Московский государственный университет имени М.В. Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра математических методов прогнозирования

Иванов Олег Юрьевич

Множественное восстановление пропущенных данных с помощью глубинных нейробайесовских моделей

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Научный руководитель:

Д. А. Кропотов

Научные консультанты:

М. В. Фигурнов, Д. П. Ветров

Москва, 2018

Содержание

Аннотация

В данной работе рассматривается задача множественного восстановления пропущенных данных. Предполагается, что данные могут быть как вещественными, так и категориальными. Для решения задачи предлагается байесовская вероятностная модель, использующая глубинные искусственные нейронные сети. Предложенная модель является обобщением одной из наиболее популярных на сегодняшний день генеративных моделей — вариационного автокодировщика. Для обучения модели в работе выводится вариационная нижняя оценка на функционал качества, которая затем максимизируется с помощью метода дважды стохастического градиентного подъема. Проводится экспериментальное исследование предложенной модели. Оно показывает, что модель применима к задаче множественного восстановления пропущенных данных, а полученные с ее помощью восстановления разнообразны, реалистичны и могут быть использованы как для улучшения качества работы других методов обучения с учителем, так и для решения задачи дорисовки изображений.

1 Введение

Машинное обучение — это наука об автоматическом поиске закономерностей в данных. В большинстве задач машинного обучения класс интересующих пользователя зависимостей известен до того, как модель будет построена, то есть при построении модели используется информация о том, зависимости между какими объектами должны быть найдены. В этой работе рассматривается модель, в которой объекты искомых зависимостей не фиксированы на этапе обучения. Изучаемая модель находит зависимости между всеми подмножествами переменных в случайном векторе фиксированной длины. Модель выучивает распределение на этот вектор, а затем может обуславливаться на произвольный набор его координат и восстанавливать условное распределение на оставшиеся компоненты. Модель строит условное распределение не в явном виде, а в виде возможности генерировать объекты из него и оценивать его плотность для данных значений пропущенных признаков. Достоинством предлагаемой модели является отсутствие ограничений на моделируемое распределение. Чтобы подчеркнуть возможность модели обуславливаться на любое подмножество переменных, было решено назвать её универсальным обуславливателем.

Помимо теоретической ценности, модель имеет ряд практических применений. Основным применением модели является множественное восстановление пропущенных данных. Пропущенные данные могут быть как ценны сами по себе, так и использоваться в качестве промежуточного этапа для применения других алгоритмов. Примером задачи, в которой восстановленные данные ценны сами по себе, является дорисовка изображений [13]. В этой задаче часть пикселей изображения считаются пропущенными и должны быть реалистично заменены искусственно сгенерированными восстановлениями. Такой алгоритм можно использовать для удаления ненужных объектов с изображений или, наоборот, для дополнения частично закрытого или поврежденного объекта. Примером задачи, где восстановленные признаки полезны для другого алгоритма обучения, является любая задача обучения с учителем на частично отсутствующих данных.

Для решения задачи в работе предлагается вероятностная графическая модель, основанная на глубоких нейронных сетях. Это генеративная модель со скрытыми переменными, которые обусловлена на наблюдаемые переменные. Наблюдаемые переменные задают априорное распределение на скрытые гауссовские переменные, которые в свою очередь используются для генерации всего объекта. Для настройки

параметров предложенной модели в данной работе выводится вариационная нижняя оценка на логарифм правдоподобия этой модели. Полученная оценка затем максимизируется с помощью метода дважды стохастического градиентного подъема. Также в работе предлагаются и изучаются теоретически и экспериментально различные модификации модели, позволяющие ускорить процесс обучения, сделать его более стабильным или расширить область применимости модели.

Статья организована следующим образом.

Радел 1 является введением в работу.

В разделе 2 производится обзор существующих методов решения задачи восстановления пропущенных признаков, дорисовки изображений, а также построения генеративных вероятностных моделей.

В разделе 3 описывается теоретическая сторона модели. В нем вводятся формальные обозначения, связанные с задачей, формальная постановка задачи. Описывается генеративная вероятностная модель, вариационная нижняя оценка и алгоритм обучения для неё. Также в разделе описываются различные модификации модели, например, эвристика гибридной модели, предложенная в [?], и, как утверждают авторы, позволяющая улучшить правдоподобие восстанавливаемых распределений, расширение модели для работы с пропущенными признаками в обучающей выборке, дополнительное априорное распределение на параметры априорного распределения в скрытом пространстве (это позволяет устранить возможность расходимости модели при некоторых входных данных). Также в этом разделе описывается оптимизационный трюк, позволяющий находить более хороший локальный оптимум в процессе обучения.

Экспериментальное исследование модели на разных задачах и наборах данных в разделе 4 показывает, что она успешно восстанавливает плотность условных распределений и генерирует объекты из них. Особенно выражены преимущества модели в случаях, когда моделируемое условное распределение имеет несколько локальных оптимумов.

В подразделе 4.1 изучается поведение модели на искусственных данных, для которых истинное условное распределение может быть вычислено аналитически.

В подразделе 4.2 показывается, что восстанавливаемые моделью пропущенные признаки позволяют повысить качество решения задач обучения с учителем на выборках с пропущенными данными из коллекции наборов данных UCI [19]. В этом подразделе предложенная модель сравнивается с несколькими популярными методами множественного восстановления пропущенных признаков.

В подразделе 4.3 демонстрируется, что модель может генерировать разнообразные и реалистичные обусловленные изображения для наборов данных MNIST [18], Omniglot [17] и CelebA [7].

В разделе 5 анализируются полученные результатыи предлагаются направления дальнейших исследований по поставленной задаче.

Раздел 6 завершает работу.

1.1 Актуальность

Задача восстановления пропущенных данных часто встречается в машинном обучении. Несмотря на то, что большинство популярных методов не предполагают наличие пропусков в данных, работа с неполными данными особенно часто возникает при применении алгоритмов машинного обучения на практике в индустриальных компаниях различного масштаба. Неполные данные возникают в силу организационных причин, например, если необходимость сбора и хранения некоторых данных была осознана в компании слишком поздно. Также пропуски в данных могут быть вызваны техническими причинами, например, несовершенством архитектуры обрабатывающего информацию программного обеспечения, ведущем к невозможности сбора необходимой информации постоянно или в определенные моменты времени, временным отказом датчиков, а также повреждением хранилища данных. Не следует также исключать принципиальную невозможность сбора некоторой информации — например, если пользователь предпочел не указывать свой возраст или уровень дохода, то точно восстановить это на основе имеющихся у компании данных невозможно. Также пропущенные данные являются скорее правилом, чем исключением в обработке медицинских данных, поскольку во-первых число различных анализов и исследований слишком велико и пациент обычно проходит через лишь минимально необходимый для постановки диагноза и назначения лечения набор из них, а во-вторых регулярно появляются новые приборы и методики измерения, значения которых не могут быть получены для давних историй болезни. Данные могут быть пропущены и по другим причинам, например, в задаче дорисовки изображений часть пикселей могут специально быть помеченными как вырезанные изображения с целью убрать какой-то нежелательный объект на фотографии и реалистично дорисовать освободившуюся часть изображения.

Таким образом, задача восстановления пропущенных данных имеет не только теоретическую ценность, но также и большую практическую значимость.

2 Обзор литературы

Этот раздел организована следующим образом: сначала проводится обзор работ по восстановлению пропущенных данных. Затем рассматриваются наиболее современные методы дорисовки изображений, что является частным случаем восстановления пропущенных данных, для которого, однако, есть множество решений, учитывающих специфику работы с изображениями. В заключении раздела будут рассмотрены генеративные вероятностные модели, на основе которых будет строиться предлагаемая в этой работе модель.

2.1 Восстановление пропущенных данных

Задача обработки пропущенных данных традиционно относится к технической работе по подготовке выборки к использованию методов машинного обучения. Существуют ряд техник для работы с пропущенными значениями.

Наиболее распространенной является замена пропущенных значений специальным значением. Специальное значение может выбираться из разных соображений.

