

6-24-2016

Using a Data Quality Framework to Clean Data Extracted from the Electronic Health Record: A Case Study.

Oliwier Dziadkowiec

University of Colorado, College of Nursing, Anschutz Medical Campus, oliwier.dziadkowiec@ucdenver.edu

Tiffany Callahan

University of Colorado, Department of Pediatrics, Anschutz Medical Campus

Mustafa Ozkaynak


University of Colorado, Anschutz Medical Campus, College of Nursing

Blaine Reeder

University of Colorado, Anschutz Medical Campus, College of Nursing

See next pages for additional authors

Follow this and additional works at: <http://repository.edm-forum.org/egems>

 Part of the [Applied Statistics Commons](#), [Biostatistics Commons](#), [Databases and Information Systems Commons](#), [Health Information Technology Commons](#), [Health Services Research Commons](#), [Other Nursing Commons](#), [Statistical Methodology Commons](#), and the [Translational Medical Research Commons](#)

Recommended Citation

Dziadkowiec, Oliwier; Callahan, Tiffany; Ozkaynak, Mustafa; Reeder, Blaine; and Welton, John (2016) "Using a Data Quality Framework to Clean Data Extracted from the Electronic Health Record: A Case Study," *eGEMs (Generating Evidence & Methods to improve patient outcomes)*: Vol. 4: Iss. 1, Article 11.

DOI: <http://dx.doi.org/10.13063/2327-9214.1201>

Available at: <http://repository.edm-forum.org/egems/vol4/iss1/11>

This Methods Case Study is brought to you for free and open access by the the Publish at EDM Forum Community. It has been peer-reviewed and accepted for publication in eGEMs (Generating Evidence & Methods to improve patient outcomes).

The Electronic Data Methods (EDM) Forum is supported by the Agency for Healthcare Research and Quality (AHRQ), Grant 1U18HS022789-01. eGEMs publications do not reflect the official views of AHRQ or the United States Department of Health and Human Services.

Using a Data Quality Framework to Clean Data Extracted from the Electronic Health Record: A Case Study.

Abstract

Objectives: Examine (1) the appropriateness of using a data quality (DQ) framework developed for relational databases as a data-cleaning tool for a dataset extracted from two EPIC databases; and (2) the differences in statistical parameter estimates on a dataset cleaned with the DQ framework and dataset not cleaned with the DQ framework.

Background: The use of data contained within electronic health records (EHRs) has the potential to open doors for a new wave of innovative research. Without adequate preparation of such large datasets for analysis, the results might be erroneous, which might affect clinical decision making or results of Comparative Effectiveness Research studies.

Methods: Two Emergency Department (ED) datasets extracted from EPIC databases (adult ED and children ED) were used as examples for examining the five concepts of DQ based on a DQ assessment framework designed for EHR databases. The first dataset contained 70,061 visits, and the second dataset contained 2,815,550 visits. SPSS Syntax examples as well as step-by-step instructions of how to apply the five key DQ concepts these EHR database extracts are provided.

Conclusions: SPSS Syntax to address each of DQ concepts proposed by Kahn et al. (2012) was developed. The dataset cleaned using Kahn's framework yielded more accurate results than the dataset cleaned without this framework. Future plans involve creating functions in R language for cleaning data extracted from the EHR as well as an R package that combines DQ checks with missing data analysis functions.

Acknowledgements

We would like to thank our hospital partners for making these two data sets available to us. We would also like to thank the Office of Research at the College of Nursing for support of this work.

Keywords

Electronic Health Records, Data Quality, Relational Databases, Applied Statistics

Disciplines

Applied Statistics | Biostatistics | Databases and Information Systems | Health Information Technology | Health Services Research | Other Nursing | Statistical Methodology | Translational Medical Research

Creative Commons License



This work is licensed under a [Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 License](https://creativecommons.org/licenses/by-nc-nd/3.0/).

Authors

Oliwier Dziadkowiec, *University of Colorado, College of Nursing, Anschutz Medical Campus*; Tiffany Callahan, *University of Colorado, Department of Pediatrics, Anschutz Medical Campus*; Mustafa Ozkaynak, *University of*

Colorado, Anschutz Medical Campus, College of Nursing; Blaine Reeder, University of Colorado, Anschutz Medical Campus, College of Nursing; John Welton, University of Colorado, Anschutz Medical Campus, College of Nursing.



Using a Data Quality Framework to Clean Data Extracted from the Electronic Health Record: A Case Study

Oliwier Dziadkowiec, PhD;^j Tiffany Callahan, MPH, PhD;ⁱⁱ Mustafa Ozkaynak, PhD;^j Blaine Reeder, PhD;^j John Welton, PhD, RN, FAANⁱ

ABSTRACT

Objectives: We examine the following: (1) the appropriateness of using a data quality (DQ) framework developed for relational databases as a data-cleaning tool for a data set extracted from two EPIC databases, and (2) the differences in statistical parameter estimates on a data set cleaned with the DQ framework and data set not cleaned with the DQ framework.

Background: The use of data contained within electronic health records (EHRs) has the potential to open doors for a new wave of innovative research. Without adequate preparation of such large data sets for analysis, the results might be erroneous, which might affect clinical decision-making or the results of Comparative Effectiveness Research studies.

Methods: Two emergency department (ED) data sets extracted from EPIC databases (adult ED and children ED) were used as examples for examining the five concepts of DQ based on a DQ assessment framework designed for EHR databases. The first data set contained 70,061 visits; and the second data set contained 2,815,550 visits. SPSS Syntax examples as well as step-by-step instructions of how to apply the five key DQ concepts these EHR database extracts are provided.

