A User Study on Data-Driven Dosage of Heparin

Joe Rowley

A thesis presented for the degree of

Bachelor of Science

Advised by:

Sri Kurniawan

Mohammad M. Ghassemi

University California, Santa Cruz

December 2015

Maybe include keywords section here:

Retrospective data driven dosing, HCI, Web Application Development

Abstract

The intention of this study was to determine the feasibility of using statistical drug dosing tools in a mock clinical setting. We wanted to learn about both the technical implementation and user experience challenges involved with creating and using computer aided dosing tools. First a web application was developed using existing models for dosing heparin in ICU patients. This application was next developed into an interactive user survey. We found that Doctors were generally more confidant and faster when dosing using the tools. [[add other results here too. It's good to avoid numbers in abstracts from what I recall]] In closing we recommend some best practices for future development of statistical drug dosing tools.

Acknowledgements

The models used in this tool were introduced Mohammad M. Ghassemi, Stefan E. Richter, Ifeoma M. Eche, Tszyi W. Chen, John Danziger, and Leo A. Celi in their paper *A data-driven approach to optimized medication dosing: a focus on heparin*. This project would not have been possible without Mohammad's support.

Additionally, data used in these models was sourced from the MIMIC-II and MIMIC-III (Saeed et al. (2011)) medical databases, a project supported by the MIT Lab for Computational Physiology.

I also have many advisors who helped me with this project:

- Mohammad M. Ghassemi
- Sri Kurniawan
- Matthew Guthaus

Additionally the following people were very influential in preparing me for this project:

- James Davis
- Phokion Kolaitis
- Philip Strong

Finally, this project would not have been possible without the unwavering moral and technical support of my parents Howard Rowley and Carol Diamond.

Table of Contents

1 Introduction

The primary goal of this study was to determine the technical implementation challenges and user experience challenges involved with a statistical dosing tool. Specifically, we implemented a heparin dosing tool that was first introduced in the paper *A data-driven approach to optimized medication dosing: a focus on heparin* (Ghassemi et al. 2014). In the paper a statistical model is generated for predicting an optimal infusion rate for heparin for patients in intensive care units (ICUs). In this study, we reimplemented their existing models into an interactive web application, which allows the user to generate dosage guidelines based on a patient's features. Next a user study application was created, which presents the doctors with a patient case. The doctor initially provides their own suggestion for the infusion rate and is then presented with the model's prediction for an optimal infusion rate. After the doctor views the model, they have the opportunity to adjust their initial dose and provide a reasoning for their dose adjustments. Their interaction with the tool yields many metrics, including number of dose adjustments, speed of dosing, number of times guidelines were viewed.

2 Background

2.1 Traditional Dosing of Heparin

Heparin is an anticoagulant medication administered intravenously to patients with blood clots. Specifically it is often used for strokes, pulmonary embolism (blockage in an artery in the lung), acute coronary syndrome

(e.g. heart attack), and many other conditions. The general protocol for dosing heparin is to initially order baseline lab results for the patient. These values combined with context of the patients condition allow a doctor to determine the infusion rate and if a bolus (a single large initial dose) is required. The primary metric used for monitoring patient progress is activated partial thromboplastin time (APTT). After the infusion is started the patient is monitored and the dose is adjust periodically. Most often new labs are ordered every 6 hours [citation to UW and Beth Israel].

2.2 Predicting Therapeutic Outcome

The objective of the dosing tool is determine an ideal initial infusion rate, such that patient has the highest probability of reaching a therapeutic state in 6 hours time. A therapeutic state is defined as an APTT value of between 60 seconds and 100 second. To predict an optimal dose Ghassemi et al. employed the use of multi-variant logistic regression.

