博士論文

深層学習と生成モデルによる
マルチモーダル学習に関する研究

指導教員 松尾 豊 特任准教授

東京大学大学院 工学系研究科
技術経営戦略学専攻

鈴木 雅大

要旨

我々は日々，多くの情報に接している．こうした情報は様々な種類（モダリティ）をとり，我々は複数のモダリティ，すなわち，マルチモーダルな情報を常に取り入れることで，確実な情報処理を行っている．近年は，コンピュータが人間のように複数のモダリティ情報を活用することで，より正確に予測や判断，推論等を行う試みが重要視されている．このような試みは，機械学習の問題と考えることができ，特に**マルチモーダル学習**と呼ばれる．

マルチモーダル学習の様々な問題設定に共通する困難な点として，モダリティ間の異種性が挙げられる．異種性とは，画像と音声のように互いに形式が大きく異なる性質のことである．さらに，異種性を持つ 2 つのデータ集合には，特徴空間の違い（各モダリティで表されるデータの形は，次元や構造が大きく異なるということ）と分布の違い（異なるモダリティは，同じ対象を表しているとしても，各モーダル集合の分布には大きな差があるということ）がある．このうち，1 つ目の特徴空間の違いへの対処としては，近年，深層ニューラルネットワークを用いた機械学習技術である深層学習によるアプローチが主流となっている．一方で，2 つ目の分布の違いについては，深層ニューラルネットワークで直接対処することはできないが，確率的生成モデルによってモダリティ間の分布の違いを明示的に考慮することができる．したがって本論文では，深層学習と生成モデルを組み合わせることで，異なるモダリティ間の異種性の問題に対処できることに着目する．

深層学習と生成モデルを組み合わせるアプローチは大きく分けて 2 つある．そのうちの 1 つは生成モデルの確率分布をニューラルネットワークで表現するアプローチであり，深層生成モデルと呼ばれる．しかし，マルチモーダル学習の問題設定のそれぞれについて，深層学習と生成モデルのアプローチが有効であることを検証した研究はほとんどない．その理由として，深層学習と生成モデルを組み合わせた研究が本格的に行われるようになってから未だ年月が経っ

ていないこと，そして深層生成モデルを実装するための枠組みが十分に整備されていないことが挙げられる．

そこで本論文では，マルチモーダル学習のそれぞれの問題設定（表現・変換・融合・共学習）において，深層学習と生成モデルを利用したモデルを提案して，有効性を検証することを目標とする．さらに，マルチモーダル情報を含んだ深層生成モデルを，実装・利用するためのライブラリを開発することを目指す．

1 つ目の研究では，マルチモーダル学習における表現と変換の 2 つの問題設定に取り組む深層生成モデルを提案する．この研究では，まず従来の表現問題に対するアプローチを単純に深層生成モデルに拡張した joint multimodal variational autoencoder（JMVAE）を検証し，双方向にモダリティを変換する場合に，情報量の大きいモダリティを欠損させると共有表現や変換したモダリティが崩れてしまい，従来知られていた欠損値補完の手法では解決できないことを示す．この問題を解決するために，本研究では JMVAE-kl と階層的 JMVAE という手法を提案する．実験から，これらの手法によって単一モダリティを入力とした場合でも適切な共有表現が推論できること，従来の 1 方向のモデルと比較して同等以上の精度で双方向の変換ができることを示す．

2 つ目の研究では，融合，すなわち，複数のモダリティから目標ラベルを予測する問題に取り組む．本研究では，異なるモーダルデータがセットで手に入りやすいのに対して，ラベル情報の獲得は人的コストがかかるという背景から，半教師あり学習に取り組む．半教師あり学習とは，少量のラベルあり集合の他に大量のラベルなし集合がある状況で，汎化性能の高い識別モデルを学習する枠組みである．この章では深層生成モデルを用いた半教師ありマルチモーダル学習の手法として，新たに semi-supervised MVAE（SS-MVAE）と semi-supervised HMVAE（SS-HMVAE）を提案する．また半教師ありマルチモーダル学習では，テスト集合に単一のモダリティしか与えられない設定があることから，モダリティが欠損しても精度を落とさずに目標ラベルを予測する手法として，SS-HMVAE を拡張した SS-HMVAE-kl というモデルを提案する．これらの手法でモダリティの欠損が補完できること，そして単一モダリティ及びマルチモーダルにおける既存の半教師あり学習と比較して，提案手法の精度が高いことを実験で確認する．

3 つ目の研究では，共学習の問題設定の一つとして知られるゼロショット学習について取り

組む．ゼロショット学習とは，一度も学習したことのない目標カテゴリのモダリティを，他の目標カテゴリでの学習結果と，異なるモダリティの情報を補助情報として用いて予測する枠組みである．本研究では，分類するモダリティとして画像，補助情報として属性を用いる属性ベースゼロショット学習に着目する．属性ベースゼロショット学習の既存研究では，属性の画像に対する分布の違い（現れやすさ）を明示的に考慮していなかった．本研究では，このような現れやすさの度合いを属性ごとの観測確率と呼び，観測確率を含めて画像や属性，ラベルの関係を記述した生成モデルを提案する．実験では，深層ニューラルネットワークから得た特徴量を利用し，モデルの妥当性の検証及び既存研究との比較実験によって，提案手法が既存手法と比較して有効性の高いモデルであることを示す．

4 つ目の研究では，深層生成モデルを実装するライブラリを提案する．深層ニューラルネットワークを用いた生成モデルを記述できるライブラリは既に存在しているが，それらは深層ニューラルネットワークと確率変数を同じレイヤーとして考えており，本研究で提案するようなマルチモーダル情報を持った複雑な深層生成モデルを実装するためには，モデルやネットワークが変更される度に新たに 1 から実装を行う必要があった．また，深層ニューラルネットワークによって定義された確率分布からサンプリングしたり尤度を計算したりすることは，従来の言語では困難であった．今日の深層生成モデルでは，目的関数として尤度を計算し，潜在変数や各モダリティのデータをサンプリング（推論や生成）することが重要になっている．こうした背景から，深層生成モデルに特化したライブラリ「Tars」を新たに提案する．このライブラリでは，確率分布がニューラルネットワークを隠蔽しており，確率分布のそれぞれでサンプリングや尤度計算できる．また，Tars を用いて様々な分布の深層生成モデルが学習できることを確認する．さらに本ライブラリを使用したアプリケーションを紹介し，本ライブラリで学習した確率分布をそのまま読み込んで画像を生成したりできることを説明する．

以上の研究を通じ，マルチモーダル学習の各問題設定において，深層学習と生成モデルを利用したモデルが有効であることを確認した．また，本論文で提案した深層生成モデルライブラリ Tars が，マルチモーダル情報を含むような複雑な深層生成モデルの実装・利用に適していることを確認した．

最後に，本論文の貢献や限界を確認し，今後への課題についてまとめる．そして本研究の産業応用の可能性ついて触れ，本研究の統一構想や汎用人工知能に向けた考察について述べる．

目次

図目次

表目次

第 1 章

序論

1.1 研究背景

1.1.1 マルチモーダル情報について

我々は日々，多くの情報に接している．自然界には，光や音（空気の振動），化学物質などの様々な形をした情報が満ちており，我々人間はそれらを多種類の感覚器官で認識している[*1]．このような情報及びそれを受け取る感覚の種類は**モダリティ（modality）**と呼ばれ[*2]，我々は複数のモダリティ，すなわち，**マルチモーダル（multimodal）**な情報を常に外界から取り入れることで，単一モーダル（unimodal）よりも確実な情報処理を行っている．

最近では，インターネット上でも多種多様な形式の情報が見られるようになっている．初期のパソコン通信ではテキスト形式に限定されていたが[*3]，インターネットの登場による通信網の発達と，通信速度や CPU 性能，ストレージ容量の向上などによって，画像，音声，動画といった様々な形式が送受信できるようになっている．このため，現在の多くの Web ページが，複数のメディア形式を併用して作られている[*4]．こうした併用は，閲覧者が内容を理解しやすくすることに大きく貢献している．たとえば，インターネットのニュース記事では，文書

[*1] 一般的にアリストテレスが提唱した視覚・聴覚・触覚・味覚・嗅覚の五感が知られているが，実際には平衡感覚や内臓感覚などもある [塚原 84]．

[*2] 言語学での「様相性」の意味とは異なることに注意されたい．

[*3] パソコン通信の歴史（http://www.kogures.com/hitoshi/history/pc-tushin/index.html，2018 年 2 月アクセス）

[*4] 初めてテキストと画像を混在して表示できるようになった Web ブラウザは，1993 年に米国立スーパーコンピュータ応用研究所（NCSA）によって開発された NCSA Mosaic である．

だけでなく，画像や動画などによって，同一の内容が示されている[*5]．事件をニュース記事にする場合，文書だけでは現場の様子などがイメージしづらくても，現場で撮った写真や記者が現場でレポートする動画などを共に伝えることで，素早く把握することができる．Twitter[*6]や Facebook[*7]でも，文書のほかに関連する画像や動画も掲載することができ，相手に対して自分が伝えたいことをより明確にすることができる．本論文では，このような文書や画像，音声などのメディア形式も異なるモダリティの情報と考える．

このように，人間はマルチモーダル情報を利用することで，物事をより正確に認知している．近年では，コンピュータが人間のように複数のモダリティ情報を活用することで，より正確に予測や判断，推論等を行う試みが重要視されている．この背景には，ビッグデータの活用とロボット技術の進展などが関連している．

ビッグデータと呼ばれる大規模なデータ集合の活用は，インターネットの発展とストレージ容量の増加ともに重要視されるようになっており，「情報通信白書」 [総務 17] によると，2017 年度は「ビッグデータ利活用元年」になるとされている．ビッグデータは一般にはデータが大量に含まれることで注目されるが，データの種類が多様であることも大きな特徴である [Beyer 12][*8]．特にインターネット上で得られるビッグデータは，前述したように自然言語，画像，動画といった様々なモダリティの非構造化データが大量に含まれる．このためビッグデータを分析する上では，様々なモダリティデータをどのように分析するかという課題が生じる．人間のようにこれらを適切に活用できれば，複数のモダリティを含んだデータからより有益な情報を得ることができると期待される．

マルチモーダル情報が注目されるもう 1 つの背景として，ロボット技術の進展が挙げられる．近年は，介護ロボットや災害派遣ロボットなどへの関心が高まっており，そうしたロボットには人間のように実世界上を自由に動き回れる能力が求められている．しかし我々が住む実世界では，身の回りの様々な物や人が障害物になりうるため，ロボットは常に周囲の状況を確実に把握している必要がある．そのため，実世界上で動作させるロボットには，外界の様々な

[*5] Yahoo!ニュースは，ニュース本文と画像の他，Yahoo!ニュース動画としてテレビニュースの動画が配信されている（https://news.yahoo.co.jp/，2018 年 2 月アクセス）．

[*6] https://twitter.com

[*7] https://www.facebook.com

[*8] 文献 [Beyer 12] によると，ビッグデータとは量，速度，種類（volume, variety, and velocity）の 3 つの特徴を持つとされる．

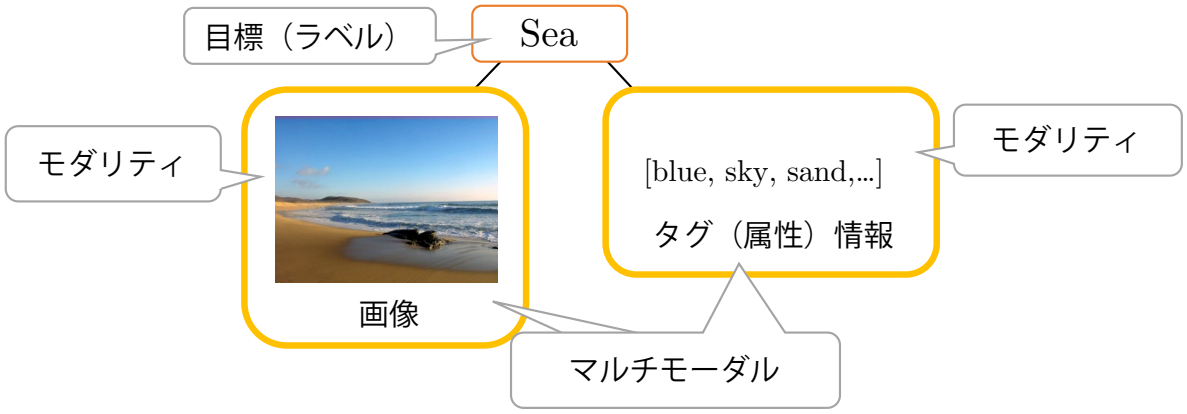


図 1.1　本論文における目標とモダリティ，マルチモーダルの関係.

種類の情報を獲得するための数多くのセンサが搭載されている．たとえば，ソフトバンクロボティクスが開発している人型ロボット Pepper の場合，マイクや RGB カメラの他，3D センサやタッチセンサなど計 27 個のセンサが搭載されている[*9]．こうしたセンサは，人間の感覚の種類と同じ役割を果たしているため，これらのセンサから得た動画，音声，角度や加速度情報，距離情報といったものはすべて異なるモダリティ情報である．ロボットは，このような得た多種の情報から現在の状況を総合的に判断し，次の行動を決定する必要がある．

ここまで，人間にとってマルチモーダル情報が重要であること，そして近年マルチモーダル情報を処理する枠組みが重要になっていることについて述べた．次節では，この枠組みを実現する方法について議論する．

ここで簡単に用語の説明をする．ある対象に対して，上記のように様々な種類の情報を取るそれぞれの枠組み，もしくはそれぞれの枠組みから得られる情報のことをマルチモーダルと呼ぶ．またその対象のカテゴリを表す情報を目標やラベルと呼ぶ．図 1.1 は，マルチモーダルデータと目標の関係を表している．これらのより形式的な定義は 3.1 節で述べる．なお，本論文ではマルチモーダルの種類を 2 つに限定して議論を進める．

[*9] Pepper 製品仕様（https://www.softbank.jp/robot/consumer/products/spec，2018 年 2 月アクセス）

1.1.2 マルチモーダル学習

機械学習は，データからパターンを自動的に抽出し，未知のパターンを用いて将来のデータを予測したり，意思決定を行う枠組みである [Murphy 12]．前述したマルチモーダルデータを活用して予測や判断を行うという試みも機械学習の問題と考えることができ，マルチモーダル機械学習（multimodal machine learning）もしくは単に**マルチモーダル学習（multimodal learning）**と呼ばれる [Ngiam 11, Lahat 15, Baltrušaitis 17]．

マルチモーダル学習の問題設定は様々考えられ，これまでも数多くの研究が行われてきた．Baltrusalitis らはこれらを表現（representation），変換（translation），アラインメント（alignment），融合（fusion），共学習（co-learning）の 5 つに分類している [Baltrušaitis 17]．

これらの問題にはそれぞれ困難な点があるが，その原因の多くは異なるモダリティ間の**異種性（異質性，heterogeneity）**にあるとされている [Karl Weiss 16, Baltrušaitis 17]．異種性の形式的な定義については 3.1 節で述べるが，画像と音声のように互いに形式が大きく異なる性質のことである．異種性を持つ 2 つのデータ集合には，次の違いが含まれる．

特徴空間の違い 各モダリティで表されるデータの形は，次元や構造が大きく異なる．たとえば，ある物体について，自然画像情報と対応するタグ情報があるとする．ただし，自然画像は行列で表される RGB カラー画像で，タグ情報は各項目について該当する／しないの 2 値の値をとるベクトル情報とする．すると，自然画像とタグ情報の空間は，実数値を取る行列と 2 値のベクトルのように大きく異なってしまう．

分布の違い 異なるモダリティは，同じ対象を表しているとしても，各モーダル集合の分布には大きな差がある．上記と同様，画像情報とタグ情報を例にとって考えると，あるタグに対応する画像は無数に考えられるが，ある画像に対応するタグの種類は非常に少ない．これは画像とタグを一対一に結び付けられないことを意味し，モダリティ間の分布に違いがあることに起因している．

これまでのマルチモーダル学習では，特徴空間の違いが注目されてきた．1.1.3 節で説明する深層学習によるアプローチも，この違いを解消することを目的としている．その一方で分布の違いについては，そもそも考慮されないか，違いはない前提で考えられてきた [Karl Weiss 16]．

これら 2 つの違いが各問題設定に及ぼす影響の度合いは，必ずしも自明ではない．しかし実際には，分布が違う状況も確かに存在する [Zhou 14]．特に変換の問題では，次元の違いだけでなく，分布の違いが大きく影響すると考えられる．モダリティ間の変換は，マルチモーダル学習の中でも主要な問題設定である．もし異なるデータ集合間の分布が異なる場合は，モダリティ間の一対一の決定論的な写像を設計できない．そのため，決定論的な関係（ルール）を作成したあとに最も近いものを近似的に選択するか [Socher 14]，ドメイン適応の手法を用いてモダリティ間の分布の違いを緩和するか [Zhou 14]，あるいは何らかの方法で決定論的でない関係を学習する必要がある．なお，特徴空間の違いが解消されたとしても，分布の違いがなくなる訳ではないことに注意されたい．たとえば，異なるモダリティ集合を任意の関数によって同じ次元の空間に写像しても，分布の違いが無くなることは保証されない [Karl Weiss 16]．

1.1.3 深層学習とマルチモーダル情報

近年，深層ニューラルネットワーク（深い階層構造をもつニューラルネットワーク）を学習器として用いる機械学習技術である**深層学習（deep learning）** [LeCun 15, Goodfellow 16] の研究が急速に進んでいる．深層学習は，入力から出力への写像の学習の過程において，ネットワークの隠れ層で入力の良い特徴量が獲得できることで注目を集めている（このことに着目して表現学習（feature learning）とも呼ばれる）．深層学習以前の特徴抽出方法は，各モダリティの性質に応じて予め人手で設計されたものを利用していた．たとえば，画像については局所鮮度勾配ヒストグラム，特に SIFT 記述子 [Lowe 04b] や HOG 記述子 [Dalal 05] がよく利用され，音声についてはメル周波数ケプストラム係数（MFCC） [Logan 00] などが使われてきた．しかし最近では，深層ニューラルネットワークによる手法に置き換わりつつある．これは，深層ニューラルネットワークによって，既存の特徴抽出器と同等もしくはそれを上回るような性能が確認されているからである．特に画像認識の分野では大きな躍進を遂げており，人間の視覚認識能力をも上回る結果が複数報告されている [Ioffe 15, He 15]

深層学習では，それぞれのモダリティの特徴空間や，性質に応じた特殊なネットワークで学習することが主流である．画像データの場合は，モダリティが格子状の形式を持ち，移動不変性があるといった仮定から，畳み込みニューラルネットワーク（CNN） [LeCun 98] が利用される．また系列データの場合は，1 次元の可変ベクトルかつ系列方向に強い依存関係があると

いう前提から，回帰結合型ニューラルネットワーク（RNN）[Gers 99] が使われる．こういった特殊なネットワークを用いることで，従来の学習器よりも高次元で複雑なデータを直接扱えるようになっている．

さらに，深層学習が従来の特徴抽出器や学習器と比べて優れている点は，学習器である深層ニューラルネットワークを自由に設計することができ，さらにネットワークの形によらず，同一の最適化方法で学習できるということである．したがって，異なるモダリティごとに，それぞれ上記のような特殊なネットワークを用いたとしても，それらを結合して end-to-end に学習することができる．

以上の背景から，近年のマルチモーダル学習では，深層ニューラルネットワークを用いる手法が主流となっている [Ngiam 11, Socher 14, Antol 15]．マルチモーダル学習の表現や融合の問題設定では，それぞれのモダリティを入力としたネットワークを用意し，各ネットワークの最終層を結合した学習器を訓練することで，全モダリティの**共有表現（joint representation）**を獲得することができる[*10]．この表現は，入力モダリティの次元や構造に依存せず，単一のモダリティよりも多くの情報を含んだ特徴量となるため，様々な問題設定でこの枠組みが利用されている [Ngiam 11, Antol 15].

しかし深層ニューラルネットワークは，決定論的な写像を学習しているため，モダリティ間の分布の違いに対処することができない．また，上記のようなネットワークを結合するアプローチでは，モダリティが欠損した場合にうまく対処できないという問題が指摘されている [Baltrušaitis 17]．さらに深層ニューラルネットワークは，関係性の方向を明示的に指定できないので，モダリティ間の明示的な関係性を記述することができない．たとえば，画像情報を補助情報として利用した文書翻訳の研究では，画像情報という大きな情報量を持つモダリティを追加したにもかかわらず，通常の翻訳の結果と比較して大きな精度改善になっていない [Caglayan 16]．これは，画像と文の情報量の違いをうまく考慮できていないためであり，翻訳文に対して画像が従属的な役割を果たしていることを，ニューラルネットワークがうまく表現できていないためとも考えられる．

このように，次元の違いの観点からマルチモーダルデータを扱う上では深層ニューラルネットワークの利用が必須となっているが，分布の違いに対処できる手法も必要である．

[*10] 詳しくは 3.2.1 節や 3.2.3 節で説明する．

1.1.4 生成モデルとマルチモーダル情報

異なる分布を持つ変数を扱う方法の 1 つとして，**生成モデル（generative models）**（もしくは確率的生成モデル（probabilistic generative models））の利用が挙げられる．生成モデルとは，変数間の関係を確率分布によって明示的にモデル化するアプローチであり[*11]，各変数の関係は，グラフィカルモデル（graphical model）を用いて記述される．特に，有向グラフィカルモデル（あるいはベイジアンネットワーク）では，有向非巡回グラフを用いて依存の方向性を明示的に記述することができる．

生成モデルによって，モダリティ間の分布の違いを明示的に考慮することができる．タグ情報から画像情報を生成する場合は，タグから画像への確率的な依存関係をモデル化することで，写像に確率的な「幅」を持たせて学習及び推論することができる．また生成モデルによるアプローチでは，表現や融合で生じる可能性のあるモダリティやラベルの欠損を自然に扱えることが知られている [Baltrušaitis 17]．

深層学習の登場以前の生成モデルを用いたマルチモーダル学習としては，トピックモデルの 1 つである pLSA をマルチモーダルデータ用に拡張したモデルがいくつかある [Monay 03, Lienhart 09, Chandrika 10, Nikolopoulos 13]．これらのモデルでは，異なるモダリティを融合したトピックを教師なしで獲得することができる[*12]．

一方で，従来の生成モデルの課題の 1 つは，自然画像のような，高次元かつ複雑なデータを直接扱えないということである．これは，生成モデルが比較的単純な確率分布で構成されており，そうした分布では表現しきれないデータは変数として取れないためである．

1.1.5 深層学習と生成モデルによるマルチモーダル学習

ここまで，マルチモーダル学習では，モダリティの違い，すなわち，異種性を考慮することが重要であることを説明した．また異種性には，特徴空間の違いだけでなく分布の違いを考慮することも重要であると指摘し，前者が深層学習で解決できることが知られている一方で，後

[*11] ただし generative adversarial network（GAN） [Goodfellow 14] に代表されるように，分布を暗黙的に定義する場合もある．

[*12] multilayer monomodal pLSA では，トピックはすべてのモダリティを統合した高レベルなトピックの他に，それより下位の各モダリティごとのトピックもモデル化されている [Lienhart 09]．

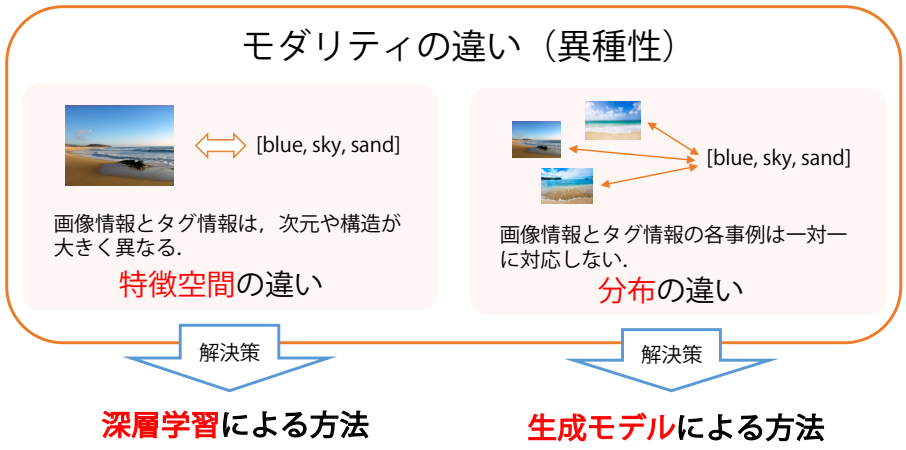


図 1.2 モダリティの異種性と，深層学習及び生成モデルによる方法の対応関係．

者は生成モデルを用いることで解消できる可能性があることを示した（図 1.2）．

近年，深層学習と生成モデルを結びつけた研究が多数提案されている．これらの研究には 2 つのアプローチがある．1 つ目は，深層ニューラルネットワークを用いて生データから特徴抽出をし，それを簡単な確率分布で表される生成モデルの入力とする方法である（**深層学習の特徴抽出 + 生成モデルのアプローチ**と呼ぶ）．2 つ目は，生成モデルの確率分布自体を深層ニューラルネットワークでモデル化する方法である．このモデルは**深層生成モデル（deep generative model）**と呼ばれ，深層学習の中でも最も急速に研究が進んでいる分野の 1 つとなっている [Kingma 13, Goodfellow 14]．

そこで本論文では，深層学習と生成モデルの両方を用いることによって，モダリティ間の異種性を考慮できる可能性に着目する．

1.2 本研究の目的

従来のマルチモーダル学習の研究では，深層学習だけでなく生成モデルの利用が，様々な問題設定において有効かどうかは検証されていなかった．これは，深層学習 + 生成モデルのモデルに対して実データが適用できるようになってから年月があまり経ってないため[*13]，そし

[*13] Kingma らと Goodfellow らによって自然画像等を学習できる深層生成モデルが提案されたのが，2014 年である [Kingma 13, Goodfellow 14]．

て特に深層生成モデルの場合，実装するには既存のライブラリでは困難であるため（3.3 節を参照）と考えられる．

そこで本研究では，マルチモーダル学習の各問題設定を実現する，深層学習と生成モデルを用いたモデルを提案することを目的とする．また深層生成モデルの実装や応用を容易にするための，簡潔かつ汎用性の高い実装が可能なライブラリを開発する．

上記の目的に対して，本研究では次の 4 つのサブ研究を行う．

1. モダリティ間を双方向に変換できる深層生成モデルの提案
2. 深層生成モデルを用いた半教師ありマルチモーダル学習モデルの提案
3. 属性と画像の関係性を明示的に記述した生成モデルによるゼロショット学習の提案
4. 深層生成モデルを実装するためのライブラリ開発

研究 1 から研究 3 は，マルチモーダル学習の各問題設定において，深層学習と生成モデルを利用した手法を提案するものである．深層学習と生成モデルの結びつけ方も研究によって異なる方法をとり，研究 1 と研究 2 では深層生成モデルによるアプローチ，研究 3 では深層学習の特徴抽出 + 生成モデルのアプローチをとる．研究 4 は，特に深層生成モデルにおいて，実装するための有効なライブラリを開発することを目標としている．

本論文は，マルチモーダル情報を扱う上で特徴空間と分布の違いに着目し，それらを両方考慮すること，すなわち，深層学習と生成モデルを用いることが重要であると主張するものである．背景で述べた通り，マルチモーダル情報の活用は様々な場面において求められており，本研究はそれらを扱うための新たな方針とモデルを提案している．

また本論文では，それら（具体的には深層生成モデルのアプローチである研究 1 と研究 2）を実装するためのライブラリの開発も行っている．これにより，本論文で提案するモデルを簡単に様々なデータに対して適用でき，さらに発展させたモデルの実装も可能になる．

1.3　本論文の構成

本論文の構成は以下の通りである．

まず 2 章では，本論文で扱う深層ニューラルネットワークと生成モデルに関する前提知識や用語について説明した後，これらを結びつける 2 つのアプローチについて説明し，そしてその

アプローチの 1 つである深層生成モデル，特に VAE についての説明を行う．

3 章では，マルチモーダルの定義を議論した後に，マルチモーダル学習の各問題設定の先行研究について説明する．その後，既存の確率モデリング言語についてまとめる．

4 章では，前提知識や関連研究を踏まえ，本研究の新規性及び意義を明らかにする．

5 章では，マルチモーダル学習における表現と変換の問題設定に取り組む深層生成モデルを提案する．本章では，まず従来の表現問題に対するアプローチを単純に深層生成モデルに拡張した joint multimodal variational autoencoder（JMVAE）を検証し，双方向にモダリティを変換する場合に，情報量の大きいモダリティを欠損させると共有表現や変換したモダリティが崩れてしまい，従来知られていた欠損値補完の手法では解決できないことを示す．この問題を解決するために，本研究では JMVAE-kl と階層的 JMVAE という手法を提案する．実験から，これらの手法によって単一モダリティを入力とした場合でも適切な共有表現が推論できること，従来の 1 方向のモデルと比較して同等以上の精度で双方向の変換ができることを確認する．

6 章では，融合，すなわち，複数のモダリティから目標ラベルを予測する問題に取り組む．本研究では，異なるモーダルデータがセットで手に入りやすいのに対して，ラベル情報の獲得は人的コストがかかるという背景から，半教師あり学習に取り組む．半教師あり学習とは，少量のラベルあり集合の他に大量のラベルなし集合がある状況で，汎化性能の高い識別モデルを学習する枠組みである．この章では深層生成モデルを用いた半教師ありマルチモーダル学習の手法として，新たに semi-supervised MVAE（SS-MVAE）と semi-supervised HMVAE（SS-HMVAE）を提案する．また半教師ありマルチモーダル学習では，テスト集合に単一のモダリティしか与えられない設定があることから，モダリティが欠損しても精度を落とさずに目標ラベルを予測する手法として，SS-HMVAE を拡張した SS-HMVAE-kl というモデルを提案する．これらの手法でモダリティの欠損が補完できること，そして単一モダリティ及びマルチモーダルにおける既存の半教師あり学習と比較して，提案手法の精度が高いことを実験で確認する．

7 章では，共学習の問題設定の 1 つとして知られるゼロショット学習について取り組む．ゼロショット学習とは，一度も学習したことのない目標カテゴリのモダリティを，他の目標カテゴリでの学習結果と，異なるモダリティの情報を補助情報として用いて予測する枠組みであ

る．本研究では，分類するモダリティとして画像，補助情報として属性を用いる属性ベースゼロショット学習に着目する．属性ベースゼロショット学習の既存研究では，属性の画像に対する分布の違い（現れやすさ）を明示的に考慮していなかった．本研究では，このような現れやすさの度合いを属性ごとの観測確率と呼び，観測確率を含めて画像や属性，ラベルの関係を記述した生成モデルを提案する．実験では，深層ニューラルネットワークから得た特徴量を利用し，モデルの妥当性の検証及び既存研究との比較実験によって，提案手法が既存手法と比較して有効性の高いモデルであることを示す．

8 章では，深層生成モデルを実装するライブラリを提案する．深層ニューラルネットワークを用いた生成モデルを記述できるライブラリは既に存在している [Salvatier 16, Tran 17a] が，それらは深層ニューラルネットワークと確率変数を同じレイヤーとして考えており，本研究で提案するようなマルチモーダル情報を持った複雑な深層生成モデルを実装するためには，モデルやネットワークが変更される度に新たに 1 から実装を行う必要があった．また，深層ニューラルネットワークによって定義された確率分布からサンプリングしたり尤度を計算したりすることは，従来の言語では困難であった．今日の深層生成モデルでは，目的関数として尤度を計算し，潜在変数や各モダリティのデータをサンプリング（推論や生成）することが重要になっている．こうした背景から，深層生成モデルに特化したライブラリ「Tars」を新たに提案する．このライブラリでは，確率分布がニューラルネットワークを隠蔽しており，確率分布のそれぞれでサンプリングや尤度計算できる．また，この章では，Tars を用いて様々な分布の深層生成モデルが学習できることを確認し，本ライブラリが上記の目的を達成していることを示す．さらに本ライブラリを使用したアプリケーションを紹介し，本ライブラリで学習した確率分布をそのまま読み込んで画像を生成したりできることを説明する．

9 章では，5 章から 8 章までの結果を踏まえて，本論文の貢献や限界を確認し，今後への課題についてまとめる．そして本研究の産業応用の可能性ついて触れ，本研究の統一構想や汎用人工知能に向けた考察について述べる．

最後に 10 章で，本論文のまとめを述べる．

第 2 章

深層学習と生成モデルに関する前提知識

本論文では，1 章で述べたように，深層学習と生成モデルの両方を用いたマルチモーダル学習の手法を提案する．そこで本章では，本論文を読み進めるために必要な深層ニューラルネットワークと生成モデルに関する前提知識や用語について説明する．

2.1 節と 2.2 節では，それぞれ深層ニューラルネットワークと生成モデルに関する前提知識や用語について説明する．2.3 節では，深層ニューラルネットワークと生成モデルを結びつける 2 つのアプローチを説明する．そして 2.4 節で，2 つのアプローチのうちの 1 つである深層生成モデル，特に VAE について説明する．

2.1　深層ニューラルネットワーク

順伝播型ニューラルネットワーク（feedforward neural network），もしくは多層パーセプトロン（multi-layer perceptron，MLP）は，（人工）ニューラルネットワーク（artificial neural network）を階層的に繋げた構造を持つ．順伝播型ニューラルネットワークの層の数が一定数以上ある場合は，特に深層ニューラルネットワーク（deep neural network）と呼ばれる．この深層ニューラルネットワークを機械学習の学習器（識別器）として用いるアプローチの総称が深層学習（deep learning）である．ここでは，深層ニューラルネットワークの構造と学習について簡単に説明する．

2.1.1 深層ニューラルネットワークの構造

入力ベクトル $\boldsymbol{x} = [x_1, ..., x_I]^T$ を受け取り，スカラー値 h を出力する情報処理を

$$\begin{aligned} h &= f(\boldsymbol{x}; \boldsymbol{\theta}) \\ &= g(\sum_i^I w_i x_i + b) = g(\boldsymbol{w}^T \boldsymbol{x} + b) \end{aligned} \tag{2.1}$$

とする．ただし，$\boldsymbol{\theta}$ は $\boldsymbol{w} = [w_1, ..., w_I]^T \in \mathcal{R}^I$ と $b \in \mathcal{R}$ で構成され，それぞれ重みベクトルとバイアスパラメータと呼ぶ．また，これらをまとめた $\boldsymbol{\theta}$ は重みやパラメータと呼ばれる．g は任意の非線形関数であり，活性化関数（activation function）と呼ばれる．活性化関数には通常，双曲線正接関数（hyperbolic tangent function，tanh）やソフトプラス関数（softplus function），正規化線形関数（rectified linear unit，ReLU）などが用いられる．

式 (2.1) の情報処理は，脳神経系のニューロンの形式的なモデル化となっている [McCulloch 43]．ニューラルネットワークでは，これをネットワークを構成する 1 つのユニットと考える．

次に，複数のユニットを考え，入力ベクトルを共有して，それぞれの重みからスカラー値 h_j が出力されると考える．すなわち，

$$h_j = g(\boldsymbol{w}_j^T \boldsymbol{x} + b_j).$$

複数のユニットの出力をベクトル $\boldsymbol{h} = [h_1, ..., h_J]^T$ で表すと

$$\boldsymbol{h} = g(\boldsymbol{W}^T \boldsymbol{x} + \boldsymbol{b}) \tag{2.2}$$

となる．ただし，$\boldsymbol{W} = [\boldsymbol{w}_1, ..., \boldsymbol{w}_J]$ 及び $\boldsymbol{b} = [b_1, ..., b_J]^T$ である．順伝播型ニューラルネットワークでは，式 (2.2) をネットワークの層（layer）と呼ぶ．すなわち，式 (2.2) のモデルは 1 層のニューラルネットワークである[*1]．

順伝播型ニューラルネットワークでは，ある層の出力が次の層の入力となる形でネットワークが構成される．たとえば，l 層目の出力 $\boldsymbol{h}^{(l)}$ は，パラメータ，活性化関数をそれぞれ $\boldsymbol{h}^{(l)}$，

[*1] 従来はこのネットワークは 2 層のニューラルネットワークとして扱われてきたが，近年は Keras [Chollet 15] などの深層学習ライブラリの影響から，式 (2.2) を層の単位として考えることが多い．

$\boldsymbol{\theta}^{(l)} = \{\boldsymbol{W}^{(l)}, \boldsymbol{b}^{(l)}\}$，$g^{(l)}$ とすると，

$$\begin{aligned}\boldsymbol{h}^{(l)} &= f^{(l)}(\boldsymbol{h}^{(l-1)}; \boldsymbol{\theta}^{(l)}) \\ &= g^{(l)}(\boldsymbol{W}^{T(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \\ &= g^{(l)}(\boldsymbol{W}^{T(l)} f^{(l-1)}(\boldsymbol{h}^{(l-2)}; \boldsymbol{\theta}^{(l-1)}) + \boldsymbol{b}^{(l)}) \\ &= g^{(l)}(\boldsymbol{W}^{T(l)} g^{(l-1)}(\boldsymbol{W}^{T(l-1)}\boldsymbol{h}^{(l-2)} + \boldsymbol{b}^{(l-1)}) + \boldsymbol{b}^{(l)})\end{aligned}$$

となり，l 層目だけでなく，$l-1$ 層目のパラメータに依存する．$l-1$ 層目の出力も $l-2$ 層目のパラメータに依存するので，結局 l 層目の出力は，それ以下のすべての層（$1 \sim l$ 層）のパラメータに依存することになる．

つまり，ネットワークが L 層で構成されている場合，最終層の L 層目の出力は $\boldsymbol{h}^{(L)} = f^{(L)}(f^{(L-1)}...(f^{(l)}...(f^{(1)}(\boldsymbol{x}; \boldsymbol{\theta}^{(1)})...; \boldsymbol{\theta}^{(l)})...; \boldsymbol{\theta}^{(L-1)}); \boldsymbol{\theta}^{(L)})$ のように各層の連鎖的な構造になる．このとき，$f^{(1)}$ は入力層（input layer），$f^{(L)}$ は出力層（output layer）と呼ばれる．また，それ以外の中間の層は隠れ層（hidden layer）と呼ばれる．隠れ層の数が一定数以上になった順伝播型ニューラルネットワークが，深層ニューラルネットワークと呼ばれる．

順伝播型ニューラルネットワークは，各層のパラメータをまとめて $\boldsymbol{\theta}$ とすることで[*2]，最終層（L 層）の出力 $\hat{\boldsymbol{y}} = \boldsymbol{h}^{(L)}$ を $\hat{\boldsymbol{y}} = f(\boldsymbol{x}; \boldsymbol{\theta})$ と簡潔に書くことができる[*3]．このことから，深層ニューラルネットワークはパラメータ $\boldsymbol{\theta}$ を変更することで出力値が変化する関数近似器とみなすことができる．ここまでの説明を図 2.1 にまとめる．

その他，入力 $\boldsymbol{x}$ の特徴空間や性質に応じて，上記の形とは異なる深層ニューラルネットワークが使われる．入力が画像データの場合は，モダリティが格子状の形式を持ち，移動不変性があるといった仮定から，畳み込みニューラルネットワーク（convolutional neural network, CNN）[LeCun 98] が利用される．また系列データの場合は，1 次元の可変ベクトルかつ系列方向に強い依存関係があるという前提から，回帰結合型ニューラルネットワーク（recurrent neural network, RNN）[Gers 99] が使われる．これらのネットワークの構造の説明については，本章では省略する．

[*2] 以降も $\boldsymbol{\theta}$ は，特に層の添字を明記していない場合，すべての層のパラメータを表す．

[*3] 活性化関数はここでは引数として明記していないが，一般的には $L-1$ 層以下の層では同じ形が使われることが多い．また，L 層の活性化関数は，出力の形式に応じて選択する．

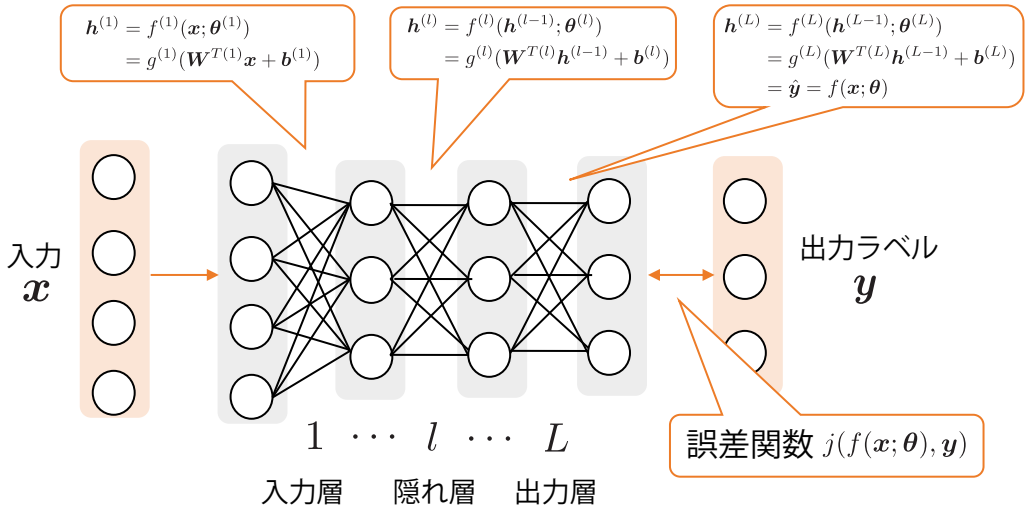


図 2.1　深層ニューラルネットワークの構造と表記のまとめ.

2.1.2　学習の目的

入力 $\boldsymbol{x}$ と目標となる出力ラベル $\boldsymbol{y}$ のペア $(\boldsymbol{x},\boldsymbol{y})$ の，真のデータ分布 $p^*(\boldsymbol{x},\boldsymbol{y})$ を考える．ニューラルネットワークを機械学習における学習器とみなすと，学習の目的は，真のデータ分布 $p^*(\boldsymbol{x},\boldsymbol{y})$ の下で，任意の $\boldsymbol{x}$ を入力としたときの出力 $\hat{\boldsymbol{y}} = f(\boldsymbol{x};\boldsymbol{\theta})$ が，対応する目標ラベル $\boldsymbol{y}$ と近くなるように，パラメータ $\boldsymbol{\theta}$ を調節することである．これは，入力 $\boldsymbol{x}$ に対する出力 $\hat{\boldsymbol{y}} = f(\boldsymbol{x};\boldsymbol{\theta})$ とラベル $\boldsymbol{y}$ の誤差を，任意の関数 $j(f(\boldsymbol{x};\boldsymbol{\theta}),\boldsymbol{y})$ で表し，$\boldsymbol{\theta}$ について最小化することで実現される．この関数 $j(f(\boldsymbol{x};\boldsymbol{\theta}),\boldsymbol{y})$ は，機械学習の分野では，誤差関数（error function）や損失関数（loss function），目的関数（objective function）と呼ばれる．

すなわち，求めるパラメータの値 $\boldsymbol{\theta}^*$ は

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} E_{p^*(\boldsymbol{x},\boldsymbol{y})}[j(f(\boldsymbol{x};\boldsymbol{\theta}),\boldsymbol{y})] \tag{2.3}$$

となる．ここで，$E_{p(\boldsymbol{x})}[f(\boldsymbol{x})]$ は確率変数 $\boldsymbol{x} \sim p(\boldsymbol{x})$ における $f(\boldsymbol{x})$ の期待値である．したがって，真のデータ分布 $p^*(\boldsymbol{x},\boldsymbol{y})$ におけるニューラルネットワークの学習は，式 (2.3) の最適化問題となる．

実際には真のデータ分布 $p^*(\boldsymbol{x},\boldsymbol{y})$ は未知なので，直接式 (2.3) の期待値を求めることができない．そこで，真のデータ分布からのサンプル $\{(\boldsymbol{x}_1,\boldsymbol{y}_1),...,(\boldsymbol{x}_N,\boldsymbol{y}_N)\}$ によって，この期待

値を近似する．すなわち，最適化問題は次のように変わる．

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \sum_{n=1}^{N} j(f(\boldsymbol{x}_n; \boldsymbol{\theta}), \boldsymbol{y}_n). \tag{2.4}$$

サンプル集合 $\{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_N, \boldsymbol{y}_N)\}$ は訓練集合（training set）と呼ばれ，$J(\boldsymbol{\theta}) = \sum_{n=1}^{N} j(f(\boldsymbol{x}_n; \boldsymbol{\theta}), \boldsymbol{y}_n)$ は訓練誤差（training error）と呼ばれる．

本来の学習の目的は，式 (2.3)，すなわち，真のデータ分布における期待値の最適化であるため，式 (2.4) の最適化では，訓練集合だけに適合する $\boldsymbol{\theta}^*$ が学習される可能性がある．この現象は，過学習と呼ばれる．

これを防ぐため，通常は，訓練（すなわち，式 (2.4) の最適化）が終わった後，訓練集合とは別にデータ分布のサンプル $\{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_M, \boldsymbol{y}_M)\}$ を用意し，これによる誤差 $\sum_{m=1}^{M} j(f(\boldsymbol{x}_m; \boldsymbol{\theta}), \boldsymbol{y}_m)$ を評価することによって，訓練集合について過学習していないかを確認する．このサンプル集合 $\{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_M, \boldsymbol{y}_M)\}$ はテスト集合（test set），テスト集合における誤差 $\sum_{m=1}^{M} j(f(\boldsymbol{x}_m; \boldsymbol{\theta}), \boldsymbol{y}_m)$ はテスト誤差（test error）や汎化誤差（generalization error）と呼ばれる．

したがって，機械学習の目的は，この汎化誤差を下げるように訓練集合を用いて学習することである．なお，訓練時にはテスト集合は手元にないため，訓練時に汎化誤差を評価するため，訓練集合の中でさらに訓練集合とテスト集合に分けて学習することがある．このときのテスト集合は，未知のテスト集合と区別して検証用集合（validation set）と呼ばれる．

以上の枠組みは，入力と対応するラベルがデータ集合（dataset）として与えられていることから，**教師あり学習（supervised learning）**と呼ばれる．一方で，入力 $\boldsymbol{x}$ の集合から学習する枠組みを**教師なし学習（unsupervised learning）**と呼ぶ．また，少数の入力 $\boldsymbol{x}$ と出力 $\boldsymbol{y}$ のペア集合と大量の入力 $\boldsymbol{x}$ の集合が与えられる場合は，**半教師あり学習（semi-supervised learning）**と呼ぶ．

2.1.3　深層ニューラルネットワークの学習方法

深層ニューラルネットワークの場合，訓練誤差は通常パラメータに関して凸関数にならないため，式 (2.4) を解析的に解くことは困難である．そのため，誤差関数の勾配を求めて，その

方向にパラメータを更新することを考える：

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}. \tag{2.5}$$

ただし η は 1 回の更新でのパラメータの変化量を決める学習率である．これを勾配降下法（gradient decent method）という．

式 (2.7) では，すべてのネットワークのパラメータにおける勾配を計算する必要があるが，誤差の計算は出力層で行なっているため，そのままでは隠れ層での勾配を計算することができない（信用割当問題）．順伝播型ニューラルネットワークでは，合成関数の微分法を利用して，出力層で計算した誤差を入力層に向かって伝播していく誤差逆伝播法（backpropagation method）によって隠れ層での勾配を計算している．

式 (2.7) では全訓練集合における誤差 $J(\boldsymbol{\theta}) = \sum_{n=1}^{N} j(f(\boldsymbol{x}_n; \boldsymbol{\theta}), \boldsymbol{y}_n)$ について更新しているが，深層学習では，訓練集合の一部を毎回ランダムに選択し（ミニバッチと呼ばれる），それについて誤差を求めて更新する場合が多い．この方法を確率的勾配降下法（stochastic gradient decent, SGD）と呼ぶ．また，SGD で 1 つの訓練集合を何回繰り返して訓練するかをエポック数と呼ぶ．

上記の SGD の他にも，慣性項を追加したモメンタム法 [Rumelhart 88] や，Adadelta [Zeiler 12]，RMSprop [Tieleman 12]，Adam [Kingma 14b] など，様々な最適化アルゴリズムが提案されている．

2.1.4 学習の工夫

深層ニューラルネットワークは，層が深くなるにつれて学習が進まなくなることが知られていた．これを解決するための様々な手法が提案されている．

ニューラルネットワークの重みの初期化は，学習の進み方や最終的な精度に大きく影響する[*4]．通常はガウス分布や一様分布で初期化されるが，分布のスケールによって各層の出力の値が大きく変化する．そのため，Xavier [Glorot 10] や He [He 15] と呼ばれる初期化のトリックが使われる．本論文でも，特に言及していない場合は，深層ニューラルネットワークの初期化に Xavier を採用している．

[*4] バイアスパラメータは 0 で初期化することが多い．

ニューラルネットワークでは，データの各要素の扱いに偏りをなくすため，データに対して予め標準化（standardization，データ集合の平均を 0，分散を 1 にする）という処理をすることが多い．しかし，ニューラルネットワークの学習が進むにつれて，各層の出力の分布がずれてしまい，層が深くなるほど収束が遅くなるという問題がある．これを解決するために，訓練中に各層の出力（ミニバッチ）を標準化するというバッチ正規化（batch normalization）[Ioffe 15] という手法が提案されている．

また，テスト集合における汎化誤差を抑える方法も複数提案されている．たとえば，ドロップアウト（dropout） [Srivastava 14] は，訓練中にユニットを確率的に消すことで正則化（regularization）の役割を果たす手法である．

2.2　生成モデル

この節では，生成モデルの概要と学習，グラフィカルモデル，そして潜在変数を含んだ生成モデルの学習について説明する．

2.2.1　生成モデルの枠組みと学習

2.1 節では，データからラベルへの写像を学習していたが，ここではデータ分布自体を学習することを考える．

データ $\boldsymbol{x}$ における真のデータ分布が $p^*(\boldsymbol{x})$ で表されるとする．このデータ分布は直接求めることができないので，代わりに確率モデル $p_{\boldsymbol{\theta}}(\boldsymbol{x})$[*5]を考え，モデルのパラメータ $\boldsymbol{\theta}$ を学習することで真のデータ分布を近似する．このとき，$p_{\boldsymbol{\theta}}(\boldsymbol{x})$ はデータ $\boldsymbol{x}$ の生成過程を確率的にモデル化したものであるため，**生成モデル（generative model）**と呼ばれる．一方で，2.1 節のように入力から出力への写像を学習するアプローチを識別モデル（discriminative model）という．

データの生成モデルを学習する利点の 1 つに，未知のデータを 1 からサンプリングできるということが挙げられる．これは，学習した生成モデルがデータ分布の構造全体を近似しているためである[*6]．その他にも，密度推定や欠損値補完など，入出力の写像のみを学習している識

[*5] $p(\boldsymbol{x}|\boldsymbol{\theta})$ とも書く．
[*6] 生成モデルがデータ分布を適切に近似できるかどうかは，最適化手法や，生成モデルのモデル化方法，そして

別モデルでは不可能なことも，生成モデルによって実現できる．

生成モデルを学習するためには，真のデータ分布とどれくらい近いのか，すなわち，どれだけ適切に近似できているのかを計測することが必要である．一般的には，生成モデル $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ とデータ分布 $p^*(\boldsymbol{x})$ との「近さ」は，カルバック・ライブラー（Kullback-Leibler，KL）ダイバージェンスを用いて，

$$D_{KL}(p^*(\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{x})) \tag{2.6}$$

のように求める．

したがって，生成モデルの学習は，式 (2.6) が最小になるようなパラメータ $\boldsymbol{\theta}$ を求める最適化問題となる．これは次のように書き換えることができる．

$$\begin{aligned}\boldsymbol{\theta}^* &= \arg\min_{\boldsymbol{\theta}} D_{KL}(p^*(\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{x})) \\ &= \arg\max_{\boldsymbol{\theta}} E_{p^*(\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x})].\end{aligned} \tag{2.7}$$

式 (2.7) は，式 (2.3) と同様，真のデータ分布における期待値となっているので直接計算できない．そのため，データ分布からのサンプルを訓練集合 $\mathcal{D} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ として式 (2.7) の期待値を近似する．

$$\begin{aligned}\boldsymbol{\theta}^* &= \arg\max_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_n) \\ &= \arg\max_{\boldsymbol{\theta}} \log \prod_{n=1}^{N} p_{\boldsymbol{\theta}}(\boldsymbol{x}_n) \\ &= \arg\max_{\boldsymbol{\theta}} \log p(\mathcal{D}|\boldsymbol{\theta}) \equiv \arg\max_{\boldsymbol{\theta}} \log L(\boldsymbol{\theta}).\end{aligned} \tag{2.8}$$

ただし，$L(\boldsymbol{\theta}) = p(\mathcal{D}|\boldsymbol{\theta})$ は $\boldsymbol{\theta}$ をパラメータとする尤度関数（likelihood function）である．

このように，尤度関数を最大化するようなパラメータの値を推定する方法を，最尤推定（maximum likelihood estimation）という[*7]．

ここまで，生成モデルのパラメータ $\boldsymbol{\theta}$ について分布を仮定しなかったが，パラメータを確率変数とみなし，任意の確率分布によって $\boldsymbol{\theta} \sim p(\boldsymbol{\theta}|\mu)$ のように生成されると考える．ただし，μ

生成モデルとデータ分布の距離を測る尺度などに依存する．

[*7] 通常は，式 (2.8) のように尤度関数の対数をとった対数尤度関数（log-likelihood function）を最大化する．

はパラメータの確率分布を制御するパラメータで，ハイパーパラメータ（hyper paramater）と呼ばれる．

訓練集合 $\mathcal{D}$ が与えられた下でのパラメータ $\boldsymbol{\theta}$ の分布 $p(\boldsymbol{\theta}|\mathcal{D},\mu)$ は，ベイズの定理（Bayes' theorem）より

$$\begin{aligned} p(\boldsymbol{\theta}|\mathcal{D},\mu) &= \frac{p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mu)}{p(\mathcal{D})} \\ &\propto p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mu) \end{aligned}$$

となる．この式は，パラメータの分布 $p(\boldsymbol{\theta}|\mu)$ が訓練集合 $\mathcal{D}$ を観測することによって $p(\boldsymbol{\theta}|\mathcal{D},\mu)$ に変化することを表している．したがって $p(\boldsymbol{\theta}|\mu)$ は事前分布（prior distribution），$p(\boldsymbol{\theta}|\mathcal{D},\mu)$ は事後分布（posterior distribution）と呼ばれる．

事後分布を最大化する $\boldsymbol{\theta}$ を推定する場合は，最大事後確率推定（maximum a posteriori estimation，MAP 推定）と呼ばれる．対数をとった事後分布について最大化すると，

$$\begin{aligned} \boldsymbol{\theta}^* &= \arg\max_{\boldsymbol{\theta}} \log p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mu) \\ &= \arg\max_{\boldsymbol{\theta}} [\log L(\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}|\mu)] \end{aligned} \tag{2.9}$$

となり，式 (2.8) の最適化問題と比較すると，$\log p(\boldsymbol{\theta}|\mu)$ という項が加わった形となっている．この項はパラメータ $\boldsymbol{\theta}$ の過学習を防ぐ正則化項の役割を果たしているとみなせる．

2.2.2　グラフィカルモデル

生成モデルでは，データの生成過程を記述するために複数の確率変数と確率分布が使われる．式 (2.9) で学習する生成モデルでは，$\boldsymbol{x}$ と $\boldsymbol{\theta}$ の 2 つの確率変数が使われている．こうした複数の確率変数の関係性（特に独立性）を記述するモデルがグラフィカルモデル（graphical model）である．

グラフィカルモデルには，変数間の関係を有向非巡回グラフで表す方法と無向グラフで表す方法がある．前者はベイジアンネットワークと呼ばれ，後者はマルコフ確率場やマルコフネットワークと呼ばれる．本論文では，グラフィカルモデルと言及した場合は，有向モデルであるベイジアンネットワークを指す．

グラフィカルモデルは，統計と機械学習の両方で使われる重要なモデルである．統計の分野では，グラフィカルモデルは多変量データの相互関係を解析するために使われることが

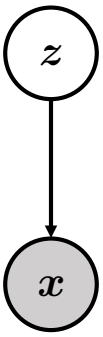


図 2.2　1 つの潜在変数を含む生成モデル.

多く，そのような統計的方法のことを「グラフィカルモデリング」という [宮川 97]．グラフィカルモデリングの例としては，多変量データの相関関係を構造学習するグラフィカル Lasso [Friedman 08] などがある．機械学習では，グラフィカルモデルは，単に生成モデルでモデル化した変数間の関係をグラフ構造として記述するための「言語」として使われる．

2.2.3　潜在変数を含んだ生成モデルの学習

入力データにおける生成モデルをモデル化するとき，入力データの確率変数やパラメータの他に，データには現れない何らかの変数を仮定することがある．このような変数を潜在変数（latent variable）と呼ぶ．それに対して，入力データのような実際に値が観測できる変数は観測変数（observable variable）と呼ぶ.

