КЛИНИЧЕСКИЕ ИССЛЕДОВАНИЯ
CLINICAL TRIALS

УДК: 53.087.7+004.658
DOI: 10.37489/2588-0519-2024-2-80-90
EDN: MCYGYH

НАУЧНО-МЕТОДИЧЕСКАЯ СТАТЬЯ
SCIENTIFIC METHODOLOGY ARTICLE

Организация данных медико-биологических исследований

© *Марцинкевич А. Ф.*

УО «Витебский государственный ордена Дружбы народов медицинский университет», Витебск, Республика Беларусь

Аннотация. В статье обсуждается важность корректной организации данных, необходимой для качественного статистического анализа. Приводятся основные проблемы, возникающие при регистрации результатов исследования из-за разнородности информации и несовершенства стандартных подходов, а также способы их решения и предотвращения. Предлагаются простые принципы формирования базы данных исследования, валидации и обеспечения целостности.

Ключевые слова: организация исследования; управление данными; база данных; валидация

Для цитирования: Марцинкевич А. Ф. Организация данных медико-биологических исследований. *Качественная клиническая практика*. 2024;(2):80–90. https://doi.org/10.37489/2588-0519-2024-2-80-90. EDN: MCYGYH.

Поступила: 02.04.2024. **В доработанном виде:** 17.04.2024. **Принята к печати:** 04.06.2024. **Опубликована:** 25.06.2024.

Medical and biological research data

© Aliaksandr F. Martsinkevich
Vitebsk State Order of Peoples' Friendship Medical University, Vitebsk, Republic of Belarus

Abstract. This paper discusses the importance of properly organizing data for effective statistical analysis. The main problems that arise when recording research results due to the heterogeneity of information and the imperfections of standard approaches, as well as ways to address and prevent them, are presented. Simple principles for creating a research database, validating, and ensuring its integrity are proposed.

Keywords: study organization; data management; database; validation

For citation: Martsinkevich AF. Medical and biological research data. *Kachestvennaya Klinicheskaya Praktika = Good Clinical Practice*. 2024;(2):80-90. https://doi.org/10.37489/2588-0519-2024-2-80-90. EDN: MCYGYH.

Received: 02.04.2024. **Revision received:** 17.04.2024. **Accepted:** 04.06.2024. **Published:** 25.06.2024

Введение / Introduction

Проведение исследований является важной частью любого научного изыскания, которое часто сопровождается накоплением большого количества данных, хранящихся на материальных или цифровых носителях. Однако вследствие разнородности поступающей информации конечная форма представления результатов исследования может быть далека от оптимальной, чему способствует оторванность биостатистика от дизайна базы данных, особенно если привлекаются сторонние специалисты.

Вместе с тем, ответственный подход к проведению исследования ещё до его начала снижает вероятность ошибок, упрощает анализ данных и обеспечивает получение качественных и надёжных результатов.

Следование простым принципам, изложенным в настоящей статье, поможет как при формировании базы данных исследования, так и во время проведения статистической обработки.

Извлечение, трансформация, загрузка / Extract, transform, load

Результаты исследования, зафиксированные в электронном или бумажном виде, представляют собой **исходные данные**: выписки из лабораторных журналов или индивидуальные регистрационные карты пациента, микрофотографии гистологических срезов с комментариями специалиста, а иногда и просто пометки в смартфоне.

Исходные данные могут быть разнородны по формату, структуре и способу представления содержания, поэтому их непосредственное использование для анализа невозможно. Если изначально данные регистрируются на бумажных носителях, то они переводятся в электронный вид, в ином случае агрегируются в одном месте, модифицируются и лишь затем могут быть сведены в единообразную **базу данных (БД)** исследования.

Извлечение, трансформация и загрузка консолидированных данных в **БД** может быть описана в рам-

ках ETL-процесса (extract → transform → load). ETL следует по возможности автоматизировать и строго регламентировать, чтобы обеспечить воспроизводимость операций.

Полученная БД будет использоваться для статистического анализа, поэтому при разработке её дизайна необходимо учитывать формат файла, структуру хранимых данных, типы и диапазоны вводимых значений.

Форматы файлов / File formats

Для представления данных в электронном виде, как правило, используются офисные табличные редакторы, хотя могут применяться и системы управления базами данных. Наиболее распространены следующие форматы:

Файлы CSV («*comma separated values*»), TSV («*tab separated values*») и аналогичные форматы с текстовым разделителем — наиболее простой и поэтому практически лишённый как преимуществ, так и недостатков формат. Нет необходимости иметь специальную программу для чтения, может быть отредактирован как в *Notepad*, так и в *MS Word* или *Libre Office*. Возможна путаница при чтении без дополнительных настроек, так как разделителем целой и дробной части числа в русскоязычных странах является запятая, которая также выступает в качестве разделителя для CSV по умолчанию.

