Geman *et al.* (1992) は“バイアス・バリアンスのトレードオフ”を提起した上で，汎化性能の高いモデルを構築するためには，問題の特性に関する事前知識をモデルに組み込んだり，データの表現方法を工夫したりすることが必要不可欠だと結論付けている [12].

[*5] 同様の現象は，単純パーセプトロンなどの非常にシンプルなモデルに限り既に知られていた．文献として，Opper *et al.* (1996, 1997) や Krogh *et al.* (1992) などが挙げられる [19–21].

[*6] 直観的には，過剰パラメータ領域ではモデル出力の推定量の分散が暗黙的に抑えられ，f^* の入出力関係が“全体として”より滑らかになるといえる（観測ノイズ等に対する過剰な適合の影響が非常に局所的になる）．また，過剰パラメータ領域で汎化するメカニズムとして，勾配降下法自体がモデルの複雑さを暗黙的に抑えるように働くというバイアス（例：パラメータ空間での決定境界とデータ点のマージンの最大化，重み行列のランクの最小化，損失形状がより平坦な極小値への収束 etc.）も重要な要素であると考えられている [22].

当化され得る今日，ニューロン数が無限大の極限における学習ダイナミクスが研究されている．これについて，次項で概観しよう．

1.1.2 学習レジームの概念

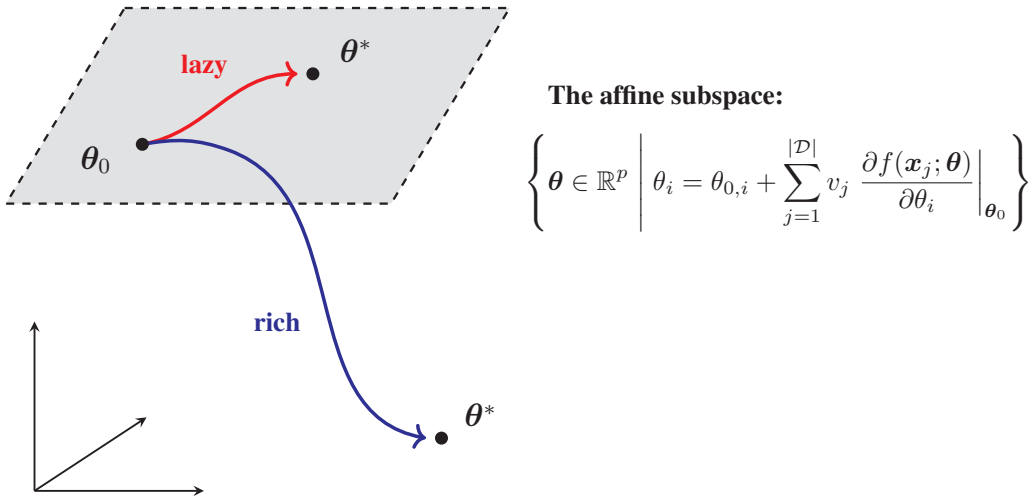

図 1.2: パラメータ空間における lazy および rich な学習ダイナミクスの概念図 ($p = 3$，$|\mathcal{D}| = 2$)．灰色の平面は，パラメータに関する初期化時の勾配で張られる affine 部分空間を表す（$v_i \in \mathbb{R}$）．勾配降下法による学習において，lazy レジームでは訓練誤差を最小化する解をこの部分空間上で得る．損失関数として二乗誤差を用いた場合，その解は初期化時の NTK $k_0(\boldsymbol{z}, \boldsymbol{z}')$ をカーネル関数として用いたときのカーネル法の解に相当する．一方，rich レジームでは学習中にパラメータが初期値から大きく離れ，線形化を脱する．その結果，通常，訓練データやタスクに内在する何かしらの“特徴”をパラメータが獲得すると考えられる．図は Kumar *et al.* (2024) を参考に作成 [23].

深層ニューラルネットワーク[*1]の中間層のニューロン数が無限大の極限における学習ダイナミクスを記述する代表的な理論の 1 つとして，Jacot *et al.* (2018) が導入した *Neural Tangent Kernel* (NTK) の理論がある [1]．深層ニューラルネットワークは，信号の affine 変換と非線形変換を行う層を多段に積み重ねることによって複雑な関数 f を構成する．一般的な定式化は 2.1 節で行うことにし，ここでは簡単のため，ニューロン数が n の中間層を 2 つだけ持ち，affine 変換における平行移動を省略したネットワークを考えよう．学習可能パラメータ $\boldsymbol{\theta}$ は，重みと呼ばれる 3 つの行列 $\boldsymbol{W}^{(1)} \in \mathbb{R}^{n\times d}$，$\boldsymbol{W}^{(2)} \in \mathbb{R}^{n\times n}$，$\boldsymbol{W}^{(3)} \in \mathbb{R}^{d'\times n}$ の要素をまとめてベクトル化したものであるとする（すなわち，$\boldsymbol{\theta}$ は $p = nd + n^2 + d'n$ 次元）．$\boldsymbol{\theta}$ を学習により調整するわけであるが，はじめは標準正規分布 $\mathcal{N}(0,1)$ で各要素を独立に初期化しておこう．

初期化したパラメータを $\boldsymbol{\theta}_0$ で表す．簡単のため，$d'=1$ とし，f の内部構造を以下で定める：

$$f(\boldsymbol{x};\boldsymbol{\theta}) \coloneqq \frac{1}{\sqrt{n}}\boldsymbol{W}^{(3)}\boldsymbol{h}^{(2)}, \tag{1.1}$$

$$\boldsymbol{h}^{(2)} = \psi(\widetilde{\boldsymbol{h}}^{(2)}), \tag{1.2}$$

$$\widetilde{\boldsymbol{h}}^{(2)} = \frac{1}{\sqrt{n}}\boldsymbol{W}^{(2)}\boldsymbol{h}^{(1)}, \tag{1.3}$$

$$\boldsymbol{h}^{(1)} = \psi(\widetilde{\boldsymbol{h}}^{(1)}), \tag{1.4}$$

$$\widetilde{\boldsymbol{h}}^{(1)} = \frac{1}{\sqrt{d}}\boldsymbol{W}^{(1)}\boldsymbol{x}. \tag{1.5}$$

ただし，$\psi(\cdot)$ は非線形な関数であり，引数のベクトルに対してその要素ごとに作用させる．$\{\boldsymbol{h}^{(1)}, \widetilde{\boldsymbol{h}}^{(1)}\}$ と $\{\boldsymbol{h}^{(2)}, \widetilde{\boldsymbol{h}}^{(2)}\}$ が，それぞれ 1 つ目と 2 つ目の中間層における中間表現（ニューロンの発火活動）である．$1/\sqrt{n}$ のように規格化するのは，伝播される信号の大きさがニューロン数 n の増加に伴って発散してしまうのを防ぐためである．上記のように，$\boldsymbol{\theta}$ の各要素を標準正規分布 $\mathcal{N}(0,1)$ で独立に初期化し，$1/\sqrt{n}$ で規格化する設定は，*NTK Parametrization* (NTP) と呼ばれる[*7]．訓練データ $\mathcal{D}$ の入出力関係をこの f で再現するために，損失関数を二乗誤差とする勾配降下法（2.1 節を参照）により $\boldsymbol{\theta}$ を初期値 $\boldsymbol{\theta}_0$ から調整していく．

中間層のニューロン数 n が大きいとき，学習可能パラメータ $\boldsymbol{\theta}$ がなす空間は，高次元で極めて複雑である．また，互いに異なるパラメータ $\boldsymbol{\theta}$ と $\boldsymbol{\theta}'$ が，任意の入力 $\boldsymbol{x}$ に対して

$$f(\boldsymbol{x};\boldsymbol{\theta}) = f(\boldsymbol{x};\boldsymbol{\theta}') \tag{1.6}$$

を満たす点を多数含む．このようなパラメータは識別不能（unidentifiable）であるという[*8]．過剰パラメータ系 $p \gg |\mathcal{D}|$ である深層ニューラルネットワークにおいて，訓練データ $\mathcal{D}$ の入出力関係を近似するパラメータも無数に存在すると考えられる．NTK の理論は，NTP の下で中間層のニューロン数 n を無限大とする極限をとると，訓練データ $\mathcal{D}$ を内挿するパラメータ $\boldsymbol{\theta}^*$ は初期値 $\boldsymbol{\theta}_0$ のすぐ近くにあり，学習ダイナミクスが初期値近傍で収束することを明らかにした．パラメータがランダムに初期化された点 $\boldsymbol{\theta}_0$ からほとんど動かずに学習できるのは，$p \gg |\mathcal{D}|$ の効果である．直観的には，勾配降下法による学習中，個々のパラメータ θ_i の初期値 $\theta_{0,i}$ からの変化 $|\theta_i - \theta_{0,i}|$ は微小であるにもかかわらず $n \to \infty$ 個で互いに干渉し合い，モデル出力としては学習を進める上で十分な変化を生むのである．このとき，パラメータの初期値周りでの線形近似（Taylor 展開による 1 次近似）が保証される [2]：

$$f(\boldsymbol{x};\boldsymbol{\theta}) \approx f(\boldsymbol{x};\boldsymbol{\theta}_0) + \sum_{i=1}^{p} \left.\frac{\partial f(\boldsymbol{x};\boldsymbol{\theta})}{\partial \theta_i}\right|_{\boldsymbol{\theta}_0} (\theta_i - \theta_{0,i}). \tag{1.7}$$

[*7] 重要なのはニューロン数に関するオーダーである．そのため，ニューロン数に依存しない定数 β，γ を用いて $\mathcal{N}(0,\beta^2)$，$\gamma/\sqrt{n}$ なども許されるが，本稿では表記の簡略化のためオーダーの定数部分は基本的に 1 とする．

[*8] パラメータ空間上の識別不能な点において Fisher 情報行列が縮退しており，特異点と呼ばれる．そのような構造を持つモデルとして，深層ニューラルネットワークの他に混合分布モデルや隠れマルコフモデル，縮小ランク回帰モデルなども該当し，一般に特異モデル（singular model）と呼ばれる．特異モデルではパラメータ空間における特定の点の周りに推定量が集中することがないため，最尤推定量における漸近正規性が成り立たない．1.1.1 項における汎化誤差の挙動に古典的な学習理論（例：AIC）が直接適用されないのもそのためである．

これは，入力 $\boldsymbol{x}$ に関して非線形な入出力関係を学習可能であるが，パラメータ $\boldsymbol{\theta}$ に関しては線形なモデルといえる．結果だけ述べると，パラメータ空間上で接空間を考えて学習ダイナミクスを線形微分方程式として解くことができ，学習後の予測関数 f^* が以下のように記述される：

$$f^*(\boldsymbol{x};\boldsymbol{\theta}^*) = f(\boldsymbol{x};\boldsymbol{\theta}_0) + \boldsymbol{K}_0(\boldsymbol{x},\mathcal{X})\boldsymbol{K}_0^{-1}(\mathcal{X},\mathcal{X})(\boldsymbol{y} - \boldsymbol{f}(\mathcal{X};\boldsymbol{\theta}_0)) \tag{1.8}$$

ただし，$\boldsymbol{y} \coloneqq (y_1, y_2, \ldots, y_{|\mathcal{D}|})^\top$，$\boldsymbol{f}(\mathcal{X};\boldsymbol{\theta}_0) \coloneqq (f(\boldsymbol{x}_1;\boldsymbol{\theta}_0), f(\boldsymbol{x}_2;\boldsymbol{\theta}_0), \ldots, f(\boldsymbol{x}_{|\mathcal{D}|};\boldsymbol{\theta}_0))^\top$ とした．また，$\boldsymbol{K}_0(\boldsymbol{x},\mathcal{X})$ と $\boldsymbol{K}_0(\mathcal{X},\mathcal{X})$ は，f の $\boldsymbol{\theta}$ に関する勾配の内積で定義される NTK と呼ばれる関数

$$k_t(\boldsymbol{z},\boldsymbol{z}') \coloneqq \sum_{i=1}^{p} \left.\frac{\partial f(\boldsymbol{z};\boldsymbol{\theta})}{\partial \theta_i}\right|_{\boldsymbol{\theta}_t} \left.\frac{\partial f(\boldsymbol{z}';\boldsymbol{\theta})}{\partial \theta_i}\right|_{\boldsymbol{\theta}_t} \tag{1.9}$$

を並べたものであり，それぞれ以下のように構成した：

$$\boldsymbol{K}_0(\boldsymbol{x},\mathcal{X}) \coloneqq \begin{pmatrix} k_0(\boldsymbol{x},\boldsymbol{x}_1) & k_0(\boldsymbol{x},\boldsymbol{x}_2) & \ldots & k_0(\boldsymbol{x},\boldsymbol{x}_{|\mathcal{D}|}) \end{pmatrix}, \tag{1.10}$$

$$\boldsymbol{K}_0(\mathcal{X},\mathcal{X}) \coloneqq \begin{pmatrix} k_0(\boldsymbol{x}_1,\boldsymbol{x}_1) & \cdots & k_0(\boldsymbol{x}_1,\boldsymbol{x}_{|\mathcal{D}|}) \\ \vdots & \ddots & \vdots \\ k_0(\boldsymbol{x}_1,\boldsymbol{x}_{|\mathcal{D}|}) & \cdots & k_0(\boldsymbol{x}_{|\mathcal{D}|},\boldsymbol{x}_{|\mathcal{D}|}) \end{pmatrix}. \tag{1.11}$$

式 1.8 のモデルに訓練データの入力 $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{|\mathcal{D}|}$ を代入すると

$$\boldsymbol{f}^*(\mathcal{X};\boldsymbol{\theta}^*) = \boldsymbol{y} \tag{1.12}$$