2.2.1 Logistic Regression Approach

The objective of the model was to determine the probability that patient would end in one of three categories: sub-therapeutic, therapeutic, or supra-therapeutic. A patient can by definition only be in one category at a time. This lead Ghassemi et al. to the following equation:

(1) $1 = P(sub{-}therapeutic) + P(therapeutic) + P(supra{-}therapeutic)$

The challenge of using logistic regression to predict therapeutic outcomes is that the probability of a therapeutic outcome is not a strictly monotonic

function. So using logistic regression directly to predict P(therapeutic) will not work. That being said, the probability of sub-therapeutic and supra-therapeutic outcomes are monotonically decreasing and increasing functions respectively. Given these properties, we can use multinomial logistic regression to determine functions for sub-therapeutic and supra-therapeutic aPTT. Using equation 1 we can then determine the probability of a therapeutic outcome.

Although at this point a case been has made that multinomial logistic regression may be an appropriate tool, there are a few considerations the Ghassemi et al. took with regards to the form of regression. There are many forms of multinomial logistic regression with subtle but important differences. There is a natural order to the classifications (sub-therapeutic, therapeutic, supra-therapeutic), so ordinal logistic regression could be considered. The challenge with ordinal logistic regression, however, is that it makes the assumption of proportional odds, so regression intercept terms vary across classes while regression coefficients are shared across classes. In the words of the original authors *"it assumes that explanatory features maintain identical effects across varying ranges of the outcome"*[citation appendix]. Instead the ordinal regression they instead used a modified version of standard multinomial logistic regression, which utilized a flexible reference category. That is instead of using a single class to predict the outcome that is not being modeled, they used all classes. So when modeling supra-therapeutic aPTT, standard multinomial logistic regression would reference either therapeutic or sub-therapeutic aPTT classes, while their approach referenced both classes. This choice was demonstrated to have significant effects on the validity of their models [citation to appendix, section 2].

2.2.2 Feature Selection

There are innumerable potential features one could consider as predictors of patient outcome. In their paper, Ghassemi et al. determined the salient features as follows: age, SOFA score, Elixhauser, Heparin dose, Measurement time, Creatinine, Ethnicity, Gender, ICU type, presence of pulmonary embolism, obesity, presence of end stage renal failure.

2.3 Summary of Prior Work

The original paper and associated appendix document the efforts taken to validate the models. To summarize, they partitioned their dataset, dedicating 70% to training and 30% for validation. They then compared the calculated the predicted classification for each patient using both a full featured model and a weight-only model. The full featured model was more accurate than the weight-only approach as measured by Volume Under the Receiver Operating Characteristic Surface (the multiple class version of Area Under Receiver Operating Characteristic Curve), with values of 0.48 vs 0.42 [citation].

In closing, they noted that a randomized controlled trial would be necessary to determine if this mechanism is more effective for dosing. This thesis project represents the initial steps towards this trial.

3 Heparin Dosage Calculator

This project is composed of two key components. The first is a "dose calculator" which contains all the functionality required to determine the optimal dose for a given patient. The second component is a survey that incorporates the calculator and is used to study how doctors interact with the dosing tool. We now describe the calculator.

3.1 Functional Requirements

The purpose of the calculator is to determine the probability functions for sub-therapeutic, therapeutic, supra-therapeutic aPTT for a range of infusion amounts for a given patient. The calculator can be broken into three steps:

1. Access and refine Dataset
2. Calculate static models
3. Make a prediction based on a given patient's features.

3.1.1 Data Processing

The data source for the original study was the MIMIC II database [citation]. MIMIC is an open access database consisting of over 40,000 deidentified patient encounters from Beth Israel Deaconess Hospital's ICUs. It is hosted at MIT's Lab for Computational Physiology. The majority of patients in an ICU do not receive heparin, so filtering by medication was needed. This was accomplished by a series of SQL queries on a local machine. Furthermore, we worked to maintain an identical dataset as was used in the original

dataset. This required filtering out all transfer patients and all patients missing required features. Additionally some features viz. Elixhauser comorbidity index, mean Sequential Organ Failure Assessment (SOFA), required manual calculation on a per patient basis, as they were not available directly from MIMIC. This yielded a dataset of approximately 1,600 patients. This number varies depending on the features supplied for a given prediction.