Файлы XLS/XLSX, ODF — генерируются соответствующими табличными процессорами. Имеют мощные дополнительные возможности обработки, форматирования и представления данных, позволяют сохранять разные наборы данных в одном файле на так называемых «листах», однако вследствие разнообразия существующего программного обеспечения имеется вероятность того, что пользователи с иным офисным пакетом не смогут получить корректный доступ к указанным дополнительным функциям. Кроме того, вследствие автоматического форматирования (которое в случае *MS Excel* невозможно отключить), есть возможность некорректной интерпретации вводимых данных, что для русскоязычного сообщества особенно актуально в плане преобразования чисел в даты. Так как в операционной системе Windows для региона «Россия» в качестве десятичного разделителя используется запятая, в ячейке с установленным по умолчанию форматом «Общий» *MS Excel* превращает значение «10.2» в дату «10. фев». Эта проблема определённым образом касается и англоязычных пользователей, что выливается в весьма курьёзные случаи. Так, в 2020 году Комитет по номенклатуре генов (HGNC) внёс изменения в названия 27 человеческих генов, что было вызвано автоматическим форматированием *MS Excel* — гены «MARCH1» и «SEPT9», например, преобразовывались в даты «1 марта» и «9 сентября». Поэтому исследователю необходимо проявлять внимательность, предварительно выставляя формат данных, либо используя средства проверки данных.

Кроме того, табличные процессоры могут испытывать определённые трудности при работе с большими файлами, количество наблюдений в которых превышает несколько миллионов. Тем не менее, несмотря на указанные проблемы и на то, что *MS Excel* распространяется на коммерческой основе, подавляющее большинство результатов сохраняется именно в этом формате.

Файлы баз данных (*SQLite*, *MySQL*, *MS Access*) — имеют наибольшие возможности в плане корректного хранения и представления результатов, однако, как правило, требуют для создания и использования определённой квалификации.

Удобным может быть приобщение к результатам исследования также **пояснительной записки**, описывающей структуру данных или содержащей иную важную для исследователя информацию, что облегчает возврат к работе спустя некоторое время.

Трансформация данных / Data transformation

Рассматривая структуру двумерных таблиц, можно выделить два формата, которые используются для представления содержимого: длинный и широкий. Широкий формат использует для хранения переменных отдельные колонки (одна колонка соответствует одной переменной), в то время как длинный формат предполагает выделение для переменных и наблюдений как минимум двух столбцов (например, наименование параметра и его значение), сопровождаемых уникальным идентификатором. Впрочем, в некоторых случаях может быть удобно частичное использование обоих способов (рис. 1).

Наибольшую распространённость имеет широкий формат, который используется для представления данных визитов, когда каждый из показателей регистрируется для одного пациента один раз, в то время как длинный может быть полезен, если неизвестно, сколько раз будет необходимо осуществить ввод. Например, при записи сопутствующей терапии рекомендуется отдать предпочтение длинному формату, так как нельзя ожидать, что все пациенты будут принимать одинаковое количество лекарственных препаратов.

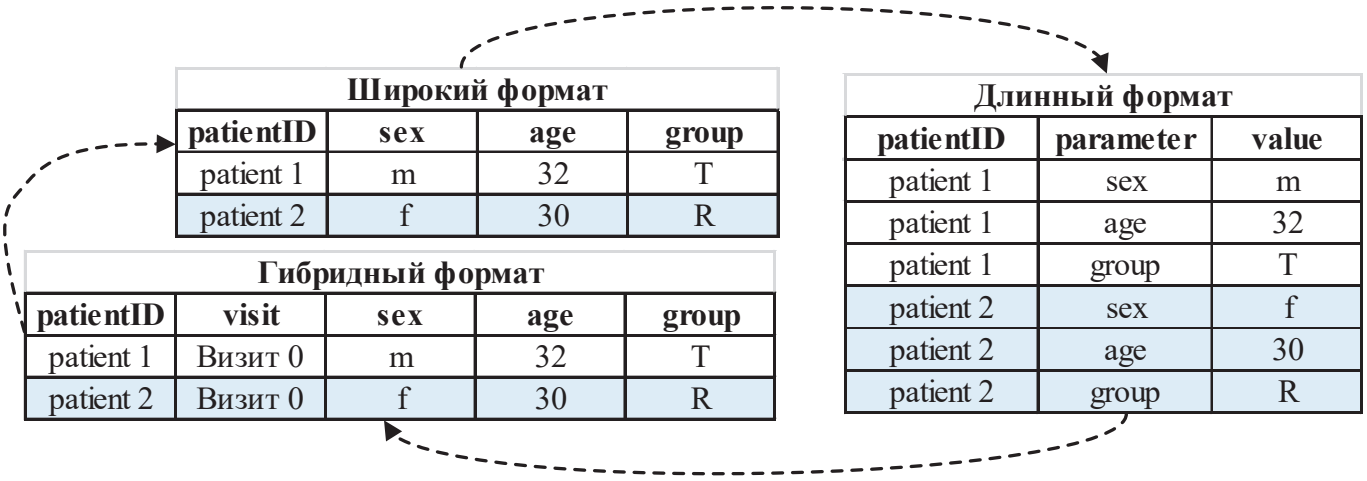

Рис. 1. Представление данных в разных форматах
Fig. 1. Data representation in various formats

Очистка / Tidying

Структурирование данные облегчает их последующую консолидацию (соединение таблиц) и очистку. Под *очисткой* (англ. *tidying*), то есть приведением данных к так называемому «опрятному» виду, подразумевается не столько удаление какой-либо информации, сколько преобразование их к некоторому стандартному состоянию.

