**Методы синтеза данных**

После описания некоторых основных методов подбора распределения в предыдущей главе мы теперь будем использовать эти концепции для генерации синтетических данных. Мы начнем с некоторых основных подходов и перейдем к более сложным по мере продвижения главы. Позже мы обратимся к более продвинутым методам, которые выходят за рамки вводного текста, но то, что мы рассмотрим, должно дать вам хорошее введение.

**Получение синтетических данных из теории**

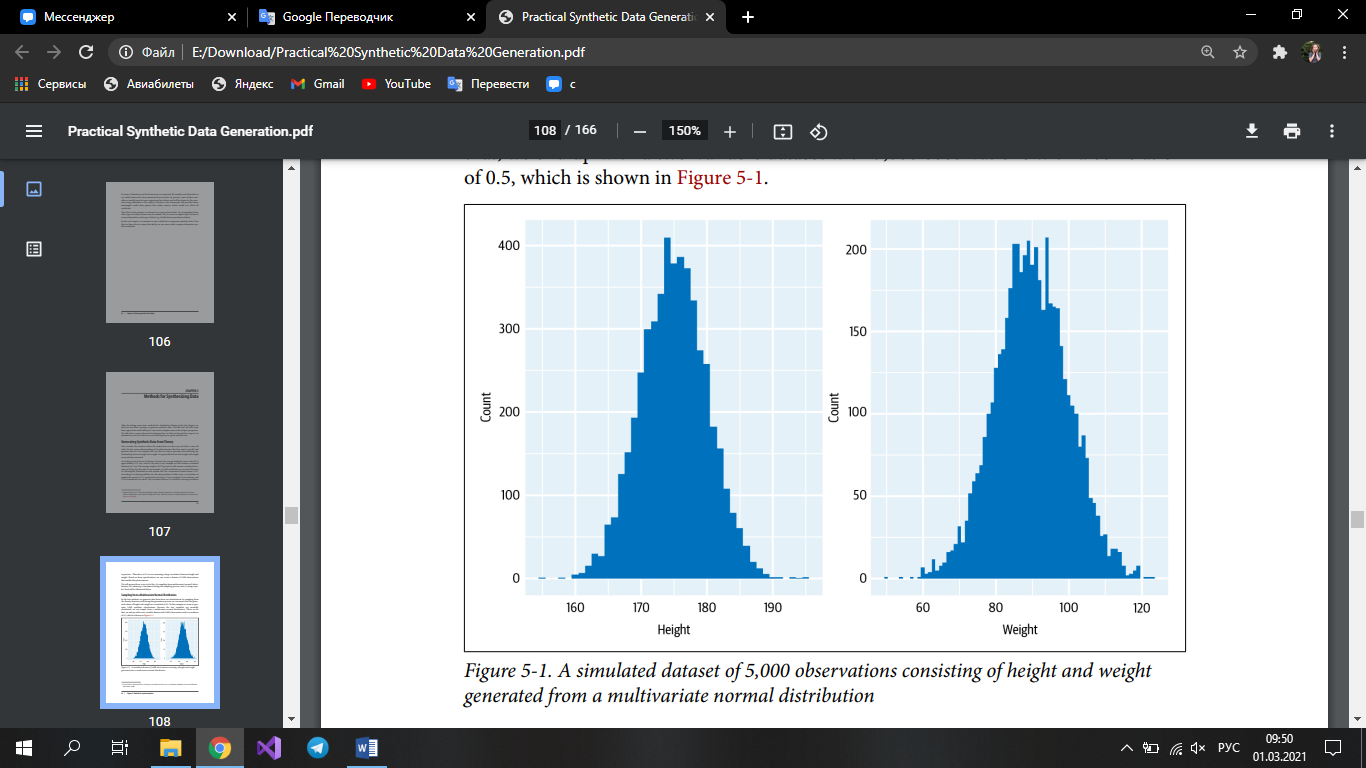
Давайте рассмотрим ситуацию, когда у аналитика нет реальных данных для начала, но он имеет некоторое представление о явлении, для которого он хочет смоделировать и сгенерировать данные. Например, предположим, что мы хотим сгенерировать данные, отражающие взаимосвязь между ростом и весом. Общеизвестно, что рост и вес положительно связаны.

По данным Центров по контролю за заболеваниями, средний рост мужчин в США составляет примерно 175 см, и для нашего примера мы примем стандартное отклонение в 5 см. Средний вес составляет 89,7 кг, и мы примем стандартное отклонение в 10 кг. Для нашего примера мы будем моделировать их как нормальные (гауссовские или колоколообразные) распределения и предположить, что корреляция между ними составляет 0,5. Согласно рекомендациям Коэна по интерпретации величины эффекта, корреляция величины, равная 0,5, считается большой, 0,3 - средней, а 0,1 - малой. Любая корреляция выше 0,5 на практике будет сильной корреляцией. Следовательно, при 0,5 мы предполагаем большую корреляцию между ростом и весом. На основе этих спецификаций мы можем создать набор данных из 5000 наблюдений, которые моделируют это явление.

Мы представим три способа сделать это: (а) выборка из многомерных (нормальных) распределений, (б) создание корреляции в процессе выборки и (в) использование копул. Каждый будет проиллюстрирован ниже.

**Выборка из многомерного нормального распределения**

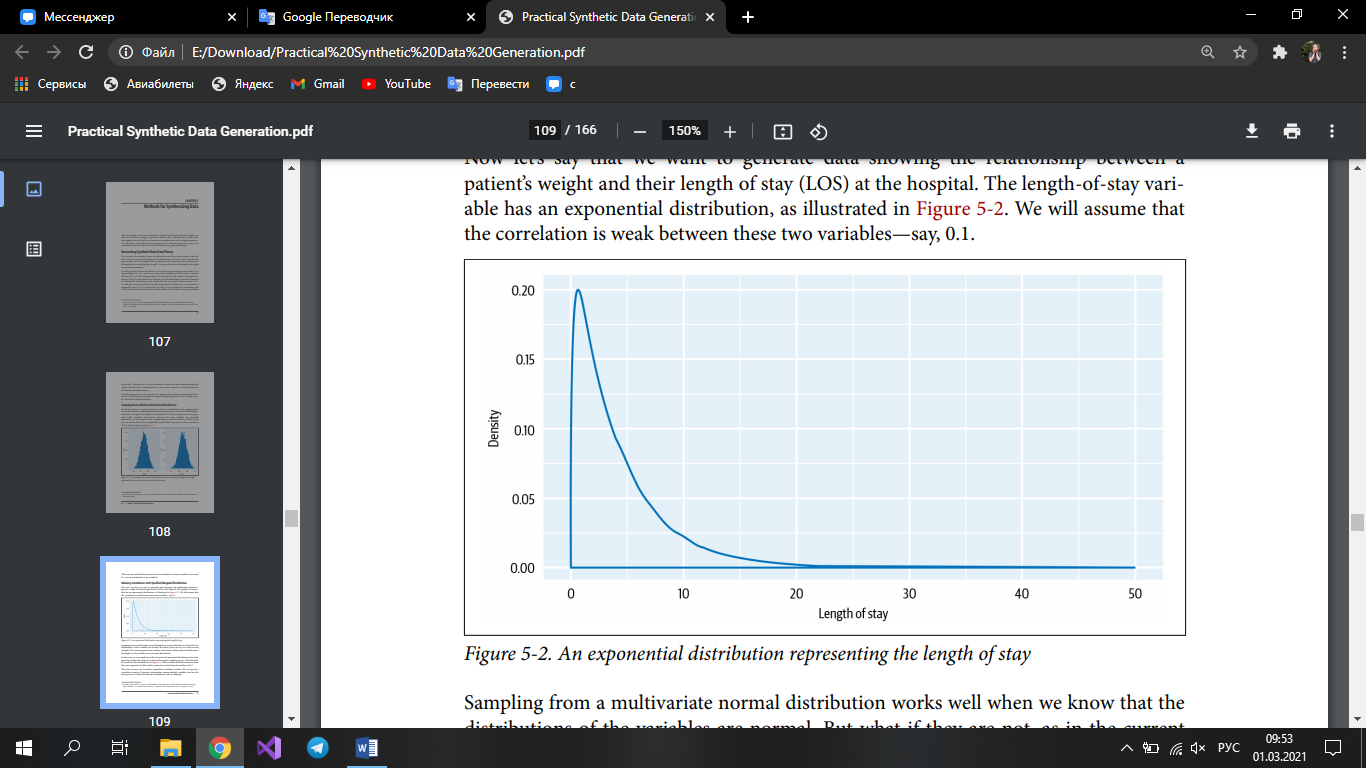
В первом методе мы генерируем данные из этих двух распределений путем выборки из функции плотности, и в процессе генерации мы можем гарантировать, что сгенерированные значения роста и веса коррелируют на 0,5. В этом примере мы хотим создать 5 000 синтетических наблюдений. Поскольку две переменные имеют нормальное распределение, мы можем сделать выборку из многомерного нормального распределения. Когда мы это сделаем, мы получим набор данных с двумя переменными с 5000 наблюдениями с корреляцией 0,5, как показано на рисунке 5-1.

  
*Рисунок 5-1. Смоделированный набор данных из 5000 наблюдений, состоящих из показателей роста и веса, созданных на основе многомерного нормального распределения.*

Это было легко. И основной процесс может быть расширен на любое количество переменных (т.е. мы не ограничены двумя переменными).

**Вызывание корреляций с заданными маржинальными распределениями**

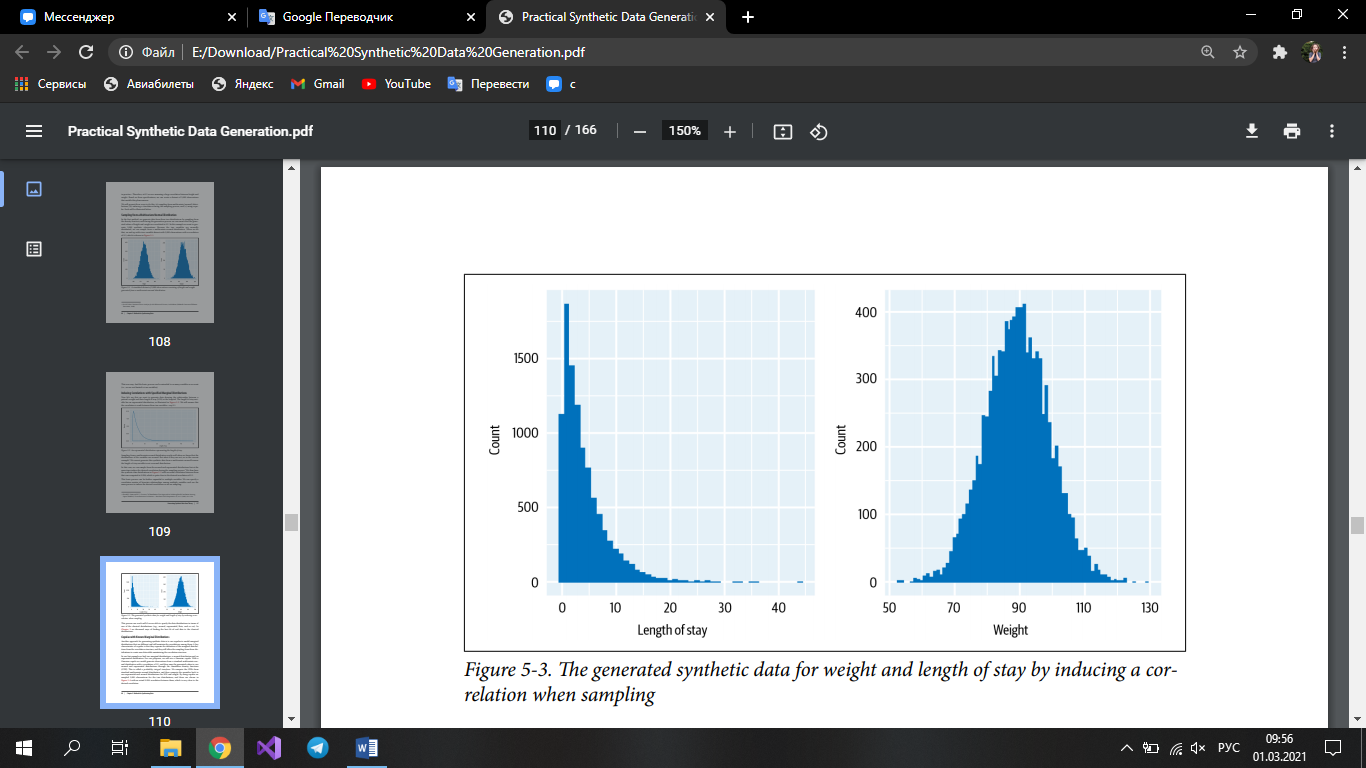
Теперь предположим, что мы хотим сгенерировать данные, показывающие взаимосвязь между весом пациента и продолжительностью его пребывания (LOS) в больнице. Переменная продолжительности пребывания имеет экспоненциальное распределение, как показано на рисунке 5-2. Мы предположим, что корреляция между этими двумя переменными слабая - скажем, 0,1.