となり，$\mathcal{D}$ の入出力関係を実現していることがわかる．NTK の理論は，NTP の下で $n \to \infty$ としたとき，勾配降下法による学習の収束先が，初期化時の NTK $k_0(\boldsymbol{z},\boldsymbol{z}')$ をカーネル関数として用いたときのカーネル法による解と等価であることを意味している [図 1.2]．カーネル法と NTK $k_0(\boldsymbol{z},\boldsymbol{z}')$ の関係を簡単に確認しよう．一般のカーネル関数 $k(\boldsymbol{z},\boldsymbol{z}') = \langle \boldsymbol{\Phi}(\boldsymbol{z}), \boldsymbol{\Phi}(\boldsymbol{z}') \rangle$ は，直観的には 2 つのデータ点 $\boldsymbol{z}, \boldsymbol{z}'$ 間の類似度を，特徴写像 $\boldsymbol{\Phi}(\cdot)$ で移した高次元空間上における内積として測る関数である[*9]．カーネル法では，そのカーネル関数を用いて $f(\boldsymbol{z}) \coloneqq \sum_{i=1}^{|\mathcal{D}|} c_i k(\boldsymbol{z},\boldsymbol{x}_i)$ のように重み付けの和としてモデルを構成する．すなわち，このモデルにおける学習可能パラメータは $\{c_i\}_{i=1}^{|\mathcal{D}|}$ であり，f は学習可能パラメータに関して線形なモデルといえる．f に $\mathcal{X}$ を代入して行列形式で書くと，

$$\begin{pmatrix} f(\boldsymbol{x}_1) \\ \vdots \\ f(\boldsymbol{x}_{|\mathcal{D}|}) \end{pmatrix} = \underbrace{\begin{pmatrix} \langle \boldsymbol{\Phi}(\boldsymbol{x}_1), \boldsymbol{\Phi}(\boldsymbol{x}_1) \rangle & \cdots & \langle \boldsymbol{\Phi}(\boldsymbol{x}_{|\mathcal{D}|}), \boldsymbol{\Phi}(\boldsymbol{x}_1) \rangle \\ \vdots & \ddots & \vdots \\ \langle \boldsymbol{\Phi}(\boldsymbol{x}_1), \boldsymbol{\Phi}(\boldsymbol{x}_{|\mathcal{D}|}) \rangle & \cdots & \langle \boldsymbol{\Phi}(\boldsymbol{x}_{|\mathcal{D}|}), \boldsymbol{\Phi}(\boldsymbol{x}_{|\mathcal{D}|}) \rangle \end{pmatrix}}_{=: \boldsymbol{K}(\mathcal{X},\mathcal{X})} \underbrace{\begin{pmatrix} c_1 \\ \vdots \\ c_{|\mathcal{D}|} \end{pmatrix}}_{=: \boldsymbol{c}} \tag{1.13}$$

すなわち

$$\boldsymbol{f}(\mathcal{X}) = \boldsymbol{K}(\mathcal{X},\mathcal{X})\boldsymbol{c} \tag{1.14}$$

[*9] 例えば，任意の $\gamma > 0$ に対し，ガウス（RBF）カーネル $k(\boldsymbol{z},\boldsymbol{z}') = \exp\left(-\gamma\|\boldsymbol{z} - \boldsymbol{z}'\|_2^2\right)$ などがよく用いられる．通常，特徴写像 $\boldsymbol{\Phi}(\cdot)$ を明示的に計算する必要がない点（カーネルトリック）がカーネル法における最大のポイントであるが，本稿では割愛する．カーネル法の理論については文献 [24] などが詳しい．

のように書ける．さて，NTK $k_0(\boldsymbol{z}, \boldsymbol{z}')$ の場合は，特徴写像 $\boldsymbol{\Phi}(\cdot)$ として，深層ニューラルネットワークの学習可能パラメータ $\boldsymbol{\theta}$ に関する勾配に初期値 $\boldsymbol{\theta}_0$ を代入した p 次元空間への写像

$$\boldsymbol{\Phi}(\boldsymbol{z}) = \left(\left.\frac{\partial f(\boldsymbol{z};\boldsymbol{\theta})}{\partial \theta_1}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \left.\frac{\partial f(\boldsymbol{z};\boldsymbol{\theta})}{\partial \theta_2}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \cdots \left.\frac{\partial f(\boldsymbol{z};\boldsymbol{\theta})}{\partial \theta_p}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_0} \right)^\top \tag{1.15}$$

を用いた形になっており，NTK はこの内積 $k_0(\boldsymbol{z}, \boldsymbol{z}') = \langle \boldsymbol{\Phi}(\boldsymbol{z}), \boldsymbol{\Phi}(\boldsymbol{z}') \rangle$ である[*10]．これを用いて式 1.11 のように $\boldsymbol{K}_0(\mathcal{X}, \mathcal{X})$ を構成し，

$$\boldsymbol{f}(\mathcal{X}) = \boldsymbol{f}(\mathcal{X};\boldsymbol{\theta}_0) + \boldsymbol{K}_0(\mathcal{X}, \mathcal{X})\boldsymbol{c} \tag{1.16}$$

というモデルの学習を考える．$\boldsymbol{f}(\mathcal{X})$ が訓練データ $\mathcal{D}$ の入出力関係を再現するように $\boldsymbol{c}$ を決めよう．教師信号（正解ラベル）$\boldsymbol{y}$ との誤差

$$\|\boldsymbol{y} - (\boldsymbol{f}(\mathcal{X};\boldsymbol{\theta}_0) + \boldsymbol{K}_0(\mathcal{X}, \mathcal{X})\boldsymbol{c})\|_2^2 \tag{1.17}$$

を最小化するように，

$$\boldsymbol{c}^* = \boldsymbol{K}_0^{-1}(\mathcal{X}, \mathcal{X})(\boldsymbol{y} - \boldsymbol{f}(\mathcal{X};\boldsymbol{\theta}_0)) \tag{1.18}$$

とすればよい[*11]．これを式 1.16 の $\boldsymbol{c}$ として定めれば，

$$\boldsymbol{f}(\mathcal{X}) = \boldsymbol{f}(\mathcal{X};\boldsymbol{\theta}_0) + \boldsymbol{K}_0(\mathcal{X}, \mathcal{X})\boldsymbol{K}_0^{-1}(\mathcal{X}, \mathcal{X})(\boldsymbol{y} - \boldsymbol{f}(\mathcal{X};\boldsymbol{\theta}_0)) \tag{1.19}$$

となり，確かに式 1.8 に訓練データの入力 $\mathcal{X}$ を代入した形になっている．

NTK の理論により，深層ニューラルネットワークが生じ得る学習形態の一種を，見通しの良いカーネル法と対応付けた記述が可能となった．初期化時の NTK で支配される学習ダイナミクスのように，パラメータ $\boldsymbol{\theta}$ の変化が初期値近傍で閉じ，モデルが線形化されるような状況は **lazy レジーム** (*lazy regime*) などと呼ばれる [25].

では，現実の機械学習モデルは，常に“lazy”な学習を行っているのだろうか．実はそうとは限らない．例えば，深層ニューラルネットワークを用いた代表的な機械学習モデルである ResNet を用いた実験によると，パラメータが学習初期に初期値から大きく離れ，その間，初期化時の NTK による解に比べて汎化性能を 3 倍程度向上させていることが知られている [26]．また，GPT (Generative Pretrained Transformer) [6–8] や BERT (Bidirectional Encoder Representations from Transformers) [27] などの大規模モデルの学習で重要な役割を果たす事前学習 (pretraining) の技術は，lazy レジームを超えた学習を示す代表例といえる．例えば，BERT における事前学習では，大量のテキストデータを用いて，部分的に隠されたトークンを予測するタスク (masked language modeling) や，隣接する文の関係性を判定するタスク (next sentence prediction) を通してパラメータ $\boldsymbol{\theta}$ を調整し，言語の意味構造を捉えた汎用的な特徴表現を獲得することを目指す [27]．こうして得られた $\boldsymbol{\theta}^*$ は，感情分析や文書分類などの下流

[*10] カーネル関数は，2 つの条件（対称性と半正定値性）を満たす必要がある．NTK がカーネル関数としての定義（特に正定値性）を満たすことの証明等は原論文 Jacot *et al.* (2018) を参照されたい [1].

[*11] ここでは $\boldsymbol{K}_0(\mathcal{X}, \mathcal{X})$ が正則行列であるとするが，特異行列であれば Moore-Penrose 逆行列を用いる．

タスクに応じて微調整される．この過程をファインチューニング (fine-tuning) と呼び，モデル出力に近い層のパラメータのみを再学習させることが多い．事前学習を通して基礎的な特徴表現をパラメータにあらかじめ埋め込むことによって，少量のデータを用いた効率的なファインチューニングを可能にしている．パラメータが初期値近傍に留まったまま訓練データ $\mathcal{D}$ を単純に内挿する lazy レジームでは，この技術は成り立たないだろう．上記のように，データやタスクに内在する何かしらの“特徴”をパラメータが獲得する能力を **特徴学習** (*feature learning*) と呼ぶ．また，このような学習ダイナミクスは，lazy レジームと対比させて **rich レジーム** (*rich regime*) などとも呼ばれる [3]．ここで，1.1.1 項の話題に戻ると，過剰パラメータ領域における深層ニューラルネットワークの性能は，この特徴学習能力と密接に関係していると考えるのが自然だろう．

1.2 主要な問いと研究結果

深層ニューラルネットワークにおける特徴学習を定量的に捉えることは難しい．なぜなら，データやタスクの“特徴”自体の明確な定義付けが困難であり，未知または複雑な場合が多いからである．学習に用いるデータの分布（種類）やモデルのアーキテクチャ，および最適化アルゴリズムなど，様々な要因にも依存するだろう．一方，パラメータ $\boldsymbol{\theta}$ が学習中に大きく変化することが特徴学習の必要条件であることは明らかである．パラメータの変化は，深層ニューラルネットワークにおける中間表現の変化をもたらす．そこで，本稿では，深層ニューラルネットワークにおける特徴学習の概念を，以下のように限定的に定義する：

> 入力 $\boldsymbol{x}$ に対する第 ℓ 層の中間表現を $\widetilde{\boldsymbol{h}}^{(\ell)} \in \mathbb{R}^{n_\ell}$，勾配降下法による最適化時の中間表現の変化を $\Delta\widetilde{\boldsymbol{h}}^{(\ell)} \in \mathbb{R}^{n_\ell}$ とする．n_ℓ は第 ℓ 層のニューロン数である．このとき，$\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ の要素 $|\Delta\widetilde{h}_i^{(\ell)}|$ の n_ℓ に関するオーダーを，その層における特徴学習の度合いとする．

$|\Delta\widetilde{h}_i^{(\ell)}|$ の n_ℓ に関するオーダーは，主に初期化と規格化の方法に依存する．例えば，NTP の場合は

$$|\Delta\widetilde{h}_i^{(\ell)}| = \Theta\left(\frac{1}{\sqrt{n_\ell}}\right) \tag{1.20}$$

となる（2.4 節）[28]．このことからも，ニューロン数の増加に伴って更新量が減少し，lazy な学習が生じることがうかがえる．一方，Yang *et al.* (2021) は，NTP と異なる初期化と規格化によって，ニューロン数が無限大の極限でも lazy レジームを回避する *Maximal Update Parametrization* (μP) を提案した [3]．μP では，更新量がニューロン数に依存しない学習を実現する．すなわち

$$|\Delta\widetilde{h}_i^{(\ell)}| = \Theta(1) \tag{1.21}$$

である（2.4 節）．これにより，ニューロン数が無限大の極限においても特徴学習能力が失われ

ないことを理論的に保証することができる[*12].

μP の理論では，すべての中間層のニューロン数（CNN ではチャネル数）が同じオーダーで無限大に漸近する状況を想定している．では，中間層ごとに異なるオーダーで増加する場合はどうだろうか．特に，μP と同様に，すべての中間層で $|\Delta\widetilde{h}_i^{(\ell)}| = \Theta(1)$ とできるだろうか．この問いは，層が深いニューラルネットワークの学習を特徴付ける上で重要である．例えば，$n_{\ell-1} = \Theta(n^2)$，$n_\ell = \Theta(n)$，$n_{\ell+1} = \Theta(n^2)$ であるような場合，n を増加させるとボトルネック構造が強調されていく．このように，中間層のニューロン数が層によって大きく異なる状況でも各層で特徴学習能力をできるだけ保ち，かつ学習が安定に進む設定を見つけたい．

本稿では，各層で効果的な最適化が行われるという条件の下で $|\Delta\widetilde{h}_i^{(\ell)}| = \Theta(1)$ とできるのは，最小のニューロン数を持つ中間層に限られることを主張する．各層で効果的な最適化が行われるという条件の 1 つとして，すべての中間層で

$$\left| \left\langle \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}^{\ell}}, \Delta\widetilde{\boldsymbol{h}}^{(\ell)} \right\rangle \right| = \Theta(1) \tag{1.22}$$

であることを要請する．ただし，損失を $\mathcal{L}$ とした．この条件は，勾配降下法による最適化に伴う中間表現の更新量 $\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ がニューロン数に依存せずに損失の減少に寄与することを意味する．L 層ニューラルネットワークの学習が安定して進むための基準として他にもいくつかの条件を考えたとき，中間層において $\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = \Theta\left(n_{\text{min}}^r\right)$，すなわち

$$|\Delta\widetilde{h}_i^{(\ell)}| = \Theta\left(\sqrt{\frac{n_{\text{min}}^{2r}}{n_\ell}}\right) \tag{1.23}$$

であることが示唆される（2.3.2 項）．ここで，n_{min} は中間層における最小のニューロン数，$r \in [0, 1/2]$ はニューロン数や ℓ に依らない共通の定数である．従って，$r = 1/2$ のときが更新量がとり得るオーダーの上限である．上記の基準は，以下の事実を根拠とする：

1. 順伝播時に $|\widetilde{h}_i^{(\ell)}| = \Theta(1)$，かつ更新時に式 1.22 が満たされるならば，中間表現の更新量 $\|\Delta\widetilde{\boldsymbol{h}}^{(1)}\|_2$，$\|\Delta\widetilde{\boldsymbol{h}}^{(2)}\|_2$，…，$\|\Delta\widetilde{\boldsymbol{h}}^{(L-1)}\|_2$ は同じオーダーを持つ [28]（2.3 節）．
2. 学習の進行に伴う勾配の発散を防ぐためには

$$\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = O\left(\sqrt{n_{\ell-1}}\right) \tag{1.24}$$