Например, можно использовать никогда не встречающееся в выборке значение, которое будет обозначать для методов машинного обучения пропущенное значение. Поскольку многие методы машинного обучения чувствительны к масштабу данных, в качестве такого значения можно использовать 0 или -1. Если 0 или -1 встречаются в данных, то можно дополнительно добавить к каждому объекту бинарную маску его пропущенных признаков, чтобы не терять информацию.

Также можно заменять пропущенные признаки их наиболее частым значением (в основном для категориальных признаков), средним, медианой или любой квантилью (для вещественных). В результате этой замены может происходить потеря информации по сравнению с исходным набором данных, однако часто такая предобработка приводит к улучшению качества построенной на получившихся данных модели.

Заметим, что замена специальным значением никак не учитывает зависимости между признаками одного объекта.

Существует ряд методов восстановления пропущенных данных с помощью разложений матрицы объектов-признаков [3], метода ближайших соседей [2], кластеризации [25], решения задачи регрессии на отсутствующие признаки. Все эти методы предлагают единственное восстановление пропущенных признаков, в то время как темой данной работы является множественное восстановление пропущенных признаков.

Следует отметить также то, что решающие деревья могут обрабатывать объекты с пропущенными признаками следующим образом [24]: если в текущей вершине дерева истинность условия не может быть вычислена, то для объекта рекурсивно вычисляется предсказание из обоих поддеревьев этой вершины, а затем усредняется по некоторому правилу — с равными весами, с весами, пропорциональными размеру поддеревьев или числу листьев в них, до которых дошел алгоритм, с весами, соответствующими априорной вероятности истинности условия, и так далее. К сожалению, этот метод может приводить к повышению вычислительной сложности алгоритма на этапе тестирования.

К методам множественного восстановления пропущенных признаков можно отнести обучение многомерной гауссианы на данных и использование аналитической формулы для вывода распределения на пропущенные признаки. Этот метод учитывает корреляции между признаками одного объекта, однако основывается на достаточно сильном предположении о распределении на данные. Также в методе требуется обращать квадратную матрицу размерности количества наблюдаемых признаков для генерации восстановлений. Это делает метод неприменимым на практике для задач с объектами большой размерности, например, для обработки изображений. Обобщением метода является обучение смеси гауссиан по данным с помощью EM-алгоритма [10]. Увеличение числа компонент смеси позволяет ослабить предположение о распределении данных, однако вычислительная сложность метода только возрастает линейно с ростом числа компонент, что делает его вычислительно неэффективным для объектов большой размерности.

2.2 Дорисовка изображений

Дорисовка изображений — классическая задача компьютерного зрения. Задача дорисовки изображений имеет ряд различных формулировок. Рассматриваемая в данной работе формулировка такова: некоторые пиксели изображения пропущены, и требуется реалистично восстановить их. Большинство ранних методов ее решения

основаны на локальной и текстурной информации или на созданных вручную для конкретной задачи признаках и эвристиках [13]. В последнее время для дорисовки изображений было предложено множество подходов, основанных на искусственных нейронных сетях.

В [6] используется полностью сверточный восстанавливающий автокодировщик (то есть автокодировщик, который восстанавливает ненаблюдаемые пиксели). Полностью сверточная архитектура приводит к отсутствию глобальных зависимостей на выходе автокодировщика. Для обеспечения декодера глобальной информацией поканально-полносвязный слой вставлен между декодером и выходом энкодера. Функция потерь декодера представляет собой сумму состязательного слагаемого и обычных потерь восстановления L_1 и L_2. Состязательное слагаемое заставляет восстановленную область быть менее размытой, поощряя сходимость к одной из мод.

В [22] авторы используют предварительно обученную глубокую сверточную генеративную модель для дорисовки изображения. Вход генеративной модели получается путем оптимизации скрытого представления изображения по сумме контекстного штрафа и штрафа восприятия. Контекстная функция штрафа представляет собой L_1-расстояние между наблюдаемыми пикселями и соответствующими им генерируемыми пикселями. Она показывает, насколько похоже сгенерированное изображение на наблюдаемую частью данного. Слагаемое штрафа восприятия основано на предварительно обученной дискриминантной модели и показывает, насколько реалистичным является сгенерированное изображение.

В [12] предлагается модель, полученная с помощью совместной оптимизации контекстного и текстурного штрафа. Пропущенные пиксели представляют собой квадрат фиксированного размера в центре изображения. Функция контекстного штрафа приводит к восстановлению глобальной структуры объекта, в то время как функция штрафа текстуры выравнивает текстуру внутри области пропущенных значений и за ее пределами. Для обученной модели дорисовка изображения выполняется путем оптимизации функции потерь по пропущенной части изображения.

В [21] сначала обучается генеративно-состязательная сеть по целым изображениям. Дорисовка состоит из процедуры оптимизации, которая находит скрытые переменные, которые лучше всего восстанавливают наблюдаемые признаки. Затем полученное скрытое представление проходит через генеративную модель для восстановления пропущенной части изображения.

Описанные методы дорисовки направлены на создание единственного реалистичного изображения, в то время как предлагаемая в этой работе модель способна генерировать разнообразные изображения. Кроме того, последние три модели имеют высокую вычислительную сложность на этапе применения обученной модели, потому что они требуют решения задачи оптимизации на генерации дорисовки. С другой стороны, предлагаемая далее в этой работе модель требует только одного прямого прохода по нейронной сети для генерации одного восстановления, и поэтому имеет существенно меньшую вычислительную сложность на этапе применения обученной модели.

2.3 Генеративные модели

В настоящее время наиболее популярными генеративными моделями являются генеративно-состязательные сети и вариационный автокодировщик.

Генеративно-состязательные сети [11] — популярная генеративная модель, способная создавать четкие и реалистичные изображения. К сожалению, эта модель имеет две проблемы, ограничивающие ее применимость.

Первая проблема — *схлопывание локальных оптимумов*: модель выучивает только подмножество возможных выходов [1, 26].

Вторая проблема заключается в том, что генеративно-состязательные сети работают только с вещественными объектами и поэтому не применимы к выборкам с категориальными признаками.

Из-за этих недостатков мы строим универсальный обуславливатель на основе вариационных автокодировщиков.

Вариационный автокодировщик [15] — это направленная генеративная модель со скрытыми представлениями для объектов. Генеративный процесс вариационного автокодировщика устроен следующим образом: сначала генерируется скрытое значение z из *априорного* распределения $p(z)$, и затем объект x генерируется из *генеративного* распределения $p_\theta(x|z)$, где θ — параметры генеративного распределения и генеративной модели. Этот процесс индуцирует распределение $p_\theta(x) = \mathbb{E}_{p(z)}\, p_\theta(x|z)$. Распределение $p_\theta(x|z)$ моделируется нейронной сетью с параметрами θ. $p(z)$ — стандартное нормальное распределение.

Параметры θ настраиваются с помощью максимизации правдоподобия обучающей выборки $\{x_i\}_{i=1}^N$ из истинного распределения на объектах $p_d(x)$:

$$\sum_{i=1}^{N} \log p_\theta(x_i) \to \max_\theta \tag{1}$$

Оптимиазционная задача (1) сложна из-за невозможности аналитически вычислить или быстро и точно численно приблизить $p_\theta(x)$. Несмотря на это, существует вариационная нижняя оценка на логарифм правдоподобия, которая может быть максимизирована с помощью метода обратного распространения ошибки и стохастического градиентного подъема:

$$\log p_\theta(x) = \mathop{\mathbb{E}}_{q_\phi(z|x)} \log \frac{p_\theta(x|z)p(z)}{q_\phi(z|x)} + \mathrm{KL}(q_\phi(z|x)\|p(z|x,\theta)) \geqslant$$

$$\geqslant \mathop{\mathbb{E}}_{q_\phi(z|x)} \log p_\theta(x|z) - \mathrm{KL}(q_\phi(z|x)\|p(z)) = L_{VAE}(x;\theta,\phi) \to \max_{\theta,\phi} \tag{2}$$