  
*Рисунок 5-2. Экспоненциальное распределение, представляющее продолжительность пребывания*

Выборка из многомерного нормального распределения работает хорошо, когда мы знаем, что распределения переменных являются нормальными. Но что, если это не так, как в текущем примере? Мы не можем сгенерировать эти синтетические данные из многомерной нормы, потому что переменная продолжительности пребывания не является нормальным распределением.

В этом случае мы можем производить выборку из нормального и экспоненциального распределений, но в то же время вызывать желаемую корреляцию в процессе выборки. Затем у нас есть синтетические распределения данных на рис. 5-3 с фактической корреляцией между ними, которая была вычислена при 0,094, что довольно близко к желаемой корреляции 0,1.

Этот базовый процесс можно расширить до нескольких переменных. Мы можем указать корреляционную матрицу двумерных отношений между несколькими переменными и использовать один и тот же процесс для получения желаемых корреляций при отборе выборки.

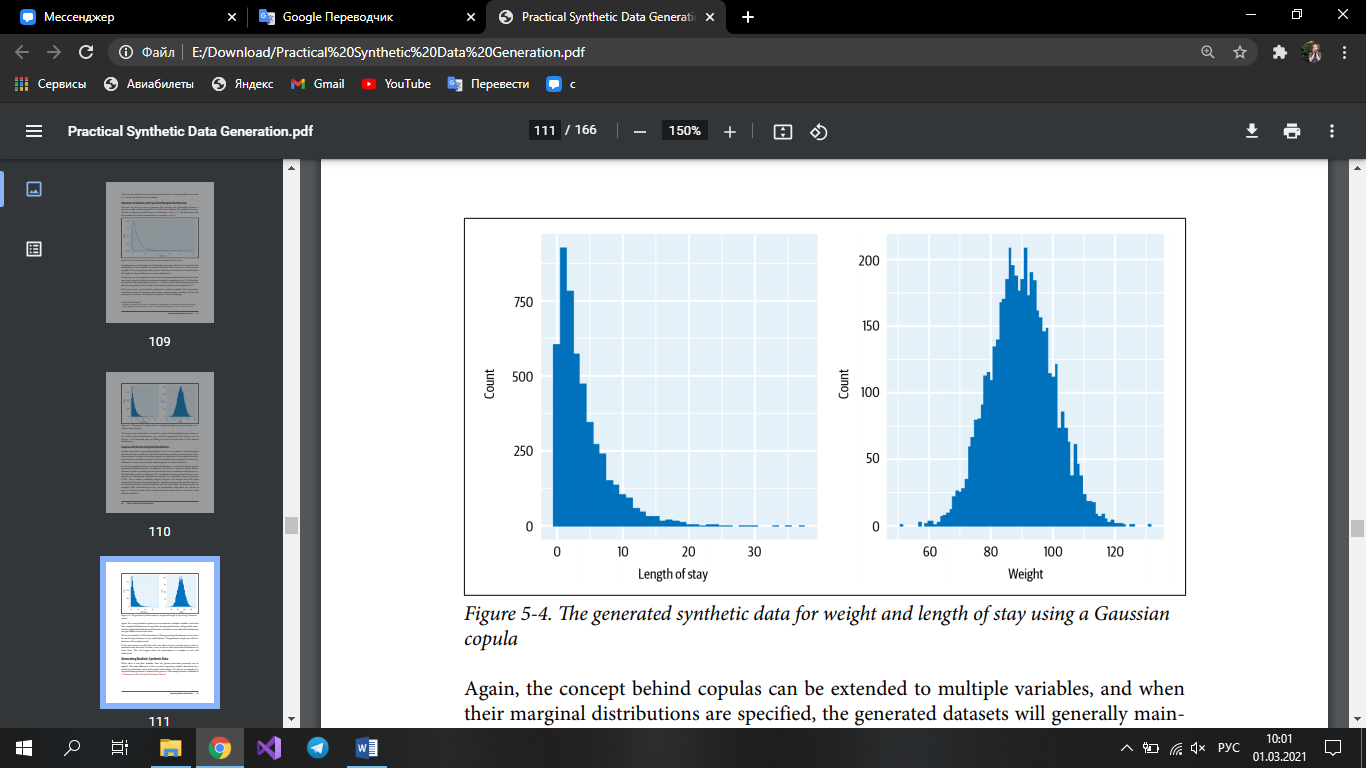
  
*Рисунок 5-3. Сгенерированные синтетические данные для веса и продолжительности пребывания путем установления корреляции при отборе образцов.*

Этот процесс может работать хорошо, если мы можем указать распределения данных в терминах одного из классических распределений (например, нормального, экспоненциального, бета-версии и т. Д.). В главе 3 мы обсудили способы нахождения наилучшего соответствия реальных данных классическим распределениям.

**Копулы с известным маргинальным распределением**

Другой подход к генерации синтетических данных - использование копул для моделирования маржинальных распределений, которые отличаются друг от друга и при этом сохраняют корреляции между ними. Ключевой характеристикой связок является то, что они отделяют определение предельных распределений от корреляционной структуры и по-прежнему позволяют производить выборку из этих распределений для создания новых данных при сохранении корреляционной структуры.

В нашем последнем примере у нас было два предельных распределения, нормальное распределение и экспоненциальное распределение. Для наших целей мы будем использовать гауссову связку. С помощью гауссовой копулы мы будем генерировать наблюдения из стандартного многомерного нормального распределения с корреляцией 0,1, а затем отображать полученные значения в наши нормальные и экспоненциальные распределения с помощью кумулятивных функций плотности (КФП). Это называется интегральным преобразованием вероятности. Мы вычисляем КФП из стандартного многомерного нормального распределения, а затем вычисляем квантили обратно к нашим экспоненциальным и нормальным распределениям для продолжительности пребывания (ПП) и веса. Используя связки, мы отобрали 5000 наблюдений для двух распределений, и они показаны на рис. 5-4 с фактической корреляцией между ними 0,094, что очень близко к желаемой корреляции.

  
*Рисунок 5-4. Сгенерированные синтетические данные для веса и продолжительности пребывания с использованием гауссовой копулы.*

Опять же, концепция, лежащая в основе связок, может быть расширена на несколько переменных, и когда их маргинальные распределения указаны, сгенерированные наборы данных, как правило, будут поддерживать маргинальные распределения и двумерные корреляции, даже если распределения сильно отличаются друг от друга.

Мы не ограничиваемся 5000 наблюдений. При создании наборов данных мы можем сделать это для гораздо больших наборов данных или очень маленьких наборов данных. Сгенерированный размер выборки будет зависеть от потребностей аналитика.

В следующем разделе мы рассмотрим случай, когда у нас есть реальные данные, и мы хотим синтезировать данные из них. В таком случае у нас нет теоретических распределений для работы. Это может произойти, если явление сложное или недостаточно изученное.

**Создание реалистичных синтетических данных**

При наличии реальных данных можно применить описанный ранее процесс. Основное отличие состоит в том, что нам нужно генерировать синтетические данные на основе модели реальных наборов данных, а не теоретических взаимосвязей. Мы будем использовать пример набора данных о выписке из больницы, чтобы проиллюстрировать этот процесс. Этот пример набора данных подробно описан в «Описание набора данных о выписке из больницы».

*Описание набора данных о выписке из больницы*

*При выписке пациента все данные, относящиеся к его опыту пребывания в больнице, собираются и помещаются в стандартизированные базы данных о выписке. Эти базы данных очень важны для анализа производительности и затрат системы здравоохранения, улучшения качества и общественного здоровья, среди других причин. Эти данные называются выписками или сводками.*

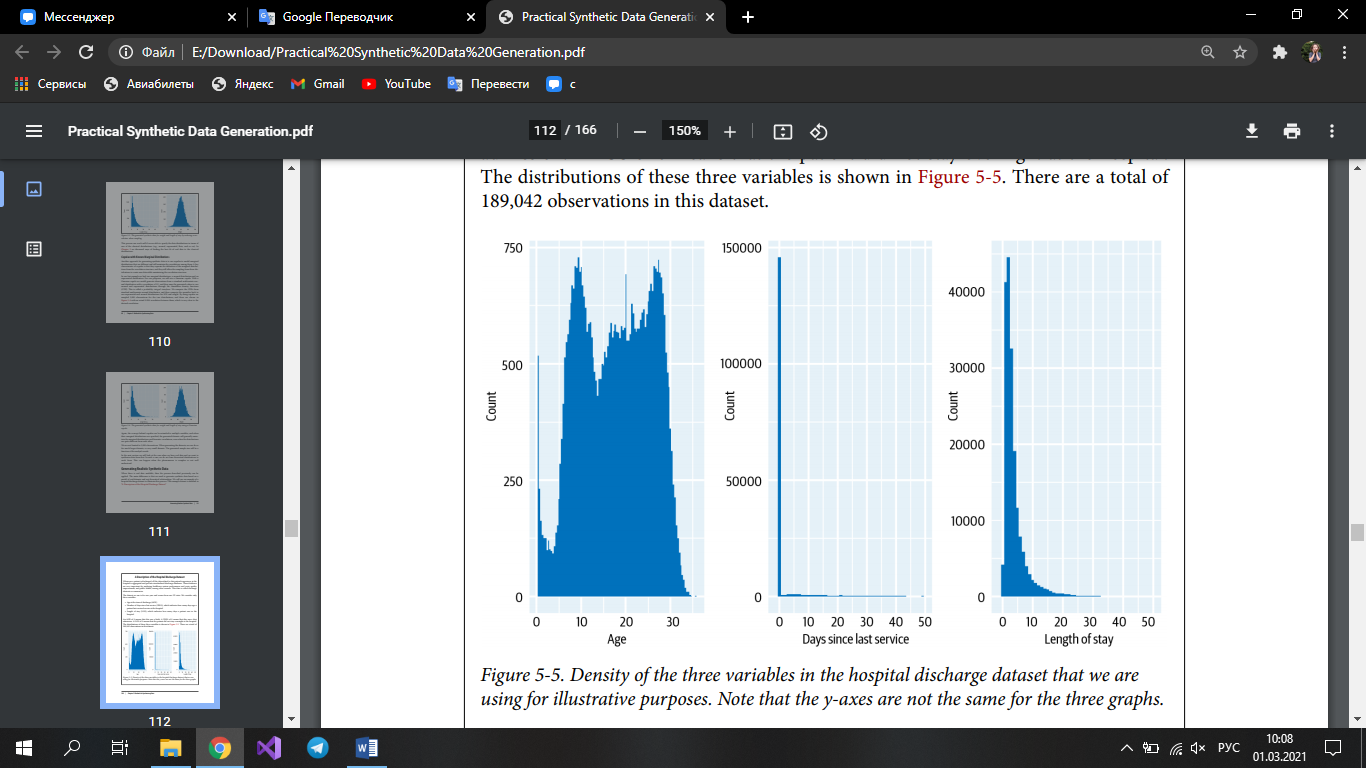
*Набор данных, который мы используем, рассчитан на один год и поступает из одного штата США. Мы рассматриваем только три переменные:*