を満たす必要がある（2.3.2 項）．

[*12] μP は，工学的に非常に有益な性質を持っている．大規模なモデルで学習する際，問題となるのはその計算コストである．特に，最適なハイパーパラメータ（例：大域的学習率）の探索には多くの試行が必要となり，計算コストが膨大となる．一方，μP の下では，比較的小規模なモデルにおけるハイパーパラメータの最適値が，大規模なモデルでもほぼそのまま最適値になっていることが実験的に検証されている [29]．そのため，計算コストの低い小規模なモデルでハイパーパラメータ探索を行い，その最適値を用いて大規模なモデルの学習を 1 度だけ行えばよい（この枠組みを μTransfer と呼ぶ）．この特性は，μP の学習ダイナミクスがニューロン数に依存しないことの恩恵である．従来の標準的な初期化手法（例：LeCun の初期化 [30] や Xavier の初期化 [31]，He の初期化 [32]）では，中間層のニューロン数を単純に増加させると，勾配の消失や発散，モデル出力の発散などの問題が生じ，最適なハイパーパラメータも一般に変化してしまう．μP を用いることで，大規模モデルのハイパーパラメータ探索に要する膨大な計算コストを大幅に削減でき，深層学習の効率的な運用を可能にする．

便宜のため，式 1.23 を満たす設定を *Dynamic Parametrization* (DP) と呼ぶことにする（式 1.23 を実現する具体的な方法については 2.4 節で示す）．

μP は，基本的に中間層のニューロン数が等しい状況，すなわち $n_1 = n_2 = \cdots = n_{L-1}$ を想定している．Yang *et al.* (2023) は，μP の拡張として，中間層のニューロン数が層によって異なる状況で $|\Delta\widetilde{h}_i^{(\ell)}| = \Theta(1)$ を実現する *Spectral Parametrization* を，重み行列とその更新量 $\boldsymbol{W}_\ell$，$\Delta\boldsymbol{W}_\ell$ の作用素ノルム（operator norm，spectral norm）のオーダーの観点から提案した [5]．本稿では，Spectral Parametrization の下で，すべての中間層のニューロン数が同じオーダーで無限大に漸近する場合には式 1.22 が成り立つが，中間層ごとに異なるオーダーを持つ場合には成り立たず，ニューロン数が層によって大きく異なる状況では効果的な最適化が困難になることを示す（2.5 節）．一方，DP では，中間層ごとに異なるオーダーを持つ場合でも式 1.22 が成り立ち，各層で効果的な最適化が行われることを確認した．すなわち，すべての中間層におけるパラメータの更新能力がニューロン数に依存しないという意味で対称性が高い状況（Spectral Parametrization）よりも，対称性が最小のニューロン数 n_{min} の方向に崩れた状況（DP）で安定することを主張する．特に，式 1.23 より，DP ではニューロン数 n_ℓ が $n_\ell \gg n_{\text{min}}$ の中間層では lazy レジームを受容する必要があり，特徴学習能力と学習の安定性にはトレードオフの関係があることが示唆される．

本稿の最後に，標準的なデータセット CIFAR-10 [4] を対象とした画像分類タスクで DP と Spectral Parametrization を比較し，中間層のニューロン数が層によって大きく異なる状況において DP が訓練性と汎化性能の両面で優位性があることを実験的に検証する（3.3 節）．

表 1.1: 順伝播と更新時における中間層のニューロン数に関するオーダー． Dynamic Parametrization (DP) では特定の n_{min} との関係が現れる．特に，ニューロン数 n_ℓ が $n_\ell \approx n_{\text{min}}$ の層では特徴学習能力を保持するが，$n_\ell \gg n_{\text{min}}$ の層では lazy レジームを受容する必要があることがわかる．ただし，ニューロン数が層によって大きく異なる状況でも安定した学習を実現することができる．

Parametrization	$\lvert\widetilde{h}_i^{(\ell)}\rvert$	$\lvert\Delta\widetilde{h}_i^{(\ell)}\rvert$
NTP	$\Theta(1)$	$\Theta\left(\frac{1}{\sqrt{n_\ell}}\right)$
μP	$\Theta(1)$	$\Theta(1)$
Spectral Parametrization	$\Theta(1)$	$\Theta(1)$
Dynamic Parametirzation $(r = 1/2)$	$\Theta(1)$	$\begin{cases} \Theta(1), & \text{if } \ell \in \{k \mid n_k = n_{\text{min}}\}, \\ \Theta\left(\sqrt{\frac{n_{\text{min}}}{n_\ell}}\right), & \text{if } \ell \in \{k \mid n_k \geq n_{\text{min}}\}. \end{cases}$

1.3　記法

本稿全体を通して，やや注意を要する記法について以下にまとめる：

- 行列やベクトルは**太字**で表し，それらの各成分やその他のスカラ値は細字で表す.
- 行列やベクトルの各成分を下付きの添え字で表す．例：行列 $\boldsymbol{A}$ の第 ij 成分は A_{ij}.
- $a \overset{\text{ass.}}{=} b$：$a$ が b であるという前提（assumption）.
- $a \overset{\text{req.}}{=} b$：$a$ が b であるという要請（requirement）.
- $\delta^{(ik)}$：クロネッカーのデルタ（$i \neq k$ のとき 0，$i = k$ のとき 1）.
- $\boldsymbol{a} \odot \boldsymbol{b}$：ベクトルのアダマール積.
- $\boldsymbol{a} \otimes \boldsymbol{b}$：ベクトルの外積.
- $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$：ベクトルの内積（ドット積）.
- $\| \cdot \|_2$：ユークリッドノルム（L_2 ノルム）.
- $\| \cdot \|_F$：フロベニウスノルム.
- $\mathcal{N}(\mu, \sigma^2)$：平均 μ，標準偏差 σ の正規分布.
- $U(a, b)$：区間 $[a, b]$ 上の一様分布.

また，深層ニューラルネットワークに関して用いる主要な記号を以下にまとめる：

- $L \in \mathbb{N}$: 深層ニューラルネットワークの層数.
- $n_\ell \in \mathbb{N}$：第 ℓ 層のニューロン数.
- $\boldsymbol{W}^{(\ell)} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$：第 ℓ 層の重み行列.
- $\boldsymbol{b}^{(\ell)} \in \mathbb{R}^{n_\ell}$：第 ℓ 層のバイアス項.
- $\widetilde{\boldsymbol{h}}^{(\ell)} \in \mathbb{R}^{n_\ell}$：第 ℓ 層の中間表現（活性化関数の適用前）.
- $\boldsymbol{h}^{(\ell)} \in \mathbb{R}^{n_\ell}$：第 ℓ 層の中間表現（活性化関数の適用後）.
- $\psi(\cdot)$：活性化関数.
- $\psi'(z)$：活性化関数 $\psi(\cdot)$ の，点 z における 1 階導関数もしくは劣微分.
- $g_\ell \in \mathbb{R}$：第 ℓ 層における信号伝播の大きさを調整する係数.
- $\sigma_\ell \in \mathbb{R}$：第 ℓ 層におけるパラメータの初期化時の標準偏差.
- $\gamma_\ell \in \mathbb{R}$：活性化関数の適用に起因する大きさの変動を吸収する係数.
- $\eta \in \mathbb{R}$：大域的学習率（ニューロン数に依存しない学習率）.

第 2 章

学習が安定に進む条件の導出

2.1　深層ニューラルネットワークの定義と学習の枠組み

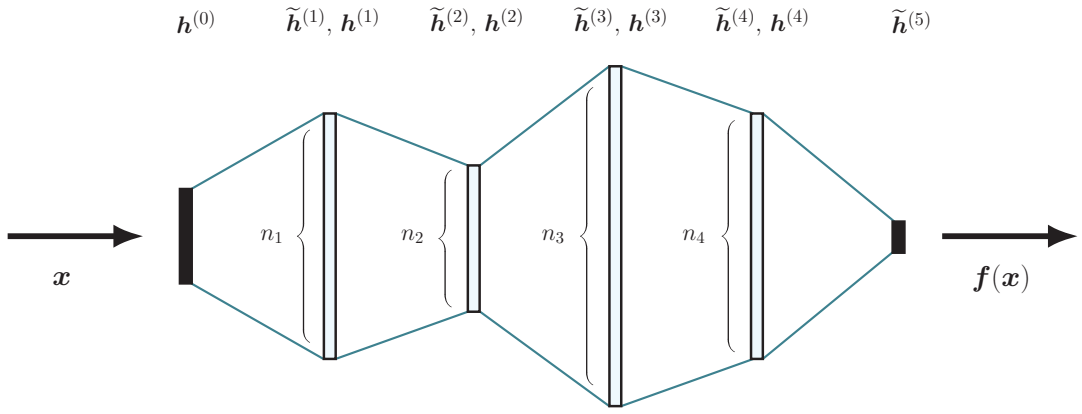

図 2.1: モデルのアーキテクチャ（$L = 5$ のとき）．n_ℓ は第 ℓ 層のニューロン数である．

n_0 次元ベクトル $\boldsymbol{x}$ を入力とする L 層の深層ニューラルネットワーク $\boldsymbol{f}$ として，以下で定義される多層パーセプトロンを考える [図 2.1]：

$$\boldsymbol{f}(\boldsymbol{x}) \coloneqq \widetilde{\boldsymbol{h}}^{(L)}, \tag{2.1}$$

$$\widetilde{\boldsymbol{h}}^{(\ell)} = g_\ell \left[\boldsymbol{W}^{(\ell)} \boldsymbol{h}^{(\ell-1)} + \sqrt{n_{\ell-1}}\, \boldsymbol{b}^{(\ell)} \right], \qquad 1 \le \ell \le L, \tag{2.2}$$

$$\boldsymbol{h}^{(\ell)} = \psi(\widetilde{\boldsymbol{h}}^{(\ell)}), \qquad 1 \le \ell \le L-1, \tag{2.3}$$

$$\boldsymbol{h}^{(0)} = \boldsymbol{x}. \tag{2.4}$$

$\psi(\cdot)$ は，標準的な非線形活性化関数（例：ReLU 関数 $\psi(z) = \max(0, z)$）であり，ベクトルに対してはその要素ごとに適用する．また，$\widetilde{\boldsymbol{h}}^{(\ell)}$ の次元（第 ℓ 層のニューロン数）を n_ℓ とし，$\{n_\ell\}_{\ell=1}^{L-1}$ はすべて十分に大きいとする．$\boldsymbol{W}^{(\ell)} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$ と $\boldsymbol{b}^{(\ell)} \in \mathbb{R}^{n_\ell}$ は，それぞれ重み行列とバイアス項であり，$\boldsymbol{f}$ における学習可能パラメータに相当する．これらの要素は，平均 0，標準偏差 $\gamma_\ell \sigma_\ell$ の正規分布 $\mathcal{N}(0, \gamma_\ell^2 \sigma_\ell^2)$ に従い独立に初期化する．$\gamma_\ell \in \mathbb{R}$ は活性化関数 $\psi(\cdot)$

の適用に起因する大きさの変化を補正するための係数，$g_\ell \in \mathbb{R}$ は信号伝播の大きさを調整するための係数である．ただし，σ_ℓ と g_ℓ は一般にニューロン数に依存するとし，次節以降で学習の安定性を考慮して決定する重要な量である[*1]．

ここで，$\boldsymbol{h}^{(\ell)}$ の次元を拡張して第 $n_\ell + 1$ 成分を常に 1 とするベクトル $\boldsymbol{h}^{*(\ell)} \in \mathbb{R}^{n_\ell+1}$ を新たに作り，同時に $\boldsymbol{W}^{*(\ell)} \coloneqq \left(\boldsymbol{W}^{(\ell)} \sqrt{n_{\ell-1}}\, \boldsymbol{b}^{(\ell)}\right) \in \mathbb{R}^{n_\ell \times (n_{\ell-1}+1)}$ とすれば，$\boldsymbol{f}$ は以下のように再定義できる：

$$\boldsymbol{f}(\boldsymbol{x}) \coloneqq \widetilde{\boldsymbol{h}}^{(L)}, \tag{2.7}$$

$$\widetilde{\boldsymbol{h}}^{(\ell)} = g_\ell \boldsymbol{W}^{*(\ell)} \boldsymbol{h}^{*(\ell-1)}, \qquad 1 \le \ell \le L, \tag{2.8}$$

$$\boldsymbol{h}^{(\ell)} = \psi(\widetilde{\boldsymbol{h}}^{(\ell)}), \qquad 1 \le \ell \le L-1, \tag{2.9}$$

$$\boldsymbol{h}^{(0)} = \boldsymbol{x}. \tag{2.10}$$

入力 $\boldsymbol{x}$ に対応する教師信号（正解ラベル）として $\boldsymbol{y}$ が与えられたとし，この入出力関係 $(\boldsymbol{x}, \boldsymbol{y})$ をモデル $\boldsymbol{f}$ で実現することを考える．ここでは，標準的な損失関数 $\zeta(\boldsymbol{z}, \boldsymbol{z}')$（例：二乗誤差 $\zeta(\boldsymbol{z}, \boldsymbol{z}') = \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{z}'\|_2^2$）を用い，損失

$$\mathcal{L} \coloneqq \zeta(\boldsymbol{y}, \boldsymbol{f}(\boldsymbol{x})) + \lambda\omega \tag{2.11}$$

を最小化するように学習可能パラメータ $\boldsymbol{W}^{*(\ell)}$ を勾配降下法によって初期値から調整していく．ただし，ω は正則化項であり，例えば，$\omega = \frac{1}{2}\sum_{\ell=1}^{L} \|\boldsymbol{W}^{*(\ell)}\|_F^2$ とする weight decay（L_2 正則化）がよく用いられる[*2]．λ は正則化の強さを決める係数であるが，以降では簡単のため原則 $\lambda = 0$ とする．このとき，勾配降下法による更新式は

$$W_{ij}^{*(\ell)} \leftarrow W_{ij}^{*(\ell)} - \eta \frac{\partial \mathcal{L}}{\partial W_{ij}^{*(\ell)}} \tag{2.13}$$