Здесь $q_\phi(z|x)$ — *предложное* распределение, заданное с помощью нейронной сети с параметрами ϕ. Оно приближает апостериорное распределение $p(z|x,\theta)$. Обычно в качестве предложного распределения используется гауссиана с диагональной матрицей ковариации. Чем ближе $q_\phi(z|x)$ к $p(z|x,\theta)$, тем точнее вариационная нижняя оценка $L_{VAE}(\theta,\phi)$. Чтобы вычислить градиент вариационной нижней оценки по ϕ, используется репараметризация математического ожидания: $z = \mu_\phi(x) + \varepsilon\sigma_\phi(x)$, где $\varepsilon \sim \mathcal{N}(0,E)$, E — единичная матрица, а μ_ϕ и σ_ϕ — детерминированные функции, задаваемые нейронными сетями. Таким образом, градиент для объекта может быть вычислен с использованием оценки Монте-Карло на первое слагаемое и аналитически для второго слагаемого:

$$\frac{\partial L_{VAE}(x;\theta,\phi)}{\partial\phi} = \mathop{\mathbb{E}}_{\varepsilon\sim\mathcal{N}(0,E)} \frac{\partial}{\partial\phi} \log p_\theta(x|\mu_\phi(x) + \varepsilon\sigma_\phi(x)) - \frac{\partial L}{\partial\phi} \mathrm{KL}(q_\phi(z|x)\|p(z)) \tag{3}$$

$$\frac{\partial L_{VAE}(x;\theta,\phi)}{\partial\theta} = \mathop{\mathbb{E}}_{\varepsilon\sim\mathcal{N}(0,E)} \frac{\partial}{\partial\theta} \log p_\theta(x|\mu_\phi(x) + \varepsilon\sigma_\phi(x)) \tag{4}$$

Таким образом, $L_{VAE}(\theta,\phi)$ может быть максимизировано с помощью стохастического градиентного подъема по параметрам ϕ и θ.

Обусловленный вариационный автокодировщик [23] приближает плотность условного распределения $p_d(x|y)$. Он существенно превосходит детерминированные модели когда распределение $p_d(x|y)$ имеет много локальных оптимумов, то есть различные x могут быть вероятны для данного y. Например, предположим, что x — вещественный случайный вектор. Тогда детерминированная регрессионная модель со

среднеквадратичной функцией потерь будет всегда возвращать условное среднее x. Для изображений это приводит к сильно размытым результатам. Однако обусловленный вариационный автокодировщик выучивает распределение над x, из которого можно генерировать различные и реалистичные объекты.

Вариационная нижняя оценка для обусловленного вариационного автокодировщика может быть выведена аналогично оценке вариационного автокодировщика. Различие в том, что все распределения теперь обуславливаются на y:

$$L_{CVAE}(x, y; \theta, \psi, \phi) = \mathop{\mathbb{E}}_{q_\phi(z|x,y)} \log p_\theta(x|z, y) - \mathrm{KL}(q_\phi(z|x, y)\|p_\psi(z|y)) \leqslant \log p_{\theta,\psi}(x|y) \quad (5)$$

Также как и для вариационного автокодировщика, эта функция потерь максимизируется с использованием репараметризации математического ожидания. Отметим, что априорное распредения $p_\psi(z|y)$ обусловлено на y и теперь также моделируется нейронной сетью с параметрами ψ. Таким образом, в обусловленном вариационном автокодировщике обучаются уже три нейронных сети, в то время как в вариационном автокодировщике только две.

Для оценки логарифма правдоподобия обусловленного вариационного автокодировщика, авторы используют два метода:

$$\log p_{\theta,\psi}(x|y) \approx \log \frac{1}{S} \sum_{i=1}^{S} p_\theta(x|z_i, y),\ z_i \sim p_\psi(z|y) \quad (6)$$

$$\log p_{\theta,\psi}(x|y) \approx \log \frac{1}{S} \sum_{i=1}^{S} \frac{p_\theta(x|z_i, y)p_\psi(z_i|y)}{q_\phi(z_i|y)},\ z_i \sim q_\phi(z|y) \quad (7)$$

Первая оценка называется *оценкой Мнте-Карло*, а вторая — *оценка выборкой по значимости*. Формально они эквивалентны, но на практике оценка Монте-Карло для вариационных автокодировщиков требует генерации большего числа скрытых представлений объекта, чтобы достигнуть такой же точности оценки. Маленькое S ведет к недооценке логарифма правдоподобия для обоих оценок [4], хотя для оценки Монте-Карло этот эффект выражен намного более ярко.

В процессе обучения модели для генерации скрытых представлений z используется предложное распределение $q_\phi(z|x, y)$, а на этапе тестирования — априорное распределение $p_\psi(z|y)$. KL-дивергенция штрафует сильное различие этих распределений, но, как утверждают авторы обусловленного вариационного автокодировщика, этого недостаточно. Скрытое представление из $p_\psi(z|y)$ может существенно отличаться от всех скрытых представлений из $q_\phi(z|x, y)$, которые подавались на вход генеративной сети на этапе обучения, поэтому на этапе теста генеративная сеть может работать

некорректно. Чтобы устранить этот недостаток, авторы предлагают *гибридную модель* (9), которая является взвешенной смесью вариационной нижней оценки (5) на логарифм правдоподобия и оценки логарифма правдоподобия с помощью метода Монте-Карло с генерацией одного скрытого представления на объект (8). Модель, соответствующая второму слагаемому, называется *гауссовской стохастической нейронной сетью* (8), потому что она может быть реализована с помощью одной нейронной сети с одним гауссовким стохастическим слоем посередине:

$$L_{GSNN}(x, y; \theta, \psi) = \mathop{\mathbb{E}}_{p_\psi(z|y)} \log p_\theta(x|z, y) \tag{8}$$

$$L(\theta, \psi, \phi) = \alpha L_{CVAE}(\theta, \psi, \phi) + (1 - \alpha) L_{GSNN}(\theta, \psi) \quad \alpha \in [0, 1] \tag{9}$$

Как будет указано в разделе 3.4, мы обнаружили существенный недостаток этой модели, который описывается в разделе 4.1.

3 Универсальный обуславливатель

3.1 Формальная постановка задачи

Рассматривается распределение $p_d(x)$ над D-мерным вектором x с вещественными или категориальными компонентами. Компоненты вектора называются *признаками*.

Пусть $U = \{1, \ldots, D\}$ — множество индексов всех признаков, а $I \subseteq U$ — множество индексов *пропущенных признаков*. Тогда $x_I = \{x_{I_1}, \ldots, x_{I_{|I|}}\}$ — вектор пропущенных признаков, а $x_{U \setminus I}$ — вектор *наблюдаемых признаков* объекта.

Целью данной работы является построить вероятностную модель распределения $p(x_I | x_{U \setminus I}, I)$ для произвольного I.

Однако истинное распределение $p_d(x_I | x_{U \setminus I}, I)$ не может быть идеально восстановлено без введения сильных предположений о виде распределения $p_d(x)$. Поэтому предлагаемая модель $p_\Theta(x_I | x_{U \setminus I}, I)$ с необходимостью будет более точной для некоторых I и менее точной для остальных. Здесь Θ — множество всех параметров используемых предлагаемой моделью на этапе тестирования.

Чтобы формализовать требования по точности модели для разных I вводится распределение над различными множествами пропущенных признаков $p(I)$. Распределение $p(I)$ произвольно и может быть задано в зависимости от решаемой задачи. Чем больше вероятность множества пропущенных значений I, тем лучше должна работать для него модель.

Таким образом, можно записать логарифм правдоподобия модели и оптимизировать его:

$$\mathop{\mathbb{E}}_{p_d(x)} \mathop{\mathbb{E}}_{p(I)} \log p_\Theta(x_I|x_{U\setminus I}, I) \to \max_\Theta \tag{10}$$

Заметим, что частными случаями предлагаемой модели (10) являются вариационный автокодировщик ($I = U$) и обусловленный вариационный автокодировщик (I — фиксированное множество).