Определение «опрятным данным» наиболее полно даёт *Jeff Leek* в работе с одноимённым названием [1]. Так, в наборе (таблице) опрятных данных:

1. Каждому исследуемому параметру (переменной) соответствует свой столбец (колонка).
2. Каждому наблюдению (случаю) соответствует своя строка.
3. Каждому значению соответствует своя ячейка.
4. Каждому исследованию соответствует один набор данных, который может быть однозначно связан с иным набором.

Пункт 4 предполагает введение для каждого из наблюдений уникального идентификатора, посредством которого можно отследить значения параметров, полученных для этого наблюдения в рамках различных исследований. Наглядный пример приведён на рис. 2.

Размещение переменных в столбцах, а наблюдений в строках не является обязательным, но, так как количество наблюдений, как правило, превышает количество изучаемых показателей, проще работать с таблицей, растянутой вертикально, а не горизонтально. Кроме того, в табличном редакторе *MS Excel* существует ограничение на количество столбцов и строк — 16384 и 1048576 соответственно, что в большинстве исследований будет достаточным, однако в некоторых случая способно привести к потере данных [2].

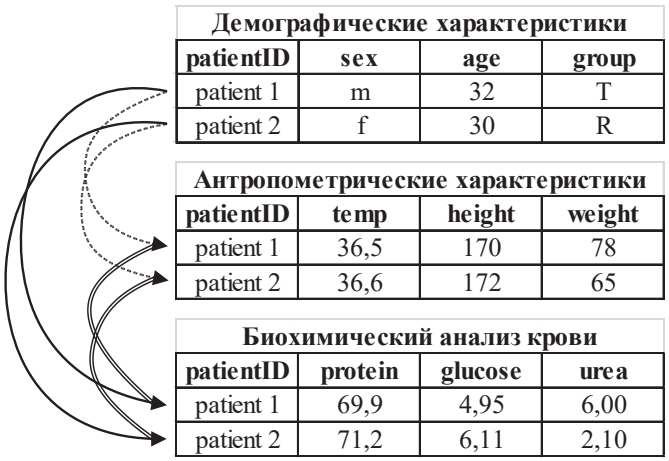

Рис. 2. Графическое представление трёх наборов данных, соответствующих концепции «опрятных» данных
Fig. 2. Graphical representation of three data sets corresponding to the concept of tidy data

В случае использования нескольких наборов данных или одного набора, разделённого на части (файлы), особое внимание следует уделить идентификаторам — в простейшем случае порядковым номерам, однозначно идентифицирующим субъект исследования. Если исследование выполняется одним исполнителем, то создание идентификаторов с единой логикой не представляется сложным, но результаты, как правило, собираются из нескольких источников, каждый из которых для идентификации может использовать свою систему. При регистрации образцов в лаборатории им присваиваются собственные номера, сквозные в рамках учётного журнала, порядковые за отчётный период

и так далее. Следует отметить, что, если исследуемые образцы в рамках одной и той же лаборатории анализируются структурными подразделениями последовательно, в каждом из подразделений могут также применяться разные идентификаторы. Кроме того, некоторые приборы могут пользоваться собственной системой кодирования образцов, основанной на предустановленных принципах и иногда защищённой от изменения во избежание фальсификации и фабрикации данных. При использовании данных от аналогичных структур (например, в случае многоцентровых исследований), идентификаторы клинических баз могут иметь сопоставимую систему кодирования, быть уникальны в пределах собственного центра, но совпадать друг с другом. Хорошей практикой будет фиксация таких промежуточных идентификаторов в отдельном файле, что может помочь в дальнейшем для уточнения данных или в случае проверки надзорными органами. Не требуется упоминать о том, что корректное сопоставление идентификаторов является критически важным при введении результатов в базу данных.

В простейшем случае идентификатором может выступать порядковый номер субъекта исследования, что, как было упомянуто выше, иногда трудно реализуемо или сопряжено со сложностями. Кроме того, иногда удобно кодировать в идентификаторе дополнительную информацию: номер исследовательского центра (идентификатор «2–10» соответствует десятому пациенту, зарегистрированному во второй клинической базе), группу («О-1» или «К-1» могут соответствовать первому пациенту из опытной или контрольной группы соответственно), номер визита пациента к врачу («Д10-2» — визит второго пациента на десятый день исследования), номер истории болезни («С001@0000» — первый пациент контрольной группы, номер истории болезни 0000) и прочее.

При использовании офисных табличных редакторов необходимо избегать некоторых плохих практик организации данных:

1. Объединённые ячейки. Очень часто используемый элемент форматирования, который практически никогда не может быть корректно обработан (объединение является визуальным эффектом, а в действительности значение хранится в первой ячейке).

2. Несколько значений в одной ячейке. Часто совмещение нескольких значений в одной ячейке является особенностями экспериментальных методик (например, определение площади поражения волосистой части головы достаточно субъективно и выражается не в конкретном процентном значении, а в интервале (10–20%); некоторые методики полуколичественного определения — экспресс-тесты для определения кетоновых тел, белка или pH мочи — также представляют результат в некотором диапазоне), однако это не отменяет того факта, что подобные данные критически сложно обрабатывать. Совмещения двух показателей (например, пола и возраста — «м40» или «жен30») в одном столбце, безусловно, следует избегать — решением может быть введение двух переменных («пол» и «возраст»).