で定義される．ここで，η はすべての層で共通でニューロン数に依存しない学習率である．これを大域的学習率（global learning rate）や定数学習率などと読び，明示的に学習率を層ごとに調整するパラメータ設定手法と区別する．g_ℓ は順伝播信号の大きさを調整するだけでなく，実質的に学習率を層ごとに調整する役割も果たす．明示的に学習率を層ごとに調整するパラメータ設定手法と g_ℓ との関係性については 2.5 節で改めて述べる．

[*1] 例えば，σ_ℓ と g_ℓ をそれぞれ以下のようにとると，1.1.2 項で述べた NTP に一致する：

$$\sigma_\ell = \Theta(1), \tag{2.5}$$

$$g_\ell = \Theta\left(\frac{1}{\sqrt{n_{\ell-1}}}\right). \tag{2.6}$$

[*2] weight decay を適用した場合，パラメータの更新式は

$$W_{ij}^{*(\ell)} \leftarrow W_{ij}^{*(\ell)} - \eta \frac{\partial \zeta}{\partial W_{ij}^{*(\ell)}} - \eta\lambda W_{ij}^{*(\ell)} \tag{2.12}$$

となり，更新に伴い確かに重みが減衰されることがわかる．これは学習レジームに影響を与えると考えられる．この weight decay の影響や，λ の適切な大きさの評価などについては今後の研究課題とする．

2.2　初期化時のオーダー評価

本稿では，学習可能パラメータ $\boldsymbol{W}^{*(\ell)}$ の初期化時や更新時において，ニューロン数に関するオーダー評価を行う．Karkada (2024) は，$L=3$ 層の線形ニューラルネットワーク[*3]を対象として，すべての中間層のニューロン数が同じオーダーで増加する場合の直観的なオーダー評価を提供している [28]．以下では，この評価を任意の層数 L の設定へ帰納的に拡張し，非線形活性化関数 $\psi(\cdot)$ およびバイアス項 $\boldsymbol{b}^{(\ell)}$ が存在する深層ニューラルネットワーク $\boldsymbol{f}$ においても同様に評価できることを確認する．非線形活性化関数の存在については，$\psi(\cdot)$ を適用する前の中間表現 $\widetilde{\boldsymbol{h}}^{(\ell)}$ について評価することで複雑な議論を回避する．バイアス項は，2.1 節で大きさを $\sqrt{n_{\ell-1}}$ 倍かつ $h^{*(\ell)}_{n_\ell+1}=1$ としたことにより，重み行列と統一的に扱うことができる．また，特に中間層のニューロン数が層ごとに異なるオーダを持つ状況に焦点を当て，それが学習ダイナミクスに与える影響について考察する．

本稿で使用する漸近記号 Θ, O, Ω を，それぞれ以下のように定義する：

$$
\begin{aligned}
\Theta(q(n)) &\coloneqq \{p(n) : \exists c_1>0, \exists c_2>0, \exists m\geq 0, \forall n\geq m, 0\leq c_1 q(n)\leq p(n)\leq c_2 q(n)\},\\
O(q(n)) &\coloneqq \{p(n) : \exists c>0, \exists m\geq 0, \forall n\geq m, 0\leq p(n)\leq cq(n)\},\\
\Omega(q(n)) &\coloneqq \{p(n) : \exists c>0, \exists m\geq 0, \forall n\geq m, 0\leq cq(n)\leq p(n)\}.
\end{aligned}
$$

以下では，ネットワークの出力次元 n_L，大域的学習率 η，層数 L，および係数 γ_ℓ は $\Theta(1)$ であるとする．すなわち，これらは定数である．一方，2.1 節で述べた通り，g_ℓ と σ_ℓ は一般に $\{n_\ell\}_{\ell=0}^{L-1}$ に依存するとし，学習の安定性を考慮して本節以降で決定していく．

勾配降下法による学習を行う以前の問題として，入力信号が消失や発散することなく順伝播される必要がある．具体的には，各層への入力である $\boldsymbol{h}^{*(\ell-1)}$ の各要素が，ニューロン数の変化に依存して消失や発散してはならない．そこで

$$
|h_i^{*(\ell-1)}| = \Theta(1) \tag{2.14}
$$

が成り立つための，g_ℓ や σ_ℓ についての制約式を求める．

まず，第 1 層（$\ell=1$）への入力について，$|h_i^{*(0)}| \overset{\text{ass.}}{=} \Theta(1)$ とする．これは，入力データ $\boldsymbol{x}$ の正規化と $\boldsymbol{h}^{*(\ell)}$ の定義を踏まえた自然な仮定であるといえる．次に，任意の ℓ' 層への入力について，$|h_i^{*(\ell'-1)}| \overset{\text{ass.}}{=} \Theta(1)$ が成り立つと仮定する．このとき，$\ell'+1$ 層への入力の第 $i\ (\neq n_{\ell'}+1)$ 成分について，

$$
\begin{aligned}
|h_i^{*(\ell')}| &= |\psi(\widetilde{h}_i^{(\ell')})| && (2.15)\\
&= \left|\psi\left(g_{\ell'}\sum_{j=1}^{n_{\ell'-1}+1} W_{ij}^{*(\ell')} h_j^{*(\ell'-1)}\right)\right| && (2.16)\\
&= \Theta\left(\psi(\gamma_\ell\, g_{\ell'}\sqrt{n_{\ell'-1}}\,\sigma_{\ell'})\right) && (2.17)
\end{aligned}
$$

[*3] 線形ニューラルネットワークは，2.1 節で定義した $\boldsymbol{f}$ において，バイアス項 $\boldsymbol{b}^{(\ell)}$ を省略して活性化関数を恒等写像 $\psi(z)=z$ とした場合に相当する．

となる[*4]．また，第 $n_{\ell'}+1$ 成分については，$\boldsymbol{h}^{*(\ell')}$ の定義により $\left|h^{*(\ell')}_{n_{\ell'}+1}\right| = \Theta(1)$ である．ここで，活性化関数 $\psi(\cdot)$ は，$\Theta(1)$ の入力に対して $\Theta(1)$ の出力を返すとする．すると，$1 \leq \ell' \leq L-1$ において

$$g_{\ell'}\sqrt{n_{\ell'-1}}\sigma_{\ell'} \overset{\text{req.}}{=} \Theta(1) \tag{2.18}$$

とすれば, ニューロン数が十分に多いとき, 帰納的に高い確率で $|\widetilde{h}^{*(\ell-1)}_i| = \Theta(1),\ (1 \leq \ell \leq L)$ が成り立つことが期待される．また，ネットワークの出力がニューロン数の増加に伴って発散してしまうことを防ぐための条件として，$|f_i(\boldsymbol{x})| = \Theta\left(g_L\sqrt{n_{L-1}}\,\sigma_L\right) \overset{\text{req.}}{=} O(1)$ すなわち

$$g_L\sqrt{n_{L-1}}\,\sigma_L \overset{\text{req.}}{=} O(1) \tag{2.19}$$

も要請する．ここで，上記で用いた活性化関数 $\psi(\cdot)$ に関する仮定は，$1 \leq \ell \leq L-1$ において

$$\|\boldsymbol{h}^{(\ell)}\|_2 \overset{\text{ass.}}{=} \Theta\left(\|\widetilde{\boldsymbol{h}}^{(\ell)}\|_2\right) \tag{2.20}$$

であることを意味する．例えば，ReLU 関数の場合，適用後は適用前の約 $1/\sqrt{2}$ になる [5]．そのため，活性化関数が適用される $1 \leq \ell \leq L-1$ では $\gamma_\ell = \sqrt{2}$，それ以外の第 L 層では $\gamma_\ell = 1$ のようにすれば，活性化関数の適用に起因する大きさの変動を吸収することができる．

2.3　勾配降下法によるパラメータの更新

2.3.1　σ_ℓ と g_ℓ が満たすべき条件

$1 \leq \ell \leq L-1$ において，勾配降下法によるパラメータの更新量は以下のように書ける：

$$\Delta W^{*(\ell)}_{ij} = -\eta\frac{\partial\mathcal{L}}{\partial W^{*(\ell)}_{ij}} = -\eta\sum_{k=1}^{n_\ell}\frac{\partial\zeta}{\partial h^{(\ell)}_k}\frac{\partial h^{(\ell)}_k}{\partial\widetilde{h}^{(\ell)}_k}\frac{\partial\widetilde{h}^{(\ell)}_k}{\partial W^{*(ij)}_\ell} \tag{2.21}$$

$$= -\eta\sum_{k=1}^{n_\ell}\frac{\partial\zeta}{\partial h^{(\ell)}_k}\,\psi'(\widetilde{h}^{(\ell)}_k)\,\left(\delta^{(ik)}g_\ell h^{*(\ell-1)}_j\right) \tag{2.22}$$

$$= -\eta g_\ell\frac{\partial\zeta}{\partial h^{(\ell)}_i}\,\psi'(\widetilde{h}^{(\ell)}_k)\,h^{*(\ell-1)}_j. \tag{2.23}$$

ただし，$\psi'(z)$ は非線形活性化関数 $\psi(\cdot)$ の点 z における 1 階導関数もしくは劣微分である．上記は，行列とベクトルを用いて

$$\Delta\boldsymbol{W}^{*(\ell)} = -\eta g_\ell\left(\frac{\partial\zeta}{\partial\boldsymbol{h}^{(\ell)}}\odot\psi'(\widetilde{\boldsymbol{h}}^{(\ell)})\right)\otimes\boldsymbol{h}^{*(\ell-1)} \tag{2.24}$$

と書ける．第 L 層においても同様に，$\Delta\boldsymbol{W}^{*(L)} = -\eta g_L\frac{\partial\zeta}{\partial\boldsymbol{f}(\boldsymbol{x})}\otimes\boldsymbol{h}^{*(L-1)}$ と書ける．また，この最適化に伴い，中間表現 $\widetilde{\boldsymbol{h}}^{(\ell)} = g_\ell\boldsymbol{W}^{*(\ell)}\boldsymbol{h}^{*(\ell-1)}$ は以下のように更新される：

$$\widetilde{\boldsymbol{h}}^{(\ell)} \leftarrow \widetilde{\boldsymbol{h}}^{(\ell)} + \Delta\widetilde{\boldsymbol{h}}^{(\ell)} = g_\ell\big((\boldsymbol{W}^{*(\ell)} + \Delta\boldsymbol{W}^{*(\ell)})(\boldsymbol{h}^{*(\ell-1)} + \Delta\boldsymbol{h}^{*(\ell-1)})\big). \tag{2.25}$$

[*4] 単に分かりやすさのため，Θ 記号の中に定数 γ_ℓ を残している．以降も同様に，漸近記号の中に定数を残している部分があることに注意.

これを展開して考えると，中間表現の更新量 $\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ は以下のように分解できることがわかる：

$$\Delta\widetilde{\boldsymbol{h}}^{(\ell)} = g_\ell \Delta\boldsymbol{W}^{*(\ell)}\boldsymbol{h}^{*(\ell-1)} + g_\ell \boldsymbol{W}^{*(\ell)}\Delta\boldsymbol{h}^{*(\ell-1)} + g_\ell \Delta\boldsymbol{W}^{*(\ell)}\Delta\boldsymbol{h}^{*(\ell-1)}. \tag{2.26}$$

ここで，学習が特定の層のみで進むことを防ぐため，$1 \le \ell \le L$ において

$$\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 \overset{\text{req.}}{=} \Theta\left(\left\|g_\ell \Delta\boldsymbol{W}^{*(\ell)}\boldsymbol{h}^{*(\ell-1)}\right\|_2\right) \tag{2.27}$$

を要請する．すなわち，中間表現の変化量 $\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ は，その層のパラメータの更新 $\Delta\boldsymbol{W}^{*(\ell)}$ に由来する変化量（式 2.26 の第 1 項）が支配的であることを意味する．式 2.27 の右辺の項の中身は，$1 \le \ell \le L-1$ において以下のように計算できる（式 2.24 を代入）：

$$g_\ell \Delta\boldsymbol{W}^{*(\ell)}\boldsymbol{h}^{*(\ell-1)} = g_\ell \left(-\eta g_\ell \left(\frac{\partial\zeta}{\partial\boldsymbol{h}^{(\ell)}} \odot \psi'(\widetilde{\boldsymbol{h}}^{(\ell)})\right) \otimes \boldsymbol{h}^{*(\ell-1)}\right) \boldsymbol{h}^{*(\ell-1)} \tag{2.28}$$

$$= -\eta g_\ell^2 \left(\frac{\partial\zeta}{\partial\boldsymbol{h}^{(\ell)}} \odot \psi'(\widetilde{\boldsymbol{h}}^{(\ell)})\right) \|\boldsymbol{h}^{*(\ell-1)}\|_2^2 \tag{2.29}$$

$$= -\eta g_\ell^2 \|\boldsymbol{h}^{*(\ell-1)}\|_2^2 \frac{\partial\zeta}{\partial\boldsymbol{h}^{(\ell)}} \odot \psi'(\widetilde{\boldsymbol{h}}^{(\ell)}). \tag{2.30}$$

第 L 層においても同様に，$g_L \Delta\boldsymbol{W}^{*(L)}\boldsymbol{h}^{*(L-1)} = -\eta g_L^2 \|\boldsymbol{h}^{*(L-1)}\|_2^2 \frac{\partial\zeta}{\partial\boldsymbol{f}(\boldsymbol{x})}$ と書ける．すなわち，$\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ が損失 $\mathcal{L}$ の勾配に沿っている．同時に，$\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ がニューロン数に依らずに損失 $\mathcal{L}$ の減少に寄与するためには，$1 \le \ell \le L$ において

$$\left|\left\langle \frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell)}},\ \Delta\widetilde{\boldsymbol{h}}^{(\ell)} \right\rangle\right| \overset{\text{req.}}{=} \Theta(1) \tag{2.31}$$

である必要があるが，$\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ が損失 $\mathcal{L}$ の勾配に沿っていることにより，これは

$$\left\|\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell)}}\right\|_2 \|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 \overset{\text{req.}}{=} \Theta(1) \tag{2.32}$$