3.2 Вариационная нижняя оценка

Проводим вывод вариационной нижней оценки для $\log p_\Theta(x_I|x_{U\setminus I}, I)$ таким же образом, как она выводилась для вариационного автокодировщика:

$$\log p_\Theta(x_I|x_{U\setminus I}, I) = \mathop{\mathbb{E}}_{q_\phi(z|x,I)} \log \frac{p_\Theta(x_I, z|x_{U\setminus I}, I)}{q_\phi(z|x, I)} + \mathrm{KL}(q_\phi(z|x, I)\|p(z|x, I, \Theta)) \geqslant$$

$$\geqslant \mathop{\mathbb{E}}_{q_\phi(z|x,I)} \log \frac{p_{\theta,\psi}(x_I, z|x_{U\setminus I}, I)}{q_\phi(z|x, I)} = \mathop{\mathbb{E}}_{q_\phi(z|x,I)} \log p_\theta(x_I|z, x_{U\setminus I}, I) - \mathrm{KL}(q_\phi(z|x, I)\|p_\psi(z|x_{U\setminus I}, I)) \tag{11}$$

В результате получаем следующую оптимизационную задачу:

$$\mathop{\mathbb{E}}_{p_d(x)} \mathop{\mathbb{E}}_{p(I)} \left[\mathop{\mathbb{E}}_{q_\phi(z|x,I)} \log p_\theta(x_I|z, x_{U\setminus I}, I) \; - \; \mathrm{KL}(q_\phi(z|x, I)\|p_\psi(z|x_{U\setminus I}, I)) \right] \to \max_{\theta,\psi,\phi} \tag{12}$$

3.3 Параметризация модели

Далее в этой работе рассматриваются только $z \in \mathbb{R}^d$, и нормальные распределения p_ψ и q_ϕ над z с диагональной матрицей ковариации.

Введем функции $\mu_\psi(x_{U\setminus I}, I)$, $\sigma_\psi(x_{U\setminus I}, I)$, параметризованные нейронными сетями с параметрами ψ и распределение $p_\psi(z|x_{U\setminus I}, I) = \mathcal{N}(z|\mu_\psi, \sigma^2_\psi)$. Распределение $q_\phi(z|x, I)$ и вещественнозначные компоненты распределения $p_\theta(x_I|z, x_{U\setminus I}, I)$ определяются аналогично.

Каждая категориальная компонента i распределения $p_\theta(x_I|z, x_{U\setminus I}, I)$ параметризуется функцией $w_{i,\theta}(z, x_{U\setminus I}, I)$, которая возвращает логарифмы вероятностей для каждой категории: $x_i \sim \mathrm{Cat}[\mathrm{Softmax}(w_{i,\theta}(z, x_{U\setminus I}, I))]$.

Компоненты скрытого представления z условно независимы при данных $x_{U\setminus I}$ и I, а компоненты x_I условно независимы при данных z, $x_{U\setminus I}$ и I.

Переменные I, x_I и $x_{U\setminus I}$ имеют переменную длину, которая зависит от I, поэтому не применимы такие архитектуры, как многослойная перцептрон и сверточная нейронная сеть. Далее описывается предлагаемое решение этой проблемы.

Пусть $b \in \{0,1\}^D$ — бинарная маска пропущенных признаков, то есть $b_i = 1$, если $i \in I$ и $b_i = 0$ в противном случае. Используя это обозначение и $\circ$ как поэлементное произведение, можно представить задачу оптимизации (12) следующим образом:

$$\log p_\theta(x_I|z, x \circ (1-b), b) = \sum_{i=1}^{D} b_i \log p_\theta(x_i|z, x \circ (1-b), b) \tag{13}$$

$$L_{UCM}(x, b; \theta, \psi, \phi) = \mathop{\mathbb{E}}_{q_\phi(z|x,b)} \log p_\theta(x_I|z, x \circ (1-b), b) \\ - \mathrm{KL}(q_\phi(z|x,b) \| p_\psi(z|x \circ (1-b), b)) \tag{14}$$

$$\mathop{\mathbb{E}}_{p_d(x)} \mathop{\mathbb{E}}_{p(b)} L_{UCM}(x, b; \theta, \psi, \phi) \to \max_{\theta,\psi,\phi} \tag{15}$$

Для этого представления распределений размер входов всегда фиксирован, поэтому стандартные архитектуры нейронных сетей теперь применимы. Нормальное распределение на скрытых представлениях позволяет выполнить репараметризацию математического ожидания по скрытым представлениям и вычислять KL-дивергенцию аналитически для максимизации (15).

3.4 Гибридная модель

Гибридная модель была предложена в [23]. Основная идея состоит в том, что выборки из $p_\psi(z|x \circ (1-b), b)$ не участвуют в процедуре обучения, но из этого распределения генерируются скрытые представления на стадии тестирования.

Такая несогласованность может приводить к некорректной работе генеративной сети. В [23] авторы предлагают использовать взвешенную смесь вариационной нижней оценки и оценки Монте-Карло логарифма правдоподобия. Модель, соответствующая последнему слагаемому, называется гауссовской стохастической нейронной сетью. В [23] авторы утверждают, что использование смеси моделей увеличивает логарифм правдоподобия на большинстве наборов данных.

Такой же прием применим и к нашей модели:

$$L_{GSNN}(x, b; \theta, \psi) = \mathop{\mathbb{E}}_{p_\psi(z|x\circ(1-b),b)} \log p_\theta(x_I|z, x \circ (1-b), b) \tag{16}$$

$$L_{hybrid}(\theta, \psi, \phi) = \alpha L_{UCM}(\theta, \psi, \phi) + (1-\alpha) L_{GSNN}(\theta, \psi), \quad \alpha \in [0,1] \tag{17}$$

В экспериментах на модельных и реальных данных в этой работе наблюдается, что использование гибридной модели приводит к лучшей оценке Монте-Карло логарифма правдоподобия, если количество генерируемых представлений в выборке Монте-Карло

фиксированное и небольшое, но к худшей оценке выборки по значимости (см. таб. 2 в подразделе 4.3).

Тем не менее, часто использование гибридной модели не является разумным. Эксперименты в разделе 4.1 показывают, что $\alpha \neq 1$ делает форму выучиваемого распределения ближе к унимодальной и более шумной. Поэтому, если истинные условные распределения имеют много локальных оптимумов и $\alpha \neq 1$, модель вряд ли выучит эти истинные распределения.

Основываясь на вышеупомянутых экспериментах, рекомендуется использовать эту модель с осторожностью или даже полностью исключить её использование, установив $\alpha = 1$, то есть оптимизируя только вариационную нижнюю оценку (15).

3.5 Работа с пропущенными данными в обучающей выборке

Оптимизируемая функция (15) требует наличия всех признаков каждого объекта на этапе обучения модели: некоторые признаки будут наблюдаемыми переменными на входе модели, а другие будут пропущенными признаками, используемыми для оценки модели.

Тем не менее, в некоторых постановках задачи обучающие данные также содержат пропущенные признаки. Модель универсального обуславливателя применима и в таких случаях. Для её применения предлагается следующая небольшая модификация задачи (15). Отсутствующие признаки, естественно, не могут быть поданы в нейронную сеть, поэтому для каждого объекта недостающие маски объектов b должны покрывать недостающие признаки объекта x. Это может быть достигнуто с помощью зависимости вероятности маски отсутствующих признаков от объекта: $p(b|x)$. Единственное требование для $p(b|x)$ — это $x_i = \omega \Rightarrow b_i = 1$, где ω обозначает пропущенное значение в данных. С другой стороны, нет разумного способа оценить $p(x_i = \omega | x_{U \setminus I}, I)$, потому что модель рассматривает только $x \in \mathbb{R}^D$. Поэтому в слагаемом потерь восстановления (13) не учитываются пропущенные признаки, что соответствует маргинализации по ним:

$$\log \int p_\theta(x_i|z, x \circ (1-b), b) dx_i = \log 1 = 0 \tag{18}$$

$$\log p_\theta(x_I|z, x \circ (1-b), b) = \sum_{i \in I:\ x_i \neq \omega} \log p_\theta(x_i|z, x \circ (1-b), b) \tag{19}$$

3.6 Априорное распределение в скрытом пространстве

При оптимизации функции потерь (15) параметры априорного распределения на z могут расти неограниченно, поскольку модель не запрещает и не штрафует большие значения этих параметров.