3. Данные с разными единицами измерения в пределах одного столбца. Рядовая ситуация, при которой масса определяемого соединения может выражаться в микро- или миллиграммах, а время как в секундах, так и в минутах. Также следует отметить, что кодирование некоторых единиц измерения, например, времени как «1 минута 30 секунд» или «01:30», очевидно для человека, но имеет сложности при последующей обработке — разумным решением может быть приведение к виду «1.5» (минут) или «90» (секунд). Следует также помнить, что использование различных единиц измерения в различных наборах данных в рамках одного исследования приводит к дополнительным трудностям — кодирование времени плавания крыс в опыте А в секундах, а в опыте В в минутах может быть логично непосредственно в ходе измерения, но при анализе данных этот момент легко упустить. К сожалению, несмотря на то, что данная проблема может быть достаточно просто решена, последствия присутствия подобных ошибок значительны.

4. Посторонние данные. Настоящий бич офисных табличных редакторов, которые пытаются дать пользователю максимально широкий набор возможностей. Как следствие, в файле с данными могут возникнуть результаты вычисления среднего значения, среднеквадратическое отклонение, результаты применения статистических критериев и прочее. Исследователь должен особенно чётко понимать, что хранение и анализ данных являются созависимыми, но структурно разделёнными процессами, поэтому подобное поведение недопустимо, так как, давая сиюминутное удобство для исследователя, сводит к минимуму возможность корректного импорта данных в сторонние программы.

5. Использование цвета в качестве идентификаторов. Действительно, иногда очень удобно использовать выделение цветом для навигации в табличном редакторе и визуального разделения групп или переменных, но это абсолютно неприемлемо как способ

хранения информации — как минимум вследствие того, что при экспорте данных информация о цветовом выделении может быть утеряна.

6. Текстовые комментарии, размещённые в произвольном месте. Табличные процессоры по умолчанию предоставляют много места, лишь часть из которого непосредственно используется для хранения данных. При непреодолимом желании делать какие-либо пометки относительно наблюдений, лучшей практикой будет выделение для этих целей дополнительного столбца (переменной «Примечание»). Вместе с тем, если комментарий касается не столько конкретного наблюдения (строки), сколько всей совокупности результатов, такие пометки следует перенести в пояснительную записку.

7. Пропущенные значения, закодированные произвольным образом. Редкое исследование может похвастаться отсутствием пропусков в данных, что в целом представляет собой штатную ситуацию. Трудности возникают тогда, когда пропущенные данные начинают кодироваться самыми разными способами, будь то пустая строка (отсутствие каких-либо значений) или прочерк, текстовая пометка («NA», «НД», «нет данных», «не доступно») или «0». Нужно чётко понимать: пустая строка и ноль пусть и имеют с точки зрения стороннего наблюдателя одинаковое информационное наполнение, но зачастую не могут быть взаимозаменяемыми — так, например, в случае кодирования результатов биохимического анализа пустая строка отнюдь не всегда означает нулевую концентрацию определяемого вещества, а может предполагать также, что забор крови или само исследование выполнены не были.

8. Опечатки и ошибки. Чаще всего возникают при ручном вводе данных, могут встречаться при наборе фамилий испытуемого, при указании инициалов, а также точек после инициалов. Порой встречаются очень парадоксальные варианты, когда при наборе данных лаборантом вместо цифры «3» была введена буква «З» (на клавиатуре ноутбука оба символа находились на одной клавише). Данная проблема в значительной степени зависит от шрифта и крайне сложно может быть отслежена без строгой типизации переменной. В другом случае при вводе времени забора биологического образца вместо значения «1440» (время в минутах, соответствующее 24 часам) было ошибочно введено «1140» — опечатка обнаружилась постфактум, так как при переводе в часы получалось не выделяющееся целое значение — «19».

9. Разнородное представление идентичных данных. Как правило, отмеченная проблема возникает в случае исследований, выполняемых несколькими исполнителями, а также работ, растянутых на продолжительный период. Наиболее яркий пример такой ошибки — кодирование пола испытуемых как «женский», «жен» или «ж» в пределах одной переменной. Указанная проблема характерна для биномиальных (дихотомических) типов данных, когда, например, наличие признака может кодироваться единицей, символом плюса, текстовыми пометками «да», «есть» и прочее. Сюда же следует отнести и использование букв различных регистров: «Диабет» и «диабет» несут аналогичную смысловую нагрузку, но при машинной обработке данных будут считаться различными терминами.

10. Дополнительные пробелы и иные неотображаемые символы. Весьма часто к текстовым данным, а иногда и к числовым, примешиваются неотображаемые символы (пробелы, символы табуляции и перевода строки), которые пользователь не может отследить без дополнительных манипуляций. В случае, например, фамилий пациентов, следует понимать, что «Иванов И.И.» и «Иванов И. И.» могут быть внешне неотличимы, но при машинной обработке будут считаться различными. Также зачастую можно встретить двукратный ввод пробела между словами, например, в словосочетаниях:

- «артериальная гипертензия»
- «артериальная гипертензия»

Вручную эта ошибка исправляется кропотливой посимвольной проверкой введённых значений, при использовании средств валидации может быть обнаружена и исправлена автоматически, что, однако, требует наличия указанных средств — как правило, они не входят в стандартную поставку пакета анализа данных и создаются индивидуально под запрос конкретного исследователя.