*• Возраст на момент выписки (ВОЗРАСТ)*

*• Количество дней с момента последнего обслуживания (КДсПО), которое показывает, сколько дней назад пациенту в последний раз оказывалась услуга в больнице.*

*• Продолжительность пребывания (ПП), которая показывает, сколько дней пациент находился в больнице.*

*ВОЗРАСТ 0 означает, что это было рождение. КДсПО 0 означает, что это было первое поступление. ПП, равное 0, означает, что пациент не оставался на ночь в больнице. Распределение этих трех переменных показано на рисунке 5-5. Всего в этом наборе данных 189 042 наблюдения.*

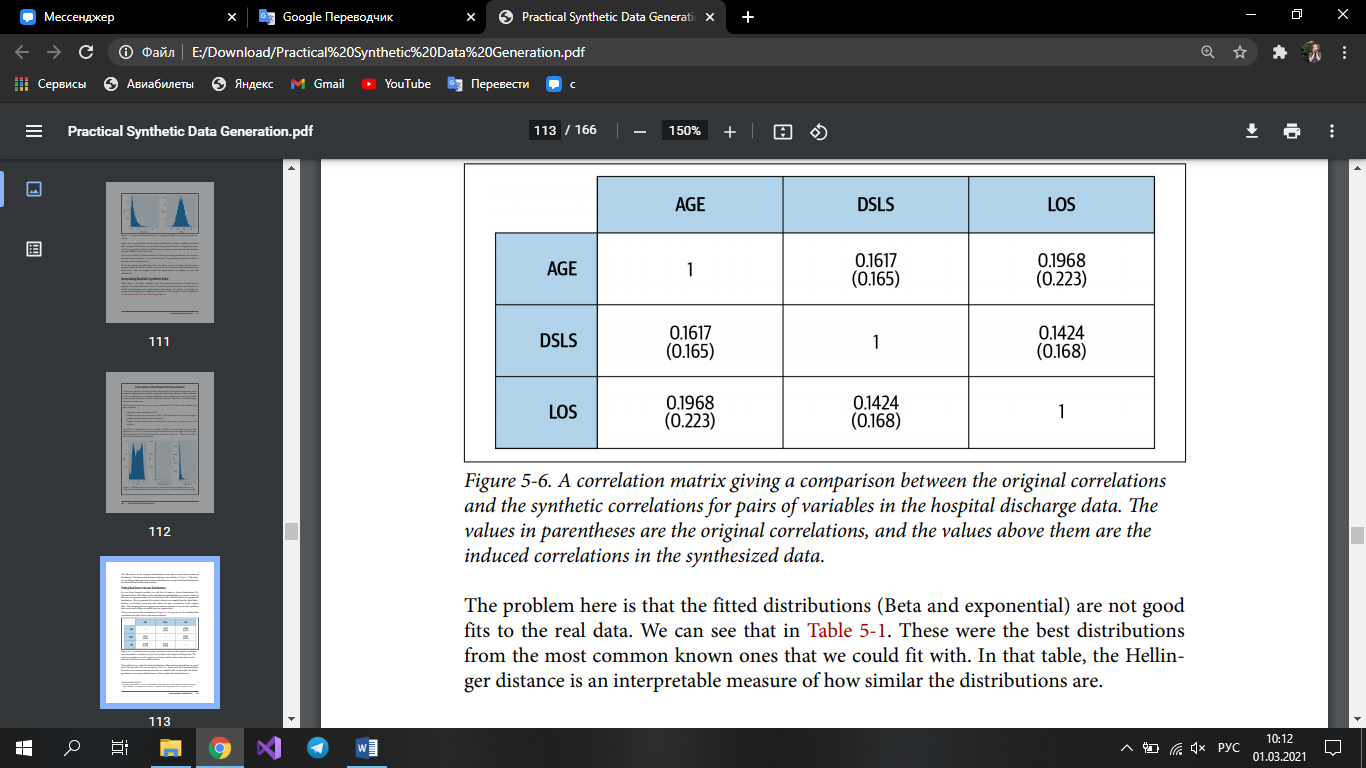
 *Рисунок 5-5. Плотность трех переменных в наборе данных о выписке из больницы, который мы используем в иллюстративных целях. Обратите внимание, что оси Y на трех графиках не совпадают.*

Нам нужно будет подогнать маржинальные распределения в наших данных к некоторому классическому распределению. Мы обсуждали подгонку распределения более подробно в главе 3. Таким образом, мы все еще генерируем данные из классических распределений, за исключением того, что эти распределения выводятся из наилучшего соответствия реальным данным.

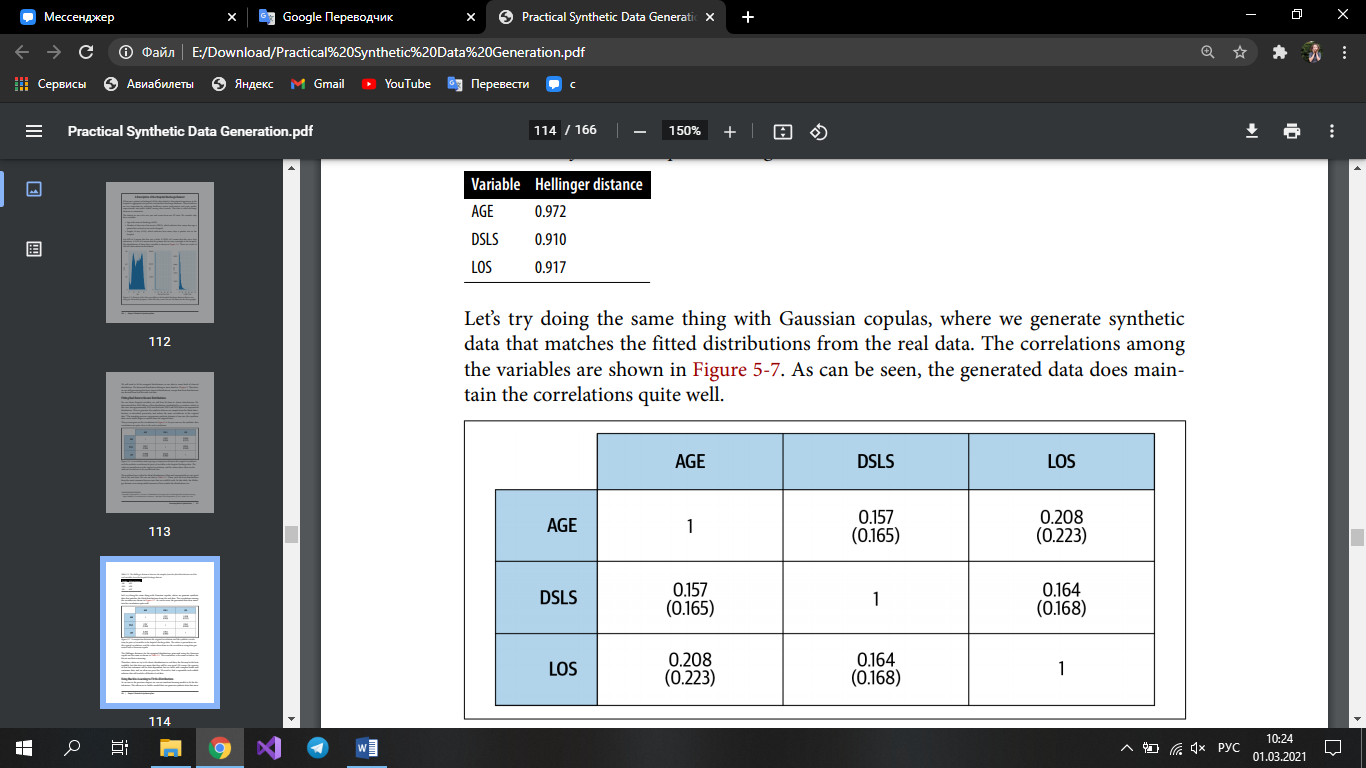
**Подгонка реальных данных к известным распределениям**

Для наших трех переменных для больниц мы сначала подгоним их к классическим распределениям. Мы определили, что ВОЗРАСТ следует бета-распределению (умноженному на константу, которая в данном случае была приблизительно 100) и что как КДсПО, так и ПП подчиняются экспоненциальному распределению. Затем, чтобы сгенерировать синтетические данные, мы можем сделать выборку из подобранных распределений, как описано ранее, и вызвать те же корреляции, что и исходные данные.

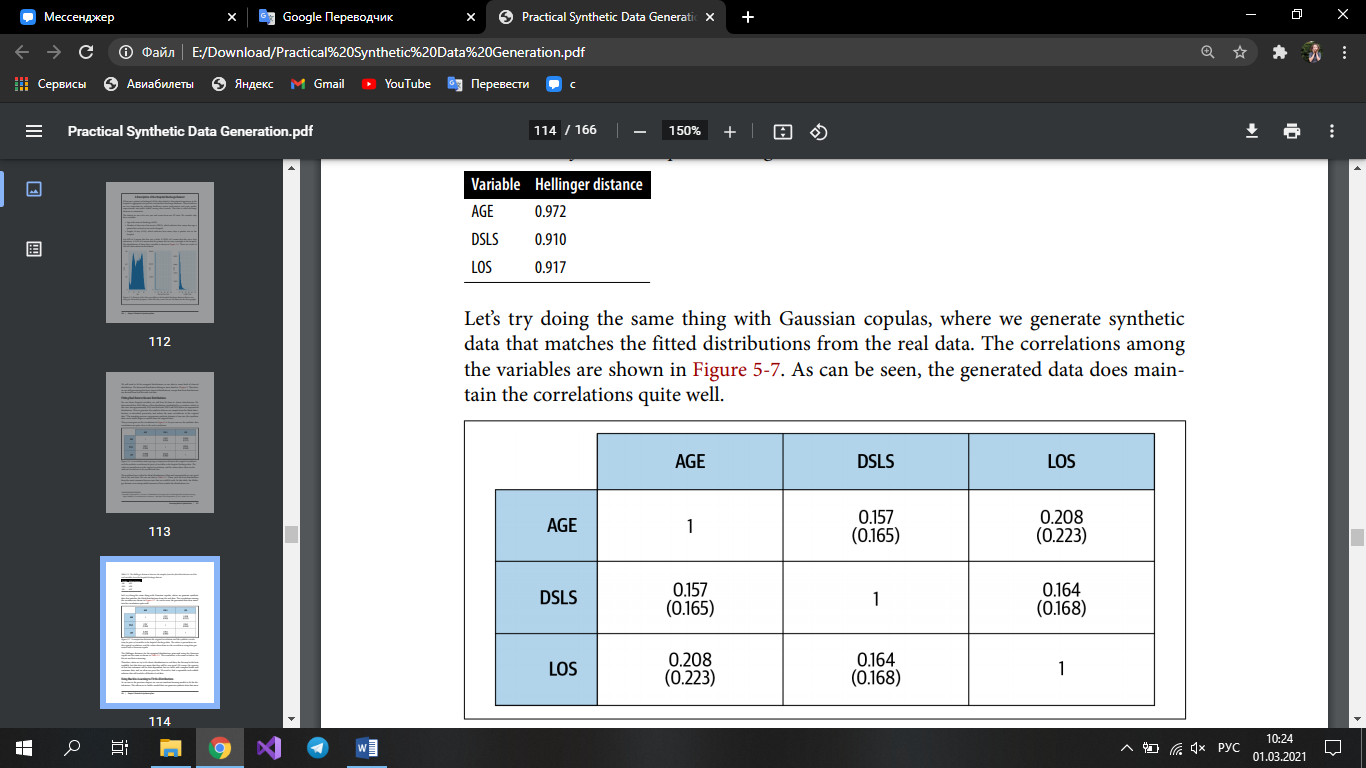
В процессе выборки можно создавать синтетические наборы данных любого размера (синтетические данные могут быть намного больше или меньше исходных данных). Этот процесс дает нам корреляции, показанные на рис. 5-6. Как видите, корреляции синтетических данных довольно близки к реальным корреляциям.