Мы наблюдаем рост $||z||_2$ во время обучения (иногда в течение первых нескольких эпох $||z||_2$ убывает и начинает возрастать после этого).

Чтобы предотвратить потенциальные численные неустойчивости, мы вводим гамма-нормальное априорное распределение на параметры априорного распределения модели, чтобы предотвратить расходимость метода. Формально мы переопределяем $p_\psi(z|x_{U\setminus I}, I)$ следующим образом:

$$p_\psi(z, \mu_\psi, \sigma_\psi|x_{U\setminus I}, I) = \mathcal{N}(z|\mu_\psi, \sigma_\psi^2)\mathcal{N}(\mu_\psi|0, \mu_0)\,\mathrm{Gamma}(\sigma_\psi|2, \sigma_0) \tag{20}$$

В экспериментах в качестве гиперпараметра μ_0 используется большое значение (10^4), а в качестве σ_0 — небольшое положительное число (10^{-4}). Таким образом, это распределение близко к равномерному около нуля, поэтому оно практически не влияет на процесс обучения, гарантируя при этом сходимость процесса обучения.

3.7 Оптимизация: коэффициент перед KL-дивергенцией

В экспериментах наблюдается, что модель может недообучаться и останавливаться в плохом локальном оптимуме: на ранней стадии обучения KL-дивергенция может мешать модели использовать скрытые переменные. Вместо этого модель пытается восстанавливать пропущенные значения основываясь только на наблюдаемых значениях, что приводит к тому, что модель обычно выдает усредненное восстановление пропущенных признаков, а не объект, сгенерированный из распределения на них.

Чтобы этого не произошло, мы используем мультипликативный коэффициент для KL-дивергенции, как это предлагается в [16].

В начале обучения коэффициент равен нулю или близок к нулю ($\approx 10^{-3}$), что дает время предложному и генеративному распределению для обучения хорошей реконструкции объектов. Во время обучения коэффициент KL-дивергенции линейно увеличивается до 1, а затем модель обучается с коэффициентом, равным 1, поэтому эта эвристика не изменяет оптимизируемую функцию (15).

Необходимость этой эвристики была обнаружена только при обучении на модельных многомодальных распределениях, но, например, для дорисовки изображений она не является обязательной.

4 Вычислительные эксперименты

4.1 Эксперименты на модельных данных

В этом разделе показывается, что модель может выучивать сложное распределение с большим числом локальных оптимумов на модельных данных. Пусть $x \in \mathbb{R}^2$ and $p(b_1 = 1) = p(b_2 = 1) = 0.5$. $p_d(x) = \frac{1}{8}\sum_{i=1}^{8}\mathcal{N}(x|\mu_i, \frac{1}{10}E)$, где $\mu_i \sim \mathcal{N}(\mu_i|0, E)$, а E — единичная матрица. Распределение $p_d(x)$ изображено на рисунке 1. Выборка содержит 100000 точек, сгенерированных из $p_d(x)$. Для решения задачи используется многослойный перцептрон с четыремя скрытыми слоями размера 400-200-100-50 и активациями ReLU, размерностью скрытого пространства 25. При обучении модель коэффициент при KL-дивергенции линейно растет с 0.001 до 1 с 16-ой эпохи до 48-ой из 80 эпох обучения.

Для различных коэффициентов гибридной смеси моделей α отображены сгенерированные точки из обученнго совместного распредления $p_\Theta(x_1, x_2)$ и условных распределений $p_\Theta(x_1|x_2)$ и $p_\Theta(x_2|x_1)$. Наблюдаемые признаки условных распределений сгенерированы из маргинальных распредлеений $p_d(x_2)$ and $p_d(x_1)$ соответственно.

Можно видеть, что при $\alpha = 1$ восстанавливает форму для каждого обусловленного распределения (рис. 2). Оценка правдоподобия методом Монте-Карло с генерацией 10 точек равна -85, а оценка методом выборки по значимости — -0.22. Использование $\alpha = 0.99$ приводит к лучшим оценкам логарифма правдоподобия методом Монте-Карло с генерацией 10 точек в скрытом пространстве — -11, но также и к худшей оценке методом выборки по значимости — -0.35. Одновременно это ведет к более сглаженному распределению, то есть генерируемые объекты содержат больше шума и находятся ближе к среднему по всем объектам (рис. 2). Наконец, при $\alpha = 0.9$ оценки Монте-Карло и выборки по значимости очень близки (-1.7 и -0.62 соответственно), однако полностью пропадает структура восстановленного распределения (рис. 2).

Таким образом, использование гибридной модели в самом деле существенно улучает оценку правдоподобия методом Монте-Карло с небольшим числом генерируемых точек в скрытом пространстве, однако уменьшает истинное правдоподобие модели, более точно оцениваемое методом выборки по значимости.

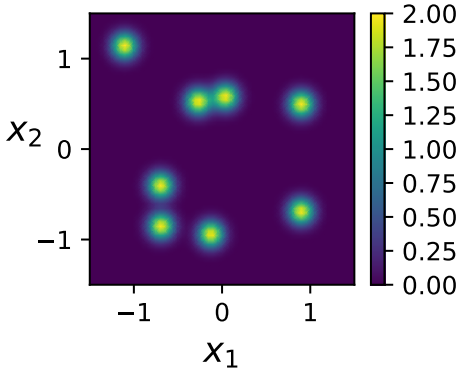

Рис. 1: Плостность распределения вероятности модельного распределения.

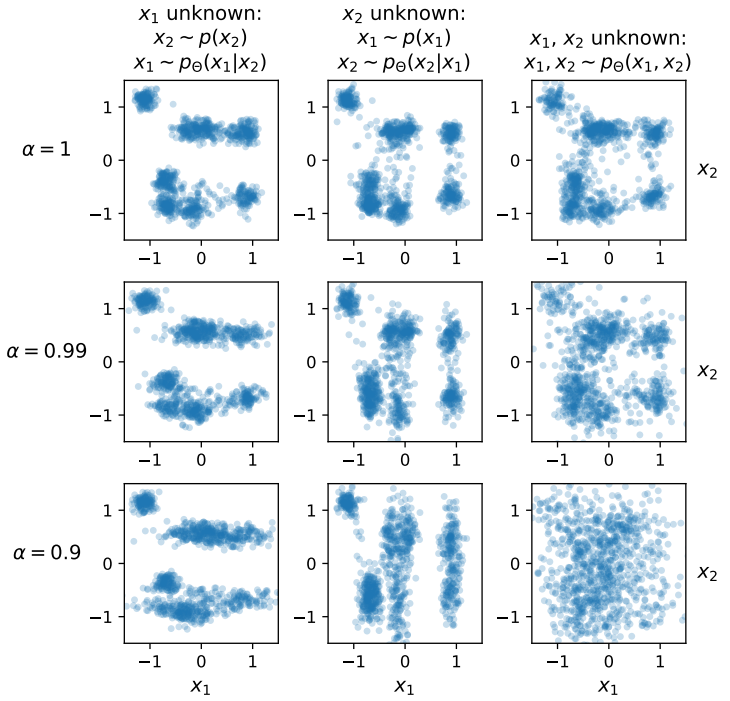

Рис. 2: Универсальный обуславливатель и гибридная модель на модельных данных.

Можно сделать вывод, что использование $\alpha \neq 1$ существенно усложняет восстановление распределений со многими локальными оптимумами, поэтому рекомендуется использовать $\alpha = 1$ или хотя бы $\alpha \approx 1$. В дальнейшем во всех экспериментах $\alpha = 1$.

4.2 Восстановление признаков в задаче обучения с учителем

Широко распространены выборки с пропущенными признаками. Рассмотрим выборку с D-мерными объектами x, где каждый признак может быть пропущен (что обозначается как $x_i = \omega$), и их целевыми значениями y. Большинство дискриминативных методов не поддерживают отсутствующие значения в объектах. Процедура заполнения пропущенных значений функций называется восстановлением пропущенных значений.