Conclusions: SPSS Syntax to address each of the DQ concepts proposed by Kahn et al. (2012)¹ was developed. The data set cleaned using Kahn's framework yielded more accurate results than the data set cleaned without this framework. Future plans involve creating functions in R language for cleaning data extracted from the EHR as well as an R package that combines DQ checks with missing data analysis functions.

ⁱUniversity of Colorado, College of Nursing, Anschutz Medical Campus, ⁱⁱUniversity of Colorado, Department of Pediatrics, Anschutz Medical Campus

Introduction

Electronic health records (EHRs) are an essential element of the Learning Health System^{2,3} and offer a new angle for examining the processes and outcomes of health care delivery.⁴⁻⁵ EHRs contain electronic data from a variety of sources including registries and administrative data sets.⁶ Though EHRs have advantages over traditional data collection methods such as by phone and through paper-and-pencil surveys, their quality has been frequently questioned. Additionally, the accuracy of information within EHRs has been frequently questioned in the literature.⁷⁻⁹⁻¹⁰

Background

There are a number of reasons for the abundance of data quality (DQ) issues found in the EHRs. First, EHR implementation is still imperfect in most health care settings; McGinn and colleagues (2011) report a number of barriers related to EHR implementation including software and hardware issues, limited user knowledge, exchanges of patients between various health settings, and additional burden to administrative staff and providers—all of which affect the quality of the data contained within EHRs.¹⁰ In addition, the current design of some EHR fields (i.e., open-text or free response) increase the likelihood of coder error; the hectic workflow further increases EHR error vulnerability.^{11,12} Finally, at times, records have delayed dates, are not closed (i.e., patients lack valid discharge, administrative procedure, or inter- and intra transfer dates), or are simply inaccurate.¹³

EHR data can be either structured (numeric data that have a predetermined format), unstructured (text fields with no predetermined format) or semistructured (a combination of structured and unstructured numeric and text fields).¹⁴ Each of these data types entails differences in the following: (1) extraction methods, (2) quality evaluation standards, (3) screening procedures, and (4) cleaning

procedures prior to being ready for secondary use or statistical analysis. Structured data are the most straightforward to work with, and thus will be the focus of this case study.

Frameworks for Examining the Quality of Electronic Health Records (EHRs)

EHRs contain both a large volume and a wide variety of patient and provider data and are relatively inexpensive to acquire.¹⁵ Given the low quality of this data for statistical analysis, having a system for properly screening and cleaning in preparation for analysis is crucial.

A number of frameworks in the literature provide information on the following: (1) how to evaluate health care DQ,^{16,17} as well as (2) how to *clean data* in preparation for statistical analysis.^{16,18-20} The DQ-related frameworks were designed for examining data in relational databases rather than in flat file formats such as Microsoft Excel (.csv). These *data cleaning* frameworks were designed for data collected through surveys lacking the complex multidimensional structure of data stored in relational *databases*.

There are ongoing initiatives dedicated to the preparation of data extracted from the EHR for analysis,^{21,22} but many of these efforts focus on data validation rather than data cleaning. A very few examples in the literature provide a step-by-step guide for cleaning EHR data extracted from relational databases, and no published examples (to the knowledge of the first author) of repurposing a DQ framework for the purpose of data cleaning currently exist.

Kahn's (2012) Pragmatic Framework for Single- and Multisite Data Quality Assessment

The five key DQ concepts identified in the Kahn et al. (2012)¹ framework will be applied to understanding DQ issues within flat file data extracts. The Kahn



et al. (2012) framework was chosen because of its usefulness for examining the DQ of data stored in relational databases. The five key data concepts include the following: (1) attribute domain constraints, (2) relational integrity rules, (3) historical data rules, (4) state-dependent rules, and (5) attribute dependency rules. We address each of these concepts by assessing the degree to which it is present in our data set using SPSS Syntax. "Data set," as used in this article, refers to "a single table of rows of records (patients or events) and columns (variables and attributes)." The data sets (single table) for this analysis are a collection of information (variables and attributes) on a group of patients and their events extracted from multiple database tables from the EHR.

Context

The adult ED is part of a larger hospital system in Colorado and serves as an educational site nearly 1,800 nursing and medical students, residents, fellows, and allied health students at the University of Colorado Denver, Anschutz Medical Campus. The hospital and its affiliated clinics provide health care services to an ethnically diverse population located in the Rocky Mountain region. The patient volume at the adult ED consists of approximately 100,000 visits per year.

The pediatric ED consists of the main ED and four satellite EDs. The patient volume at the main ED consists of approximately 70,000 visits per year and the four satellite pediatric EDs account for approximately 60,000 visits per year. This pediatric ED is also a Level I Regional Pediatric Trauma Center, providing emergency care services to a large geographic area in the Rocky Mountain region. The main pediatric ED has 40 private patient care rooms and 8 observation rooms.

Clinical Significance of the Case Study

Timely administration of medication in the emergency room is critically important for patient outcomes. For instance, there is a growing literature that shows that timely antibiotic administration for patients presenting to an emergency department (ED) with fever and other signs of infection decreases their chances of developing sepsis and of many going into septic shock and dying.^{23,24} Timely administration of corticosteroid to pediatric patients presenting with acute asthma exacerbation decreases their length of stay²⁵ and reduces the chances of hospital admission.²⁶ Also timely administration of intravenous recombinant tissue plasminogen activator (rt-PA) in the ED has been shown to improve outcomes in patients who recently had a ischemic stroke.²⁷ In sum, decrease in time between the points when medication is ordered and when it's administered is an important outcome for adult and pediatric patients in the ED.