潜在変数 $\boldsymbol{z}$ を含んだ生成モデルの学習について考える．ここでは，各変数の生成過程を

$$\begin{aligned}\boldsymbol{z} &\sim p_{\boldsymbol{\theta}}(\boldsymbol{z}),\\ \boldsymbol{x} &\sim p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\end{aligned}$$

とする．この生成過程をグラフィカルモデルで記述すると，図 2.2.3 のようになる．なお，図 2.2.3 の中で，白丸が潜在変数，黒丸が観測変数を表している.

この生成モデルのすべての変数における同時分布（joint distribution）は，

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}) = p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p_{\boldsymbol{\theta}}(\boldsymbol{z}) \tag{2.10}$$

となる．

式 (2.8) のように，観測変数における対数尤度 $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ について最大化したいが，そのためには，潜在変数について積分した尤度 $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ を求める必要がある．

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}. \tag{2.11}$$

このように，積分によって特定の変数を消去する操作を周辺化（marginalization）といい，$p_{\boldsymbol{\theta}}(\boldsymbol{x})$ を周辺尤度（marginal distribution）と呼ぶ[*8]．

一般に，この周辺化の計算は困難になることが多い．そのため，次の式で表される $\mathcal{L}(\boldsymbol{x}; q, \boldsymbol{\theta})$ を計算することを考える．

$$\mathcal{L}(\boldsymbol{x}; q, \boldsymbol{\theta}) = \int q(\boldsymbol{z}) \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} d\boldsymbol{z}. \tag{2.12}$$

ここで $q(\boldsymbol{z})$ は任意の分布である．

式 (2.11) と式 (2.12) から，$\mathcal{L}(\boldsymbol{x}; q, \boldsymbol{\theta})$ は対数周辺尤度 $\log p_{\boldsymbol{\theta}}(\boldsymbol{x})$ と次のような関係があることがわかる．

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \mathcal{L}(\boldsymbol{x}; q, \boldsymbol{\theta}) + D_{KL}(q(\boldsymbol{z}) || p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})). \tag{2.13}$$

$D_{KL}(q(\boldsymbol{z}) || p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}))$ は常に正となるため，$\mathcal{L}(\boldsymbol{x}; q, \boldsymbol{\theta})$ は必ず周辺尤度の下界を抑える．そのため，$\mathcal{L}(\boldsymbol{x}; q, \boldsymbol{\theta})$ は対数周辺尤度の変分下界（variational lower bound）もしくはエビデンス下界（evidence lower bound，ELBO）と呼ばれる．また，負の変分自由エネルギー（variational free energy）と呼ばれることもある．

式 (2.13) からわかるように，変分下界は，$\boldsymbol{\theta}$ と q をパラメータとしてもつ．したがって，$\boldsymbol{\theta}$ についての下界の最大化と，q についての KL ダイバージェンスの最小化（すなわち，$q(\boldsymbol{z}) = p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$）を繰り返すことで周辺尤度を最大化することができる．これは，EM アルゴリズムと呼ばれる．一方，$p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$ が解析的に求まらない場合は，$q(\boldsymbol{z})$ の分布族を制限することで（平均場近似という）最適化を行う．この方法を変分推論（variational inference）という．

[*8] エビデンス（evidence）とも呼ばれる．また物理学では分配関数（partition function）と呼ばれる．

2.3　深層ニューラルネットワークと生成モデルを結びつけるアプローチ

生成モデルは確率分布でモデル化されるので，次元が大きく構造が複雑なデータを直接入力としてとることができない．そこで近年，深層学習と生成モデルを結びつける研究が行われている．この方法には，大きく分けて 2 つある．

1 つ目は，深層ニューラルネットワークを用いて生データから特徴抽出（feature extraction）をし，それを簡単な確率分布で表される生成モデルの入力として学習する方法である．1 章で説明したように，深層ニューラルネットワークは，隠れ層で入力の良い表現が獲得できることが知られている．「良い表現」の基準としては，情報量，独立性，説明性，スパース性，不変性，ロバスト性，平滑性などが挙げられる [麻生 15][*9]．深層学習によって得られる特徴量はこれらの基準を満たすと考えられ，確率分布の入力として直接扱うことができる．これを**深層学習の特徴抽出 + 生成モデルのアプローチ**と呼ぶことにする．

2 つ目は，生成モデルの確率分布自体を深層ニューラルネットワークでパラメータ化（モデル化）する方法である．これが**深層生成モデル（deep generative model）によるアプローチ**である．

これらの違いを表したのが図 2.3 である[*10]．深層学習の特徴抽出 + 生成モデルによるアプローチは，確率変数に意味を持たせ，それらの複雑な関係性を明示的に記述することに優れている．一方深層生成モデルによるアプローチは，確率分布を直接ニューラルネットワークでパラメータ化するため，高次元で複雑な構造のデータを生成することができる．

2.4 節では，2 つ目のアプローチで利用される深層生成モデルについて説明する．

2.4　深層生成モデル

深層生成モデルは，深層学習と生成モデルを組み合わせる方法の 1 つで，深層ニューラルネットワークによって直接生成モデルの確率分布を定義する．

[*9] 良い表現とはなにか，については文献 [Bengio 13a] や文献 [Goodfellow 16] の 13 章でも議論されている．
[*10] 本論文では有向モデルのみを考えるので，ここでも有向グラフで描いている．

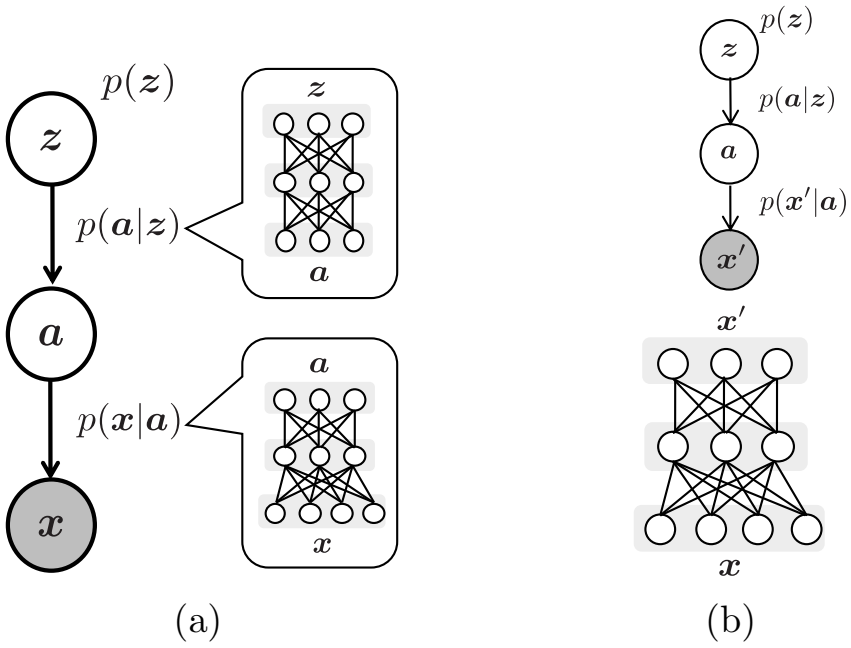


図 2.3　深層ニューラルネットワークと生成モデルを結びつけるアプローチの違い．(a) 深層生成モデルによるアプローチ，(b) 深層学習の特徴抽出 + 生成モデルによるアプローチ．

深層生成モデルとしては，これまで無向グラフの deep bolzmann machine（DBM）[Salakhutdinov 09] などが提案されていた．しかし，DBM の学習則は MCMC 法に基づくので，これらのモデルは自然画像のような高次元のデータを入力として学習できないという課題があった．

近年，高次元で複雑なデータを入力に取れる深層生成モデルが提案されている．その代表的な手法が variational autoencoder（VAE） [Kingma 13, Rezende 14] と generative adversarial network（GAN） [Goodfellow 14] である．これらの手法は，いずれも通常の深層ニューラルネットワークと同様に SGD を用いて学習することができる．VAE と GAN の大きな違いは，VAE は明示的に生成モデルの分布を仮定して学習するのに対して，GAN では生成モデルの分布の形は暗黙的にして学習することができる．本論文では深層生成モデルとしては VAE のみを扱うので，以下 VAE について説明する．

2.4.1 Variational autoencoder (VAE)

図 2.2.3 のように，1 つの潜在変数と観測変数を含む生成モデルを考える．また，これらの生成過程を $\boldsymbol{z} \sim p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ 及び $\boldsymbol{x} \sim p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ とする．

2.2.3 節で説明したように，周辺尤度を直接最大化できないので，変分下界 $\mathcal{L}(\boldsymbol{x}; q, \boldsymbol{\theta}) = \int q(\boldsymbol{z}) \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})}{q(\boldsymbol{z})} d\boldsymbol{z}$ を最大化することを考える．

ここで，$q(\boldsymbol{z})$ をモデルパラメータ $\boldsymbol{\phi}$ を用いて $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ のように $\boldsymbol{x}$ から $\boldsymbol{z}$ への確率的な写像で近似する．すると変分下界は

$$\begin{aligned} \mathcal{L}(\boldsymbol{x}; q, \boldsymbol{\theta}) &= \mathcal{L}(\boldsymbol{x}; \boldsymbol{\phi}, \boldsymbol{\theta}) \\ &= \int q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})} d\boldsymbol{z} \end{aligned} \tag{2.14}$$

となる．このように $q(\boldsymbol{z})$ を $\boldsymbol{x}$ から $\boldsymbol{z}$ への写像で近似する方法は，学習による変分推論（learned variational inference）と呼ばれる．また，$q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ は事後分布 $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$ を近似しているとみなせるので，近似分布と呼ばれる．

この変分下界は，さらに次のように式変形できる．

$$\mathcal{L}(\boldsymbol{x}; \boldsymbol{\phi}, \boldsymbol{\theta}) = -D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})) + E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})]. \tag{2.15}$$

これを目的関数として，$\boldsymbol{\phi}$ と $\boldsymbol{\theta}$ について同時に下界を最大化することで，周辺尤度を最大化することができる．それぞれの分布について，$q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ が入力から潜在変数への**エンコーダ（encoder）**，$p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ が潜在変数から入力への**デコーダ（decoder）**とみなせるので，このモデルは変分オートエンコーダ（variational autoencoder，VAE）[Kingma 13, Rezende 14] と呼ばれる．式 (2.15) において，第 1 項は正則化項，第 2 項は負の再構成誤差を表している．

次に，エンコーダとデコーダを深層ニューラルネットワークで表現する方法について説明する．エンコーダ $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ をガウス分布とすると，次のように深層ニューラルネットワークでパラメータ化できる．

$$\begin{aligned} q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) &= \mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma}^2)), \\ \boldsymbol{\mu} &= f_{\boldsymbol{\mu}}(f_{\mathrm{MLP}}(\boldsymbol{x})), \\ \boldsymbol{\sigma}^2 &= \mathrm{Softplus}(f_{\boldsymbol{\sigma}^2}(f_{\mathrm{MLP}}(\boldsymbol{x}))). \end{aligned} \tag{2.16}$$

ただし，$f_{\boldsymbol{\mu}}$ と $f_{\boldsymbol{\sigma}^2}$ はそれぞれ線形の単層ニューラルネットワーク，$f_{\mathrm{MLP}}(\boldsymbol{x})$ は $\boldsymbol{x}$ を入力とす

る任意の層数を持つ深層ニューラルネットワークを表す．また，Softplus はベクトルの各要素に対してソフトプラス関数を活性化関数として適用することを意味する．

デコーダ $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ については，$\boldsymbol{x}$ の各要素が実数値を取るときはガウス分布として式 (2.16) と同様に深層ニューラルネットワークでパラメータ化することができる．$\boldsymbol{x}$ の各要素がそれぞれ独立に 2 値をとる場合はベルヌーイ分布とし，次のようにパラメータ化できる．

$$\begin{aligned} p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) &= \mathrm{Bern}(\boldsymbol{x};\boldsymbol{\mu}), \\ \boldsymbol{\mu} &= \mathrm{Sigmoid}(f_{\boldsymbol{\mu}}(f_{\mathrm{MLP}}(\boldsymbol{z}))). \end{aligned} \tag{2.17}$$

ただし Sigmoid はシグモイド関数とする．2 値の場合でも，$\boldsymbol{x}$ が one-hot（1 つの要素のみが 1 で残りは 0）のときはカテゴリ分布とし，次のようにパラメータ化できる．

$$\begin{aligned} p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) &= \mathrm{Cat}(\boldsymbol{x};\boldsymbol{\mu}), \\ \boldsymbol{\mu} &= \mathrm{Softmax}(f_{\boldsymbol{\mu}}(f_{\mathrm{MLP}}(\boldsymbol{z}))). \end{aligned} \tag{2.18}$$

ただし Softmax はソフトマックス関数である．

VAE を，通常の深層ニューラルネットワークと同様 SGD などで学習するためには，パラメータ $\boldsymbol{\theta}, \boldsymbol{\phi}$[*11]について下界 $\mathcal{L}(\boldsymbol{x})$ の勾配を計算する必要がある．しかし，式 (2.15) における負の再構成誤差項は，パラメータ $\boldsymbol{\phi}$ をもつエンコーダにおける期待値になっている．したがって，$\boldsymbol{\phi}$ の勾配が期待値の中に入らないため，直接勾配を計算することができない．

$\boldsymbol{z} \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ をガウス分布とすると，$\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$（ただし $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$）のように再パラメータ化（reparameterize）できる．すると，式 (2.15) の負の再構成誤差項（第 2 項）の $\boldsymbol{\theta}$ と $\boldsymbol{\phi}$ に関する勾配は，

$$\begin{aligned} \nabla_{\boldsymbol{\theta},\boldsymbol{\phi}} E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] &= \nabla_{\boldsymbol{\theta},\boldsymbol{\phi}} E_{\mathcal{N}(\boldsymbol{\epsilon};\mathbf{0},\boldsymbol{I})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon})] \\ &= E_{\mathcal{N}(\boldsymbol{\epsilon};\mathbf{0},\boldsymbol{I})}[\nabla_{\boldsymbol{\theta},\boldsymbol{\phi}} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon})] \end{aligned}$$

のように計算することができる．この手法を再パラメータ化トリック（reparameterization trick） [Kingma 13, Rezende 14] という．なお，再パラメータ化した式 (2.15) の第 2 項は，モンテカルロサンプリングによって

$$E_{\mathcal{N}(\boldsymbol{\epsilon};\mathbf{0},\boldsymbol{I})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon})] \simeq \frac{1}{L} \sum_{l=1}^{L} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}^{(l)})$$

[*11] 確率分布を深層ニューラルネットワークでパラメータ化したので，これらは深層ニューラルネットワークのパラメータである．

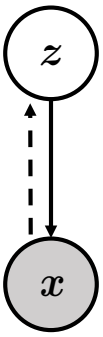


図 2.4　VAE のグラフィカルモデル.

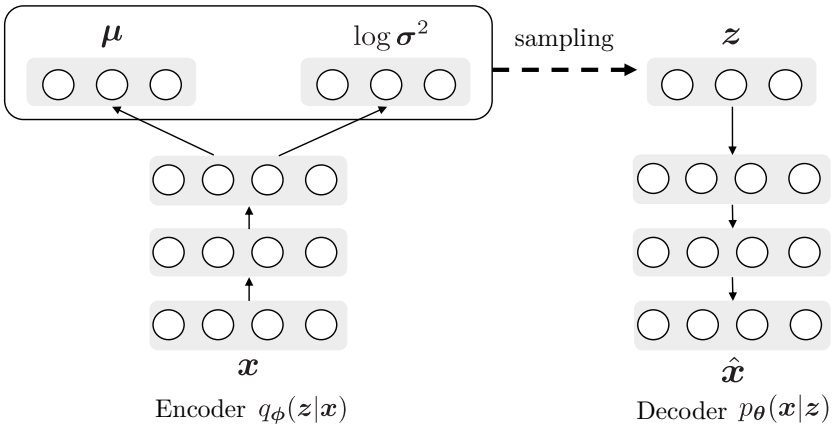


図 2.5　VAE における深層ニューラルネットワークの構造.

のように近似できる．ただし，$\boldsymbol{\epsilon}^{(l)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ であり，$L = 1$ とすることが多い.

式 (2.15) の正則化項（第 1 項）の勾配については，解析的に求めることができる．したがって，式 (2.15) は SGD など通常の最適化アルゴリズムを用いて最適化することができる.

図 2.4.1 は VAE のグラフィカルモデルである．生成モデルは図 2.2.3 と同じだが，近似分布 $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ が推論モデル（推論分布）として点線で表されている.

図 2.5 はエンコーダをガウス分布，デコーダをベルヌーイ分布としたときの VAE における深層ニューラルネットワークの構造を示している.

VAE は生成モデルなので，データ集合を学習した後，新たなデータをサンプリング（生成）

図 2.6　VAE のデコーダによってランダムに生成した MNIST 画像.

することができる．図 2.6 は，VAE を MNIST [LeCun 98] と呼ばれる手書き数字画像で学習した後，ランダムな $\boldsymbol{z}$ からデコーダ $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ を使って画像 $\boldsymbol{x}$ を生成した結果である．これらの画像は，いずれも MNIST データ集合の中には存在せず，VAE が 1 から生成した画像である．このように，VAE は画像のような高次元データを直接学習できるため，それらを新たに作り出すこともできる.

2.4.2　Conditional variational autoencoder (CVAE)

次に，観測変数 $\boldsymbol{x}$ の生成過程を，$\boldsymbol{z} \sim p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ 及び $\boldsymbol{x} \sim p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{y})$ と考える．ここで，$\boldsymbol{y}$ はもう 1 つの確率変数であり，観測変数とする．すると，$\boldsymbol{y}$ が与えられた下での $\boldsymbol{x}$ における条件付き尤度は $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{y}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{y})p(\boldsymbol{z})d\boldsymbol{z}$ となる．この生成モデルを VAE を拡張する形で深層ニューラルネットワークでパラメータ化したものが，条件付き VAE（conditional variational autoencoder，CVAE）である [Sohn 15].

CVAE では，近似分布（エンコーダ）を $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})$ と置く．すなわち，CVAE の目的関

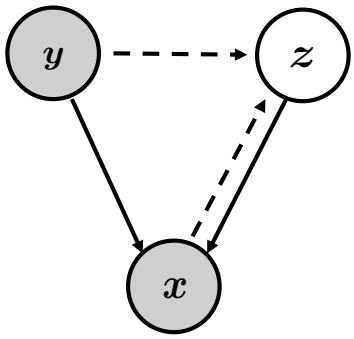


図 2.7　CVAE のグラフィカルモデル.

数は,

$$\begin{aligned}\mathcal{L}(\boldsymbol{x},\boldsymbol{y}) &= E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{y})p(\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})}] \\ &= -D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})||p(\boldsymbol{z})) + E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{y})]\end{aligned} \tag{2.19}$$

となる．このモデルは，VAE と同様の方法で学習できる.

図 2.4.2 は CVAE のグラフィカルモデルである．図 2.4.1 と比較すると，観測変数 $\boldsymbol{y}$ が加わっているのがわかる.

観測変数 $\boldsymbol{y}$ は，2 つの異なる観点でみることができる．まず，$\boldsymbol{x}$ に対応する目標ラベルとみる観点である．そうすると，通常の VAE が訓練集合として $\boldsymbol{x}$ しか与えられない教師なし学習であったのに対して，CVAE は $\boldsymbol{x}$ と $\boldsymbol{y}$ の両方が与えられる教師あり学習とみなすことができる．たとえば，$\boldsymbol{x}$ を手書き数字画像，$\boldsymbol{y}$ を対応する数字ラベルとして学習できる．また，CVAE の生成モデルは $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{y})$ となっているため，学習後に数字ラベル $\boldsymbol{y}$ と潜在変数 $\boldsymbol{z}$ の値を変更して，対応する手書き数字画像を生成することができる．$\boldsymbol{z}$ は $\boldsymbol{y}$ と独立であるため，$\boldsymbol{z}$ は数字ラベルに依存しない「筆跡」のような情報が獲得される．図 2.8 は CVAE を MNIST で学習した後，様々な「筆跡」で各数字を生成した結果である.

もう 1 つは，$\boldsymbol{y}$ を $\boldsymbol{x}$ とは異なる**もう 1 つのモダリティ**とする観点である．この観点では，CVAE は $\boldsymbol{y}$ から $\boldsymbol{x}$ への確率的な変換モデルを学習していることになる．また $\boldsymbol{z}$ は，変換の「不確かさ」を表していると考えることができる．したがって CVAE は，$\boldsymbol{x}$ と $\boldsymbol{y}$ のデータが一

図 2.8　CVAE のデコーダによって数字ごとにランダムに生成した MNIST 画像．縦列が各数字ラベル（すなわち，異なる $\boldsymbol{y}$ の値）に対応し，横列はそれぞれ様々な「筆跡」（すなわち，異なる $\boldsymbol{z}$ の値）に対応している．

対一に対応していなくても，確率的な対応関係を学習することができる．

本論文では，この 2 つを区別するために，$\boldsymbol{x}$ に対応するラベル情報は $\boldsymbol{y}$ と表す一方で，$\boldsymbol{x}$ に対応するもう 1 つのモダリティ情報は $\boldsymbol{w}$ と表すことにする[*12]．

2.4.3　半教師あり学習のための VAE

VAE は半教師あり学習のためのモデルとしても使われる [Kingma 14a, Maaløe 16]．ここでは，Kingma らによって提案された M2 モデル [Kingma 14a] について説明する．

半教師あり学習では，訓練集合として少数のラベルあり集合 $\mathcal{D}_{\mathcal{L}} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_N, \boldsymbol{y}_N)\}$ と大量のラベルなし集合 $\mathcal{D}_{\mathcal{U}} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_{N'}, \boldsymbol{y}_{N'})\}$ から識別モデル $p(\boldsymbol{y}|\boldsymbol{x})$ を学習する．

[*12] 近年のマルチモーダル学習の研究ではもう 1 つのモダリティを $\boldsymbol{y}$ と表すことが多い [Vedantam 17, Higgins 17] が，本論文ではこのような理由から，以降も $\boldsymbol{w}$ と表している．

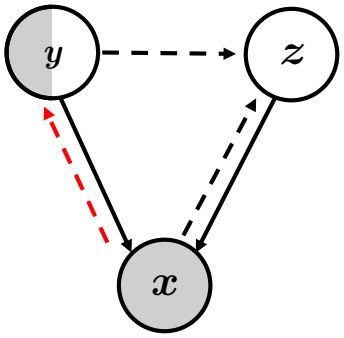


図 2.9　M2 モデルのグラフィカルモデル.

CVAE と同様に 2 つの観測変数 $\boldsymbol{x}$，$\boldsymbol{y}$ を考える．それらの同時分布の下界 $\mathcal{L}(\boldsymbol{x}, \boldsymbol{y})$ は，

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{y}) \geq E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{y})p(\boldsymbol{z})p(\boldsymbol{y})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})}] \equiv \mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) \tag{2.20}$$

となる．式 (2.19) と比べると，式 (2.20) では $p(\boldsymbol{y})$ が加わっているが，K クラス分類問題の場合では $p(\boldsymbol{y}) = \mathrm{Cat}(\boldsymbol{y}; \boldsymbol{\pi})$ であり，$\boldsymbol{\pi} = [\frac{1}{K}, ..., \frac{1}{K}]^T$ のように定数とされることが多いので，結果的に最適化する下界は同じとなる．

式 (2.20) はラベルあり集合 $\mathcal{D}_{\mathcal{L}}$ において学習するが，半教師あり学習の枠組みではラベルなし集合 $\mathcal{D}_{\mathcal{U}}$ も学習に利用する．そこで，ラベル情報を含まない周辺分布 $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ の下界を求めて目的関数とする．この目的関数は，識別モデル $q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x})$ を導入して次のように求められる．

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq E_{q_{\boldsymbol{\phi}}(\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{y})p(\boldsymbol{z})p(\boldsymbol{y})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{x})}] \equiv \mathcal{U}(\boldsymbol{x}). \tag{2.21}$$

ただし $q_{\boldsymbol{\phi}}(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{x}) = q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{y})q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x})$ である．

さらに，ラベルあり集合において識別モデルを学習するために，以下のようにラベルあり集合における識別モデルの対数尤度を式 (2.21) に加える．

$$\mathcal{L}_l(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) + \alpha \cdot \log q_{\phi}(\boldsymbol{y}|\boldsymbol{x}). \tag{2.22}$$

ただし α は，識別モデルと生成モデルの割合を調節するパラメータである．

したがって，ラベルあり・なし集合の両方における目的関数 $\mathcal{J}$ は，

$$\mathcal{J} = \frac{1}{N} \sum_{(\boldsymbol{x}_n, \boldsymbol{y}_n) \in \mathcal{D}_{\mathcal{L}}} \mathcal{L}_l(\boldsymbol{x}_n, \boldsymbol{y}_n) + \frac{1}{M} \sum_{\boldsymbol{x}_m \in \mathcal{D}_{\mathcal{U}}} \mathcal{U}(\boldsymbol{x}_j) \tag{2.23}$$

となる．VAE や CVAE と同様に，これを最大化するように深層ニューラルネットワークを学習することで，end-to-end に生成モデル，推論モデル，そして識別モデルを同時に学習することができる．

図 2.4.3 は M2 モデルのグラフィカルモデルである．なお，ラベル $\boldsymbol{y}$ が黒と白の半分ずつになっているのは，ラベルありとラベルなしの両方（すなわち，観測変数と潜在変数の両方）として考えるためである．また赤い点線は，識別モデルを表している．

深層生成モデル（生成モデル）の利点として，このように教師あり学習と教師なし学習を統一的に扱えることがある．深層生成モデルでは，教師あり・なしの違いは，ラベルに対応する確率変数が，観測変数か潜在変数かの違いに過ぎない．また識別モデルも，入力が与えられた下でラベルを生成する確率分布として，生成モデルの中に統一的に組み込むことができる．式 (2.23) の目的関数は，これらすべての枠組みを含んでおり，この目的関数を最大化することで統一的に学習できる．

これ以外の VAE に基づく半教師ありモデルとして，M2 モデルを発展させた ADGM と SDGM [Maaløe 16] などがある．

第 3 章

マルチモーダル情報の定義と関連研究

本章では，マルチモーダル情報の定義と，5 章以降の各研究についての関連研究を説明する．

まず 3.1 節で，マルチモーダル学習で扱われるマルチモーダル情報の定義について説明する．そして 3.2 節で，マルチモーダル学習の各問題設定とそれらの従来研究を概観する．最後に，3.3 節で確率モデリングのライブラリについて概観し，それらをマルチモーダル学習に用いる上での利点及び欠点について議論する．

3.1　マルチモーダルの定義について

1 章で説明したとおり，本論文では，マルチモーダル情報とは異種性をもつデータのことを指すとする．また異種性は，2 つのモダリティの特徴空間と分布が異なるものとする．本節では，様々なサーベイ論文などを参考に，この定義の妥当性について議論する．

3.1.1　ドメインとタスクについて

異種性について議論するにあたり，機械学習におけるドメイン及びタスクの定義について説明する．

ドメインとタスクの定義は，これまで多少の混乱があり，明確な定義が定まっていなかった．たとえば，Daumé はドメインとタスクの違いを，転移学習（transfer learning）とマルチ

タスク学習（multi-task lerning）に対応すると述べている[*1]．しかし，今日では転移学習はドメイン適応も含む，より広い領域を指す言葉と捉える方が主流である [神嶌 10, Pan 10]．一方で，神嶌による転移学習のサーベイでは，ドメインとタスクは違いが明確でないとして，いずれもドメインという言葉で統一されている [神嶌 10]．しかし前述の Daumé はじめ，複数の論文ではドメインとタスクの違いは明確に区別されている [Pan 10, Saenko 10]．

本論文では，Pan らの転移学習のサーベイ [Pan 10] を参考にドメインとタスクの定義を明確にする．この定義は，のちに Weiss らのサーベイ [Karl Weiss 16] でも採用されたことから，現在のところ最も標準的な定義であると思われる．

入力データの集合 $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \in \mathcal{X}$ が与えられるとき，**ドメイン（domain）**とは，データの特徴空間 $\mathcal{X}$ 及び分布 $p(\boldsymbol{X})$ で構成されるものである．すなわち，$\{\mathcal{X}, p(\boldsymbol{X})\}$ なので，ドメインが異なるということは

1. 特徴空間 $\mathcal{X}$ が異なる
2. 分布 $p(\boldsymbol{X})$ が異なる
3. 特徴空間 $\mathcal{X}$ と分布 $p(\boldsymbol{X})$ の両方が異なる

という 3 種類が考えられる．Pan らはこれらを文書で例え，1 を異なる言語，2 を異なるトピックとしている [Pan 10]．

ドメイン $\{\mathcal{X}, p(\boldsymbol{X})\}$ が与えられるとき，**タスク（task）**は目標のラベル空間 $\mathcal{Y}$ と分類確率（条件付き分布）$p(\boldsymbol{y}|\boldsymbol{X})$ によって $\{\mathcal{Y}, p(\boldsymbol{y}|\boldsymbol{X})\}$ のように構成される．ただし $\boldsymbol{y}$ は入力データ集合 $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$ に対応する目標のラベル集合 $\boldsymbol{y} = \{y_1, ..., y_n\} \in \mathcal{Y}$ である．また任意のラベル y_n が取りうる実現値を**クラス（class）**といい，目標のカテゴリを数値に符合化したものである．ラベル空間が K 個のクラスから構成されるなら $\mathcal{Y} = \{l_1, ..., l_K\}$ となる．さらにドメインの議論と同様に，タスクが異なるということは，

1. ラベル空間 $\mathcal{Y}$ が異なる
2. 分類確率 $p(\boldsymbol{y}|\boldsymbol{X})$ が異なる
3. ラベル空間と分類確率 $p(\boldsymbol{y}|\boldsymbol{X})$ が異なる

[*1] https://nlpers.blogspot.jp/2007/11/domain-adaptation-vs-transfer-learning.html，2018 年 2 月アクセス．

という 3 種類が考えられる．これについても Pan らは文書に例えて，1 をクラス分類の違い（10 クラス分類と 2 クラス分類），2 を定義されたクラスについてインバランスな状態としている．

これらの議論を踏まえると，ドメインとタスクはそれぞれ入力データの特徴空間と目標のラベル空間に着目していることがわかる．

ドメインやタスクが 2 つ与えられ，あるドメインから別のドメイン，もしくはあるタスクから別のタスクに変換したり転用する問題設定は**転移学習（transfer learning）**と呼ばれる．一般に，転移する元は元ドメイン／タスク，転移する先は目標ドメイン／タスクと呼ばれる．

マルチモーダル学習でいうと，変換の問題設定（3.2.2 節）はドメインの転移，共学習（3.2.4 節，特にゼロショット学習）はタスクの転移と考えることができる．

3.1.2　異種性とモダリティ

異種性（heterogeneity）もしくは**異種（heterogeneous）**という用語は，ドメインと関連して使われるようになっているものである．これは，転移学習の研究において異種転移学習（heterogeneous transfer learning）のように用いられるようになって以降広まったものである [Shi 10, Zhu 11]．Pan らのサーベイ [Pan 10] では異種性についてはまだ明記されていなかったが，Machine Learning Summer School 2011 のトークでは，転移学習の分類において，新たに異種という項目が追加されている[*2]．

Weiss らのサーベイ [Karl Weiss 16] では Pan らのサーベイでの定義を踏襲しつつ，異種転移学習についても定義している．それによると，転移学習において特徴空間 $\mathcal{X}$ が異なる場合に異種転移学習と呼ぶとしている．つまり，上記のドメインの定義でいうと「1. 特徴空間 $\mathcal{X}$ が異なる」もしくは「3. 特徴空間 $\mathcal{X}$ と分布 $p(\boldsymbol{X})$ の両方が異なる」データが異種性を持つということになる．しかし，文献 [Karl Weiss 16] によると，多くの異種転移学習では分布が同じであることが仮定されている．実際，深層学習を用いて異種情報を扱うアプローチの多くが，暗黙的に分布の同一性を仮定している [Socher 13]．

しかし Zhou らが指摘するように，分布が異なる場合も確かに存在する [Zhou 14]．彼らは

[*2] http://www.ntu.edu.sg/home/sinnopan/tutorials/[MLSS11]Transfer%20Learning.pdf，2018 年 2 月アクセス．

英語とドイツ語という異なる 2 言語（異種情報と考える）を転移する場合に生じる 2 つの問題を説明している．

1 つめは，英語文書のラベルありデータ集合から，ドイツ語の文書の分類器を学習するというものである．簡単な方法としては，いくつかのドイツ語の文書を Google 翻訳で英語に変換することで，ドイツ語と英語の対応関係を作ることである．これができれば，分類したいドイツ語を英語に翻訳して，英語の分類器を用いることができる．しかし，これら単語帳作成用文書と訓練集合の文書では，文書の種類の違いからドイツ語から英語への単語ごとの対応関係が異なる可能性がある（文献 [Zhou 14] ではドイツ語の「betonen」の訳を例に挙げている）．これは，英語の訓練集合とドイツ語から翻訳した英語文書で，単語分布が異なるためである．よって，英語の分類器を適切にドイツ語に転移することはできない．

2 つめは，多言語のセンチメント分類問題である．英語における本のレビューのラベルあり集合と，英語とドイツ語の曲レビューのペア（ラベルなし）が与えられた下で，ドイツ語の曲レビューをセンチメント分類する．英語からドイツ語への変換を学習するために，曲レビューのペアを利用できる．しかし英語の本のレビューは，本と曲で使われている言葉の種類が大きく異なるため，学習に有用でない可能性がある．

これら 2 つの例はいずれも 2 つの異種情報の分布の違いを説明しているものだが，前者は異種性による違い，後者は事例集合による違いを示している．後者の問題は異種性に関わらずドメインの違いとしてあるものである．本研究では，主に同じ集合間での分布の違いについて考慮するので，異種情報の分布の違いという場合には，一対一対応ができないことを指し，後者については明示的に考慮しない．

3.1.3　本論文での定義のまとめ

ここまでの議論を踏まえて，本論文では，異なるデータ集合 $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \in \mathcal{X}$ と $\boldsymbol{W} = \{\boldsymbol{w}_1, ..., \boldsymbol{w}_m\} \in \mathcal{W}$ が異種性をもつというのは，集合の特徴空間が異なる場合 $\mathcal{X} \neq \mathcal{W}$ を指すものとする．ただし，特徴空間が異なることと，分布が異なることは排他ではないため，分布が異なる可能性も考慮する必要があることに留意されたい．前述の通り，これがこれまでのマルチモーダル研究や異種転移学習では着目されていなかった部分である．したがって本論文では特に，**特徴空間と分布の両方が異なる場合**を指すものとする（図 3.1）．

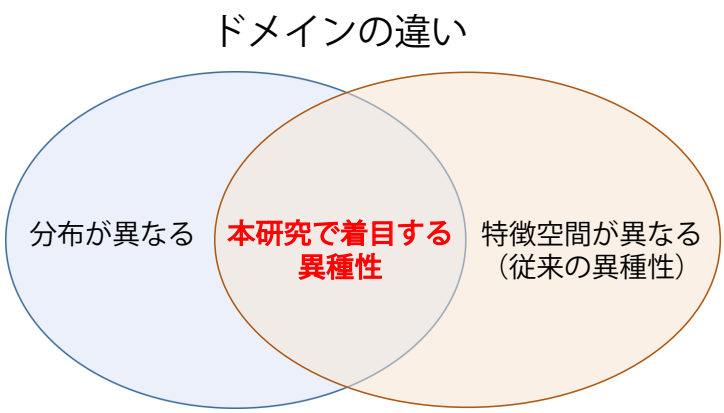


図 3.1　ドメインの違いの定義と本研究で着目する異種性との関係.

一方モダリティとは，データを知覚する様々な方式及び状況のそれぞれを指す用語である [Lahat 15, Baltrušaitis 17]．つまり，モダリティはデータを得る枠組みを指す用語なのに対して，異種データはそれによって得られるデータそのものを指す用語である．通常，モダリティが異なっても得られる情報が異種性を持つとは限らないが，本論文では，逆説的にそれぞれの枠組みで得られたデータが異種性を持つ場合に，それらの枠組みや，枠組みから得られるデータを，まとめて異なる**モダリティ（modality）**と呼ぶことにする．またそのような異なるモダリティの集まりを**マルチモーダル（multimodal）**と呼ぶ．マルチモーダルがデータや情報を指すことを強調する場合には，マルチモーダル情報／データ（multimodal information/data）と呼ぶ.

3.2　マルチモーダル学習の問題設定と既存研究

1 章で述べたように，マルチモーダル学習には 5 つの問題設定が存在する [Baltrušaitis 17]．再掲すると，次のとおりである.

表現（representation）　マルチモーダルデータをどのように表現し要約するか.

変換（translation）　あるモダリティから別のモダリティへデータをどのように変換するか.

アラインメント（alignment）　異なるモダリティ間の部分要素ごとの直接的な関係をどのように特定するか.

融合（fusion）　異なるモダリティをどのように結合しどのように目標ラベルの予測を行うか.

共学習（co-learning）　モダリティやその表現及び予測モデル間でどのように知識を伝えるか.

本節では，それぞれの問題設定について代表的な先行研究を挙げて，深層学習と生成モデルの観点から議論する．なお，アラインメントについては系列データで主に扱われる問題設定であり，本研究では系列データを扱わないため，省略する．

また，5 章から 7 章の各提案手法に直接関係する先行研究については，改めて各章にて説明する．

3.2.1　表現について

マルチモーダルな表現とは，ある目標について，複数の異なるモダリティから得られた情報を用いて表現することである．機械学習では良い表現が精度に大きく貢献することが知られている．一般的な「良い表現」については 2.3 節で説明したとおりだが，特にマルチモーダル学習における複数のモダリティの情報を統合した共有表現については，Srivastava らによって次の 3 つが示されている [Srivastava 12].

1. 共有表現は，表現空間における類似性が対応する「概念」の類似性に関連していなければならない.
2. 共有表現は幾つかのモダリティが欠損しても容易に得ることができ，欠損したモダリティは他のモダリティから補完することができる.
3. 共有表現は，識別タスクに有用でなければならない.

3 つ目については，目標を予測する問題，すなわち，融合問題に関連している．共有表現を獲得するアプローチは，ニューラルネットワークを用いた方法と，DBM などの生成モデルを用いた方法に大別される.

ニューラルネットワークを用いた方法

ニューラルネットワークによるアプローチはとてもシンプルである．まず，各単一モダリティを入力とするネットワークを用意し，それらの最終層または隠れ層を結合することで，

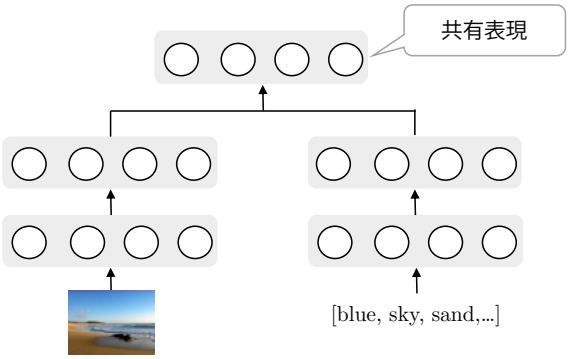


図 3.2　異なるモダリティからの共有表現の獲得.

その層で共有表現を獲得するというものである（図 3.2）．この手法は，共有表現を得るための単純かつ最も利用されている方法である．Ngiam らは深層オートエンコーダを用いたマルチモーダル学習を提案し，単一のモダリティよりも良い表現が得られることを示している [Ngiam 11]．しかし，ニューラルネットワークは片方のモダリティの入力を欠損させると，共有表現が崩れてしまうという課題がある．これは，結合したニューラルネットワークが決定論的な写像で構成されているためである．また，異なるモダリティの分布の違い，すなわち，情報量の違いを考慮できないという課題もある．

DBM などの生成モデルを用いた方法

一方，Srivastava らによって提案された DBM の手法は，生成モデルによってすべてのモダリティと潜在変数の同時分布モデル化されているため，モダリティが欠損しても共有表現を推論でき，他のモダリティから簡単に補完することができる [Srivastava 12]．さらに Sohn らは Srivastava らの手法を拡張し，variation of information 最小化に基づいて DBM を学習することで，2 つのモダリティが双方向に変換でき，かつ Srivastava らよりも識別精度の高い共有表現を獲得できるモデルを提案した [Sohn 14]．このように DBM に基づく手法は，上記の良い共有表現の要件をすべて満たしている．しかし，前述のように DBM の学習則はマルコフ連鎖モンテカルロ（MCMC）法に基づくので，これらのモデルは自然画像のような高次元のデータを入力として学習できないという問題がある．

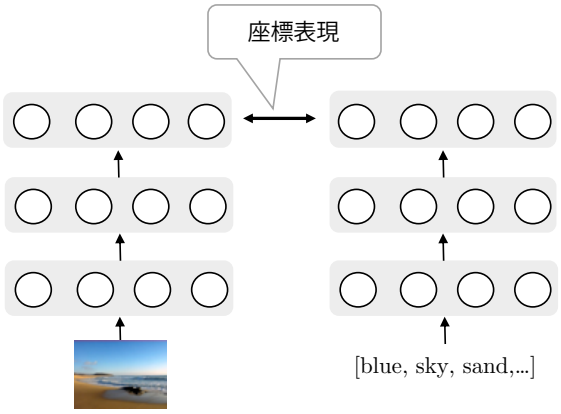


図 3.3　異なるモダリティからの座標表現の獲得.

座標表現によるアプローチ

表現を獲得する問題設定では，共有空間に写像する以外にも，異なるネットワークを共有するのではなく，それぞれの表現空間を互いに近づけるというアプローチがある（図 3.3）．こうして得られる表現は，座標表現（coordinate representation）と呼ばれる．Deep visual-semantic embedding（DeViSE）と呼ばれるゼロショット学習の手法は，CNN による画像からの特徴表現とラベルから word2vec による埋め込み空間の誤差（具体的には hinge rank loss）を近づけるように学習する [Frome 13]．同様のアプローチで，ラベルの部分を RNN に変えたり [Socher 14]，画像を動画に変えたりした手法 [Ouyang 14] が提案されている．このアプローチは，各モダリティのネットワークを結合しないので，一方のモダリティの入力が欠損しても，もう片方の表現が崩れることはない．ただし，分布の違いを考慮できないという課題が残っている．

これらについてまとめると，表 3.1 のようになる．

アプローチ	高次元入力	欠損モダリティの対処	分布の違いの考慮
共有表現（ニューラルネットワーク）	○	○	×
共有表現（DBM などの生成モデル）	×	○	○
座標表現	○	○	×

表 3.1　表現についての既存のアプローチの比較.

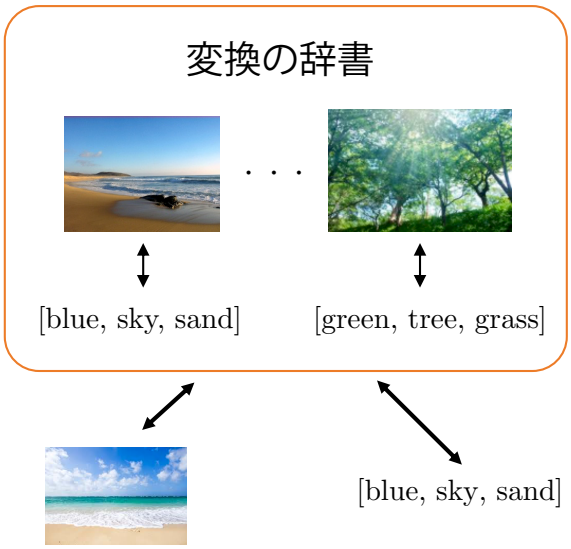


図 3.4　事例ベースの概要.

3.2.2　変換について

モダリティ間の変換は，マルチモーダル学習の中でも最も研究されており，音声合成 [Hunt 96] やクロスモーダル検索 [Rasiwasia 10] などは初期のモダリティ変換の研究といえる．最近では，Microsoft COCO データ集合 [Chen 15] などの登場により，自然言語処理と画像を結びつける研究，特に，画像キャプショニング [Farhadi 10] や動画キャプショニング [Venugopalan 14] の研究が盛んである．

変換の手法は非常に広範囲にわたり，様々な手法が提案されているものの，[Baltrušaitis 17] では大きく事例ベースの方法と生成による方法に分けている．

事例ベースの方法

事例ベースの方法は，訓練データから「辞書」と呼ばれる変換元の事例と変換後の事例の対応関係を作成することに基づく（図 3.4）．この対応関係は，学習した決定論的な写像関数から得られる．ある事例を別のモダリティに変換する際には辞書から最も近い事例を検索することで，対応する変換先のデータを求めることができる．このとき得られる変換先のモダリティデータは，実際に辞書に存在する事例であり，生成する訳ではないことに注意された

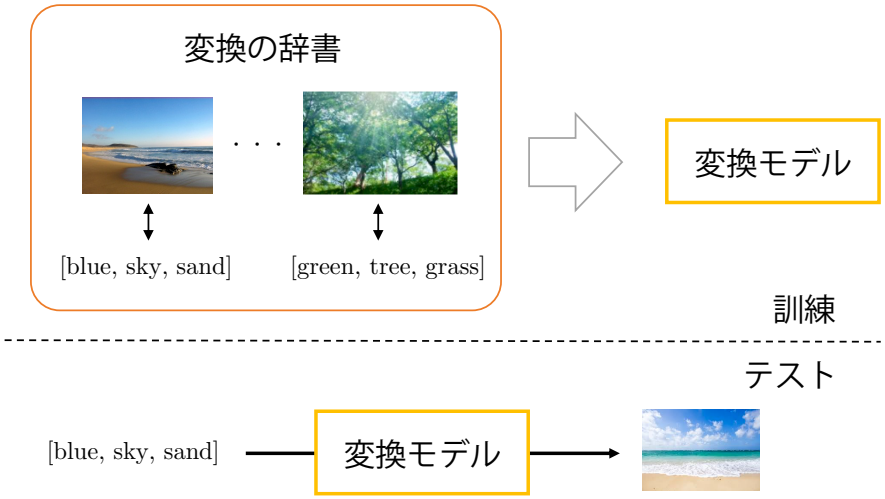


図 3.5　生成による方法の概要.

い．Socher らは，画像とキャプションの変換において，係り受け木をベースとした DT-RNN を学習してキャプションの分散表現を獲得し，画像の CNN 特徴ベクトルからキャプション空間への線形写像を学習することで，画像と説明文の関係を獲得する方法を提案している [Socher 14]．このように，2 つのモダリティを中間的な意味表現に写像するような関数が学習できれば，その意味表現上で検索することで，変換先のモダリティの事例を推定することができる．

この手法の利点は，双方向の変換が容易であることである．訓練集合から中間表現への写像が学習されてしまえば，いずれの方向でもモダリティ間の変換を行うことができる．一方で，この手法の欠点は，辞書を常に保持していなければならないためメモリコストが増大すること，そして事例が増えると検索に時間がかかってしまうことである．Cao らは検索時間の問題を解決するため，中間意味表現をハッシュ空間にしたモデルを提案している [Cao 16]．しかし，それらよりも大きな問題が，学習したモダリティ間の決定論的な関係は自明ではないということである．それぞれのモダリティの分布が異なる場合は，この関係は決定論的にはならない．

生成による方法

生成によるアプローチは，変換先のデータを辞書から検索するのではなく，辞書から変換モデルを学習し，その変換モデルを使って 1 から変換先のデータを生成する手法である（図 3.5）．これまで最も主流だったのは，変換モデルとして RNN のエンコーダ・デコーダモデルを利用するアプローチである．エンコーダに変換元のモダリティデータを入れると，デコーダで変換先のモダリティデータが得られるというものである．画像からキャプションを生成する場合 [Farhadi 10] や動画からキャプションを生成する場合 [Venugopalan 14] によく使われ，エンコーダの特徴表現としては CNN から得たものが利用される．この手法では，1 つの画像や動画事例から対応する文書を生成しなければならないため，変換元の画像からできるだけ文書に有用な情報を選択して変換に利用する必要がある．最近では，アテンションメカニズムによって，ある単語が画像のどの部分に対応しているかを学習することで，これを実現している [Xu 15]．

しかし，上記のアプローチには 2 つの制限がある．まず 1 つは，事例ベースのアプローチのように双方向の変換は不可能であるということである．そしてもう 1 つは，やはり決定論的な変換を学習しているため，完全に対応関係のある変換しかできないということである．上記のような画像から文書に変換する場合はあまり問題にならないかもしれないが，文書から画像を生成する場合は重大な問題となる．何故なら，キャプションが同じでも対応する画像は大きく変わりうるためである．これはモダリティ間の情報量の違い，すなわち，分布の違いに起因する．つまり，この問題を解決するためには，各モダリティの分布の違いを考慮した変換モデルで変換を行う必要がある[*3]．

近年では，2.4.2 節で説明したように，あるモダリティ $\boldsymbol{w}$ が与えられた下でのもう片方のモダリティ $\boldsymbol{x}$ の条件付き分布 $p(\boldsymbol{x}|\boldsymbol{w})$ をモデル化するような深層生成モデル（条件付きモデルと呼ばれる）が提案されている [Kingma 14a, Sohn 15]．詳細については 5.1 章で説明するが，こうした条件付きモデルを変換モデルとして用いることで，異なるモダリティ間の分布の違いを考慮することができる．しかし，これらの手法は条件付き分布のみをモデル化しているため，双方向に生成することはできない．

[*3] 画像から文書の変換や，文書間の変換の場合は，多くの研究でアテンション（attention）メカニズムによってこの問題の解決を試みている．これについては 9.2.1 節でも改めて議論する．

以上についてまとめると，表 3.2 のようになる．

アプローチ	双方向変換	分布の違いの考慮
事例ベースの方法	○	×
生成による方法（エンコーダ・デコーダモデル）	×	×
生成による方法（深層生成モデル，条件付きモデル）	×	○

表 3.2　変換についての既存のアプローチの比較．

3.2.3　融合について

一般的に「マルチモーダル学習」という言葉が使われる場合は，複数のモダリティから目標ラベルを予測する問題を指すことが多い．これを融合の問題設定という．マルチモーダル学習の中では最も研究が多く，これまでも様々なアプローチが提案されている．

マルチモーダル学習における融合が最も注目を集めているのは，異なるモダリティには相補性という性質があるからである [Lahat 15]．これは，あるモダリティからは，他のモダリティにはない付加価値を得られるということを指す．ここでの価値とは，目標ラベルを予測する上で大きく貢献するという意味である．こうした付加価値はモダリティの多様性（diversity）とも呼ばれる [Lahat 15]．この観点は，ある目標 $\boldsymbol{y}$ について異なるモダリティ $\boldsymbol{X}$ と $\boldsymbol{W}$ が得られる場合，$p(\boldsymbol{X}|\boldsymbol{y}) \neq p(\boldsymbol{W}|\boldsymbol{y})$ となることを暗示している．このことからも，異なるモダリティは分布が異なることが示唆される[*4]．

上記の相補性や多様性の性質から，異なるモダリティを多く得られれば，より多くの目標ラベルに関する情報を得られようになる．しかし，これらの表現には互いに冗長性があるため，これらを単純に結合するのではなく，なんらかの方法で適切に目標への写像を学習する必要がある．

従来とられたアプローチは，マルチカーネル学習（multiple kernel learning, MKL）を用いる方法である．これは，サポートベクトルマシン（support vector machines, SVM）によって，異なるモダリティに対して異なるカーネルを用意して学習する方法である [Bucak 14]．しかし最近では，深層ニューラルネットワークによる方法に置き換わっている．

[*4] $p(\boldsymbol{y})$ が一定ならば，$\int p(\boldsymbol{X}|\boldsymbol{y})p(\boldsymbol{y})d\boldsymbol{y} \neq \int p(\boldsymbol{W}|\boldsymbol{y})p(\boldsymbol{y})d\boldsymbol{y}$ なので，$p(\boldsymbol{X}) \neq p(\boldsymbol{W})$ となる．

深層ニューラルネットワークによるアプローチでは，表現と融合に大きな違いはなく，特に近年は区別されることが少なくなっている．たとえば，Visual question answering では，画像とそれに対する質問文から共有表現に写像し，そこから答えの単語を予測している [Antol 15]．

MKL によるアプローチと比較したときの深層学習の利点は，大量のデータから学習できる容量があること，すべてのモダリティに関するネットワークを end-to-end に学習できること，そして非常に複雑な非線形写像も学習できるということが挙げられる [Baltrušaitis 17]．しかしその一方で，表現の場合と同様に，欠損モダリティをうまく扱えないという問題がある．また融合の問題設定は，表現とは異なり，目標ラベルを予測する教師あり学習である．したがって精度を高めるには，大量のラベルありデータが必要になる．

この問題を解決する方法の 1 つが，半教師あり学習の枠組みを用いることである．これまでも，いくつかの半教師ありマルチモーダル学習が提案されている [Guillaumin 10, Cheng 16]．たとえば，Guillaumin らは，画像とタグからなるマルチモーダルデータに少数のラベルしかない場合に，ラベルのないマルチモーダルデータを用いて，画像からラベルをより高精度に予測する半教師あり学習の枠組みを提案した [Guillaumin 10]．

その一方で，近年深層生成モデルによる半教師あり学習が提案されている [Kingma 14a, Maaløe 16, Salimans 16]．これらの手法は，ラベルありデータとラベルありデータを統一的に扱えるため，従来の手法よりも end-to-end に効率よく高い精度で半教師あり学習を実行することができる．しかし，マルチモーダルデータを入力とした場合の深層生成モデルによる半教師あり学習の手法は，これまで殆ど提案されていない．

3.2.4　共学習について

共学習とは，あるモダリティのデータが不足していてモデリングが困難な場合に，他のモダリティを頼りにして学習を進める枠組みである．特に，訓練集合に，ある入力に対して予測したい目標クラスのラベルが全くない場合は，**ゼロショット学習（zero-shot learning）**と呼ばれる．より形式的な問題設定については 7.1.1 節を参照されたい．

ゼロショット学習

ゼロショット学習に関する初出の研究は Larochelle らによるゼロデータ学習（zero-data learning） [Larochelle 08] である．この研究では手書き文字認識を対象とし，元タスクを数字，目標タスクをアルファベットなどとし，数字に関する学習を行った後，学習していない目標タスクに関する分類を行うというものである．この場合，目標タスク即ちアルファベットに関する知識がないため，文字をドット表現したものを中間表現として導入して学習を行っている．

その後現在まで様々な方法が提案されているが，大きく分けると以下の 2 つのアプローチがある．

セマンティックな知識への埋め込み　ゼロショット学習で最も用いられている手法で，補助情報としてセマンティックな埋め込み空間を与えて，入力からの写像を学習するというものである．このとき埋め込み空間は，すべてのクラスについて予め定義されているので，元タスクで学習した写像を使って，目標タスクに関する予測を行う．上述のゼロデータ学習もドット表現を埋め込み空間として知識の転移を行っている．セマンティックな知識には，属性が最もよく使われる [Lampert 09, Sharmanska 12, Akata 13, Lampert 14]．本研究も属性を用いたゼロショット学習を提案しており，次節で属性についての説明をする．属性以外では，単語ベクトルや，Wikipedia や WordNet から自然言語処理によって特徴量を抽出して補助情報として利用する手法がある [Akata 13, Fu 14a]．また，セマンティックな知識とデータ同士の類似度を併用することで，転移学習を行うアプローチもある [Rohrbach 10].

クラス間の関係　もう 1 つが，補助情報として元タスク・目標タスク間の関係グラフを利用する手法である [Rohrbach 10, Fu 14c]．この手法では，元タスクに関する通常の学習を行ったあと，関係グラフを利用して目標タスクでのクラスを予測する．利用する関係グラフは，2 部グラフ [Rohrbach 10]，吸収的マルコフ過程 [Fu 14c] などが提案されている．また，Lampert らの IAP モデルも関係グラフを利用したものと考えられる [Lampert 09, Lampert 14].

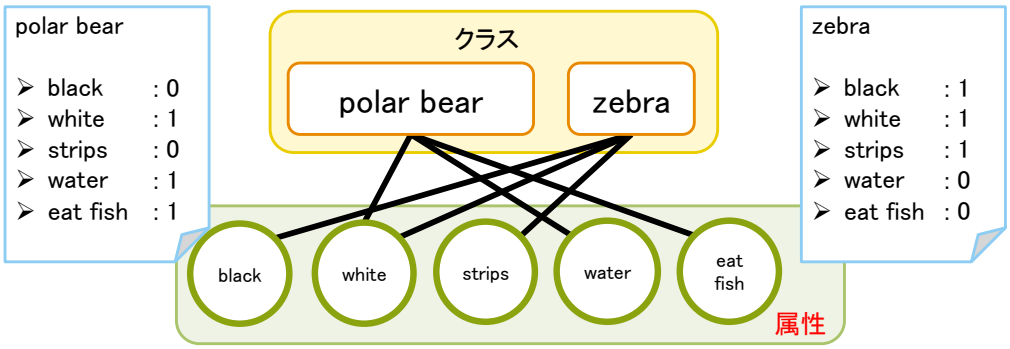


図 3.6　目標クラスと属性の例.