Стандартным алгоритмом является замена значений отсутствующих признаков объекта средним значением этого признака или некоторым специальным значением. Универсальный обуславливатель обеспечивает более мощный способ восстановления признаков. Он позволяет генерировать несколько разных восстановлений для каждого объекта из распределения по пропущенным признакам. В экспериментах каждый объект с пропущенными признаками заменяется $n = 10$ объектами со сгенерированными восстановлениями. Таким образом размер выборки увеличивается

в n раз. Затем классификатор или регрессор обучаются на полученной таким образом обучающей выборке. Для тестовой выборки алгоритм применяется к каждому из n восстановленных объектов, а предсказания затем усредняются в зависимости от функции потерь задачи обучения с учителем. Для среднеквадратичной ошибки оптимальным усреднением является просто среднее из всех прогнозов, для средней абсолютной ошибки — медиана, для точности в классификации — наиболее частный вариант ответа и так далее. В этой работе для регрессионных задач используется среднее значение прогнозов, а для классификации — наиболее частое.

Эксперименты показывают, что лучшие результаты достигаются, когда модель выучивает распредление на конкатенацию объектов x их целевых функций y. Рассмотрим модельный набор данных, где $x_1 = 1$, $x_2 \sim \mathcal{N}(x_2|y, 1)$, $p_d(y = 0) = p_d(y = 5) = 0,5$. В этом случае $p_d(x_2|x_1 = 1) = 0,5\mathcal{N}(x_2|0, 1) + 0,5\mathcal{N}(x_2|5, 1)$. Можно видеть, что генерация данных из $p_d(x_2|x_1)$ может только вводить в заблуждение классификатор, приводя к неправильным предсказаниям в половине случаев. С другой стороны, $p_d(x_2|x_1, y) = \mathcal{N}(x_2|y, 1)$. Заполнение пробелов с использованием $p_d(x_2|x_1, y)$ может только улучшить классификатор или регрессор, предоставив ему некоторую информацию из совместного распределения $p_d(x, y)$ и, таким образом, упрощая зависимость, которая должна быть найдена во время обучения. Поэтому y рассматривается как дополнительный признак, которая всегда пропущен во время тестирования. Предсказанные универсальным обуславливателем значения y не подаются в классификатор или регрессор.

Чтобы сохранить информацию о различии между восстановленным значением и наблюдаемым значением, можно расширить признаковое описание x бинарной маской пропущенных значений.

Для обучения предложенной модели используется распределение $p(b_i|x)$, в котором $p(b_i|x_i = \omega) = 1$ и $p(b_i|x) = 0,5$ в противном случае.

Для воспроизводимости выборка разделена на обучающую и тестовую так, чтобы каждый третий объект находился в тестовой выборке. Перед обучением производится удаление 50% случайных признаков как в обучающей, так и в тестовой выборке. Признаки удаляются независимо по объектам. Также вещественные признаки нормализуются. После этого обучается универсальный обуславливатель, гауссовкая стохастическая нейронная сеть и многомерная гауссиана. Признаки восстанавливаются с помощью метода подставновки среднего и вышеперечисленных алгоритмов. Для обучения используется Gradient Boosting Classifier или Regressor из бибилиотеки XG-

Таблица 1: Средний по тестовой выборке R2-score (задачи регрессии) или точность (задачи классификации) для сравниваемых методов. Чем больше, тем лучше. УО — Универсальный обуславливатель, ГСНС — гауссовская стохастическая нейронная сеть (частный случай УО), МГ — многомерная гауссиана.

Dataset	Среднее	XGBoost	**УО**	**ГСНС**	МГ
Boston	0.505 ± 0.061	0.502 ± 0.056	$\mathbf{0.577} \pm 0.069$	0.563 ± 0.069	0.564 ± 0.055
Concrete	0.452 ± 0.042	0.458 ± 0.040	$\mathbf{0.494} \pm 0.032$	0.453 ± 0.030	0.484 ± 0.030
CASP	0.840 ± 0.002	0.842 ± 0.002	$\mathbf{0.856} \pm 0.002$	$\mathbf{0.856} \pm 0.002$	0.850 ± 0.003
Wine	0.230 ± 0.012	0.236 ± 0.008	0.232 ± 0.016	$\mathbf{0.243} \pm 0.014$	0.238 ± 0.011
Yeast	0.423 ± 0.025	0.426 ± 0.026	$\mathbf{0.436} \pm 0.019$	0.419 ± 0.025	0.430 ± 0.019

Boost [5] с `max_depth` $= 5$. Также результаты работы на восстановленных признаках сравниваются со встроенным в XGBoost методом обработки пропущенных данных. Эта процедура повторяется 10 раз с различными пропущенными признаками, а затем вычисляется среднее и стандартное отклонение качества решения задачи обучения с учителем.

Как видно из таблицы 1, использование предложенной модели повышает качество работы классификатора или регрессора, которые были созданы для решения задачи обучения с учителем. В большинстве случаев использование универсального обуславливателя увеличивает качество работы классификатора или регрессора, обученного на восстановленных данных. В случаях, когда универсальный обуславливатель не превосходит гауссовскую стохастическую нейронную сеть, можно предположить, что условное распределение для обучения не является достаточно сложным и не имеет много локальных оптимумов, поэтому более простая модель обучается лучше.

4.3 Применение модели в задаче дорисовки изображений

Задача дорисовки изображений имеет ряд различных формулировок. Рассматриваемая в данной работе формулировка такова: некоторые пиксели изображения пропущены, и требуется реалистично восстановить их. В отличие от большинства статей, требуется восстановить не только одну наиболее вероятную или среднюю дорисовку, но и распределение по всем возможным дорисовкам, из которых пользователь может выбрать наиболее подходящую. Это распределение имеет очень много локальных

опимумов, потому что часто существует множество различных правдоподобных способов дорисовки изображения.

В отличие от предыдущего подраздела, в обучающей выборке для модели находятся неповрежденные изображения без пропущенных признаков, поэтому модификация из подраздела 3.5 не используется и $p(b|x) = p(b)$. Маска пропущенных пикселей обычно не задается распределением Бернулли, так как в этом случае наблюдаемые пиксели равномерно распределяются по изображению, и дорисовка становится больше похожей на задачу интерполяции, чем на структурное предсказание. Другая причина — отсутствие разумных практических приложений для дорисовок с такой маской.

Типичный случай в задаче дорисовки изображений — когда пропущенные пиксели образуют прямоугольник. Чтобы сделать задачу сложнее, чем обычная интерполяция, мы произвольно равновероятно генерируем угловые точки прямоугольников на изображении, но отклоняем те прямоугольники, площадь которых меньше четверти площади изображения.

Другим способом задания маски является выборка равновероятно горизонтальной линии некоторой ширины на изображении и восстановление остальной части изображения.

Как было изложено в подразделе 2.2, современные подходы обычно используют разные состязательные функции потерь для генерации более четких и реалистичных изображений. Модель универсального обуславливателя может быть адаптирована к задаче дорисовки изображений с помощью использования взвешенной суммы этих состязательных, текстурных и других функций потерь как части функции ошибки восстановления $p_\theta(x_I|z, x \circ (1-b), b)$. Тем не менее, такие построения выходят за рамки этого исследования, поэтому они оставлены для будущей работы. В настоящей работе мы показываем, что даже без них модель может генерировать как разнообразные, так и реалистичные дорисовки.

Таблица 2 показывает, что универсальный обуславливатель лучше выучивает условные распределения на различные дорисовки, чем другие модели.

Во всех экспериментах использовался метод оптимизации Adam [14] и короткие связи между априорной и генеративной нейронными сетями, предложенные в [20] и [16].

MNIST [18] — это выборка из 60000 обучающих 10000 тестовых черно-белых картинок цифр от 0 до 9 размера 28x28. Для MNIST используется правдоподобие распределения Бернулли как функция потерь восстановления: $\log p_\theta(x_I|z, x \circ (1-b), b) = \sum_{i \in I} \log \text{Bernoulli}(x_i|p_{\theta,i}(z, x \circ (1-b), b))$, где $p_{\theta,i}(z, x \circ (1-b), b)$ — выход генеративной

Таблица 2: Средний отрицательный логарифм правдоподобия дорисовок для 1000 объектов. ВЗ-S обозначает оценку лографима правдоподобия с помощью выборки по значимости с генерацией S скрытых представлений для каждого объекта (7). МК-S обозначает оценку лографима правдоподобия с помощью метода Монте-Карло с генерацией S скрытых представлений для каждого объекта (6). Стандартное отклонение оценено с помощью бутстрапа по объектам. Наивный Байессовский регрессор — стандартный метод, предплагающий независимость пикселей.