Methods

Design: Data extracted from EHR systems in two EDs, from two separate hospital systems located in Colorado were cleaned using Kahn et al.'s (2012)' five DQ concepts. Both of the data sets were extracted from relational databases developed by Epic (Verona, WI).

Sample: The first data set contained 70,061 patient events in adult ED between 2012 and 2013, and the second data set contained 2,815,550 patient events in a pediatric ED in 2013. The analysis conducted in this paper was part of a larger study, which has been ruled a Category 4 study (de-identification was used) and thus received an Exempt review by the Colorado Multiple Institutional Review Board (COMIRB).

We wrote all the code to examine the five DQ concepts using IBM SPSS version 22 Syntax. Our EHR data set preparation process (Figure 1) consisted of the following four phases:

The first phase was completed by our hospital partners, who provided a de-identified data set to use via secure file transfer in the form of a Microsoft Excel spreadsheet (.csv format). The second phase involved recoding variables in the Microsoft Excel (.csv) data set; while these steps were performed before phases 3 and 4, further discussion of the details regarding phase 2 is outside the scope of this paper.

Analytic Framework

Table 1 presents Kahn et al.'s (2012)¹ framework (phases 3 and 4), which was used as the analytic framework for the data cleaning in this case study. The first column includes the DQ concept; the second column includes the data set dimensions to be assessed in order to address the data concept; and the third column includes guidance on how to assess each of the concepts.

Application of the Kahn et al. (2012) Framework: Explanation of the Five Data Concepts

Attribute Domain Constraints

In our case study, *attribute domain constraints* refers to “variable coding structure and response options.” We suggest that the first important step in this phase of data cleaning is to identify whether or not the data match the predefined EHR data structure. In addition, it is important for the researcher to check whether the current format of the specific variables is appropriate for the intended analysis. In most cases, there may be data provided that match the predefined EHR structure, but the current format of the data may not be conducive to analysis. For example, variables coded as string can be recoded into value-labeled numeric categorical variables, or date and time information may be combined into a single variable when it is more appropriate for this information to be contained in two separate variables.

Figure 1. Data Set Preparation Phases

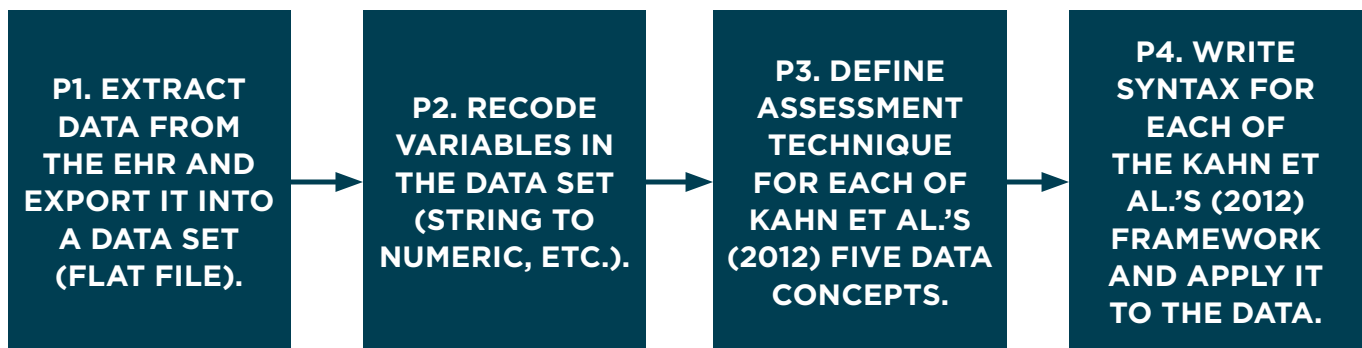




Table 1. Practical Guide to Examining EHR Data Sets Based on Kahn et al (2012)

KEY DATA CONCEPT	WHAT TO ASSESS	ASSESSMENT TECHNIQUE
Attribute Domain Constraints	Accuracy and response validity	Coding and recoding checks and frequency analysis (Figure 2). <i>Example: Do the responses match the predefined coding pattern? Are there variables currently coded as string variables that can be recoded into a value-labeled numeric variable?</i>
	Missing data	Missing data analysis. <i>Example: Are the missing values logical or is there a potential source of input or output error that should be considered?</i>
Relational Integrity Rules	Between database consistency	Compare patient IDs after a merge or compare the same patient ID on demographic variables (Figure 3). <i>Example: Prior to merging two data sets based on the primary key, are there missing values where you would expect them to be? Do the number of rows and variables in the merges data set add up to what were in the data sets that the merge comprised?</i>
	Between site consistency	Compare results of merging data sets (by comparing primary keys or patient IDs) between sites. <i>Example: Consider adding variables that code sites prior to merging so that errors can be easily traced back to the correct premerge file.</i>
Historical Data Rules	Time interval coding	Make sure that the time intervals are coded in the same units for all records and capture the desired time frame (Figure 4).
	Time stamps	Check that time stamps fall in expected intervals (weekly or monthly) and don't exceed a preestablished frequency.
State-Dependent Objects Rules	Event sequences per person and within a site	Make sure that the last event time occurs before the first event time (Figure 5). <i>Example: Verify multiple events to ensure that the primary event recording is accurate.</i>
	Sequence timing by event	Make sure that events have appropriate concurrent event times. <i>Example: Ensure that not only the event dates are correct, but for those events that occur on the same day, ensure that the recorded times make sense.</i>
Attribute Dependency Rules	Qualifying events	Check to make sure that events that depend on a previous event (treatment that follows a certain diagnosis) make sense. (Figure 6). <i>Example: An individual who gets admitted to psychiatric ED needs to have a psychiatric diagnosis; find both the diagnosis and psychiatric ED visit variable and compare their frequencies.</i>
	Dependent events	Find chief complaint variable and compare to the first ED event frequency <i>Example: Patients with discharge and departure events should also have arrival event information.</i>

Checking attribute domain constraints also refers to verifying the accuracy of missingness; specifically, answering questions about your data sets such as, “Do the current patterns of missing data make sense?” For example, patients who have complicated injuries may have multiple treatment

steps within each event, and thus will appear to have duplicate arrival date and time information. Having a clear understanding of the underlying structure of the database is crucial to verifying the accuracy of the extracted data set intended for secondary use.