属性ベースゼロショット学習

属性（attribute）は 2009 年以来，コンピュータビジョンの主要なトピックの 1 つとして研究されている．画像の他に動画 [Fu 14b] などに用いられ，ゼロショット学習の他に，見慣れない物体の認識 [Farhadi 09] にも用いられる．属性の定義として Lampert らは “We call a property of an object an attribute, if a human has the ability to decide whether the property is present or not for a certain object” と述べている [Lampert 14]. 即ち人間がある/ないを判別できるようなモダリティが属性である．属性はクラスに対して [Lampert 09, Lampert 14] の他，各画像ごと [Farhadi 09] に定義される．一般的には Yes / No の 2 値をとるが，属性同士の大小関係で定義する研究 [Parikh 11b] もある．属性の学習方法として，入力と属性の間に特徴を挟んだり [Hwang 11]，多層ニューラルネットワークを用いる研究 [Chung 12] がある．

図 3.6 は目標クラスと属性の関係を表した例である．この図では，ライオンとクジラというクラスに対して属性が定義されている．この場合，画像の特徴量（即ちドメイン）が異なっていてもクラスが同じならば定義される属性は同じである．よって，属性の知識が与えられれば，クラスを選ぶことができる．このような属性の性質は次に述べる DAP モデルで活用されている．

属性ベースのゼロショット学習は Lampert らによって提案された [Lampert 09, Lampert 14]．特に埋め込み空間として属性を利用した Direct Attribute Prediction (DAP) モデルは属性ベースゼロショット学習の代表的手法として知られ，このモデルを基に様々な手

法がこれまで提案されている [Li 13, Kankuekul 12]．DAP モデルの基本的なアイディアは単純で，画像を属性空間に写像する関数を学習することで，未知のクラスラベルについても予測できるというものである．このモデルは，属性という人手で事前に作成された知識を，画像の予測に効果的に用いることができる．

しかしその一方で，モダリティ間の分布の違いから，属性と画像の関係は必ずしも一対一にならないという問題がある．この問題について，詳しくは 7.1.2 節で述べる．

3.3　確率モデリングのためのライブラリ

本節では，生成モデルなどの確率モデリングを行うためのライブラリを俯瞰する．また，それぞれについて深層生成モデル，特に複数のモダリティをもつ複雑な深層生成モデルが実装できるかどうかに焦点を当てる．

3.3.1　確率モデリングの推論計算について

確率モデリング，特にベイズモデリングを行う際に重要なのは，どのように確率分布の推論を行うかということである．

観測変数 $\boldsymbol{x}$ 及び潜在変数 $\boldsymbol{z}$ において，観測変数の周辺分布が $p(\boldsymbol{x}) = \int p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}$ とモデル化されるとする．このとき，推論とは事後分布 $p(\boldsymbol{z}|\boldsymbol{x})$ を計算することである．この事後分布は

$$p(\boldsymbol{z}|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{\int p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}} \tag{3.1}$$

と計算できる．しかし，分母に期待値の計算が入っているため，解析的に計算することは困難である．そのため，一般的に以下の 2 種類の近似手法が取られる．

- サンプリングによって確率的に近似する
- 近似事後分布 $q(\boldsymbol{z})$ を置き，真の事後分布との距離を近づける形で近似する

1 つ目のサンプリング近似には，マルコフ連鎖モンテカルロ法 [Robert 05] が使われ，その中でもメトロポリス・ヘイスティングス法やハイブリッド・モンテカルロ法 [Neal 11] などが用いられる．以下に挙げる確率プログラミング言語も，この手法をとるものが多い．

2 つ目の方法は，KL ダイバージェンスを最小化する方策が取られ，特に

$$\min_{q} D_{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{x})) \tag{3.2}$$

の形をとる場合は，変分推論で求めることできる[*5]．学習による変分推論，すなわち，$q(\boldsymbol{z})$ を $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ のように $\boldsymbol{x}$ から $\boldsymbol{z}$ への写像で近似すると，

この KL ダイバージェンスは

$$\begin{aligned} D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z}|\boldsymbol{x})) &= \log p(\boldsymbol{x}) - \int q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \log \frac{p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})} d\boldsymbol{z} \\ &= \log p(\boldsymbol{x}) - \mathcal{L}(\boldsymbol{x};\boldsymbol{\phi}) \end{aligned} \tag{3.3}$$

と分解できる．このとき，$\mathcal{L}(\boldsymbol{x};\boldsymbol{\phi})$ は変分下界である．$\log p(\boldsymbol{x})$ は最適化したいパラメータに依存しないので，KL ダイバージェンスの最小化は，変分下界の最大化で達成できる．

変分下界の計算は，従来は決定的に求める方法が主流であったが，VAE などで用いられる再パラメータ化トリック [Kingma 13, Rezende 14] の登場以降は，積分をモンテカルロ近似によって計算する方法が取られることが多くなっている．

以上の 2 つの近似方法があることを踏まえた上で，既存の確率プログラミング言語について見ていく．

3.3.2 Stan

Stan は Andrew Gelman らによって開発されている確率プログラミングを行うための言語である[*6]．C++ で書かれているため高速にサンプリングできることが特徴で，2012 年頃から本格的な開発が進んでいる．C++ で書かれているものの，モデル等を記述した Stan 形式のファイルを Python や R で読み込んで実行するインターフェイスもある（PyStan[*7]，RStan[*8]）．

Stan では，主にサンプリングによるモデルパラメータの近似を行う．特に，ハミルトニアン・モンテカルロ法の 1 つの実装方式である NUTS（No-U-Turn Sampler）[Hoffman 14] が

[*5] KL ダイバージェンスの向きが逆の場合は期待値伝播（expectation propagation，EP）法として知られている．

[*6] http://mc-stan.org

[*7] https://github.com/stan-dev/pystan

[*8] https://github.com/stan-dev/rstan

デフォルトのサンプラーとして採用されている．また近年は，変分推論によってパラメータの事後分布を近似する方法である ADVI（Automatic Differentiation Variational Inference）[Kucukelbir 15] も使うことができる．

実装は，Stan ファイルに data，parameters，model の 3 つを記述する．data は入力とするデータを記述する．model にパラメータからデータへの生成過程 $\boldsymbol{x} \sim p(\boldsymbol{x}|\boldsymbol{\theta}), \boldsymbol{\theta} \sim p(\boldsymbol{\theta})$ を書き，parameters に事後分布を推定したいパラメータを書く．

確率モデルには，ガウス分布やベルヌーイ分布といった通常の確率分布を使うことができる．しかし，確率的ニューラルネットワークで近似した確率分布をモデルに用いることができない．したがって，ベイズ的ニューラルネットワーク（Bayesian neural network）や深層生成モデルの実装をすることはできない．

3.3.3　PyMC3

PyMC3 [Salvatier 16] は，John Salvatier や Thomas V. Wiecki らによって開発が進められている確率モデリングライブラリである．Python で実装されているが，自動微分や高速なサンプリングを実現するために，内部の計算に Theano [Theano Development Team 16] を利用している[*9]．Stan とは異なり，モデルの記述をすべて Python 上で行うことができる．

Stan と同様に，NUTS や ADVI によるサンプリングと変分推論ができるが，PyMC3 ではさらに確率分布のパラメータとして Theano で記述したニューラルネットワークを適用することができる．そのため，ベイズ的ニューラルネットワークや確率的ニューラルネットワークによるベイズ推論を行うことができる．

また，ニューラルネットワークで近似分布をモデル化した学習による変分推論も，ADVI の実装を拡張する形で実行することができる[*10]．しかし，あくまで簡易的なものに過ぎないので，確率変数が複数あるような複雑な深層生成モデルを記述することはできない．また，近似分布にガウス分布以外の確率分布を用いることができない．

[*9] Theano をバックエンドとして使っているものの，現在（2017 年 7 月）のところ，GPU での計算は公式にサポートされていない．また Tran らによると，すべての計算に Theano が使われているわけではないという [Tran 17a].

[*10] local_rv というオプションで近似分布であるガウス分布のパラメータを指定する．

3.3.4 Edward

確率モデリング言語の中で，今最も注目を集めているライブラリが Edward [Tran 17a] である．Edward は Tran らによって開発が進められており，PyMC3 と同様 Python で実装されている．バックエンドには Tensorflow が使われており，GPU による高速計算もサポートしている．

Edward はモデル，推論，評価の 3 つの段階で構成されており（Box's loop と呼ばれる），モデルを設計した後，データとモデルから推論を行い，そのモデルを評価してまた設計する，という流れを実行することができる．生成モデルは確率分布の他に Tensorflow [Abadi 15] で記述したニューラルネットワークを混ぜて記述することができる．そのため，複雑な生成モデルを容易に記述することができる．

推論は，変分推論とモンテカルロサンプリングの両方が実装されている．特に変分推論は，PyMC3 と異なり，KLqp というクラスに近似分布を与える仕様になっている．そのため，近似分布は生成モデルと同様，任意の確率分布やニューラルネットワークで柔軟に記述することができる．また変分推論以外にも EP 法や MAP 推定も実装できるが，それぞれ KLpq と MAP クラスを用いて同じ枠組みで記述することができる．

このように Edward は確率モデリングのためのライブラリとして必要な機能をほぼすべて備えており，しかもこれまでのライブラリと比較しても比較的簡単に記述することができる．しかし，複雑な深層生成モデルを実装する上ではいくつか問題がある．たとえば，近年の VAE モデルでは，深層ニューラルネットワークによって記述された確率分布が多層になったり，複雑な形をとることが多い [Sønderby 16, Maaløe 16]．特に本研究で提案するマルチモーダル学習のための深層生成モデルは，モダリティごとに異なる生成モデルや推論分布を持つ形になっている．また Edward では，個々の分布を別々に書く仕様にはなっていないため，この場合，個々の尤度計算ができなくなり，下界を求められなくなってしまう．さらに利用の観点からは，個々の確率分布から推論したり生成したりすることが求められるが，Edward では上記の理由により不可能となる．

第 4 章

深層学習と生成モデルによるマルチモーダル学習

4.1　関連研究を踏まえた本論文の目標

2 章では，深層ニューラルネットワークと生成モデル，深層生成モデルに関する前提知識について説明した．そして 3 章では，マルチモーダルの定義を明確にし，特徴空間の違いだけでなく，分布の違いを考慮すべきことを指摘した．そして，マルチモーダル学習における各問題設定の関連研究について概観した．

表現の問題設定では，良い共有表現を得るためには，深層ニューラルネットワークでは異なる分布を考慮できないため，DBM などの生成モデルが良いとされる一方で，高次元のデータを扱えないという問題があることを確認した．また共有表現以外にも，異なるモダリティネットワークを近づける座標表現という方法があるが，こちらも分布の違いを考慮できないことを確認した．

変換の問題設定では，まず辞書を利用した事例ベースと呼ばれる手法があり，中間意味表現上への写像が学習できれば，双方向に変換できることを確認した．しかし，メモリコストや検索時間が膨大になるといった問題や，辞書による決定論的な関係が必ずしも自明でないことをみた．その一方で生成によるアプローチ，特に深層生成モデルを用いるアプローチでは，辞書を用いずに異なるモダリティを 1 から生成できるものの，1 方向でしか変換できないことを確認した．

融合の問題設定では，深層学習の手法が主流になっているものの，ラベルありデータが大量に必要であることが問題であり，半教師あり学習による融合モデルが提案されていることを確認した．一方で，近年は深層生成モデルによる半教師あり学習も提案されており，他の半教師あり学習と異なり end-to-end で学習できるものの，半教師ありマルチモーダル学習のためのモデルはまだ存在しないことを指摘した．

共学習の問題設定では，ゼロショット学習というタスクがあり，特に属性ベースゼロショット学習では，属性空間に画像情報を写像することで学習できることをみた．しかし既存研究である DAP モデルでは，属性と画像の分布の違いを考慮していないことを指摘した．

このように，マルチモーダル学習では深層学習が多く使われるようになっている一方で，マルチモーダル学習の課題として挙げられているものの多くが，モダリティ間の分布の違いに影響していることがわかる．1 章で説明したように，この問題は生成モデルで解決できる可能性がある．

3 章で説明したように，深層学習と生成モデルを用いたマルチモーダル情報の学習の研究は幾つか行われているが，上記の様々な問題設定において検証した研究はない．その理由として，まず深層学習と生成モデルを組み合わせた研究が本格的に行われるようになってから，未だ年月が経っていないことがあげられる．特に，SGD で学習できる深層生成モデルは 2014 年頃に提案されていたが [Kingma 13, Goodfellow 14]，それらが小規模の自然画像データを入力として学習できるようになったのが 2015~2016 年頃である [Radford 15]．深層生成モデルの研究は他の関連研究と比較しても発展が速く，2017 年に入ってからは深層生成モデルの一手法の GAN に関する研究が爆発的に増えている[*1]ものの，それでも理論の完成には程遠く，既存のモデルの改良論文が大量に溢れている状況である．また，実際のデータに用いるような応用研究でも，今あるデータをどうモデルに適用するかを議論している段階であり，それらに比べると深層生成モデルを用いたマルチモーダル学習に関する研究はほとんどないのが現状である．

上記の検証が進んでいないもう 1 つの理由は，深層生成モデルを実装するための枠組みが十分に整備されていないことが挙げられる．3.3 節で述べてたように，これまでも深

[*1] The GAN zoo（https://github.com/hindupuravinash/the-gan-zoo）というページには，2017 年前後からの GAN に関する arXiv に上がった論文の一覧が公開されており，2017 年 12 月現在まで指数関数的な勢いで論文が出ていることがわかる．

層ニューラルネットワークを含めた確率分布を実装するライブラリは複数提案されており [Salvatier 16, Tran 17a]，様々なモデルの実装を行うことができる．しかし，これらは深層生成モデルの実装に特化している訳ではない．近年の深層生成モデルは複数の確率分布で複雑に構成されており [Sønderby 16, Maaløe 16]，特に本研究のように複数のモダリティ情報を扱う場合には，それらを記述するために必要な確率分布はさらに増える．従来の確率モデリング言語は，それぞれの尤度を計算する仕様ではないため，これらのモデルの実装には対応していない．さらに，開発した生成モデルを様々なアプリケーションで利用するためには，学習した確率分布を簡単に保存したり読み込んだりできる必要がある．これらのライブラリではネットワークを保存することは可能だが，確率分布自体を保存したり，後から読み込んでサンプリングすることは困難である．

以上のことから，本研究では次の 2 つを達成することを目標とする．

1. マルチモーダル学習のそれぞれの問題設定において，深層学習と生成モデルを利用したモデルを提案して，有効性を検証する．
2. マルチモーダル情報を含んだ深層生成モデルを，実装・利用するためのライブラリを開発する．

4.2　本章以降の位置付け

以降の章では，上記の目標 1 が 5 章から 7 章にあたり，目標 2 が 8 章に該当する．図 4.1 は，マルチモーダル学習の諸問題とそれに取り組んだ章（研究）の対応関係を表したものである．

5 章から 7 章では，マルチモーダル学習の 4 つの問題設定のそれぞれにおいて，深層学習と生成モデルを用いたモデルを提案する．2.3 節で述べた通り，深層学習と生成モデルを結びつける方法は大きく分けて 2 種類ある．本論文では，2 種類の利点を考慮し，モダリティや問題設定の関係性の設定に応じて，使い分けることにする．図 4.1 では，それぞれの研究がどのアプローチを用いたかを示している．

なお，1.1.1 節でも少し述べたように，本論文を通して，扱うモダリティの数は同時に 2 つに限定し，各モダリティ情報も，画像とタグ情報のような比較的簡単なものとしている．これは，

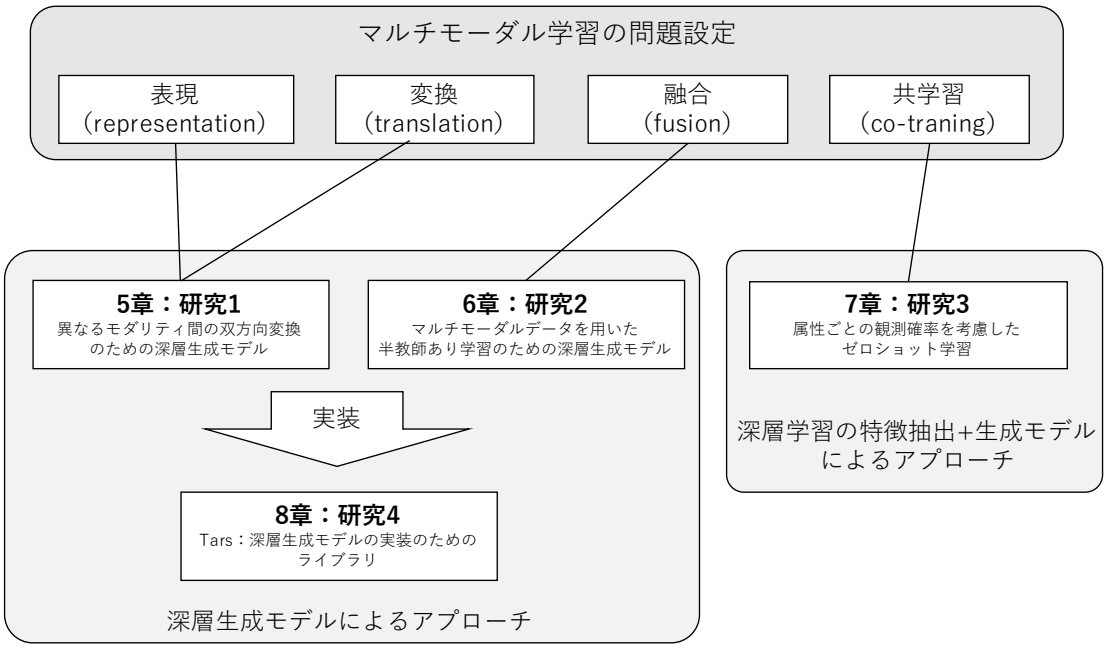


図 4.1　本論文の各章（5 章から 8 章）のマルチモーダル学習の問題設定との関係.

データ集合はなるべく簡素にして様々な実験を行えるようにするためである．また，3.2 節で述べたように，本論文では系列情報は扱わないので，マルチモーダル学習のもう 1 つの問題設定であるアラインメントについては考慮しない．系列情報を含んだマルチモーダル学習については，考察（9 章）で議論する．

5 章では，異なるモダリティを双方向に変換する問題に取り組む．この研究では，辞書を用いた決定論的なモデルではなく，深層生成モデルを用いた生成によるアプローチを採用する．方法としては，表現の問題設定と同様に，異なるモダリティのネットワークを結合することで，深層生成モデルで共有表現が獲得できるようにする．良い共有表現が獲得されれば，生成したいモダリティを欠損させることによって他のモダリティから補完され，双方向の変換が可能となる．しかし欠損させるモダリティの情報量が大きい場合は，共有表現が崩れ，適切に補完できないことが実験によりわかった．本研究ではこれを改善する手法を提案し，モダリティが欠損しても適切に表現が獲得できること，1 方向のモデルと同等以上の精度で双方向にモダリティが変換できることを示す．

6 章では，融合問題，特に半教師ありマルチモーダル学習に取り組む．この章では深層生成

モデルを用いた半教師ありマルチモーダル学習の手法の他に，テスト集合で片方のモダリティのみが与えられる場合でも精度が落ちない手法を提案する．単一モダリティ・マルチモーダルの両方で，既存手法と比較して精度の高い半教師あり学習ができることを示す．

7 章では，属性というモダリティ情報を補助情報として利用した属性ベースゼロショット学習に取り組む．従来の属性ベースゼロショット学習では，分布の違いを考慮せずに，画像特徴量を属性空間に写像していた．本研究では，属性の画像に対する影響（観測確率）を考慮した生成モデルを提案することで，従来の手法よりも正解率が向上することを確認する．なおこの研究では，目標カテゴリ・属性・画像間の明示的な有向関係を表現するために，画像の特徴抽出に深層ニューラルネットワークを利用し，それぞれの関係は生成モデルで構築している．

8 章では，深層生成モデルに特化したライブラリ Tars を開発する．マルチモーダル情報を含むような複雑な深層生成モデルの実装・利用のためのライブラリ開発という目標を達成するために，ネットワークが確率分布に隠蔽される枠組みを提案する．これによって，ネットワークや確率分布の形に依存せずにサンプリングや尤度計算を行うことが可能となる．また，それぞれの確率分布はクラスとして独立しているので，それぞれを保存して，アプリケーション用にそのまま利用することができる．これらのことを，実験と応用例から確認する．

第 5 章

異なるモダリティ間の双方向変換のための深層生成モデル

本章では，異なるモダリティ間で双方向に変換できるような深層生成モデルを提案する．

深層生成モデルによって異なるモダリティを双方向に変換する方法として，各モダリティの最も上位の隠れ層を共有する共有表現のアプローチを選択する．[Srivastava 12] で示されているように，適切な共有表現を獲得できれば，共有表現を介して双方向にモダリティを変換することができる．双方向変換のためのもう 1 つの単純な手法として，1 方向で変換するネットワークを別々に学習するということが考えられる．モダリティを 1 方向で変換する深層生成モデルについては，これまでも複数提案されている [Kingma 14a, Sohn 15, Pandey 16]．しかし，モダリティを双方向に変換する場合，このアプローチでは，必要なネットワークの数がモダリティの数に対して指数関数的に増大してしまう．さらに，それぞれの方向のネットワークは独立に学習されるので，隠れ層は共有されず，それぞれ異なる表現が獲得されてしまう．したがって，双方向に変換するという目標に対して，この単純な方法はあまり適さない．

異なるモダリティを変換するためには，共有表現を確率的な潜在変数としてモデル化することが重要である．これは 1 章や 3 章で述べたように，異なるモダリティは異なる分布を持つため，それらの関係性は決定論的にならないためである．これを実現する手法としては，深層生成モデルの deep Boltzmann machine（DBM） [Srivastava 12, Sohn 14] を用いたモデルが知られている．しかし，DBM の学習則は MCMC 法に基づいており，大規模で高次元なデータを入力として学習するのは困難である．

近年，変分推論によって柔軟に深層生成モデルを学習できるモデルとして，variational autoencoder（VAE） [Kingma 13, Rezende 14] が提案されている（2.4.1 節）．このモデルは通常の深層ニューラルネットワークと同様，学習時に誤差逆伝播法を用いることができるため，従来の DBM のような MCMC 法による学習モデルに比べて，大規模で高次元のデータ集合を学習することができる．本研究では，まず共有表現のアプローチを VAE に適用することで，マルチモーダルデータを扱えるようにする．このモデルを joint multimodal variational autoencoder（JMVAE）と呼ぶ．確率モデルとして考えると，JMVAE は潜在変数の下での各モダリティの条件付き分布によって，全モダリティの同時分布を構成している．この同時分布から，あるモダリティの下での別のモダリティの条件付き分布を求めることができるので，これを用いて双方向にモダリティを変換することができる．

あるモダリティから対応する別のモダリティを変換するとき，変換先の生成するモダリティは入力では欠損として扱われる．しかし，欠損モダリティの次元が他のモダリティの次元よりも大きい場合，潜在表現や生成したサンプルが崩れてしまう可能性がある．本章では，この問題が実際に生じることを実験的に確認し，従来提案されていた欠損値補完の手法を単純に用いるだけでは解消できないことを示す．本研究では，この問題を解決するための追加的な手法として，階層的 JMVAE と JMVAE-kl という 2 つの異なる手法を提案する．階層的 JMVAE は，潜在変数を確率的多層構造にすることで，潜在表現の崩壊を防ぎ，適切なサンプルを生成できる．JMVAE-kl は，本研究で新たに提案するアプローチで，各モダリティを単一で入力とする新たな近似分布を用意し，全モダリティを入力とする JMVAE の近似分布との距離を近づけることで学習する．これらの手法によって，欠損モダリティによる問題は解消され，異なるモダリティ間を双方向に変換できるようになる．図 5.1 では，JMVAE によって共有表現となる潜在変数を介して，画像から属性，属性から画像のように，次元も構造も異なるモダリティ間を双方向に変換できることを示している．さらに画像は属性よりも情報量が大きいので，同じ属性から対応する複数の画像を生成することができる．

本研究の主な貢献は以下の通りである．

- VAE でマルチモーダル情報を統合するモデルである JMVAE では，入力で欠損させるモダリティが他のモダリティよりも次元が大きい場合，変換先の生成したサンプルが崩れてしまう問題があること明らかにし，従来の欠損値補完手法を単純に用いるだけでは

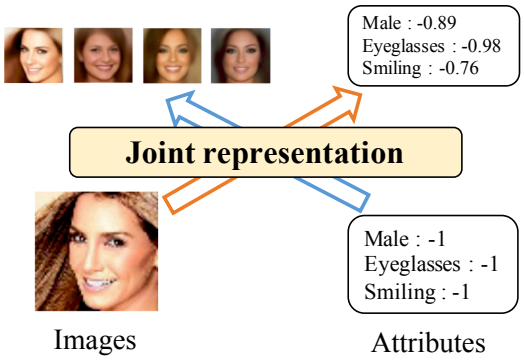


図 5.1　JMVAE による，共有表現を介した異なるモダリティ間の双方向生成.

解決できないことを実験的に示す.

- 欠損モダリティ問題を解決するために，階層的 JMVAE 及び JMVAE-kl という提案手法を導入する.
- 上記の手法によって欠損モダリティ問題が解決し，すべてのモダリティを統合した共有表現が獲得され，適切に異なるモダリティ間の双方向の変換ができることを定量的・定性的実験によって示す.

5.1　関連研究

本節では，深層生成モデルを用いた 2 つのモダリティ間の変換に関する関連研究を述べる.

Sohn らは，あるモダリティが別のモダリティによって条件づけられた深層生成モデルである conditional VAE（CVAE）を提案した [Sohn 15]（2.4.2 節）．このモデルは，変分推論によって条件付き尤度を最大化するように学習され，条件づけられたモダリティからもう片方のモダリティを生成することができる．また，各モダリティの分布の違いを考慮しているため，たとえば，数字ラベルから対応する複数の手書き数字を生成することができる [Kingma 14a]．このアプローチは，他のモダリティでも用いられ，回転角度から物体画像 [Kulkarni 15]，属性から顔画像 [Larsen 15b, Yan 15]，そしてキャプションから画像 [Mansimov 15]，といったものがある．CVAE の最も大きな特徴は，モダリティ間の変換が 1 方向であること，そして

潜在変数が条件づけられたモダリティの情報を含んでいないことである[*1]．このため，CVAE は双方向に変換できず，複数のモダリティを統合した共有表現は獲得されない．

Pandey らは CVAE と同様，VAE で条件付き対数尤度を最大化するモデルとして conditional multimodal autoencoder（CMMA）を提案している [Pandey 16]．CVAE との違いは，潜在変数が 2 つのモダリティに対応する変数と接続しているため，その潜在変数で 2 つのモダリティ情報を統合した共有表現が得られる可能性があるということである．しかし，CMMA も 1 方向でしか変換できない．

一方，GAN [Goodfellow 14] は生成分布の尤度を明示的に指定せずに学習できるため，VAE よりも鮮明な画像を生成できることが特徴である．CVAE のように，別のモダリティで条件づけた conditional GAN（CGAN） [Mirza 14] を用いた研究が一般的で，キャプションから画像を生成する研究も複数提案されている [Reed 16]．しかし，この手法も条件付き分布をモデル化しているので，1 方向でしか変換できない．

最近では，画像から画像について双方向に変換できる GAN によるモデルが提案されている [Liu 16, Liu 17, Zhu 17]．これらのモデルは，同一サイズの画像間のピクセル単位での完全な対応関係をモデル化しており，生成の際に確率的要素はなるべく無視する傾向にある．しかし，3.1 節で明確にしたように，異なるモダリティとは特徴空間が異なる表現のことであるため，ピクセル単位で完全な対応関係がつくれる訳ではない．

5.2　マルチモーダル情報のための VAE

本節では，2.4.1 節で説明した VAE をマルチモーダルデータに拡張した JMVAE について述べる．そして，JMVAE の片方の入力が欠損した場合の推定方法として，新たに JMVAE-kl と階層的 JMVAE を提案する．

5.2.1　Joint multimodal variational autoencoder

データ集合 $\{(\boldsymbol{x}_1, \boldsymbol{w}_1), ..., (\boldsymbol{x}_N, \boldsymbol{w}_N)\}$ を考える．ただし，$\boldsymbol{x}$ と $\boldsymbol{w}$ はそれぞれ異なる種類の次元や分布をもつとし，それぞれを異なるモダリティとする．データ集合の各事例のモダリ

[*1] Louizos らは，CVAE のエンコーダには潜在変数と条件づけられた変数との間に依存関係が残っているため，潜在変数と条件づけられた変数は厳密に独立になる訳ではないと指摘している [Louizos 15]．

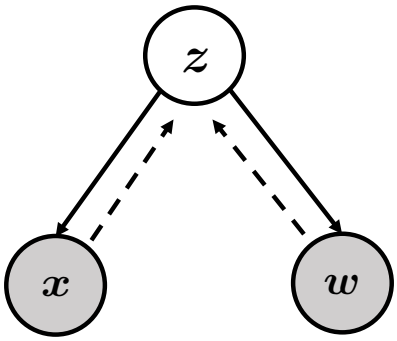


図 5.2　JMVAE のグラフィカルモデル.

ティの組 $(\boldsymbol{x}_i, \boldsymbol{w}_i)$ は同じ対象を表現しているものとする．本研究の目標は，これら 2 種類のモダリティ間を双方向に変換することである．ここでの「双方向に変換する」とは，$\boldsymbol{x}$ から $\boldsymbol{w}$ の生成と $\boldsymbol{w}$ から $\boldsymbol{x}$ の生成の両方を行うことを指す.

ここでは，これらは同一の潜在的な概念 $\boldsymbol{z}$，すなわち，共有表現の下で条件付き独立であるとし，各モダリティは異なる分布から生成されると仮定する．したがって，潜在変数及び各モダリティの生成過程は $\boldsymbol{z} \sim p(\boldsymbol{z})$ 及び $\boldsymbol{x}, \boldsymbol{w} \sim p(\boldsymbol{x}, \boldsymbol{w}|\boldsymbol{z}) = p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z})$ となる.

この過程をグラフィカルモデルで表したものが図 5.2 である.

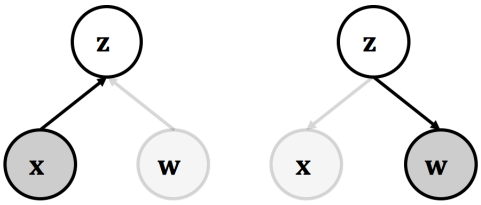
このモデルはすべてのモダリティの同時分布（**joint** distribution）をモデル化していることから，本研究ではこのモデルを **Joint Multimodal Variational AutoEncoder**（**JMVAE**）と呼ぶ.

近似事後分布を $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})$ とすると，対数尤度 $\log p(\boldsymbol{x}, \boldsymbol{w})$ の変分下界は次のようになる.

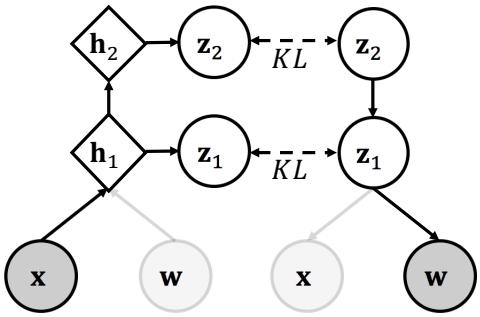
$$\begin{aligned}\mathcal{L}_{JM}(\boldsymbol{x}, \boldsymbol{w}) &= E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})}] \\ &= -D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})||p(\boldsymbol{z})) + E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})] + E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z})].\end{aligned} \tag{5.1}$$

この式には，各モダリティに対応した 2 つの負の再構成誤差項がある．VAE と同様に $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})$ をエンコーダ，$p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ と $p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z})$ をデコーダと呼ぶ.

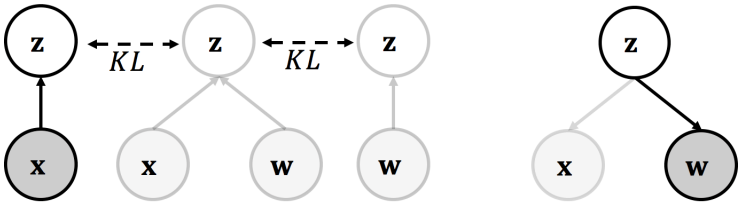
式 (5.1) のエンコーダとデコーダは，深層ニューラルネットワークでパラメータ化し，VAE



(a) JMVAE



(b) Hierarchical JMVAE



(c) JMVAE-kl

図 5.3　(a) JMVAE，(b) 階層的 JMVAE，及び (c) JMVAE-kl の推論分布（エンコーダ，左）と生成分布（デコーダ，右）．各手法での $q(\boldsymbol{z}|\boldsymbol{x})$ と $p(\boldsymbol{w}|\boldsymbol{z})$ のモデル化を表している．丸は確率変数，菱形は決定論的変数を表す．

と同様に各パラメータ（$\boldsymbol{\theta}$ と $\boldsymbol{\phi}$）に関して最適化できる．各モダリティは異なる特徴表現を持つので，デコーダ $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ と $p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z})$ に対して異なる分布や異なる構造のネットワークを設定する必要がある．分布やネットワーク構造の種類は，データ集合における各モダリティに依存する．たとえば，あるモダリティの事例の各次元要素が連続値をとるならばガウス分布，2 値をとるならばベルヌーイ分布，2 値かつ one-hot ならばカテゴリ分布となる．エンコーダ $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})$ はガウス分布とし，各モダリティに対して異なるネットワークを用意し，最終層で結合することでパラメータ化する．

JMVAE は，CVAE や CMMA とは異なり全モダリティの同時分布をモデル化し，各モダ

リティは潜在変数の下で条件付き独立となっている．そのため，すべてのモダリティを含んだ共有表現を抽出できることが期待される．さらに，同時分布について各モダリティで条件付けることで，双方向の条件付き分布が得られるため，テキストから画像，画像からテキストのように，双方向のモダリティの変換も可能となる．加えて JMVAE は $p(\boldsymbol{x}, \boldsymbol{w}_1, \boldsymbol{w}_2, ...)$ のように 3 つ以上のモダリティも入力として扱うことができる．

5.2.2 欠損モダリティの推定

JMVAE では，訓練後のテスト段階に，エンコーダ $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})$ をつかって，複数のモダリティから共有された潜在表現を推論できる．本章の目標は双方向に異なるモダリティを変換することなので，対応する変換先の生成したいモダリティの事例は手元にない．図 5.3(a) は JMVAE で $\boldsymbol{x}$ から $\boldsymbol{w}$ を生成する流れを示しており，エンコーダにおいて $\boldsymbol{w}$ は欠損しているものとして扱われる．従来の識別的なマルチモーダル学習の設定でも，あるモダリティから別のモダリティを推定する場合は，0 やランダムなノイズが設定される [Ngiam 11]．

VAE において，入力が欠損している場合の補完方法としては，遷移カーネルを用いたマルコフ連鎖による反復サンプリングの方法が提案されている [Rezende 14]．JMVAE の場合，$\boldsymbol{x}$ が欠損したときの遷移カーネル $T(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{w})$ は次のようになる．

$$T(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{w}) = \int p(\tilde{\boldsymbol{x}}|\boldsymbol{z}) q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}) d\boldsymbol{z}. \tag{5.2}$$

$\boldsymbol{x}$ の初期値を $\boldsymbol{x} \sim p(\boldsymbol{x})$ のようにランダムなノイズとし，式 (5.2) を用いて反復的にサンプリングすることで，欠損モダリティを推定できる．本章では，この手法を反復サンプリング手法と呼ぶ．

欠損モダリティが，他のモダリティと比べて高次元で複雑な構造の場合，エンコーダによって推論された潜在変数は不完全となり，デコーダで生成（補完）したサンプルは崩れてしまう可能性がある．本章では，単純に反復サンプリング手法を用いるだけでは，この現象は防げないことを実験で示す．

本研究では，この問題を解決するために，階層的 JMVAE と JMVAE-kl という 2 つの異なる提案手法を導入する．

5.2.3　階層的 JMVAE

近年，VAE の潜在変数を確率的な階層構造に拡張して，モデルの表現力や尤度を向上させる手法がいくつか提案されている [Burda 15, Sønderby 16, Gulrajani 16]．本研究で提案する JMVAE は，すべてのモダリティの情報を統合した潜在変数をモデル化しているため，確率的階層構造に容易に拡張できる．潜在変数を L 層の階層 $\boldsymbol{z}_1, ..., \boldsymbol{z}_L$ とすると[*2]，JMVAE の同時分布は次のようになる．

$$p(\boldsymbol{x}, \boldsymbol{w}) = \int ... \int p(\boldsymbol{z}_L) p_{\boldsymbol{\theta}}(\boldsymbol{z}_{L-1}|\boldsymbol{z}_L) ... p_{\boldsymbol{\theta}}(\boldsymbol{z}_1|\boldsymbol{z}_2) p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_1) p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z}_1) d\boldsymbol{z}_1 ... d\boldsymbol{z}_L. \quad (5.3)$$

ただし，それぞれの条件付き確率分布 $p_{\boldsymbol{\theta}}(\boldsymbol{z}_{l-1}|\boldsymbol{z}_l)$ はすべてガウス分布とし，深層ニューラルネットワークによってパラメータ化されるとする．

階層的な潜在変数をもつ VAE における，近似分布の因数分解の方法は様々提案されているが，本章では Gulrajani らの分解方法 [Gulrajani 16] に従った．この方法では，近似分布は次のように分解される．

$$q(\boldsymbol{z}_1, ..., \boldsymbol{z}_L|\boldsymbol{x}, \boldsymbol{w}) = q_{\boldsymbol{\phi}}(\boldsymbol{z}_1|\boldsymbol{x}, \boldsymbol{w}) \dots q_{\boldsymbol{\phi}}(\boldsymbol{z}_L|\boldsymbol{x}, \boldsymbol{w}). \quad (5.4)$$

各条件付き分布はガウス分布とし，式 (5.4) からそれぞれ独立となっている．本研究では，Gulrajani らの手法と同様，入力から最終層まで決定論的な写像で構成され（それぞれの写像は深層ニューラルネットワークでパラメータ化される），各階層の決定論的な出力 $\boldsymbol{h}_l$ から確率的な出力 $\boldsymbol{z}_l$ が得られるとした（図 5.3(b) を参照）．したがって，確率的階層化した JMVAE の下界は次のようになる．

$$\begin{aligned} \mathcal{L}_{JM_h}(\boldsymbol{x}, \boldsymbol{w}) = &- \sum_{l=1}^{L} E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_{l+1}|\boldsymbol{x},\boldsymbol{w})}[D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}_l|\boldsymbol{x}, \boldsymbol{w})||p_{\boldsymbol{\theta}}(\boldsymbol{z}_l|\boldsymbol{z}_{l+1}))] \\ &+ E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_1|\boldsymbol{x},\boldsymbol{w})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_1)] + E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_1|\boldsymbol{x},\boldsymbol{w})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z}_1)]. \end{aligned} \quad (5.5)$$

第 1 項は，エンコーダとデコーダの各確率的層間のカルバック・ライブラー（KL）ダイバージェンスを最小化するようになっている．

潜在変数の確率的階層化によって，式 (5.4) のように複雑な近似分布を構成することができるので，より入力での欠損に対して頑健な潜在表現が得られ，さらに反復サンプリングに

[*2] ここでの確率的階層は，深層ニューラルネットワークにおける決定論的な階層構造とは異なる．

よって適切な欠損モダリティの生成が可能となると期待される．本章ではこの手法を階層的 JMVAE（hierarchical JMVAE）と呼び，実験を通して，本手法が欠損モダリティの問題を緩和できることを示す．図 5.3(b) は階層的 JMVAE で $\boldsymbol{x}$ から $\boldsymbol{w}$ を生成する流れを示している．

5.2.4　JMVAE-kl

確率的階層構造の手法では，欠損モダリティの生成のために，依然として反復サンプリング手法が重要となる．しかし高次元のサンプルを生成する際には，時間がかかってしまうという問題がある．したがって，反復サンプリング手法を用いずに適切な欠損サンプルを生成できる手法として，JMVAE-kl を提案する．

単一のモダリティ入力をとるエンコーダ $q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w})$ を考える（ただし λ はモデルパラメータ）．もしこれらを適切に得ることができれば，これらのうち生成元のモダリティに対応する分布を使って $\boldsymbol{z}$ を直接推論できる．たとえば，テスト時に $\boldsymbol{x}$ のみから潜在変数を推論したい場合は，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$ をつかって $\boldsymbol{z} \sim q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$ のように推論できる (図 5.3(c) 左参照)．この際，入力に欠損値をとらないので，階層的 JMVAE と異なり，反復サンプリングをせずに異なるモダリティ間を双方向に変換することが可能となる．

JMVAE-kl では，単一のモダリティ入力をとるエンコーダを JMVAE のエンコーダ $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})$ に近づけるようにして学習する (図 5.3(c) 左参照)．分布間の距離をカルバック・ライブラー距離とすると，JMVAE-kl の目的関数は次のようになる．

$$\mathcal{L}_{JM_{kl}}(\boldsymbol{x},\boldsymbol{w}) = \mathcal{L}_{JM}(\boldsymbol{x},\boldsymbol{w}) - [D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})) + D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w}))]. \tag{5.6}$$

他の観点からみると，式 (5.6) を最大化することは，パラメータ化された分布において変分推論による variation of information（VI）の最小化とみなせる．証明は付録 A.2 を参照されたい．VI は 2 つの変数間の距離を計測する指標であり，$p_{\mathcal{D}}$ をデータ分布とすると $-E_{p_{\mathcal{D}}}[\log p(\boldsymbol{x}|\boldsymbol{w}) + \log p(\boldsymbol{w}|\boldsymbol{x})]$ のように，2 つの負の条件付き対数尤度の和で表される．この指標をを最小化するということは，双方向にモダリティ間の変換が適切に行われるように学習していることになる．Sohn らによるマルチモーダル学習の手法では，VI 最小化によってモデルを学習する方法を取っている [Sohn 14]．しかし，彼らの手法では MCMC 法によって学習する DBM をモデルとして用いているため，本研究のように自然画像のような次元の大きい

データを直接学習することは困難である．

JMVAE-kl は，モダリティの数が増えると，元の JMVAE のエンコーダのネットワークが増えるだけでなく，モダリティが欠損した場合を考慮したエンコーダも必要となるので，ネットワーク数が膨大になってしまうという問題がある．モダリティが 2 個の場合，JMVAE-kl では 5.2.4 節で述べたように，元の JMVAE のエンコーダ $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})$ と各モダリティが欠損した場合のエンコーダ $q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w})$ が必要となる．ここで，各モダリティから潜在変数への写像を表現したネットワークがそれぞれ同じパラメータ数であると仮定し，そのパラメータ数を M とする．すると，$q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})$ では $2M$，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$ と $q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w})$ ではそれぞれ M のパラメータが必要なので，エンコーダに必要なパラメータ数は合計 $4M$ となる．モダリティが 3 個になると，元の JMVAE のエンコーダ $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w}_1,\boldsymbol{w}_2)$ の他に，全モダリティのすべての欠損の組み合わせを考慮したエンコーダ $q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w}_1)$，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w}_2)$，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w}_1)$，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w}_1,\boldsymbol{w}_2)$，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w}_2,\boldsymbol{x})$ が必要になる．よって，ネットワークのパラメータ数の合計は $12M$ になる．モダリティ数が K 個の場合では，エンコーダのパラメータ数は $2^{K-1}KM$ となり，モダリティ数に対してネットワークのパラメータ数が指数的に増大してしまう．

一方階層的 JMVAE では，モダリティの数が増えても，対応する元の JMVAE のエンコーダのネットワークが増えるだけである．階層的 JMVAE のエンコーダが式 (5.4) のように因数分解され，それぞれの確率的階層間の決定論的な写像がパラメータ数 M のネットワークで表現されていると仮定すると，K 個のモダリティにおける L 層の階層的 JMVAE では，エンコーダのパラメータ数が合計 $(K+L-1)M$ となる．よって階層的 JMVAE では，JMVAE-kl と比べて，モダリティの数に対するパラメータ数を大幅に抑えられる．

したがって, 階層的 JMVAE と JMVAE-kl にはそれぞれ長所と短所がある．階層的 JMVAE はモダリティが 3 種類以上で欠損モダリティの次元がそれほど大きくない場合に有効で，JMVAE-kl はモダリティが 2 種類のみで欠損モダリティが高次元の場合に有効であるといえる．

5.3 実験

5.3.1 データ集合

本実験の目的は，(1) 欠損モダリティ問題が確かに発生し提案手法でそれが改善されること，(2) 異なるモダリティを統合した共有表現が獲得されていること，(3)1 方向の変換と同等（もしくはそれ以上）の精度で双方向の変換ができること，の 3 点について確認することである．

既存のマルチモーダル学習のデータ集合としては，MIR Flickr25k [Huiskes 08] や Washington RGB-D データ集合 [Lai 11] などがある．しかし，これらのデータ集合は異なるモダリティでも次元や構造が同じである例が多く，特に RGB-D データ集合は物体画像の RGB 情報と深さの情報を異なるモダリティとしていて，画像サイズはモダリティ間で同じとなっている．よって (1) を検証するには不向きである．一方，次元や構造が異なっている例として，一般物体画像とタグを異なるモダリティとする MIR Flickr25k などがあるが，一般物体画像の生成自体が今現在も困難なタスクであるため (2) や (3) の検証に向かない．そもそも，通常のマルチモーダル学習では，通常複数のモダリティからより良い表現を獲得して識別精度を高めることが目的であり，本研究の目的とは異なる．そのため，既存のマルチモーダル学習のデータ集合は本実験では用いないこととした．

その代わり，本実験では上記の目的を達成するため，MNIST と CelebA [Liu 15] の 2 つのデータ集合を用いることにした．

MNIST は，本来マルチモーダル学習のためのデータ集合ではない．しかし，ラベルを one-hot のタグと考えて 1 つのモダリティとすると，手書き数字画像とラベルでは次元や構造が大きく異なること，one-hot の情報なので潜在空間で多様体学習ができているか判断しやすいこと，そしてデータ集合のサイズが小さく生成しやすいことから，上記の目的に適していると考えられる．前処理として，数字画像の各ピクセル値は $[0, 1]$ になるよう正規化した．全データ集合のうち 50,000 を訓練集合とし，残りの 10,000 をテスト集合とした．

CelebA は 202,599 枚のカラー顔画像と対応する 40 の 2 値属性（男性，メガネ，髭など）によって構成される，より一般的なマルチモーダルデータ集合である．モダリティ間で次元や構造がより大きく異なるため，MNIST よりも困難な設定だが，画像は顔に限定されているため，一般物体画像などと比べて生成しやすいデータ集合と考えられる．前処理として，各画像を顔

を中心に正方形に切り取り，64 × 64 にリサイズしたあと標準化した．本実験では，OpenCV によって顔を特定できた 191,899 をデータ集合として用いた．全データ集合のうち 90% を訓練集合とし，残りの 10% をテスト集合とした．

5.3.2　モデル構造

MNIST

数字画像を $\boldsymbol{x} \in [0,1]^{784}$，対応するラベルを $\boldsymbol{w} \in \{0,1\}^{10}$ とした．ここでは構造の表記のため，出力が k ユニットの線形全結合層-ReLU（正規化線形関数）を `DkR`，`DkR` から ReLU を除いた構造を `Dk` とした．また，2 つのネットワーク `I`，`J` の最終層を連結して 1 つの層とする場合は `(I,J)` と表記する．これは深層ニューラルネットワークでは concatenate と呼ばれる処理で，本研究では複数のモダリティのネットワークを結合する場合に用いる．

エンコーダの式 (2.16) の f_{MLP} を `(D512R-D512R, D512R-D512R)` とし，$f_{\boldsymbol{\mu}}$ と $f_{\boldsymbol{\sigma}^2}$ は，それぞれ `D64` とした．デコーダは $p(\boldsymbol{x}|\boldsymbol{z})$ をベルヌーイ分布，$p(\boldsymbol{w}|\boldsymbol{z})$ をカテゴリ分布とした[*3]．いずれのモダリティについても f_{MLP} を `D512R-D512R` とし，$p(\boldsymbol{x}|\boldsymbol{z})$ の $f_{\boldsymbol{\mu}}$ は `D784`，$p(\boldsymbol{x}|\boldsymbol{z})$ の $f_{\boldsymbol{\mu}}$ は `D40` とした．

本実験では階層的 JMVAE は 2 層まで（$L = 2$）とし，2 層目のエンコーダとデコーダはいずれも f_{MLP} が `D512R-D512R`，$f_{\boldsymbol{\mu}}$ と $f_{\boldsymbol{\sigma}^2}$ を `D64` とした．JMVAE-kl の $q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$ の $f_{\boldsymbol{\mu}}$ は `D512R-D512R`，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w})$ の $f_{\boldsymbol{\mu}}$ は `D512R-D512R` とし，$f_{\boldsymbol{\mu}}$ と $f_{\boldsymbol{\sigma}^2}$ は，いずれの場合もすべて `D64` とした．

CelebA

顔画像を $\boldsymbol{x} \in \mathcal{R}^{32\times32\times3}$，対応する属性を $\boldsymbol{w} \in \{-1,1\}^{40}$ とする．フィルタのサイズが 4×4 でチャネル数が k，ストライドが 2 の畳込み層-バッチ正規化-ReLU を `CkBR` とし，`CkBR` からバッチ正規化を除いた構造を `CkR`，前述と同じフィルタ構造の逆畳込み層-バッチ正規化-ReLU を `DCkBR`，`DCkBR` からバッチ正規化を除いた構造を `DCkR`，さらに ReLU を除いた構造を `DCk` とする．また，k ユニットの線形全結合層-バッチ正規化-ReLU を `DkBR`，平坦化層を `F` と表記する．

[*3] MNIST は $[0,1]$ の実数値をとるが，訓練時とテスト時は要素の実数値に応じて確率的に 0 または 1 に離散化している．これは，過学習を防ぐ役割も果たしている [Burda 15, Sønderby 16].

エンコーダの f_{MLP} を (C64R-C128BR-C256BR-C256BR-F, D512R-D512BR)-D1024R とし，$f_{\boldsymbol{\mu}}$ と $f_{\boldsymbol{\sigma}^2}$ は，それぞれ D128 とした．デコーダは $p(\boldsymbol{x}|\boldsymbol{z})$ と $p(\boldsymbol{w}|\boldsymbol{z})$ はどちらもガウス分布とした．ただし式 (2.16) のパラメータ化と異なり，$\boldsymbol{\mu}$ は $\mathrm{Tanh}(f_{\boldsymbol{\mu}}(f_{\mathrm{MLP}}(\boldsymbol{z})))$ (ただし Tanh は双曲線正接関数) とし，σ^2 の各要素を 1 に固定した．ネットワーク構造は，$p(\boldsymbol{x}|\boldsymbol{z})$ の f_{MLP} を D4096R-DC256BR-DC128BR-DC64BR，$f_{\boldsymbol{\mu}}$ を DC3 とし，$p(\boldsymbol{w}|\boldsymbol{z})$ の f_{MLP} を D4096R-D512BR，$f_{\boldsymbol{\mu}}$ を D40 とする．

階層的 JMVAE の 2 層目は，MNIST と同じネットワーク構造とした．JMVAE-kl の $q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$ の f_{MLP} は C64R-C128BR-C256BR-C256BR-F-DR1024，$f_{\boldsymbol{\mu}}$ と $f_{\boldsymbol{\sigma}^2}$ はそれぞれ D128 とし，$q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w})$ の f_{MLP} は D512R-D512BR-D1024R，$f_{\boldsymbol{\mu}}$ と $f_{\boldsymbol{\sigma}^2}$ はそれぞれ D128 とした．

5.3.3 学習パラメータ設定

最適化アルゴリズムは Adam [Kingma 14b] を使い，学習率は MNIST で 10^{-3}，CelebA で 10^{-4} とした．式 (5.1) の正則化項によって学習の初期に局所解に陥ることを防ぐために，ウォームアップ法 [Bowman 15, Sønderby 16] を用いた．これは，学習の初期は再構成誤差のみを学習し，N_t エポックまで線形に正規化項を大きくしていく方法である．MNIST では $N_t = 200$ とし，訓練エポック数を $2,000$ とした．CelebA では，$N_t = 20$ とし，訓練エポック数を 50 とした．

モデルの実装には，Theano [Theano Development Team 16] と Lasagne [Dieleman 15] に基づく深層生成モデルライブラリ Tars（8 章参照）を用いた．

5.3.4 評価指標

本実験では，モデルの評価にテスト条件付き対数尤度 $\log p(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{w})$（もしくは $\log p(\tilde{\boldsymbol{w}}|\boldsymbol{x}, \boldsymbol{w})$）を用いた．条件付き対数尤度 $\log p(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{w})$ は，次のように近似できる．

$$\begin{aligned}\log p(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{w}) &\simeq \log \int q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}) p(\tilde{\boldsymbol{x}}|\boldsymbol{z}) d\boldsymbol{z} \\ &\geq \int q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}) \log p(\tilde{\boldsymbol{x}}|\boldsymbol{z}) d\boldsymbol{z} \\ &\simeq \frac{1}{N} \sum_{i=1}^{N} \log p(\tilde{\boldsymbol{x}}|\boldsymbol{z}^{(l)}). \end{aligned} \tag{5.7}$$

ただし，$\boldsymbol{z}^{(l)} \sim q(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w})$ である．式 (5.7) では，Jensen の不等式で下界を求めてからサンプル近似を行っているが，これは対数尤度で直接サンプル近似を行うとサンプル数に対して偏るためである[*4]．この下界は負の再構成誤差と考えることができるので，高い値になるほどモダリティが適切に再構成できることを意味する．本実験では，サンプル数は $N = 10$ とした．

JMVAE では，一方のモダリティしか与えられない場合，即ち生成するモダリティが欠損している場合の条件付き対数尤度 $p(\tilde{\boldsymbol{x}}|\boldsymbol{w})$（または $\log p(\tilde{\boldsymbol{w}}|\boldsymbol{x})$）も考えられる．この対数尤度を求めるためには，近似分布 $q(\boldsymbol{z}|\boldsymbol{w})$（または $q(\boldsymbol{z}|\boldsymbol{x})$）を求め，式 (5.7) のように潜在変数をサンプリングする必要がある．この近似分布を求める方法は，JMVAE の各手法によって異なる．JMVAE や階層的 JMVAE では，生成するモダリティを欠損値として扱うため，まず欠損モダリティの初期値を 0 とする．次に反復サンプリングを複数回行うことで欠損モダリティを補完し，その後補完したモダリティをエンコーダに入力して潜在変数をサンプリングする．このため，条件付き尤度は反復サンプリングの回数や補完能力に依存し，適切に補完ができれば尤度は高くなる．JMVAE-kl では近似分布 $q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{x})$（または $q_{\boldsymbol{\lambda}}(\boldsymbol{z}|\boldsymbol{w})$）を学習の際に求めているので，これを用いて直接条件付き尤度を求める．

本章では便宜上，$\log p(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \boldsymbol{w})$ を**複数**条件付き対数尤度もしくは単に条件付き対数尤度と呼び，$\log p(\tilde{\boldsymbol{x}}|\boldsymbol{w})$ を**単数**条件付き対数尤度と呼ぶ．単数条件付き対数尤度と複数条件付き対数尤度の両方を用いて，既存研究であり条件付き分布をモデル化する CVAE [Kingma 14a, Sohn 15] 及び CMMA [Pandey 16] と比較する．適切に双方向に異なるモダリティを変換できるかを検証する際は，単数条件付き対数尤度を用いる．複数条件付き対数尤度では，モダリティの再構成の精度を評価する．階層的 JMVAE や CVAE，CMMA での単数及び複数条件付き対数尤度の近似方法は付録 A.1 節を参照されたい．

5.3.5　実験 1：MNIST

実験 1-1：欠損モダリティ問題と提案手法による改善の確認

本節では最初に，JMVAE で双方向にモダリティを変換する際，欠損モダリティが他のモダリティと比べて高次元の場合，生成したサンプルが壊れてしまうことを確認する．また，反復

[*4] 文献 [Burda 15] によると，対数尤度のサンプル近似は，サンプル数が十分に大きければ，期待値が真の対数尤度に近づく．一方下界は不偏推定量であるため，サンプル数に対して偏らない．

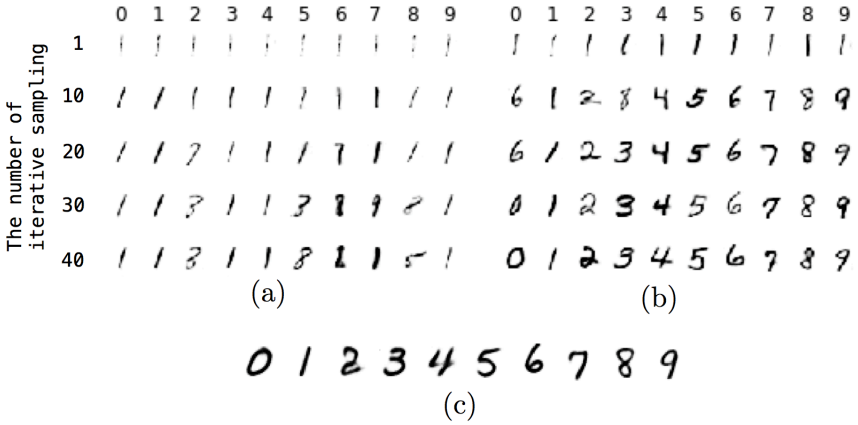


図 5.4 MNIST におけるラベル（$\boldsymbol{w}$）から画像（$\boldsymbol{x}$）の生成．各列は $\boldsymbol{w}$ 空間の各要素，即ち 0 から 9 のラベルに対応している．下の行にいくにつれ，$\boldsymbol{x}$ を生成するための反復サンプリングの回数が増えている．(a) JMVAE．(b) 階層的 JMVAE． (c) JMVAE-kl.