11. Использование неправильного разделителя для целой и дробной части числа. В русскоязычных странах, как правило, для отделения целой части числа от дробной используется запятая, в то время как в западных странах для этой цели служит точка. Вместе с тем, вследствие невнимательности достаточно просто перепутать разделители и внести в данные дополнительную неразбериху.

12. Хранение данных в заголовках. Встречается при разделении некоторой переменной на интервалы или при кодировании мультиномиальных переменных. Например, исследователь может выделить в заголовки возраст пациента («<20», «21–30», «>30») или полиморфизм генов («С» или «Т»), делая в ячейку пометку при соответствии условию («да/нет» или «+/-»). Технически такие данные нарушают принцип

«опрятности», так как одна переменная «размазывается» по нескольким заголовкам. Решением может быть введение переменной, содержащей возможные значения — например переменной «возраст», которая содержит значения <20», «21–30», «>30» или переменной «полиморфизм» в которой делается соответствующая отметка — «C» или «T»; фактически же этот приём представляет собой трансформацию из широкого в длинный формат.

13. Именование заголовков столбцов (переменных). В некоторых программных продуктах даже в настоящее время сохраняются определённые трудности с воспроизведением кириллицы, что может приводить к ошибкам, если данную кодировку использовать в заголовке столбца («пол», «возраст»). Оптимальным является использование латиницы и терминов на английском языке или транслите («sex», «age» или «pol», «vozrast» — справедливости ради следует отметить, что применение транслита пусть и допускается, но в определённой степени порицается). Английский язык не является непреложным условием, но предпочтителен, так как в большинстве слов не содержит символов с диакритическими знаками, которые в случае немецкого (*умляут* ä) или французского (*седиль* ç) языка могут также привести к непредвиденным результатам при именовании заголовков столбцов.

Вследствие непредсказуемого поведения следует также избегать в заголовках использования пробелов и специальных символов — косой черты, вопросительных и восклицательных знаков, символов процента и амперсанда, двоеточия, скобок и прочих. Для визуального разделения слов в заголовке предпочтительно использовать чередование прописных и строчных букв (так называемый *PascalCase* или *camelCase* — «GlucoseBlood» или «glucoseBlood» соответственно), символа нижнего подчеркивания («glucose_blood») или точки («glucose.blood»). Не рекомендуется использовать *kebab-case* (glucose-blood) и *Train-Case* (Glucose-Blood), которые как минимум будут преобразованы к точечной нотации или вызовут ошибку.

Кроме предотвращения ошибок, связанных с импортом, именование столбцов при осмысленном подходе поможет значительно облегчить последующую обработку данных. Очевидно, что названия должны кратко, но полно отражать значение показателя: столбцы «g1» и «g2» для исследователя могут означать многое, но быть полностью бессмысленными для коллег (а спустя некоторое время и для самого исследователя). Например, «glucose_1h» и «glucose_2h» ненамного длиннее, но нативно подсказывают, что показатели могут соответствовать уровням глюкозы спустя 1 и 2 часа от какой-то временной точки. Если существуют несколько источников данных или иных группирующих факторов (различные биологические среды, ткани или точки отсчёта) целесообразно использование более сложных названий, например «blood.glucose», «blood.cholesterol» или «blood.albumin». Обратите внимание, что при таком подходе стоит придерживаться иерархичности: первое место занимает наиболее общая или представляющая наибольший интерес характеристика. Так, если объектом исследования является липидный состав липопротеиновых комплексов крови, то есть в каждом из комплексов (высокой, низкой и очень низкой плотности) происходит определение содержания холестерола, фосфатидилхолина, фосфатидилэтаноламина и прочих фосфолипидов, названия показателей могут быть следующие: «HD.CHS», «HD.PCH», «HD.PEA», «LD.CHS», «LD.PCH» и далее. Однако, если объектом исследования является непосредственно содержание холестерола в различных формах, в том числе и общего холестерола, более логично вынести определяемое вещество на первое место: «cholesterol.HD», «cholesterol.LD», «cholesterol.VLD» и «cholesterol.total».

Дубликаты наблюдений (строк). Примитивная ошибка, возникающая при повторном введении данных для субъекта исследования. Без использования идентификаторов выявляется крайне трудно. В случае использования нескольких наборов данных и объединения их по идентификаторам приводит к дублированию строк в итоговой таблице (см. рис. 3).

Во избежание подобных ситуаций возможно введение в этап очистки данных дополнительной процедуры поиска дубликатов или специальных условий соединения таблиц.

Некорректное представление отсутствующих наблюдений (строк). При исследовании динамики показателя во времени он многократно измеряется через некоторые интервалы. Однако по непредсказуемым причинам не все из испытуемых могут посетить врача-исследователя или некоторые из подопытных животных могут в промежутках между исследованиями погибнуть; в таком случае пропущенными данными становится не один из неудавшихся анализов, а вся строка, отвечающая за это наблюдение. Перед исследователем возникает дилемма: оставить отведённую выбывшему субъекту строку без результатов или же удалить её полностью (см. рис. 4).

patientID	sex	age	group
patient 1	m	32	T
patient 2	f	30	R

patientID	temp	height	weight
patient 1	36,5	170	78
patient 1	36,5	170	78
patient 2	36,6	172	65

patientID	sex	age	group	temp	height	weight
patient 1	m	32	T	36,5	170	78
patient 1	m	32	T	36,5	170	78
patient 2	f	30	R	36,6	172	65