  
*Рисунок 5-6. Матрица корреляции, дающая сравнение между исходными корреляциями и синтетическими корреляциями для пар переменных в данных о выписке из больницы. Значения в скобках - это исходные корреляции, а значения над ними - индуцированные корреляции в синтезированных данных.*

Проблема здесь в том, что подобранные распределения (бета и экспоненциальные) плохо подходят для реальных данных. Мы можем видеть это в Таблице 5-1. Это были лучшие дистрибутивы из наиболее распространенных известных, с которыми мы могли согласиться. В этой таблице расстояние Хеллингера является интерпретируемой мерой сходства распределений.

*Таблица 5-1. Расстояния Хеллингера между выборками из подобранных распределений и реальными переменными из набора данных при выписке из больницы.*

Давайте попробуем проделать то же самое с гауссовыми связками, где мы генерируем синтетические данные, которые соответствуют подобранным распределениям из реальных данных. Корреляции между переменными показаны на Рисунке 5-7. Как видно, сгенерированные данные действительно хорошо поддерживают корреляции.

  
*Рисунок 5-7. Сравнение исходных корреляций и синтетических корреляций для пар переменных в данных о выписке из больницы. Значения в скобках - это исходные корреляции, а значения над ними - это корреляции с использованием данных, сгенерированных с помощью гауссовой копулы.*

Расстояния Хеллингера для маргинальных распределений, созданных с использованием гауссовой связки, такие же, как показано в таблице 5-1. Вывод такой же, как и раньше: посадки не так убедительны.

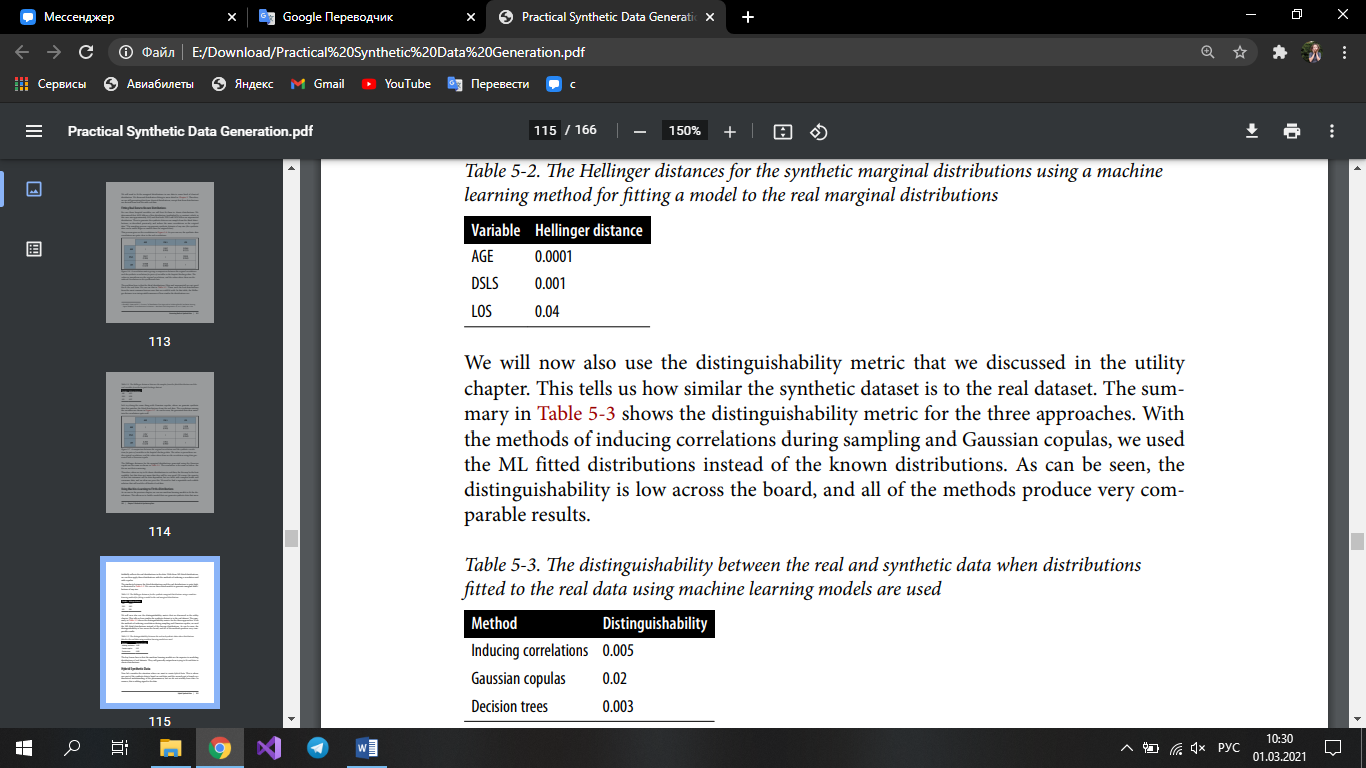
Поэтому, когда мы пытаемся подогнать классические распределения к реальным данным, они могут быть наилучшими из имеющихся, но это не означает, что они будут очень хорошими. Конечно, правдивость этого последнего утверждения будет зависеть от данных, но мы работаем со сложными данными о здоровье и потребителях, и мы часто видим, что они плохо подходят. Нам нужно найти повторяемое и масштабируемое решение, которое будет работать для всех видов реальных данных.

**Использование машинного обучения для соответствия дистрибутивам**

Как мы видели в предыдущей главе, мы можем использовать модели машинного обучения, чтобы соответствовать дистрибутивам. Это позволяет нам построить модель, которая может генерировать синтетические данные, более точно отражающие реальное распределение данных. С этими подобранными распределениями ML мы можем затем применить эти распределения с помощью методов индукции корреляции и копул.

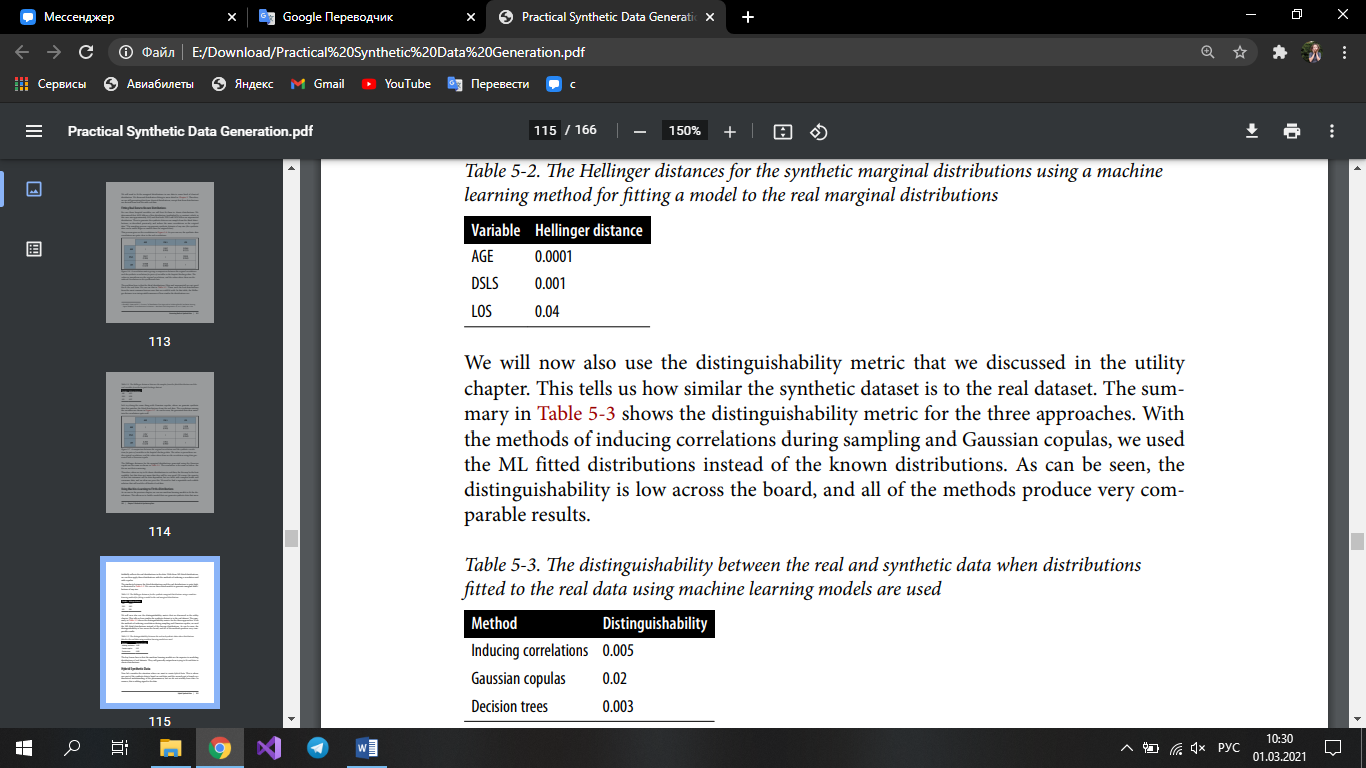
Сходство между подобранными распределениями и реальными распределениями довольно велико, как показано в Таблице 5-2. Мы можем использовать эти подогнанные модели для создания предельных распределений любого размера.

*Таблица 5-2. Расстояния Хеллингера для синтетических маржинальных распределений с использованием метода машинного обучения для подгонки модели к реальным маржинальным распределениям*



Теперь мы также будем использовать метрику различимости, которую мы обсуждали в главе полезности. Это говорит нам о том, насколько синтетический набор данных похож на настоящий. Сводка в таблице 5-3 показывает метрику различимости для трех подходов. Используя методы наведения корреляций во время выборки и гауссовские связки, мы использовали аппроксимирующие распределения ML вместо известных распределений. Как можно видеть, различимость невысока по всем направлениям, и все методы дают очень сопоставимые результаты.