Figure 2. Sample SPSS Code for Checking Attribute Domain Constraints

*Recode the Psych_enc variable from string to numeric and provide value labels.

RECODE Psych_enc ('Yes'=1) ('No'=0) **INTO** Psych_encr.

VARIABLE LABELS Psych_encr 'Psych_enc variable (Psych_enc) recoded from String to numeric'.

EXECUTE.

VALUE LABELS

Psych_encr

1 'Yes'

0 'No'.

EXECUTE.

*Recode the Final EDUC Disposition variable so that it is no longer a string variable and provide value labels.

RECODE Final_EDUC_Dispo ('Admit'=1) ('DC Home'=2) ('Expired'=3) ('LEFT AMA'=4) ('LWBS'=5)

('LWT'=6) ('Observation'=7) ('Transfer between sites'=8) ('Transfer to OSH'=9) **INTO** Final_EDUCr.

VARIABLE LABELS Final_EDUCr 'Final EDUC Dispo variable (Final_EDUC_Dispo) recoded from String to numeric'.

EXECUTE.

VALUE LABELS

Final_EDUCr

1 'Admit'

2 'DC Home'

3 'Expired'

4 'LEFT AMA'

5 'LWBS'

6 'LWT'

7 'Observation'

8 'Transfer between sites'

9 'Transfer to OSH'.

EXECUTE.

*Run frequencies on newly recoded variables to ensure that the recoding worked properly and that all possible values have been coded.

DATASET ACTIVATE DataSet1.

FREQUENCIES **VARIABLES=**Final_EDUCr Psych_encr

/ORDER=ANALYSIS.



Relational Integrity Rules

In our case study, *relational integrity rules* refer to “between-and-within-database consistency for data extracted from the same EHR.” Very often an analysis of data extracted from EHRs will consist of merging multiple data sets, and it is crucial not only to verify the consistency of each record within one data set but also to verify the consistency of records between data sets after each merge. The first step in checking for response integrity rules is identifying the primary key or the main identifying variable that should exist for each patient across all databases (i.e., patient ID).

Identifying a primary key is important in order to ensure that you can merge or join multiple files containing information on a group of patients. It also allows you to verify accuracy of the data. For example, if you have multiple observations for a primary key and need to verify which one is correct, you can utilize information from other data sets (i.e., known demographic information) for the particular

primary key to select and the correct observation.

Historical Data Rules

In our case study, *historical data rules* refer to “coding consistency and validity of variables indicating the time of an EHR event.” It is important to make sure that, if the data are supposed to be tracked in a monthly or yearly predefined format, there are no records capturing different time intervals (e.g., biweekly or semiannual format). It is also important to ensure that the time stamps for each of the events, for each patient, follow a logical predetermined pattern and that the gaps between events are not unrealistic. For instance, if a patient is supposed to have monthly visits, but the data indicate a different pattern of visits, there might be a problem with using that record for analysis. Unless there is a reason for gaps between events, such as between hospital patient transfers, the researcher needs to ensure that each record follows an expected timeline.

Figure 3. Sample SPSS code for Checking Relational Integrity Rules

```
*Merge datasets 1 and 2 together by Patient_idr(primary key).
DATASET ACTIVATE DataSet2.
MATCH FILES /FILE=*
/FILE='DataSet2'
/BY Patient_idr.
EXECUTE.

**Merging files together with a primary key that contains multiple events.
*Sort DataSet2 by Patient_idr(primary key) and Encounter_idr variables(event id).
DATASET ACTIVATE DataSet2.
SORT CASES BY Patient_idr(A) Encounter_idr(A).

*Sort DataSet1 dataset by Patient_idr.
DATASET ACTIVATE DataSet1.
SORT CASES BY Patient_idr(A).

*Merge datasets together by Patient_idr first and then by Encounter_idr second.
MATCH FILES FILE = DataSet2 / TABLE = DataSet1 / BY Patient_idr Encounter_idr.
```

Figure 4. Sample SPSS Code for Checking Historical Data Rules

```

*EVENTS DATASET EDITS.
*Change the event date and time formatted variable into separate
date only and time only formatted variables.

*Compute a new variable which will be used to split the date information from the time information for this event.
STRING E_DATEr (A11).
COMPUTE E_DATEr=substr(EVENT_TIME, 1, 9).
VARIABLE LABELS E_DATEr 'Split event time variable and remove date'.
EXECUTE.

*Create a new variable that will contain only the needed pieces of the event date.
STRING event_d (A13).
Compute event_d=concat(substr(E_DATEr,6,9),"/",substr(E_DATEr,3,3),"/",substr(E_DATEr,1,2)).
VARIABLE LABELS event_d 'Concatenate event time date'.
EXECUTE.

*Reformat the newly created date variable to be in the following format: MM/DD/YYYY.
COMPUTE EVENT_DATE=number(event_d, SDATE11).
VARIABLE LABELS EVENT_DATE 'Recoded and reformatted event date variable (EVENT_TIME)'.
VARIABLE LEVEL EVENT_DATE (SCALE).
FORMATS EVENT_DATE (ADATE10).
VARIABLE WIDTH EVENT_DATE(10).
EXECUTE.

*Create a new variable that will contain only the needed pieces of the event time.
STRING EVENT_Tr (A8).
COMPUTE EVENT_Tr=substr(EVENT_TIME, 11, 19).
VARIABLE LABELS EVENT_Tr 'Split event time variable and remove time'.
EXECUTE.

*Reformat the newly created time variable to be in the following format: HH:MM:SS.
COMPUTE EVENT_TIME2=number(EVENT_Tr, TIME8).
VARIABLE LABELS EVENT_TIME2 'Recoded event time variable (EVENT_TIME)'.
VARIABLE LEVEL EVENT_TIME2 (SCALE).
FORMATS EVENT_TIME2 (TIME8).
VARIABLE WIDTH EVENT_TIME2(8).
EXECUTE.