Рис. 3. Дублирование наблюдений при объединении двух наборов данных
Fig. 3. Duplicate observations when combining two data sets

patientID	day	protein
patient 1	1	69,9
patient 2	1	71,2
patient 3	1	68,4

patientID	day	protein
patient 1	2	71,1
patient 2	2	
patient 3	2	69,6

patientID	protein_1	protein_2
patient 1	69,9	71,1
patient 2	71,2	
patient 3	68,4	69,6

patientID	day	protein
patient 1	2	71,1
patient 3	2	69,6

patientID	protein_1	protein_2
patient 1	69,9	71,1
patient 2	71,2	69,6

Рис. 4. Обработка отсутствующих наблюдений
Fig. 4. Handling missing observations

Возникшая неоднозначность не является проблемой, но требует отдельного внимания. Если исследователь примет решения не вносить в базу данных пустую строку, соответствующую пропущенному наблюдению, при анализе следует особенно тщательно проверить соответствие идентификаторов наблюдений в различные периоды друг другу — иначе, согласно приведённому на рис. 4 примеру, результатам пациента 2 во второй день будут сопоставлены данные пациента 3, а данные пациента 3, в свою очередь «исчезнут». Как правило, трансформации подобного рода производятся инструментом анализа данных автоматически и широко внедрены, но не будет лишним заранее знать, способно ли ваше программное обеспечение на подобные манипуляции.

Если исследователь введёт в базу данных пустую строку каких-либо дополнительных действий не потребуется, однако в случае большого количества наблюдений (сотни, тысячи и десятки тысяч) навигация по такой базе может быть затруднительной. Универсального совета по разрешению подобных ситуаций не существует, однако следует заметить, что нужно заранее быть готовым к такому развитию событий, когда регистрация отсутствующих наблюдений в виде пустых строк будет невозможна — в случае незапланированных визитов или досрочного завершения исследования вследствие каких-либо причин. Кроме того, умение использовать идентификаторы при соединении данных двух таблиц упростит ситуации, когда одному субъекту соответствует несколько

наблюдений — например при учёте данных анамнеза или регистрации нежелательных явлений.

Использование специализированных средств управления базами данных избавляет от ошибок подобного рода, однако, требует дополнительного обучения для начала работы.

Хорошей практикой является выделение в отдельный текстовый файл **описания процедуры очистки**, чтобы при необходимости спустя некоторое время можно было без особых усилий её повторить или воспроизвести на ином, но имеющем сходную структуру наборе результатов.

Валидация / Validation

Кроме упомянутой выше необходимости приведения базы данных в «опрятный» вид, наиболее распространёнными причинами для очистки данных является небрежность в работе и некорректное или ошибочное применение программного обеспечения, используемого для формирования базы результатов исследования.

Переменные в базе данных могут подвергаться различным проверкам:

- проверка на уникальность (важно для идентификаторов, особенно в исследованиях с повторными измерениями);
- соответствие типа (предотвращает запись текста в числовую переменную);
- корректность кодирования (если было оговорено, что пол кодируется как «муж» и «жен» не допускается введение «м», «мужчина», «жен», «female» и прочее; осуществляется путём сверки со «словарём», который как правило включает до нескольких десятков терминов);
- вхождение в диапазон (результаты лабораторных анализов в подавляющем большинстве должны быть положительны, а, например, температура не может превышать 40–50 °C, таким образом значение 366, полученное вследствие пропуска десятичного разделителя, будет автоматически помечено как ошибочное);
- соответствие формату (важно для дат и времени, но может использоваться и для других строго регламентированных показателей, когда размер «словаря» слишком велик — при кодировании сопутствующей терапии по АТХ или заболеваний по MedDRA);
- проверка консистентности (даты последовательных визитов должны быть хронологичны; если стоит пометка о том, что общий анализ крови выполнен — должны быть приведены его результаты; тест на определение беременности не должен быть отмечен как выполненный у биологических мужчин);
- проверка на наличие значения (пропуски в данных могут встречаться, если пациент не явился на визит или образец был утерян, но отсутствие значения глюкозы, при условии, что белок и мочевина известным скорее всего являются ошибкой);
- проверка на длину значения (может использоваться для проверки номера телефона или почтового индекса).

Упомянув проблемы, сопряжённые с использованием офисных табличных редакторов, нельзя не отметить, что многие из них содержат функции, предназначенные для валидации данных и предотвращения ошибок ввода. Например, *MS Excel* позволяет задать проверку данных и ограничить вводимые данные по следующим параметрам:

- целое или действительное число (удобно для предотвращения ввода текстовых данных, а также некорректных десятичных разделителей);
- список (перечень возможных значений устанавливается **заранее** и может быть использован для ввода пола, расы пациента, отнесения к опытной или контрольной группе);
- дата или время (позволяет предотвратить опечатки и описки);
- длина текста (для указания группы или подгруппы АТХ);
- пользовательское условие (используется для проверки сложных критериев, определяемых при помощи формул, например, уникальности вводимого значения идентификационного кода пациента — для предотвращения возникновения дубликатов; также можно ограничить ввод даты, предшествующей текущей и прочее).