Метод	MNIST	Omniglot	CelebA
Универсальный обуславливатель ВЗ-10^2	$\mathbf{83} \pm 2$	$\mathbf{275} \pm 17$	$\mathbf{34035} \pm 1609$
Универсальный обуславливатель МК-10^4	98 ± 4	1452 ± 109	41513 ± 2163
Универсальный обуславливатель МК-10^2	135 ± 6	2203 ± 150	53904 ± 3121
Гауссовская стохастическая нейросеть МК-10^4	139 ± 3	1199 ± 62	53427 ± 2208
Гауссовская стохастическая нейросеть МК-10^2	139 ± 3	1200 ± 62	53486 ± 2210
Наивный Байесовский регрессор	205	2490	269480

нейронной сети. Это стандартная практика для черно-белых изображений, используемая например в [9]. Наблюдаемые пискели образуют горизонтальную линию толщиной 3 пикселя. Используется сверточная нейронная сеть с MaxPooling и размерностью скрытого пространства 8. Примеры генерируемых дорисовок приведены на рисунке 3.

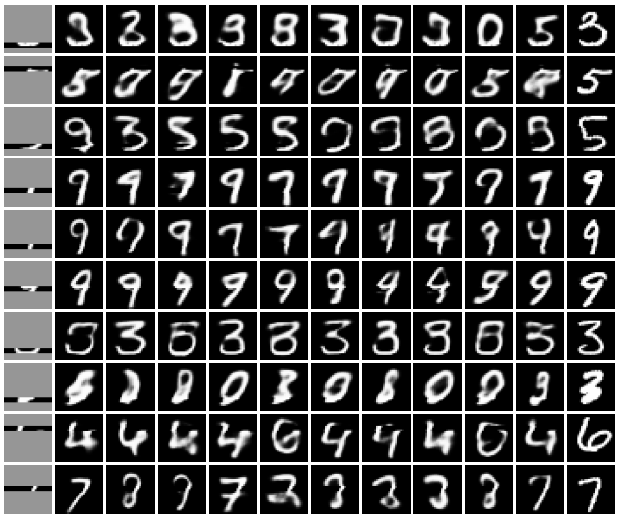

Рис. 3: Дорисовки для MNIST. Левый столбец: входное значение. Серый цвет обозначает пропущенные данные. Середина: объекты, сгенерированные универсальным обуславливателем. Правый столбец: истинное изображение.

Omniglot [17] — выборка из 19280 обучающих и 13180 тестовых черно-белых изображений символов из различных алфавитов размером 105x105. Как и в предыдущем эксперименте, яркость каждого пикселя интерпретируется как вероятность распределения Бернулли. Используемая маска — случайный прямоугольник — описана выше. Используемые архитектуры нейронных сетей основаны на сверточных ResNet блоках [8], AveragePooling и размерности скрытого пространства 64. Модель обучается 50 эпох, что занимает примерно 2.5 часа на графическом ускорителе GeForce GTX 1080 Ti. Примеры генерируемых дорисовок приведены на рисунке 4.

Рис. 4: Дорисовки для Omniglot. Левый столбец: входное значение. Серый цвет обозначает пропущенные данные. Середина: объекты, сгенерированные универсальным обуславливателем. Правый столбец: истинное изображение.

CelebA [7] — это выборка из 162770 обучающих, 19867 валидационных и 19962 тестовых цветных изображений лиц знаменитостей размера 178x218. До обучения мы нормируем каналы в выборке. Используемая маска — случайный прямоугольник — описана выше. Используемые архитектуры нейронных сетей основаны на сверточных ResNet блоках [8], AveragePooling и размерности скрытого пространства 32. Модель обучается 20 эпох, что занимает примерно 30 часов на графическом ускорителе GeForce GTX 1080 Ti. Примеры генерируемых дорисовок приведены на рисунке 5.

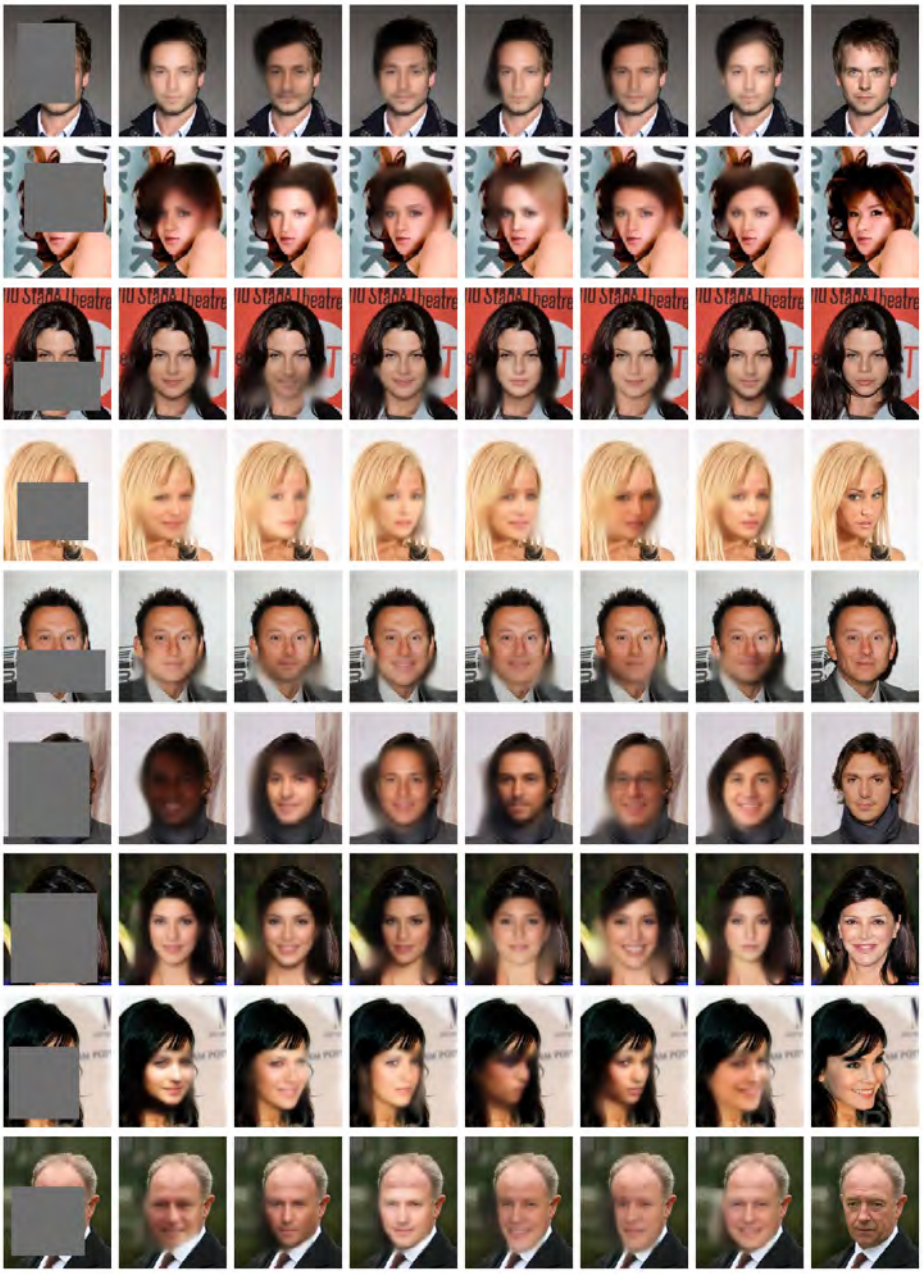

Рис. 5: Дорисовки для CelebA. Левый столбец: входное значение. Серый цвет обозначает пропущенные данные. Середина: объекты, сгенерированные универсальным обуславливателем. Правый столбец: истинное изображение.