```



State Dependent Objects Rules

In our case study, *state dependent objects rules* refer to “ensuring that the ordering and the temporal ordering of patient events follow an appropriate sequence for each EHR record.” For example, it is important to verify that the patient date of birth occurs before any other events in the data set (i.e.,

that the patient is born before the recorded hospital arrival time). It is also important to verify that the sequence of steps within each event follow an expected, logical order. Events occurring out of the expected order or predetermined timeline may be indicative of a potential DQ error. It is important to verify not only the dates for each of the events but also the individual times for each event.

Figure 5. Sample SPSS Code for Checking State Dependent Objects Rules

```

*Verify the ordering of the ED departure date and First ED event date variables .
IF (ED_Dept_DATE > FirstED_Date) CDATE=1.
IF (ED_Dept_DATE < FirstED_Date) CDATE=0.
IF (ED_Dept_DATE = FirstED_Date) CDATE=2.
VARIABLE LABELS CDATE 'Verify the ordering of the ED departure date and First ED event date variables'.
EXECUTE.
VALUE LABELS
CDATE
1 'ED Departure date occurs AFTER FirstED event Date'
0 'ED Departure date occurs BEFORE FirstED event Date'
2 'ED Departure date and FirstED event Date are the same'.
EXECUTE.

*Verify the ordering of the ED departure time and First ED event time variables for the records that have the
same date for both of these variables .
IF (ED_Dept_DATE = FirstED_Date) AND (EDDept_TIME > FirstED_TIME) CTIME=1.
IF (ED_Dept_DATE = FirstED_Date) AND (EDDept_TIME < FirstED_TIME) CTIME=0.
IF (ED_Dept_DATE = FirstED_Date) AND (EDDept_TIME = FirstED_TIME) CTIME=2.
VARIABLE LABELS CTIME 'Verify the ordering of the ED departure time and First ED event time variables'.
EXECUTE.
VALUE LABELS
CTIME
1 'ED Departure time occurs AFTER First ED Event time'
0 'ED Departure time occurs BEFORE First ED Event time'
2 'ED Departure time and Date occur AT THE SAME TIME as ED Event time and date'.
EXECUTE.

*Composite of date and time verificaiton variables into a single variable.
IF (CTIME=1) C_Verify=1.
IF (CDATE=0) C_Verify=0.
IF (CTIME=0) C_Verify=0.
IF (CTIME=2) C_Verify=2.
VARIABLE LABELS C_Verify 'Verify that the ED Departure DATE & TIME occurs after the First ED event DATE & TIME'.
EXECUTE.
VALUE LABELS
C_Verify
1 'CORRECT: ED Departure Date/Time occurs AFTER First ED Event Date/Time'
0 'INCORRECT: ED Departure Date/Time occurs BEFORE First ED Event Date/Time'
2 'CORRECT: ED Departure Date/Time occur at the sametime as First ED Event Date/Time'.
EXECUTE.

```

Attribute Dependency Rules

In our case study, *attribute dependency rules* refers to “verifying the existence of dependencies between variables that are sequentially dependent upon each other.” For example, if patients have data for a variable that indicates what their chief complaint was, they should also have data for the variable that indicates that they had an ED event. Understanding the dependency between variables is an important part of verifying EHR data as this data is unidirectional; each event comprises a series of steps, and each step is dependent upon the steps that occurred before it. Variable dependency can also be used to help fill in missing data. For example, if there are multiple data sets and each data set represents a single step within an event that occurred (i.e., data set for chief complaint, data set for admission and discharge), then variables that appear in both data sets are the same and thus can be used to complete data that may be missing.

Post Quality Framework Data Checks and Treatment of Missing Data

In addition to applying the Kahn et al. (2012)¹ framework, more traditional DQ and validity checks—such as examining the randomness of missing data, examining outliers, examining if the data fits the assumptions of the chosen analysis, and checking variable distributions for skewness and kurtosis—should also be utilized. A frequently used framework for screening and cleaning data can be found in Chapter Four of *Using Multivariate Statistics*.¹⁹ Missingness is a difficult problem to address; specific techniques will depend on the purpose of the study as well as the properties of the missing data: *missing at random* (MAR), versus *missing not at random* (MNAR). SPSS²⁸ has a built-in multiple imputation menu, which uses chained equations to predict missing values. This method has been shown to be more effective for dealing with missing data than more traditional mean replacement or incomplete case deletion,²⁹ except when the amount of missing data is large (>50 percent). The code below shows an example code for doing multiple imputation in SPSS using five iterations (NIMPUTATIONS=5).