としてよい．また，$1 \le \ell \le L-1$ において，損失の中間表現に関する勾配は以下のように再帰的に書ける：

$$\frac{\partial\mathcal{L}}{\partial\widetilde{h}_i^{(\ell)}} = \sum_{k=1}^{n_{\ell+1}} \frac{\partial\mathcal{L}}{\partial\widetilde{h}_k^{(\ell+1)}} \frac{\partial\widetilde{h}_k^{(\ell+1)}}{\partial h_i^{*(\ell)}} \tag{2.33}$$

$$= \sum_{k=1}^{n_{\ell+1}} \frac{\partial\mathcal{L}}{\partial\widetilde{h}_k^{(\ell+1)}} g_{\ell+1} W_{ki}^{*(\ell+1)} \tag{2.34}$$

$$= g_{\ell+1} \sum_{k=1}^{n_{\ell+1}} \frac{\partial\mathcal{L}}{\partial\widetilde{h}_k^{(\ell+1)}} W_{ki}^{*(\ell+1)}. \tag{2.35}$$

行列とベクトルを使って書くと $\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell)}} = g_{\ell+1} {\boldsymbol{W}^{*(\ell+1)}}^\top \frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell+1)}}$ であり，このノルムは以下で評価できる：

$$\left\|\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell)}}\right\|_2 = \Theta\left(g_{\ell+1}\sigma_{\ell+1}\sqrt{n_\ell}\left\|\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell+1)}}\right\|_2\right). \tag{2.36}$$

以上により，中間表現の更新量 $\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ のノルムを以下で評価できる（式 2.32 と式 2.36 を適用）：

$$\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = \Theta\left(\left\|\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell)}}\right\|_2^{-1}\right) \tag{2.37}$$

$$= \Theta\left(\frac{1}{g_{\ell+1}\sigma_{\ell+1}\sqrt{n_\ell}}\left\|\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell+1)}}\right\|_2^{-1}\right) \tag{2.38}$$

$$= \Theta\left(\frac{1}{g_{\ell+1}\sigma_{\ell+1}\sqrt{n_\ell}}\|\Delta\widetilde{\boldsymbol{h}}^{(\ell+1)}\|_2\right). \tag{2.39}$$

ここで，初期化時のオーダー評価（2.2 節）で要請した $g_\ell\sqrt{n_{\ell-1}}\,\sigma_\ell \overset{\text{req.}}{=} \Theta(1),\ (1 \le \ell \le L-1)$ を上式に適用すると，中間層の更新量

$$\|\Delta\widetilde{\boldsymbol{h}}^{(1)}\|_2,\ \|\Delta\widetilde{\boldsymbol{h}}^{(2)}\|_2,\ \ldots,\ \|\Delta\widetilde{\boldsymbol{h}}^{(L-1)}\|_2 \tag{2.40}$$

はニューロン数に関して同じオーダーを持つ必要があることがわかる [28]．もし，中間層のニューロン数が層によって異なるオーダーで増加する場合，特徴学習能力について層間で非対称性な構造が生じ得ることが示唆される．なぜなら，初期化時のオーダー評価（2.2 節）において，各層 ℓ に対して $|h_i^{*(\ell)}| = \Theta(1)$ を保証したため $\|\boldsymbol{h}^{*(\ell)}\|_2 = \Theta\left(\sqrt{n_\ell}\right)$ となる一方で，更新量 $\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2$ は必ずしも各層のニューロン数 n_ℓ のオーダーに依存するとは限らず，すべての中間層で同じオーダーを持つことが求められているからである．この条件については，後の議論で再考する（2.3.2 項）．

以下では，g_ℓ と σ_ℓ が満たすべき条件を求める．まず，式 2.31 の $1 \le \ell \le L-1$ において，式 2.27 および式 2.30 を代入し，式 2.32 を適用すると

$$\left|\left\langle\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}^{(\ell)}},\ \Delta\widetilde{\boldsymbol{h}}^{(\ell)}\right\rangle\right| = \Theta\left(\left\langle\frac{\partial\mathcal{L}}{\partial\boldsymbol{h}^{(\ell)}}\odot\psi'(\widetilde{\boldsymbol{h}}^{(\ell)}),\ \ \eta g_\ell^2\|\boldsymbol{h}^{*(\ell-1)}\|_2^2\frac{\partial\zeta}{\partial\boldsymbol{h}^{(\ell)}}\odot\psi'(\widetilde{\boldsymbol{h}}^{(\ell)})\right\rangle\right)$$

$$= \Theta\left(g_\ell^2\|\boldsymbol{h}^{*(\ell-1)}\|_2^2\left\|\frac{\partial\mathcal{L}}{\partial\boldsymbol{h}^{(\ell)}}\odot\psi'(\widetilde{\boldsymbol{h}}^{(\ell)})\right\|_2^2\right) \tag{2.41}$$

$$= \Theta\left(g_\ell^2\|\boldsymbol{h}^{*(\ell-1)}\|_2^2\frac{1}{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2^2}\right) \tag{2.42}$$

となる．これを g_ℓ について解くと，

$$g_\ell = \Theta\left(\frac{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2}{\|\boldsymbol{h}^{*(\ell-1)}\|_2}\right) \tag{2.43}$$

$$= \Theta\left(\frac{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2}{\sqrt{n_{\ell-1}}}\right) \tag{2.44}$$

となる．第 L 層においても同じ結果を得る．ただし，モデル出力 $f_i(\boldsymbol{x}) \coloneqq \widetilde{h}_i^{(L)}$ がニューロン数に依存せずに変化し，消失や発散することなく学習が進むことを保証するため，$|\Delta\widetilde{h}_i^{(L)}| = \Theta(1)$，すなわち

$$\|\Delta\widetilde{\boldsymbol{h}}^{(L)}\|_2 \overset{\text{req.}}{=} \Theta(\sqrt{n_L}) = \Theta(1) \tag{2.45}$$

とし，

$$g_L = \Theta\left(\frac{\sqrt{n_L}}{\sqrt{n_{L-1}}}\right) \tag{2.46}$$

$$= \Theta\left(\frac{1}{\sqrt{n_{L-1}}}\right) \tag{2.47}$$

を得る．また，初期化時のオーダー評価（2.2 節）で要請した $g_\ell\sqrt{n_{\ell-1}}\,\sigma_\ell \overset{\text{req.}}{=} \Theta(1),\ (1 \le \ell \le L-1)$ において，σ_ℓ について解くと

$$\sigma_\ell = \Theta\left(\frac{1}{g_\ell\sqrt{n_{\ell-1}}}\right) \tag{2.48}$$

$$= \Theta\left(\frac{\sqrt{n_{\ell-1}}}{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2\sqrt{n_{\ell-1}}}\right) \tag{2.49}$$

$$= \Theta\left(\frac{1}{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2}\right) \tag{2.50}$$

となる．σ_L については，式 2.39 で $\ell = L-1$ のとき，以下のように解ける：

$$\sigma_L = \Theta\left(\frac{\|\Delta\widetilde{\boldsymbol{h}}^{(L)}\|_2}{g_L\sqrt{n_{L-1}}\|\Delta\widetilde{\boldsymbol{h}}^{(L-1)}\|_2}\right) \tag{2.51}$$

$$= \Theta\left(\frac{\sqrt{n_L}}{\sqrt{\frac{n_L}{n_{L-1}}}\sqrt{n_{L-1}}\|\Delta\widetilde{\boldsymbol{h}}^{(L-1)}\|_2}\right) \tag{2.52}$$

$$= \Theta\left(\frac{1}{\|\Delta\widetilde{\boldsymbol{h}}^{(L-1)}\|_2}\right). \tag{2.53}$$

以上により，学習が安定に進む上で g_ℓ と σ_ℓ が満たすべきオーダーが，中間表現の更新量 $\{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2\}_{\ell=1}^{L-1}$ に依存する形で得られた．

2.3.2　勾配とモデル出力の発散を防ぐための基準

中間表現の更新量 $\{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2\}_{\ell=1}^{L-1}$ は，ニューロン数に関してどのような条件を満たす必要があるだろうか．過大である場合，勾配の発散につながり不安定な学習を招く．ここでは，最適化の進行に伴う勾配の拡大を防ぐための基準を，中間層のニューロン数に依存する形で導出する．最適化の時間経過を t で表し，時間 $t \ge 2$ における逆伝播を考える．損失の中間表現に関する勾配は，以下のように書ける：

$$\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}_{t-1}^{(\ell)}} = g_{\ell+1}\left(\boldsymbol{W}_{t-2}^{*(\ell+1)} + \Delta\boldsymbol{W}_{t-1}^{*(\ell+1)}\right)^\top \frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}} \tag{2.54}$$

$$= g_{\ell+1}\boldsymbol{W}_{t-2}^{*(\ell+1)\top}\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}} + g_{\ell+1}\Delta\boldsymbol{W}_{t-1}^{*(\ell+1)\top}\frac{\partial\mathcal{L}}{\partial\widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}}. \tag{2.55}$$

ここで，第 1 項のノルムは，

$$\left\| g_{\ell+1} \boldsymbol{W}_{t-2}^{*(\ell+1)\top} \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}} \right\|_2 = \Theta\left(g_{\ell+1}\sigma_{\ell+1}\sqrt{n_\ell} \left\| \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}} \right\|_2 \right) \tag{2.56}$$

で評価できる．また，第 2 項のノルムは，$\ell \leq L-2$ において

$$\begin{aligned} \left\| g_{\ell+1} \Delta \boldsymbol{W}_{t-1}^{*(\ell+1)\top} \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}} \right\|_2 &= \left\| -\eta g_{\ell+1}^2 \left(\frac{\partial \zeta}{\partial \boldsymbol{h}_{t-2}^{(\ell+1)}} \odot \psi'(\widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}) \otimes \boldsymbol{h}_{t-2}^{*(\ell)} \right)^\top \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}} \right\|_2 \\ &= O\left(g_{\ell+1}^2 \sqrt{n_\ell} \left\| \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}} \right\|_2^2 \right) \end{aligned} \tag{2.57}$$

で評価できる（$\ell = L-1$ でも同じ結果を得る）．ただし，以下が成り立つと仮定した：

$$\left| \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}}^\top \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}} \right| \overset{\text{ass.}}{=} O\left(\left\| \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}} \right\|_2^2 \right). \tag{2.58}$$

式 2.55 において，第 2 項が支配的でないという条件を考えると，以下のように整理できる：

$$g_{\ell+1}^2 \sqrt{n_\ell} \left\| \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}} \right\|_2^2 \overset{\text{req.}}{=} O\left(g_{\ell+1}\sigma_{\ell+1}\sqrt{n_\ell} \left\| \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}} \right\|_2 \right), \tag{2.59}$$

$$\frac{\|\Delta \widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}\|_2^2}{n_\ell} \sqrt{n_\ell} \frac{1}{\|\Delta \widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}\|_2^2} \overset{\text{req.}}{=} O\left(\frac{\|\Delta \widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}\|_2}{\sqrt{n_\ell}} \frac{1}{\|\Delta \widetilde{\boldsymbol{h}}_{t-2}^{(\ell+1)}\|_2} \sqrt{n_\ell} \frac{1}{\|\Delta \widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}\|_2} \right), \tag{2.60}$$

$$\|\Delta \widetilde{\boldsymbol{h}}_{t-1}^{(\ell+1)}\|_2 \overset{\text{req.}}{=} O(\sqrt{n_\ell}). \tag{2.61}$$

従って，$2 \leq \ell \leq L$ において

$$\|\Delta \widetilde{\boldsymbol{h}}^{(\ell)}\|_2 \overset{\text{req.}}{=} O\left(\sqrt{n_{\ell-1}}\right) \tag{2.62}$$

を得る．ここで，学習が安定に進むためには $\|\Delta \widetilde{\boldsymbol{h}}^{(1)}\|_2$, $\|\Delta \widetilde{\boldsymbol{h}}^{(2)}\|_2$, $\ldots$, $\|\Delta \widetilde{\boldsymbol{h}}^{(L-1)}\|_2$ が同じオーダーである必要性（2.3.1 項）を踏まえると

$$\|\Delta \widetilde{\boldsymbol{h}}^{(\ell)}\|_2 \overset{\text{req.}}{=} O(\sqrt{n_{\min}}) \tag{2.63}$$

のように，中間層における最小のニューロン数 $n_{\min} \coloneqq \min(n_1, n_2, \ldots, n_{L-1})$ が上限として現れることが示唆される．また，初期化時のオーダー評価（2.2 節）で要請した $g_L\sqrt{n_{L-1}}\,\sigma_L \overset{\text{req.}}{=} O(1)$ より，

$$g_L \sqrt{n_{L-1}}\, \sigma_L \overset{\text{req.}}{=} O(1), \tag{2.64}$$

$$\frac{1}{\sqrt{n_{L-1}}} \sqrt{n_{L-1}} \frac{1}{\|\Delta \widetilde{\boldsymbol{h}}^{(L-1)}\|_2} \overset{\text{req.}}{=} O(1), \tag{2.65}$$

$$\|\Delta \widetilde{\boldsymbol{h}}^{(L-1)}\|_2 \overset{\text{req.}}{=} \Omega(1) \tag{2.66}$$

を得る．

以上の議論をまとめると，中間層における最小のニューロン数を $n_{\min}$ とし，$1 \leq \ell \leq L-1$ で共通の $r \in [0,\, 1/2]$ に対して

$$\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = \Theta\left(n_{\min}^{r}\right) \tag{2.67}$$

で学習が安定に進むと考えられる．

2.4　最小のニューロン数に基づくパラメータ設定

前節までに得られた主要な量を再掲する．中間層 $1 \leq \ell \leq L-1$ において

$$g_\ell = \Theta\left(\frac{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2}{\sqrt{n_{\ell-1}}}\right), \tag{2.68}$$

$$\sigma_\ell = \Theta\left(\frac{1}{\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2}\right). \tag{2.69}$$

第 L 層において

$$g_L = \Theta\left(\frac{1}{\sqrt{n_{L-1}}}\right), \tag{2.70}$$

$$\sigma_L = \Theta\left(\frac{1}{\|\Delta\widetilde{\boldsymbol{h}}^{(L-1)}\|_2}\right). \tag{2.71}$$