*Таблица 5-3. Различие между реальными и синтетическими данными при использовании распределений, подогнанных к реальным данным с использованием моделей машинного обучения.*

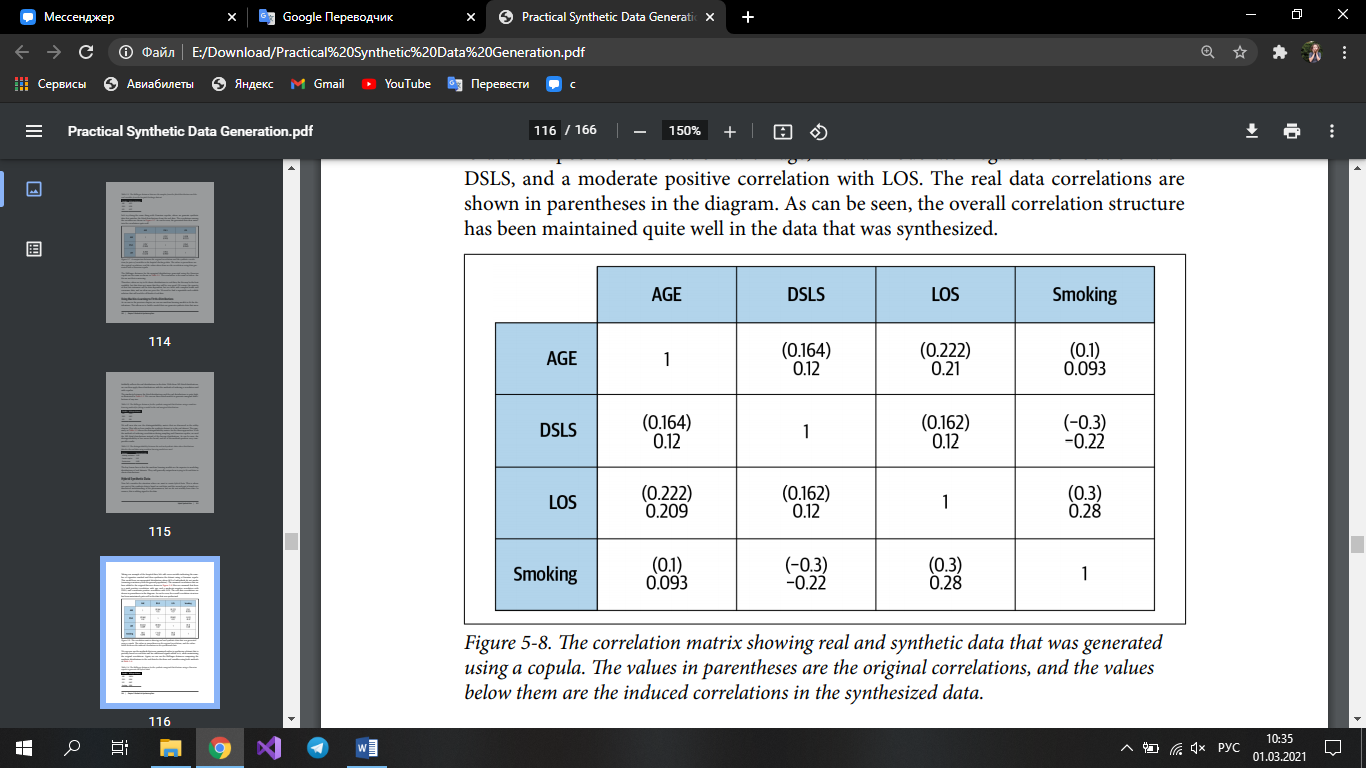


Ключевой урок здесь заключается в том, что модели машинного обучения намного превосходят моделирование распределений реальных наборов данных. Как правило, они превосходят попытки подогнать реальные данные к классическим распределениям.

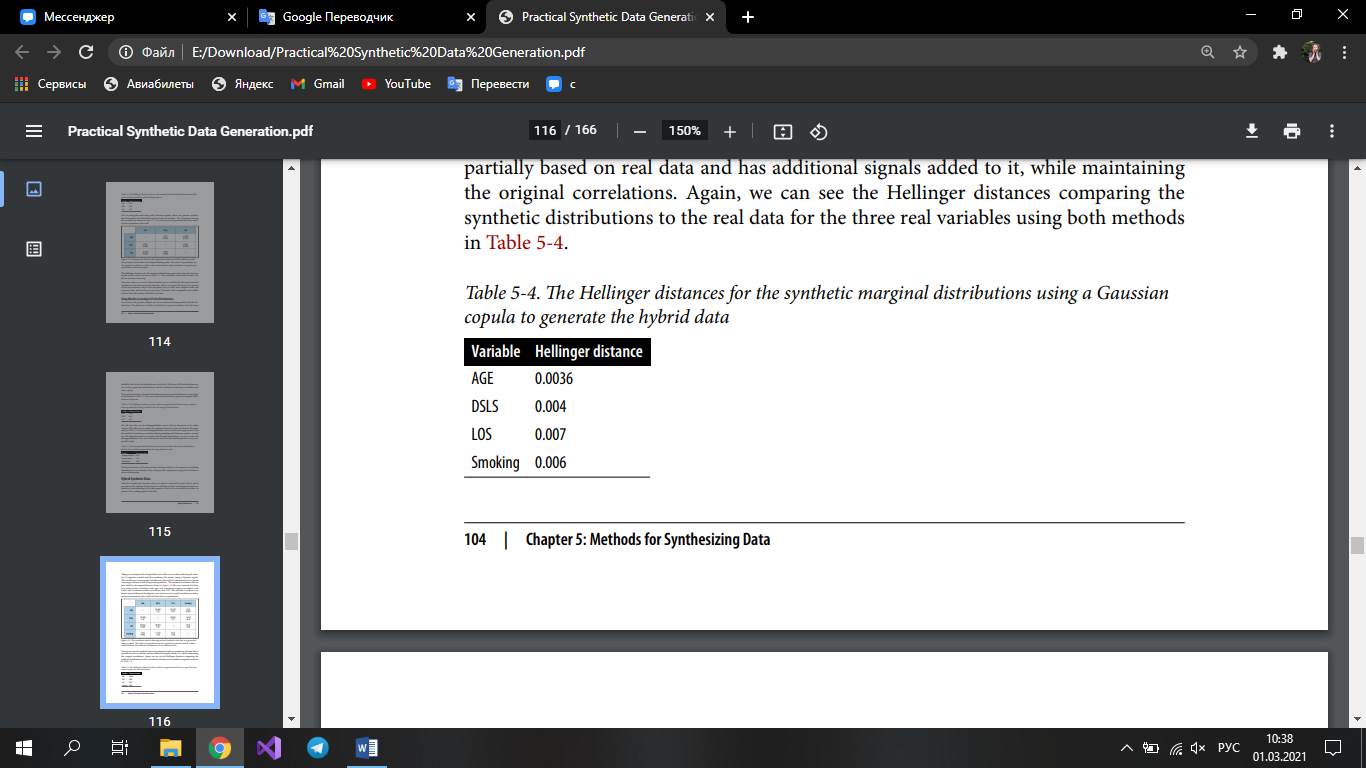
**Гибридные синтетические данные**

Теперь давайте рассмотрим ситуацию, когда мы хотим создать гибридные данные. Здесь одна часть синтетических данных основана на реальных данных, а вторая часть основана на теоретическом понимании явления, но на самом деле у нас нет данных. По сути, это добавление сигнала к данным.

Взяв наш пример с данными о больнице, давайте добавим новую переменную, указывающую количество выкуриваемых сигарет, а затем синтезируем набор данных с помощью гауссовой копулы. Это будет экспоненциальное распределение, когда 86% людей не курят (что обеспечивает согласованность с населением в целом). Предполагаемые корреляции, которые мы добавили к исходным данным, показаны на рисунке 5-8. Здесь мы предположили, что существует слабая положительная корреляция с возрастом, умеренная отрицательная корреляция с КДсПО и умеренная положительная корреляция с ПП. Реальные корреляции данных показаны на диаграмме в скобках. Как можно видеть, общая корреляционная структура довольно хорошо поддерживается в синтезированных данных.

  
*Рисунок 5-8. Корреляционная матрица, показывающая реальные и синтетические данные, которые были сгенерированы с помощью копулы. Значения в скобках - это исходные корреляции, а значения под ними - индуцированные корреляции в синтезированных данных.*

Теперь мы можем использовать методы, которые были исследованы ранее, для синтеза набора данных, который частично основан на реальных данных и имеет добавленные к нему дополнительные сигналы, сохраняя при этом исходные корреляции. Опять же, мы можем увидеть расстояния Хеллингера, сравнивая синтетические распределения с реальными данными для трех реальных переменных, используя оба метода в таблице 5-4.

*Таблица 5-4. Расстояния Хеллингера для синтетических маргинальных распределений с использованием гауссовой копулы для генерации гибридных данных.*  


Этот синтетический набор данных объединил реальную информацию с гипотетической информацией для создания гибрида. Основные принципы могут быть легко распространены на большее количество переменных и использоваться с другими методами.

Описанный здесь набор методов предоставляет набор инструментов для создания искусственных, реалистичных и гибридных данных. Кроме того, методы могут быть расширены до произвольного числа переменных для создания довольно сложных наборов данных.

*Последовательный синтез машинного обучения*

*Один из способов генерации синтетических данных - использовать обычно используемые алгоритмы регрессии и классификации. Обычно используемые алгоритмы - это деревья классификации и регрессии (ДКиР), хотя также могут использоваться их варианты. Также могут использоваться другие алгоритмы, такие как машины опорных векторов. Для иллюстрации предположим, что ДКиР используется для синтеза.*

*Допустим, у нас есть пять переменных: A, B, C, D и E. Генерация выполняется последовательно, поэтому нам нужна последовательность. Для выбора последовательности можно использовать различные критерии. В нашем примере мы определяем последовательность как A → E → C → B → D.*

*Пусть штриховое обозначение означает, что переменная синтезирована. Например, A 'означает, что это синтезированная версия A. Ниже приведены шаги для последовательной генерации:*