サンプリング手法 [Rezende 14] のような従来の欠損値補完手法を用いるだけでは，この問題は解決できないことを示す．そして，本研究で提案する階層的 JMVAE と JMVAE-kl によってこの問題が解決されることを示す．

図 5.4 (a) は，JMVAE で $\boldsymbol{w}$ から $\boldsymbol{x}$ を生成した結果を示している．一番上の行，つまり 1 回だけ反復サンプリングを適用して生成した画像は，不鮮明で適切に生成されていない．反復サンプリング数が増えるにつれ，多少鮮明になるが，数字画像は明らかにラベルに対応していない．この結果から，反復サンプリング手法でも欠損モダリティの問題は解決できないことがわかる．

次に，階層的 JMVAE と JMVAE-kl が，この問題を解決できることを示す．図 5.4 (b) は階層的 JMVAE の場合の結果を示している．反復サンプリングが 1 回の場合は，(a) と同様ラベルに対応した数字が生成されていないが，サンプリング数を増やすと，ラベルにほぼ対応した数字画像が生成されることが確認できる．図 5.4 (c) は，JMVAE-kl の場合の結果である．5.2.4 節で述べたように，JMVAE-kl の場合は反復サンプリングせずに，数字に条件づけられた数字を生成することができる．また，JMVAE-kl は階層的 JMVAE よりもはっきりとした数字を生成できることが確認できる．

図 5.5 は，JMVAE，階層的 JMVAE，JMVAE-kl のそれぞれの単数条件付き尤度 $\log p(\tilde{\boldsymbol{w}}|\boldsymbol{x})$

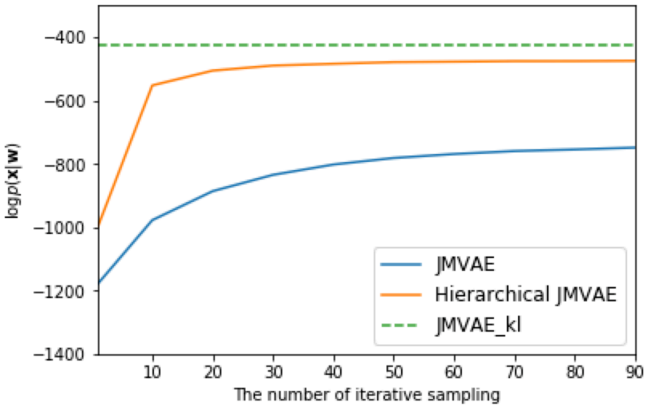


図 5.5　MNIST データ集合における異なる反復サンプリング数での JMVAE，階層的 JMVAE，及び JMVAE-kl の単数条件付き対数尤度の値.

をプロットしたもので，図の横軸は反復サンプリング数を表している（ただし，JMVAE-kl は反復サンプリング手法をとらないので点線で横一直線で表している）．サンプリング数が増えると，JMVAE と階層的 JMVAE の両方の場合で対数尤度は高くなり，反復サンプリングの手法が精度向上に貢献していることがわかる．しかし通常の JMVAE では，JMVAE-kl の尤度よりもはるかに低く，サンプリング回数をかなり増やさないと尤度が高くならない．階層的 JMVAE の場合は，JMVAE よりも少ないサンプリング回数で尤度が大きくなり，最終的な尤度も通常の JMVAE より高くなることがわかる．一方 JMVAE-kl の場合は，反復サンプリングせずに高い尤度を得ることができる．

なお，階層的 JMVAE の条件付き尤度の評価値は，他のモデルの評価値と比較すると過小評価されている可能性があることに留意されたい．これは，条件付き尤度の近似式が，他のモデルよりも下界をおさえているためである（付録 A.1 を参照）．よって階層的 JMVAE は，サンプリング回数を増やした場合に，実際は JMVAE-kl よりも高い尤度となっている可能性もある．

実験 1-2：潜在空間での共有表現の確認

続いて，潜在空間で異なるモダリティの共有表現が獲得されていることを確認する．

図 5.6 は，各手法での潜在表現を可視化したものである．図 5.6 の左がすべてのモダリティを入力としたエンコーダからサンプリングしたもので，右が単一のモダリティ $\boldsymbol{w}$ からサンプ

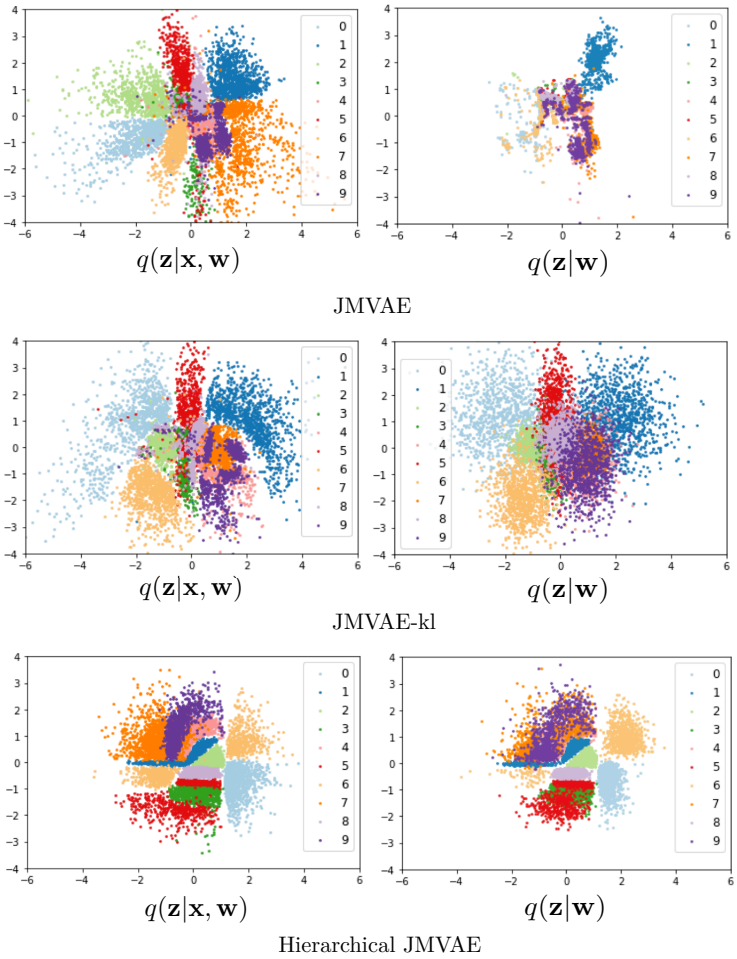


図 5.6 2-D の潜在表現の可視化．異なる色の点は数字ラベルに対応している．JMVAE と階層的 JMVAE の反復サンプリング数は 100 に設定した．

リングした潜在表現である．なお，階層的 JMVAE の場合は一番上の確率的層でサンプリングしている．ここでは，エンコーダの $f_{\boldsymbol{\mu}}$ と $f_{\boldsymbol{\sigma}^2}$ を D2 として訓練したものを用いている．

まず図 5.6 左に着目すると，JMVAE と JMVAE-kl は，いずれもラベルごとに分かれて分布しており，2 つのモダリティを含んだ共有情報が獲得できていることが確認できる．一方，階層的 JMVAE では，よりラベルごとにまとまった表現になっていて，確率的階層化の効果が確認できる．

次に図 5.6 右に着目する．JMVAE の結果をみると，左と比べてかなり小さい領域にサンプルの分布が押しつぶされているのがわかる．また，ラベルごとのまとまりも殆ど見られなく

表 5.1　MNIST におけるテスト条件付き対数尤度の評価.

	$\log p(\tilde{\boldsymbol{x}}\|\boldsymbol{w})$	$\log p(\tilde{\boldsymbol{w}}\|\boldsymbol{x})$	$\log p(\tilde{\boldsymbol{x}}\|\boldsymbol{w},\boldsymbol{x})$	$\log p(\tilde{\boldsymbol{w}}\|\boldsymbol{x},\boldsymbol{w})$
CVAE	-448.8	-5.293	-70.42	-5.304
CMMA	-451.1	-0.2971	-69.89	**-0.002574**
JMVAE	-747.1	**-0.2286**	**-69.57**	-0.2026
JMVAE-kl	**-422.35**	-0.2628	-76.94	-0.1874
Hierarchical JMVAE	-475.5	-2.640	-196.9	-0.4456

なっている．この結果から，画像情報が欠損されると潜在表現が崩れてしまうことが実際に確認できる．一方 JMVAE-kl の結果をみると，左のすべてのモダリティからのサンプリングとほぼ変わらない潜在表現が得られていることがわかる．右の図では楕円にまとまった表現となっているが，これは $\boldsymbol{w}$ からの情報量が $\boldsymbol{x}$ と $\boldsymbol{w}$ の両方の情報量と比べて小さいため，不確かさが大きくなり，比較的単純な分布となっているためである．JMVAE の場合のように，小さい領域にまとまっていないことから，明らかに欠損モダリティの問題が解決されていることがわかる．最後に階層的 JMVAE の結果をみると，欠損していない左の結果とほぼ同様の，ラベルごとに分離した分布が獲得されている．この結果から，階層的 JMVAE でも潜在表現が崩れてしまう問題点は解消されたといえる．

実験 1-3：条件付き対数尤度による評価

本節では，モダリティの双方向の変換と再構成を定量的に評価するため，単数条件付き対数尤度と複数条件付き対数尤度で評価する．また，対数尤度の評価を既存の条件付き VAE である CVAE や CMMA と比較する．ここで，CVAE と CMMA は条件付き分布をモデル化していることに注意されたい．つまり，これら従来のモデルは 1 方向でしかモダリティを変換できないため，双方向変換のためには，各変換方向についてモデルを用意して独立に学習する必要がある．そのため，学習時間のコストが増えるだけでなく，実験 1-2 で示したようなすべてのモダリティを統合した共有表現を獲得することができない．一方 JMVAE は，潜在変数が与えられた下での各モダリティの条件付き分布によって同時分布をモデル化しているので，潜在変数で共有表現を獲得でき，その表現を介して双方向にモダリティを変換することができる．

表 5.1 は両方向のモダリティにおける単数条件付き対数尤度と複数条件付き対数尤度の評価

である．この評価では，JMVAE と階層的 JMVAE の反復サンプリング数を 100 とした．既存手法と提案手法，特に通常の JMVAE を比較すると，複数条件付き対数尤度については，既存手法と同等もしくはそれ以上の結果となっている．既存モデルが各方向の変換に別々のモデルを用意して学習する必要があることを考えると，それらと同等の精度で双方向に変換できるという結果は十分であると考えられる．また単数条件付き尤度についても，$\boldsymbol{x}$ から $\boldsymbol{w}$ の生成は，他の既存モデルよりも適切に生成できていることがわかる．一方 $\boldsymbol{w}$ から $\boldsymbol{x}$ の尤度は低くなっているが，これは実験 1-1 でも示したとおり，欠損モダリティ問題のためである．JMVAE-kl や階層的 JMVAE によって，この問題は解決され，特に JMVAE-kl では従来のモデルよりも高い尤度となることがわかる．

次に表 5.1 における提案手法内での比較をする．まず，$\boldsymbol{w}$ の単数条件付き尤度で評価した元の JMVAE と JMVAE-kl の差は，$\boldsymbol{x}$ の場合の差と比べると，あまり大きくなっていない．このことから，欠損モダリティ問題が発生するのは，生成したいモダリティ，即ち欠損モダリティがそれ以外のモダリティと比較して高次元の場合だけであることがわかる．次に，複数条件付き尤度の結果から，JMVAE-kl はそれぞれのモダリティの再構成の精度向上には必ずしも貢献しないことがわかる．これは JMVAE-kl が，VI 最小化と等価であることから，同一のモダリティの再構成ではなく，異なる種類のモダリティ間をより適切に生成するようにモデル化されているためと考えれられる．なお，階層的 JMVAE の結果をみると，いずれも JMVAE-kl よりも低い結果となっているが，これは実験 1-1 でも述べたとおり，階層的 JMVAE の尤度の評価値が過小評価されているためと考えられる．つまり，階層的 JMVAE については他のモデルとの数値による厳密な比較は難しいということに留意されたい．

5.3.6 実験 2：CelebA

実験 2-1：欠損モダリティ問題と提案手法による改善の確認

図 5.7 は，属性から顔画像を生成した結果を示している．まず JMVAE の場合（図 5.7(b)）では，属性に対応する顔画像が適切に生成されないことがわかる．さらに，MNIST の場合と異なり，反復サンプリング数を増やすと，生成した顔画像が崩れてしまうことが確認できる．このことから，CelebA の顔画像のような高次元のモダリティの場合では，反復サンプリング手法が全くうまく働かないことがわかる．次に，階層的 JMVAE の場合（図 5.7(c)）をみる

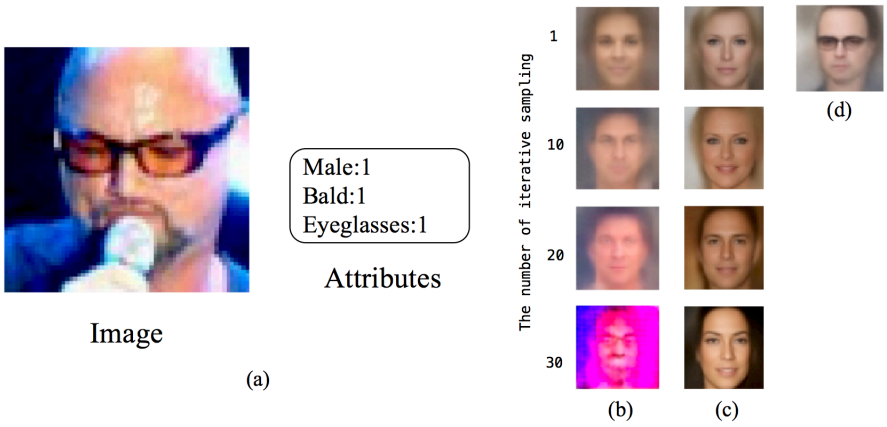


図 5.7　CelebA データ集合における属性（$\boldsymbol{w}$）から顔画像（$\boldsymbol{x}$）の生成．(a) は CelebA 画像のテスト集合の中の一事例．(b) から (d) は，それぞれのモデルで (a) の事例の属性から生成した顔画像であり，下の行にいくにつれ反復サンプリング回数が増えている．(b) JMVAE．(c) 階層的 JMVAE．(d) JMVAE-kl.

と，反復サンプリング数を増やすことで，より綺麗な顔画像が生成されることがわかる．しかし，属性に対応した顔画像にはなっていないことも確認でき，特に Eyeglasses という特徴的な属性が生成された顔画像には全く反映されていない．CelebA でこのような結果になる理由の 1 つとして，反復サンプリングの際，固定される属性よりも，サンプル毎にランダムに変化する画像の方が次元や情報量が大きいため，潜在空間で大きく移動してしまい，その結果属性とは異なる画像が生成されてしまう，ということが考えられる．また確率的階層が深くなると，潜在変数におけるモード（今回の場合は属性の分布）の隔たりが小さくなることが知られており [Bengio 13b]，このことも，潜在変数において生成したい属性とは異なる属性の分布に移動しやすくなる原因かもしれない．一方 JMVAE-kl の場合（図 5.7(d)）は，反復サンプリングをせずに，属性に対応した顔画像を生成することができる．

図 5.8 は，CelebA において，反復サンプリング数を変更した場合の提案モデルにおける単数条件付き尤度をプロットしたものである．この結果を見ると，図 5.5 と異なり，反復サンプリング数が 1 のときに JMVAE や階層的 JMVAE は JMVAE-kl よりも高い尤度となり，サンプリング数が増えるごとに尤度が下がることがわかる．これは，CelebA の実験設定では，単数条件付き尤度 $\log p(\tilde{\boldsymbol{x}}|\boldsymbol{w})$ が，元の顔画像と対応する属性から生成した顔画像との平均二乗誤差と実質的に等価（生成分布 $\log p(\boldsymbol{x}|\boldsymbol{w})$ が分散 1 で固定されたガウス分布であるため）で

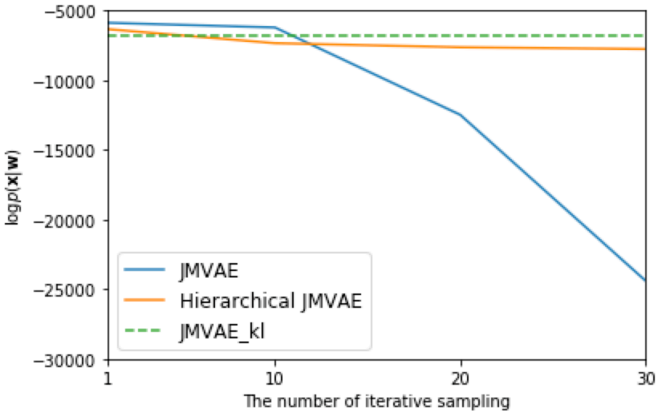


図 5.8 CelebA データ集合における異なる反復サンプリング数での JMVAE，階層的 JMVAE，及び JMVAE-kl の単数条件付き対数尤度の値.

表 5.2 CelebA におけるテスト条件付き対数尤度の評価.

	$\log p(\tilde{\boldsymbol{x}}\|\boldsymbol{w})$	$\log p(\tilde{\boldsymbol{w}}\|\boldsymbol{x})$	$\log p(\tilde{\boldsymbol{x}}\|\boldsymbol{w},\boldsymbol{x})$	$\log p(\tilde{\boldsymbol{w}}\|\boldsymbol{x},\boldsymbol{w})$
CVAE	**-6825**	-44.28	-4031	-44.28
CMMA	-6920	-44.57	**-4026**	**-38.74**
JMVAE	-48763	**-43.97**	**-4026**	-43.43
JMVAE-kl	-6852	-44.13	-4089	-43.83
Hierarchical JMVAE	-7355	-47.61	-4901	-47.32

あることと関連している．平均二乗誤差で測った場合，ぼやけた平均的な顔画像と比較した方が全サンプルで比較したときの平均誤差は小さくなるため，サンプリング数が 1 のときに尤度が高くなる．一方，サンプリング数を増やすと，はっきりした顔画像が生成されるようになるので，元画像との誤差が大きくなるサンプルが複数あらわれ，結果的にサンプリング数が 1 の場合よりも尤度が低くなると考えられる．図 5.8 をみると，階層的 JMVAE の場合は，サンプリング数が増えるにつれて JMVAE-kl よりも少し低い尤度で収束することが確認できる．しかし JMVAE の場合は，大きく尤度が下がってしまっている．これは図 5.7 の結果と対応しており，高次元モダリティの場合，反復サンプリング手法では欠損モダリティ問題が解決できないことを示している．

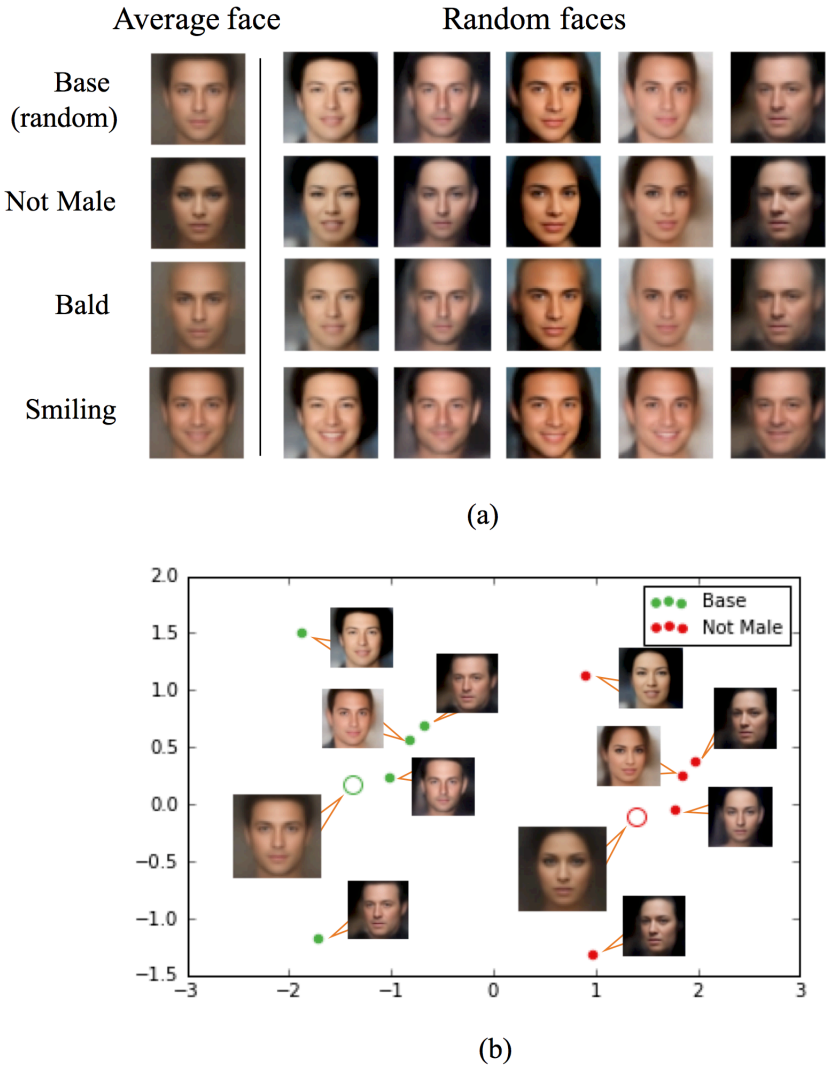


図 5.9　(a) 平均顔とランダムな顔画像の生成．各行は凡例の属性に対応しており，ランダムな顔画像の各列は同じバリエーションをもつ．(b) 潜在表現の PCA による可視化．それぞれの色は各サンプルが条件づけられている属性に対応している．

実験 2-2：条件付き対数尤度による評価

表 5.2 は条件付き尤度の評価を示している．JMVAE と階層的 JVMAE の反復サンプリング数は 40 とした．この表から，MNIST の場合と同様，既存の条件付きモデルと比べて，JMVAE による条件付き尤度の評価値が同等もしくはわずかに高い結果となった．また，実験

2-1 で示したように，JMVAE-kl と階層的 JMVAE の単数条件付き尤度の値が JMVAE よりも大幅に高くなっていることから，欠損モダリティ問題が解消されることがわかる．しかし，これらの値は CVAE の尤度より僅かに低くなっていることも確認できる．これは CelebA は異なるモダリティ間での情報量の違いが大きいため，双方向の変換が MNIST より困難なためと考えられる．

提案手法内での違いについては，MNIST の場合（表 5.1）とほぼ同様の結果となった．

実験 2-3：属性から顔画像の生成と共有表現の確認

次に，JMVAE が CelebA データ集合の属性から画像を生成できることを確認する．以降の実験では，実験 2-1 の結果から JMVAE-kl を用いることとし，また鮮明な画像を生成するために，GAN と組み合わせることとする．具体的には，$p(\boldsymbol{x}|\boldsymbol{z})$ のネットワークを GAN における**生成器**とみなし，JMVAE-kl の下界とともに GAN の誤差関数を最適化する．これは，VAE-GAN モデル [Larsen 15b] と同じ方法である．なお，学習に用いる GAN の識別器のネットワークは `C64R-C128BR-C256BR-C256BR-F-D1024R-D1S` とした（ただし `DkS` は `Dk` の活性化関数をシグモイド関数としたもの）．

図 5.9(a) では，様々な属性で条件づけて生成した顔画像を示している．ここでは，まずすべての属性を $\{-1, 1\}$ からランダムに選択したものを Base とし，Base の設定したい属性（ここでは男性，禿げている，笑っている，に該当する属性）の値を 2（Not の場合は -2）とすることで，各属性における $\boldsymbol{w}$ を設定した．各属性の平均顔は，$p(\boldsymbol{x}|\boldsymbol{z}_{mean})$（ただし $\boldsymbol{z}_{mean}$ は q の平均）からサンプリングした．さらに，$\boldsymbol{x} \sim p(\boldsymbol{x}|\boldsymbol{z})$（ただし，$\boldsymbol{z} = \boldsymbol{z}_{mean} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}$ 及び $\epsilon \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\zeta})$ であり，この図では $\boldsymbol{\zeta} = 0.6 \cdot \boldsymbol{I}$ とした）のように，同じ属性から様々なバリエーションの顔を生成することができる．結果から，各属性に応じて適切に顔が生成できていることがわかる．

図 5.9(b) では，生成した各顔画像の潜在空間における位置をプロットしている．この図をみると，顔画像のサンプルが，対応する属性ごとにまとまって配置されていることがわかる．また，属性ごとのまとまりの中では，平均顔がほぼ中心に位置し，その周りにランダムな複数の顔画像が配置されており，さらにこれらの配置は，Base と Not Male でほぼ同じとなることが確認できる．これらの結果から，顔画像と属性を含んだ共有表現の多様体学習が適切に行われていることがわかる．

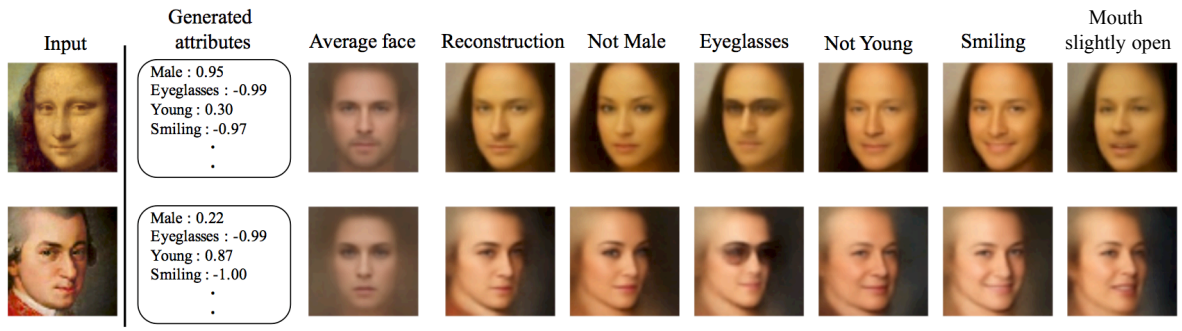


図 5.10 モナリザ（上）とモーツァルト（下）の肖像画[*6]と，それらの属性の生成，及び変更した属性で条件づけて再構成した画像.

実験 2-4：顔画像と属性の双方向の変換

最後に，JMVAE が顔画像と属性間を双方向に変換できることを示す．図 5.10 では，訓練集合に含まれない画像から属性を生成し，さらに属性値を変更することで様々な顔画像を生成できることを示している．これらの画像は次のようにして作成している．まず，JMVAE でラベルのない画像 $\boldsymbol{x}$ から対応する属性 $\boldsymbol{w}$ を生成する．次に，生成した属性 $\boldsymbol{w}$ から平均顔 $\boldsymbol{x}_{mean}$ を生成する．そして，変更したい属性の値を変更した $\boldsymbol{w}$ から $\boldsymbol{x}'_{mean}$ を生成する．最後に，$\boldsymbol{x} + \boldsymbol{x}'_{mean} - \boldsymbol{x}_{mean}$ とすることで，変更した属性値に対応した顔画像 $\boldsymbol{x}'$ を得ることができる．なお図 5.10 の Reconstruction は，入力画像のみから再構成によって生成した画像である．

図 5.10 から，モダリティの次元が大きく異なるデータ集合においても，提案手法が両方のモダリティを双方向に変換できることがわかる．なお，上記の属性から画像への生成方法は CMMA [Pandey 16] と類似しているが，CMMA は 1 方向でしか変換できないため，属性情報のない画像を変化させることはできない.

なお，本研究の提案手法である JMVAE-kl が，属性から顔画像の生成を正しく行えるかについては文献 [Vedantam 17] でも検証されている．この論文では，JMVAE-kl の他に，BiVCCA [Wang 16] や，彼らの提案手法である TELBO と比較している[*7]．論文の中では，

[*6] モナリザ（https://en.wikipedia.org/wiki/Mona_Lisa），モーツァルト（https://en.wikipedia.org/wiki/Wolfgang_Amadeus_Mozart）

[*7] この論文内では，本研究の JMVAE が JVAE（Joint VAE），JMVAE-kl が JMVAE と呼ばれている.

JMVAE-kl は BiVCCA よりも大幅に精度が高く変換できること，そして TELBO と比較して同等以上の精度となることが確認されている．

5.4 結論

本章では，異なる次元や構造をもつモダリティ間を双方向に変換する問題に取り組んだ．まず，複数のモダリティが共有表現に写像されるように VAE を拡張した．このモデルを JMVAE と呼ぶ．本モデルでは，すべてのモダリティは共有表現で条件づけられており，全モダリティの同時分布をモデル化している．このため，既存の条件付き分布をモデル化した手法とは異なり，双方向に異なるモダリティを変換することができる．また，異なるモダリティ間で変換する際，情報量の大きいモダリティを欠損させるとうまく補完できない問題があることを確認した．これを解決するために，新たに階層的 JMVAE と JMVAE-kl という追加的手法を提案した．実験によって，これらの手法で欠損モダリティ問題が解決されることを確認した．さらに，全モダリティを統合した共有表現が適切に獲得され，既存の 1 方向しか変換できないモデルと比較して，同等もしくはそれ以上に適切にモダリティ間を変換できることを確認した．最後に，片方のモダリティを変化させることで，もう片方の該当する部分が変化することが確認できた．

第 6 章

マルチモーダルデータを用いた半教師あり学習のための深層生成モデル

本章では，マルチモーダル学習の融合設定において，深層生成モデルを用いた半教師あり学習モデルを提案する．

マルチモーダル学習の融合問題とは，マルチモーダルデータから対応する目標ラベルを予測するというものである．この問題設定が注目されているのは，異なるモダリティには相補性の性質があり，あるモダリティからは，ラベルを予測する上で他のモダリティにはない付加価値（多様性）が得られるからである [Lahat 15]．

たとえば，ロボットは画像情報だけでなく距離情報や感覚センサ情報など，様々なモダリティ情報を同時に取得している．ロボットが物体認識をする問題設定を考えると，距離情報やセンサ情報も利用することで，画像情報だけではわからない物体の形や質感を知ることができる．したがってそれらの情報をすべて物体カテゴリを予測する識別モデルの訓練集合として利用することで，精度の高いカテゴリラベルの予測が期待できる．

近年のマルチモーダル学習は，識別器として深層ニューラルネットワークが使われることが多い [Ngiam 11]．これは，従来の識別器や特徴抽出器と比較して識別精度が高く，ネットワークの隠れ層で入力の良い特徴表現が獲得できるからである．また，識別器がネットワークで構成されているため，設計や結合が容易である．したがって，各モダリティを入力として

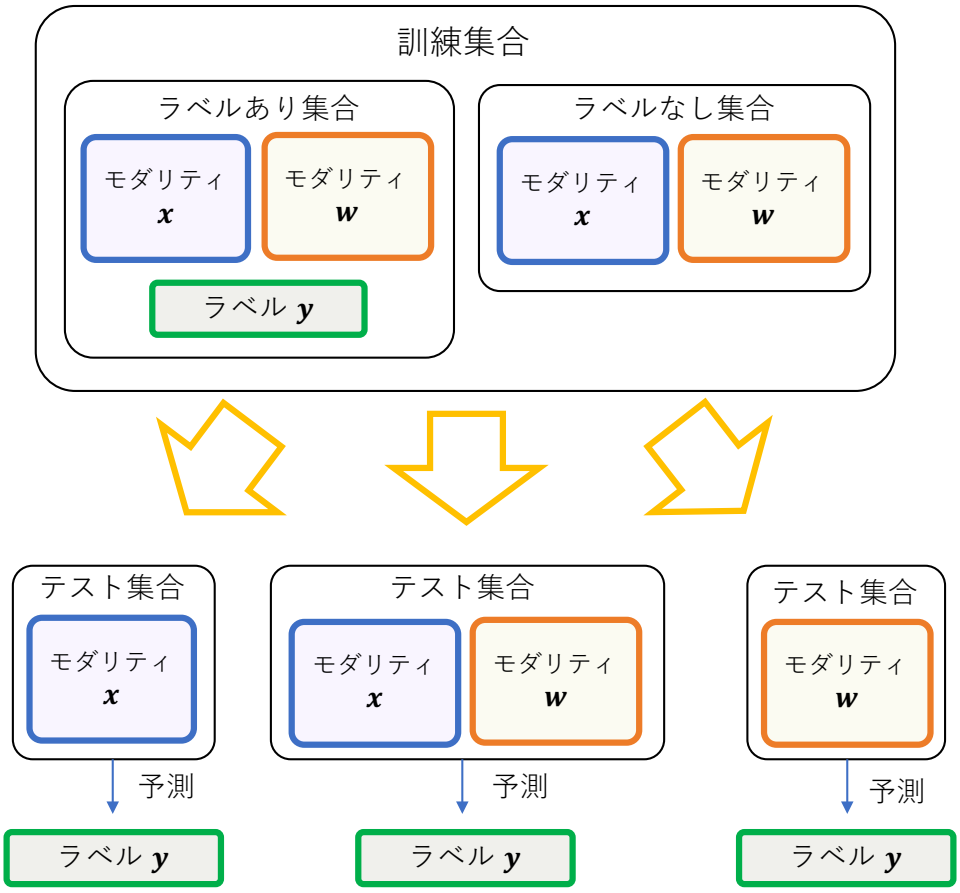


図 6.1　本研究での半教師ありマルチモーダル学習の問題設定の概要.

とるネットワークを用意し，それらのネットワークの最も上位の隠れ層を共有したものを訓練することで，複数のモダリティの情報を統合した表現（共有表現）を獲得することができる [Ngiam 11].

各モダリティのネットワークは，特徴空間に応じて設定できるため，このアプローチはモダリティの種類によらずに成功を収めてきた．たとえば，画像とそれに関する質問文が与えられた場合に解答を予測するという visual question answering では，画像のモダリティに対して畳み込みニューラルネットワーク，質問文に対して LSTM を用意して，それらを結合することで解答を予測するモデルを設計している [Antol 15].

一般に，深層ニューラルネットワークの学習には大量のラベルありデータ集合が必要である．ニューラルネットワークの入力に対応する各モダリティの情報については，比較的安価に

手にいれることができる．たとえば，前述したロボットの物体認識の場合，ロボットに搭載され ている様々なセンサから，ある物体に関する複数のモダリティ情報を入手することができる．その一方で，出力に対応するラベル情報は人手で与える必要があるため，データが大量にある場合，ラベル付けのための人的コストがかかってしまうという課題がある．

これを解決する方法の 1 つが半教師あり学習の適用である．半教師あり学習は，人手でラベル付けしたデータだけでなく，大量のラベルなしデータも訓練に用いることで識別モデルの汎化性能を高める枠組みである．前述したように，マルチモーダルデータの組み合わせは安価に入手できることが多いので，これらをラベルなしデータ集合として利用することで，識別モデルの精度を向上させることが考えられる．

複数のモダリティを入力とした半教師あり学習，すなわち，半教師ありマルチモーダル学習は，これまでもいくつか提案されている．Cheng らは，深層ニューラルネットワークを用いた共訓練（co-training）によって，各モダリティ及びすべて（ここでは 2 種類）のモダリティをそれぞれテスト集合としたときの識別精度が向上する枠組みを提案している [Cheng 16]．この問題設定を示したものが図 6.1 である．本研究では，これを半教師ありマルチモーダル学習の問題設定とする．

一方で近年，半教師あり学習のモデルとして深層生成モデルが注目されている [Kingma 14a]．2.4.3 節で述べたように，深層生成モデルを用いることで，ラベルあり集合とラベルなし集合を統一的に扱うことができる．したがって，上記の共訓練のアプローチ等とは異なり，end-to-end に半教師あり学習を実行することができる．深層生成モデルの中でも VAE に基づく方法は，既存の半教師あり学習と比べて高い精度となることが知られている [Kingma 14a, Maaløe 16]．しかし，これまで深層生成モデルを用いた，マルチモーダルデータのための半教師あり学習は殆ど提案されていない．

こうした背景から，本研究では深層生成モデルを用いた半教師ありマルチモーダル学習のモデルとして，次の 2 つを新たに提案する．1 つ目が，既存の VAE による半教師モデルを単純にマルチモーダルに拡張した semi-supervised MVAE（SS-MVAE）で，もう 1 つが，SS-MVAE に複数のモダリティを統合する共有表現（潜在変数）を追加した semi-supervised HMVAE（SS-HMVAE）である．

これらのモデルは，テスト集合として複数のモダリティが与えられた場合を想定している

が，図 6.1 で示したように，1 つのモダリティのみが与えられた場合も精度の良い識別結果となることが求められる．しかし上記の提案モデルの場合，その他のモダリティの入力がラベル予測の際に欠損することになり，識別精度が落ちてしまう恐れがある．そこで本研究では，マルチモーダル深層生成モデルの欠損補完手法 [Suzuki 16] に基づき，SS-HMVAE を拡張する形で，SS-HMVAE-kl という追加的アプローチを提案する．

本研究の主な貢献は以下の通りである．

- 深層生成モデルによる半教師ありマルチモーダル学習について新たに取り組み，SS-MVAE と SS-HMVAE を提案する．
- テスト集合として 1 つのモダリティしか与えられない問題設定に対処するため，SS-HMVAE をベースに，SS-HMVAE-kl を提案する．
- MNIST を用いた実験で，教師あり学習の設定において，SS-HMVAE-kl によって，モダリティを欠損させても識別精度が大きく落ちないことを確認する．そして半教師あり学習の設定で，マルチモーダルな入力でも半教師あり学習が行えること，そしてテスト集合に単一モダリティしかない設定でも精度よく予測でき，従来の単一モダリティのみで訓練やテストをする半教師あり学習モデルの精度を上回ることを確認する．
- マルチモーダルデータ集合である RGB-D Object データ集合を用いた実験で，SS-HMVAE によって，従来の単一モーダル及びマルチモーダルの半教師あり学習モデルよりも精度の高い結果が得られることを確認する．特に SS-HMVAE-kl によって，識別精度の低いモダリティの精度が大幅に向上することを確認する．

6.1 関連研究

6.1.1 半教師ありマルチモーダル学習の既存研究

Guillaumin らは，マルチモーダルデータ（画像とタグ）のラベルあり集合とラベルなし集合から，画像を入力してラベルを予測する識別器を学習する枠組みを提案している [Guillaumin 10]．この手法は 2 段階からなっており，まずマルチカーネル学習によって，ラベルありデータを用いてラベルを予測する識別器を学習し，その識別器を使ってラベルなしデータのラベル情報を予測する．次に，ラベルありデータ及びラベルなしデータから予測

したラベル情報を用いて，画像からラベルを予測するもう 1 つの識別器を学習する，という流れである．この手法は，実験によって，次に述べる共訓練よりも高い精度となることが示されている．この手法をもとに，いくつかの拡張が提案されている [Luo 13, Xie 14]．しかしこの手法は，タグからラベルを予測する識別器を訓練時に同時に学習できない．

Cheng らは，深層ニューラルネットワークを用いた共訓練による半教師ありマルチモーダル学習を提案している [Cheng 16]．共訓練とは，入力データを 2 つに分けて，それぞれに識別器を用意し，片方の識別器で高い確信度となった事例をもう片方の訓練事例とすることで訓練事例を増やす半教師あり学習の手法である．この手法は 2 つに分けたデータがラベルの下で条件付き独立である必要があるが，マルチモーダルデータの場合，前述した相補性の性質から，この要件を満たしている可能性が高い．Cheng らは，通常の共訓練に加えて，訓練データとして追加する事例が各カテゴリごとに多様性を保つように凸クラスタリング法を用いている．さらに各モダリティの識別器を訓練したあと，両方のモダリティを入力とした識別器を学習している．これによって，Guillaumin らの枠組みとは異なり，各モダリティを入力とした識別器と，両方のモダリティを入力とした識別器が得られる．ただし，この手法は上記のように精度向上のために複数の手法を併用しており，RGB-D データ集合に特化したものである．

6.1.2　深層生成モデルによる半教師あり学習

深層生成モデルは，半教師あり学習にも優れた性能を示すことで知られている．VAE では，M2 モデル [Kingma 14a]（2.4.3 節）が最初に提案され，それを拡張する形で ADGM と SGDM [Maaløe 16] が提案されている．なお，VAE とは別の深層生成モデルである GAN も半教師あり学習のモデルとして使われる [Salimans 16, Salimans 16]．ただし，本研究では VAE に基づく手法を提案するので，これらの GAN による手法との比較は行わない．

マルチモーダルデータで半教師あり学習を行う深層生成モデルは，これまでのところ殆ど提案されていない．唯一，Du らによってマルチモーダル感情認識のための半教師あり深層生成モデルが提案されている [Du 17]．ここでは，semiMVAE というモデルが提案されているが，これは本研究で提案する SS-MVAE（6.3.1 節）と同じ生成モデルの形をしている[*1]．一方本

[*1] ただし Du らの論文が出たのは 2017/4/25 であり，筆者らが初めて SS-MVAE を発表した時期とほぼ同じである [鈴木 17]．また semiMVAE は，推論モデルに混合ガウス分布を仮定しており，単純にガウス分布を仮定している SS-MVAE とは厳密には異なる．

章では，SS-MVAE を拡張した SS-HMVAE を提案し，さらに一方のモダリティを入力とした識別モデルを同時に学習する方法を提案している．

6.2 問題設定

データ集合 $\mathcal{D}_{\mathcal{L}} = \{(\boldsymbol{x}_1, \boldsymbol{w}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_N, \boldsymbol{w}_N, \boldsymbol{y}_N)\}$ が訓練集合として与えられるとする．ただし，$\boldsymbol{x}$ と $\boldsymbol{w}$ は異なるモダリティであり，$\boldsymbol{y} \in \{0, 1\}^K$ はそれらの目標カテゴリを表すラベル情報とする[*2]．また訓練集合の各事例 $(\boldsymbol{x}_n, \boldsymbol{w}_n, \boldsymbol{y}_n)$ は同じ対象を表現しているものとする．

本研究で取り組む問題設定は，ラベルあり訓練集合の他にラベルなし訓練集合 $\mathcal{D}_{\mathcal{U}} = \{(\boldsymbol{x}_1, \boldsymbol{w}_1), ..., (\boldsymbol{x}_M, \boldsymbol{w}_M)\}$（ただし $N << M$）が与えられた下で，Cheng らの枠組みのように，$\boldsymbol{x}$ と $\boldsymbol{w}$ を入力とする識別モデル $p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w})$ だけでなく，各モダリティを入力とする識別モデル，すなわち，$p(\boldsymbol{y}|\boldsymbol{x})$ 及び $p(\boldsymbol{y}|\boldsymbol{w})$ を獲得することである（図 6.1）．本研究ではこのタスクを**半教師ありマルチモーダル学習（semi-supervised multimodal learning）**と呼ぶ．

6.3 提案手法

本節では，本研究で提案する SS-MVAE と SS-HMVAE について説明する．そして，モダリティが欠損した場合に対処する手法について説明する．

6.3.1 SS-MVAE

モダリティ $\boldsymbol{x}, \boldsymbol{w}$ の他にラベル $\boldsymbol{y}$ を考えて，生成過程を $\boldsymbol{y} \sim p(\boldsymbol{y}) = \mathrm{Cat}(\boldsymbol{y}; \boldsymbol{\pi})$，$\boldsymbol{z} \sim p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$，$\boldsymbol{x}, \boldsymbol{w} \sim p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{w}|\boldsymbol{z}, \boldsymbol{y}) = p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{y}) p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z}, \boldsymbol{y})$ とする．このとき，全モダリティとラベルの同時分布は $p(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) = \int p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{y}) p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z}, \boldsymbol{y}) p(\boldsymbol{z}) p(\boldsymbol{y}) d\boldsymbol{z}$ で与えられる．

この深層生成モデルは，JMVAE（5.2.1 節）にラベル情報が加わったものとみることができる．もう 1 つの見方として，モダリティが 1 つだとすると，Kingma らによって提案された VAE の半教師あり学習モデルである M2 モデル [Kingma 14a]（2.4.3 節）と同じ生成モデルになる．したがって，この深層生成モデルは，M2 モデルをマルチモーダルデータに拡張した

[*2] 本研究ではマルチラベル（1 つの事例が複数の対象に該当する）を想定しないので，one-hot（1 つの要素のみが 1 で残りは 0）表現となる．

形とみなすことができる．本研究ではこのモデルを *Semi-Supervised Multimodal Variational AutoEncoder*（*SS-MVAE*）と呼ぶ．

SS-MVAE の変分下界 $\mathcal{L}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y})$ は次のようになる．

$$\begin{aligned}\log p(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) &= \log \int p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z}, \boldsymbol{y}) d\boldsymbol{z} \\ &\geq E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{y}) p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z}, \boldsymbol{y}) p(\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y})}] \equiv \mathcal{L}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) \end{aligned} \tag{6.1}$$

ただし $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y})$ は推論モデルである．

確率分布を深層ニューラルネットワークで表す方法や，式 (6.1) の勾配をとるために再パラメータ化トリックを用いることについては，2.4.1 節での説明のとおりである．

この目的関数はラベルあり集合において学習するが，半教師あり学習の枠組みではラベルなし集合も学習に利用する．そのために，ラベル情報を含まない同時分布を考えて目的関数を設計する．この目的関数は，識別モデル $q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w})$ を導入して次のように求められる．

$$\begin{aligned}\log p(\boldsymbol{x}, \boldsymbol{w}) &= \log \int \int p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z}, \boldsymbol{y}) d\boldsymbol{z} d\boldsymbol{y} \\ &\geq E_{q_{\boldsymbol{\phi}}(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{y}) p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{z}, \boldsymbol{y}) p_{\boldsymbol{\theta}}(\boldsymbol{z}) p_{\boldsymbol{\theta}}(\boldsymbol{y})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w})}] \equiv \mathcal{U}(\boldsymbol{x}, \boldsymbol{w}). \end{aligned} \tag{6.2}$$

ただし，$q_{\boldsymbol{\phi}}(\boldsymbol{z}, \boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}) = q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w})$ である．

なお識別モデルは，問題設定からカテゴリ分布となるが，通常離散分布の場合は再パラメータ化トリックを使うことはできない．M2 モデルや SDGM ではカテゴリ分布の積分を各ラベルの総和の形に直すことで対処されている．しかし，これらの論文では実験に 10 クラスのデータ集合を使用しているのに対して，本研究の実験では 51 クラスある RGB-D データ集合を用いるため，計算コスト的に問題がある．そこで，本研究では Gumbel-softmax [Jang 16] を用いて，近似的に再パラメータ化する．

さらに，ラベルあり集合において識別モデルを学習するために，以下のようにラベルあり集合における識別損失を式 (6.2) に加える．

$$\mathcal{L}_l(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) = \mathcal{L}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) + \alpha \cdot \log q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}). \tag{6.3}$$

ただし，α は学習において識別モデルと生成モデルの割合を調節するパラメータである．

したがって，ラベルあり・なし集合の両方における目的関数 $\mathcal{J}_{MVAE}$ は，

$$\mathcal{J}_{MVAE} = \frac{1}{N} \sum_{(\boldsymbol{x}_n, \boldsymbol{w}_n, \boldsymbol{y}_n) \in \mathcal{D}_{\mathcal{L}}} \mathcal{L}_l(\boldsymbol{x}_n, \boldsymbol{w}_n, \boldsymbol{y}_n) + \frac{1}{M} \sum_{(\boldsymbol{x}_j, \boldsymbol{w}_j) \in \mathcal{D}_{\mathcal{U}}} \mathcal{U}(\boldsymbol{x}_j, \boldsymbol{w}_j) \tag{6.4}$$

となる．

この目的関数をラベルあり・なし集合を用いて最大化することで，生成モデル，推論モデル，そして識別モデル $q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w})$ を同時に学習することができる．また式 (6.4) の計算には，ラベルあり集合だけでなく，ラベルなし集合でも利用できるので，end-to-end な半教師あり学習が実現できる．

図 6.2(a) は SS-MVAE のグラフィカルモデルである．

6.3.2 SS-HMVAE

SS-MVAE では，推論モデルは $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y})$ と設定していたが，変分推論において，推論モデルは複雑な形になっていることが望ましい．たとえば，観測変数 $\boldsymbol{x}$ と潜在変数 $\boldsymbol{z}$ をもつ生成モデル $p(\boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$ を考えたとき，変分下界は $\mathcal{L}(\boldsymbol{x}) = \log(\boldsymbol{x}) - D_{KL}(q(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z}|\boldsymbol{x}))$ となる（ただし $q(\boldsymbol{z}|\boldsymbol{x})$ は $p(\boldsymbol{z}|\boldsymbol{x})$ の近似分布となっている推論モデルである）．このとき，推論モデル $q(\boldsymbol{z}|\boldsymbol{x})$ が複雑になればなるほど，下界は真の対数尤度の良い近似となる．また，本研究では 2 つのモダリティを入力として扱うので，その点からも，潜在変数を推論する分布はガウス分布のような単純な分布ではなく，より複雑な形となることが望ましい．

そこで，新たな潜在変数として $\boldsymbol{a}$ を導入し，推論モデルを $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) = \int q_{\boldsymbol{\phi}}(\boldsymbol{z}, \boldsymbol{a}|\boldsymbol{x}, \boldsymbol{w})d\boldsymbol{a} = \int q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{a})q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x}, \boldsymbol{w})d\boldsymbol{a}$ とする．また潜在変数 $\boldsymbol{a}$ を使って，生成モデルも $p(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) = \int\int p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{a})p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{a})p_{\boldsymbol{\theta}}(\boldsymbol{a}|\boldsymbol{z}, \boldsymbol{y})p(\boldsymbol{z})p(\boldsymbol{y})d\boldsymbol{a}d\boldsymbol{z}$ とする．すると，ラベルあり集合における目的関数は，

$$\begin{aligned} &\log p(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) \\ &\geq E_{q_{\boldsymbol{\phi}}(\boldsymbol{a}, \boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{a})p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{a})p_{\boldsymbol{\theta}}(\boldsymbol{a}|\boldsymbol{z}, \boldsymbol{y})p(\boldsymbol{z})p(\boldsymbol{y})}{q_{\boldsymbol{\phi}}(\boldsymbol{a}, \boldsymbol{z}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y})}] \equiv \mathcal{L}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{y}) \end{aligned} \tag{6.5}$$

となる．

また，識別モデルについても $\boldsymbol{a}$ を使って $q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}) = \int q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{a})q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x}, \boldsymbol{w})d\boldsymbol{a}$ とする．すると識別モデルの対数尤度（負の損失関数）は $\log q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}) = \log E_{q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x}, \boldsymbol{w})}[q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{a})]$ とな

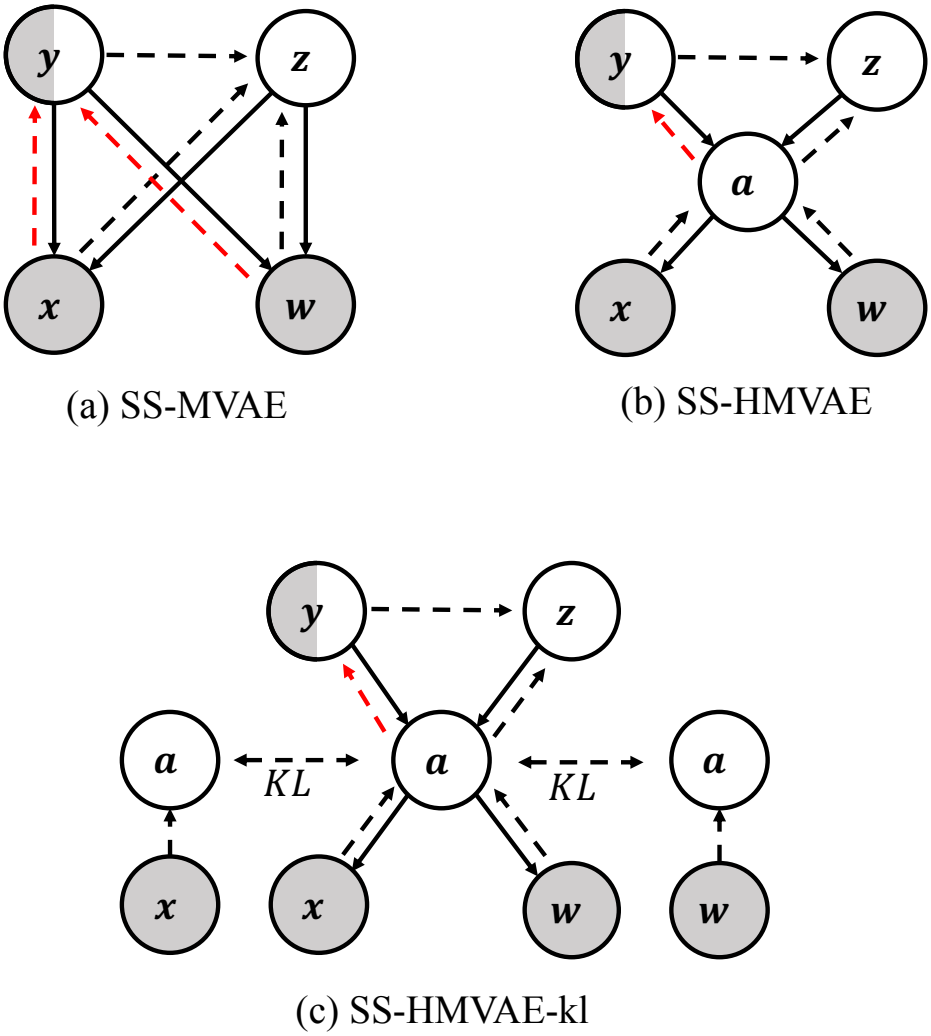


図 6.2　本研究で提案する半教師ありマルチモーダルモデルのグラフィカルモデル.

るため，より柔軟な確率分布で表現することができる．この識別モデルを用いることで，ラベルなし集合における目的関数は次のようになる.

$$\begin{aligned}&\log p(\boldsymbol{x},\boldsymbol{w})\\&\geq E_{q_{\boldsymbol{\phi}}(\boldsymbol{a},\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{a})p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{a})p_{\boldsymbol{\theta}}(\boldsymbol{a}|\boldsymbol{z},\boldsymbol{y})p(\boldsymbol{z})p(\boldsymbol{y})}{q_{\boldsymbol{\phi}}(\boldsymbol{a},\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})}] \equiv \mathcal{U}(\boldsymbol{x},\boldsymbol{w}).\end{aligned} \tag{6.6}$$

ただし $q_{\boldsymbol{\phi}}(\boldsymbol{a},\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) = q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{a},\boldsymbol{y})q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{a})q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})$ である.