Кроме того, задаются дополнительные условия: включение или невключение в диапазон, равенство или неравенство указанному значению, что может быть удобно, если результаты методики испытания или сам определяемый параметр имеет некоторый допустимый интервал — например температура тела, удельная плотность мочи или рабочий диапазон прибора.

При попытке же ввода некорректного значения пользователь увидит предупреждение, текст которого также может быть предварительно установлен, и содержать инструкции и пояснения. Существует возможность установки подсказки, отображаемой при наведении на ячейку, в которой могут быть раз-

мещены дополнительные сведения, такие как полное название параметра, единицы измерения или расшифровка закодированных значений. При применении функции проверки данных на уже введённых значениях есть возможность графически выделить ячейки, содержание которых проверку не проходит. К сожалению, следует также отметить, что проверка данных в *MS Excel* работает только при их ручном введении с клавиатуры, в случае вставки из буфера обмена заданные ограничения игнорируются.

Описанный инструмент проверки данных характерен для табличных редакторов, однако, если есть квалифицированные специалисты, то может быть реализован и на других платформах, вплоть до создания отдельных программ (скриптов *R*, *SPSS* и пр.), определяющих валидность данных.

Анализ / Analysis

Корректная БД обеспечивает удобную обработку (экспорт, тестирование статистических гипотез, представление результатов в форме таблиц и графиков), выполняемую специализированным программным обеспечением. Последовательность действий пользователя в ходе анализа данных, или **инструкции анализа данных**, могут быть сохранены в различных форматах, зависящих от инструмента, предпочитаемого исследователем (*Statistica* [3] использует файлы с расширением *.sta, *SPSS* [4] — *.sav, скрипты *R* [5] — *.R и так далее). Полученные результаты фиксируются в структурированной форме в виде **итогового отчёта**, служащего для подготовки научных статей или отчёта о проведении клинического исследования. Написание итогового отчёта является заключительным этапом процедуры анализа данных.

Блок-схема организации данных / Data organization flow chart

Таким образом общая схема извлечения, трансформации, выгрузки, анализа и интерпретации данных в исследовании может быть представлена следующим образом (рис. 5).

Виртуальное рабочее пространство исследователя представлено следующими компонентами:

1. Исходные данные (нативные файлы специализированного оборудования, а также файлы, подготовленные сотрудниками лаборатории, копии и выписки из журналов). Формат исходных данных практически не зависит от исследователя и должен отвечать единственному требованию: возможности извлечения результатов. Отдельно хотелось бы отметить, что для надёжности не будет лишним сфотографировать лист лабораторного журнала или распечатки данных прибора и в электронном виде приобщить к исходным данным.
2. Пояснительная записка к данным (описание исходных данных и способа соотнесения исходных данных посредством идентификаторов, имени файла и прочее; описание переменных, включая единицы измерения; описание цели исследования; описание дизайна исследования) в текстовом формате doc/odt/txt.
3. Описание трансформации исходных данных в БД (процедура очистки данных) в текстовом формате doc/odt/txt.
4. База данных исследования (соответствует структуре опрятных данных и, как правило, подготавливается непосредственно исследователем). Рабочий набор данных не всегда существует в виде отдельного файла — вполне возможна ситуация, когда исследователь программно выполняет процедуры консолидации и очистки исходных данных, а БД формируется прямо в оперативной памяти компьютера.
5. Инструкции анализа данных (генерируются специализированным программным обеспечением, имеют различные форматы, зависят от используемого инструмента). Могут быть как достаточно примитивными, осуществляющими воспроизведение ранее выполненных расчётов и манипуляций, так и сложными, обеспечивающими генерирование итогового отчёта с заданным форматированием, автоматическое обновление результатов при добавлении новых данных и прочее.
6. Итоговый отчёт (создаётся исследователем вручную или автоматически при наличии соответствующих шаблонов) — текстовый документ, отражающий в удобной для восприятия форме основные результаты исследования: описательную

Рис. 5. Блок-схема жизненного цикла данных
Fig. 5. Flow chart of data life cycle

статистику, протестированные гипотезы, модели, прогнозы. Так как практически всегда использует достаточно сложное форматирование, сохраняется в текстовом формате word/odt.

Целостность данных / Data integrity

После создания итогового отчёта работа исследователя не завершается, так как все полученные данные и созданные документы должны быть помещены в архив и зарезервированы. Более того, хорошей практикой будет постоянное резервное копирование всей информации, касающейся исследования на всём его протяжении.

Рекомендации по целостности данных были сформулированы в 2016 году FDA [6], впоследствии дополнены EMA [7] и описываются 10 принципами, закодированными в акрониме ALCOA++, согласно которым данные должны быть:

- A — *attributable* (соотносимые) — любая запись может быть соотнесена с уполномоченным лицом, её сделавшим;
- L — *legible* (читаемые) — записи на бумажных носителях должны быть хорошо читаемы, а сжатие, шифрование или кодирование электронных данных обратимым;
- C — *contemporaneous* (своевременные) — регистрация данных производится непосредственно после их получения, по возможности указывается также время и дата записи;
- O — *original* (оригинальные) — первичные документы сохраняются, а регистрация данных происходит в документах с заранее определёнными шаблонами;
- A — *accurate* (точные) — записи полностью идентичны полученным значениям, а электронные данные — бумажным носителям;
- + *C* — *complete* (полные) — записи не должны удаляться или модифицироваться безвозвратно, все изменения фиксируются;
- + *C* — *consistent* (постоянные) — данные регистрируются последовательно в хронологическом порядке и имеют отметку о времени события;
- + *E* — *enduring* (долговечные) — должно быть обеспечено резервное копирование и архивация данных, по возможности как локальное, так и облачное;
- + *A* — *available* (доступные) — данные должны не просто надёжно хранится, но и быть всегда доступны для обзора и проверки;
- + *T* — *traceable* (отслеживаемые) — источник данных и их изменений должны фиксироваться (например, в виде метаданных).