*• Выборка из распределения A для получения A ′*

*• Постройте модель F1: E ~ A*

*• Синтезируйте E как E ′ = F1 (A ′)*

*• Постройте модель F2: C ~ A + E*

*• Синтезировать C как C ′ = F2 (A ′, E ′)*

*• Постройте модель F3: B ~ A + E + C*

*• Синтезировать B как B ′ = F3 (A ′, E ′, C ′)*

*• Постройте модель F4: D ~ A + E + C + B*

*• Синтезировать D как D ′ = F4 (A ′, E ′, C ′, B ′)*

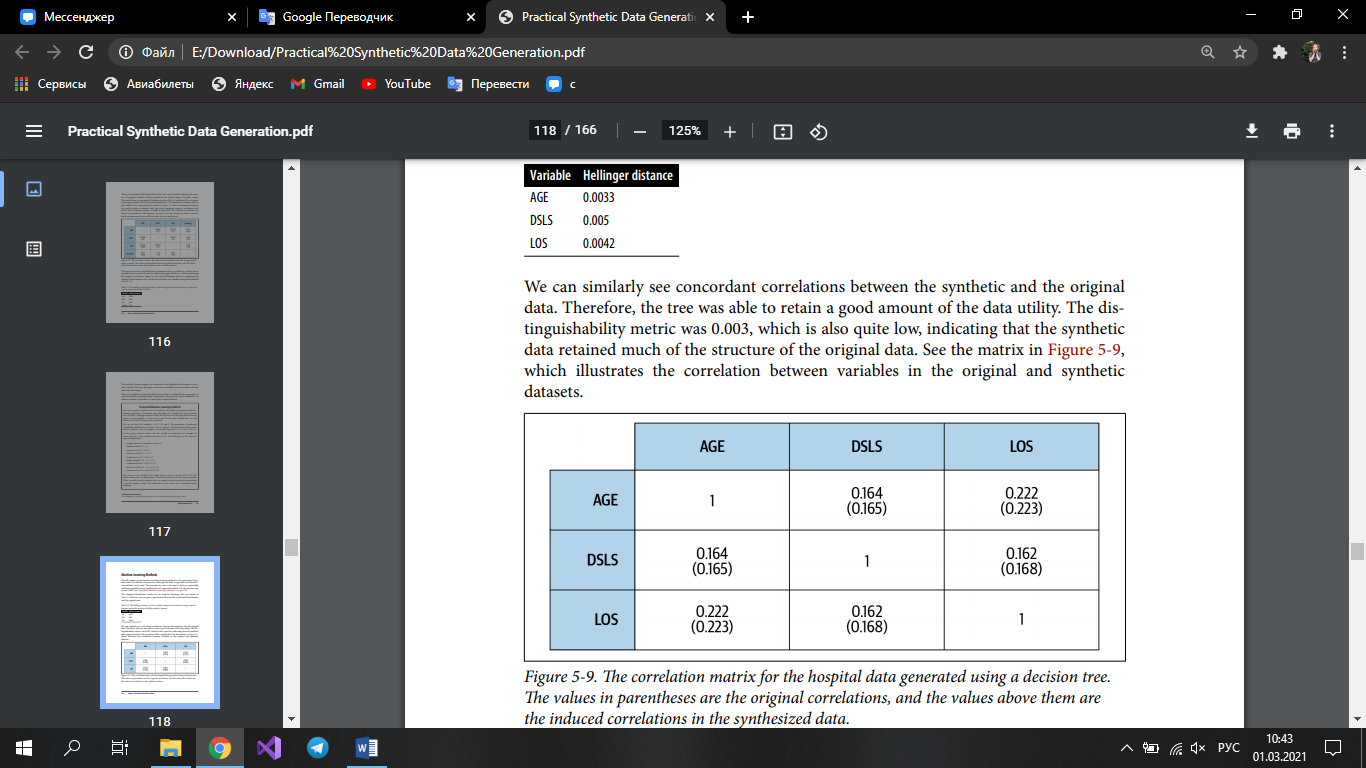
*Процесс можно рассматривать как первоначальную установку серии моделей {F1, F2, F3, F4}. Эти модели составляют генератор. Затем эти модели можно использовать для генерации данных. Когда модель используется для генерации данных, мы делаем выборку из предсказанного конечного узла, чтобы получить синтетические значения. Перед выборкой распределение в узле можно сгладить.*

**Методы машинного обучения**

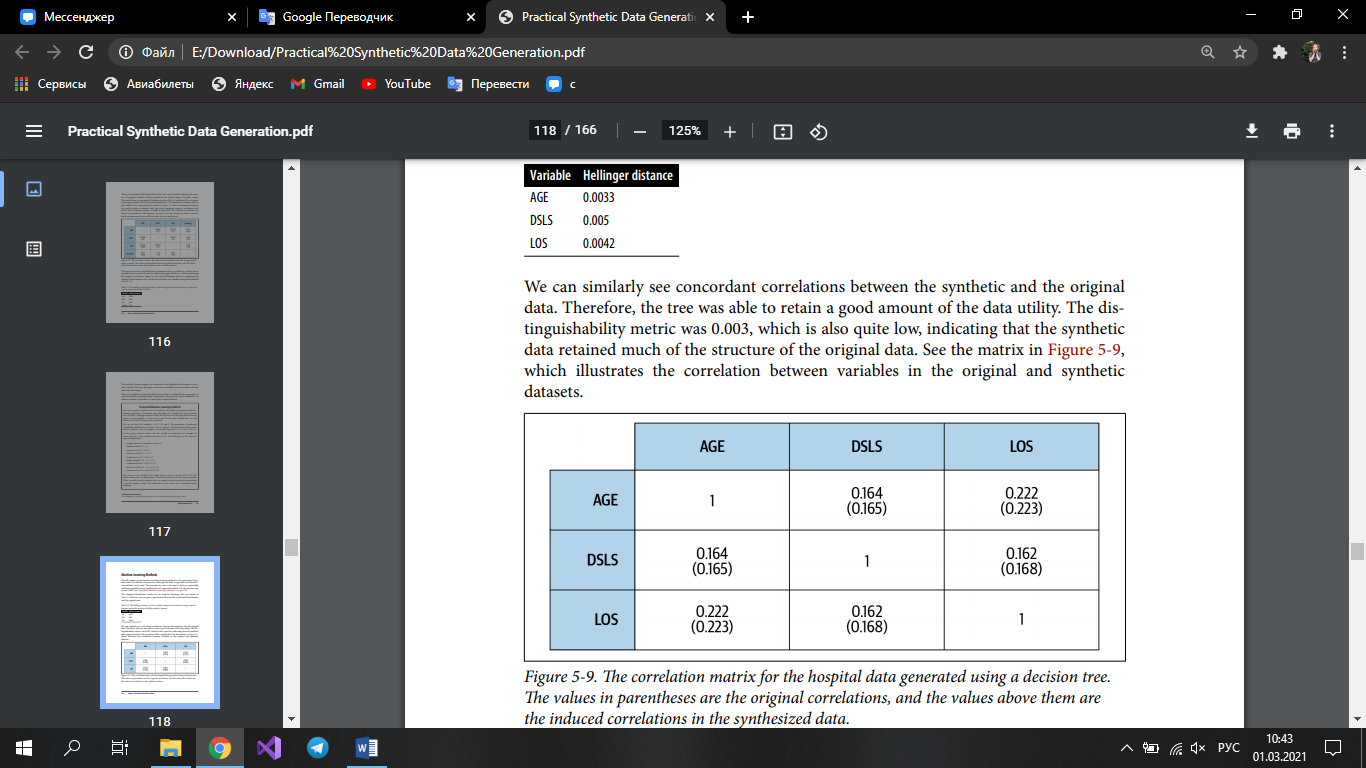
Мы рассмотрим репрезентативный метод машинного обучения для генерации синтетических данных. Мы будем использовать дерево решений, хотя можно использовать любой метод регрессии и классификации. Принцип для каждого один и тот же в том, что мы последовательно синтезируем переменные, используя модели классификации и регрессии. Для дерева решений мы используем CART (см. «Последовательный синтез машинного обучения» на стр. 105).

Результаты предельного распределения по нашим данным о выписке из больницы показаны в Таблице 5-5. Здесь мы видим неплохое совпадение синтезированных дистрибутивов с исходными.

*Таблица 5-5. Расстояния Хеллингера для синтетических маржинальных распределений с использованием метода машинного обучения для создания всех синтетических наборов данных.*



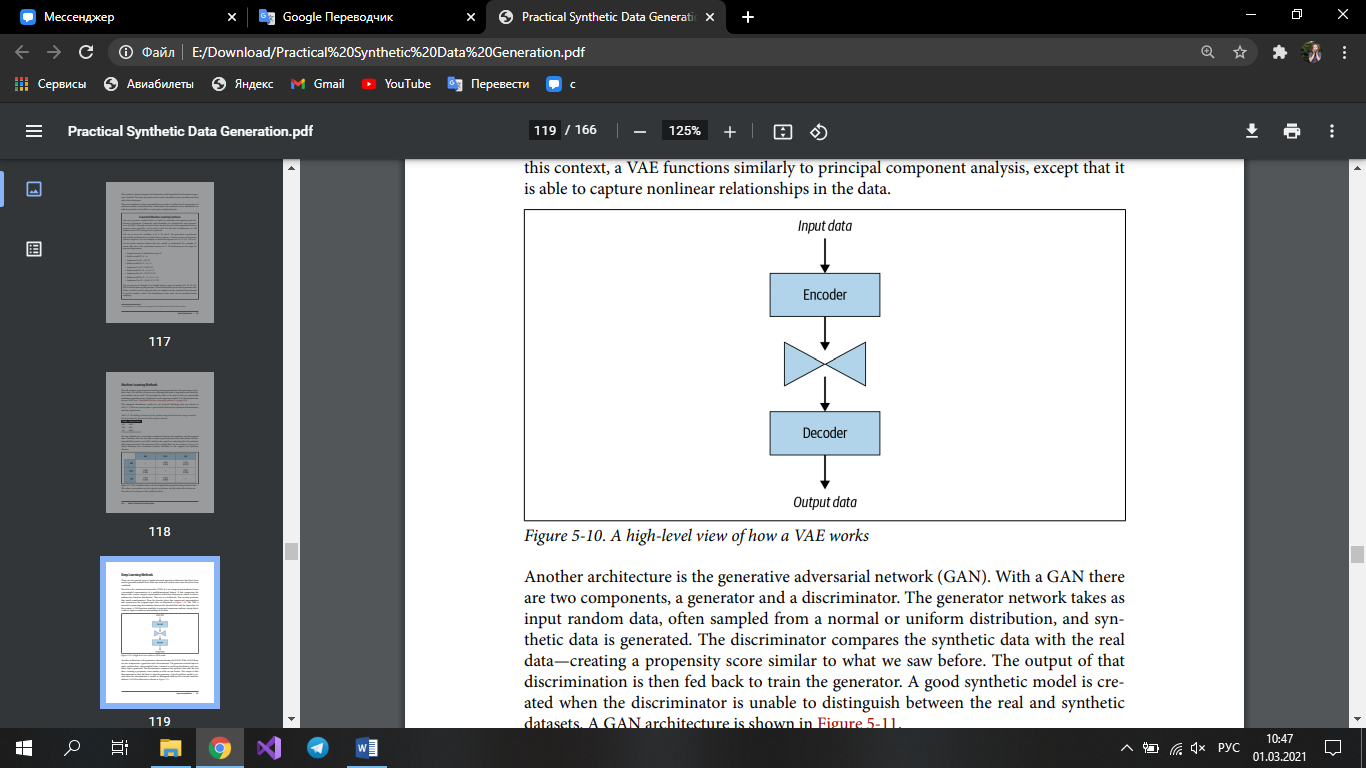
Мы также можем видеть согласованные корреляции между синтетическими и исходными данными. Таким образом, дерево смогло сохранить значительную часть полезности данных. Показатель различимости составил 0,003, что также довольно мало, что указывает на то, что синтетические данные сохранили большую часть структуры исходных данных. См. Матрицу на рис. 5-9, которая иллюстрирует корреляцию между переменными в исходных и синтетических наборах данных.