なお，式 (6.6) の下界は

$$
\begin{aligned}
&E_{q_{\boldsymbol{\phi}}(\boldsymbol{a},\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{a})p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{a})p_{\boldsymbol{\theta}}(\boldsymbol{a}|\boldsymbol{z},\boldsymbol{y})p(\boldsymbol{z})p(\boldsymbol{y})}{q_{\boldsymbol{\phi}}(\boldsymbol{a},\boldsymbol{z},\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})}] \\
&= E_{q_{\boldsymbol{\phi}}(\boldsymbol{z},\boldsymbol{y}|\boldsymbol{a})}[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{a}|\boldsymbol{z},\boldsymbol{y})p(\boldsymbol{z})p(\boldsymbol{y})}{q_{\boldsymbol{\phi}}(\boldsymbol{z},\boldsymbol{y}|\boldsymbol{a})}] \\
&\quad + E_{q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{a})] + E_{q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{w}|\boldsymbol{a})] + \mathcal{H}(q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w}))
\end{aligned} \tag{6.7}
$$

と分解することができる．ここで，第 1 項は観測変数を $\boldsymbol{a}$ とした M2 モデルと同じであり，第 2 項と第 3 項は JMVAE（式 (5.1)）の再構成誤差の項に等しい．また第 4 項は平均情報量である．このことから，SS-HMVAE の目的関数は JMVAE の学習と M2 モデルの両方の下界を含んでいることがわかる．したがって $\boldsymbol{a}$ では，JMVAE と同様に複数のモダリティを統合した共有表現が獲得され，それを入力とした半教師あり学習が行なわれると解釈できる．

このように，潜在変数が階層的な構造になっていることから，本モデルを *Semi-Supervised Hierarchical Multimodal Variational AutoEncoder*（*SS-HMVAE*）と呼ぶ．図 6.2(b) がグラフィカルモデルである．

$\boldsymbol{a}$ のような潜在変数を追加してより柔軟な分布を表現する研究は，これまでもいくつか行なわれている [Maaløe 16, Sønderby 16, Gulrajani 16]．Maale らは，深層生成モデルに補助変数と呼ばれる潜在変数を導入した半教師あり学習モデルである ADGM と SDGM を提案している [Maaløe 16]．

6.3.3 欠損モダリティへの対処

SS-MVAE や SS-HMVAE を訓練する際，マルチモーダルデータが入力として識別モデルに与えられることが前提となっている．一方で問題設定や図 6.1 で示したように，テスト時に単一のモダリティしか与えられない場合でも適切にカテゴリラベルを識別したい．

テスト集合に単一のモダリティデータしかない場合，ラベルを推定する最も単純な方法は，そのモダリティ以外の識別モデルの入力を欠損させることである．しかし，識別モデルは深層ニューラルネットワークで設計されており，ニューラルネットワークでは欠損モダリティに対して適切に対処できないという問題が指摘されている [Baltrušaitis 17]．こうした問題を解決方法としては，5.2.2 節で説明した反復サンプリング手法 [Rezende 14] がある．しかし，5 章で示したとおり，欠損モダリティの次元や情報量が大きい場合，充分に補完できない可能性

がある．そのため本研究では，SS-HMVAE を拡張したモデルである SS-HMVAE-kl を提案する．

以下では，反復サンプリング手法と SS-HMVAE-kl について，それぞれ説明する．

反復サンプリング

反復サンプリング手法 [Rezende 14] は，5.2.2 節で説明したように，VAE で入力の欠損値を補完する方法である．SS-MVAE は，識別モデルに潜在変数を持たないが，SS-HMVAE は $q(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) = \int q(\boldsymbol{y}|\boldsymbol{a})q(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})d\boldsymbol{a}$ のように潜在変数 $\boldsymbol{a}$ が含まれており，この変数が異なるモダリティ情報を統合した共有表現の役割を果たしている．よって，識別モデルの入力 $\boldsymbol{x}$ が欠損した場合の遷移カーネルは $\boldsymbol{a}$ を用いて次のように書ける．

$$T(\tilde{\boldsymbol{x}}|\boldsymbol{x},\boldsymbol{w}) = \int p(\tilde{\boldsymbol{x}}|\boldsymbol{a})q(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})d\boldsymbol{a}. \tag{6.8}$$

したがって，反復サンプリングの手段は次のとおりである．まず $\boldsymbol{x}$ の初期値を $\boldsymbol{x} \sim p(\boldsymbol{x})$ のようにランダムなノイズとして，推論モデル $q(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})$ を使って $\boldsymbol{a}$ をサンプリングする．次に，生成モデル $p(\boldsymbol{x}|\boldsymbol{a})$ から $\boldsymbol{x}$ をサンプリングし，新たな入力値として置き換える．そして，このサンプリングを複数回反復することで，最終的に欠損モダリティ $\boldsymbol{x}$ を補完できる．本研究では，このようにして補完したモダリティを識別モデルの入力として与えて，目標ラベルを予測する．

SS-HMVAE-kl

次に，5.2.4 節で欠損モダリティ問題への対処のため提案した JMVAE-kl を参考に，SS-HMVAE を拡張した SS-HMVAE-kl を提案する．

SS-HMVAE-kl のアイディアは，JMVAE-kl と同じアプローチである．

まず，モダリティごとの推論モデルである $q_{\boldsymbol{\lambda}}(\boldsymbol{a}|\boldsymbol{x})$ と $q_{\boldsymbol{\lambda}}(\boldsymbol{a}|\boldsymbol{w})$ を新たに用意する．そして，SS-HMVAE の推論モデル $q_{\boldsymbol{\lambda}}(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})$ との距離（ここではカルバック・ライブラー（KL）ダイバージェンスを考える）を近づけるように学習する．したがって，SS-HMVAE-kl の目的関数 $\mathcal{J}_{kl}$ は，

$$\mathcal{J}_{kl} = \mathcal{J}_{HMVAE} - \frac{\beta}{M+N}\mathcal{J}_{div} \tag{6.9}$$

となる．ただし，$\mathcal{J}_{HMVAE}$ は SS-HMVAE の目的関数，$\boldsymbol{\lambda}$ は各モダリティの推論モデルのモデルパラメータ，β は KL 項の影響を調節するパラメータ，そして

$$\mathcal{J}_{div} = \sum_{(\boldsymbol{x}_n, \boldsymbol{w}_n) \in \mathcal{D}_{\mathcal{L}} \bigcup \mathcal{D}_{\mathcal{U}}} [D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x}_n, \boldsymbol{w}_n)||q_{\boldsymbol{\lambda}}(\boldsymbol{a}|\boldsymbol{x}_n)) + D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x}_n, \boldsymbol{w}_n)||q_{\boldsymbol{\lambda}}(\boldsymbol{a}|\boldsymbol{w}_n))] \tag{6.10}$$

である．

この式を最適化することで，SS-HMVAE のモデル及び各モダリティの推論モデルを同時に学習できる．適切に各モダリティの推論モデルが学習できれば，単一モダリティ $\boldsymbol{x}$ を入力とする識別モデルを $q_{\boldsymbol{\phi},\boldsymbol{\lambda}}(\boldsymbol{y}|\boldsymbol{x}) = E_{q_{\boldsymbol{\lambda}}(\boldsymbol{a}|\boldsymbol{x})}[q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{a})]$ のように求めることができる．

なお，このように各モダリティの推論モデルを用意して KL ダイバージェンスを近づけるアプローチは，5.2.4 節で説明したように，変分推論による variation of information の最小化と数式的に等価である（証明は付録 A.2 を参照）．Variation of information を最小化するということは，モダリティ間の双方向変換がより適切になるように促されることを意味する．JMVAE や SS-HMVAE では双方向変換がうまくいくということは，欠損補完が適切に行われることなので，直感的な JMVAE-kl や SS-HMVAE-kl の効果と対応している．

Cheng らの半教師ありマルチモーダル学習では，各モダリティのネットワークを共訓練したあとに，2 つのモダリティネットワークの特徴量を入力とした識別器を学習していた [Cheng 16]．一方 SS-HMVAE-kl では，マルチモーダル入力と各モダリティ入力の識別モデルを同時に学習することができる．

図 6.2(c) が SS-HMVAE-kl のグラフィカルモデルである．KL と書かれている部分は，KL ダイバージェンスを近づけているという意味である．

6.4 実験

6.4.1 データ集合

本研究では，MNIST と Washington RGB-D の 2 つのデータ集合を用いて実験した．

MNIST

5 章の実験では，MNIST の手書き数字画像と数字ラベルを異なるモダリティとして実験に用いた．本研究ではマルチモーダルデータとして扱うために，手書き数字画像を左右半分にし

てそれぞれ異なるモダリティとし，ラベルは数字ラベルとした．このとき，半分にした画像それぞれは 1D ベクトル表現に平坦化され，かつ移動不変な表現と考えるので，ベクトルを結合しただけでは元の 2D 数字画像を復元できるわけではない．この設定は文献 [Sohn 14] でもマルチモーダルデータの簡単な問題設定として用いられている．なお，実際のマルチモーダルデータは特徴空間や情報量が大きく異なることが多いので，このデータ集合は提案手法を検証するための簡単な問題であることに留意されたい．

分割した各画像をそれぞれ $\boldsymbol{x} \in \{0,1\}^{392}$，$\boldsymbol{w} \in \{0,1\}^{392}$ とし，ラベル情報は $\boldsymbol{y} \in \{0,1\}^{10}$ とする．前処理として，数字画像の各ピクセル値が $[0,1]$ になるよう正規化した．全データ集合のうち 50,000 を訓練集合とし，残りの 10,000 をテスト集合とした．半教師あり学習の設定では，訓練集合のうちの一部（割合は実験設定による）をラベルあり集合とし，残りをラベルなし集合とした．

RGB-D データ集合

RGB-D とは，カラー画像（RGB）とそれに対応する奥行き（Depth）画像で構成される情報である．本実験で用いる Washington RGB-D データ集合 [Lai 11] は Microsoft の Kinect によって撮られたものである．300 種類の家庭用品の画像で構成され，それらは 51 のカテゴリにグループ分けされている．先行研究 [Eitel 15] と同様，それぞれのグループを画像のクラスとして扱う．RGB と Depth は特徴空間も情報量も異なることから，RGB-D はマルチモーダル情報として扱われる [Eitel 15]．本研究では，MNIST よりも実践的なマルチモーダルデータ集合として実験で検証する．図 6.3 が，Washington RGB-D の例である．

51 クラスのそれぞれには 600 枚の事例があり，各クラスについて 5 枚飛ばしに事例を選択することで 41,877 枚のデータ集合を作った．41,877 枚のうち，文献 [Lai 11] が指定した 10 通りの分割方法に従い，訓練集合とテスト集合に分けた．平均しておよそ 35,000 枚が訓練集合，6,877 枚がテスト集合となっている．さらに訓練集合から 5% をランダムに選択してラベルある集合とし，残りをラベルなし集合とした．

RGB-D データ集合の前処理について説明する．まず，RGB 画像と Depth 画像の両方を 148×148 にリサイズする．元画像は縦長や横長なので，元画像の長い方を 148 で固定し，短い方は端のピクセルを拡張する形で補間する．次に，Depth 画像の距離情報が欠損している部分を最近傍の距離の値で補間し，距離の値を 0 から 225 に正規化する．さらに，jet colormap

図 6.3 Washington RGB-D の例．左が RGB 画像で右が対応する Depth 画像．ここでは Depth 画像の色が奥行きの値に対応していて，赤いほど手前の距離であることを表している．

処理によって，単一チャネルの Depth 画像を 3 チャネルに拡張する．以上の前処理の方法は，すべて文献 [Eitel 15] に従ったものである．ただし画像サイズは文献 [Cheng 16] に従い，文献 [Eitel 15] とは異なりデータ拡張は行っていない．

また本実験では，RGB-D については深層生成モデルの入力として直接扱うのではなく，深層ニューラルネットワークで特徴抽出したものを使うとする[*3]．これは，本研究では画像生成が目的でないため，画像を直接入力として入れる必要がないからである．特徴抽出用の深層ニューラルネットワークは，ILSVRC2012 データ集合で事前学習済みの VGG16 [Simonyan 14] とし，特徴量は fc1 層（4096 次元）での出力値を特徴量とする．まず VGG16 をモダリティごとに用意し，訓練集合で再学習する．ただし半教師あり学習の設定なので，訓練集合のうちラベルあり集合のみを訓練に用いる．訓練には Adam [Kingma 14b] を用い，過学習を防ぐため学習率は 10^{-5} として 200 エポック学習した．前述のとおり，訓練・テストの分割方法は 10 通りあるので，それぞれのラベルあり集合で訓練して特徴抽出する．

したがって，RGB 画像と Depth 画像の入力特徴量は，それぞれ $\boldsymbol{x} \in \mathcal{R}_{>0}^{4096}$ と $\boldsymbol{w} \in \mathcal{R}_{>0}^{4096}$

[*3] このため 2.3 節で述べた分類だと，「深層学習の特徴抽出 + 生成モデルによるアプローチ」と考えることもできる．しかし，このアプローチでは，ガウス分布やベルヌーイ分布といった簡単な分布で構成される生成モデルのことを指している．また本研究は深層生成モデルの研究であるので，図 4.1 では「深層生成モデルによるアプローチ」に分類している．

となる[*4].

6.4.2　モデル構造

MNIST

SS-MVAE の分布とそのネットワーク構造は以下のようにした．なお，モデル構造の表記方法は 5.3.2 節と同じである．

- $p(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{y})$, $p(\boldsymbol{w}|\boldsymbol{z},\boldsymbol{y})$ (ベルヌーイ分布)
  - f_μ: `D392`
  - f_{MLP}: `(z, y-D512R)-D512R-D512R-D512R`
- $q(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w})$ (カテゴリ分布)
  - f_μ: `D10`
  - f_{MLP}: `(x-D512R-D512R, w-D512R-D512R)-D512R-D512R`
- $q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w},\boldsymbol{y})$ (ガウス分布)
  - f_μ and f_{σ^2}: `D64`
  - f_{MLP}: `(y-D512R, x-D512R-D512R, w-D512R-D512R)-D512R`

また，SS-HMVAE については次のように設定した．

- $p(\boldsymbol{x}|\boldsymbol{a})$, $p(\boldsymbol{w}|\boldsymbol{a})$ (ベルヌーイ分布)
  - f_μ: `D392`
  - f_{MLP}: `a-D512R-D512R-D512R`
- $q(\boldsymbol{y}|\boldsymbol{a})$ (カテゴリ分布)
  - f_μ: `D10`
  - f_{MLP}: `a-D512R-D512R-D512R`
- $q(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})$ (ガウス分布)
  - f_μ and f_{σ^2}: `D64`
  - f_{MLP}: `(x-D512R-D512R, w-D512R-D512R)-D512R`

[*4] 定義域が正の実数となるのは，fc1 層の活性化関数に ReLU を使っているためである

- $q(\boldsymbol{a}|\boldsymbol{z},\boldsymbol{y})$ (ガウス分布)
  - f_μ and f_{σ^2}: `D64`
  - f_{MLP}: `(a-D512R-D512R, y-D512R)-D512R`
- $q(\boldsymbol{z}|\boldsymbol{a},\boldsymbol{y})$ (ガウス分布)
  - f_μ and f_{σ^2}: `D64`
  - f_{MLP}: `(z-D512R-D512R, y-D512R)-D512R`

さらに，SS-HMVAE-kl の各モダリティの推論モデルは次のように設定した.

- $q(\boldsymbol{a}|\boldsymbol{x})$, $q(\boldsymbol{a}|\boldsymbol{w})$ (ガウス分布)
  - f_μ and f_{σ^2}: `D64`
  - f_{MLP}: `x or w-D512R-D512R-D512R`

RGB-D データ集合

SS-HMVAE の分布そのネットワーク構造は以下のようにした.

- $p(\boldsymbol{x}|\boldsymbol{a})$, $p(\boldsymbol{w}|\boldsymbol{a})$
  - f_μ: `D1024`
  - f_{MLP}: `a-D1024R-D1024R`
- $q(\boldsymbol{y}|\boldsymbol{a})$ (カテゴリ分布)
  - f_μ: `D51`
  - f_{MLP}: `a-D1024R-Dropout0.5`
- $q(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})$ (ガウス分布)
  - f_μ and f_{σ^2}: `D1024`
  - f_{MLP}: `(x-D1024R, w-D1024R)`
- $q(\boldsymbol{a}|\boldsymbol{z},\boldsymbol{y})$ (ガウス分布)
  - f_μ and f_{σ^2}: `D1024`
  - f_{MLP}: `(z-D1024R, y-D1024R)`
- $q(\boldsymbol{z}|\boldsymbol{a},\boldsymbol{y})$ (ガウス分布)
  - f_μ and f_{σ^2}: `D1024`

– f_{MLP}: (a-D1024R, y-D1024R)

ただし，DropoutRate はドロップアウト率 Rate のドロップアウト層である．

さらに，SS-HMVAE-kl の各モダリティの推論モデルは次のように設定した．

- $q(\boldsymbol{a}|\boldsymbol{x})$, $q(\boldsymbol{a}|\boldsymbol{w})$ (ガウス分布)
  - f_{μ} and f_{σ^2}: D1024
  - f_{MLP}: x or w-D1024R

6.4.3　学習パラメータ設定と評価指標

NIST データ集合での評価はクラス分類誤り率（エラー率）とし，値が低いほど精度が高いことを意味する．RGB-D データ集合はクラス分類正解率で評価する．

最適化アルゴリズムは Adam を利用した．バッチサイズは 128，学習率は 10^{-4} として，MNIST データ集合は 1000 エポック，RGB-D データ集合は 100 エポック学習した．

SS-HMVAE の反復サンプリング数は 100 回とした．また $\alpha = 1$（識別モデルのパラメータ）及び $\beta = 1$（KL ダイバージェンスのパラメータ）とした．

実装は Theano [Theano Development Team 16] と Lasagne [Dieleman 15] をベースとした深層生成モデルライブラリ Tars（8 章）を使用した．

6.4.4　実験 1：MNIST

実験 1 では，MNIST データ集合を利用して，次のことを確認する．

- 教師あり設定で，提案手法のマルチモーダル学習の識別精度を評価する．また，モダリティを欠損させると精度がどの程度悪化するのか，そして反復サンプリングと SS-HMVAE-kl によってどの程度改善されるのかを確認する（実験 1-1）．
- テスト集合に両方のモダリティが含まれる場合の半教師ありマルチモーダル学習の性能を確認する（実験 1-2）．
- テスト集合に片方のモダリティのみを含む場合の半教師ありマルチモーダル学習の性能を確認する（実験 1-3）．

表 6.1 MNIST での教師ありマルチモーダル学習のテスト集合における分類誤り率（%）．
† の結果は，元論文からの引用．

モデル	$\boldsymbol{x}$	$\boldsymbol{w}$	$\boldsymbol{x}+\boldsymbol{w}$
MRBM（最尤学習）† [Sohn 14]	15.0	11.1	1.6
MRBM（VI 最小化による学習）† [Sohn 14]	6.6	7.3	1.7
SS-MVAE	36.0	40.0	1.2
SS-HMVAE	34.6	37.1	**1.1**
SS-HMVAE（反復サンプリング）	11.3	12.9	**1.1**
SS-HMVAE-kl	**4.4**	**6.0**	2.6

実験 1-1

提案モデルによる教師ありマルチモーダル学習では，各モデルのラベルあり集合の目的関数 $\mathcal{D}_{\mathcal{L}}$ を学習する．SS-MVAE は識別モデルと生成モデルの分布が完全に分かれているので，識別モデルの項のみを学習すればよい．一方，SS-HMVAE の識別モデルは $\log q_{\boldsymbol{\phi}}(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{w}) = \log E_{q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})}[q(\boldsymbol{y}|\boldsymbol{a})]$ となり，推論モデル $q_{\boldsymbol{\phi}}(\boldsymbol{a}|\boldsymbol{x},\boldsymbol{w})$ を含んでいるため，$\mathcal{D}_{\mathcal{L}}$ 全体を学習する必要がある．また SS-HMVAE-kl では，$\mathcal{D}_{\mathcal{L}}$ だけでなく，式 (6.10) も目的関数に加える（ただしラベルあり集合だけで学習する）．

比較のために，既存の深層生成モデルによるマルチモーダル学習モデルである MRBM による，同様の設定のもとでの実験結果 [Sohn 14] も合わせて転載する．この結果は，通常の最尤学習と，VI 最小化に基づく訓練によるもので，訓練したあとに MRBM の共有表現を入力として線形 SVM でクラス分類器を訓練している．VI は双方向の変換が最大になるような目的関数になっているため，最尤学習よりも高い精度で各モダリティからのラベル予測が実行できる．

表 6.1 が実験結果である．SS-MVAE，SS-HMVAE 及び SS-HMVAE-kl が，提案モデルのマルチモーダル教師あり学習の結果である．SS-HMVAE では，反復サンプリングをしなかった場合とした場合の両方の結果を載せている．

まず SS-MVAE の結果をみると，マルチモーダル学習の結果（$\boldsymbol{x}+\boldsymbol{w}$）は，既存の MRBM よりも高い精度となっている．しかし，各モダリティにおける分類精度（$\boldsymbol{x}$，$\boldsymbol{w}$）はマルチモー

ダル学習での結果と比べると大きく下回っており，MRBM の最尤学習の結果よりも低くなっている．これは，SS-MVAE の識別モデルでは，欠損モダリティを適切に扱えないことを示している．

次に SS-HMVAE の結果を確認する．SS-HMVAE は SS-MVAE と比べてマルチモーダル学習の精度が向上している．これは，SS-HMVAE の識別モデルが推論分布における期待値になっていて，より柔軟な確率分布を表現できるためと考えられる．しかし，やはり一方のモダリティを欠損させると，MRBM の結果と比べて精度が落ちてしまっている．

表 6.1 の下から 2 つ目は，SS-HMVAE において反復サンプリング手法を適用した結果である．表からわかるように，サンプリングしない場合と比べて両方のモダリティで大幅に精度が向上していることがわかる．また MRBM と比較すると，最尤学習の場合と同じくらいの結果になっている．これは，SS-HMVAE が反復サンプリングによって，従来の生成モデルと同じように欠損モダリティを扱えるようになったことを意味している．

そして，その下が SS-HMVAE-kl の結果である．マルチモーダル学習の結果は SS-MVAE や SS-HMVAE に劣っているが，各モダリティにおける精度は，反復サンプリングの手法を上回り，提案手法の中で最も良い結果となっている．また，MRBM の結果と比較すると，最尤学習の手法だけでなく，VI 最小化の手法よりも良い結果となっている．6.3.3 節で述べたように，KL ダイバージェンス最小化の項を入れた下界を最大化することは，VI 最小化と等価である．今回の実験結果で SS-HMVAE-kl が VI 最小化手法と同等以上の精度となったことからも，この等価性が裏付けられたといえる．

以上の結果から，教師ありマルチモーダル学習で，モダリティを欠損させると精度が大きく落ちるが，欠損サンプリングで既存の生成モデルと同等程度の精度となること，そして SS-HMVAE-kl によって VI 最小化による MRBM をも上回る結果となることを確認した．

実験 1-2

訓練集合内のラベルあり事例数を 100，500，1000 に減らし，そのラベルあり集合での教師あり学習と，本研究で提案する半教師ありマルチモーダル学習の精度を比較する．なお，訓練集合のうちラベルあり集合以外は，すべて半教師あり学習に利用した．

表 6.2 は提案モデルで半教師ありマルチモーダル学習を実行した結果である．一番上は，ラベルあり集合で教師あり学習した（ラベルなし集合は利用していない）結果である．ネット

表 6.2 提案手法による MNIST での半教師ありマルチモーダル学習．ラベルあり事例数を変えた時のテスト集合における分類誤り率（%）で評価．

モデル	100	500	1000
教師あり学習	27.7	11.4	8.5
SS-MVAE	25.2	**2.9**	**2.6**
SS-HMVAE	**9.6**	2.9	2.7
SS-HMVAE-kl	10.1	3.5	2.8

表 6.3 MNIST での単一モダリティ $\boldsymbol{x}$（上）と $\boldsymbol{w}$（下）の半教師あり学習．ラベルあり事例数を変えた時のテスト集合における分類誤り率（%）で評価．

モデル	100	500	1000
M2 [Kingma 14a]	23.1	14.6	10.5
SDGM [Maaløe 16]	15.8	8.2	7.5
SS-HMVAE（反復サンプリング）	22.4	13.7	13.3
SS-HMVAE-kl	**12.3**	**6.2**	**5.5**

モデル	100	500	1000
M2 [Kingma 14a]	28.2	17.7	11.8
SDGM [Maaløe 16]	27.2	10.3	8.9
SS-HMVAE（反復サンプリング）	20.4	15.1	13.8
SS-HMVAE-kl	**12.2**	**6.9**	**6.5**

ワーク構造は SS-MVAE の識別モデルと同じとした．

すべてのラベルあり事例数の場合で教師あり学習の精度を上回っていることから，提案モデルによってマルチモーダルデータを入力とした半教師あり学習ができることがわかる．モデル間の違いをみると，ラベルあり事例数が大きいときは，いずれもほぼ同じ精度となっている．その一方で，ラベルあり事例数が 100 のときは，SS-MVAE があまり教師あり学習の結果と変わらないのに対して，SS-HMVAE は教師あり学習や SS-MVAE よりもはるかに精度が高くなっていることがわかる．SS-HMVAE-kl は，KL ダイバージェンスの項も目的関数に含めて学習しているため，いずれの結果でも SS-HMVAE より少し精度が低くなる傾向があることが

わかる．それでも SS-HMVAE と同様，ラベルあり事例数が 100 のときに SS-MVAE よりも精度が高くなることが確認できる．

実験 1-3

本実験では，単一モダリティにおける既存の半教師あり学習モデル（M2 モデル [Kingma 14a]，SDGM [Maaløe 16]）と比較する．ネットワーク構造はなるべく SS-MVAE や SS-HMVAE と合わせた．

表 6.3 は，反復サンプリング法を適用した SS-HMVAE と SS-HMVAE-kl による，単一モダリティによるラベル予測の誤り率である．M2 モデルと SDGM は，いずれも単一モダリティのみを入力にとるため，各モダリティで別々に訓練している．訓練集合とテスト集合の事例数は提案手法と M2 や SDGM で同じだが，提案手法では訓練時にマルチモーダルデータ（$\boldsymbol{x}$ と $\boldsymbol{w}$ の両方）が与えられている．テスト時には，いずれの手法も単一モダリティのみを入力とする．

反復サンプリングを適用した SS-HMVAE は，M2 モデルと同等かそれより少し高い精度となったが，SDGM よりは低い精度に留まった．SS-HMVAE が訓練時に 2 つのモダリティを用いていることを考えると，単一モダリティについては，あまり良い精度とはなっていない．一方で SS-HMVAE-kl は，すべてのラベルあり事例数の場合と両方のモダリティの場合で，既存の単一モダリティの半教師あり学習モデルを上回る結果となった．M2 や SDGM が，特にラベルあり事例数が 100 のときに大きく精度が落ちるのに対して，SS-HMVAE-kl は比較的高い精度を保っている．ラベル数が 100 というのは，各クラスあたりのラベルありデータが 10 しかないことに該当し，数字画像の半分しか入力として与えられない状況では困難な設定といえる．SS-HMVAE-kl では，訓練で両方のモダリティデータが与えられたときに，片方のモダリティだけでも予測に役立つような知識を適切に抽出できていたため，精度が高く予測できたと考えられる．これまでの半教師ありマルチモーダル学習では，単一モダリティの半教師あり学習よりも精度が高いことは確認されていなかったが [Guillaumin 10]，提案手法ではそれが確かめられたことになる．

なお，表 6.2 と表 6.3 の両方の提案手法の識別モデルは，1 回の学習で得たものである．テスト集合で単一のモダリティが与えられる問題設定を考案した Guillaumin らの提案手法では，単体の学習では一方のモダリティの予測モデルしか獲得できなかった [Guillaumin 10]．

表 6.4 RGB-D データ集合での単一モダリティ及びマルチモーダルにおける半教師あり学習．テスト集合における正解率（%）で評価．† の結果は，元論文からの引用．

	モデル	$\boldsymbol{x}$	$\boldsymbol{w}$	$\boldsymbol{x}+\boldsymbol{w}$
単一モダリティ	M2 [Kingma 14a]	85.6 ± 1.6	72.0 ± 1.7	-
	SDGM [Maaløe 16]	85.6 ± 1.9	75.8 ± 1.7	-
マルチモーダル	CT+SVM† [Cheng 15]	78.7	75.4	83.7
	共訓練 † [Cheng 16]	85.5 ± 2.0	**82.6** ± 2.3	89.2 ± 1.3
提案手法	SS-MVAE	79.6 ± 1.8	34.4 ± 7.0	89.9 ± 1.7
	SS-HMVAE（反復サンプリング）	86.4 ± 2.1	54.8 ± 1.9	**90.6** ± 1.6
	SS-HMVAE-kl	**86.8** ± 2.2	81.1 ± 2.4	90.2 ± 1.4

そのため，訓練前にどのモダリティを入力にした識別モデルを設計するかを決める必要があった．本研究で提案するモデルは，図 6.1 で示した問題設定の識別モデルを同時に学習することが可能である．また SS-HMVAE-kl の結果からもわかるように，同時に学習した方が，それぞれのモダリティの知識を活用できるため，SS-HMVAE-kl は計算量と識別精度の両面で利点があると考えられる．

6.4.5 実験 2：RGB-D データ集合

実験 1 は MNIST を用いた 2 つのモダリティの情報量が同程度の設定だったが，実験 2 では情報量が異なり，より複雑な RGB-D データ集合を使って実験する．

この実験では，提案手法を，単一モダリティ及びマルチモーダルにおける既存の半教師あり学習の結果と比較する．単一モダリティについては実験 1-3 と同様，M2 モデル [Kingma 14a]，SDGM [Maaløe 16] と比較する．マルチモーダルについては，CT+SVM [Cheng 15]，共訓練 [Cheng 16] と比較する．共訓練の手法は，マルチモーダル入力の深層ニューラルネットワークを使い，事前学習に ILSVRC2012 データ集合を用いるなど，本研究での設定に近い．ただし，この研究では Depth 画像の前処理に surface normal を用いている．これは本研究で用いた jet colormap 処理よりも精度が高くなることが知られている [Eitel 15]．

表 6.4 が実験結果である．10 分割したそれぞれの訓練・テスト集合での正解率を平均した値と標準偏差を載せている．

両方のモダリティを入力としたときの結果（$\boldsymbol{x}+\boldsymbol{w}$）は，SS-HMVAE が最も良い．SS-

表 6.5　RGB-D データ集合での各設定における教師あり学習．訓練集合全体をラベルあり集合としている．テスト集合における正解率（%）で評価．† の結果は，元論文からの引用．

モデル	$\boldsymbol{x}$	$\boldsymbol{w}$	$\boldsymbol{x}+\boldsymbol{w}$
[Cheng 16] による設定 †	86.3	84.0	91.3
本研究での設定	88.9	81.6	92.0

HMVAE-kl は SS-HMVAE よりも落ちるものの，既存手法よりも良い精度となっている．

次に，各モダリティについての結果をみる．SS-MVAE は，単一モダリティ・マルチモーダルのいずれの既存手法と比較しても低い精度となっている．一方 SS-HMVAE については，RGB 画像（$\boldsymbol{x}$）については既存手法よりもよい精度となっているが，Depth 画像（$\boldsymbol{w}$）については，反復サンプリングによる補完をしているにも関わらず，大幅に精度が落ちている．これは RGB 画像と比べて，Depth 画像の方がラベル情報についての情報量が小さいためと考えられる．5 章でも示したように，情報量の異なるマルチモーダルデータで学習すると，情報量の大きいモダリティが欠損したとき，反復サンプリングでも潜在変数の崩壊が改善されない．今回の結果も，反復サンプリングでは補完しきれなかったことを示している．

一方 SS-HMVAE-kl は，SS-HMVAE と比べると，特に Depth 画像（$\boldsymbol{w}$）について大幅に向上している．これは単一モダリティの既存手法よりも高い結果である．さらに興味深いのは，RGB 画像（$\boldsymbol{x}$）についても精度が向上していることである．

図 6.4 は訓練・テストの各分割における，SS-HMVAE と SS-HMVAE-kl の RGB 画像と Depth 画像の精度比較である．これをみると，どの分割においても SS-HMVAE-kl での Depth 画像の精度が大幅に向上していることがわかる．一方 RGB 画像での SS-HMVAE-kl の結果をみると，SS-HMVAE の結果と比較して精度の減少はほぼ発生しておらず，むしろ僅かに上昇していることがわかる．このことから，情報量が異なるマルチモーダル情報で SS-HMVAE-kl を訓練した場合は，情報量が少ない方に合わせて精度が低くなるのではなく，それらを組み合わせた情報（$\boldsymbol{x}+\boldsymbol{w}$）に向かって両方のモダリティの精度が向上することがわかる．

表 6.4 に戻ると，SS-HMVAE-kl は，マルチモーダルの既存手法と比べても，RBG 画像のときに精度を上回っていることがわかる．その一方で，Depth 画像では共訓練の結果 [Cheng 16]

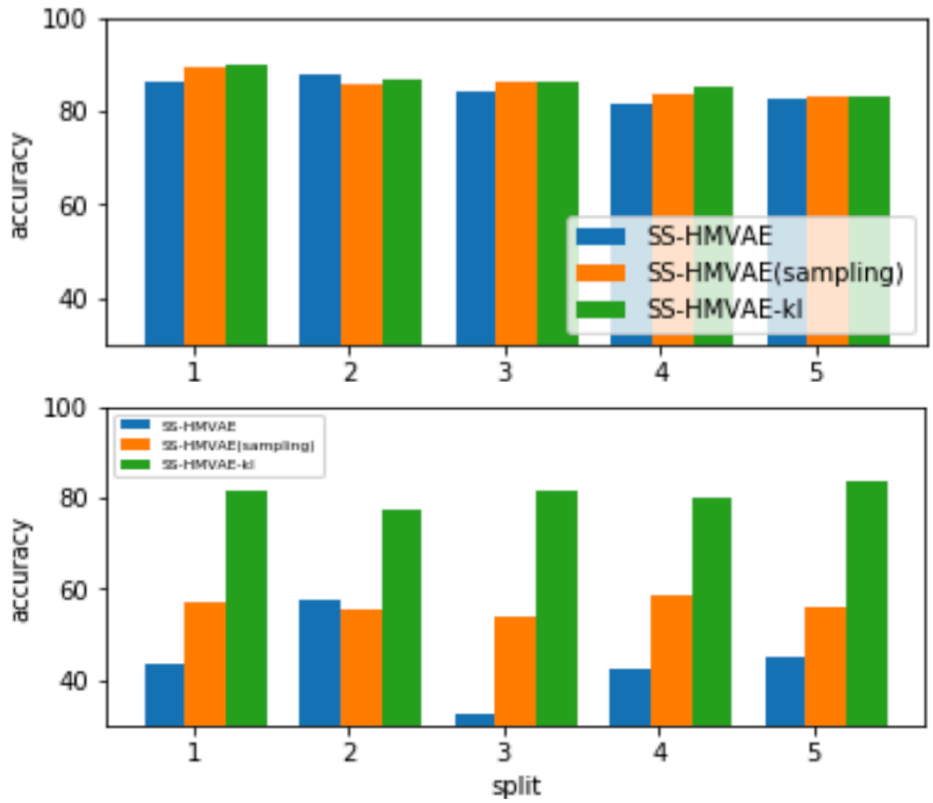


図 6.4 それぞれの訓練・テスト分割での RGB 画像と Depth 画像の正解率．上が RBG 画像の結果で下が Depth 画像の結果．横軸の各番号は，それぞれ異なる訓練・テスト分割に対応している．

よりも僅かに低い結果となっている．これは文献 [Cheng 16] では，より精度の高い surface normal 処理で Depth 画像を前処理しているためと考えられる．

ここで，前処理やネットワーク設定による違いを確認するために，文献 [Cheng 16] での設定と本研究での設定のそれぞれで，全訓練集合をラベルあり集合として教師あり学習した場合の結果を確認する．表 6.5 がその結果である．これは，それぞれの設定での半教師あり学習の精度の上限である[*5]．[Cheng 16] での設定では，前処理の影響によって，本研究の設定と比べて Depth 画像の精度が高くなっている．一方 RGB 画像の方は，本研究が VGG16 を用いて特徴抽出していることから，文献 [Cheng 16] よりも高い精度となっている．マルチモーダル

[*5] 半教師あり学習の上限を確認する意図があるので，この結果では特徴抽出用ネットワークを再学習する際に，訓練集合すべてを用いずに，表 6.4 と同様，5% のラベルあり集合のみを用いる．

入力の場合は，文献 [Cheng 16] も本研究も同じくらいの精度となっている．

表 6.4 の結果と見比べると，本研究での提案手法が，文献 [Cheng 16] の結果と比較しても高い精度で半教師あり学習を実行できていることがわかる．特に SS-HMVAE-kl における Depth 画像については，教師あり学習とほぼ変わらない精度まで向上している．これは，SS-HMVAE-kl で，半教師あり学習の効果と KL 最小化の効果が，敵対せずに共に発揮されていることを示している．

また本研究での設定で，半教師あり設定のラベルあり集合のみ（すなわち，訓練集合の 5%）を用いて教師あり学習をした場合は，$\boldsymbol{x}$，$\boldsymbol{w}$，$\boldsymbol{x}+\boldsymbol{w}$ のそれぞれで 86.1，74.0，88.6 という結果になった．この結果から，SS-HMVAE-kl における Depth 画像だけでなく，SS-HMVAE と SS-HMVAE-kl でのマルチモーダル入力の識別精度も，半教師あり学習によって大きく向上したことがわかる．

なお，文献 [Cheng 16] の手法が，凸クラスタリング法を併用するなど，RGB-D データ集合に特化して様々な改良を施しているのに対して，本研究の提案手法は，汎用的な半教師ありマルチモーダル学習モデルであることにも留意されたい．

6.4.6　考察

ここまでの実験で，提案手法が既存の単一モダリティ及びマルチモーダルにおける半教師あり学習よりもよい精度となることが確認できた．特に SS-HMVAE-kl は，情報量が異なるモダリティの場合，精度の低い方を大幅に向上させることがわかった．

一方，反復サンプリングについては，あまり大幅な欠損補完の改善にはならなかった．反復サンプリングの解決策の 1 つは，階層的 JMVAE（5.2.3 節）から示唆されるように，潜在変数を更に増やすことである．しかし，SS-HMVAE は既に確率的層（観測変数や潜在変数を層と考えた場合の層数）が 3 層になっており，また一般には深層生成モデルの確率的層は，単純に多く積み重ねるだけではうまくいかないことが知られている [Sønderby 16]．さらに多層にするためには，probabilistic ladder network [Sønderby 16] や MatNet [Bachman 16] のような手法を用いる必要がある．

また，今回は識別モデルのパラメータ α と KL 項のパラメータ β をすべて 1 に固定して実験した．パラメータ α は教師なし学習と教師あり学習のバランスを調節し，パラメータ β は，

各モダリティの分布を近づける強さを調節する．したがって，どちらのパラメータもモデルの識別精度に大きく影響すると考えられる．こちらもモダリティの情報量の違いと合わせて，影響の度合いを検証する必要がある．

6.5 結論

本稿では，半教師ありマルチモーダル学習について取り組み，深層生成モデルによる SS-MVAE と SS-HMVAE を提案した．また，入力に 1 つのモダリティしか与えられない問題設定に対処するため，SS-HMVAE を拡張した SS-HMVAE-kl を提案した．

まず，MNIST を用いた実験を行い，SS-HMVAE-kl によって，モダリティ欠損問題が解決すること，そしてテスト集合に 2 つのモダリティが含まれる場合と，単一モダリティの場合の両方で半教師あり学習が適切に行われることを確認した．特に単一モダリティの場合に，従来の単一モダリティにおける半教師あり学習モデルの精度を上回ることが確認された．

そして RGB-D Object データ集合を用いた実験を行い，SS-HMVAE-kl によって，識別精度の低いモダリティの精度が大幅に向上することを確認した．また，単一モダリティ及びマルチモーダルにおける既存の半教師あり学習と比較して，同等以上の結果が得られることを確認した．

第 7 章

属性ごとの観測確率を考慮したゼロショット学習

本章では，共学習の 1 つであるゼロショット学習に取り組む．

ゼロショット学習は画像認識領域における問題設定で，Larochelle らによる研究 [Larochelle 08] 以来様々な手法が提案されている．本章では画像認識に限定して議論を進めるので．本章で言及する目標のクラスは，一般物体認識問題におけるカテゴリに該当する．ゼロショット学習において，テスト集合のクラスラベルを推定するために必要な「共通する情報」は，補助情報（side information）と呼ばれており，何を補助情報とするかが極めて重要となる．Lampert らはゼロショット学習の補助情報にセマンティックな属性（attribute）を用いた効果的な属性ベースのゼロショット学習として，Direct Attribute Prediction（DAP）モデルを提案した [Lampert 09, Lampert 14]．

属性ベースのゼロショット学習において，画像の特徴ベクトル（以降画像特徴量とする）への現れやすさは，属性によって異なると考えられる．たとえば，色情報で表された画像特徴量では，色に関する属性（「blue」など）は現れやすいが，色とは関係無い属性（「hunter」など）は画像特徴量に現れにくい．特に現れにくい属性は，画像特徴量から適切に学習できないため，目的であるテスト集合のクラス推定に負の影響を与える恐れがある．これは，属性と画像特徴量の分布が異なるために生じる問題である．しかし，このような問題について，DAP モデルを始め様々な属性を補助情報としたゼロショット学習 [Yu 10, Rohrbach 10, Rohrbach 11, Fu 14c, Fu 14a] では考慮されていなかった．本研究

ではこの現れやすさの度合いを属性ごとの観測確率と呼ぶこととし，観測確率を考慮した新たな属性ベースゼロショット学習のモデルを提案する．そして実験によって提案モデルの妥当性と有効性を評価する．

本研究の貢献は次のとおりである．

- 観測確率という各属性の画像特徴量への現れやすさの度合いに着目し，その度合いが属性ベースのゼロショット学習において重要であることを示す．
- 観測確率を取り入れた新しい属性ベースゼロショット学習の生成モデルを提案し，DAP モデル [Lampert 09, Lampert 14] と比較して高い正解率となることを示す．
- その他のゼロショット学習 [Lampert 09, Lampert 14, Akata 13, Fu 14c, Fu 14a] とも比較をし，これらの研究のように属性以外の補助情報を追加せずに，同等以上の精度となることを示す．
- 提案手法が DAP モデルの利点を保持しつつ，DAP モデルでの問題点を解決し，扱えなかった問題設定にも適用可能であることを示す．

7.1 提案手法

本節では提案手法の内容について述べる．まず 7.1.1 節で属性ベースのゼロショット学習の問題設定を確認する．7.1.2 節では，本研究で新たに着目した属性ごとの観測確率について，及び観測確率を含めた提案モデルについて説明する．そして 7.1.3 節で，提案モデルによる学習，推定方法について説明する．

7.1.1 問題設定

本節では，属性ベースゼロショット学習の問題設定を定式化し，解くべき問題を確認する．

画像の実現値である画像特徴量を $\boldsymbol{x}$ とし，定義域を $\mathcal{X} = \mathcal{R}^M$ とする．また画像特徴量 $\boldsymbol{x}$ に対応するクラスを表すクラスラベルを y とし，定義域 $\mathcal{Y}$ を $\{l_1, ..., l_K\}$ という K 個のクラス集合とする．

また，本章ではタスクという概念を用いる．タスクとは，3.1 節で説明したように転移学習の枠組みで用いられる用語で，学習や推定するクラスラベルの定義域 $\mathcal{Y}$ を考えたとき，それ

が同じならば同じタスク，異なるならば異なるタスクとする．ゼロショット学習の場合，訓練集合とテスト集合はクラスラベルの定義域が異なるため，それぞれを別々のタスクと考える．また，訓練集合のクラスラベルの定義域に関して学習や推定を行う問題設定を元タスク，テスト集合のクラスラベルの定義域に関して学習や推定を行う問題設定を目標タスクと呼んで区別する．

ある事例が元タスクもしくは目標タスクの事例であることを，それぞれ添え字の s と t で表す．よって元タスクの訓練集合は $\mathcal{D}_s = \{\boldsymbol{X}_s, \boldsymbol{y}_s\} = \{\boldsymbol{x}_{sn}, y_{sn}\}_{n=1}^{N_s}$，目標タスクのテスト集合は $\mathcal{D}_t = \{\boldsymbol{X}_t\} = \{\boldsymbol{x}_{tn}\}_{n=1}^{N_t}$ と表記する．また，元タスクのクラスラベルの定義域は $\mathcal{Y}_s = \{l_{s1}, ..., l_{sK_t}\}$，目標タスクのクラスラベルの定義域は $\mathcal{Y}_t = \{l_{t1}, ..., l_{tK_t}\}$ となる．なお本章の問題設定では，画像特徴量の定義域はどちらのタスクでも同じ M 次元のベクトル空間とする．

ゼロショット学習の目的は，重複しない元タスクと目標タスクのクラス集合（$\mathcal{Y}_s \cap \mathcal{Y}_t = \emptyset$）が与えられたとき，元タスクの集合 $\mathcal{D}_s = \{\boldsymbol{x}_{sn}, y_{sn}\}_{n=1}^{N_s}$ を用いて，目標タスクの集合 $\mathcal{D}_t = \{\boldsymbol{x}_{tn}\}_{n=1}^{N_t}$ のクラスラベル $\{y_{tn}\}_{n=1}^{N_t}$ を推定することである．このとき $\mathcal{D}_s$ は訓練集合，$\mathcal{D}_t$ はテスト集合としてのみ用いることに留意されたい．

属性ベースのゼロショット学習では，補助情報として異種情報である属性を導入する．属性は M 個与えられていて，その実現値を $\boldsymbol{w} = [w_1, ..., w_M]$ とし，定義域は $\mathcal{W} = \{0, 1\}^M$ とする．以降，属性の実現値のことを，単に属性値と呼ぶ．m 番目の属性値 w_m は，0 ならば m 番目の属性を持たず，1 ならば m 番目の属性を持つことを意味する．

属性値は，事例ごと，すなわち，各事例の画像特徴量とクラスラベルの両方 $\{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$ によって $\boldsymbol{W}^y = [\boldsymbol{w}^{y_1}, ..., \boldsymbol{w}^{y_N}]^T$ として定義される．元タスクの場合，属性値は $\boldsymbol{W}^{y_s} = [\boldsymbol{w}^{y_{s1}}, ..., \boldsymbol{w}^{y_{sN_s}}]^T$ となる．ただし，クラスラベルが同じになるような事例の属性値をすべて同じにする場合（即ち $y_i = y_j$ のとき $\boldsymbol{w}^{y_i} = \boldsymbol{w}^{y_j}$）は，特にクラスラベルごとの定義と呼ぶ．目標タスクの場合も $\boldsymbol{W}^{y_t} = [\boldsymbol{w}^{y_{t1}}, ..., \boldsymbol{w}^{y_{tN_t}}]^T$ のように事例ごとに属性値が定義されるが，実際の問題設定では事例集合が与えられないため，クラス集合の任意のクラス l_t に対する定義 $\boldsymbol{W}^{l_t} = [\boldsymbol{w}^{l_{t1}}, ..., \boldsymbol{w}^{l_{tK_t}}]^T$ を事前情報としてクラス推定に利用する．

以上の議論を表 7.1 でまとめる．

表 7.1 属性ベースゼロショット学習の問題定式化.

事前情報	$\mathcal{Y}_s, \mathcal{Y}_t$（ただし $\mathcal{Y}_s \cap \mathcal{Y}_t = \emptyset$） $\boldsymbol{W}^{y_s}, \boldsymbol{W}^{l_t}$
訓練集合	$\mathcal{D}_s = \{\boldsymbol{x}_{sn}, y_{sn}\}_{n=1}^{N_s}$
テスト集合	$\mathcal{D}_t = \{\boldsymbol{x}_{tn}\}_{n=1}^{N_t}$
学習目的	テスト集合のクラスラベルを推定する

7.1.2 観測確率を考慮した属性ベースゼロショット学習

本節では，属性ベースゼロショット学習を解くための新たな手法として，観測確率という概念と，それを考慮した属性ベースゼロショット学習の提案モデルの説明をする．

観測確率

それぞれの属性が入力である画像特徴量にどれくらい現れるか，即ち画像特徴量に対する各属性の現れやすさについては，属性によって異なると考えられる．たとえば，クジラの画像が RGB color histogram のような色の画像特徴量で表されているとする（図 7.1）．「blue」という属性は色の知識なので画像特徴量に現れやすいが，「hunter」属性は色情報では解釈しにくいので，この種類の画像特徴量には現れにくい．このように，用いる画像特徴量の種類が同じでも，属性が異なれば画像特徴量への現れ方は異なると考えられる．

この現れやすさは，各属性の属性値 w_m を画像特徴量 $\boldsymbol{x}$ から任意の分類器 f_m で学習するときに大きく影響する．特に現れにくい属性の場合，画像特徴量にその属性に該当するような情報が少ないため，分類器 f_m は適切に学習できないと考えられる．そのため，分類器 f_m から推定結果として得られる確信度 $p(w_m|\boldsymbol{x})$ も，同様に正しく求められない．属性ベースゼロショット学習の代表的な手法である DAP モデル [Lampert 09, Lampert 14] では，クラスラベル y の事後分布 $p(y|\boldsymbol{x})$ を確信度 $p(w_m|\boldsymbol{x})$ を用いて $p(y|\boldsymbol{x}) \propto \prod_m \frac{p(w_m^y|\boldsymbol{x})}{p(w_m^y)}$ として求めて，クラスラベル y の推定をしている．しかし，画像特徴量への現れやすさを考慮していないため，ある属性 w_m が画像特徴量 $\boldsymbol{x}$ に現れにくい場合，確信度 $p(w_m|\boldsymbol{x})$ が正しく推定できなくなり，その影響からクラスラベル y の事後分布 $p(y|\boldsymbol{x})$ の良い推定を得られなくなる恐れが

図 7.1　属性による画像特徴量への現れやすさの違い.

ある.

以上の理由から，本研究では画像特徴量に対する各属性の現れやすさという度合いが重要であると考え，この度合いを考慮した手法を提案する．今後の議論のためにこの度合いを属性ごとの**観測確率**と呼ぶこととする.

提案モデル

本研究では，観測確率を導入した新たな属性ベースゼロショット学習の生成モデルを提案する．生成モデルとは，データの生成過程を明示的に記述するモデルであり，本研究ではクラスラベルから画像特徴量が生成される過程をモデル化する．図 7.2 は提案する生成モデルをグラフィカルモデルで表したものである．この生成過程は，元タスク・目標タスク両方で共通とする.

前述したように，画像特徴量から観測される属性値と，クラスラベルに対して予め定義されている属性値が異なる場合がある．本研究ではこの 2 つを異なる属性として明確に区別し，前者を**観測された属性**と呼び，その実現値を $\boldsymbol{c} = [c_1, ..., c_M]$，定義域を $\mathcal{C} = \{0, 1\}^M$ とする．また後者については，7.1.1 節で設定した属性と同義であるため，その属性値も同様に $\boldsymbol{w}$ と表記する．ただし観測された属性と区別する場合は，**真の属性**と呼ぶ．この 2 つの属性値が同じならば，その属性は画像特徴量に現れやすく，異なるならば現れにくいということになる.

次に，生成過程とその分布の形について順番に説明する．事例 n のクラスラベル y_n が与えられれば，事前情報として与えられている定義 $\boldsymbol{w}^{y_n}$ によって真の属性の属性値が定まる．ただし，属性値の定義は事例ごとにされているため，クラスラベルが同じでも真の属性値が同じ値とは限らないことに留意する．この生成過程は $\boldsymbol{w}_n \sim p(\boldsymbol{w}_n|y_n) = [[\boldsymbol{w}_n = \boldsymbol{w}^{y_n}]]$ と表現できる．ただし $[[P]]$ は Iverson の記法で，P が真のとき 1，偽のとき 0 をとる.

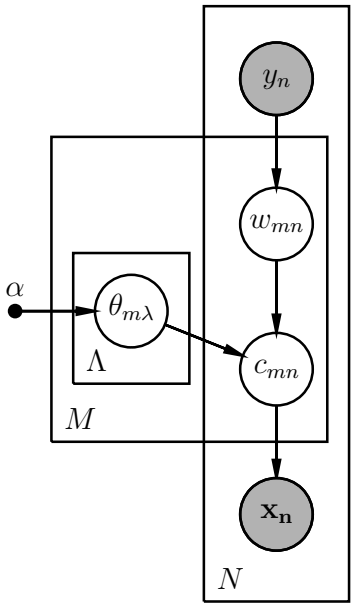


図 7.2 提案モデルのグラフィカルモデル.

真の属性から観測された属性への生成過程を考えると，生成される間に何からの要因によって，属性値が変化していると考えられ，その変化の確率は現れやすさ，即ち観測確率と同義である．よって観測確率を，m 番目の真の属性の属性値 $\lambda = w_{mn}$ から，観測された属性の属性値 c_{mn} を生成する確率変数 $\theta_{m\lambda}$ として定義する．属性値の定義域は $\{0, 1\}$ の 2 値なので，観測された属性はベルヌーイ分布に従って生成されると考え，観測確率はベルヌーイ分布のパラメータとする．したがって，観測された属性の生成過程は $c_{mn} \sim p(c_{mn}|w_{mn}, \theta_{m\lambda}) = p(c_{mn}|\theta_{mw_{mn}}) = \mathrm{Bern}(c_{mn}; \theta_{mw_{mn}}) = \theta_{mw_{mn}}^{c_{mn}}(1 - \theta_{mw_{mn}})^{1-c_{mn}}$ となる．

さらに，ベルヌーイ分布の共役事前分布はベータ分布なので，パラメータである観測確率 $\theta_{m\lambda}$ はハイパーパラメータ $\boldsymbol{\alpha} = [\alpha_0, \alpha_1]$ によって $\theta_{m\lambda} \sim p(\theta_{m\lambda}|\boldsymbol{\alpha}) = \mathrm{Beta}(\theta_{m\lambda}; \boldsymbol{\alpha}) = \frac{\theta_{m\lambda}^{\alpha_0-1}(1-\theta_{m\lambda})^{\alpha_1-1}}{B(\boldsymbol{\alpha})}$ のように生成される．ただし，$B(\boldsymbol{\alpha})$ はベータ関数である．

画像特徴量 $\boldsymbol{x}_n$ は観測された属性から $\boldsymbol{x}_n \sim p(\boldsymbol{x}_n|\boldsymbol{c}_n)$ として生成される．ただし，この生成過程の形については次節で説明する．

この生成モデルでは観測確率について次の 2 つの仮定をしている．

仮定 1 観測確率は属性によって値が異なる

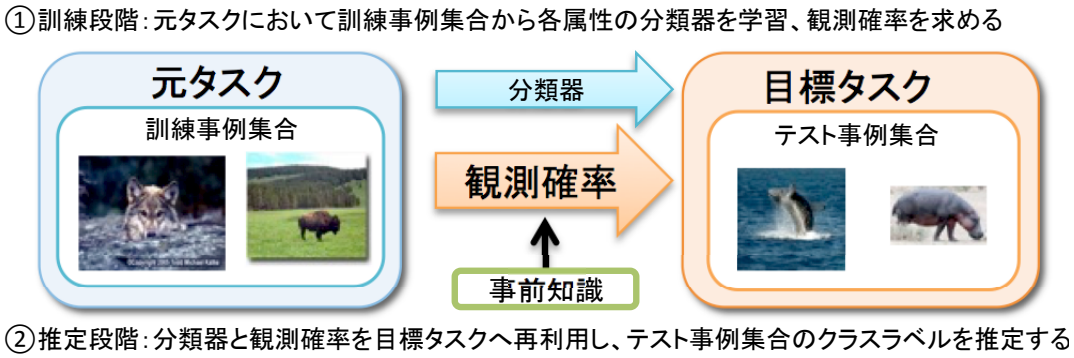


図 7.3　訓練段階と推定段階の概要図.

仮定 2　画像特徴量の種類が同じならば，異なるタスクで観測確率が等しい

これらは提案モデルの前提となる仮定であり，実験でこれらの仮定の妥当性を検証する必要がある.

7.1.3　提案モデルの訓練・推定

提案手法では，元タスクの訓練集合から提案モデルのパラメータを訓練する**訓練段階**と，目標タスクで元タスクで学習したパラメータを再利用し，テスト集合からクラスラベルを推定する**推定段階**がある.

まず，提案モデルから訓練段階及び推定段階の計算を導出する．次に，観測確率を異なるタスクで近づける工夫を説明する.

本節で最終的に得られる訓練段階と推定段階のアルゴリズムをそれぞれ Algorithm 1 と Algorithm 2 に示す．また，図 7.3 で訓練，推定段階の概要を示す.

訓練段階

訓練段階では，元タスクの訓練集合 $\mathcal{D}_s$ が与えられたときの提案モデルのパラメータ，即ち観測確率を学習する.

生成モデルからパラメータを推定する一般的な方法として，EM アルゴリズムがよく知られているが，本研究ではそれとは異なるアプローチをとる．これは，既存研究の DAP モデルには，訓練段階で画像特徴量に応じて任意の分類器を利用できるという特徴があり，本研究の提案モデルでもその特徴を残すためである．提案モデルでは以降説明する工夫によって，生成

Algorithm 1 Training algorithm of our proposed model.

Input: $\mathcal{D}_s$, $\boldsymbol{W}^{y_s}$

Output: $\tilde{\Theta}, \mathbf{f}$

Separate $\mathcal{D}_s$ into $\mathcal{D}_{tr}$ and $\mathcal{D}_v$
for m in $\{1, ..., M\}$ **do**
  Train probabilistic classifier $f_m : \mathcal{X} \to \mathcal{W}_m$
  $\tilde{\theta}_{m0} = \tilde{\theta}_{m1} = N_{vm0} = N_{vm1} = 0$
  for n in $\{1, ..., N_v\}$ **do**
    Estimate $p(c_{mn}|\boldsymbol{x}_{vn})$ from f_m
    if $w_m^{y_n} == 0$ **then**
      $\tilde{\theta}_{m0} = \tilde{\theta}_{m0} + p(c_m = 1|\boldsymbol{x}_{vn})$
      $N_{vm0} = N_{vm0} + 1$
    else
      $\tilde{\theta}_{m1} = \tilde{\theta}_{m1} + p(c_m = 1|\boldsymbol{x}_{vn})$
      $N_{vm1} = N_{vm1} + 1$
    end if
  end for
end for
return $\tilde{\Theta} = \{\tilde{\theta}_{m0}, \tilde{\theta}_{m1}, N_{vm0}, N_{vm1}\}_{m=1}^{M}$, $\mathbf{f} = \{f_m\}_{m=1}^{M}$

モデルでありながら DAP モデルと同様に，任意の分類器で学習できる．またこの工夫によって，真の属性，観測された属性及び観測確率に明確な解釈が与えられる．

モデルの設計の段階で $p(\boldsymbol{x}_n|\boldsymbol{c}_n)$ の分布の形を明示化しなかったが，この分布をベイズの定理により $p(\boldsymbol{x}_n|\boldsymbol{c}_n) = \frac{p(\boldsymbol{c}_n|\boldsymbol{x}_n)p(\boldsymbol{x}_n)}{p(\boldsymbol{c}_n)} = \prod_m \frac{p(c_{mn}|\boldsymbol{x}_n)p(\boldsymbol{x}_n)}{p(c_{mn})}$ とする．ここで観測された属性が独立，及び画像特徴量に対して条件付き独立であると仮定している．この仮定は，DAP モデルでも暗黙的に設定されていたものであり，本章でもこの仮定を採用する．

次に，条件付き独立とした条件付き確率 $p(c_{mn}|\boldsymbol{x}_n)$ に着目する．本章ではこの条件付き確率を，分類器 $f_m : \mathcal{X} \to \mathcal{W}_m$ の確信度として得たものと考える．その理由を以下で説明する．

元タスクの訓練集合 $\mathcal{D}_s$ を $\mathcal{D}_{tr}$ と $\mathcal{D}_v$ に分け，$\mathcal{D}_{tr}$ を訓練集合として画像特徴量から属性への写像を得る．具体的には各属性の属性値 w_m についてそれぞれ任意の分類器 f_m を用意し $f_m : \mathcal{X} \to \mathcal{W}_m$ を学習する．なお，この学習に必要な各事例の属性値ラベル $\boldsymbol{w}_n$ は予め定義さ

Algorithm 2 Estimation algorithm of our proposed model.