ただし，$1 \leq \ell \leq L-1$ において共通の $r \in [0,\, 1/2]$ に対して

$$\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = \Theta\left(n_{\min}^{r}\right) \tag{2.72}$$

であった．

簡単のためオーダーの定数部分は基本的に 1 とすると， 2.1 節で定義した $\boldsymbol{f}(\boldsymbol{x}) \coloneqq \widetilde{\boldsymbol{h}}^{(L)}$ は，結果的に以下のように書ける（$r \in [0,\, 1/2]$ は ℓ に依らない共通の定数）：

$$\widetilde{\boldsymbol{h}}^{(L)} = \frac{1}{\sqrt{n_{L-1}}}\left[\boldsymbol{W}^{(L)}\boldsymbol{h}^{(L-1)} + \sqrt{n_{L-1}}\,\boldsymbol{b}^{(L)}\right], \tag{2.73}$$

$$\widetilde{\boldsymbol{h}}^{(\ell)} = \frac{n_{\min}^{r}}{\sqrt{n_{\ell-1}}}\left[\boldsymbol{W}^{(\ell)}\boldsymbol{h}^{(\ell-1)} + \sqrt{n_{\ell-1}}\,\boldsymbol{b}^{(\ell)}\right], \qquad 1 \leq \ell \leq L-1, \tag{2.74}$$

$$\boldsymbol{h}^{(\ell)} = \psi(\widetilde{\boldsymbol{h}}^{(\ell)}), \qquad 1 \leq \ell \leq L-1, \tag{2.75}$$

$$\widetilde{\boldsymbol{h}}^{(0)} = \boldsymbol{x}. \tag{2.76}$$

ただし，重み行列とバイアス項の各要素は，

$$W_{ij}^{(\ell)},\; b_i^{(\ell)} \sim \mathcal{N}\left(0,\; \frac{\gamma_\ell^2}{n_{\min}^{2r}}\right) \tag{2.77}$$

で独立に初期化する[*5]．便宜のため，初期化や規格化を上記のように行う設定を *Dynamic Parametrization* (DP) と呼ぶことにする．

[*5] $\gamma_\ell \in \mathbb{R}$ については 2.2 節を参照のこと．

すべての中間層のニューロン数が同じオーダーで増加し，すなわち $\forall \ell \in \{1, \dots, L-1\}$ に対して $n_\ell = \Theta(n)$ であるとき，$1 \leq \ell \leq L-1$ で $\|\Delta \widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = \Theta(n^r)$ となる．各成分について書くと

$$|\Delta \widetilde{h}_i^{(\ell)}| = \Theta\left(\sqrt{\frac{n^{2r}}{n}}\right) \tag{2.78}$$

である．従って $n \to \infty$ の極限では，$r \in [0, 1/2]$ において $r = 1/2$ のときにのみ $|\Delta \widetilde{h}_i^{(\ell)}| = \Theta(1)$ となり lazy レジームを回避することができる．この $r = 1/2$ のときが *Maximal Update Parametrization* (μP) であり，$r = 0$ のときが *NTK Parametrization* (NTP) である[*6]．NTP の場合，ニューロン数が増えると $1/\sqrt{n}$ のオーダーで更新量が減少し，lazy な学習を生じることがうかがえるだろう．

本稿では，安定な学習を実現する条件の下，中間層によって異なるオーダーを持つニューロン数への依存性を追い，$1 \leq \ell \leq L-1$ で $\|\Delta \widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = \Theta\left(n_{\text{min}}^r\right)$，すなわち

$$|\Delta \widetilde{h}_i^{(\ell)}| = \Theta\left(\sqrt{\frac{n_{\text{min}}^{2r}}{n_\ell}}\right) \tag{2.79}$$

という基準を得た．

2.5　学習率を層ごとに調整するパラメータ設定との関係

本稿では大域的学習率 η は $\Theta(1)$ であるとし，各層における信号伝播の大きさを制御する g_ℓ をニューロン数に関して適切なオーダーで設定することを考えた．本節では，$g_\ell = \Theta(1)$ とし，明示的に学習率を層ごとに調整する場合の Dynamic Parametrization (DP) を考える．第 ℓ 層の学習率を η_ℓ とする．

式 2.30 において $g_\ell = \Theta(1)$ とし，η をニューロン数に依存する η_ℓ として調整する場合，変換前の g_ℓ を用いて

$$\eta_\ell \leftarrow g_\ell^2 \tag{2.80}$$

としても同じことである．式 2.26 の右辺における残りの 2 項についても同様の変換が帰納的に適用でき，$\Delta \widetilde{\boldsymbol{h}}^{(\ell)}$ を層ごとの学習率 η_ℓ を用いた形で書くことができる．また，変換前の g_ℓ と σ_ℓ を用いて

$$\sigma_\ell \leftarrow g_\ell \sigma_\ell \tag{2.81}$$

を新たな σ_ℓ とすれば，順伝播信号の大きさも元々のネットワークと等価になる [28]．従って，DP において，$g_\ell = \Theta(1)$ として明示的に層ごとの学習率 η_ℓ を調整する場合，以下のように定

[*6] μP や NTK の一般論については原論文 [3] や解説 [33,34] 等を参照されたい．

めれば良いことがわかる：

$$\eta_\ell = \Theta\left(\frac{n_{\min}^{2r}}{n_{\ell-1}}\right), \tag{2.82}$$

$$\eta_L = \Theta\left(\frac{1}{n_{L-1}}\right), \tag{2.83}$$

$$\sigma_\ell = \Theta\left(\frac{1}{\sqrt{n_{\ell-1}}}\right), \tag{2.84}$$

$$\sigma_L = \Theta\left(\frac{1}{n_{\min}^{r}\sqrt{n_{L-1}}}\right). \tag{2.85}$$

なお，従来の標準的な初期化手法（例：LeCun の初期化 [30] や Xavier の初期化 [31]，He の初期化 [32]）では，$n_{\ell-1} \approx n_\ell$ のとき

$$\eta_\ell = \Theta\left(1\right), \tag{2.86}$$

$$\sigma_\ell = \Theta\left(\frac{1}{\sqrt{n_{\ell-1}}}\right) \tag{2.87}$$

のオーダーになる[*7]．ここで，DP において $r = 0$，すなわち NTP の場合は

$$\eta_\ell = \Theta\left(\frac{1}{n_{\ell-1}}\right), \tag{2.88}$$

$$\sigma_\ell = \Theta\left(\frac{1}{\sqrt{n_{\ell-1}}}\right) \tag{2.89}$$

となるため，NTP では標準的な初期化手法における学習率を $1/n_{\ell-1}$ する（学習率を非常に小さくとる）ことに対応していることがわかる．

[*7] LeCun の初期化 [30]，Xavier の初期化 [31]，He の初期化 [32] は，それぞれ以下のような初期化手法を指す．ただし，バイアス項は 0 で初期化することが多い．

- LeCun の初期化（活性化関数として tanh 型の関数を想定）：

$$W_{ij}^{(\ell)} \sim \mathcal{N}\left(0,\ \frac{1}{n_{\ell-1}}\right) \quad \text{または} \quad W_{ij}^{(\ell)} \sim U\left(-\sqrt{\frac{3}{n_{\ell-1}}},\ \sqrt{\frac{3}{n_{\ell-1}}}\right).$$

- Xavier の初期化（活性化関数として tanh 型の関数を想定）：

$$W_{ij}^{(\ell)} \sim \mathcal{N}\left(0,\ \frac{2}{n_\ell + n_{\ell-1}}\right) \quad \text{または} \quad W_{ij}^{(\ell)} \sim U\left(-\sqrt{\frac{6}{n_\ell + n_{\ell-1}}},\ \sqrt{\frac{6}{n_\ell + n_{\ell-1}}}\right).$$

- He の初期化（活性化関数として ReLU 型の関数を想定）：

$$W_{ij}^{(\ell)} \sim \mathcal{N}\left(0,\ \frac{2}{n_{\ell-1}}\right) \quad \text{または} \quad W_{ij}^{(\ell)} \sim U\left(-\sqrt{\frac{6}{n_{\ell-1}}},\ \sqrt{\frac{6}{n_{\ell-1}}}\right).$$

第 3 章

計算機実験による理論検証

3.1 Spectral Parametrization による最適化の性質

Yang *et al.* (2023) は, ニューロン数が層によって異なる設定において $|\Delta\widetilde{h}_i^{(\ell)}| = \Theta(1)$ を実現する *Spectral Parametrization* を，重み行列とその更新量 $\boldsymbol{W}_\ell$，$\Delta\boldsymbol{W}_\ell$ の作用素ノルム（operator norm, spectral norm）のオーダーの観点から提案した [5]．NTP, μP, Spectral Parametrization, DP について，順伝播と更新時における中間層のニューロン数に関するオーダーを表 1.1 にまとめる．Spectral Parametrization は，$n_1 = n_2 = \cdots = n_{L-1}$ を基本的に想定した μP の拡張になっており，具体的には以下で定義される：

$$\eta_\ell = \Theta\left(\frac{n_\ell}{n_{\ell-1}}\right), \tag{3.1}$$

$$\sigma_\ell = \Theta\left(\frac{1}{\sqrt{n_{\ell-1}}}\min\left\{1, \sqrt{\frac{n_\ell}{n_{\ell-1}}}\right\}\right). \tag{3.2}$$

2.5 節における変換方法を逆向きに適用し，層ごとの学習率 η_ℓ ではなく，学習率 η を $\Theta(1)$ としてニューロン数に依存する g_ℓ を用いた形で書き直すと

$$g_\ell = \Theta\left(\sqrt{\frac{n_\ell}{n_{\ell-1}}}\right), \tag{3.3}$$

$$\sigma_\ell = \begin{cases} \Theta\left(\dfrac{1}{\sqrt{n_\ell}}\right), & \text{if } n_{\ell-1} \le n_\ell, \\ \Theta\left(\dfrac{1}{\sqrt{n_{\ell-1}}}\right), & \text{if } n_{\ell-1} > n_\ell \end{cases} \tag{3.4}$$

となる．

Spectral Parametrization では，$|\Delta\widetilde{h}_i^{(\ell)}| = \Theta(1)$ すなわち $\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = \Theta\left(\sqrt{n_\ell}\right)$ となるため，中間層のニューロン数が同じオーダーでない場合，中間表現の更新量 $\|\Delta\widetilde{\boldsymbol{h}}^{(1)}\|_2$, $\|\Delta\widetilde{\boldsymbol{h}}^{(2)}\|_2, \ldots, \|\Delta\widetilde{\boldsymbol{h}}^{(L-1)}\|_2$ も一般にそれぞれ異なるオーダーを持つことになる（また，式 2.62 を満たさない層も生じ得る）．従って，Spectral Parametrization では，$\Delta\widetilde{\boldsymbol{h}}^{(\ell)}$ が

ニューロン数に依らずに損失 $\mathcal{L}$ の減少に寄与することを要請する

$$\left| \left\langle \frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}^{(\ell)}},\, \Delta \widetilde{\boldsymbol{h}}^{(\ell)} \right\rangle \right| = \Theta(1) \tag{3.5}$$

が満たされないと考えられる．これについて，計算機実験により検証する．

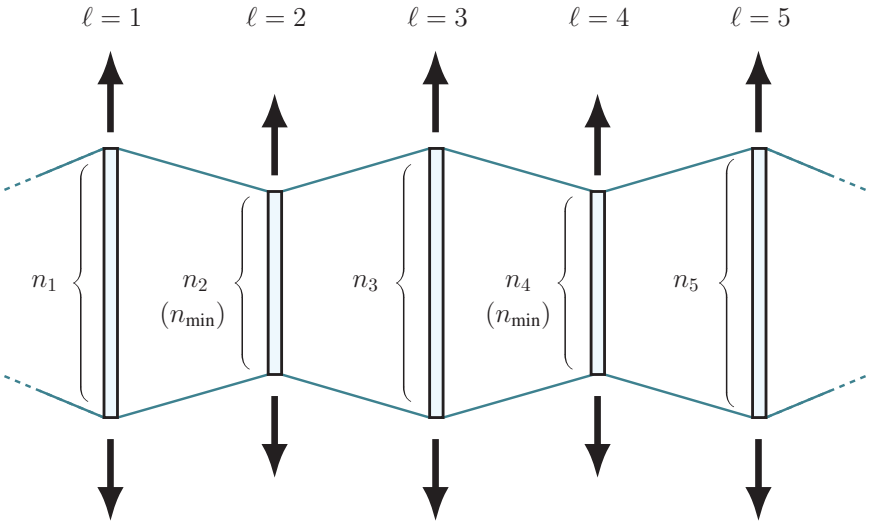

図 3.1: 実験で用いるモデル f の中間層． $n \coloneqq n_1 = n_3 = n_5$，$n_{\text{min}} \coloneqq n_2 = n_4$，とする．

実験の詳細は以下の通りである：

データセットとタスク: CIFAR-10 における `airplane` と `automobile` のクラスから 2000 枚抽出した画像を訓練データとして使用し，2 クラス分類を行う．ただし，教師信号は one-hot 形式とする．

モデル: $L = 6$，$\psi(z) = \max(0, z)$，$\gamma_1 = \cdots = \gamma_{L-1} = \sqrt{2}$，$\gamma_L = 1$，$n_0 = 3072$，$n_6 = 2$，$n \coloneqq n_1 = n_3 = n_5$，$n_{\text{min}} \coloneqq n_2 = n_4$ とする．n を 10^3 から 10^4 まで log-scale で 10 段階用意し，n_{min} について以下の 2 つの設定を考える [図 3.1]：

- Constant Ratio：$n_{\text{min}} = 1/5\, n$．すなわち n_{min} も $\Theta(n)$ であり，n を増やしたときに中間層のニューロン数の比が一定に保たれる．
- Dynamic Ratio：$n_{\text{min}} = 6\sqrt{n}$．すなわち n_{min} が $\Theta(n)$ ではなく $\Theta\left(\sqrt{n}\right)$ であり，n を増やしたときに中間層のニューロン数の比が動的に変わっていく（ボトルネック構造がより強調されていく）．