Figure 6. Sample SPSS Code for Checking Attribute Dependency Rules

```

**Fill in missing time data for AdmitDecisionDate from DataSet1 with corresponding data from the same
variable in DataSet2.
Dataset1.
If (AdmitDecisionDate_bk = '.') AdmitDecisionDate_bk = AdmitDecisionDate_im.
If (AdmitDecisionTime_bk = '.') AdmitDecisionTime_bk = AdmitDecisionTime_im.
If (CDUDecisionDate_bk = '.') CDUDecisionDate_bk = CDUDecisionDate_im.
If (CDUDecisionTime_bk = '.') CDUDecisionTime_bk = CDUDecisionTime_im.

```

Figure 7. Sample SPSS Code for Executing Multiple Imputation

```

DATASET DECLARE SPSSImputations.
MULTIPLE IMPUTATION Variable1 Variable 2 Variable3
/IMPUTE METHOD=AUTO NIMPUTATIONS=5 MAXPCTMISSING
/MISSINGSUMMARIES NONE
/IMPUTATIONSUMMARIES MODELS DESCRIPTIVES
/OUTFILE IMPUTATIONS=SPSSImputations .

```



Findings

Applying Quality Framework Data Checks: An Example

Table 2 illustrates the importance of properly preparing an extracted data set prior to conducting analysis—specifically, the number of patients and encounters, number of records needing recoding, number of variables with missing observations, number of variables that have erroneous time sequences, and the number of variables with dependent events by data set. Table 2 provides valuable descriptive information about the potential impact of each of the various checks by data set.

As described above, state-dependent object rules are necessary when working with multiple events. The primary purpose of these rules is to ensure that the times within a given event occur in the expected order. To better illustrate the potential issues that can arise when time sequences are not properly

examined prior to analysis, a one-way Kruskal-Wallis (K-W) ANOVA with Bonferroni adjustment for pairwise comparisons was run. Specifically, the K-W ANOVA was used to determine whether there were significant differences between five ED locations—main ED and five satellite clinics (SC)—and the time between when an order for medication was given and when the medication was taken.

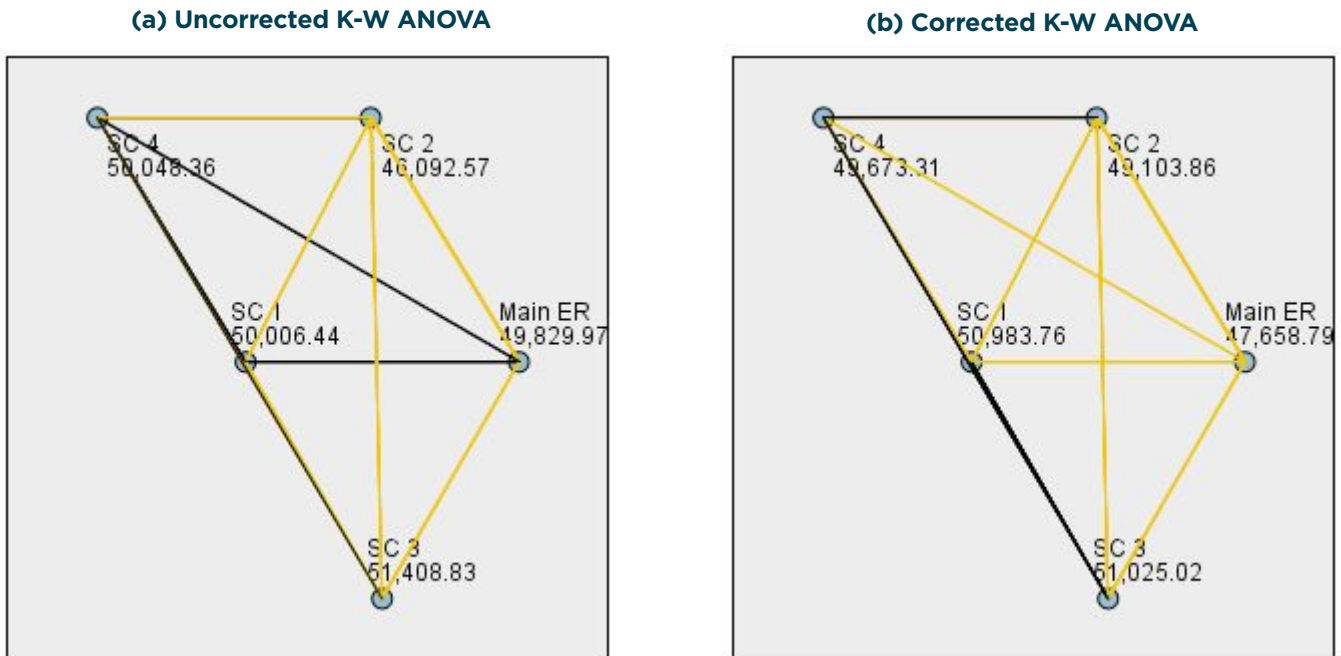
Corrected and Uncorrected Differences

K-W ANOVAs analyses were run on the uncorrected (Figure 8a) and corrected (Figure 8b) data sets to illustrate the importance of utilizing properly prepared data. While the results from the K W ANOVAs indicated the EDs were significantly different both pre- and postcorrection, the pre- and postcorrection pairwise comparison results were very different (Figures 8a, 8b). Specifically, prior to correction, the pairwise comparisons indicated differences between the EDs that—when time sequence errors were corrected—no

Table 2. Initial Descriptive Screening by Data Set

DESCRIPTIVE SCREENING	DATA SET 1	DATA SET 2
Number of patients	241,773	70,061
Number of encounters	2,815,550	70,061
Records needing recoding (check for Attribute Domain Constraints-related issues)	25	13
Number of potential primary keys (check for Relational Integrity Rules)	2	1
Variables with missing observations (check for Attribute Domain Constraints-related issues)	32	4
Variables with time sequences (check for Historical Data Rules, Relational Integrity Rules, State Dependent Object Rules)	8	6
Variables with dependent events (Check for Attribute Dependency Rules and Relational Integrity Rules)	7	5

Figure 8. Pairwise Comparison Results for the Kruskal-Wallis ANOVA on Uncorrected (a) and Corrected (b)



Note: Yellow lines represent significant relationships ($p < 0.05$); black lines represent nonsignificant relationships ($p > 0.05$).

longer existed. This result is especially important because the large sample sizes in these types of extracted data allow for even small differences to be statistically significant. Failing to properly correct for time sequence errors can result in very different interpretations of results and thus could have very different clinical implications.