Input: $\mathcal{D}_t$, $\tilde{\Theta}$, $\mathbf{f}$, $\boldsymbol{W}^{l_t}$, $\boldsymbol{\alpha}$, $\boldsymbol{\eta}$

Output: $\boldsymbol{y}_t$

for n in $\{1, ..., N_t\}$ **do**
  for m in $\{1, ..., M\}$ **do**
    Estimate $p(c_{mn}|\boldsymbol{x}_{tn})$ from f_m
  end for
end for
for m in $\{1, ..., M\}$ **do**
  $p(c_m) = \frac{\sum_n p(c_{mn}|\boldsymbol{x}_{tn})}{N_t}$
  for λ in $\{0, 1\}$ **do**
    $\alpha_\lambda^{sample} = 0$
    for j in $\{1, ..., \eta_\lambda N_{vm\lambda}\}$ **do**
      $Sample \sim p(c_m = \lambda)$
      $\alpha_\lambda^{sample} = \alpha_\lambda^{sample} + Sample$
    end for
  end for
  $\theta_{m0} = \frac{\tilde{\theta}_{m0}+\alpha_0+\alpha_0^{sample}-1}{N_{vm0}+\alpha_0+\alpha_0^{sample}+\alpha_1+\alpha_1^{sample}-2}$
  $\theta_{m1} = \frac{\tilde{\theta}_{m1}+\alpha_0+\alpha_1^{sample}-1}{N_{vm1}+\alpha_0+\alpha_0^{sample}+\alpha_1+\alpha_1^{sample}-2}$
end for
for n in $\{1, ..., N_t\}$ **do**
  Estimate y_{tn} by Equation (7.1)
end for
return $\boldsymbol{y}_t = \{y_{tn}\}_{n=1}^{N_t}$

れていることに留意されたい．学習後 $\mathcal{D}_v$ を検証用集合とし，その画像特徴量 $\boldsymbol{x}_{vn}$ を各属性の分類器 f_m の入力として与えることで出力として各属性の確信度 $\hat{p}(w_{mn}|\boldsymbol{x}_{vn})$ を得る．しかし，この確信度は真の属性の確信度ではなく，学習していない未知の検証用集合から属性を観測できた度合いである．よって，この確信度を観測された属性の c_{mn} の確信度 $p(c_{mn}|\boldsymbol{x}_{vn})$ とする．

以上の議論によって，真の属性，観測された属性及び観測確率について，分類器 f_m によっ

て明確な解釈が与えられる．真の属性とは，分類器 f_m のラベルとして与えられる値である．また観測された属性は，分類器 f_m が未知の画像特徴量から得られた値であり，その度合いは確信度として得られる．そして観測確率は，分類器 f_m が画像特徴量から属性を観測できた確率であり，生成モデルとしては定義された属性値と未知の画像特徴量から得られた属性値の違いの原因として解釈できる．

$\mathcal{D}_v$ の画像特徴量 $\boldsymbol{X}_v$ が与えられたときの観測確率の事後分布は $p(\theta_{m\lambda}|\boldsymbol{X}_v)$ であり，訓練段階では，この確率を最大にするような観測確率 $\theta_{m\lambda}$ を求めたい．本研究では次のような最大事後確率推定（MAP 推定）によって解く．

$$\hat{\theta}_{m\lambda} = \arg\max_{\theta_{m\lambda}} p(\theta_{m\lambda}|\boldsymbol{X}_v). \tag{7.1}$$

$p(\theta_{m\lambda}|\boldsymbol{x}_v)$ を直接最大化することは困難だが，対数をとって下界を求めると解析的に解くことができ，次のように求まる．

$$\hat{\theta}_{m\lambda} = \frac{\sum_{n:w_m^{y_{vn}}=\lambda} p(c_{mn}=1|\boldsymbol{x}_{vn}) + \alpha_0 - 1}{N_{vm\lambda} + \alpha_0 + \alpha_1 - 2}. \tag{7.2}$$

ただし $p(c_{mn}=1|\boldsymbol{x}_{vn})$ は検証用集合から得た確信度，$N_{vm\lambda} = \sum_{n:w_m^{y_{vn}}}$ である．式 (7.2) の詳しい導出は付録 B に記載する．

なお，式 (7.2) の $\sum_{n:w_m^{y_{vn}}}$ の部分から，$\boldsymbol{x}_{vn}$ にクラスラベル y_{vn} が与えられていないと，式 (7.2) は解くことができないことに留意されたい．

推定段階

推定段階では，訓練段階で学習した観測確率と分類器を再利用して，目標タスクのテスト集合 $\mathcal{D}_t$ のクラスラベルを推定する．

まず，提案モデルからクラスラベル y_{tn} の事後分布 $p(y_{tn}|\boldsymbol{x}_{tn})$ を求める式を導出する．

テスト集合 $\mathcal{D}_t$ の画像特徴量 $\boldsymbol{x}_{tn}$ 及びクラスラベル y_{tn} の同時分布は，提案モデルより次の

ようになる．

$$
\begin{aligned}
&p(\boldsymbol{x}_{tn}, y_{tn}|\boldsymbol{\theta}) \\
&= \sum_{\boldsymbol{c}} \sum_{\boldsymbol{w}} p(\boldsymbol{x}_{tn}, y_{tn}, \boldsymbol{c}, \boldsymbol{w}|\boldsymbol{\theta}) \\
&= p(y_{tn}) \prod_{m} \sum_{c_m} \sum_{w_m} p(w_m|y_{tn}) p(\boldsymbol{x}_{tn}|c_m) p(c_m|w_m, \theta_{m\lambda}) \\
&\simeq p(y_{tn}) p(\boldsymbol{x}_{tn}) \prod_{m} \sum_{c_m} \frac{p(c_m|\theta_{mw_m^{y_{tn}}}) p(c_m|\boldsymbol{x}_{tn})}{p(c_m)}.
\end{aligned} \tag{7.3}
$$

式 (7.3) から画像特徴量 $\boldsymbol{x}_{tn}$ に対するクラスラベル y_{tn} の事後分布は，次のようになる．

$$
\begin{aligned}
p(y_{tn}|\boldsymbol{x}_{tn}) &= \frac{p(\boldsymbol{x}_{tn}, y_{tn})}{p(\boldsymbol{x}_{tn})} \\
&= p(y_{tn}) \prod_{m} \sum_{c_m} \frac{p(c_m|\theta_{mw_m^{y_{tn}}}) p(c_m|\boldsymbol{x}_{tn})}{p(c_m)} \\
&\propto \prod_{m} \sum_{c_m} \frac{p(c_m|\theta_{mw_m^{y_{tn}}}) p(c_m|\boldsymbol{x}_{tn})}{p(c_m)}.
\end{aligned} \tag{7.4}
$$

ただし，式変形の中で $p(y_{tn})$ を一様分布と仮定した．

式 (7.4) を計算するためには，$p(c_m|\theta_{mw_m^y})$，$p(c_m|\boldsymbol{x})$，$p(c_m)$ をそれぞれ求める必要がある．

確信度 $p(c_m|\boldsymbol{x})$ は，元タスクの訓練集合で学習した分類器 f_m を再利用し $p(c_m|\boldsymbol{x}) = f_m(\boldsymbol{x})$ として推定できる．

周辺分布 $p(c_m)$ は $p(c_m) = \int p(\boldsymbol{x}, c_m) d\boldsymbol{x} = \int p(\boldsymbol{x}) p(c_m|\boldsymbol{x}) d\boldsymbol{x} \simeq \frac{1}{N_t} \sum_n p(c_{mn}|\boldsymbol{x}_{tn})$ として求められる．ただし $p(c_{mn}|\boldsymbol{x}_{tn})$ は分類器 f_m から求めた確信度である．

$p(c_m|\theta_{mw_m^y})$ はベルヌーイ分布であり，分布のパラメータ，即ち観測確率 $\theta_{m\lambda}$ は，訓練段階において元タスクで推定したものを再利用する．

推定するクラスラベル $\hat{y}_{tn}$ は目標タスクのクラス集合 $\mathcal{Y}_t$ の要素の何れかであり，式 (7.4) の事後分布が最大になるようなクラスを選択する（MAP 推定）．

$$
\begin{aligned}
\hat{y}_{tn} &= \arg\max_{l_{tk} \in \mathcal{Y}_t} p(y_{tn} = l_{tk}|\boldsymbol{x}_{tn}) \\
&= \arg\max_{l_{tk} \in \mathcal{Y}_t} \prod_{m} \sum_{c_m} \frac{p(c_m|\theta_{mw_m^{l_{tk}}}) p(c_m|\boldsymbol{x}_{tn})}{p(c_m)}.
\end{aligned} \tag{7.5}
$$

この推定は，目標タスクのすべてのクラスについて，それぞれ式 (7.4) の事後分布を計算し，それらの中で最大となる事後分布のクラスを選択することで，容易に求まる．

タスク間の観測確率の違いの補正

ここで仮定 2 の妥当性について，式 (7.2) で求めた観測確率の推定式から検討する．式 (7.2) 中の観測された属性の事後分布 $p(c_{mn}=1|\boldsymbol{x}_n)$ は，画像特徴量の種類が同じならば，事例が異なっても似た傾向になると考えられる．この前提に従えば，求めた観測確率も事例によらないことになり，仮定 2 は成り立つと考えられる．しかし，属性値は定義より，事例，即ち画像特徴量とクラスラベルが与えられたときに一意に定まるため，事例が異なる場合には事後分布は厳密には異なる値となる．つまり，元タスクの検証用事例を増やして観測確率の精度を上げようとしても，属性値の定義が異なるため，目標タスクをテスト事例としたときの観測確率とは同じにはならない．よって，目標タスクの推定に利用する観測確率は，正確には同じ目標タスクの事例の事後分布 $p(c_{mn}=1|\boldsymbol{x}_{tn})$ から求めた観測確率でなければならない．この観測確率は，目標タスクの事例を用いて得られた観測確率であるため，目標タスクの観測確率 θ^{target} と呼ぶ．しかし実際の問題設定では目標タスクのクラスラベルが得られないため，事後分布を求めても式 (7.2) から目標タスクの観測確率を計算することができない．

この問題を解決するため，本章ではタスク間の観測確率の違いを補正する手法を提案する．式 (7.2) の $\boldsymbol{\alpha}$ は，元々観測確率を生成するベータ分布 $\mathrm{Beta}(\theta_{m\lambda};\boldsymbol{\alpha})=\frac{\theta_{m\lambda}^{\alpha_0-1}(1-\theta_{m\lambda})^{\alpha_1-1}}{B(\boldsymbol{\alpha})}$ のハイパーパラメータであり，どれだけ属性値を観測したかという有効観測数に該当する．さらにハイパーパラメータを m 番目の属性ごと（$\alpha_{m\lambda}$）に考えることで，各属性の観測数，即ち事前知識を取り入れることが可能となる．本手法では，目標タスクの属性値に従ってハイパーパラメータを調節することで，観測確率が目標タスクに近づくように補正する．

目標タスクで求めた事例ごとの事後分布 $p(c_{mn}=1|\boldsymbol{x}_{tn})$ から，観測された属性の周辺分布 $p(c_m)$ を求める．この周辺分布は目標タスクにおいて属性値 c_m の観測される傾向を表している．これを事前知識として反映させるために，周辺分布 $p(c_m=\lambda)$ をベルヌーイ分布として任意の数サンプリングし，その総和 $\alpha_{m\lambda}^{sample}$ をハイパーパラメータとして式 (7.2) の分子に加える．また，式 (7.2) の分母には正規化のため $p(c_m=0)$ と $p(c_m=1)$ のサンプリング総和 α_{m0}^{sample} と α_{m1}^{sample} を加える．よって，タスクの違いを補正した観測確率の推定式は次のよう

になる．

$$\hat{\theta}_{m\lambda} = \frac{\sum_{n:w_m^{y_{vn}}=\lambda} p(c_{mn}=1|x_{vn}) + \alpha_0 + \alpha_{m\lambda}^{sample} - 1}{N_{vm\lambda} + \alpha_0 + \alpha_{m0}^{sample} + \alpha_1 + \alpha_{m1}^{sample} - 2}. \tag{7.6}$$

事前知識を取り入れる程度はサンプリング回数によって決まる．本研究では，元タスクの検証用集合の総数 $N_{vm\lambda}$ に対する割合 η_λ で調節する．ただし割合 η_λ を大きくしすぎると，観測確率 $\theta_{m\lambda}$ が $p(c_m = \lambda)$ と等価になってしまうので，なるべくサンプリング回数を検証用集合数以下（$\eta_\lambda \leq 1$）にするべきと考えられる．

なお，本手法では目標タスクの事後分布を求めないと観測確率が求まらないため，元タスクで正規化前の観測確率 $\tilde{\theta}_{m\lambda} = \sum_{n:w_m^{y_{vn}}=\lambda} p(c_{mn}=1|\boldsymbol{x}_{vn})$ と正規化定数 $N_{vm\lambda} = \sum_{n:w_m^{y_{vn}}=\lambda}$ を求めておく．推定段階でこれらを再利用し，目標タスクでのサンプリング値から，式 (7.6) を計算して観測確率を求める．

訓練段階と推定段階のアルゴリズムの擬似コードである Algorithm 1 と Algorithm 2 では，本節で述べたサンプリングの方法を用いた学習・推定アルゴリズムが記述されている．

7.2　関連研究

ゼロショット学習の関連研究については 3.2.4 節で説明したが，ここでは既存研究と本研究との違いを挙げて，提案手法の新規性を明確化する．

本研究のベースとなる既存研究は Lampert らの DAP モデル [Lampert 09, Lampert 14] である．DAP モデルでは，予め属性のリストを考え，訓練集合とテスト集合で考えているすべてのクラスについて属性リストとの関係を定義し，それを補助情報として用いる．

本研究の提案モデルが DAP モデルと最も異なる点は，観測確率という新しい概念を導入したことである．また，DAP モデルが識別モデルであるのに対して，提案モデルは画像特徴量の生成過程を記述した生成モデルで構築されており，モデルの種類が異なる．さらに，DAP モデルがゼロショット学習しか対応していないのに対して，提案モデルは観測確率によってゼロショット以外の任意のショット数でも学習が可能である．その一方で，DAP モデルの学習に任意の分類器を利用できるという利点は，本研究でも共通である．

本研究で導入する各属性の観測確率は，各属性の画像特徴量への現れやすさの度合いを定式化したものであり，属性ベースゼロショット学習において，本研究で新たに着目した概念であ

る．観測確率と似たような概念に着目した研究として Parikh らの研究がある [Parikh 11a]．Parikh らは，属性には人間にとって理解しやすい（understandable）もしくは識別しやすい（discriminative）属性があり，そのような属性を人間とインタラクションして生成するシステムを提案している．また，Yu らは一般的に人間が画像から認識できるかに応じて属性を visual 属性と non-visual 属性に分けている [Yu 10]．このように，観測確率と似た概念はこれまでも着目されてきたが，本質的な部分でこれらの概念と観測確率は異なる．これまでの研究では，通常の画像に対して人間が判別できるかどうかに着目していたが，本研究の観測確率は，画像特徴量に対してモデルもしくは分類器が観測できるかどうかに焦点を当てている．これは，実際に学習や推定を行うのは分類器であり，人間にとって観測できるような属性を選んだところで，必ずしもうまく学習器に認識されるとは限らないからである．このような人間と分類器の判断が異なる現象が発生することは，後述する実験で実証する．さらに，これまでの研究では学習前に人間が属性を選別する必要があったが，本研究の提案モデルでは，観測確率はパラメータとして含まれているため，学習と同時に観測確率を求め，予め属性を選別しなくても影響を考慮することができる．

他に観測確率に近い概念として，筆者らは各属性のラベルの偏りや正解率を表す予測能力を定義し，それを DAP モデルに重み付けすることを提案した [Suzuki 14a]．しかし，この手法の有効性は DAP モデルと正解率で比較したときの僅かな向上しか確認できなかった．また文献 [Suzuki 14b] では本章と同様の観測確率を含めたモデルを提案したが，訓練集合とテスト集合のそれぞれで求めた観測確率が異なる場合を考慮できていなかった．本章ではこの点を改善し，さらに複数の検証実験によって，提案手法の有効性を確認している．

生成モデルによる属性ベースゼロショット学習は Yu らにも提案されている [Yu 10]．Yu らは Author-Topic モデルをヒントに Category-Topic モデルを提案している．このモデルと本章の提案モデルが異なる点は，Yu らのモデルがトピックモデルのため画像特徴量として visual word を使わなければならないのに対し，本研究の提案手法は画像特徴量に関する制限がないというところである．その他にも，本章の提案モデルには観測確率が含まれているなど，同じ生成モデルでも異なる点が多い．

Fu らは，クラスによって属性の分布が異なるという問題があることを，projection domain shift 問題として指摘している [Fu 14a]．Fu らは属性の他に Wikipedia から抽出した言語ベ

クトル，さらに画像特徴量の 3 つを併用し正準相関分析をすることで，この問題を解決している．本章の提案手法では，観測確率の事前知識としてテスト集合での属性の周辺分布を考慮することで，属性以外の知識を加えずに，訓練集合とテスト集合での観測確率の違いの解消を試みている．

7.3　検証実験

本節では既存研究との比較実験の前に，提案モデルで設定した仮定などについて実験によって検証し，提案手法の妥当性について議論する．

7.3.2 節で仮定 1 と仮定 2 の検証実験をし，7.3.3 節で人間と分類器での属性の観測しやすさの違いを検証して観測確率を用いる妥当性について議論する．そして 7.3.4 節で目標タスクの観測確率に近づける手法の有効性を検証する．

7.3.1　データ集合

本節では Lampert ら [Lampert 09, Lampert 14] によって作成された Animals with Attributes（以下 AwA）[*1] と Farhadi ら [Farhadi 09] によって作成された aPascal-aYahoo（以下 aP-aY）[*2] の 2 種類のデータ集合で実験を行った．これら 2 つのデータ集合の最も大きな違いは，aP-aY では属性が事例ごとに定義されている一方で，AwA ではクラスごと（$y_i = y_j$ のとき $w^{y_i} = w^{y_j}$）に定義されているという点である．また AwA の属性は画像とは無関係なものが多い一方で，aP-aY は画像から認識しやすいような属性が付けられている．その他の違いは表 7.2 でまとめた．

7.3.2　検証 1：仮定 1 と仮定 2 の検証

本実験では，提案モデルでの 2 つの仮定を検証する．

用いるデータ集合は AwA とし，元タスクと目標タスクのクラスは，それぞれデフォルトで分けられている訓練集合とテスト集合のクラスラベルの定義域とした（元タスクが 40 クラス，目標タスクが 10 クラス）．元タスクでの検証用事例数は，元タスクの集合全体の 10% と

[*1] http://attributes.kyb.tuebingen.mpg.de
[*2] http://vision.cs.uiuc.edu/attributes

表 7.2 実験で用いるデータ集合の違い.

データ集合	Animals with Attributes	aPascal-aYahoo
画像数	30475	15339
クラス数	50	32（Pascal 20・Yahoo 12）
属性の数	85	64
属性の定義	クラスごと	事例ごと
クラスの種類	動物	動物・乗り物・家電

した．ハイパーパラメータは $\alpha_0 = \alpha_1 = 2$ とした．なお，本実験では 7.1.3 節で示したサンプリングの方法は利用していない．画像特徴量には，ILSVRC2012 データ集合で事前学習済みの 7 層の畳み込み深層ニューラルネットワーク AlexNet [Krizhevsky 12] の fc7 層目の出力（4096 次元）を用いた．この特徴量は DeCAF [Donahue 14] と呼ばれているため，本章では以降 DeCAF と呼ぶ．分類器には L2 正則化ロジスティック回帰を用いた．検証 1 では，分類器を固定した上で提案手法による効果を確認するため，ロジスティック回帰のパラメータ C は 1.0 と固定した．実装は Python 2.7[*3]で行い，機械学習ライブラリ scikit-learn 0.15.2[*4]を利用した．

仮定 1 と仮定 2 が妥当かどうか検証するために，各属性の元タスクで求めた観測確率と目標タスクの観測確率をプロットする．AwA はクラスごとに用意されている事例数が異なるので，偏りをなくすため各クラスの事例数を 90 枚とした．7.1.3 節でも述べたとおり，目標タスクの観測確率は実際の問題設定では求められないが，この実験では目標タスクにもラベルがあるものとして計算している．

図 7.4 が検証結果である．上の図が元タスクで求めた観測確率 θ_{m0} と目標タスクの観測確率 θ_{m0}^{target} の値で，下の図が θ_{m1} と θ_{m1}^{target} の値を示す．横軸が m 番目の属性，縦軸が観測確率の値を表す．青が元タスクの観測確率で，緑が目標タスクの観測確率を表す．

まず，元タスクの観測確率について着目する．図 7.4 から，仮定 1 で仮定したとおり属性によって観測確率の値が大きく異なることがわかる．もしすべての属性が完全に画像に現れて適

[*3] https://www.python.org
[*4] http://scikit-learn.org

切に学習できれば，θ_{m0} はすべて 0，θ_{m1} はすべて 1 となっているはずである．図 7.4 をみると全体の傾向として θ_{m0} は 0 に近く，θ_{m1} は 1 に近いように分布しているが，個々の属性にはかなりばらつきがあることがわかる．

次に図 7.4 について，観測確率と目標タスクの観測確率の分布の違いに着目する．この図から仮定 2 で仮定したとおり，おおよそ近い分布になっていることが確認できる．しかし完全に一致している訳ではなく，7.1.3 節で議論したようにタスクの違いが影響していると考えられる．

以上の検証によって提案モデルでの仮定は概ね妥当であることを示した．

7.3.3　検証 2：観測確率の妥当性の検証

提案手法では，分類器の各属性の確信度から属性ごとの観測確率を求めている．似たような概念として，Parikh ら [Parikh 11a] は人間にとって理解しやすい属性を考え，人間とのインタラクションによってこのような属性を生成するシステムを提案している．また，Yu ら [Yu 10] は AwA の属性のうち一般的に人間が画像から理解できるであろう属性を visual 属性とし，それ以外を non-visual 属性として区別した．

本節では，分類器にとって画像特徴量から観測しやすい属性と人間が通常の画像から判断できる属性を比較検証し，本研究の問題設定において，観測確率を用いること妥当性について議論する．

観測確率が高い属性と低い属性がどのようなものかを検証するために，図 7.4 の結果から，観測確率による属性の上位 5 個と下位 5 個を表 7.3 に示した．表中の太文字の属性が Yu らによる visual 属性である．また，visual 属性と観測確率の関係の定量的な評価値として Area Under the Curve（AUC）による評価も載せている．表中の AUC は観測確率が visual 属性と non-visual 属性を適切に分類できたかを表している．表 7.3 では，θ_{m1} が順位が適切に分類できた属性の順位であるのに対し，θ_{m0} は逆に順位が低い属性がより適切に分類できた属性となることに注意する．これは AUC による評価値も同様で，θ_{m0} のときは 1 で完全に正しく分類，0.5 でランダム，0 ですべて逆に分類したことになるが，θ_{m1} ではすべて逆となる．

表 7.3 から，visual 属性，即ち人間が画像から判断できるような属性が，必ずしも高い観測確率になるとは限らないということがわかる．θ_{m1} の oldworld や fast などは，人間は明らか

表 7.3 観測確率による属性の順位と AUC による評価（太文字の属性は visual 属性）.

観測確率	上位 5 位	下位 5 位	AUC
θ_{m0}	oldworld	skimmer	0.365
	fast	**desert**	
	chewtheeth	**red**	
	tail	flys	
	newworld	plankton	
θ_{m1}	quadrapedal	**red**	0.431
	furry	flys	
	fast	**cave**	
	ground	scavenger	
	oldworld	insects	

に画像に現れないと判断すると思われる属性である．さらに，oldworld や fast，red などのように θ_{m0} と θ_{m1} のどちらにも上位または下位にくる属性がよく見られる．これは，属性ラベルがインバランスであるなどの理由で，真の属性の属性値がどちらの場合も観測された属性の片方の属性値しか観測できない状態になっているためと考えられる．また，θ_{m0} と θ_{m1} の両方の場合で AUC が 0.5 に近いことから，どちらの場合もほぼランダムな分類となっていることがわかる．

以上のように，分類器にとって画像特徴量から観測しやすい属性と，人間が通常の画像から判断できる属性には，関係性が低いことが示された．この結果から，予め人間が分類器が適切に学習できそうな属性を選択することには限界があると考えられる．また，そのような属性を人間が判断しようとすると，それだけ人的コストもかかってしまう．その一方，観測確率を用いれば，人間が予め判別せずに訓練段階で分類器が画像特徴量からうまく学習できるかどうかを推定することができる．よって本研究の問題設定では，人手で定義した属性の観測しやすさではなく，観測確率を用いる方が妥当であると考えられる．

7.3.4　検証 3：「タスク間の観測確率の違いの補正」の検証

7.1.3 節で，観測確率を目標タスクの観測確率に近づける方法として，サンプリングによるタスク間の違いの軽減を提案した．本実験では，この手法の効果を検証実験によって評価する．AwA のすべての訓練集合を利用し，検証用集合は訓練集合のうち各クラスについて 30 枚とする．観測確率と目標タスクの観測確率の分布がどれだけ近いのかを定量的に評価するため，本章では χ^2 値を利用する．正規化定数 $N_{vm\lambda}$ あたりのサンプル割合 $\boldsymbol{\eta}$ を $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$ と変化させて，サンプル数による χ^2 値の変化を検証する．また，その際のクラス平均正解率も併せて評価する．ただし η_0 と η_1 は同じ値とする．各割合で 5 回実行し，その平均で評価する．

表 7.4　サンプリングの割合を変化させた際の χ^2 値（上）とクラス平均正解率（下）．

観測確率	サンプリングの割合					
	0	0.2	0.4	0.6	0.8	1.0
θ_{m0}	2.26	1.27	1.60	2.06	2.43	2.82
θ_{m1}	15.20	5.62	4.29	3.96	3.84	3.91

サンプリングの割合						目標タスクの観測確率
0	0.2	0.4	0.6	0.8	1.0	
.457	.457	.460	.465	.465	.461	.511

表 7.4 が実験結果である．サンプル数を増やすことで χ^2 値が低くなっていることから，タスク間の観測確率の距離が縮まっていることがわかる．また，それに応じてクラス平均正解率が向上し，目標タスクの観測確率に近くなっていることが確認できる．しかし，割合が 1.0 のように増えすぎると χ^2 値が大きくなり，クラス平均正解率も低下してしまうことがわかった．これは 7.1.3 節で述べたように，事前知識である $p(c_m)$ の影響が大きくなってしまうためと考えられる．

7.4　既存研究との比較実験

本節では提案手法を既存研究と比較実験し，提案手法の有効性について議論する．

7.4.1 節では，AwA を用いた既存研究との比較実験をし，7.4.2 節で aP-aY を使って既存研究と比較する．そして 7.4.3 節で，観測確率の近さに関する考察をする．

7.4.1　実験 1：Animals with Attributes

実験 1(a)：DAP モデルとの比較

ベースライン手法を DAP モデル [Lampert 09, Lampert 14] とし，本章の提案手法との比較を行った．この実験では提案モデルと DAP モデルでデータ集合を AwA とした際のクラス平均正解率で評価した．元タスクと目標タスクのクラスは，それぞれデフォルトで分けられている訓練集合とテスト集合のクラスラベルの定義域とする．元タスクの訓練事例数は各クラスあたり 10 枚から 90 枚へと 10 枚ずつ増やし，それぞれで訓練段階を行った．また，元タスクでの検証用事例数は，元タスクの訓練集合の 10% とした．目標タスクのテスト集合は各クラスあたり 90 枚で固定し，10 クラス分類を推定する．画像特徴量は DeCAF を使い，分類器は既存手法と DAP モデル共にパラメータ C を 1.0 とした L2 ロジスティック回帰を用いた．ハイパーパラメータは $\alpha_0 = \alpha_1 = 2$ とした．また，7.1.3 節で示したサンプリングによる方法を採用し，正規化定数 $N_{vm\lambda}$ あたりのサンプル割合 $\boldsymbol{\eta}$ は，$\eta_0 = \eta_1 = 0.2$ とした．元タスクの訓練集合は，データ集合の訓練集合から各クラスあたりランダムに選択するが，選び方の影響を軽減するため，評価指標は各枚数で 5 回実験したそれぞれのクラス平均正解率の平均とする．

クラス平均正解率の比較結果が，図 7.5 である．横軸が元タスクの各クラスごとの訓練事例数で縦軸がクラス平均正解率である．青線が提案手法，緑線が DAP モデルを表し，各グラフのエラーバーは 5 回実験した結果の標準偏差を表す．また，提案手法において観測確率を目標タスクの観測確率とした場合の実験結果も点線で表示した．目標タスクの観測確率は，7.1.3 節で述べたように，目標タスクのテスト集合にクラスラベルがあるものとして求めた値である．なお目標タスクの観測確率は，実際のゼロショット学習では求めることができないことに留意する．

図 7.5 より，提案モデルが DAP モデルと比較して正解率が 2〜3% 高いことが確認された．更に，目標タスクの観測確率の場合，それ以上に正解率が高くなることを確認した．目標タスクの観測確率による結果は提案手法の潜在的な精度を示しており，この結果から潜在的には更に高い正解率となることが示された．

実験 1(b)：既存のゼロショット学習との比較

本実験では，提案手法と DAP モデル以外の既存のゼロショット学習との比較をする．提案手法と比較する手法は，本章のベースライン手法である DAP モデルの他，同じく Lampert らによる IAP モデル [Lampert 09, Lampert 14]，Akata らの ALE, HLE, AHLE モデル [Akata 13]，Fu らの Semantic Graph [Fu 14c]，Fu らの TMV-BLP [Fu 14a] と比較した．

データ集合はすべての手法で AwA とし，元タスク・目標タスクのクラスは，それぞれデフォルトで分けられている訓練集合とテスト集合のクラスラベルの定義域とし，AwA のすべての訓練集合とテスト集合を利用する．これらの設定はすべての手法で共通である．本研究の提案モデルでは，元タスクの検証用集合は訓練集合のうち各クラスにつき 30 枚とし，$C = 1.0$ の L2 ロジスティック回帰を分類器とした．その他のパラメータは $\alpha_0 = \alpha_1 = 2$, $\eta_0 = \eta_1 = 0.8$ とした．提案モデルでは画像特徴量の種類として DeCAF を使っているが, その他の手法では Semantic Graph が同様に DeCAF を用いている以外は HSV color histograms, SIFT [Lowe 04a], rgSIFT [Van De Sande 10], PHOG [Bosch 07], SURF [Bay 08], local self-similarity histograms [Shechtman 07] の 6 種類の画像特徴量の種類を使っている．その他の違いとして，AHLE は属性の他に WordNet の知識を補助情報とし，Semantic Graph は属性の代わりに各クラスの Wikipedia から skip-gram モデルで抽出した言語ベクトルを補助情報としている．さらに TMV-BLP については補助情報として属性と Wikipedia から skip-gram モデルで抽出した言語ベクトルを併用している．このように，そもそも利用した補助情報が異なる上，入力とする画像特徴量の種類も異なるので，正解率のみから手法自体の比較を行うことは困難であることに留意されたい．

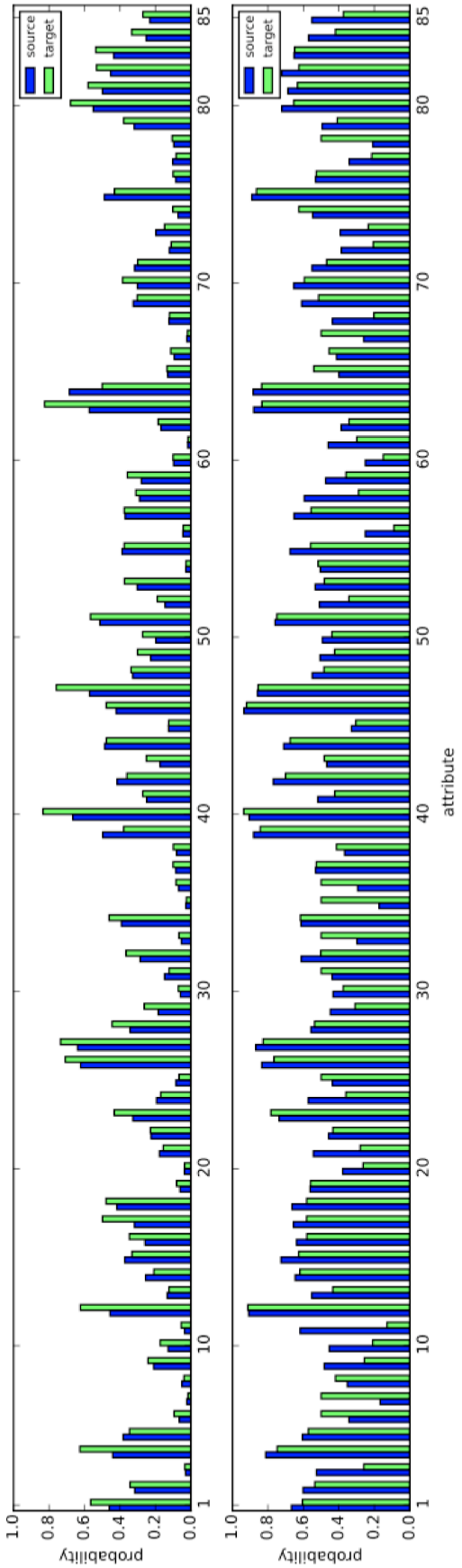


図 7.4　元タスクの観測確率と目標タスクの観測確率の比較

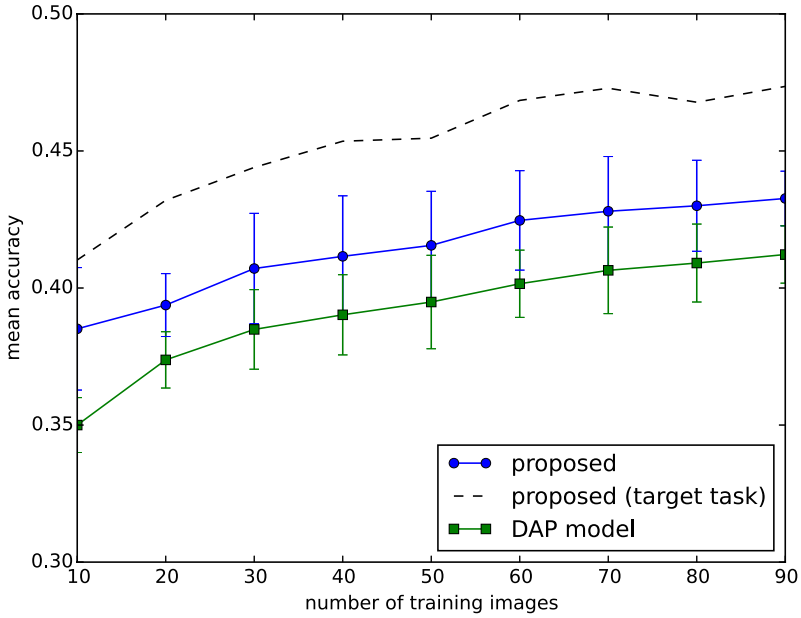


図 7.5　提案手法と DAP モデルの比較.

表 7.5 ゼロショット学習の既存研究との比較.

手法	補助情報	クラス平均正解率（%）
提案手法	属性	46.5（51.1，using the observation probabilities on the target task）
DAP	属性	40.5（[Lampert 09]）／41.4（[Lampert 14]）／46.2（our implementation）
IAP	属性	27.8（[Lampert 09]）／ 42.2（[Lampert 14]）
ALE／HLE／AHLE [Akata 13]	属性／ WordNet ／属性と WordNet	37.4／39.0／43.5
Semantic Graph [Fu 14c]	言語ベクトル（Wikipedia）	43.1
TMV-BLP [Fu 14a]	属性と言語ベクトル（Wikipedia）	**47.1**

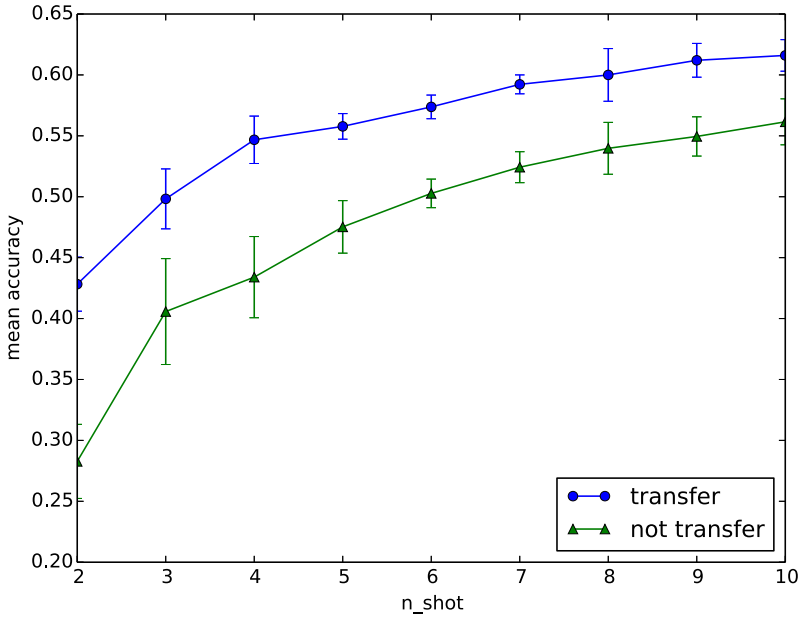


図 7.6　n ショット学習（転移なしと転移ありの比較）.

表 7.5 がクラス平均正解率で比較した結果である．既存手法の正解率はすべてそれらの論文に書かれている結果を引用し，DAP モデルは本章で利用した DeCAF での実験結果も併せて載せた．また，それぞれの手法で使われている補助情報も記載した．表 7.5 から，これらの手法の中では，TMV-BLP モデルが最も高い正解率であることが確認できる．本章の提案手法は，TMV-BLP には劣るが，同じ DeCAF 特徴量を用いている Semantic Graph より高い結果となっている．ただし上記のとおり，表 7.5 の下 3 つの既存手法では，補助情報として属性以外の WordNet や Wikipedia の知識を利用もしくは併用している．このことから，提案手法は補助情報について前処理や追加を一切しなくても，他のゼロショット学習の手法と同等以上の精度となることを示した．また，提案モデルにおいて，目標タスクの観測確率を用いた場合の正解率は 51.1% となり，潜在的には提案モデルは他の手法を上回る精度となることが確認できた.

実験 1(c)：n ショット学習での実験

本実験では，ゼロショットでない問題設定でも提案モデルが有効であるであることを示す．属性ベースゼロショット学習の既存手法である DAP モデルでは，元タスクの分類器を目標タスクで再利用することでゼロショット学習を実現していた．よって，元タスクとは別に分類器を用意して目標タスクの訓練集合を学習することはできなかった．これは，DAP モデルがゼロショット学習の場合しか対応していないことを意味する[*5]．一方，提案モデルでは分類器だけではなく，元タスクで学習したパラメータである観測確率も目標タスクで再利用している．よって，例えそれぞれのタスクで分類器を別々に学習しても，観測確率は再利用できるので，元タスクの知識を目標タスクに移すことができる．即ち，ゼロショット学習だけではなく，目標タスクで少数の訓練集合（たとえば，各クラスについて 2 枚) を用意して学習した場合でも適用できる．本章ではこれを n ショット学習 (n は目標タスクの各クラスごとの訓練事例数) と呼ぶ．

本節では，観測確率によって元タスクの知識を目標タスクに移すことを「転移」と呼ぶ．実験では n ショット学習において，転移しない場合と，提案手法で転移した場合で比較する．データ集合は AwA とし，元タスクと目標タスクのクラスは，それぞれデフォルトで分けられている訓練集合とテスト集合のクラスラベルの定義域とする．転移しない場合は，DAP モデルで目標タスクのみで訓練・推定をし，元タスクの訓練事例数は各クラスについて n 枚，テスト事例数は 80 枚として，10 クラス分類を推定する．転移する場合は提案モデルを用い，目標タスクについては同様の設定とし，元タスクの訓練事例数は各クラスあたり 90 枚，検証用事例数はそのうち 10% とした．転移しない場合と転移した場合の両方で，画像特徴量はこれまでと同様 DeCAF とし，分類器は $C = 1.0$ の L2 ロジスティック回帰とする．転移した場合のパラメータ設定は，$\alpha_0 = \alpha_1 = 2$，$\eta_0 = \eta_1 = 0.2$ とした．評価はクラス平均正解率とし，n を 2 から 10 までの 9 通りの値として，それぞれ 5 回実験した平均とする．$n = 1$ の場合は，目標タスクの訓練事例が各クラス 1 枚しかないことになり，転移しない場合において厳しすぎる設定のため除外した．

実験結果が図 7.6 である．横軸が目標タスクの各クラスあたりの訓練事例数，縦軸がクラス

[*5] 逐次学習可能な分類器を利用すれば，ゼロショット以外でも対応可能だが，任意の分類器が使えるという利点は失われる．

平均正解率である．また，緑線が転移しない場合，青線が転移した場合を表し，各グラフのエラーバーは標準偏差を表す．提案モデルによって転移した結果の方が高い正解率となっている．特に $n = 2$ のとき，転移しない場合と比べて約 15% 正解率が向上していることが確認できた．このように提案手法が DAP モデルができなかった n ショット学習でも有効であることを示した．

7.4.2　実験 2：aPascal-aYahoo

本実験ではデータ集合に aP-aY を用いる．aP-aY は AwA とは異なり，属性が各事例に対して定義されている．よって，本来の属性の定義どおり事例ごとに考えることができるが，AwA のようにクラスごとの定義も考えることができる．これは同じクラスラベルである全事例の属性値を平均して，0 より大きければそのクラスラベルを持つすべての事例の属性値を 1 とすることで求まる．よって，このデータ集合での実験では，事例ごとの定義の他に，クラスごとの定義の属性値ラベルを考えることができ，前者の実験設定を Per-Image，後者を Per-Class としてそれぞれ実験する．

実験はこれまでの実験と同様，DAP モデルと提案モデルを比較する．画像特徴量の種類はテクスチャやカラー，エッジ，HOG 特徴量による visual word などから作成した 9751 次元の画像特徴量を利用した．分類器は画像特徴量の種類に合わせて χ^2 カーネルの SVM を利用し，確信度は Platt scaling [Platt 99] によって求めた．パラメータ C は 1.0 としパラメータ γ は元タスクの訓練集合の χ^2 距離の逆数とした．また，元タスク・目標タスクのクラスラベルの定義域は，それぞれ Pascal の 20 クラス，Yahoo の 12 クラスとした．訓練事例数は各クラスについて $\{10, 20, 30, 40, 50\}$ と変化させ，目標タスクのテスト事例数は各クラスごとに 50 枚とした．以上の設定は比較する両モデルで共通である．提案モデルについては，検証用事例数を訓練事例数の 10% とし，パラメータ設定は，$\alpha_0 = \alpha_1 = 2$，$\eta_0 = \eta_1 = 0.2$ とした．評価はクラス平均正解率で，5 回実行した平均をとる．

実験 2(a)：Per-Class

Per-Class での実験結果が図 7.7 である．横軸は元タスクにおける各クラスごとの訓練事例数で縦軸がクラス平均正解率である．青線が提案手法，緑線が DAP モデルを表し，各グラフ

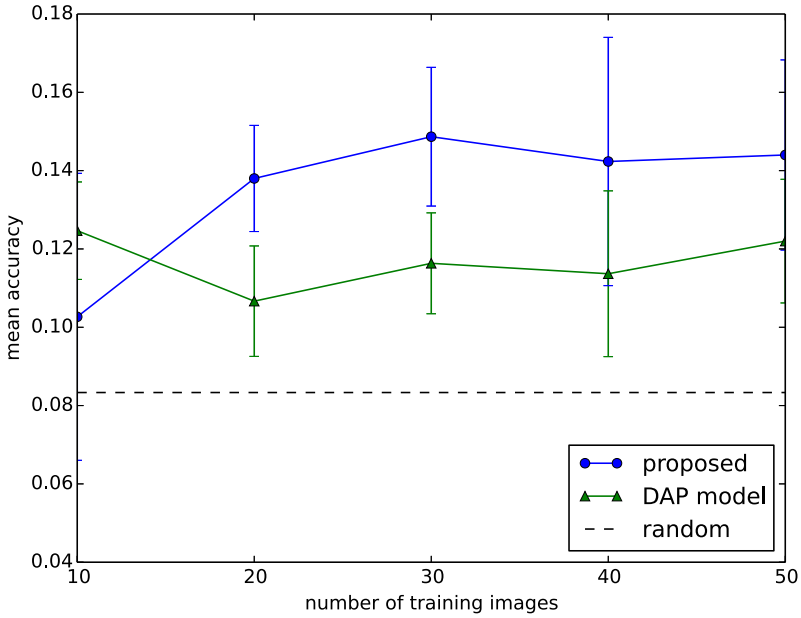


図 7.7　提案手法と DAP モデルの比較（Per-Class）.

のエラーバーは標準偏差を表す．また，点線はランダムにクラス分類した際の正解率を表す．提案手法によって概ね正解率が向上したことが確認できた．しかし，訓練事例数が各クラス 10 枚の際に提案手法が DAP モデルよりも悪くなっている．この理由の 1 つとして，元タスクにおける訓練事例数が少ないことで，モデルのパラメータである観測確率の推定が上手くいかなかった可能性が考えられる．

実験 2(b)：Per-Image

Lampert らによると，Per-Image は画像ごとのクラスを考慮せずに属性を定義していること，そして事例ごとの属性値のラベルは適切に学習できない，という理由によって Per-Class よりもうまく推定できないとされている [Lampert 14]．Per-Image の実験結果は図 7.8 である．グラフについての説明は図 7.7 と同様である．この結果から，DAP モデルは Per-Class の結果と比較して明らかに適切に分類できておらず，ランダムな分類の正解率とほぼ等しくなっていることがわかる．一方提案手法は，DAP モデルよりも正解率が高く，訓練事例数を増やすことによって徐々に正解率が高くなっていることがわかる．ただし図 7.8 で，元タス

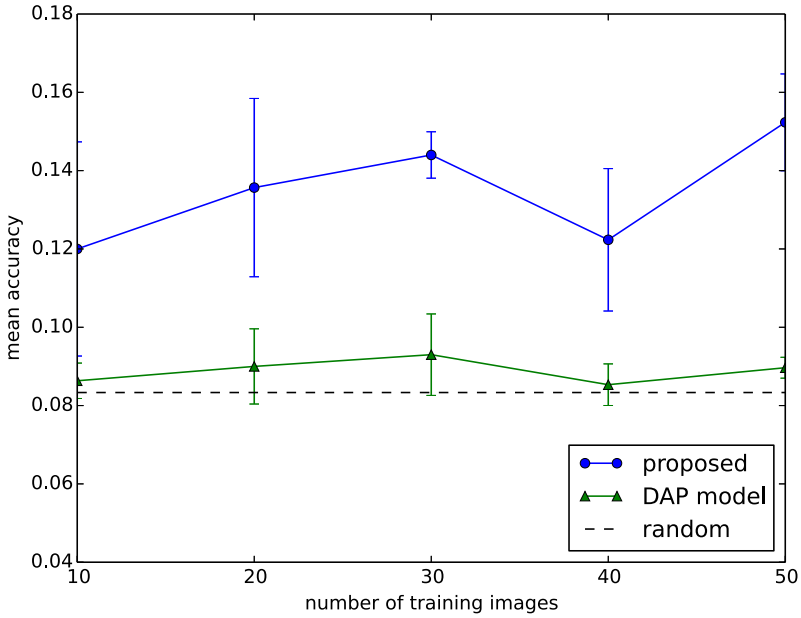


図 7.8　DAP モデルと提案手法の比較（Per-Image）.

クにおける訓練事例数が 40 枚のとき正解率が下がっていることが確認できる．この現象は DAP モデルの結果でも同様にみられることから，提案モデルの問題ではなく，訓練集合もしくは分類器の性質によるものと考えられる．今度は，Per-Class における提案手法の結果と比較すると，同等もしくはそれ以上の正解率となっていることがわかる．このことから，提案手法では DAP モデルの Per-Image では適切に学習できないという問題点を解消し，どちらの問題設定でも同様の性能を得ることができることを示した．この理由としては，提案モデルは上手く学習できなかった分類器を観測確率が考慮し，クラス推定への影響を押さえることができたということが考えられる．

7.4.3　観測確率の近さに関する考察

実験結果によって，提案手法が DAP モデルと比較して正解率が向上することが確認できたが，目標タスクの観測確率で求めた正解率よりも低くなることも明らかになった．本章では，目標タスクの属性の周辺分布をサンプリングによって事前知識として考慮することで，目標タ

表 7.6　目標タスクの観測確率にノイズを加えた際のクラス平均正解率.

手法	各クラスの訓練事例数				
	10	30	50	70	90
目標タスクの観測確率	.411	.445	.454	.473	.475
$\sigma^2 = 0.05$	.388	.445	.452	.458	.466
$\sigma^2 = 0.1$	.335	.409	.430	.441	.450
$\sigma^2 = 0.15$	.335	.368	.394	.411	.426
DAP モデル	.354	.387	.401	.412	.417

表 7.7　目標タスクの観測確率にノイズを加えた際の χ^2 値.

観測確率	ノイズ	各クラスの訓練事例数				
		10	30	50	70	90
θ_{m0}	$\sigma^2 = 0.05$	1.61	2.00	2.41	2.57	2.71
	$\sigma^2 = 0.1$	4.00	5.00	5.73	6.06	6.35
	$\sigma^2 = 0.15$	7.52	9.63	10.85	11.47	11.97
θ_{m1}	$\sigma^2 = 0.05$	3.73	3.88	4.33	4.95	4.75
	$\sigma^2 = 0.1$	5.01	5.27	5.70	6.45	6.16
	$\sigma^2 = 0.15$	6.92	7.37	7.75	8.62	8.23

スクの観測確率の結果に近づけようとした．検証 3 の結果から，この提案手法によって，観測確率が近づき正解率が向上することを確認した．しかし，実験 1(a) と実験 1(b) の結果から，目標タスクの観測確率の結果との差は開いていることが確認できる．ここでは，目標タスクの観測確率にガウス分布によるノイズを与えることで意図的に分布が遠くなるようにし，観測確率の近さと正解率の高さにどこまで相関があるのかを検証する．分布の距離とクラス平均正解率で評価する．

ガウス分布の分散を変更した時のクラス平均正解率の変化が表 7.6 であり，χ^2 値の変化が表 7.7 である．

表 7.6 から，分散を大きくすると正解率が低くなっていることがわかる．また表 7.7 から，分散を大きくすると χ^2 値が大きくなることが確認できる．一方実験 1(a) などではサンプル割

合 $\boldsymbol{\eta}$ を $\eta_0 = \eta_1 = 0.2$ としており，このときの χ^2 値は表 7.4 から確認できる．しかし，この値は表 7.7 での分散 0.05 のときの χ^2 値よりも小さい値である．更に表 7.6 より，分散 0.05 のときの正解率は目標タスクの観測確率のときとほぼ同じであることから，χ^2 値から判断すれば，提案手法でも目標タスクの観測確率の正解率とほぼ同じになるはずである．しかし，実験 1(a) でも示したように，実際には目標タスクの観測確率の方が高い正解率となっている．したがって，目標タスクの観測確率の場合と比べて提案手法の正解率が低いことの原因は，χ^2 値でわかるような観測確率全体を分布とした場合の違いだけではないことが示された．本章では，大まかに近さを測る手段として χ^2 値を採用したが，他の評価方法で検証する，または観測確率の差が大きくなると結果に影響されるような属性を見つける，といった工夫が必要と考えられる．

7.5　結論

本章では，属性ベース転移学習において，これまで着目されていなかった各属性の画像特徴量への現れやすさに着目した．そしてこの度合いを観測確率として考慮した新しい属性ベースゼロショット学習のモデルを提案した．モデルを提案するにあたり，いくつかの仮定をしたが，それぞれについて検証を行い，概ね妥当であることを示した．また異なるタスクでの観測確率を近づけるために，目標タスクでの属性の周辺分布をサンプリングによって観測確率の事前知識とすることで，タスク間の観測確率をより近づけることが可能であることを示した．

実験では，提案モデルが DAP モデルよりもよい正解率となることを示した．また，既存のゼロショット学習とも比較し，これらの研究が別の知識を補助情報として精度を上げようとする中，本章の提案手法はデータに関する前処理を一切せずに同等以上の結果になることを確認した．更に DAP モデルではできなかった n ショット学習を行い，元タスクからの転移によって，正解率が向上することを確認した．また，属性が事例ごとに定義されている場合に DAP モデルが上手く推定できないことが知られていたが，提案モデルによってこの問題が解消されることが確認できた．このような結果から，本章の提案手法は属性ベースゼロショット学習のモデルとして有効であることが示された．

一方で，観測確率を目標タスクの観測確率に近づけても，本研究の結果では目標タスクの観測確率による理想的な正解率に到達できなかったことが確認された．また観測確率に関する考

察から，この原因は，観測確率全体の分布の違いだけではないことが示された.

第 8 章

Tars：深層生成モデルの実装のためのライブラリ

8.1　確率モデリング言語と深層生成モデル

ここまでの章では，マルチモーダルデータのための深層学習と生成モデルを用いたモデルを提案し，それぞれの問題設定において有効性を確認した．しかし，これらを実際に運用するためには，なるべく簡潔かつ汎用性のある実装であることが望ましい．特に深層生成モデルでは，それぞれの確率分布が深層ニューラルネットワークで表されており，それらの分布が生成モデルや推論モデルを構成している．そのため，愚直に実装すると，とても冗長なコードになってしまい，可読性が下がりデバッグも困難になる．また，マルチモーダルデータを入力に取るので，データ集合に応じてモダリティは変化し，必要なニューラルネットワークや分布も変化する．そのため，データ集合が変化したときに，既存のコードからの変更がなるべく少なく済むようにしたい．

これらの理由から，深層生成モデルの実装は 1 から実装するのではなく，何らかのライブラリを用いることが望ましい．深層ニューラルネットワークのライブラリは，Theano [Theano Development Team 16] や Tensorflow [Abadi 15]，PyTorch[*1]，Keras [Chollet 15] などがあり，通常深層ニューラルネットワークを実装するときは，これらのいずれかを使う．特に Keras などの抽象度の高いライブラリは，簡単にニューラルネット

[*1] https://github.com/pytorch/pytorch

ワークが記述できるようになっている．しかしこの枠組みは，深層生成モデルの実装には適していないことが指摘されている[*2]．

3.3 節では，確率モデリングのためのライブラリとして，Stan，PyMC3，Edward を挙げた．特に PyMC3 と Edward は Theano と Tensorflow をそれぞれバックエンドとして利用していて，深層ニューラルネットワークを混ぜて記述することが可能である．しかし，深層生成モデルを実装する上では，必ずしも最適とは言えないことも指摘した．

近年の深層生成モデルを実装する上で，本研究では次の 2 点に着目する．

- 尤度の計算：深層生成モデルの中でも，VAE は深層ニューラルネットワークによって記述された確率分布が多層になったり，複雑な形をとることが多い [Sønderby 16, Maaløe 16]．また，本論文で提案している深層生成モデルは，モダリティごとに確率分布を設定している．そのため変分下界を計算するためにも，各分布の簡潔な尤度計算の方式が望まれる．
- 確率分布からのサンプリング：近年の深層生成モデルは，分布からサンプリング，すなわち，推論や生成ができることで注目を集めている，たとえば，5 章で提案した JMVAE は，与えられた顔画像の属性を変化させて再構成することができる．このようなシステムを実装したいとき，学習した生成モデルの任意の確率分布を直接システムで読み込んでサンプリングができると便利である．さらに変分下界を計算するためにもサンプリングが必要であるが，勾配が伝わるようなサンプリング方法でなければならない．

これらが深層生成モデルにおいて重要であることを例を用いて説明する．図 8.1 は，深層生成モデルとその変分下界の式を示した例である．深層生成モデルでは，それぞれの分布がニューラルネットワークで構成されているのは前述のとおりだが，この例ではその分布が全部で 5 つある．深層生成モデルの目的関数は変分下界であるが，その式には図 8.1 で示してあるように，サンプリングする分布と尤度計算する分布がある．したがって，変分下界を計算する

[*2] Edward の作者の Tran は Twitter 上で次のような発言をしている（@dustinvtran，2017/04/08）．“DeepMind’s Sonnet is cool. Though it’s weird people still try to shoehorn unsupervised/generative models into these neural net frameworks. ”(https://twitter.com/dustinvtran/status/850393525644644352), “It’s clear from Keras that GANs and probabilistic generative models beyond vanilla autoencoders just don(t fit this paradigm”(https://twitter.com/dustinvtran/status/850394381794373634).