5 Обсуждение и выводы

Эксперименты на модельных данных показывают, что использование в гибридной модели $\alpha \neq 1$ существенно усложняет восстановление распределений со многими локальными оптимумами, поэтому рекомендуется использовать $\alpha = 1$ или $\alpha \approx 1$.

Эксперимент по обучению с учителем показывает, что предложенный метод восстановления пропущенных данных часто приводит к лучшим результатам решения задач обучения с учителем, чем существующие аналоги. Несмотря на то, что

формально метод не привносит новую информацию в обучающую выборку и поэтому не может улучшить качество работы применяемой после него модели обучения с учителем, на практике предварительная обработка данных существенно влияет на качество работы алгоритма машинного обучения. Хорошая предварительная обработка упрощает зависимости в данных, что приводит к более хорошему качеству решения задачи. Эксперименты показывают, что универсальный обуславливатель частично устраняет сложность данных, вызванную пропусками признаков и тем самым облегчает задачу для применяемых после него моделей обучения с учителем.

Эксперименты по дорисовке изображений наглядно показывают преимущества множественного восстановления пропущенных данных по сравнению с их единичным восстановлением. Генерируемые моделью изображения различны и реалистичны. Модель обучается за разумное время и может использовать внутри себя нейронные сети произвольной сложности и архитектуры. Также в экспериментах по дорисовке изображений демонстрируется возможность использования нестандартного распределения на маски пропущенных признаков для решения конкретной задачи.

Интересным направлением для будущей работы является добавление состязательного слагаемого в функцию потерь во избежание генерации размытых изображений, что является общей проблемой вариационных автокодировщиков.

6 Заключение

В данной работе была рассмотрена задача множественного восстановления пропущенных данных. Эта задача актуальна и имеет ряд практических приложений. Была предложена формализация задачи в терминах вероятностных моделей, введено распределение над масками пропущенных признаков и с его помощью правдоподобие модели. На основе существующих подходов был предложен конкретный вид вероятностной модели, основанной на нейронных сетях и приближающей распределения с помощью гауссовских скрытых переменных — универсальный обуславливатель. Была выведена вариационная нижняя оценка для логарифма правдоподобия этой модели, которая может быть эффективно максимизирована по параметрам модели. Модель эффективно работает как с большими выборками, так и с объектами большой размерности. В работе были предложены и изучены различные модификации универсального обуславливателя, такие как гауссовская стохастическая нейронная сеть и гибридная модель, вариация коэффициента при KL-дивергенции в процессе обучения,

работа с пропущенными данными в обучающей выборке, введение дополнительных априорных распределений в пространстве скрытых представлений объектов. Также в работе даются рекомендации по выбору гиперпараметров модели. Были проведены вычислительные эксперименты с модельными и реальными данными на разных задачах с различными метриками качества. Результаты экспериментов показывают, что универсальный обуславливатель способен выучивать сложные условные распределения, генерируемые им восстановления могут как использоваться другими методами машинного обучения, так и быть полезными сами по себе. Также в экспериментах показывается преимущество предлагаемой модели перед существующими методами восстановления пропущенных данных.

Список литературы

[1] Adagan: Boosting generative models / I. O. Tolstikhin, S. Gelly, O. Bousquet et al. // Advances in Neural Information Processing Systems. "— 2017. "— Pp. 5430–5439.

[2] *Batista G., Monard M. C.* A study of k-nearest neighbour as a model-based method to treat missing data // Argentine Symposium on Artificial Intelligence. "— 2001.

[3] *Brand M.* Incremental singular value decomposition of uncertain data with missing values // European Conference on Computer Vision / Springer. "— 2002. "— Pp. 707–720.

[4] *Burda Y., Grosse R., Salakhutdinov R.* Importance weighted autoencoders // *arXiv preprint arXiv:1509.00519*. "— 2015.

[5] *Chen T., Guestrin C.* XGBoost: A scalable tree boosting system // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. "— KDD '16. "— New York, NY, USA: ACM, 2016. "— Pp. 785–794.

[6] Context encoders: Feature learning by inpainting / D. Pathak, P. Krähenbühl, J. Donahue et al. // *CoRR*. "— 2016. "— Vol. abs/1604.07379.

[7] Deep learning face attributes in the wild / Z. Liu, P. Luo, X. Wang, X. Tang // Proceedings of International Conference on Computer Vision (ICCV). "— 2015. "— December.

[8] Deep residual learning for image recognition / K. He, X. Zhang, S. Ren, J. Sun // Proceedings of the IEEE conference on computer vision and pattern recognition. "— 2016. "— Pp. 770–778.

[9] Draw: A recurrent neural network for image generation / K. Gregor, I. Danihelka, A. Graves et al. // Proceedings of the 32nd International Conference on Machine Learning / Ed. by F. Bach, D. Blei. "— Vol. 37 of *Proceedings of Machine Learning Research*. "— Lille, France: PMLR, 2015. "— 07–09 Jul. "— Pp. 1462–1471.

[10] Em algorithm with gmm and naive bayesian to implement missing values / X.-Y. Zhou, J. S. Lim, I.-K. Kwon et al. // *Advanced Science and Technology Letters*. "— 2014. "— Vol. 46. "— Pp. 1–5.

[11] Generative adversarial nets / I. Goodfellow, J. Pouget-Abadie, M. Mirza et al. // Advances in neural information processing systems. "— 2014. "— Pp. 2672–2680.

[12] High-resolution image inpainting using multi-scale neural patch synthesis / C. Yang, X. Lu, Z. Lin et al. // The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). "— Vol. 1. "— 2017. "— P. 3.

[13] Image inpainting / M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester // Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. "— SIGGRAPH '00. "— New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000. "— Pp. 417–424.

[14] *Kingma D. P., Ba J.* Adam: A method for stochastic optimization // *arXiv preprint arXiv:1412.6980*. "— 2014.

[15] *Kingma D. P., Welling M.* Auto-encoding variational bayes // *CoRR*. "— 2013. "— Vol. abs/1312.6114.

[16] Ladder variational autoencoders / C. K. Sønderby, T. Raiko, L. Maaløe et al. // Advances in neural information processing systems. "— 2016. "— Pp. 3738–3746.

[17] *Lake B. M., Salakhutdinov R., Tenenbaum J. B.* Human-level concept learning through probabilistic program induction // *Science*. "— 2015. "— Vol. 350, no. 6266. "— Pp. 1332–1338.

[18] *LeCun Y., Cortes C., Burges C. J.* The mnist database of handwritten digits. "— 1998.

[19] *Lichman M.* UCI machine learning repository. "— 2013.

[20] *Mao X., Shen C., Yang Y.-B.* Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections // Advances in neural information processing systems. "— 2016. "— Pp. 2802–2810.

[21] Semantic image inpainting with deep generative models / R. A. Yeh, C. Chen, T. Y. Lim et al. // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. "— 2017. "— Pp. 5485–5493.

[22] Semantic image inpainting with perceptual and contextual losses / R. Yeh, C. Chen, T. Lim et al. // *CoRR*. "— 2016. "— Vol. abs/1607.07539.

[23] *Sohn K., Lee H., Yan X.* Learning structured output representation using deep conditional generative models // Advances in Neural Information Processing Systems 28 /

Ed. by C. Cortes, N. D. Lawrence, D. D. Lee et al. "— Curran Associates, Inc., 2015. "— Pp. 3483–3491.

[24] Techniques for dealing with missing values in classification / W. Z. Liu, A. P. White, S. G. Thompson, M. A. Bramer // International Symposium on Intelligent Data Analysis / Springer. "— 1997. "— Pp. 527–536.

[25] Towards missing data imputation: a study of fuzzy k-means clustering method / D. Li, J. Deogun, W. Spaulding, B. Shuart // International Conference on Rough Sets and Current Trends in Computing / Springer. "— 2004. "— Pp. 573–579.

[26] Veegan: Reducing mode collapse in gans using implicit variational learning / A. Srivastava, L. Valkoz, C. Russell et al. // Advances in Neural Information Processing Systems. "— 2017. "— Pp. 3310–3320.