各モデルは Spectral Parametrization で設定する．バイアス項は省略する．

学習: 大域的学習率 $\eta = 0.1$，バッチサイズ 1，損失関数は二乗誤差 $\zeta(\boldsymbol{z}, \boldsymbol{z}') = \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{z}'\|_2^2$．最適化ステップでは，すべての訓練データからランダムに取り出した 1 つのデータを用いて勾配降下法によりパラメータを更新する．

実験手順: まず，Constant Ratio の設定かつ Spectral Parametrization でモデルを初期化し，$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}^{(\ell)}}$ を各層で計算する．その後，最適化ステップを 1 回行ったときの $\Delta \widetilde{\boldsymbol{h}}^{(\ell)}$ を各層で

計算し，$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}^{(\ell)}}$ との内積の絶対値を記録する．この試行を，ランダムなシード値を用いて 30 回独立に行う．また，$n = 10^3$ から $n = 10^4$ まで行う．次に，Dynamic Ratio の設定かつ Spectral Parametrization で初期化したモデルに対しても同様に行う．実験終了後，試行に関する平均と標準偏差を計算し，グラフを描画する．

実験結果を図 3.2 に示す．すべての中間層のニューロン数が $\Theta(n)$ である Constant Ratio では式 3.5 が成り立つが，中間層によってニューロン数のオーダーが異なる Dynamic Ratio では成り立たず，中間層のニューロン数が層によって大きく異なる状況において絶対値が小さくなることがわかる．従って，Spectral Parametrization をそのまま適用した場合，ボトルネック構造を持つ深層ニューラルネットワークの最適化は困難である可能性が示唆される．

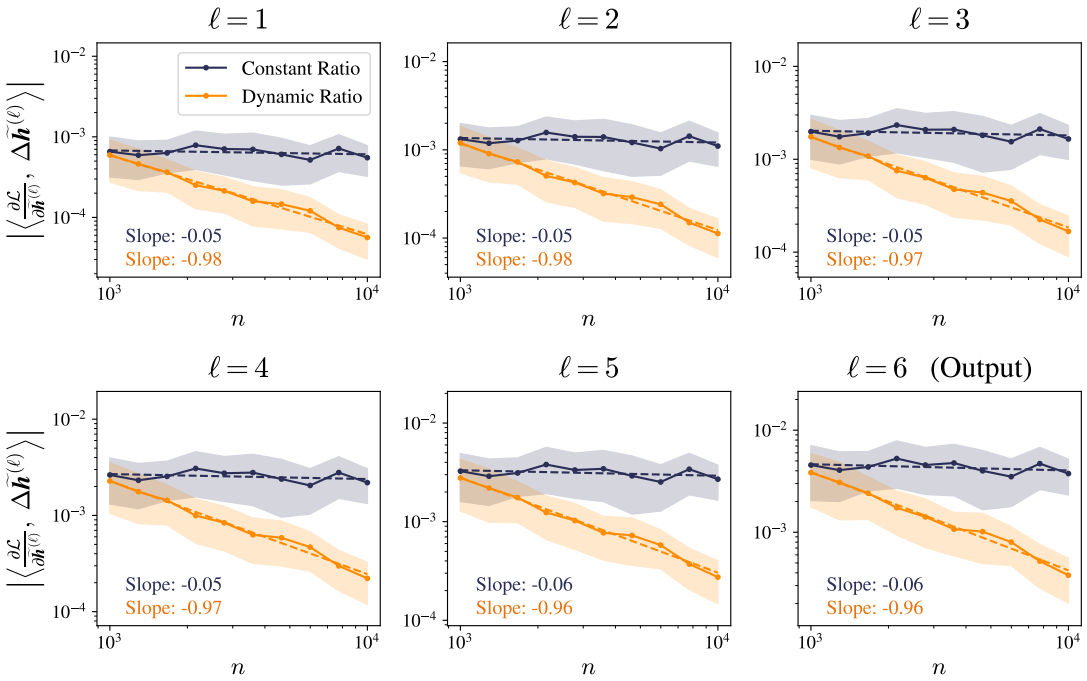

図 3.2: Spectral Parametrization では Dynamic Ratio の状況において式 3.5 が成り立たない．破線は，試行に関する平均に対して最小二乗法でフィッティングしたものである．また，その傾きを Slope として示している．

3.2　Dynamic Parametrization による効果的な最適化

前節では Spectral Parametrization において，中間層のニューロン数が層によって異なるオーダーを持つ Dynamic Ratio の設定では式 3.5 が成立しないことを確認した．本節では，Dynamic Parametrization (DP) を用いることで，Dynamic Ratio の設定でも式 3.5 が成立することを確認する[*1]．

[*1] Constant Ratio の設定かつ DP において式 3.5 が成り立つことは明らかである．

実験の詳細は以下の通りである：

データセットとタスク: 前節と同様.

モデル: $L = 6$, $\psi(z) = \max(0, z)$, $\gamma_1 = \cdots = \gamma_{L-1} = \sqrt{2}$, $\gamma_L = 1$, $n_0 = 3072$, $n_6 = 2$, $n \coloneqq n_1 = n_3 = n_5$, $n_{\text{min}} \coloneqq n_2 = n_4$ とする．n を 10^3 から 10^4 まで log-scale で 10 段階用意し，n_{min} について以下の設定を考える [図 3.1]：

- Dynamic Ratio：$n_{\text{min}} = 150\, n^{1/5}$．すなわち n_{min} が $\Theta(n)$ ではなく $\Theta\left(n^{1/5}\right)$ であり，n を増やしたときに中間層のニューロン数の比が動的に変わっていく（ボトルネック構造がより強調されていく）.

各モデルは DP $(r = 1/2)$ および Spectral Parametrization で設定する．バイアス項は省略する.

学習: 前節と同様.

実験手順: まず，Dynamic Ratio の設定かつ DP でモデルを初期化し，$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}^{(\ell)}}$ を各層で計算する．その後，最適化ステップを 1 回行ったときの $\Delta \widetilde{\boldsymbol{h}}^{(\ell)}$ を各層で計算し，$\frac{\partial \mathcal{L}}{\partial \widetilde{\boldsymbol{h}}^{(\ell)}}$ との内積の絶対値を記録する．この試行を，ランダムなシード値を用いて 30 回独立に行う．また，$n = 10^3$ から $n = 10^4$ まで行う．次に，Dynamic Ratio の設定かつ Spectral Parametrization で初期化したモデルに対しても同様に行う．実験終了後，試行に関する平均と標準偏差を計算し，グラフを描画する.

実験結果を図 3.3 に示す．前節と同様に Spectral Parametirzation では式 3.5 が成り立たない一方で，DP では成り立つことがわかる．すなわち，DP によって，中間層のニューロン数が層によって大きく異なる状況においても，最適化能力がニューロン数に依存しない形で保証されることが確認できた.

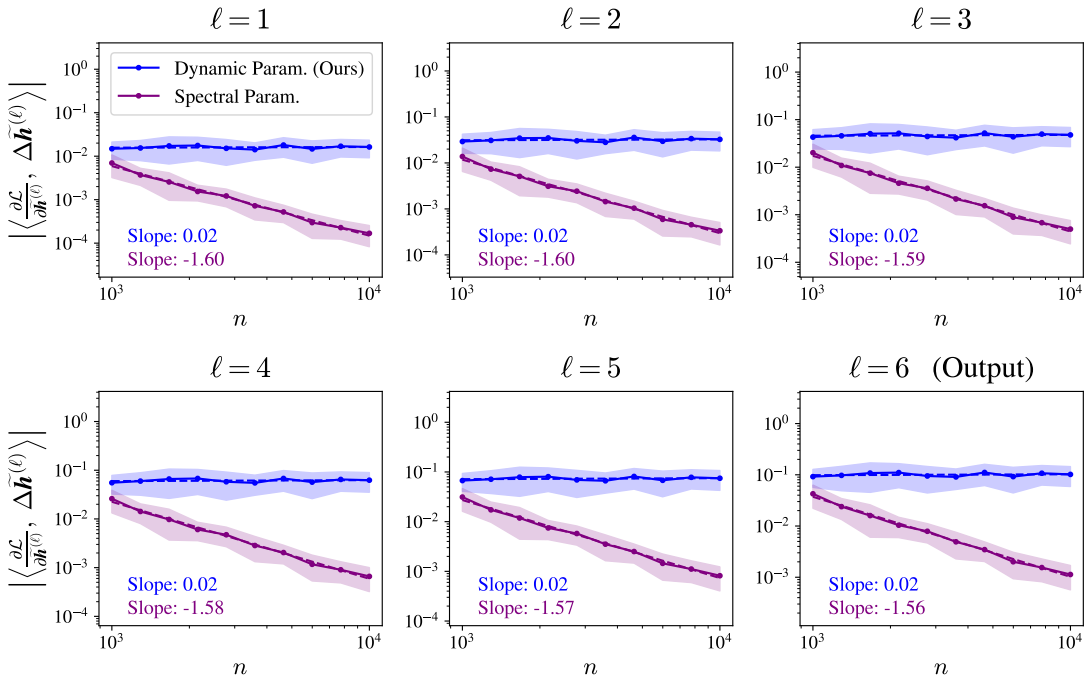

図 3.3: Dynamic Parametrization (DP) では各層で $\Theta(1)$ を保つことができる．破線は，試行に関する平均に対して最小二乗法でフィッティングしたものである．また，その傾きを Slope として示している．

3.3　性能への影響

Spectral Parametrization では，すべての中間層のニューロン数が同じオーダーであるとき，$1 \leq \ell \leq L-1$ において

$$|\Delta \widetilde{h}_i^{(\ell)}| = \Theta\left(1\right) \tag{3.6}$$

を実現する．一方，Dynamic Parametrization (DP) のとき，$r = 1/2$ の設定では

$$|\Delta \widetilde{h}_i^{(\ell)}| = \Theta\left(\sqrt{\frac{n_{\min}}{n_\ell}}\right) \tag{3.7}$$

となる．すなわち，DP では $n_{\min}$ を定数として n_ℓ を増加すると，ニューロン数 n_ℓ が $n_\ell \approx n_{\min}$ の中間層では特徴学習能力が保持される一方で，$n_\ell \gg n_{\min}$ の中間層では $1/\sqrt{n_\ell}$ のオーダーで lazy な学習に近づくことが示唆される．しかし，DP では中間層によってニューロン数が異なるオーダーを持つ場合にも式 2.62 や式 3.5 を満たすため，学習の安定性は高いといえる．従って，特徴学習能力と学習の安定性にはトレードオフの関係があることが示唆される．本節では，中間層のニューロン数が層によって大きく異なる状況において，DP の訓練性および汎化性能が Spectral Parametrization と比較して実践的にどの程度優れているかを検証する．実験の詳細は以下の通りである：

データセットとタスク: CIFAR-10 における `airplane` と `automobile` のクラスから合計 4000 枚抽出した画像を使用して 2 クラス分類を行う．ただし，4000 枚のうち 2000 枚を訓練データとし，残りをテストデータとする．また，教師信号は one-hot 形式とする．

モデル: $L = 6$，$\psi(z) = \max(0, z)$，$\gamma_1 = \cdots = \gamma_{L-1} = \sqrt{2}$，$\gamma_L = 1$，$n_0 = 3072$，$n_6 = 2$，$n_1 = n_3 = n_5 = 10^4$，$n_2 = n_4 = 500$ とする．DP $(r = 1/2)$ および Spectral Parametrization で設定する．バイアス項は省略する．

学習: 大域的学習率 $\eta = 0.1$，バッチサイズ 64，モーメンタム $= 0.9$，損失関数は二乗誤差 $\zeta(\boldsymbol{z}, \boldsymbol{z}') = \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{z}'\|_2^2$ とし，1 つのバッチに含まれるデータに対する損失の平均値を $\mathcal{L}$ とする．最適化ステップでは，訓練データをバッチごとに分割し，各バッチについて勾配降下法を適用してパラメータを更新する．全てのバッチを 1 回ずつ処理し，訓練データ全体を 1 巡する操作を 1 エポックとする．

実験手順: まず，DP でモデルを初期化し，学習を 100 エポック行ったときの訓練損失とテスト損失を記録する．この試行を，ランダムなシード値を用いて 100 回独立に行う．次に，Spectral Parametrization で設定したモデルに対しても同様に行う．実験終了後，試行に関する平均と標準偏差を計算し，グラフを描画する．

実験結果を図 3.4 に示す．DP では学習の進行に伴い，Spectral Parametirzation と比較して訓練損失がより効果的に減少し，テスト損失も低い傾向にあることがわかる．また，使用するデータクラスを変更した場合も同様の結果が得られた（図 3.5，図 3.6，図 3.7）．これらの結果は，ボトルネック構造を持つ深層ニューラルネットワークに対して DP を適用することで，訓練性および汎化性能の向上が期待できることを示唆している[*2]．

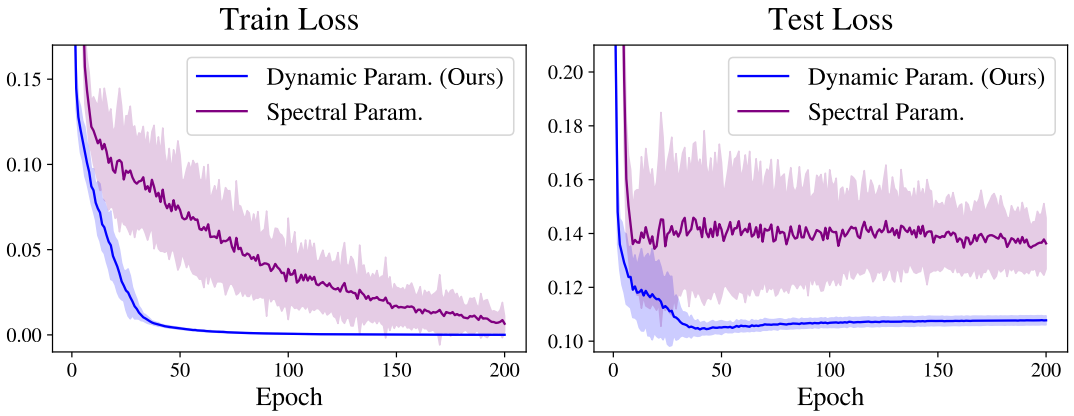

図 3.4: Dynamic Parametrization (DP) は訓練性と汎化性能の両面で優位性がある． `airplane` と `automobile` で 2 クラス分類を行った．試行に関する平均を実線，標準偏差を色付きの帯で示している．