Uncorrected Time Sequences

As shown in Figure 8a, results indicated that the average length of time between when a medication order was given and when it was taken significantly differed by ED ($F(4, 99434) = 111.52, p < .0001$). Pairwise comparisons indicated that the main ED medication time significantly differed from SC 2 and SC 3 medication times ($p < .0001$); SC

1 medication time was significantly different than SC 2 ($p = .01$) and marginally different than SC 3 ($p = 0.05$) medication times; SC 2 medication time was significantly different than the main ED, SC 1, SC 3, and SC 4 ($p < .0001$) medication times; SC 3 medication time was significantly different than SC 2 ($p < .0001$) and marginally different than the main ED ($p = .004$) and SC 1 ($p = 0.05$) medication times; SC 4 medication time was significantly different than SC 2 ($p < .0001$) medication time.

Corrected Time Sequences

Checking the accuracy of the order in which these two events occurred revealed 2,519 encounters with incorrect time sequences. As shown in Figure 8b, results indicated that the average length of time



between when a medication order was given and when it was taken significantly differed by ED ($F(4, 96915) = 215.90, p < .0001$). Pairwise comparisons indicated that the main ED medication time significantly differed from SC 1, SC 3, and SC 4 ($p < .0001$) and marginally differed from SC 2 ($p = 0.003$) medication times; SC 1 medication time was marginally different than SC 4 ($p = .05$) medication time; SC 2 medication time was marginally different than SC 3 ($p = .009$) medication time.

Corrected and Uncorrected Differences

The two separate analyses were run to illustrate the importance of utilizing properly prepared data. While the results from the Kruskal Wallance one-way ANOVA indicated the EDs were significantly different both pre- and postcorrection, the pre- and postcorrection pairwise comparison results were very different. Specifically, prior to corrections, the pairwise comparisons indicated differences between the EDs that—when time sequence errors were corrected—no longer existed. Failing to properly correct for time sequence errors can result in very different interpretations of results and thus could have very different clinical implications.

Discussion and Lessons Learned

In this paper we adapted a pragmatic framework for assessing EHR DQ by Kahn et al. (2012),¹ originally designed for relational databases for use with cleaning extracted data sets for secondary use. The data used in the current case study was extracted from two different (adult and pediatric) EDs from two different hospital systems in the same geographical region. The Kahn et al. (2012)¹ framework allowed us to examine the data extracted from a relational database within a framework addressing specific DQ concepts such as complex, dependent time series data; combining multiple database tables that traditional data cleaning and screening guides, such as those found in as

Tabachnick & Fidell (2001)¹⁹, are not designed to address.

SPSS syntax was used to write code to examine the Kahn et al. (2012)¹ DQ concepts, and example SPSS syntax commands were provided to help potential researchers using data extracted from EHRs, especially ED EHRs, to examine the quality of their data. These examples are not exhaustive and should not be thought of as a finished product but as a starting point that can be customized for each researcher's specific analytical needs.

Although we attempted to provide generalizable examples, our sample commands and indicators of quality might not capture the full spectrum of EHR databases, especially the unstructured databases. We would like to emphasize that this case study is meant to be an example of how researchers and statisticians might adapt a DQ frameworks for data cleaning purposes in order to combine “data preparation” activities such as data validation and data cleaning, thus the number of errors found by other researchers might be greatly different.

For instance, the hospital system that provided us with this data has stronger than average data validation protocols, thus we found a smaller number of errors than we expected. We think that an important finding of our case study is that even a small amount of errors in the data set can have an impact on statistical analysis. Other data sets extracted from similar EHR system might have many more errors.

We think that our overarching goal of making sure that the patient data and event sequences within the data were what would be expected in the “real world” was unique to this case study. This is important because it assures us that we have removed important bias (improbable or erroneous values and event sequences) from our statistical analysis.

In the future, we will translate this syntax to other programming languages such as R and SAS, and will build an R package that will automate the DQ check processes as well as include more diverse DQ indicators. We will also include functions to visualize data and algorithms to deal with missing data specifically designed for time-series analysis, an important set of methods for analyzing outcomes of pragmatic trials. The primary purpose of this R package will be to provide a set of tools that allows for easy DQ checking based on concepts adopted from Kahn's et al. (2012)¹ DQ framework and appropriate data cleaning on data extracted from the EHR.

Conclusion

The Kahn et al. (2012)¹ framework, which incorporates five DQ concepts, designed for examining relational databases, was useful in examining EHR-extracted data intended for secondary use or for particular statistical analyses. We were able to use SPSS to examine each of the five data concepts and have provided detailed, step-by-step code via syntax for each customized to each of these categories. Future research should aim to replicate work done in this case study as well as provide additional examples of how to apply this framework to other statistical software packages.

Acknowledgments

We would like to thank our hospital partners for making these two data sets available to us. We would also like to thank the Office of Research at the College of Nursing for support of this work.