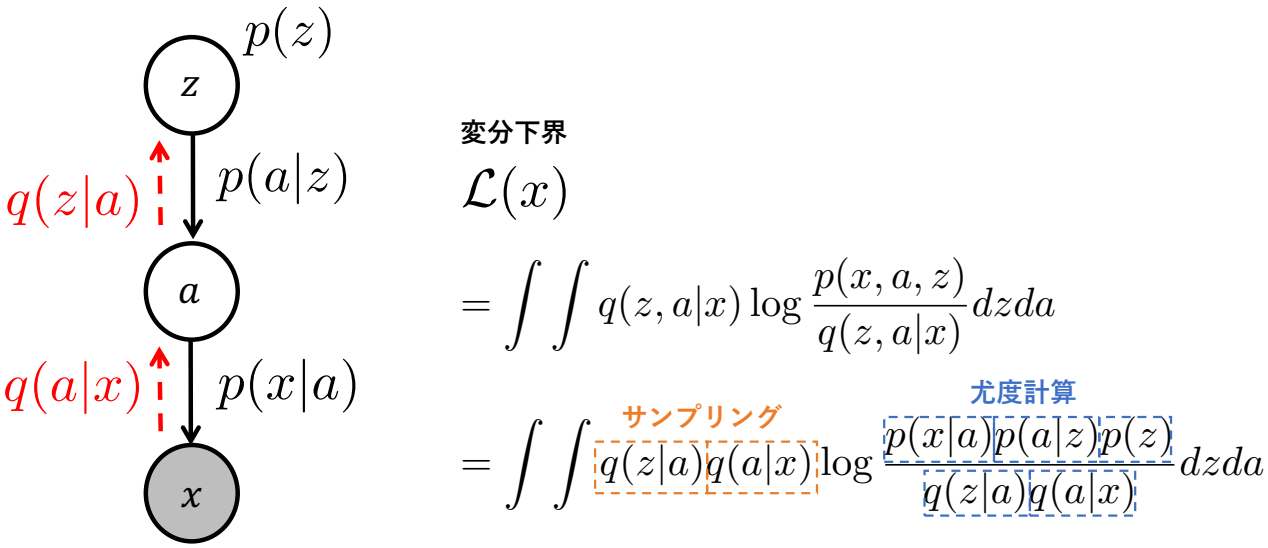


図 8.1　深層生成モデルとその変分下界の例.

ためには，すべての分布について，サンプリングと尤度計算ができる必要がある．さらにそれらは微分可能でなければならないし，学習後もサンプリング（生成や推論）ができる必要がある．PyMC3 と Edward は汎用性の高いライブラリであるが，分布ごとのサンプリングや尤度計算ができないため，このような深層生成モデルを実装できない．

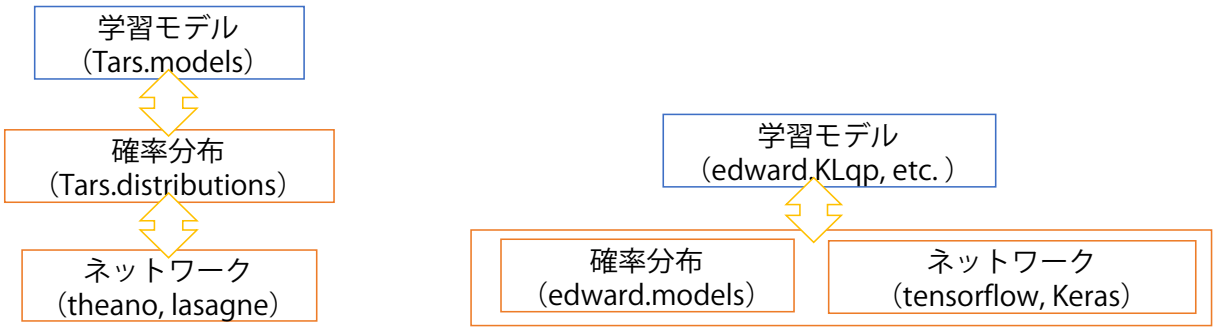
以上のことから，本研究ではライブラリとしての簡潔さと汎用性を満たしつつ，上記の要件を満たした深層生成モデル特化型ライブラリの開発を目指す．

8.2　提案手法

8.2.1　概要

本研究では深層生成モデルライブラリ Tars を提案する．このライブラリは Github 上で公開されており（https://github.com/masa-su/Tars），本章で説明する機能を実際に確認することができる．

Tars は深層生成モデル，特に様々な学習による変分推論（learned variational inference）の計算に特化したライブラリである．バックエンドには，Theano 及び Lasagne [Dieleman 15] を用いている．Lasagne は Theano のラッパーで，Keras のように簡単にネットワークが記述



(a) Tars　　(b) 既存の確率モデリング言語（例：Edward）

図 8.2　Tars と既存の確率モデリング言語.

できつつ，非常に軽い実装となっている．Tars では，VAE 系の他，GAN 系や自己回帰モデル系といった数多くの深層生成モデルを実装することができる．

図 8.2(a) は Tars の全体像である．Tars は**ネットワーク**，**確率分布**，**学習モデル**の 3 つのレイヤーで構成されている．ネットワークは Lasagne で記述し，確率分布は `Distribution` クラスを継承した各確率分布クラス（Gaussian，Bernoulli など），学習モデルは `Model` クラスを継承した各モデルクラス（VAE，GAN など）で実装する．

PyMC3 や Edward といった既存のライブラリとの最も大きな違いは，**確率分布がニューラルネットワークを隠蔽している**ことである．図 8.2(b) は Edward の構成を表している．図 8.2(a) と比べると，学習モデルの下に，確率分布とニューラルネットワークが同じレイヤーとして位置付けられている．これは，確率分布とニューラルネットワークを対等な関係として混ぜて記述できることを示している．しかし前述したとおり，この方法では確率分布ごとに尤度計算やサンプリングを計算することが困難となる．

Tars では，確率分布のパラメータをニューラルネットワークで定義する形となっているものの，その確率分布をニューラルネットワークの入力にするような混ぜた書き方はできない．その代わり，確率分布の尤度計算や確率分布からのサンプリングは，確率分布クラスの関数として呼び出すことができる．このため，確率分布クラスに隠蔽されているニューラルネットワークの実装がどのような形になっていても，尤度計算やサンプリングを実行することができる．また，すべての確率分布の API は統一されているため，分布を変更しても同様の方法で

サンプリングや尤度を計算することができる．

計算した尤度は，Theano で書かれているため自動微分が可能である．したがって，確率分布クラスでサンプリングや尤度計算することで，微分可能な下界を求めることができる．サンプリングや尤度計算によって下界を計算する部分は，学習モデルに書かれていて，学習モデルの引数に確率分布を与える形で初期化し，学習することができる．

学習後，生成モデルを構成するそれぞれの確率分布からサンプリングをすることで，推論や生成が可能となる．後述するように，確率分布クラスの中で Theano で書かれたサンプリングがコンパイルされているので，Numpy[*3]形式の入力からサンプルを抽出することができる．

以上が Tars 全体の概要である．次に，図 8.2(a) における各レイヤーの詳しい説明をする．

8.2.2　ネットワーク

前述のとおり，深層ニューラルネットワークは Lasagne を使って実装する．その他 Tars では，セル単位の LSTM [Hochreiter 97] や GRU [Chung 14]，畳み込み LSTM [Xingjian 15]，PixelCNN [Oord 16] など Lasagne にはないネットワークも実装されていて，`Tars.layers` 以下で呼び出して利用することができる．

8.2.3　確率分布

確率分布は，Tars で最も特徴的な部分といえる．確率分布には，`DistributionSample` と `Distribution` という 2 つの抽象クラスがある．

`DistributionSample` クラス

`DistributionSample` クラスは，ガウス分布やベルヌーイ分布といった様々な分布が実装されている．`sample` メソッドは，パラメータを引数にとって，選択した確率分布からのサンプリングの結果を返す．`log_likelihood` メソッドは，確率分布のパラメータ $\boldsymbol{\lambda}$ と確率変数の真の実現値 $\boldsymbol{x}$ を引数にとり，対数尤度 $\log p(\boldsymbol{x}|\boldsymbol{\lambda})$ を返す．これらのメソッドの引数と戻り値はすべて Theano のグラフ形式である．

表 8.1 は `DistributionSample` クラスを継承して実装されている確率分布である．最大の

[*3] http://www.numpy.org

表 8.1 `DistributionSample` クラスを継承して実装されている確率分布.

確率分布	再パラメータ化トリックによるサンプリング	対数尤度
退化分布	-	×
ガウス分布	○	○
標準ガウス分布	○	○
ラプラス分布	○	○
ガンベル分布	○	○
ベルヌーイ分布	○（Gumbel-softmax）	○
カテゴリ分布	○（Gumbel-softmax）	○
ガンマ分布	○（rejection sampling）	○
ディリクレ分布	○（rejection sampling）	○
ベータ分布	○（rejection sampling）	○
クマラスワミー分布	○	○

特徴は，サンプリングがすべて再パラメータ化トリック [Kingma 13, Rezende 14] によって実装されていることである．特に，離散サンプリングには Gumbel-softmax [Jang 16]，ガンマ分布は棄却サンプリング [Naesseth 16] を利用している．したがって，いずれの分布でも，サンプリングの勾配を計算することができ，推論モデルとして用いることができる．またすべて尤度計算ができるので，生成モデルの分布としても用いることができる．

`Distribution` クラス

`Distribution` クラスは，Lasagne で書かれたネットワークを格納するように設計されている．ネットワークは，各分布のパラメータ数に応じて用意し，各分布クラスの初期化時に設定する．たとえば，ガウス分布の場合は，平均と分散がパラメータなので，それに対応するネットワークを用意して設定する．また初期化の段階で，条件付ける変数を `given` という引数で設定する．たとえば，$p(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{y})$ という確率分布を設定するとき，条件付ける変数は $\boldsymbol{z},\boldsymbol{y}$ なので，これらを `given` に設定する．なお，Tars では `lasagne.layers.InputLayer` を確率変数と解釈する．これによって，任意の変数で条件けられた分布を記述することができる．ソースコード 8.1 は，Tars での確率分布クラスの設定例である．

ソースコード 8.1　ガウス分布（Tars.distributions.Gauss）$p(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{y})$ の設定例

```
# 確率変数
z = InputLayer((None,10))
y = InputLayer((None,10))

# 確率分布のパラメータ（Lasagne で記述）
p_0 = DenseLayer(ConcatLayer([z,y]), num_units=512, nonlinearity=rectify)
p_1 = DenseLayer(p_0, num_units=512, nonlinearity=rectify)
p_mean = DenseLayer(p_1, num_units=784, nonlinearity=linear)
p_var = DenseLayer(p_1, num_units=784, nonlinearity=softplus)

# 確率分布の定義
p = Gauss(p_mean, p_var, given=[z,y])
```

一度確率分布を初期化すると，以降はニューラルネットワークの構造を気にせずにサンプリングや尤度計算ができる．Distribution クラスは，同じ確率分布の名前の DistributionSample クラスをインスタンスとして持っていて，それを利用して尤度計算やサンプリングができる，たとえば，Gauss クラスは GaussSample クラスをインスタンスとして持っている．

Distribution クラスでのサンプリング

サンプリングのために，3 種類のメソッドが用意されている．それぞれ，fprop，sample_mean_given_x，sample_given_x である．それぞれのメソッドの引数は，先に指定した given の型に対応していて，Theano のグラフを渡すようになっている．fprop は，ニューラルネットワークで順伝播した出力をそのまま返す．ガウス分布の場合，平均と分散のそれぞれのネットワークの出力を返す．これは，尤度計算のときなどに有用である．sample_mean_given_x はサンプルの平均を返す．ガウス分布 $\mathcal{N}(\boldsymbol{x};\mu,\sigma)$ の場合，平均値 μ のみが返される．決定論的に画像を生成したいときに用いることができる．sample_given_x はサンプリングした結果が出力される．即ち，引数にニューラルネットワークを渡した出力を DistributionSample の sample メソッドの引数とした出力である．これらの出力はすべて自動微分可能である．

深層生成モデルでは，確率分布を学習したあとに，サンプリング，即ち生成や推論をしたいことが多い．このとき，確率分布の入力として Theano のシンボルではなく，Numpy 形式の画像やノイズを取ることができると便利である．Theano では Numpy 形式を入力として取る

ためには，theano.function というメソッドで入力のシンボルとグラフを指定してコンパイルする必要があるが，確率分布が増えると，すべての分布についてこれを実行するのが厄介となる．そこで Tars の確率分布クラスでは，引数として Numpy 形式が取れるようなメソッドを用意している．それが np_fprop，np_sample_mean_given_x，np_sample_given_x である．機能は上記の 3 種類のメソッドと同様だが，引数として Numpy 形式をとることができる．これにより，学習した確率分布クラスをそのままアプリケーション等に利用することができる．

ソースコード 8.2 は，これらのサンプリング用のメソッドの具体例である．

ソースコード 8.2 確率分布クラスのサンプリングメソッドの例

```
13 # Theano シンボルでサンプリング・順伝播
14 mean_sample, var_sample = q.fprop(x) # 順伝播
15 samples = q.sample_given_x(x) # ランダムサンプル
16 samples = q.sample_mean_given_x(x) # 平均サンプル
17
18 # Numpy 形式でサンプリング・順伝播
19 mean_sample, var_sample = q.np_fprop(np_x) # 順伝播
20 samples = q.np_sample_given_x(np_x) # ランダムサンプル
21 samples = q.np_sample_mean_given_x(np_x) # 平均サンプル
```

Distribution クラスでの尤度計算

Distribution での尤度計算は，log_likelihood_given_x で行う．2 つの確率変数 $\boldsymbol{x}, \boldsymbol{z}$ をとり，対数尤度の計算 $\log p(\boldsymbol{z}|\boldsymbol{x})$ をする．分布クラスが持つネットワークを MLP とすると，対数尤度の計算は $\log p(\boldsymbol{z}|\boldsymbol{x}) = \log p(\boldsymbol{z}|\boldsymbol{\lambda} = \mathrm{MLP}(\boldsymbol{x}))$ となる．即ち，条件づける変数 $\boldsymbol{x}$ を入力として順伝播し，出力 $\boldsymbol{\lambda}$ を得て，それをパラメータとして尤度計算をする．$\log p(\boldsymbol{z}|\boldsymbol{\lambda})$ の計算のために，インスタンスとして持っている DistributionSample の log_likelihood メソッドを利用している．サンプリングと同様に，Numpy 形式を入力に取れるように np_log_likelihood_given_x メソッドも用意されている．

ソースコード 8.3 は，尤度計算用メソッドの具体例である．

ソースコード 8.3 確率分布クラスの対数尤度メソッドの例

```
22 # Theano シンボルで尤度計算
23 log_likelihood = q.log_likelihood_given_x(x, samples)
24
25 # Numpy 形式で尤度計算
26 log_likelihood = q.np_log_likelihood_given_x(np_x, np_samples)
```

複数の確率分布クラスを組み合わせたメタ確率分布クラス

ここまで，確率分布クラスが自動微分可能なサンプリングや尤度を求めることができることを見てきた．しかし図 8.1 で示したような，複雑な生成モデルを記述するためには，確率分布を組み合わせるような枠組みが必要である．

Tars では，初期化時に確率分布クラスを引数に取るような，**メタ確率分布クラス**でこれを実現している．これはちょうど確率分布の積に対応している．たとえば，図 8.1 では，生成モデルは $p(\boldsymbol{x}, \boldsymbol{a}|\boldsymbol{z}) = p(\boldsymbol{x}|\boldsymbol{a})p(\boldsymbol{a}|\boldsymbol{z})$ というように複数の分布の積で表されている．メタ確率分布クラスでは $p(\boldsymbol{x}|\boldsymbol{a})$，$p(\boldsymbol{a}|\boldsymbol{z})$ が初期化時の引数となり $p(\boldsymbol{x}, \boldsymbol{a}|\boldsymbol{z})$ がメタ確率分布クラスが表現する確率分布となる．

メタ確率分布クラスの最大の利点は，引数がどのような分布（ガウス分布，ベルヌーイ分布など）であっても，それらに関係なくサンプリングや尤度計算ができるということである．メタ確率分布クラスと確率分布クラスの関係は，確率分布クラスとネットワークの関係と同じと考えることができる．また，メタ確率分布クラスには確率分布クラスと全く同じサンプリング・尤度計算用メソッドが用意されている．これは即ち，外部からはメタ確率分布クラスを通常の確率分布クラスと同じものとして扱えることを意味する．

Tars では，どのような確率分布の積で表されるかによって，いくつかのメタ確率分布クラスを用意している．ここでは，`MultipleDistribution` クラスと `MergeDistribution` クラスについて説明する．

`MultipleDistribution` クラスは，メタ確率分布クラスが各分布の連鎖積で表されている場合，即ち $p(\boldsymbol{x}|\boldsymbol{a}_1)...p(\boldsymbol{a}_n|\boldsymbol{z}) = p(\boldsymbol{x}, \boldsymbol{a}_1, ..., \boldsymbol{a}_n|\boldsymbol{z})$ となるときに用いる．これによって作成されるメタ確率分布クラスは，`given` が $\boldsymbol{z}$，サンプリングの出力が $\boldsymbol{x}, \boldsymbol{a}_1, ..., \boldsymbol{a}_n$ となる．

`MergeDistribution` クラスは，同じ変数で条件づけられた分布の積となる分布を表現する．つまり，$p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{w}|\boldsymbol{z})... = p(\boldsymbol{x}, \boldsymbol{w}, ...|\boldsymbol{z})$ という形である．この場合，`given` が $\boldsymbol{z}$，サンプリングの出力が $\boldsymbol{x}, \boldsymbol{w}, ...$ となる．このクラスは，5 章や 6 章で提案した潜在変数から複数のモダリティの変数が生成されるモデルを実装するのに便利である．

ソースコード 8.4 は，これらの実装例である．その他にも，確率分布を組み合わせるメタ確率分布クラスが複数実装されているが，本章では以上の説明に留める．

表 8.2　Tars で実装されている学習モデルの一覧.

深層生成モデルの種類	学習モデル
VAE 系	VAE [Kingma 13, Rezende 14]，conditional VAE [Sohn 15] M2 model [Kingma 14a]（semi-supervised VAE） CMMA [Pandey 16] SDGM [Maaløe 16] JMVAE（5 章） VAE-GAN [Larsen 15a] DRAW [Gregor 15] Variational RNN [Chung 15]
GAN 系	GAN [Goodfellow 14]，conditional GAN [Mirza 14] WGAN [Arjovsky 17] pix2pix [Isola 16] cycle GAN [Zhu 17]
自己回帰系	pixel CNN [Oord 16]（ただし Tars.layers で実装）

ソースコード 8.4　メタ確率分布の実装例

```
27 # p(x|a)p(a|z)=p(x,a|z)を表現するメタ確率分布(ネットワークは省略)
28 gauss = Gaussian(mean, var, given=[z])
29 bernoulli = Bernoulli(mean, given=[a])
30 p = MultiDistributions([gauss, bernoulli])
31
32 # p(x|z)p(w|z)=p(x,w|z)を表現するメタ確率分布(ネットワークは省略)
33 gauss = Gaussian(mean, var, given=[z])
34 bernoulli = Bernoulli(mean, given=[z])
35 p = MergeDistributions([gauss, bernoulli])
```

8.2.4　学習モデル

`Model` クラス

学習モデルでは，定義された確率分布を受け取り，サンプリングや尤度計算の操作によって下界を計算し，与えられたデータから学習を行う．Tars には複数の学習モデルが実装されており，表 8.2 にその一覧を載せる．

ここで，学習モデル内の変分下界は，与えられた確率分布のサンプリングと尤度計算のみから求まるということに注意されたい．上記で述べたように，サンプリングや尤度計算の API はすべての確率分布クラス及びメタ確率分布クラスで統一されている．つまり，学習モデルに与える確率分布の種類が変わったり，条件付けされる変数が変わったり，複数の分布で構成されていたとしても，学習モデルクラスは一切変更せずに，変分下界を求めて学習できる．従来の手法は，確率分布を変更するたびに別の実装を 1 から書く必要があったことを考えると，Tars はそれらに比べて実装が簡潔になり，汎用性が高くなったといえる．

学習モデルは `Model` クラスを継承しており，API も統一されている．学習モデルは，初期化するときに引数として確率分布を与える．与える確率分布の数や役割は，学習モデルによって変わる．初期化の段階で下界の計算が行われ，学習とテスト用の Theano の関数がコンパイルされる．訓練データを与えて学習するときは，`train` メソッドを用いる．また，テストデータで検証する場合は `test` メソッドを用いる．

ソースコード 8.6 に学習モデルの 1 つ，`VAE` クラスの実装例を載せる．ネットワークと確率分布を与えれば，このように簡単に学習やテストを実行することができる．

ソースコード 8.5　VAE の実装例

```
36 # 学習モデルの初期化(q や p は確率分布クラス)
37 model = VAE(q, p, n_batch=128, optimizer=adam)
38
39 for i in range(100):
40         # 学習(train_x は訓練データ)
41         lower_bound_train = model.train([train_x])
42
43         # テスト(test_x はテストデータ)
44         lower_bound_test = model.test([test_x])
```

`Model` クラス以外の実装

深層生成モデルのアルゴリズムを開発するのではなく，深層生成モデルをアプリケーションで利用したいと考える人は，データに合わせてネットワークや分布を設定し，表 8.2 の利用したい学習モデルのクラスに渡すだけで学習することができる．しかし，深層生成モデルの研究者にとっては，新しい学習アルゴリズムやアーキテクチャを設計したいという要求がある．

Tars の学習モデルの変分下界の計算は，確率分布クラスによる抽象化によって，非常に簡潔に書かれている．ソースコードは VAE での下界の計算例である．このように，サンプリン

グや尤度計算といった面倒な計算はすべて確率分布内で行われているので，研究者にとっても新しい学習モデルを書くのが容易になっている．また，2 つの確率分布間の KL ダイバージェンスを計算する関数 `analytical_kl` など，確率分布に関する計算をサポートするような関数も用意されている．実際に新しい学習モデルを書きたい場合は，VAE クラスなどを継承して，下界やコストの計算部分のメソッドをオーバーライドする．

ソースコード 8.6 VAE の変分下界の計算例

```
45 # self.q, self.p, self.prior は初期化時に定義された確率分布
46
47 # 入力の確率変数の設定
48 x = self.q.inputs
49
50 # KL ダイバージェンスの項の計算
51 kl_divergence = analytical_kl(self.q, self.prior
52                                given=[x, None],
53                                deterministic=deterministic)
54
55 # 対数尤度の項の計算
56 z = self.q.sample_given_x(x, repeat=l,
57                            deterministic=deterministic)
58 inverse_z = self._inverse_samples(z)
59 log_likelihood =\
60         self.p.log_likelihood_given_x(inverse_z,     deterministic=deterministic)
61
62 # 誤差関数(負の変分下界)
63 loss = -T.mean(log_likelihood - annealing_beta * kl_divergence)
64
65 # 学習パラメータ
66 params = self.q.get_params() + self.p.get_params()
```

以上のように，深層生成モデルの研究者にとっても，アプリケーションのために利用したい人にとっても，Tars は実装コストを大幅に抑えられる工夫が施されている．

8.3 評価実験

本節では，Tars が各分布をモジュールとして自由に変更したりサンプリングできることを，簡単な実験で示す．実験では VAE クラスを利用し，確率分布の種類を実際に簡単に変更できること，そして学習した分布から簡単にサンプリングできることを確認する．

VAE では通常エンコーダとしてガウス分布が使われるが，ここでは分布をベルヌーイ分布，

カテゴリ分布，ガンマ分布，ディリクレ分布のそれぞれに変更して学習し，対数尤度の評価と変更した分布からのサンプリングを行う．

データ集合は MNIST を用いた．エンコーダ，デコーダの両方で，活性化関数を ReLU とした 200 次元の 2 層ニューラルネットワークを用いた．また，デコーダの確率分布はベルヌーイ分布に固定した．潜在変数の次元は 100 次元とし，カテゴリ分布，ディリクレ分布ではカテゴリの次元も持つので，カテゴリの次元を 10 次元，即ち潜在変数は 100×10 次元とした．

VAE の事前分布 $p(x)$ は，エンコーダの分布に応じて次のように設定した

- ガウス分布：$\mathcal{N}(x; 0, 1)$
- ガンマ分布：$\mathrm{Gamma}(x; 1, 1)$
- ベータ分布：$\mathrm{Beta}(x; 1, 1)$
- ベルヌーイ分布：$\mathrm{Bern}(x; 0.5)$
- カテゴリ分布：$\mathrm{Cat}(x; \boldsymbol{\pi} = [\frac{1}{10}, ..., \frac{1}{10}])$

評価は対数尤度で行い，[Burda 15] に従って importance weighted sampling を 5000 回行った値を対数尤度とした．また，MNIST データは訓練時にランダムに 2 値化して学習した．

表 8.3 は，エンコーダを様々な確率分布に変更した場合の結果である．同じ VAE クラスで実装しているが，エンコーダの分布を変更するだけで，簡単に様々なタイプの VAE を実装することができる．特に，ガンマ分布とベータ分布の場合の VAE はこれまで提案されていなかったが，このように学習できること確認できる．

次に，それぞれの分布（ガウス分布，ガンマ分布，ベータ分布）でモデル化した VAE からサンプリングをする．図 8.3 は $\boldsymbol{z} \sim p(\boldsymbol{z}), \boldsymbol{x} \sim p(\boldsymbol{x}|\boldsymbol{z})$ のようにして $\boldsymbol{x}$ をサンプリングした結果である．Tars では，確率分布クラスから直接サンプリングできるので，この確率過程のとおりにコード上で NumPy でサンプリングできる．それぞれの事前分布 $p(\boldsymbol{x})$ は上で設定したとおりである．いずれの分布でも，数字画像が適切に生成できていることがわかる．

8.4　応用例：FacialVAE

本節では，Tars を用いて学習した確率モデルをアプリケーションに応用した例を示す．

FacialVAE は，顔画像を入れると，「笑っている」，「男性」などの属性が予測され，さらに

表 8.3　様々なエンコーダの VAE の対数尤度.

エンコーダの分布	$\log p(\boldsymbol{x})$
ガウス分布	-91.02
ベルヌーイ分布	-96.13
カテゴリ分布	-98.46
ガンマ分布	-92.82
ベータ分布	-93.67

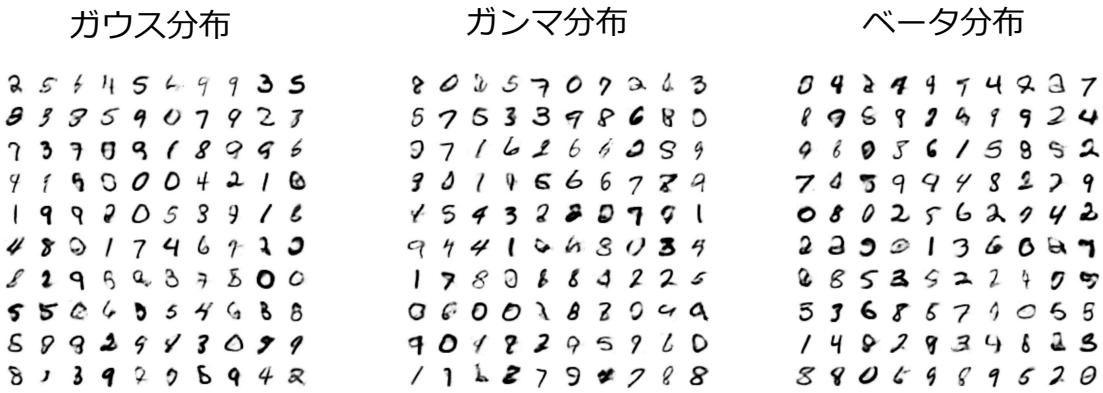


図 8.3　様々なエンコーダで学習した VAE のサンプリング結果.

その属性を変更すると，その値に従って顔画像が変化するシステムである．

このシステムは conditional VAE [Sohn 15] と M2 モデル [Kingma 14a] がベースとなっている．顔画像を $\boldsymbol{x}$，属性を $\boldsymbol{y}$，潜在変数を $\boldsymbol{z}$ とすると，$p(\boldsymbol{x}|\boldsymbol{y}) = \int p(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}$ という生成モデルを学習している．また，与えられた画像から属性が予測できるように $q(\boldsymbol{y}|\boldsymbol{x})$ という分布も別に学習している．図 8.4 は，このモデルを説明したものである．

FacialVAE では，あらかじめ Tars で学習した確率分布クラスをそれぞれ保存しておき，そのままシステム上（Flask[*4]で書かれている）で読み込んで，属性の予測や画像の生成を行っている．これは，Tars が確率分布をクラスとして書くことができること，そしてそれぞれの確率分布クラスで NumPy 形式を入力として簡単にサンプリングできることから可能になって

[*4] http://flask.pocoo.org

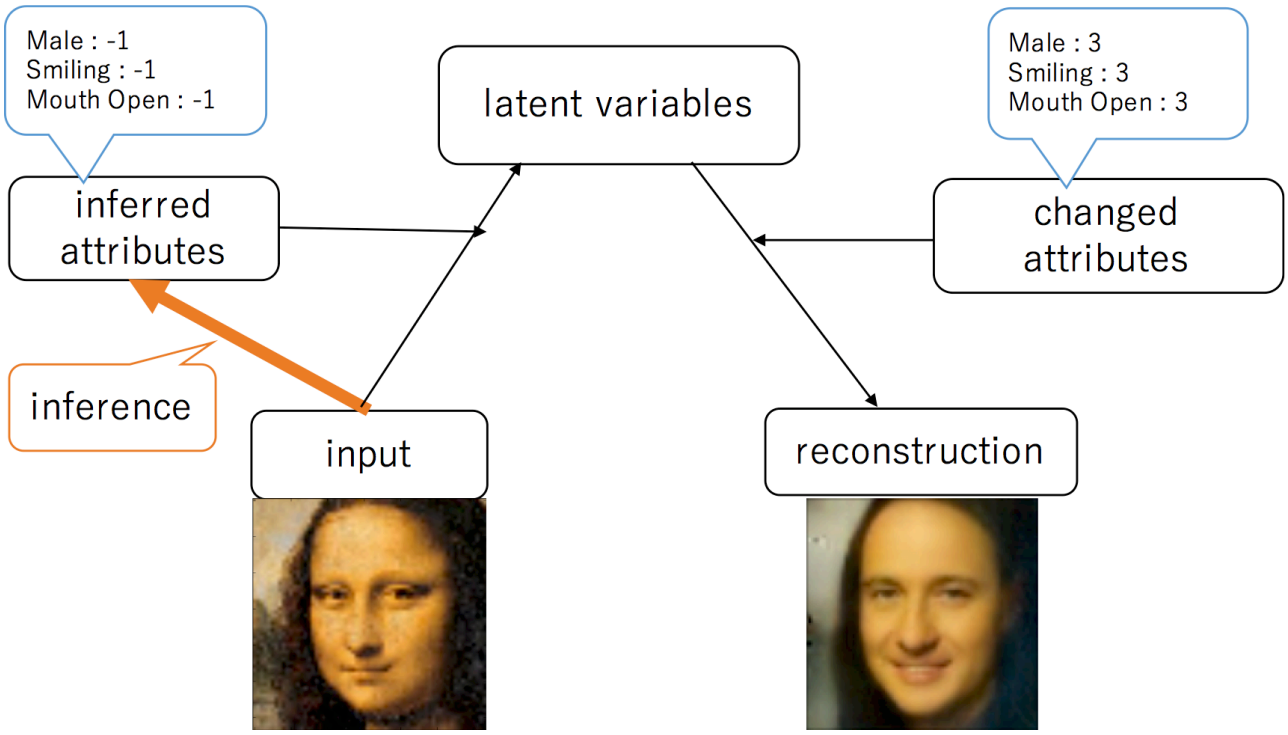


図 8.4 FacialVAE で使われている conditional VAE.

いる．

FacialVAE は Web アプリケーションとして公開されており（図 8.5）[*5]，誰でも試すことができる．また，このデモはテレビ番組でも紹介された[*6]．

8.5 結論

本章では，深層生成モデルを実装するためのライブラリ Tars を提案した．Tars では，確率分布の尤度計算とサンプリング抽出が深層生成モデルにおいて重要であることを踏まえて，確率分布がネットワークを隠蔽するような設計としている．またメタ確率分布クラスによって，多層構造などの確率分布をまとめて 1 つの分布として扱えるようになっている．これらの分布は種類によらず同じ API で尤度計算やサンプリングができるため，深層生成モデルの研究者

[*5] http://fvae.ail.tokyo

[*6] ワールドビジネスサテライト．公式サイト（http://txbiz.tv-tokyo.co.jp/wbssp/vod/post_117430）で動画が公開されている．

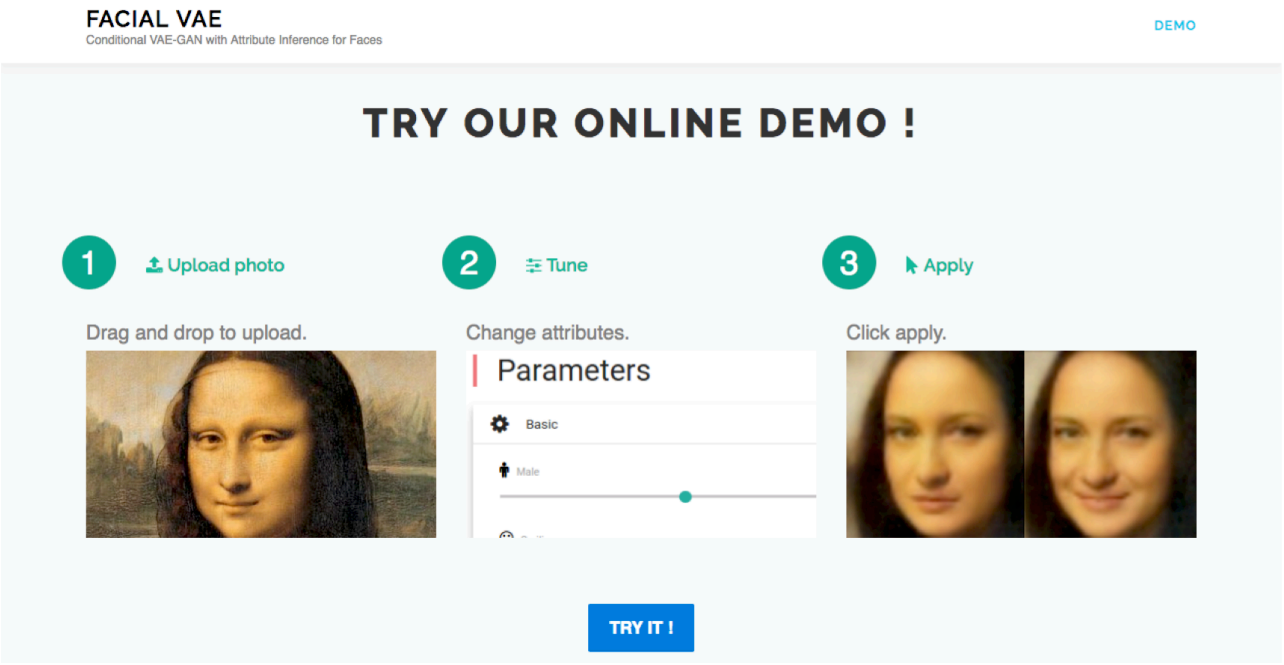


図 8.5　FacialVAE のデモページ.

にとっても，アプリケーションで利用したい人にとっても，簡潔かつ汎用性の高い実装を書くことができる．実験から，様々な分布や下界を持つ深層生成モデルを，最小限の実装の変更で学習できることを確認した．また，Tars で学習した確率分布が Web アプリケーションで使われている事例をみて，Tars がモデルの実装からアプリケーションでの活用まで幅広く使用できる可能性があることを確認した．

第 9 章

考察

本章では本研究の成果について整理した上で，本研究の貢献についてまとめる．その上で本研究の成果の限界について論じ，今後の課題について議論する．

9.1　各章の整理

5 章から 7 章では，マルチモーダル情報を扱うにあたり，深層学習と生成モデルを用いたモデルを提案し，それらが様々な問題設定において有用であることを確認した．

5 章では，モダリティを双方向変換することを目的として，共有表現のアプローチを深層生成モデルに適用した JMVAE を提案し，片方のモダリティを欠損させると共有表現や変換したモダリティが崩れてしまい，従来の欠損値補完手法では解決しないことを確認した．そして JMVAE-kl と階層的 JMVAE を提案し，欠損させても共有表現が崩れないこと，そして双方向にモダリティを変換できることを確認する．さらに，片方のモダリティを変化させることで，もう片方のモダリティの該当部分も変化することがわかった．

6 章では，深層生成モデルを用いた半教師ありマルチモーダル学習モデルとして SS-MVAE と SS-HMVAE を提案した．また半教師ありマルチモーダル学習では，テスト集合に単一のモダリティしか与えられない設定があることから，モダリティが欠損しても精度を落とさずに目標ラベルを予測する手法として，SS-HMVAE を拡張した SS-HMVAE-kl を提案した．この手法でモダリティの欠損が補完できること，そして単一モダリティ及びマルチモーダルの両方で，既存手法よりも精度の高い半教師あり学習ができることを確認した．

7 章では，共学習の 1 つである，ゼロショット学習に取り組んだ．属性ベースゼロショット学習において，属性ごとの画像特徴量への観測確率を考慮した生成モデルを提案することによって，従来よりも高い正解率で未知のタスクのクラスを予測できることを確認した．これは，単純に従来の手法の特徴量を深層ニューラルネットワークに置き換えた結果を上回る精度である．また他のモダリティを補助情報として加えている研究と比較しても同等以上の精度となったため，本研究の手法は，適切に複数モダリティを活用しているといえる．

5 章から 7 章をまとめると，マルチモーダル学習のそれぞれの問題設定において，モダリティ間の特徴空間と分布の違いを考慮する，すなわち，深層学習と生成モデルを用いたモデルを利用することで，それぞれ精度の改善や問題の解消ができることを確認した．

8 章では，マルチモーダル情報を含むような複雑な深層生成モデルを実装するためのライブラリとして Tars を提案した．Tars では，ネットワークが確率分布に隠蔽される方法を取っており，それぞれがモジュールのように別々に尤度計算したりサンプリングすることができる．簡単な実験や応用例から，Tars が簡潔かつ汎用性の高いライブラリであることを示した．

9.2　提案手法の貢献と今後の技術的課題

9.2.1　マルチモーダル情報の扱いとモデル化について

本論文では，マルチモーダル学習に取り組むにあたって，特徴空間と分布の両方の違いを考慮する必要性を議論し，実験から，マルチモーダル学習の各問題設定において，深層学習と生成モデルを用いるアプローチによって対処できることを示した．これが本論文の貢献である．しかし，それぞれの問題設定のあらゆる場合に対して，この主張が成り立つことを示した訳ではない．

まず，本論文ではモダリティを 2 つに限定しているため，3 つ以上モダリティがある場合でも同様の主張が成り立つかどうかは定かではない．現実世界の多くの問題設定で，モダリティは 3 つ以上与えられる．1 章で述べたように，ロボットは複数のセンサを持ち，様々なマルチモーダル情報が得られる．これらの情報を用いてロボットが的確に状況判断するために，すべてのモダリティを入力として取ることができるマルチモーダル学習モデルが求められる．

3.2.1 節でみたように，表現の問題設定では，共有表現と座標表現のアプローチが考えられる．共有表現の場合，3 つ以上の多くのモダリティを 1 つの層あるいは 1 つの潜在変数にすべ

て統合するのは困難な可能性がある．文献 [Pang 15] では，モダリティの種類によって統合する段階 を分ける工夫をしている．しかし，これはモダリティの種類に大きく依存するため，どのモダリティをどの段階で結合すべきか検証する必要がある．また，5 章では JMVAE-kl もしくは階層的 JMVAE による欠損補完手法を提案したが，JMVAE-kl はモダリティが増えるとネットワークの数が指数関数的に増えてしまう（5.2.4 節）．階層的 JMVAE ではネットワークが指数関数的に増えることはないが，どの程度の数のモダリティ欠損ならば共有表現が崩れないかといったことは明らかではない．

座標表現のアプローチでは，それぞれのモダリティ用のネットワークがあれば，任意の方法でネットワークの層同士の距離を近づけることで，共通する表現を学習できる．しかし 3 つ以上のネットワークがある場合は，どのネットワークを基準として近づければいいか定かではなくなる．したがって，現状でも多数のモダリティを適切に扱う研究は確立していないと考えられる．

また，本研究では画像やタグ，属性といったシンプルなモダリティ情報のみに限定していた．一方，現実の問題設定ではネット上の閲覧数や販売数，そしてロボットやウェアラブルデバイスなどのセンサから得たデータといった多くの情報が，系列データである．本研究では固定次元のモダリティのみとしていたので，系列データのモダリティの違いについては定義でも議論せず，マルチモーダル学習の 1 つであるアラインメントの問題設定については扱わなかった．現在のところ，系列情報間の分布の違いの定義は曖昧であり，そうした違いがどのくらい影響するのかもわかっていない. 最新の教師なし翻訳を行う研究でも，言語間の分布の違いや近さについては考慮されていない [Lample 17]．しかし Zhou らが指摘するように，系列間では一般的に分布は異なる [Zhou 14]．現在は，アテンションメカニズムによって，これに対処できていると考えられている．たとえば，異なる系列同士（翻訳など）や固定次元から系列（画像から文書など）の変換については，多くの場合でアテンション付き RNN が利用されている．最近では，深層生成モデルと併用して，文書から画像の生成も実現している [Reed 16]．しかし根本的には，そもそも系列の場合のモダリティの違いとは何か，特に分布の違いとは何かについて考える必要があると思われる．

異なるモダリティ間の関係についても様々な考え方があり，モデルの設計に直接影響する．本論文では，異なるモダリティはある目標（概念）$\boldsymbol{y}$ から生じるもので，その意味では異なる

モダリティ同士は概念の下では対等と考えている（特に 5 章と 6 章）．一方で Higgins らは，画像と言語は情報量の違いから非対称であり，階層的な関係，すなわち，言語の方がより抽象的な概念としてモデル化すべきと主張している [Higgins 17]．彼らは言語を画像の上位概念とし，さらに言語間で階層性があるモデルとして SCAN を提案した．実験では，本論文の 5 章にあたる JMVAE と比較しており，正解率や多様性の面で，SCAN が JMVAE よりも優れていると主張している．

さらに本論文では，それぞれの問題に対してタスクを考慮しないか，もしくは 1 つに固定している．現実には数多くのタスクがあり，人間は常に異なるタスクの知識を転移して行動している．そうしたタスクはそれぞれ異種の情報を用いているので，マルチタスクかつマルチモーダルな問題を解いていることになる．Kaiser らはマルチモーダルかつマルチタスクな複数の問題設定を同時に解くモデルとして MultiModel を提案している [Kaiser 17]．このモデルは各モダリティ（言語，画像，音声，カテゴリ）用のネットワークがあり，異なるタスクで使いまわすようになっている．また自己回帰モデルとして in と out の両方が設計され，共有表現に写像するようになっている．実験では，8 つの異なるタスクのデータ集合を MultiModel で同時に学習し，任意のタスクによって精度が下がる負の転移は起こらず，多くのタスクで精度が上がる結果が報告されている．このモデルは，人間のようにマルチモーダルかつマルチタスクな転移学習が実行できる可能性を示しているが，本論文で着目した異種間の分布の違いについては従来のマルチモーダル学習同様考慮していない．こうした問題設定にも，深層生成モデルを適用することが考えられる．

ゼロショット学習に関しては，本研究では属性ベースの手法での検証に留まった．近年は，ワンショット学習でも深層生成モデルの手法が提案されており [Rezende 16]，最近では深層生成モデルによるゼロショット学習も提案されている [Kansky 17]．これらの手法においてのマルチモーダルデータの扱いについても，本論文の知見を活かせるか検証する必要がある．

9.2.2 深層生成モデルライブラリについて

Tars は深層生成モデルに特化したライブラリである．本ライブラリは Github 上での Star 数[*1]は，少ないながら，Dustin Tran（Edward の開発者）や Thomas Wiecki（PyMC3 の開発者）など，著名な確率プログラミング言語の開発者からつけられており，一定の評価がされているといえる．

確率分布自体に尤度やサンプリングできる機能を持たせ，それらをモジュールとして接続して（Tars ではメタ確率分布クラスで隠蔽して）生成モデルを組み上げていくという Tars の理念は，アーキテクチャ的な考え方である．それぞれをモジュールと見立てて組み上げるという発想は，機械学習の様々なライブラリやプラットフォームでも実践されている．Tensorflow は学習モデルをモジュールと見立ててマルチタスク学習が行える Tensor2Tensor[*2]を開発している．また，全脳アーキテクチャ・イニシアティブ[*3]という団体では，人間の脳は複数の機械学習をモジュールとして組み合わせることで実装できるという考え方をとり，Brica[*4]というプラットフォームを開発している．

しかし Tars は未だ発展途上であり，制限の多いライブラリである．最も大きい問題は，Tars のバックエンドとして利用している Lasagne [Dieleman 15] の開発がほぼ止まっていることである．また，Lasagne がラップしている Theano [Theano Development Team 16] もバージョン 1.0 公開後に開発を終了するとしている[*5]．今後幅広く利用してもらうためには，Keras [Chollet 15] など広く普及しているネットワーク実装ライブラリに切り替える必要がある．また，メタ確率分布クラスによって複数の確率分布をまとめて記述できるようになったものの，それでも書ける生成モデルには制限があるという課題が残っている．

2017 年に入ってから，GAN の確率分布の形を暗黙的にできるという特徴 [Uehara 16] を利用し，VAE の推論分布などに併用する深層生成モデルが登場している [Mescheder 17, Tran 17b, Mohamed 16, Li 17]．しかし現在の Tars ではこれらを適切に実装することができ

[*1] リポジトリを見た中でどのくらいの人数が気に入ったかを表す指標．Facebook や Twitter の「いいね」に該当する．

[*2] https://github.com/tensorflow/tensor2tensor

[*3] https://wba-initiative.org

[*4] http://brainvalley.jp/brica

[*5] https://groups.google.com/forum/#!topic/theano-users/7Poq8BZutbY，2018 年 2 月アクセス．

ない（これは Edward でも同様である）．このように急速に進歩する深層生成モデルの研究に対応できるような開発を，今後も進めていく必要がある．

9.3 本研究の技術の適用可能性について

ここでは，本研究で提案した技術の適用可能性について議論する．

1.1.1 節で説明したように，ビッグデータの活用やロボット技術の進展によって，マルチモーダル情報を処理する枠組みは重要になりつつある．本研究は 2 つの点で，実世界への適用可能性があると考えられる．

まず 1 つが，特徴空間や分布が異なるモダリティにおいて適切に学習できるという点である．たとえば，研究 2（6 章）の実験で用いた RGB-D は，ロボットが空間認識のためにセンサとして搭載しているため，そのまま技術を利用できることが期待される．また，研究 1（5 章）や研究 3（7 章）で用いた属性情報と画像といった情報は，Flickr[*6]や Pixiv[*7]といった Web サイトでも得られ，これらのページや内容を表すより良い表現を獲得することが可能になる．その一方で，本論文では，9.2 節で議論したように，多くのモダリティや系列データにおいて，提案アプローチが有効であることを検証しきれていない．したがって，多くのセンサを搭載したロボットや，大量の情報が含まれるビッグデータに直接適用して，有益な結果が得られるかは不明である．

2 つ目は，本研究で提案する枠組みが，モダリティやラベルがなかったり欠損したりした場合に対処できることである．これは，本研究のアプローチが，モダリティやラベルの分布を考慮した生成モデルによるものだからである．実データにおいても，学習時や学習後に欠損していることは非常に多い．複数のセンサを搭載しているロボットは，あるセンサが故障や不具合を起こす可能性があり，その場合でも適切に状況判断することが求められる．研究 1 や研究 2 で提案したモダリティの欠損補間の技術を使うことで，こうした問題に対処できると考えられる．また，実世界のマルチモーダルデータにはラベルはついておらず，すべてのカテゴリを考慮してラベルづけをしたり，大量のマルチモーダルデータすべてにラベルづけをすることは困難である．こうした場合にも，研究 2 や研究 3 の技術を使うことで，大幅に人的コストを抑え

[*6] https://www.flickr.com
[*7] https://www.pixiv.net

ることが可能となる．

9.4　本研究の統一構想について

本研究では，マルチモーダル学習のそれぞれの問題設定に応じて，深層ニューラルネットワークと生成モデルによるモデルを設計することで，対処できることを示している．したがって，マルチモーダル学習の様々な問題設定に対して，汎用性のあるモデルを提案している訳ではないことを明記しておく．

2.3 節で説明したように，深層学習と生成モデルを組み合わせる方法は「深層生成モデルのアプローチ」と「深層学習の特徴抽出 + 生成モデルのアプローチ」の 2 種類があり，研究 1, 2 は前者，研究 3 は後者を採用している．この 2 つのアプローチは，マルチモーダル学習において異なる利点を持つ．「深層生成モデルのアプローチ」は自然画像などの次元の大きい複雑なデータを扱うことができるが，潜在変数の階層を深くすることは難しいとされ，また潜在変数には明示的な意味を持たせないことが多い．一方「深層学習の特徴抽出 + 生成モデルのアプローチ」は， 階層ベイズに代表されるように，階層的かつ明示的に確率変数間の複雑な関係を記述することができ，確率変数それぞれにも意味を持たせることができるが，自然画像などを生成することができない．よって，これらの特徴を組み合わせることができれば，より汎用的なマルチモーダル学習のモデルが構築できる．

2 つのアプローチを組み合わせる方法の 1 つが，深層生成モデルの潜在変数を，階層ベイズ等の確率モデルの入力としてとるという方法である．たとえば，Goyal らは VAE の潜在変数の事前分布として，ノンパラメトリックトピックモデルを用いる手法を提案している [Goyal 17]．これは，トピックモデルと VAE をそれぞれモデル化し VAE の潜在変数 $\boldsymbol{z}$ をトピックモデルの入力にするというものである．この発想に従って統一モデルを考える．まず，画像や文書，音声といった，深層学習によって扱う必要があるものは研究 1 のように深層生成モデルでモデル化する．そして，属性やタグなど深層学習で扱う必要がなく，研究 3 の観測確率のように明示的に関係を設計したい場合は，マルチモーダル pLSA [Monay 03, Lienhart 09, Chandrika 10, Nikolopoulos 13] のような階層ベイズモデルを用いる．そして階層ベイズモデルの 1 つの入力モダリティをマルチモーダル深層生成モデルの潜在変数とすることで，これらのモデルを統合することができる．この統合モデルのイメー

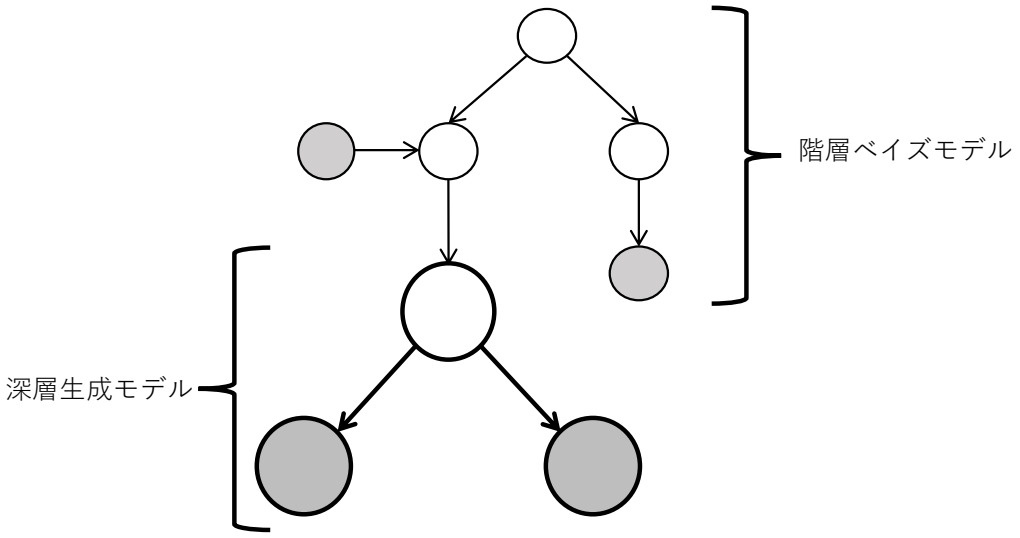


図 9.1　マルチモーダル学習のための生成モデルの構想（統合モデル）.

ジを図 9.1 に示す.

ここで示した統一モデルは，あくまで構想であり，実際にこのアプローチが適切にマルチモーダル情報を学習できるかは定かではない．数多くのモダリティを統合するという意味では，前述した MultiModel の方が学習は容易に行える可能性は高い．しかし，モダリティ間の関係を明示的に記述したり分布の違いを考慮するためには，生成モデルを利用しなければならない．逆に，後述するように「内部モデル」としての生成モデルを考えると，生成モデルは複数のモダリティが統合できる必要がある．その場合の 1 つのアイディアとして，こうした統一モデルが候補になると思われる.

9.5　汎用的な人工知能に向けた考察

最後に，汎用的な人工知能を念頭に，それに向けた本研究の考察をする.

本研究では，生成モデルはデータの生成過程を表現するものとして扱っているが，人間の脳の観点からすると，世界をモデル化する**内部モデル（internal model）**と考えることができる．人間は，外界からの刺激を元に，外界世界をシミュレートするモデルを学習しているとされる．これが内部モデルと呼ばれ，人間は内部モデルに基づいて行動することで，実際に外界に対して行動する前に，何が起こるかを予測することができる．人工知能でこれをモデル化す

ると，内部モデルは生成モデルによって実現される．

内部モデルを機械学習における生成モデルと捉え，行動と結びつけた枠組で有名なのが Friston による自由エネルギー原理（free-energy principle）[Friston 10] である．自由エネルギー原理では，生物学的なシステムが内部状態の自由エネルギーを最小化することによって秩序を維持していると考えている．

状態 $\boldsymbol{x}$[*8] と潜在変数 $\boldsymbol{z}$ を持つ生成モデル $p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})$ を考えて，近似分布を $q_{\boldsymbol{\phi}}(\boldsymbol{z})$ とする．また，負の周辺尤度の上界である変分自由エネルギー（負の変分下界）を $\mathcal{F}(\boldsymbol{x}; \boldsymbol{\phi}, \boldsymbol{\theta}) = -E_{q_{\boldsymbol{\phi}}(\boldsymbol{z})}[\log p(\boldsymbol{x}, \boldsymbol{z})] + \mathcal{H}[q_{\boldsymbol{\phi}}(\boldsymbol{z})]$ とする．自由エネルギー原理では，内部パラメータ $\boldsymbol{\phi}$ と行動 $\boldsymbol{a}$ は，（変分）自由エネルギーを最小化するように更新すると考える．

$$\hat{\boldsymbol{\phi}} = \arg\min_{\boldsymbol{\phi}} \mathcal{F}(\boldsymbol{x}; \boldsymbol{\phi}, \boldsymbol{\theta}),$$
$$\hat{\boldsymbol{a}} = \arg\min_{\boldsymbol{a}} \mathcal{F}(\boldsymbol{x}; \boldsymbol{\phi}, \boldsymbol{\theta}).$$

なお，ここでの $\arg\min_{\boldsymbol{a}}$ は，自由エネルギーが最小になるような $\boldsymbol{x}$ を選ぶ行動 $\boldsymbol{a}$ を取るということである．また，生成モデルのパラメータ $\boldsymbol{\theta}$ については，上記の更新を一定数繰り返した後に更新する．

自由エネルギー原理では，入力は単純に状態 $\boldsymbol{x}$ として考えられている．ある状態 $\boldsymbol{x}$ を受け取ったときに内部状態が更新され，その後生成モデルを元に，自由エネルギーが最小になるような状態 $\boldsymbol{x}$ を選ぶ行動 $\boldsymbol{a}$ が取られる．しかし実際には，外界からの刺激は五感を通じてマルチモーダル情報として得られるため，自由エネルギーは複数のモダリティ $\boldsymbol{x}$ や $\boldsymbol{w}$ を含んだ $\mathcal{F}(\boldsymbol{x}, \boldsymbol{w}; \boldsymbol{\phi}, \boldsymbol{\theta})$ という形になり，生成モデルは $p_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z})$ となる．つまり，複数のモダリティ入力を統合して扱う仕組みが脳の内部モデルに含まれていると考えられる（図 9.2）．したがって，複数のモダリティを扱える生成モデルが必要となる．

本論文では，マルチモーダル情報を扱うために深層学習と生成モデルが重要であることを議論した．しかし上記のように考えると，本論文では，マルチモーダル情報を含んだ内部モデル $p(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z})$（ラベルを含めると $p(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{z}, \boldsymbol{y})$）のモデル化の方法と学習方法を提案していると考えることもできる．研究 1 では，モダリティの欠損によって内部表現（潜在変数）が崩れない内部モデルを提案し，研究 2 ではラベルを含んだ内部モデルにおいて訓練集合にラベル情報があまりない場合に対処するモデルと学習方法を提案した．研究 3 では，訓練集合とテスト集

[*8] 通常，状態は $\boldsymbol{s}$ で表されることが多いが，ここではこれまでの入力の表記と合わせている．

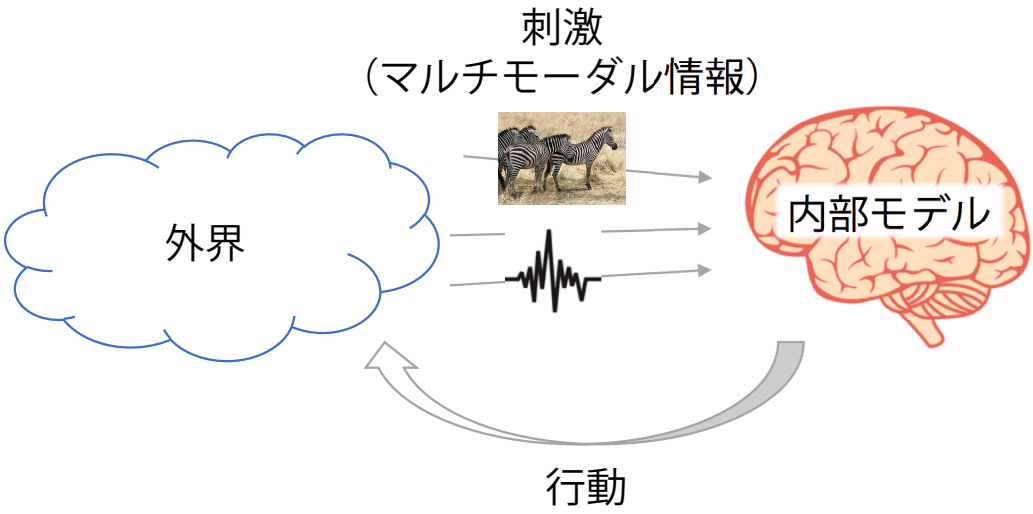


図 9.2　マルチモーダル情報と内部モデル.