[*2] 2.5 節で述べた通り，g_ℓ は層ごとの学習率に対応しているため，2 つのパラメータ設定における性能の比較には議論の余地があると考えている．ただし，層ごとの学習率としては DP は Spectral Parametrization に比べて常に小さいにもかかわらず訓練損失の減少が速い点は，DP の安定性や最適化能力の高さが現れており興味深い．

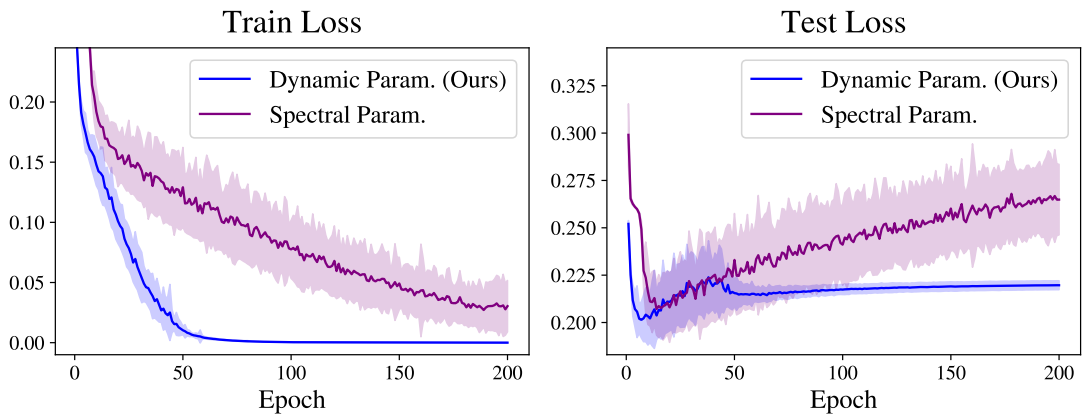

図 3.5: `bird` と `cat` で 2 クラス分類した場合の損失.

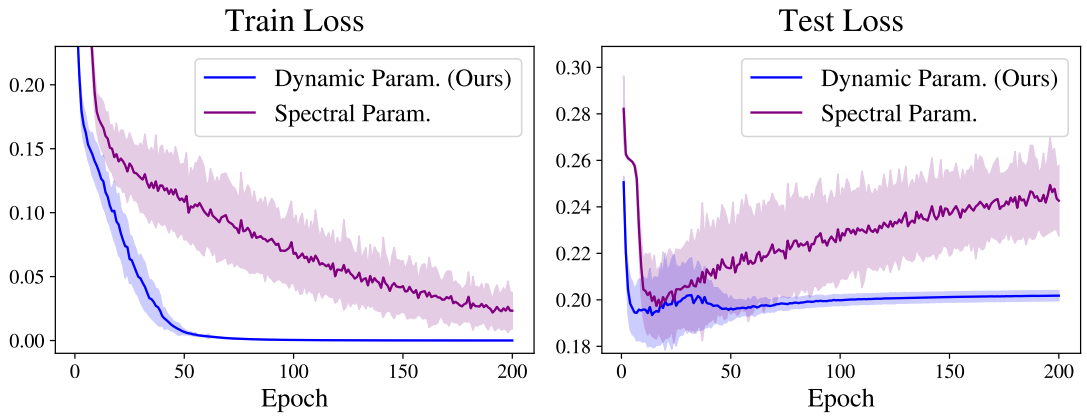

図 3.6: `deer` と `dog` で 2 クラス分類した場合の損失.

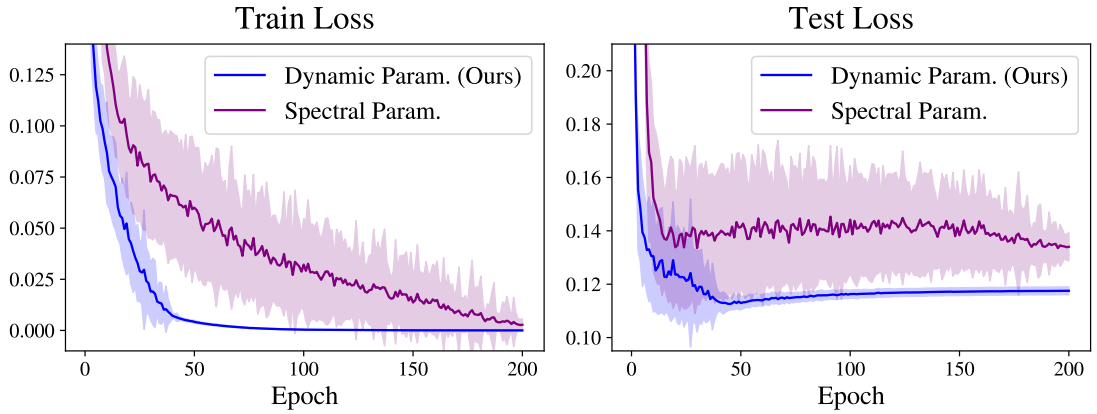

図 3.7: `frog` と `horse` で 2 クラス分類した場合の損失.

第 4 章

おわりに

本研究では，中間層のニューロン数が層によって異なるオーダーで増加する状況に焦点を当て，学習が安定に進むための基準として $\|\Delta\widetilde{\boldsymbol{h}}^{(\ell)}\|_2 = \Theta(n_{\text{min}}^r),\ r \in [0,\, 1/2]$ を導くとともに，この基準を満たす具体的なパラメータ設定を提案した．さらに，μP の拡張である Spectral Parametrization との比較を通じて，中間層のニューロン数が層によって大きく異なる場合においても，提案手法では各層で効果的な最適化が行われ，訓練性および汎化性能の向上が期待できることを実験的に確認した．

深層ニューラルネットワークにおけるボトルネック構造は，層数が少ない浅いモデルには現れない，深層特有の情報処理機構である．そのため，この構造の理解は，深層ニューラルネットワークの情報処理様式を特徴付ける上で重要である．本研究では，ボトルネック構造が強調されるときに生じる情報処理の一端を，シンプルなオーダー評価の枠組みで捉えた．

一方で，本研究の解析にはいくつかの限界がある．特に，ニューロン数が相対的に少ない層では，漸近的な議論における確率的な揺らぎの影響も大きくなると考えられるが，この影響については評価していない．また，本研究は主に学習初期のダイナミクスに焦点を当てたものであり，学習が進み，パラメータが初期値から大きく離れた後の振る舞いについては議論できていない．今後の研究として，このような影響をより詳細に解析し，特徴学習能力と学習の安定性のトレードオフをより厳密に評価することが求められる．また，本研究では特徴学習の概念を中間表現の更新量 $|\Delta\widetilde{h}_i^{(\ell)}|$ のニューロン数に関するオーダーに限定して議論した．そのため，学習によって獲得される具体的な表現については保証されない．例えば，深層ニューラルネットワークが訓練データの特徴をどのような階層構造として抽出するのか，また，ボトルネック部分でどのような情報圧縮のメカニズムが働くのかといった問題は，依然として興味深い研究課題である．

過剰パラメータ系が獲得する情報の表現について，何かしらの普遍則を記述することはできるだろうか．多くの問いが残されている．

謝辞

本研究を遂行するにあたり，日頃から数理脳科学や機械学習に関する議論を通じて多大な影響を与えてくださった XXXXXXXX に深く感謝申し上げます．また，文章表現や内容の整理に関して貴重な助言やご確認をいただき，本論文の執筆において重要な支えとなりました．

さらに，研究活動を通じて共に議論し，学びの場を共有した研究室のメンバーにも心より感謝いたします．皆様からいただいた刺激と支えが本研究を進める原動力となり，学びを深めることができました．

参考文献

[1] Arthur Jacot, Franck Gabriel, and Clément Hongler. (2018). Neural tangent kernel: Convergence and generalization in neural networks, *Advances in Neural Information Processing Systems*, **31**, 8580-8589.

[2] Jaehoon Lee, Lechao Xiao, Samuel S. Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. (2019). Wide neural networks of any depth evolve as linear models under gradient descent, *Advances in Neural Information Processing Systems*, **32**, 8570-8581.

[3] Greg Yang and Edward J. Hu. (2021). Tensor programs IV: Feature learning in infinite-width neural networks, *International Conference on Machine Learning*, *PMLR* **139**, 11727-11737.

[4] Alex Krizhevsky. (2009). Learning multiple layers of features from tiny images, Technical Report, University of Toronto.

[5] Greg Yang, James B. Simon, and Jeremy Bernstein. (2023). A spectral condition for feature learning, *arXiv preprint arXiv:2310.17813*.

[6] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. (2018). Improving language understanding by generative pre-training, Technical Report, OpenAI.

[7] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. (2019). Language models are unsupervised multitask learners, Technical Report, OpenAI.

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. (2020). Language models are few-shot learners, *Advances in Neural Information Processing Systems*, **33**.

[9] 岡崎直観，荒瀬由紀，鈴木潤，鶴岡慶雅，宮尾 祐介 (2022).『IT Text 自然言語処理の基礎』オーム社.

[10] George Cybenko. (1989). Approximation by superpositions of a sigmoidal function, *Mathematics of Control, Signals, and Systems*, **2**(4), 303-314.

[11] Sho Sonoda and Noboru Murata. (2017). Neural network with unbounded activation func-

tions is universal approximator, *Applied and Computational Harmonic Analysis*, **43**(2), 233-268.

[12] Stuart Geman, Elie Bienenstock, and Ren Doursat. (1992). Neural networks and the bias/-variance dilemma, *Neural Computation*, **4**(1), 1-58.

[13] Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro. (2019). The role of over-parametrization in generalization of neural networks, *International Conference on Learning Representations*, https://openreview.net/forum?id=BygfghAcYX.

[14] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. (2019). Reconciling modern machine learning practice and the classical bias-variance trade-off, *Proceedings of the National Academy of Sciences*, **116**(32), 15849-15854.

[15] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. (2020). Deep double descent: Where bigger models and more data hurt, *International Conference on Learning Representations*, https://openreview.net/forum?id=B1g5sA4twr.

[16] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J. Tibshirani. (2022). Surprises in high-dimensional ridgeless least squares interpolation, *The Annals of Statistics*, **50**(2), 949-986.

[17] Stéphane d'Ascoli, Levent Sagun, and Giulio Biroli. (2020). Triple descent and the two kinds of overfitting: Where & why do they appear?, *Advances in Neural Information Processing Systems*, **33**.

[18] Ben Adlam and Jeffrey Pennington. (2020). The neural tangent kernel in high dimensions: Triple descent and a multi-scale theory of generalization, *International Conference on Machine Learning*, *PMLR* **119**, 74-84.

[19] Manfred Opper and Wolfgang Kinzel. (1996). Statistical mechanics of generalization, *Models of Neural Networks III: Association, Generalization, and Representation*, edited by Eytan Domany, J. Leo van Hemmen, and Klaus Schulten, Springer, 151-209.

[20] Siegfried Bös and Manfred Opper. (1997). Dynamics of training, *Advances in Neural Information Processing Systems*, **9**, 141-147.

[21] Anders Krogh and John A. Hertz. (1992). Generalization in a linear perceptron in the presence of noise, *Journal of Physics A: Mathematical and General*, **25**(25), 1135-1147.

[22] Gal Vardi. (2022). On the Implicit Bias in Deep-Learning Algorithms, *arXiv preprint arXiv:2208.12591*.

[23] Tanishq Kumar, Blake Bordelon, Samuel J. Gershman, and Cengiz Pehlevan. (2024). Grokking as the transition from lazy to rich training dynamics, *International Conference on Learning Representations*, https://openreview.net/forum?id=vt5mnLVIVo.

[24] 瀬戸道生，伊吹竜也，畑中健志 (2021).『機械学習のための関数解析入門 ヒルベルト空間とカーネル法』内田老鶴圃.

[25] Lenaic Chizat, Edouard Oyallon, and Francis Bach. (2019). On lazy training in differentiable programming, *Advances in Neural Information Processing Systems*, **32**, 2933-2943.

[26] Stanislav Fort, Gintare Karolina Dziugaite, Mansheej Paul, Sepideh Kharaghani, Daniel M. Roy, and Surya Ganguli. (2020). Deep learning versus kernel learning: an empirical study of loss landscape geometry and the time evolution of the neural tangent kernel, *Advances in Neural Information Processing Systems*, **33**.

[27] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding, *arXiv preprint arXiv:1810.04805*.

[28] Dhruva Karkada. (2019). The lazy (NTK) and rich (μP) regimes: a gentle tutorial, *arXiv preprint arXiv:2404.19719*.

[29] Greg Yang, Edward J. Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. (2021). Tuning large neural networks via zero-shot hyperparameter transfer, *Advances in Neural Information Processing Systems*, **34**, 17084-17097.

[30] Yann LeCun, Léon Bottou, Genevieve B. Orr, and Klaus-Robert Müller. (1998). Efficient BackProp, *Neural Networks: Tricks of the Trade*, edited by Genevieve B. Orr and Klaus-Robert Müller, Springer Berlin Heidelberg, 9-50.

[31] Xavier Glorot and Yoshua Bengio. (2010). Understanding the difficulty of training deep feedforward neural networks, *International Conference on Artificial Intelligence and Statistics*, *JMLR Proceedings*, **9**, 249-256.

[32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. (2015). Delving deep into rectifiers: Surpassing human level performance on ImageNet classification, *International Conference on Computer Vision*, 1026-1034.

[33] 唐木田亮 (2024).「機械学習の仕組み：統計力学的アプローチ」橋本幸士 編『学習物理学入門』朝倉書店.

[34] 唐木田亮 (2024).「幅が大きいニューラルネットに現れる学習の普遍則の今」『日本神経回路学会誌』**31**(4), 167-176.