References

1. Kahn MG, Raebel MA, Glanz JM, Riedlinger K, Steiner JF. A pragmatic framework for single-site and multisite data quality assessment in electronic health record-based clinical research. *Medical care*. 2012;50.
2. Grossman C, McGinnis JM. *Digital Infrastructure for the Learning Health System: The Foundation for Continuous Improvement in Health and Health Care: Workshop Series Summary*. National Academies Press; 2011.
3. Marsolo K. In Search of a Data-in-Once, Electronic Health Record-Linked, Multicenter Registry—How Far We Have Come and How Far We Still Have to Go. *EGEMS*. 2013;1(1).
4. Ozkaynak M, Brennan PF, Hanauer DA, et al. Patient-centered care requires a patient-oriented workflow model. *J Am Med Inform Assoc*. 2013;20(e1):e14-16.
5. Welton JM. Business Intelligence and Nursing Administration. *Journal of Nursing Administration*. 2014;44(5):245-246.
6. Talbert S, Sole ML. Too much information: research issues associated with large databases. *Clin Nurse Spec*. 2013;27(2):73-80.
7. Greiver M, Barnsley J, Glazier RH, Harvey BJ, Moineddin R. Measuring data reliability for preventive services in electronic medical records. *BMC health services research*. 2012;12(1):116.
8. Dixon BE, Siegel JA, Oemig TV, Grannis SJ. Electronic health information quality challenges and interventions to improve public health surveillance data and practice. *Public health reports (Washington, DC: 1974)*. 2012;128(6):546-553.
9. Price M, Bowen M, Lau F, Kitson N, Bardal S. Assessing accuracy of an electronic provincial medication repository. *BMC medical informatics and decision making*. 2012;12(1):42.
10. McGinn CA, Grenier S, Duplantie J, et al. Comparison of user groups' perspectives of barriers and facilitators to implementing electronic health records: a systematic review. *BMC Med*. 2011;9:46.
11. Parsons A, McCullough C, Wang J, Shih S. Validity of electronic health record-derived quality measurement for performance monitoring. *J Am Med Inform Assoc*. 2012;19(4):604-609.
12. Johnson KE, Kamineni A, Fuller S, Olmstead D, Wernli KJ. How the Provenance of Electronic Health Record Data Matters for Research: A Case Example Using System Mapping. *EGEMS*. 2014;2(1).
13. Hersh WR, Weiner MG, Embi PJ, et al. Caveats for the use of operational electronic health record data in comparative effectiveness research. *Medical care*. 2013;51:S30-S37.
14. Capurro D, van Eaton E, Black R, Tarczy-Hornoch P. Availability of Structured and Unstructured Clinical Data for Comparative Effectiveness Research and Quality Improvement: A Multisite Assessment. *EGEMS*. 2014;2(1).
15. Bayley KB, Belnap T, Savitz L, Masica AL, Shah N, Fleming NS. Challenges in using electronic health record data for CER: experience of 4 learning organizations and solutions applied. *Med Care*. 2013;51(8 Suppl 3):S80-86.
16. Redman TC. *Data quality: the field guide*. Digital Press; 2001.
17. Weiskopf NG, Weng C. Methods and dimensions of electronic health record data quality assessment: enabling reuse for clinical research. *J Am Med Inform Assoc*. 2013;20(1):144-151.
18. Savik K, Fan Q, Bliss D, Harms S. Preparing a large data set for analysis: using the minimum data set to study perineal dermatitis. *J Adv Nurs*. 2005;52(4):399-409.
19. Tabachnick B, Fidell L. Cleaning up your act: screening data prior to analysis. *Using multivariate statistics*. 2001;5:61-116.
20. Van den Broeck J, Cunningham SA, Eeckels R, Herbst K. Data cleaning: detecting, diagnosing, and editing data abnormalities. *PLoS Med*. 2005;2(10):e267.



21. Devine EB, Capurro D, van Eaton E, et al. Preparing Electronic Clinical Data for Quality Improvement and Comparative Effectiveness Research: The SCOAP CERTAIN Automation and Validation Project. *EGEMS*. 2013;1(1).
22. Kahn MG, Brown JS, Chun AT, et al. Transparent Reporting of Data Quality in Distributed Data Networks. *eGEMs*. 2015;3(1).
23. Monroe K, Kutny M. Protocol for Reducing Time to Antibiotics in Pediatric Patients Presenting to an Emergency Department With Fever and Neutropenia. 2015.
24. Kumar A, Roberts D, Wood KE, et al. Duration of hypotension before initiation of effective antimicrobial therapy is the critical determinant of survival in human septic shock*. *Critical care medicine*. 2006;34(6):1589-1596.
25. Davis SR, Burke G, Hogan E, Smith SR. Corticosteroid timing and length of stay for children with asthma in the emergency department. *Journal of Asthma*. 2012;49(8):862-867.
26. Bhogal SK, McGillivray D, Bourbeau J, Benedetti A, Bartlett S, Ducharme FM. Early administration of systemic corticosteroids reduces hospital admission rates for children with moderate and severe asthma exacerbation. *Annals of emergency medicine*. 2012;60(1):84-91. e83.
27. Lees KR, Bluhmki E, Von Kummer R, et al. Time to treatment with intravenous alteplase and outcome in stroke: an updated pooled analysis of ECASS, ATLANTIS, NINDS, and EPITHET trials. *The Lancet*. 2010;375(9727):1695-1703.
28. Nie NH, Bent DH, Hull CH. *SPSS: Statistical package for the social sciences*. Vol 227: McGraw-Hill New York; 1975.
29. Van Buuren S. Multiple imputation of multilevel data. *Handbook of advanced multilevel analysis*. 2011:173-196.