Системы контроля версий / Version control systems

Без специализированного программного обеспечения для управления данными принципы ALCOA++ достаточно сложно реализовать, а следование им может принести больше проблем, чем преимуществ. В качестве компромиссного решения можно использовать системы управления версиями — программы, позволяющие фиксировать и отслеживать изменения, происходящие с данными, будь то текстовые документы, электронные таблицы или изображения.

Наиболее распространён в настоящее время *Git* [8], который имеет открытый исходный код, бесплатен и представлен на различных операционных системах. *Git* отслеживает заданную директорию в файловой системе персонального компьютера и при редактировании содержащихся в ней документов фиксирует информацию о произошедших изменениях так, чтобы спустя некоторое время была возможность вернутся к сохранённым версиям. *Git* поддерживает «ветвление» версий, когда, предположим, исследователь кардинальным образом перерабатывает отчёт, сохраняя его параллельно исходному варианту, а не заменяя. Оценив результат, пользователь может объединить ветви, либо же удалить ненужные.

Немаловажно, что изменения на локальном компьютере могут быть синхронизированы с удалённым сервером в Интернет и распространены среди иных авторизованных пользователей. Существует большое количество сервисов, предоставляющих как частные, так и публичные сервера для этих целей, также возможно использование и собственного сервера. Работа посредством *Git* позволяет получить доступ к свежей версии данных из любой точки мира как непосредственно самому исследователю, так любому из его коллег или статистику. Кроме того, хранение копий в нескольких местах создаёт дополнительную страховку на случай утери документов, что может произойти абсолютно случайным образом и вне зависимости от желания исследователя (поломка компьютера, повреждение данных вредоносными программами, кража устройства).

В завершение хотелось бы привести поговорку: «*garbage in, garbage out*» (мусор на входе, мусор на выходе), которая подразумевает, что невозможно получить качественный результат при использовании некачественных данных. Применение описанных выше подходов значительно снизит вероятность ошибок и обеспечит быстрый и комфортный анализ.

ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ

Конфликт интересов
Автор заявляет об отсутствии конфликта интересов.

Финансирование
Работа выполнялась без спонсорской поддержки.

ADDITIONAL INFORMATION

Conflict of interests
The author declares no conflict of interest.

Financing
The work was carried out without sponsorship.

СВЕДЕНИЯ ОБ АВТОРАХ

Марцинкевич Александр Францевич — к. б. н., доцент, кафедра общей и клинической биохимии, Витебский государственный ордена Дружбы народов медицинский университет, Витебск, Республика Беларусь
Автор, ответственный за переписку
e-mail: argentum32@gmail.com
РИНЦ SPIN-код: 5472–8541

ABOUT THE AUTHORS

Aliaksandr F. Martsinkevich — PhD, Cand. Sci. (Biol.), associate professor, Vitebsk State Order of Peoples' Friendship Medical University, Vitebsk, Republic of Belarus
Corresponding author
e-mail: argentum32@gmail.com
RSCI SPIN-code: 5472–8541

Список литературы / References

1. Wickham H. Tidy Data. J. Stat. Soft. [Internet]. 2014 Sep. 12 [cited 2024 Apr. 2];59(10):1-23. Available from: https://www.jstatsoft.org/index.php/jss/article/view/v059i10.
2. Covid: how Excel may have caused loss of 16,000 test results in England. Доступно по: https://www.theguardian.com/politics/2020/oct/05/how-excel-may-have-caused-loss-of-16000-covid-tests-in-england. Ссылка активна на 23.03.2024.
3. STATISTICA: Data Mining, анализ данных, контроль качества, прогнозирование, обучение, консалтинг. Доступно по: http://statsoft.ru. Ссылка активна на 23.03.2024.
4. SPSS Statistics | IBM. Доступно по: https://www.ibm.com/products/spss-statistics. Ссылка активна на 23.03.2024.
5. R: The R Project for Statistical Computing. Доступно по: https://www.r-project.org. Ссылка активна на 23.03.2024.
6. Data Integrity and Compliance With CGMP Guidance for Industry. Доступно по: https://www.fda.gov/files/drugs/published/Data-Integrity-and-Compliance-With-Current-Good-Manufacturing-Practice-Guidance-for-Industry.pdf. Ссылка активна на 23.03.2024.
7. Guideline on computerised systems and electronic data in clinical trials. Доступно по: https://www.ema.europa.eu/en/documents/regulatory-procedural-guideline/guideline-computerised-systems-and-electronic-data-clinical-trials_en.pdf. Ссылка активна на 23.03.2024.
8. Git. Доступно по: https://git-scm.com. Ссылка активна на 23.03.2024.