  
*Рисунок 5-9. Матрица корреляции для данных больницы, созданная с помощью дерева решений. Значения в скобках - это исходные корреляции, а значения над ними - индуцированные корреляции в синтезированных данных.*

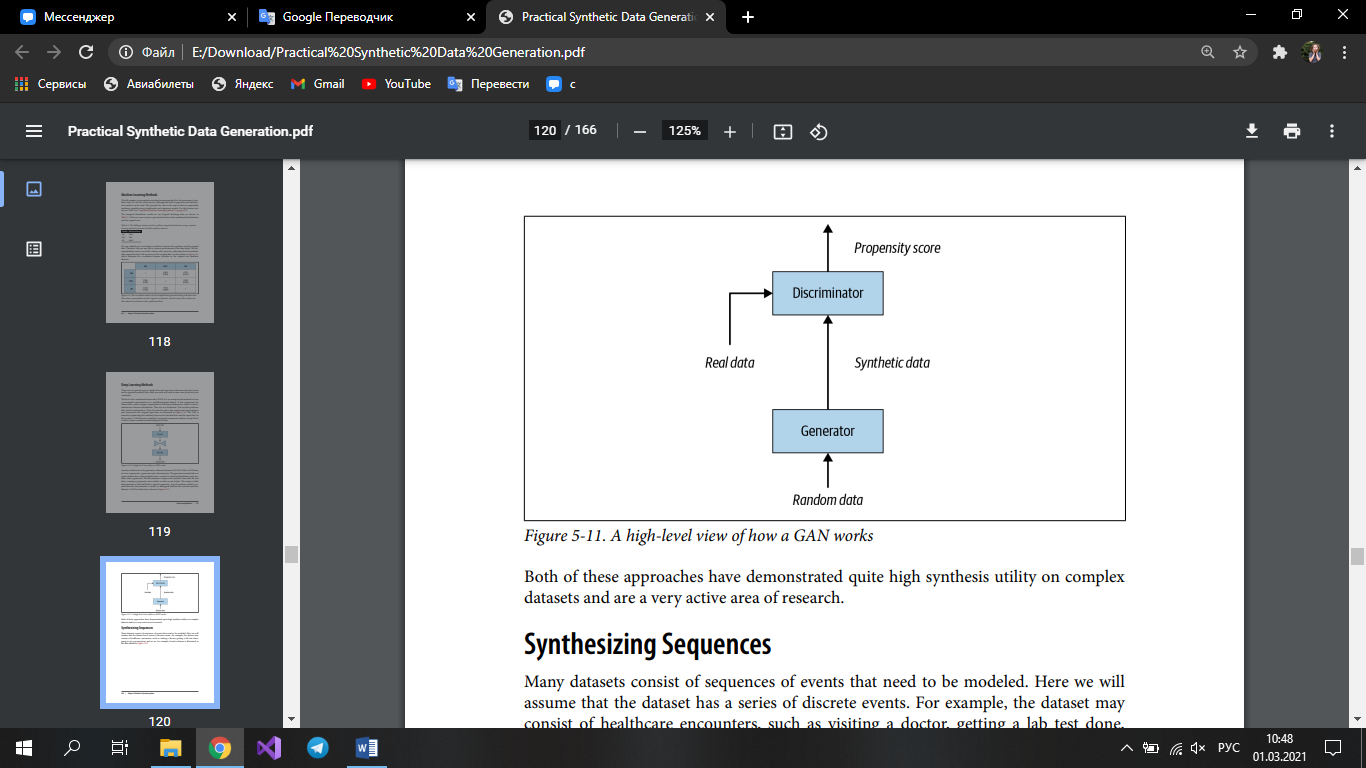
**Методы глубокого обучения**

Существует два основных типа архитектур искусственных нейронных сетей, которые используются для генерации синтетических данных. Оба могут хорошо работать, а в некоторых случаях их можно было комбинировать.

Первый - вариационный автоэнкодер (VAE). Это неконтролируемый метод изучения осмысленного представления многомерного набора данных. Сначала он сжимает набор данных в более компактное представление с меньшим количеством измерений, которое часто является многомерным распределением Гаусса. Это действует как узкое место. Кодировщик выполняет это начальное преобразование. Затем декодер берет это сжатое представление и восстанавливает исходные входные данные, как показано на рисунке 5-10. VAE обучается путем оптимизации сходства между декодированными данными и входными данными. В этом контексте функция VAE аналогична анализу главных компонентов, за исключением того, что он может фиксировать нелинейные отношения в данных.

  
*Рисунок 5-10. Общее представление о том, как работает VAE*

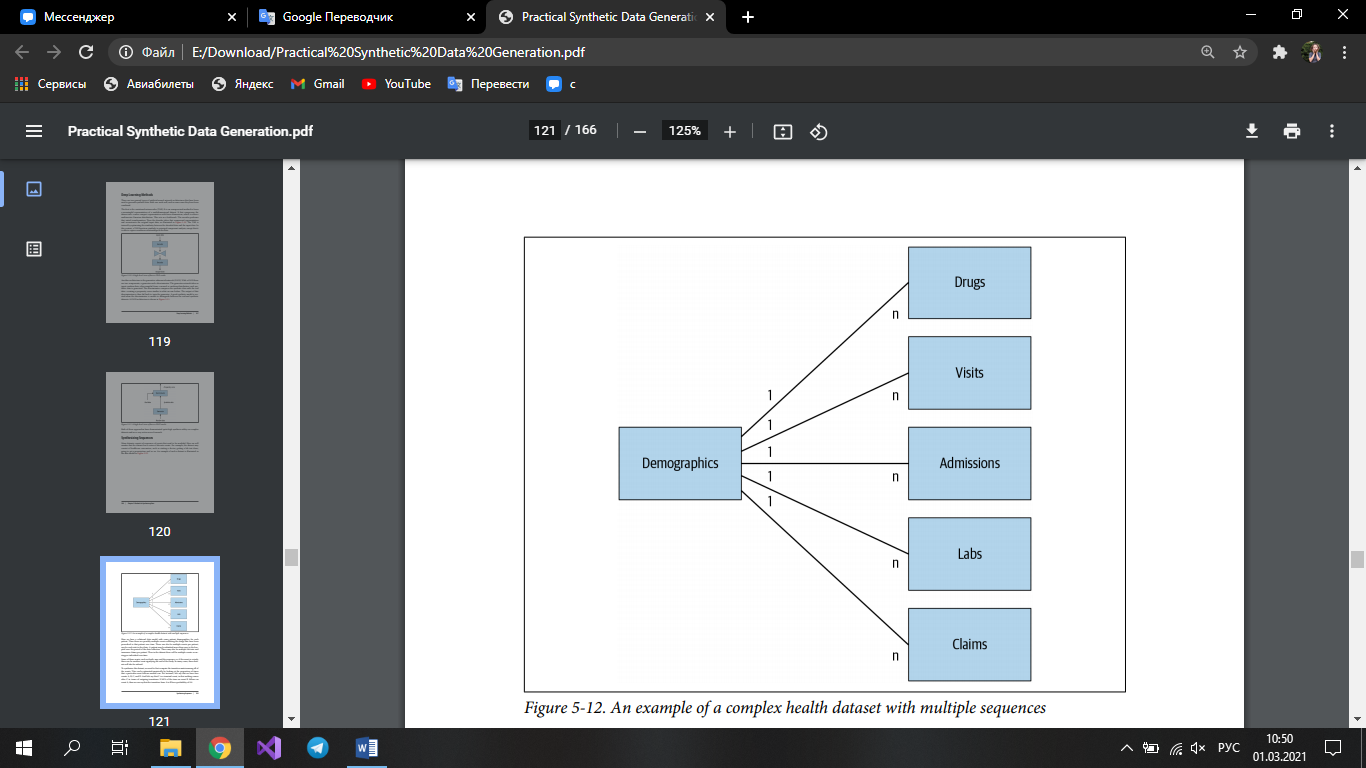
Другая архитектура - это генерирующая состязательная сеть (GAN). GAN состоит из двух компонентов: генератора и дискриминатора. Сеть генератора принимает на вход случайные данные, часто выбираемые из нормального или равномерного распределения, и генерируются синтетические данные. Дискриминатор сравнивает синтетические данные с реальными данными, создавая показатель склонности, аналогичный тому, что мы видели раньше. Затем выходной сигнал этой селективности возвращается для обучения генератора. Хорошая синтетическая модель создается, когда дискриминатор не может различить реальные и синтетические наборы данных. Архитектура GAN показана на рисунке 5-11.

  
*Рисунок 5-11. Общее представление о том, как работает GAN*

Оба этих подхода продемонстрировали довольно высокую эффективность синтеза сложных наборов данных и являются очень активной областью исследований.

**Синтез последовательностей**

Многие наборы данных состоят из последовательностей событий, которые необходимо смоделировать. Здесь мы предположим, что набор данных содержит серию дискретных событий. Например, набор данных может состоять из обращений за медицинской помощью, таких как посещение врача, прохождение лабораторных анализов, получение рецепта и т. Д. Пример такого набора данных проиллюстрирован в модели данных на рисунке 5-12.

  
*Рисунок 5-12. Пример сложного набора данных о здоровье с несколькими последовательностями*

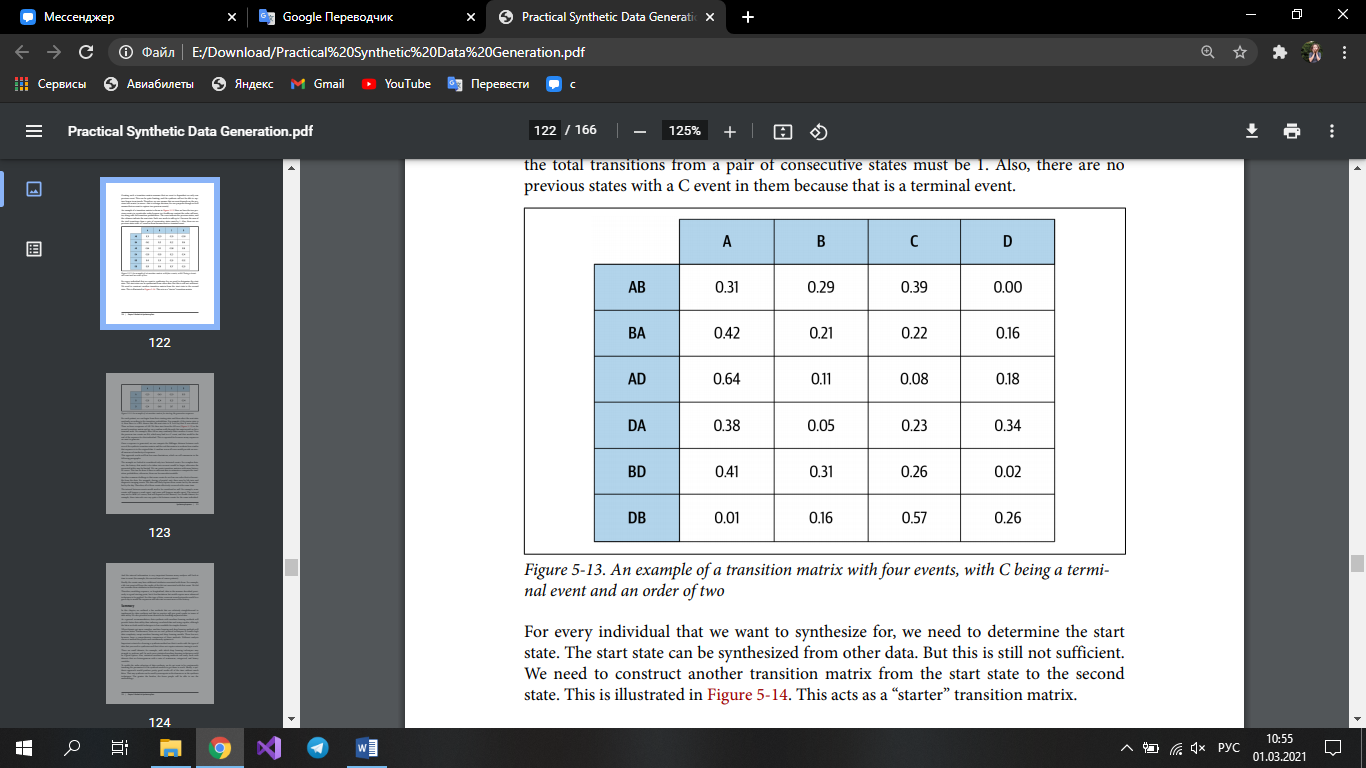
Здесь у нас есть реляционная модель данных с некоторыми демографическими данными для каждого пациента. Тогда, возможно, есть несколько событий, отражающих лекарства, которые были прописаны этому пациенту с течением времени. Также может быть несколько событий для каждого пациента, по одному на каждый визит в клинику. Пациент может быть госпитализирован более одного раза за период сбора данных. Также может быть несколько лабораторных анализов и страховых требований для каждого пациента. Таким образом, в наборе данных будет несколько событий, происходящих на человека с течением времени.