合でタスクが異なる場合に他のモダリティによって対処できる内部モデルを提案した．そして研究 4 では，内部モデルを設計するためのライブラリを提案した.

なお，状態 $\boldsymbol{x}$ が与えられたときにどのような行動 $\boldsymbol{a}$ を選択するかという設定は，強化学習（reinforcement learning）の問題設定である．内部モデルを含んだ強化学習はモデルベース強化学習とみなされ，これまでも研究が進められている．しかし従来のモデルベース強化学習では，内部モデルは教師あり学習で学習するものとされ，生成モデルと組み合わせた方法は現時点ではほとんどない．自由エネルギー原理からも，行動の選択と内部モデルが密接に繋がっているのは明らかである．したがって，今後は深層生成モデルと深層強化学習の研究は連携して進められるべきであり，そのときに，マルチモーダル情報を如何に取り入れるかという観点も同時に重要となる.

第 10 章

結論

本論文は，マルチモーダル学習における複数の問題設定において，深層学習と生成モデルに基づく新たなモデルを提案した．また，深層生成モデルを簡潔かつ汎用性が高く実装できるライブラリとして Tars を提案した．

1 章では，本研究の背景や目的について述べて，2 章と 3 章では本研究の前提知識及び関連研究について説明した．4 章では 3 章を踏まえた本論文の目標及びそれ以降の研究の位置付けについて明確化した．5 章から 7 章では，マルチモーダル学習の問題設定のそれぞれにおいて，深層学習と生成モデルに基づく手法が有効であることを示した．また，8 章では本論文で提案したような複雑な深層生成モデルを実装・利用するためのライブラリの開発するという目標に対して，Tars を開発した．

5 章では，異なる種類のモダリティ間を双方向に変換できる深層生成モデルとして，JMVAE を提案した．また，推論で情報量の大きいモダリティを欠損させると，推論した潜在変数や生成したモダリティが崩れてしまう可能性があることを確認し．既存の欠損値補完の手法でも解決できないことを明らかにした．この問題を解決するために，JMVAE-kl と階層的 JMVAE という追加的な手法を提案した．実験から，これらの手法によって，欠損モダリティ問題が解決すること，すべてのモダリティを統合した適切な共有表現が獲得されること，従来の 1 方向しか生成できないモデルと比較して，同等以上の精度で双方向に変換できることを確認した．

6 章では，深層生成モデルを用いた半教師ありマルチモーダル学習のモデルとして SS-MVAE と SS-HMVAE をそれぞれ提案した．また，モダリティが欠損しても精度を落とさずに目標ラベルを予測する手法として，SS-HMVAE を拡張した SS-HMVAE-kl を導入した．こ

の手法でモダリティの欠損が補完できること，そして単一モダリティ・マルチモーダルの両方で，既存手法よりも精度の高い半教師あり学習ができることを確認した．

7 章では，マルチモーダル情報である属性を補助情報として用いた，属性ベースゼロショット学習に取り組んだ．本研究では，既存研究では考慮していなかった，属性の画像に対する分布の違い（現れやすさ）を属性ごとの観測確率と呼び，それを含めた生成モデルを提案した．既存研究との比較実験によって，提案手法が既存手法と比較して有効性の高いモデルであることを示した．

8 章では，深層生成モデルライブラリ Tars を提案した．Tars は，ネットワークが確率分布に隠蔽されており，またサンプリングや尤度計算が，分布に依存しない形となっている．さらに，メタ確率分布クラスによって，複数の確率分布をまとめて 1 つの分布として書くことができるようになっている．このため，ネットワークや確率分布に依存せずに下界の計算ができるようになり，簡潔かつ汎用性の高い実装を実現している．また，学習した確率分布は保存したり読み込んだりできるので，アプリケーションにそのまま利用することができる．簡単な実験及び応用例の紹介によって，以上のことを示した．

9 章では，ここまでの各論を踏まえて，提案手法の貢献と限界について議論した．また本研究の産業応用の可能性ついて触れ，本研究の統一構想や汎用人工知能に向けた考察をした．

マルチモーダル情報処理は，今後人工知能が発展する中で，ますます重要になっていくと考えられる．本論文で得られた知見や提案したモデル，ライブラリを活用して，マルチモーダル学習や，深層学習と生成モデルによる新たな研究が生まれることを期待する．

付録 A

異なるモダリティ間の双方向生成のための深層生成モデル

A.1 階層的 JMVAE，CVAE，CMMA における条件付き対数尤度の導出

CVAE の複数条件付き対数尤度 $\log p(\tilde{\boldsymbol{x}}|\boldsymbol{x},\boldsymbol{w})$ は，次のようになる．

$$\begin{aligned}\log p(\tilde{\boldsymbol{x}}|\boldsymbol{x},\boldsymbol{w}) &\simeq \log \int q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})p(\tilde{\boldsymbol{x}}|\boldsymbol{z},\boldsymbol{w})d\boldsymbol{z} \\ &\geq \int q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})\log p(\tilde{\boldsymbol{x}}|\boldsymbol{z},\boldsymbol{w})d\boldsymbol{z} \\ &\simeq \frac{1}{N}\sum_{i=1}^{N}\log p(\tilde{\boldsymbol{x}}|\boldsymbol{z}^{(l)},\boldsymbol{w}). \end{aligned} \tag{A.1}$$

ただし，$\boldsymbol{z}^{(l)} \sim q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})$ である．

CMMA の複数条件付き対数尤度は，JMVAE と同様式 (5.7) で計算できる．

階層的 JMVAE の複数条件付き対数尤度は

$$\begin{aligned}\log p(\tilde{\boldsymbol{x}}|\boldsymbol{x},\boldsymbol{w}) &\simeq \log \int q(\boldsymbol{z}_2|\boldsymbol{x},\boldsymbol{w}) \int p(\boldsymbol{z}_1|\boldsymbol{z}_2)p(\tilde{\boldsymbol{x}}|\boldsymbol{z}_1)d\boldsymbol{z}_1 d\boldsymbol{z}_2 \\ &\geq \int q(\boldsymbol{z}_2|\boldsymbol{x},\boldsymbol{w}) \log \int p(\boldsymbol{z}_1|\boldsymbol{z}_2)p(\tilde{\boldsymbol{x}}|\boldsymbol{z}_1)d\boldsymbol{z}_1 d\boldsymbol{z}_2 \\ &\geq \int q(\boldsymbol{z}_2|\boldsymbol{x},\boldsymbol{w}) \int p(\boldsymbol{z}_1|\boldsymbol{z}_2) \log p(\tilde{\boldsymbol{x}}|\boldsymbol{z}_1)d\boldsymbol{z}_1 d\boldsymbol{z}_2 \\ &\simeq \frac{1}{N}\sum_{i=1}^{N} \int p(\boldsymbol{z}_1^{(l)}|\boldsymbol{z}_2) \log p(\tilde{\boldsymbol{x}}|\boldsymbol{z}_1)d\boldsymbol{z}_1 \end{aligned} \tag{A.2}$$

と計算できる．ただし，$\boldsymbol{z}_2^{(l)} \sim q(\boldsymbol{z}_2|\boldsymbol{x},\boldsymbol{w})$ であり，積分部分については次のように近似できる．

$$\int p(\boldsymbol{z}_1^{(l)}|\boldsymbol{z}_2) \log p(\tilde{\boldsymbol{x}}|\boldsymbol{z}_1)d\boldsymbol{z}_1 \simeq \frac{1}{N}\sum_{j=1}^{N} \log p(\tilde{\boldsymbol{x}}|\boldsymbol{z}_1^{(k)}).$$

ただし，$\boldsymbol{z}_1^{(k)} \sim q(\boldsymbol{z}_1|\boldsymbol{z}_2^{(l)})$.

式 (A.2) の近似式は，式 (5.7) や式 (A.1) よりも下界をおさえた式となっている．したがって他の評価値よりも，実際の尤度と比べたときの過小評価の度合いが高くなる可能性があることに留意されたい．

単数条件付き対数尤度の近似式は，CVAE と階層的 JMVAE については，通常の JMVAE と同様，片方の入力を欠損させたエンコーダからサンプリングすることで導出する．CMMA については $p(\boldsymbol{z}|\boldsymbol{x})$ または $p(\boldsymbol{z}|\boldsymbol{w})$ からサンプリングして導出する．

A.2 JMVAE-kl の目的関数と variation of information の関係について

Variation of information（VI）は $-E_{p_{\mathcal{D}}(\boldsymbol{x},\boldsymbol{w})}[\log p(\boldsymbol{x}|\boldsymbol{w}) + \log p(\boldsymbol{w}|\boldsymbol{x})]$（ただし $p_{\mathcal{D}}$ はデータ分布）と表現される．以降の導出では，この式の負の対数尤度の和に着目し，期待値については考慮しない．

対数尤度の和 $\log p(\boldsymbol{x}|\boldsymbol{w}) + \log p(\boldsymbol{w}|\boldsymbol{x})$ の変分下界は次のように計算できる．

$$
\begin{aligned}
&\log p(\boldsymbol{x}|\boldsymbol{w}) + \log p(\boldsymbol{w}|\boldsymbol{x}) \\
&\geq E_{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}[\log \frac{p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}|\boldsymbol{w})}{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}] + E_{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}[\log \frac{p(\boldsymbol{w}|\boldsymbol{z})p(\boldsymbol{z}|\boldsymbol{x})}{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}] \\
&= E_{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}[\log p(\boldsymbol{x}|\boldsymbol{z})] + E_{q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})}[\log p(\boldsymbol{w}|\boldsymbol{z})] \\
&\quad - D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||p(\boldsymbol{z}|\boldsymbol{x})) - D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||p(\boldsymbol{z}|\boldsymbol{w})) \\
&= \mathcal{L}_{JM}(\boldsymbol{x},\boldsymbol{w}) - [D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||p(\boldsymbol{z}|\boldsymbol{x})) + D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||p(\boldsymbol{z}|\boldsymbol{w}))] \\
&\quad + D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||p(\boldsymbol{z})).
\end{aligned} \tag{A.3}
$$

もし，すべての確率分布がニューラルネットワークなどでパラメータ化されていれば，同じネットワーク構造で表現できるので，$p(\boldsymbol{z}|\boldsymbol{x})$ と $p(\boldsymbol{z}|\boldsymbol{w})$ のそれぞれを，$q(\boldsymbol{z}|\boldsymbol{x})$ と $q(\boldsymbol{z}|\boldsymbol{w})$ と置き換えることができる．したがって，上記の置き換えをした式 (A.3) は，

$$
\begin{aligned}
&\mathcal{L}_{JM}(\boldsymbol{x},\boldsymbol{w}) - [D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||q(\boldsymbol{z}|\boldsymbol{x})) + D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||q(\boldsymbol{z}|\boldsymbol{w}))] + D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||p(\boldsymbol{z})) \\
&= \mathcal{L}_{JM_{kl}}(\boldsymbol{x},\boldsymbol{w}) + D_{KL}(q(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{w})||p(\boldsymbol{z})) \geq \mathcal{L}_{JM_{kl}}(\boldsymbol{x},\boldsymbol{w}).
\end{aligned} \tag{A.4}
$$

となる．

したがって，式 (5.6) を最大化することは，パラメータ化された分布による対数尤度の和の変分下界の最大化，即ち変分推論における VI 最小化に等しい．

付録 B

属性ごとの観測確率を考慮したゼロショット学習

B.1 観測確率の導出

事後分布 $p(\theta_{m\lambda}|\boldsymbol{X})$ が最大となるような観測確率の推定量 $\hat{\theta}_{m\lambda}$ を求める．

$p(\theta_{m\lambda}|\boldsymbol{X})$ について対数を取ると，

$$
\begin{aligned}
&\log p(\theta_{m\lambda}|\boldsymbol{X}) \\
&\propto \log p(\boldsymbol{X}|\theta_{m\lambda}) + \log p(\theta_{m\lambda}|\boldsymbol{\alpha}) \\
&= \sum_{n:w_{mn}=\lambda} \log p(\boldsymbol{x}_n|\theta_{m\lambda}) + \log p(\theta_{m\lambda}|\boldsymbol{\alpha}) \\
&= \sum_{n:w_{mn}=\lambda} \log \sum_{c_{mn}} p(\boldsymbol{x}_n, c_{mn}|\theta_{m\lambda}) + \log p(\theta_{m\lambda}|\boldsymbol{\alpha}) \\
&= \sum_{n:w_{mn}=\lambda} \log \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n) \frac{p(\boldsymbol{x}_n, c_{mn}|\theta_{m\lambda})}{p(c_{mn}|\boldsymbol{x}_n)} + \log p(\theta_{m\lambda}|\boldsymbol{\alpha})
\end{aligned}
$$

となる．ここで $\log(x)$ は凸関数なので，Jensen の不等式を適用すると

$$\begin{aligned}
&\sum_{n:w_{mn}=\lambda} \log \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n) \frac{p(\boldsymbol{x}_n, c_{mn}|\theta_{m\lambda})}{p(c_{mn}|\boldsymbol{x}_n)} + \log p(\theta_{m\lambda}|\boldsymbol{\alpha}) \\
&\geq \sum_{n:w_{mn}=\lambda} \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n) \log \frac{p(\boldsymbol{x}_n, c_{mn}|\theta_{m\lambda})}{p(c_{mn}|\boldsymbol{x}_n)} + \log p(\theta_{m\lambda}|\boldsymbol{\alpha}) \\
&= \sum_{n:w_{mn}=\lambda} \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n)[\log p(\boldsymbol{x}_n|c_{mn}) + \log p(c_{mn}|\theta_{m\lambda})] \\
&\quad - \sum_{n:w_{mn}=\lambda} \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n) \log p(c_{mn}|\boldsymbol{x}_n) + \log p(\theta_{m\lambda}|\boldsymbol{\alpha})
\end{aligned}$$

となり、下界が求まる．

下界を $\theta_{m\lambda}$ について最大化するために偏微分をとる．

$$\begin{aligned}
&\sum_{n:w_{mn}=\lambda} \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n) \frac{\partial}{\partial \theta_{m\lambda}} \log p(c_{mn}|\theta_{m\lambda}) + \frac{\partial}{\partial \theta_{m\lambda}} \log p(\theta_{m\lambda}|\boldsymbol{\alpha}) \\
&= \sum_{n:w_{mn}=\lambda} \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n) \frac{\partial}{\partial \theta_{m\lambda}} \log[{\theta_{m\lambda}}^{c_{mn}} (1-\theta_{m\lambda})^{1-c_{mn}}] \\
&\qquad + \frac{\partial}{\partial \theta_{m\lambda}} \log[\frac{1}{B(\alpha_0, \alpha_1)} {\theta_{m\lambda}}^{\alpha_0 - 1} (1-\theta)^{\alpha_1 - 1}] \\
&= \sum_{n:w_{mn}=\lambda} \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n) \frac{\partial}{\partial \theta_{m\lambda}} [c_{mn} \log \theta_{m\lambda} + (1-c_{mn}) \log(1-\theta_{m\lambda})] \\
&\qquad + \frac{\partial}{\partial \theta_{m\lambda}} [-\log B(\alpha_0, \alpha_1) + (\alpha_0 - 1) \log \theta_{m\lambda} + (\alpha_1 - 1) \log(1-\theta_{m\lambda})] \\
&= \sum_{n:w_{mn}=\lambda} \sum_{c_{mn}} p(c_{mn}|\boldsymbol{x}_n) [\frac{c_{mn}}{\theta_{m\lambda}} - \frac{1-c_{mn}}{1-\theta_{m\lambda}}] + \frac{\alpha_0 - 1}{\theta_{m\lambda}} - \frac{\alpha_1 - 1}{1-\theta_{m\lambda}} \\
&= \sum_{n:w_{mn}=\lambda} [\frac{p(c_{mn}=1|\boldsymbol{x}_n)}{\theta_{m\lambda}} - \frac{p(c_{mn}=0|\boldsymbol{x}_n)}{1-\theta_{m\lambda}}] + \frac{\alpha_0 - 1}{\theta_{m\lambda}} - \frac{\alpha_1 - 1}{1-\theta_{m\lambda}} \\
&= \sum_{n:w_{mn}=\lambda} [\frac{p(c_{mn}=1|\boldsymbol{x}_n)}{\theta_{m\lambda}} - \frac{1-p(c_{mn}=1|\boldsymbol{x}_n)}{1-\theta_{m\lambda}}] + \frac{\alpha_0 - 1}{\theta_{m\lambda}} - \frac{\alpha_1 - 1}{1-\theta_{m\lambda}}.
\end{aligned}$$

この式が 0 となるような $\theta_{m\lambda}$ を $\hat{\theta}_{m\lambda}$ とすると，$\hat{\theta}_{m\lambda}$ は

$$\hat{\theta}_{m\lambda} = \frac{\sum_{n:w_{mn}=\lambda} p(c_{mn}=1|\boldsymbol{x}_n) + \alpha_0 - 1}{\sum_{n:w_{mn}=\lambda} + \alpha_0 + \alpha_1 - 2}$$

となる．

発表文献

論文誌

1. **鈴木雅大**，松尾豊，深層生成モデルを用いた半教師ありマルチモーダル学習，情報処理学会論文誌（投稿中）
2. **鈴木雅大**，松尾豊，異なるモダリティ間の双方向生成のための深層生成モデル，情報処理学会論文誌，Vol. 59, No. 3, pp. 1-15, 2018.
3. **鈴木雅大**，佐藤晴彦，小山聡，栗原正仁，松尾豊，属性ごとの観測確率を考慮したゼロショット学習，情報処理学会論文誌, Vol. 57, No. 5, pp. 1-15, 2016.

国際会議発表（査読あり）

1. **Masahiro Suzuki**, Haruhiko Sato, Satoshi Oyama, Masahito Kurihara, Transfer Learning Based on the Observation Probability of Each Attribute, Proc. 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2014), pp.3648-3652, October 2014.
2. **Masahiro Suzuki**, Haruhiko Sato, Satoshi Oyama, Masahito Kurihara, Image Classification by Transfer Learning Based on the Predictive Ability of Each Attribute, Proc. the International MultiConference of Engineers and Computer Scientists (IMECS 2014), Vol I, pp.75-78, March 2014.
3. Keiki Zen, **Masahiro Suzuki**, Haruhiko Sato, Satoshi Oyama, Masahito Kurihara, Monophonic Sound Source Separation by Non-negative Sparse Autoencoders, Proc. 2014 IEEE International Conference on Systems, Man, and Cybernetics

(SMC 2014), pp.3644-3647, October 2014.

国際会議ワークショップ発表（査読あり）

1. **Masahiro Suzuki**, Yutaka Matsuo, Semi-supervised Multimodal Learning with Deep Generative Models, submitted to ICLR 2018 workshop.
2. **Masahiro Suzuki**, Kotaro Nakayama, Yutaka Matsuo, Joint Multimodal Learning with Deep Generative Models, ICLR 2017 workshop, April 2017.
3. Masatoshi Uehara, Issei Sato, **Masahiro Suzuki**, Kotaro Nakayama, Yutaka Matsuo, Workshop on Adversarial Training, NIPS 2016 workshop, December 2016.

国際会議発表（査読なし）

1. Takeshi Itoh, Jumpei Ukita, Ayaka Kato, Takaaki Kaneko, **Masahiro Suzuki**, Yusuke Iwasawa, Modeling the development of place cells in hippocampus, 2015 Annual International Conference on Biologically Inspired Cognitive Architectures (BICA 2015), November 2015.

国内学会口頭発表（査読なし）

1. **鈴木雅大**・松尾豊, 半教師ありマルチモーダル学習のための深層生成モデル，2017 年度人工知能学会全国大会，2017.
2. 冨山翔司・味曽野雅史・**鈴木雅大**・中山浩太郎・松尾豊, 画像とテキストの潜在的な意味情報を用いたニューラル翻訳モデルの提案，2017 年度人工知能学会全国大会，2017.
3. **鈴木雅大**・松尾豊，深層生成モデルを用いたマルチモーダル学習，2016 年度人工知能学会全国大会，2016.
4. **鈴木雅大**・佐藤晴彦・小山聡・栗原正仁，属性ごとの観測確率を考慮した転移学習，2014 年度人工知能学会全国大会，2014.
5. 数原良彦・**鈴木雅大**・戸田浩之・鷲崎誠司，少数の正解ラベルを用いた移動履歴からの

移動手段判定，2014 年度人工知能学会全国大会，2014.

6. **鈴木雅大**・佐藤晴彦・小山聡・栗原正仁，属性ごとの予測能力に基づく属性ベース転移学習，第 12 回情報科学技術フォーラム講演論文集，Vol.2，pp. 371-372，2013.

国内学会ポスター発表（査読なし）

1. 上原雅俊・佐藤一誠・**鈴木雅大**・中山浩太郎・松尾豊，b-GAN: 密度比推定の視点から見た Generative Adversarial Nets，第 19 回情報論的学習理論ワークショップ，2016.
2. **鈴木雅大**・佐藤晴彦・小山聡・栗原正仁，属性ごとの予測能力を考慮した属性ベース転移学習，情報処理北海道シンポジウム 2013 講演論文集，pp. 43-46，2013.

arXiv プレプリント論文

1. **Masahiro Suzuki**, Kotaro Nakayama, Yutaka Matsuo, Improving Bi-directional Generation between Different Modalities with Variational Autoencoders, arXiv preprint arXiv:1801.08702, 2018.
2. **Masahiro Suzuki**, Kotaro Nakayama, Yutaka Matsuo, Joint Multimodal Learning with Deep Generative Models, arXiv preprint arXiv:1611.01891, 2016.
3. Joji Toyama, Masanori Misono, **Masahiro Suzuki**, Kotaro Nakayama, Yutaka Matsuo, Neural Machine Translation with Latent Semantic of Image and Text, arXiv preprint arXiv:1611.08459, 2016.
4. Masatoshi Uehara, Issei Sato, **Masahiro Suzuki**, Kotaro Nakayama, Yutaka Matsuo, Generative Adversarial Nets from a Density Ratio Estimation Perspective, arXiv preprint arXiv:1610.02920, 2016.

訳書

1. Ian Goodfellow，Yoshua Bengio，Aaron Courville 著，深層学習，監訳・分担翻訳（本書冒頭〈ウェブサイト，謝辞，表記〉，第 I 部冒頭，第 II 部冒頭，第 III 部冒頭，第 16

章，第 17 章，第 18 章，第 19 章，第 20 章)，KADOKAWA，2018 年 2 月 28 日発売予定.

謝辞

本博士論文は，多くの方のご指導やご支援がなければ，完成させることができませんでした．

指導教官である松尾 豊特任准教授には，私が松尾研究室に在籍してから 3 年間に渡り，熱心なご指導を賜りました．研究のご指導の他にも，取材や勉強会といった様々な場に出る機会を積極的に与えて下さいました．松尾研究室に在籍した 3 年間で，自分自身の視野を大きく広げることができたと思います．ここに感謝の意を表します．

また，本論文の副査を担当していただいた東京大学 大学院工学系研究科 縄田 和満教授，古田 一雄教授，森 純一郎准教授，大学院情報理工学系研究科 佐藤 一誠講師には，貴重なお時間を割いていただき，丁寧かつ的確なご指摘を数多くいただきました．特に佐藤講師には，私が北海道大学に在籍した頃からお世話になり，博士課程進学に際して松尾研究室を紹介して下さいました．ここに深く感謝申し上げます．

本論文の一部は，北海道大学在籍時に進めた研究を発展させたものです．当時の指導教官である北海道大学 大学院情報科学研究科 栗原 正仁教授，小山 聡准教授，北海学園大学 工学部 電子情報工学科 佐藤 晴彦准教授には，学部・修士課程を通じて，様々なご指導をいただきました．博士論文を無事に完成させることができたのは，北海道大学在籍時の経験があったからこそです．深く感謝致します．

松尾研究室の中山 浩太郎特任講師には，研究面だけでなく，日常的に様々なサポートをしていただきました．特に本論文に掲載した FacialVAE の Web アプリケーションは，中山講師と退職された Toma Sakai 氏が中心となって制作したものです．また，中山講師をはじめ，Emilio Castillo 氏，Alfredo Solano 氏，Michael Bawiec 氏など学術支援専門職員の皆様による計算機資源等の整備がなければ，研究を円滑に遂行することはできませんでした．深く感謝いたします．

松尾研究室秘書の永本 登代子さん，小泉 恵美子さん，吉田 和美さん，阿部 美和子さん，そしてすでに退職されましたが中野 佐恵子さん，浪岡 亮子さん，木全 弥栄さんには，書面作成や経理など様々な面でサポートをいただきました．ありがとうございました．

そして松尾研究室の研究仲間の皆様がいなければ，継続的に研究活動を進めることはできませんでした．特に，岩澤 有祐特任研究員には，松尾研究室に入ってから数多くの場面でご指導をいただき，研究について何度も相談にのっていただきました．野中 尚輝氏には，日常的に様々な場面でお世話になりました．金子 貴輝氏には，同学年であり，ほぼ唯一松尾研究室で理論方面について議論できる相手として，時間を割いて相談にのっていただきました．本研究の内容のいくつかは金子氏との議論の中で着想を得たものです．ここに改めて感謝いたします．

最後に，博士課程修了までの気の遠くなるような長い学生生活を，金銭面と精神面から支えていただいた家族全員に心から深く感謝致します．１つのことを最後までやり遂げるのが苦手な私でしたが，父からの「お前は研究者に向いている」という言葉が大きな励みとなり，こうして博士論文を完成させることができました．本当にありがとうございました．

東京大学 大学院工学系研究科
技術経営戦略学専攻
鈴木 雅大

参考文献

[Abadi 15] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems (2015), Software available from tensorflow.org

[Akata 13] Akata, Z., Perronnin, F., Harchaoui, Z., and Schmid, C.: Label-embedding for attribute-based classification, in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pp. 819–826IEEE (2013)

[Antol 15] Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., and Parikh, D.: Vqa: Visual question answering, in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2425–2433 (2015)

[Arjovsky 17] Arjovsky, M., Chintala, S., and Bottou, L.: Wasserstein gan, *arXiv preprint arXiv:1701.07875* (2017)

[Bachman 16] Bachman, P.: An architecture for deep, hierarchical generative models, in *Advances in Neural Information Processing Systems*, pp. 4826–4834 (2016)

[Baltrušaitis 17] Baltrušaitis, T., Ahuja, C., and Morency, L.-P.: Multimodal Machine Learning: A Survey and Taxonomy, *arXiv preprint arXiv:1705.09406* (2017)

[Bay 08] Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L.: Speeded-up robust features

(SURF), *Computer vision and image understanding*, Vol. 110, No. 3, pp. 346–359 (2008)

[Bengio 13a] Bengio, Y., Courville, A., and Vincent, P.: Representation learning: A review and new perspectives, *IEEE transactions on pattern analysis and machine intelligence*, Vol. 35, No. 8, pp. 1798–1828 (2013)

[Bengio 13b] Bengio, Y., Mesnil, G., Dauphin, Y., and Rifai, S.: Better mixing via deep representations, in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pp. 552–560 (2013)

[Beyer 12] Beyer, M. A. and Laney, D.: The Importance of 'Big Data': A Definition, *Stamford, CT, USA: Gartner* (2012)

[Bosch 07] Bosch, A., Zisserman, A., and Munoz, X.: Representing shape with a spatial pyramid kernel, in *Proceedings of the 6th ACM international conference on Image and video retrieval*, pp. 401–408ACM (2007)

[Bowman 15] Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Jozefowicz, R., and Bengio, S.: Generating sentences from a continuous space, *arXiv preprint arXiv:1511.06349* (2015)

[Bucak 14] Bucak, S. S., Jin, R., and Jain, A. K.: Multiple kernel learning for visual object recognition: A review, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 36, No. 7, pp. 1354–1369 (2014)

[Burda 15] Burda, Y., Grosse, R., and Salakhutdinov, R.: Importance weighted autoencoders, *arXiv preprint arXiv:1509.00519* (2015)

[Caglayan 16] Caglayan, O., Barrault, L., and Bougares, F.: Multimodal Attention for Neural Machine Translation, *arXiv preprint arXiv:1609.03976* (2016)

[Cao 16] Cao, Y., Long, M., Wang, J., Yang, Q., and Philip, S. Y.: Deep Visual-Semantic Hashing for Cross-Modal Retrieval., in *KDD*, pp. 1445–1454 (2016)

[Chandrika 10] Chandrika, P. and Jawahar, C.: Multi modal semantic indexing for image retrieval, in *Proceedings of the ACM International Conference on Image and Video Retrieval*, pp. 342–349ACM (2010)

[Chen 15] Chen, X., Fang, H., Lin, T.-Y., Vedantam, R., Gupta, S., Dollár, P., and Zit-

nick, C. L.: Microsoft COCO captions: Data collection and evaluation server, *arXiv preprint arXiv:1504.00325* (2015)

[Cheng 15] Cheng, Y., Zhao, X., Huang, K., and Tan, T.: Semi-supervised learning and feature evaluation for rgb-d object recognition, *Computer Vision and Image Understanding*, Vol. 139, pp. 149–160 (2015)

[Cheng 16] Cheng, Y., Zhao, X., Cai, R., Li, Z., Huang, K., and Rui, Y.: Semi-Supervised Multimodal Deep Learning for RGB-D Object Recognition (2016)

[Chollet 15] Chollet, F., et al.: Keras, https://github.com/fchollet/keras (2015)

[Chung 12] Chung, J., Lee, D., Seo, Y., and Yoo, C. D.: Deep attribute networks, in *Deep Learning and Unsupervised Feature Learning NIPS Workshop*, Vol. 3 (2012)

[Chung 14] Chung, J., Gulcehre, C., Cho, K., and Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling, *arXiv preprint arXiv:1412.3555* (2014)

[Chung 15] Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A. C., and Bengio, Y.: A recurrent latent variable model for sequential data, in *Advances in neural information processing systems*, pp. 2980–2988 (2015)

[Dalal 05] Dalal, N. and Triggs, B.: Histograms of oriented gradients for human detection, in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, Vol. 1, pp. 886–893IEEE (2005)

[Dieleman 15] Dieleman, S., Schlüter, J., Raffel, C., Olson, E., Sønderby, S. K., Nouri, D., Maturana, D., Thoma, M., Battenberg, E., Kelly, J., Fauw, J. D., Heilman, M., Almeida, de D. M., McFee, B., Weideman, H., Takács, G., Rivaz, de P., Crall, J., Sanders, G., Rasul, K., Liu, C., French, G., and Degrave, J.: Lasagne: First release. (2015)

[Donahue 14] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition, in *International conference on machine learning*, pp. 647–655 (2014)

[Du 17] Du, C., Du, C., Li, J., Zheng, W.-l., Lu, B.-l., and He, H.: Semi-supervised

Bayesian Deep Multi-modal Emotion Recognition, *arXiv preprint arXiv:1704.07548* (2017)

[Eitel 15] Eitel, A., Springenberg, J. T., Spinello, L., Riedmiller, M., and Burgard, W.: Multimodal deep learning for robust RGB-D object recognition, in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 681–687IEEE (2015)

[Farhadi 09] Farhadi, A., Endres, I., Hoiem, D., and Forsyth, D.: Describing objects by their attributes, in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 1778–1785IEEE (2009)

[Farhadi 10] Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., and Forsyth, D.: Every picture tells a story: Generating sentences from images, in *European conference on computer vision*, pp. 15–29Springer (2010)

[Friedman 08] Friedman, J., Hastie, T., and Tibshirani, R.: Sparse inverse covariance estimation with the graphical lasso, *Biostatistics*, Vol. 9, No. 3, pp. 432–441 (2008)

[Friston 10] Friston, K.: The free-energy principle: a unified brain theory?, *Nature Reviews Neuroscience*, Vol. 11, No. 2, pp. 127–138 (2010)

[Frome 13] Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al.: Devise: A deep visual-semantic embedding model, in *Advances in neural information processing systems*, pp. 2121–2129 (2013)

[Fu 14a] Fu, Y., Hospedales, T. M., Xiang, T., Fu, Z., and Gong, S.: Transductive multi-view embedding for zero-shot recognition and annotation, in *European Conference on Computer Vision*, pp. 584–599Springer (2014)

[Fu 14b] Fu, Y., Hospedales, T. M., Xiang, T., and Gong, S.: Learning multimodal latent attributes, *IEEE transactions on pattern analysis and machine intelligence*, Vol. 36, No. 2, pp. 303–316 (2014)

[Fu 14c] Fu, Z.-Y., Xiang, T., and Gong, S.: Semantic Graph for Zero-Shot Learning, *arXiv preprint arXiv:1406.4112* (2014)

[Gers 99] Gers, F. A., Schmidhuber, J., and Cummins, F.: Learning to forget: Continual

prediction with LSTM (1999)

[Glorot 10] Glorot, X. and Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks, in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256 (2010)

[Goodfellow 14] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y.: Generative adversarial nets, in *Advances in Neural Information Processing Systems*, pp. 2672–2680 (2014)

[Goodfellow 16] Goodfellow, I., Bengio, Y., and Courville, A.: *Deep learning*, MIT press (2016)

[Goyal 17] Goyal, P., Hu, Z., Liang, X., Wang, C., and Xing, E.: Nonparametric Variational Auto-encoders for Hierarchical Representation Learning, *arXiv preprint arXiv:1703.07027* (2017)

[Gregor 15] Gregor, K., Danihelka, I., Graves, A., Rezende, D. J., and Wierstra, D.: DRAW: A recurrent neural network for image generation, *arXiv preprint arXiv:1502.04623* (2015)

[Guillaumin 10] Guillaumin, M., Verbeek, J., and Schmid, C.: Multimodal semi-supervised learning for image classification, in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 902–909IEEE (2010)

[Gulrajani 16] Gulrajani, I., Kumar, K., Ahmed, F., Taiga, A. A., Visin, F., Vazquez, D., and Courville, A.: PixelVAE: A Latent Variable Model for Natural Images, *arXiv preprint arXiv:1611.05013* (2016)

[He 15] He, K., Zhang, X., Ren, S., and Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034 (2015)

[Higgins 17] Higgins, I., Sonnerat, N., Matthey, L., Pal, A., Burgess, C. P., Botvinick, M., Hassabis, D., and Lerchner, A.: SCAN: Learning Abstract Hierarchical Compositional Visual Concepts, *arXiv preprint arXiv:1707.03389* (2017)

[Hochreiter 97] Hochreiter, S. and Schmidhuber, J.: Long short-term memory, *Neural*

computation, Vol. 9, No. 8, pp. 1735–1780 (1997)

[Hoffman 14] Hoffman, M. D. and Gelman, A.: The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo, *Journal of Machine Learning Research*, Vol. 15, pp. 1593–1623 (2014)

[Huiskes 08] Huiskes, M. J. and Lew, M. S.: The MIR flickr retrieval evaluation, in *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pp. 39–43ACM (2008)

[Hunt 96] Hunt, A. J. and Black, A. W.: Unit selection in a concatenative speech synthesis system using a large speech database, in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, Vol. 1, pp. 373–376IEEE (1996)

[Hwang 11] Hwang, S. J., Sha, F., and Grauman, K.: Sharing features between objects and their attributes, in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1761–1768IEEE (2011)

[Ioffe 15] Ioffe, S. and Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift, in *International Conference on Machine Learning*, pp. 448–456 (2015)

[Isola 16] Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A.: Image-to-image translation with conditional adversarial networks, *arXiv preprint arXiv:1611.07004* (2016)

[Jang 16] Jang, E., Gu, S., and Poole, B.: Categorical Reparameterization with Gumbel-Softmax, *arXiv preprint arXiv:1611.01144* (2016)

[Kaiser 17] Kaiser, L., Gomez, A. N., Shazeer, N., Vaswani, A., Parmar, N., Jones, L., and Uszkoreit, J.: One Model To Learn Them All, *CoRR*, Vol. abs/1706.05137, (2017)

[Kankuekul 12] Kankuekul, P., Kawewong, A., Tangruamsub, S., and Hasegawa, O.: Online incremental attribute-based zero-shot learning, in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 3657–3664IEEE (2012)

[Kansky 17] Kansky, K., Silver, T., Mély, D. A., Eldawy, M., Lázaro-Gredilla, M., Lou, X., Dorfman, N., Sidor, S., Phoenix, S., and George, D.: Schema Networks: Zero-

shot Transfer with a Generative Causal Model of Intuitive Physics, *arXiv preprint arXiv:1706.04317* (2017)

[Karl Weiss 16] Karl Weiss, T. M. K. and Wang, D.: A survey of transfer learning, *Journal of Big Data* (2016)

[Kingma 13] Kingma, D. P. and Welling, M.: Auto-encoding variational bayes, *arXiv preprint arXiv:1312.6114* (2013)

[Kingma 14a] Kingma, D. P., Mohamed, S., Rezende, D. J., and Welling, M.: Semi-supervised learning with deep generative models, in *Advances in Neural Information Processing Systems (NIPS)*, pp. 3581–3589 (2014)

[Kingma 14b] Kingma, D. and Ba, J.: Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* (2014)

[Krizhevsky 12] Krizhevsky, A., Sutskever, I., and Hinton, G. E.: Imagenet classification with deep convolutional neural networks, in *Advances in neural information processing systems*, pp. 1097–1105 (2012)

[Kucukelbir 15] Kucukelbir, A., Ranganath, R., Gelman, A., and Blei, D.: Automatic variational inference in Stan, in *Advances in neural information processing systems*, pp. 568–576 (2015)

[Kulkarni 15] Kulkarni, T. D., Whitney, W. F., Kohli, P., and Tenenbaum, J.: Deep convolutional inverse graphics network, in *Advances in Neural Information Processing Systems*, pp. 2539–2547 (2015)

[Lahat 15] Lahat, D., Adali, T., and Jutten, C.: Multimodal data fusion: an overview of methods, challenges, and prospects, *Proceedings of the IEEE*, Vol. 103, No. 9, pp. 1449–1477 (2015)

[Lai 11] Lai, K., Bo, L., Ren, X., and Fox, D.: A large-scale hierarchical multi-view rgb-d object dataset, in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 1817–1824IEEE (2011)

[Lampert 09] Lampert, C. H., Nickisch, H., and Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer, in *Computer Vision and Pattern*

Recognition, 2009. CVPR 2009. IEEE Conference on, pp. 951–958IEEE (2009)

[Lampert 14] Lampert, C. H., Nickisch, H., and Harmeling, S.: Attribute-based classification for zero-shot visual object categorization, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 36, No. 3, pp. 453–465 (2014)

[Lample 17] Lample, G., Denoyer, L., and Ranzato, M.: Unsupervised Machine Translation Using Monolingual Corpora Only, *arXiv preprint arXiv:1711.00043* (2017)

[Larochelle 08] Larochelle, H., Erhan, D., and Bengio, Y.: Zero-data learning of new tasks., in *AAAI*, Vol. 1, p. 3 (2008)

[Larsen 15a] Larsen, A. B. L., Sønderby, S. K., Larochelle, H., and Winther, O.: Autoencoding beyond pixels using a learned similarity metric, *arXiv preprint arXiv:1512.09300* (2015)

[Larsen 15b] Larsen, A. B. L., Sønderby, S. K., and Winther, O.: Autoencoding beyond pixels using a learned similarity metric, *arXiv preprint arXiv:1512.09300* (2015)

[LeCun 98] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P.: Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, Vol. 86, No. 11, pp. 2278–2324 (1998)

[LeCun 15] LeCun, Y., Bengio, Y., and Hinton, G.: Deep learning, *Nature*, Vol. 521, No. 7553, pp. 436–444 (2015)

[Li 13] Li, F., Zhou, S.-R., Zhang, J.-M., Zhang, D.-Y., and Xiang, L.-Y.: Attribute-based knowledge transfer learning for human pose estimation, *Neurocomputing*, Vol. 116, pp. 301–310 (2013)

[Li 17] Li, Y. and Turner, R. E.: Gradient Estimators for Implicit Models, *arXiv preprint arXiv:1705.07107* (2017)

[Lienhart 09] Lienhart, R., Romberg, S., and Hörster, E.: Multilayer pLSA for multimodal image retrieval, in *Proceedings of the ACM International Conference on Image and Video Retrieval*, p. 9ACM (2009)

[Liu 15] Liu, Z., Luo, P., Wang, X., and Tang, X.: Deep learning face attributes in the wild, in *Proceedings of the IEEE International Conference on Computer Vision*, pp.

3730–3738 (2015)

[Liu 16] Liu, M.-Y. and Tuzel, O.: Coupled generative adversarial networks, in *Advances in Neural Information Processing Systems*, pp. 469–477 (2016)

[Liu 17] Liu, M.-Y., Breuel, T., and Kautz, J.: Unsupervised image-to-image translation networks, in *Advances in Neural Information Processing Systems*, pp. 700–708 (2017)

[Logan 00] Logan, B., et al.: Mel Frequency Cepstral Coefficients for Music Modeling., in *ISMIR* (2000)

[Louizos 15] Louizos, C., Swersky, K., Li, Y., Welling, M., and Zemel, R.: The variational fair auto encoder, *arXiv preprint arXiv:1511.00830* (2015)

[Lowe 04a] Lowe, D. G.: Distinctive image features from scale-invariant keypoints, *International journal of computer vision*, Vol. 60, No. 2, pp. 91–110 (2004)

[Lowe 04b] Lowe, D. G.: Method and apparatus for identifying scale invariant features in an image and use of same for locating an object in an image (2004), US Patent 6,711,293

[Luo 13] Luo, Y., Tao, D., Geng, B., Xu, C., and Maybank, S. J.: Manifold regularized multitask learning for semi-supervised multilabel image classification, *IEEE Transactions on Image Processing*, Vol. 22, No. 2, pp. 523–536 (2013)

[Maaløe 16] Maaløe, L., Sønderby, C. K., Sønderby, S. K., and Winther, O.: Auxiliary deep generative models, *arXiv preprint arXiv:1602.05473* (2016)

[Mansimov 15] Mansimov, E., Parisotto, E., Ba, J. L., and Salakhutdinov, R.: Generating images from captions with attention, *arXiv preprint arXiv:1511.02793* (2015)

[McCulloch 43] McCulloch, W. S. and Pitts, W.: A logical calculus of the ideas immanent in nervous activity, *The bulletin of mathematical biophysics*, Vol. 5, No. 4, pp. 115–133 (1943)

[Mescheder 17] Mescheder, L., Nowozin, S., and Geiger, A.: Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks, *arXiv preprint arXiv:1701.04722* (2017)

[Mirza 14] Mirza, M. and Osindero, S.: Conditional generative adversarial nets, *arXiv*

preprint arXiv:1411.1784 (2014)

[Mohamed 16] Mohamed, S. and Lakshminarayanan, B.: Learning in implicit generative models, *arXiv preprint arXiv:1610.03483* (2016)

[Monay 03] Monay, F. and Gatica-Perez, D.: On image auto-annotation with latent space models, in *Proceedings of the eleventh ACM international conference on Multimedia*, pp. 275–278ACM (2003)

[Murphy 12] Murphy, K. P.: *Machine Learning: A Probabilistic Perspective*, The MIT Press (2012)

[Naesseth 16] Naesseth, C. A., Ruiz, F. J., Linderman, S. W., and Blei, D. M.: Rejection Sampling Variational Inference, *arXiv preprint arXiv:1610.05683* (2016)

[Neal 11] Neal, R. M., et al.: MCMC using Hamiltonian dynamics, *Handbook of Markov Chain Monte Carlo*, Vol. 2, No. 11 (2011)

[Ngiam 11] Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. Y.: Multimodal deep learning, in *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 689–696 (2011)

[Nikolopoulos 13] Nikolopoulos, S., Zafeiriou, S., Patras, I., and Kompatsiaris, I.: High order pLSA for indexing tagged images, *Signal Processing*, Vol. 93, No. 8, pp. 2212–2228 (2013)

[Oord 16] Oord, van den A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al.: Conditional image generation with pixelcnn decoders, in *Advances in Neural Information Processing Systems*, pp. 4790–4798 (2016)

[Ouyang 14] Ouyang, W., Chu, X., and Wang, X.: Multi-source deep learning for human pose estimation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2329–2336 (2014)

[Pan 10] Pan, S. J. and Yang, Q.: A survey on transfer learning, *IEEE Transactions on knowledge and data engineering*, Vol. 22, No. 10, pp. 1345–1359 (2010)

[Pandey 16] Pandey, G. and Dukkipati, A.: Variational methods for Conditional Multimodal Learning: Generating Human Faces from Attributes, *arXiv preprint*

arXiv:1603.01801 (2016)

[Pang 15] Pang, L., Zhu, S., and Ngo, C.-W.: Deep multimodal learning for affective analysis and retrieval, *IEEE Transactions on Multimedia*, Vol. 17, No. 11, pp. 2008–2020 (2015)

[Parikh 11a] Parikh, D. and Grauman, K.: Interactively building a discriminative vocabulary of nameable attributes, in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1681–1688IEEE (2011)

[Parikh 11b] Parikh, D. and Grauman, K.: Relative attributes, in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 503–510IEEE (2011)

[Platt 99] Platt, J., et al.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, *Advances in large margin classifiers*, Vol. 10, No. 3, pp. 61–74 (1999)

[Radford 15] Radford, A., Metz, L., and Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks, *arXiv preprint arXiv:1511.06434* (2015)

[Rasiwasia 10] Rasiwasia, N., Costa Pereira, J., Coviello, E., Doyle, G., Lanckriet, G. R., Levy, R., and Vasconcelos, N.: A new approach to cross-modal multimedia retrieval, in *Proceedings of the 18th ACM international conference on Multimedia*, pp. 251–260ACM (2010)

[Reed 16] Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H.: Generative adversarial text to image synthesis, *arXiv preprint arXiv:1605.05396* (2016)

[Rezende 14] Rezende, D. J., Mohamed, S., and Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models, *arXiv preprint arXiv:1401.4082* (2014)

[Rezende 16] Rezende, D., Danihelka, I., Gregor, K., Wierstra, D., et al.: One-shot generalization in deep generative models, in *International Conference on Machine Learning*, pp. 1521–1529 (2016)

[Robert 05] Robert, C. P. and Casella, G.: *Monte Carlo Statistical Methods (Springer*

Texts in Statistics), Springer-Verlag New York, Inc., Secaucus, NJ, USA (2005)

[Rohrbach 10] Rohrbach, M., Stark, M., Szarvas, G., Gurevych, I., and Schiele, B.: What helps where–and why? semantic relatedness for knowledge transfer, in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 910–917IEEE (2010)

[Rohrbach 11] Rohrbach, M., Stark, M., and Schiele, B.: Evaluating knowledge transfer and zero-shot learning in a large-scale setting, in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1641–1648IEEE (2011)

[Rumelhart 88] Rumelhart, D. E., Hinton, G. E., Williams, R. J., et al.: Learning representations by back-propagating errors, *Cognitive modeling*, Vol. 5, No. 3, p. 1 (1988)

[Saenko 10] Saenko, K., Kulis, B., Fritz, M., and Darrell, T.: Adapting visual category models to new domains, in *Proceedings of the European Conference on Computer Vision (ECCV)*, Vol. 6314, pp. 213–226 (2010)

[Salakhutdinov 09] Salakhutdinov, R. and Hinton, G.: Deep boltzmann machines, in *Artificial Intelligence and Statistics*, pp. 448–455 (2009)

[Salimans 16] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X.: Improved techniques for training gans, in *Advances in Neural Information Processing Systems*, pp. 2226–2234 (2016)

[Salvatier 16] Salvatier, J., Wiecki, T. V., and Fonnesbeck, C.: Probabilistic programming in Python using PyMC3., *PeerJ Computer Science*, Vol. 2, p. e55 (2016)

[Sharmanska 12] Sharmanska, V., Quadrianto, N., and Lampert, C. H.: Augmented attribute representations, in *European Conference on Computer Vision*, pp. 242–255Springer (2012)

[Shechtman 07] Shechtman, E. and Irani, M.: Matching local self-similarities across images and videos, in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pp. 1–8IEEE (2007)

[Shi 10] Shi, X., Liu, Q., Fan, W., Philip, S. Y., and Zhu, R.: Transfer learning on heterogenous feature spaces via spectral transformation, in *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pp. 1049–1054IEEE (2010)

[Simonyan 14] Simonyan, K. and Zisserman, A.: Very deep convolutional networks for large-scale image recognition, *arXiv preprint arXiv:1409.1556* (2014)

[Socher 13] Socher, R., Ganjoo, M., Manning, C. D., and Ng, A.: Zero-shot learning through cross-modal transfer, in *Advances in neural information processing systems*, pp. 935–943 (2013)

[Socher 14] Socher, R., Karpathy, A., Le, Q. V., Manning, C. D., and Ng, A. Y.: Grounded compositional semantics for finding and describing images with sentences, *Transactions of the Association for Computational Linguistics*, Vol. 2, pp. 207–218 (2014)

[Sohn 14] Sohn, K., Shang, W., and Lee, H.: Improved multimodal deep learning with variation of information, in *Advances in Neural Information Processing Systems*, pp. 2141–2149 (2014)

[Sohn 15] Sohn, K., Lee, H., and Yan, X.: Learning structured output representation using deep conditional generative models, in *Advances in Neural Information Processing Systems*, pp. 3483–3491 (2015)

[Sønderby 16] Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., and Winther, O.: Ladder variational autoencoders, in *Advances in Neural Information Processing Systems*, pp. 3738–3746 (2016)

[Srivastava 12] Srivastava, N. and Salakhutdinov, R. R.: Multimodal learning with deep boltzmann machines, in *Advances in neural information processing systems (NIPS)*, pp. 2222–2230 (2012)

[Srivastava 14] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting, *The Journal of Machine Learning Research*, Vol. 15, No. 1, pp. 1929–1958 (2014)

[Suzuki 14a] Suzuki, M., Sato, H., Oyama, S., and Kurihara, M.: Image classification by transfer learning based on the predictive ability of each attribute, in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, Vol. 1 (2014)

[Suzuki 14b] Suzuki, M., Sato, H., Oyama, S., and Kurihara, M.: Transfer learning based on the observation probability of each attribute, in *Systems, Man and Cybernetics*

(SMC), 2014 IEEE International Conference on, pp. 3627–3631IEEE (2014)

[Suzuki 16] Suzuki, M., Nakayama, K., and Matsuo, Y.: Joint Multimodal Learning with Deep Generative Models, *arXiv preprint arXiv:1611.01891* (2016)

[Theano Development Team 16] Theano Development Team, : Theano: A Python framework for fast computation of mathematical expressions, *arXiv e-prints*, Vol. abs/1605.02688, (2016)

[Tieleman 12] Tieleman, T. and Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, *COURSERA: Neural networks for machine learning*, Vol. 4, No. 2, pp. 26–31 (2012)

[Tran 17a] Tran, D., Hoffman, M. D., Saurous, R. A., Brevdo, E., Murphy, K., and Blei, D. M.: Deep probabilistic programming, *arXiv preprint arXiv:1701.03757* (2017)

[Tran 17b] Tran, D., Ranganath, R., and Blei, D. M.: Deep and Hierarchical Implicit Models, *arXiv preprint arXiv:1702.08896* (2017)

[Uehara 16] Uehara, M., Sato, I., Suzuki, M., Nakayama, K., and Matsuo, Y.: Generative Adversarial Nets from a Density Ratio Estimation Perspective, *arXiv preprint arXiv:1610.02920* (2016)

[Van De Sande 10] Van De Sande, K., Gevers, T., and Snoek, C.: Evaluating color descriptors for object and scene recognition, *IEEE transactions on pattern analysis and machine intelligence*, Vol. 32, No. 9, pp. 1582–1596 (2010)

[Vedantam 17] Vedantam, R., Fischer, I., Huang, J., and Murphy, K.: Generative Models of Visually Grounded Imagination, *arXiv preprint arXiv:1705.10762* (2017)

[Venugopalan 14] Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., and Saenko, K.: Translating videos to natural language using deep recurrent neural networks, *arXiv preprint arXiv:1412.4729* (2014)

[Wang 16] Wang, W., Yan, X., Lee, H., and Livescu, K.: Deep variational canonical correlation analysis, *arXiv preprint arXiv:1610.03454* (2016)

[Xie 14] Xie, W., Lu, Z., Peng, Y., and Xiao, J.: Graph-based multimodal semi-supervised image classification, *Neurocomputing*, Vol. 138, pp. 167–179 (2014)

[Xingjian 15] Xingjian, S., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-c.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting, in *Advances in neural information processing systems*, pp. 802–810 (2015)

[Xu 15] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention, in *International Conference on Machine Learning*, pp. 2048–2057 (2015)

[Yan 15] Yan, X., Yang, J., Sohn, K., and Lee, H.: Attribute2Image: Conditional Image Generation from Visual Attributes, *arXiv preprint arXiv:1512.00570* (2015)

[Yu 10] Yu, X. and Aloimonos, Y.: Attribute-based transfer learning for object categorization with zero/one training example, in *European conference on computer vision*, pp. 127–140Springer (2010)

[Zeiler 12] Zeiler, M. D.: ADADELTA: an adaptive learning rate method, *arXiv preprint arXiv:1212.5701* (2012)

[Zhou 14] Zhou, J. T., Pan, S. J., Tsang, I. W., and Yan, Y.: Hybrid Heterogeneous Transfer Learning through Deep Learning., in *AAAI*, pp. 2213–2220 (2014)

[Zhu 11] Zhu, Y., Chen, Y., Lu, Z., Pan, S. J., Xue, G.-R., Yu, Y., and Yang, Q.: Heterogeneous Transfer Learning for Image Classification., in *AAAI* (2011)

[Zhu 17] Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A.: Unpaired image-to-image translation using cycle-consistent adversarial networks, *arXiv preprint arXiv:1703.10593* (2017)

[宮川 97] 宮川 雅巳：グラフィカルモデリング (統計ライブラリー), 朝倉書店 (1997)

[神嶌 10] 神嶌敏弘：転移学習, 人工知能学会誌, Vol. 25, No. 4, pp. 572–580 (2010)

[総務 17] 総務省：平成２９年版情報通信白書 (2017)

[塚原 84] 塚原仲晃：脳の情報処理, 朝倉書店 (1984)

[麻生 15] 麻生 英樹, 安田 宗樹, 前田 新一, 岡野原 大輔, 岡谷 貴之, 久保 陽太郎, ボレガラ ダヌシカ：深層学習 Deep Learning (監修:人工知能学会), 近代科学社 (2015)

[鈴木 17] 鈴木 雅大, 松尾 豊：半教師ありマルチモーダル学習のための深層生成モデル, 2017年度 人工知能学会全国大会（第 31 回） (2017)