Некоторые из этих событий, например смерть, могут закончить последовательность; или, если событие является исследованием, может быть другое событие, означающее окончание исследования. Во многих случаях эти наборы данных также будут заказаны.

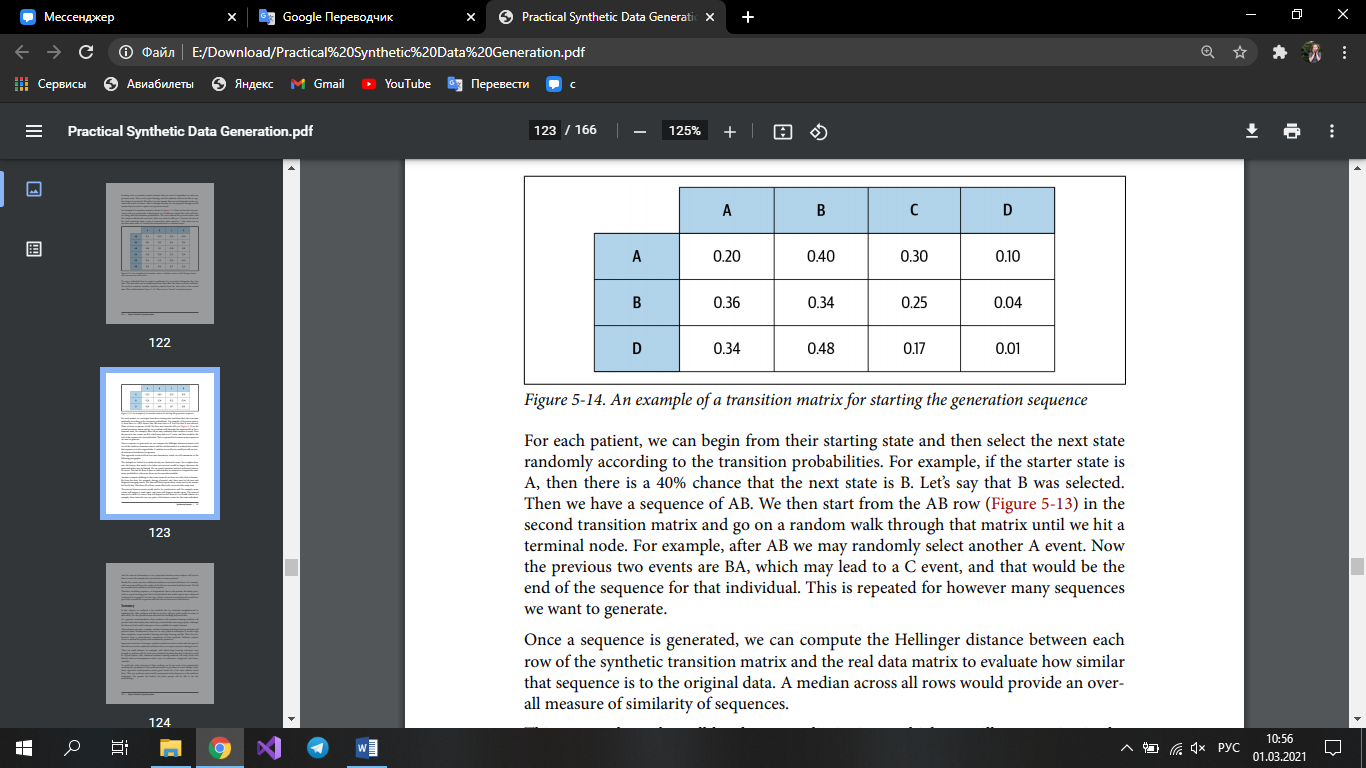
Чтобы синтезировать этот набор данных, нам нужно сначала вычислить матрицу перехода между всеми событиями. Это можно оценить эмпирически, посмотрев, сколько раз одно событие следует за другим. Например, предположим, что у нас есть четыре события A, B, C и D. И предположим, что C является конечным событием, в котором ничего не происходит после C с точки зрения исходящих переходов. Если в 40% случаев событие B следует за событием A, то мы можем сказать, что переход от A к B имеет вероятность 0,4.

Создание такой матрицы перехода предполагает, что событие зависит только от одного предыдущего события. Это может быть весьма ограничивающим фактором, и синтез не сможет уловить долгосрочные тенденции. Следовательно, мы можем предположить, что событие зависит от двух предыдущих событий (или более - это проектное решение; для наших целей мы будем предполагать, что мы хотим зафиксировать два предыдущих события).

Пример матрицы перехода показан на рисунке 5-13. Здесь у нас есть два предыдущих события в определенном порядке, потому что в контексте здравоохранения порядок имеет значение вместе с вероятностями перехода. Строки указывают предыдущие состояния, а столбцы указывают следующее состояние. В каждой строке необходимо добавить до 1, потому что сумма общих переходов из пары последовательных состояний должна быть 1. Кроме того, нет предыдущих состояний с событием C в них, потому что это конечное событие.

  
*Рисунок 5-13. Пример матрицы перехода с четырьмя событиями, где C является конечным событием и порядка двух*

Для каждого человека, для которого мы хотим синтезировать, нам нужно определить начальное состояние. Начальное состояние может быть синтезировано из других данных. Но этого все же недостаточно. Нам нужно построить еще одну матрицу перехода от начального состояния ко второму состоянию. Это показано на Рисунке 5-14. Это действует как «стартовая» переходная матрица.

  
*Рисунок 5-14. Пример матрицы перехода для запуска последовательности генерации*

Для каждого пациента мы можем начать с его начального состояния, а затем случайным образом выбрать следующее состояние в соответствии с вероятностями перехода. Например, если начальное состояние - A, то с вероятностью 40% следующим состоянием будет B. Допустим, было выбрано B. Тогда у нас есть последовательность AB. Затем мы начинаем со строки AB (рис. 5-13) во второй матрице перехода и идем случайным образом по этой матрице, пока не дойдем до конечного узла. Например, после AB мы можем случайным образом выбрать другое событие A. Теперь два предыдущих события - это BA, которые могут привести к событию C, и это будет концом последовательности для этого человека. Это повторяется для любого количества последовательностей, которые мы хотим создать.

После того, как последовательность сгенерирована, мы можем вычислить расстояние Хеллингера между каждой строкой синтетической матрицы перехода и реальной матрицей данных, чтобы оценить, насколько эта последовательность похожа на исходные данные. Медиана по всем строкам обеспечит общую меру сходства последовательностей.

Этот подход работает хорошо, но имеет некоторые ограничения, которые мы кратко изложим в следующих параграфах.

В рассмотренном нами примере учитывались только два исторических события. Для сложных наборов данных история, которую необходимо учитывать, будет больше, в противном случае сгенерированная полезность может быть ограничена. Конечно, мы можем создавать матрицы переходов с более подробной историей. Это можно сделать, если имеется достаточно данных для оценки или вычисления вероятностей перехода; в противном случае они могут быть несколько нестабильными.

Другой распространенной проблемой является то, что некоторые события не имеют порядка, который можно различить из данных. Например, во время визита в больницу могут проводиться лабораторные анализы и мероприятия по диагностической визуализации. Данные, вероятно, будут фиксировать эти события не по минутам, а по дням. Следовательно, все эти события фактически произошли одновременно.

Также необходимо учитывать интервал между событиями. Например, некоторые события будут происходить с разницей в неделю, а некоторые - с разницей в несколько месяцев. Интервал не может быть фиксированным (конечно, это будет зависеть от набора данных). Например, в наборе данных о здоровье эти интервалы могут довольно сильно различаться между событиями для одного и того же человека.

И информация об интервале очень важна, потому что многие анализы будут смотреть на время до события (например, время выживания больных раком).

Наконец, с событиями могут быть связаны дополнительные атрибуты. Например, событие лабораторного тестирования будет иметь результаты лабораторного теста, связанные с этим событием. Мы не рассматривали эти атрибуты в этом описании.

Следовательно, последовательность моделирования или продольные данные описанным выше способом являются хорошей отправной точкой, но у нее есть ограничения, которые потребуют применения более продвинутых методов. Для этого типа данных рекуррентные нейронные сети были бы хорошим способом смоделировать последовательности и учесть больше истории.

Резюме

В этой главе мы обрисовали в общих чертах несколько методов, которые относительно просто реализовать для синтеза данных и которые на практике дадут хорошие результаты с точки зрения полезности данных. Мы также предоставили некоторые инструкции по обработке последовательных данных.

В качестве общей рекомендации синтез данных с помощью методов машинного обучения обеспечит лучшую полезность данных, чем получение коррелированных данных и использование копул, хотя оба последних являются полезными методами, доступными для более простых наборов данных.

Когда наборы данных станут более сложными, методы машинного обучения и глубокого обучения будут работать лучше. Кроме того, не существует реальных практических методов обработки данных высокой сложности, кроме моделей машинного обучения и глубокого обучения. Однако всестороннего сравнения этих методов не проводилось. Разные аналитики выбирают метод, который им больше нравится, и постоянно его оптимизируют.

Важным критерием выбора метода синтеза является то, что он работает с типами данных, которые необходимо синтезировать, и не требует обширной настройки для работы.

Например, есть небольшие наборы данных, с которыми методы глубокого обучения могут плохо работать. В таких случаях хорошим вариантом могут быть методы статистического машинного обучения. Кроме того, методы статистического машинного обучения могут легко работать с наборами данных, которые неоднородны и содержат непрерывные, категориальные и двоичные переменные.

Чтобы обеспечить более широкое внедрение синтеза данных, мы не хотим постоянно настраивать параметры моделей синтеза, чтобы заставить их работать. В идеале, синтезирующий подход всегда давал бы довольно хорошие результаты без особых усилий. Таким образом, синтез может быть использован неспециалистами в области или в методах синтеза. Чем больше нагрузка, тем меньше людей сможет использовать методологию.

В следующей главе мы рассмотрим другую сторону реестра: конфиденциальность. Хотя мы можем создавать высокоэффективные данные, также важно обеспечить управление рисками конфиденциальности. Гарантия конфиденциальности - важная возможность при синтезе данных. В сегодняшней нормативно-правовой среде ответственность перед организацией может быть значительной, если она использует синтетические данные, как если бы это не личные данные, а затем обнаруживает, что риски конфиденциальности по-прежнему высоки.