This data munging was primarily done in Python using the Pandas library, but also required PostgreSQL for hosting the database. SQL queries were composed by hand. To prepare the data for model generation we needed to code several variables. Specifically, we needed to add binary classifiers for sub-therapeutic and supra-therapeutic to each patients dataset. Additionally a binary classifier was needed for ethnicity, and we followed the original coding used in the paper (white and non-white).

3.1.2 Model Generation

Once the dataset was prepared, generating the models was fairly simple. We used the Python library Scikit-learn's implementation of multinomial Logistic Regression to compute the two logistic regressions. Once the two logistic regressions for sub-therapeutic and supra-therapeutic aPTT are generated they can be stored for later use as long as the dataset is static. We avoided doing this to simulate a real world scenario where data is constantly being added overtime.

3.1.3 Model Prediction

To generate the final dose curves for a specific patient we use scikit-learn's logistic regression predict_proba function. A function was created to output the probability of sub-therapeutic, sup-therapeutic aPTT given a patients features and a dose. For graphical purposes we call this function repeatedly for varying infusion rates in the range of 0 to 34 units/Kg/HR in steps of 0.5 units/Kg/HR. Predicting the specific infusion rate which maximizes the probability of a therapeutic outcome can also be thought of as finding the point which minimizes the sum of the probabilities of sub-therapeutic and supra-therapeutic (see equation 1). Using this fact we used the python library scipy's minimize_scalar function which implements Brent's Method and let it solve for the highest probability with a tolerance of .01%. The infusion rate curves and the optimal dose are then cached in Postgres for future queries that match the same patient features, given that the initial dataset has not changed.

3.2 Architecture

3.2.1 Frontend

The frontend for the application was built in Angular.js. Angular.js is client-side MVC (Model View Controller) framework that was chosen for it's strengths in quick prototyping. The user interface was built in Twitter Bootstrap, as it allows for rapid prototyping and easy implementation responsive design. Additionally, it has many css components such as progress bars that are useful for the design.

3.2.2 Backend

The backend for the application utilizes Python's Flask webframe work. Data is stored in postgreSQL and accessed through the sqlalchemy ORM library. The frontend interfaces with the server via a stateless REST api which enables the exchange of json data. The API definition is available in the appendix.

4 Heparin Survey

4.1 Survey Design

The primary objective of the survey was to learn about the user experience challenges involved with data-driven dosage techniques. To do this a test interface was developed and several medical doctors of varying disciplines attempted to use it. The survey consisted of 5 patient cases developed by domain experts to cover multiple different use cases for heparin. A doctor would first agree to a research disclosure, then fill out a short questionnaire about their background and relevant experience. Next an instruction page was displayed explaining the survey. Once the doctor read over the instructions and guidelines, they were presented with their first case which consists of some lab results and a short admission note. It was their task to determine a potential infusion rate based on traditional heparin dosing guidelines that were available to them. After choosing an infusion rate, the optimal dose model was displayed, along with the location of the users dose. The doctor then had the option of either adjusting the dose or sticking with their

initial dose. Whenever dosing the doctor was able to leave notes to explain their reasoning. This task then was repeated for the remaining four patients. After completing the test, there was a short follow-up questionnaire consisting of a few questions to gauge receptiveness to the tool and measure any change in confidence in the tool. Additionally, we asked for other places they thought the tool could be used.

To do limitations of funding we could not compensate doctors for their time taking the test. As a result all volunteers were acquaintances or friend's of friends of the author.

4.2 Architecture and Implementation

The survey was implemented using the same technological stack as the calculator (Angular.js and Bootstrap on the frontend). By utilizing a SPA (single page application) methodology we were able to quickly iterate and make many changes over time. There were several candidate versions of the survey developed over time, and tested with domain experts for feedback. The final iteration allows for the most direct input of comments from doctors, instead of relying on proxy metrics such as time to complete cases that are similar with and without the heparin calculator.

5 Survey Results

5.1 Cumulative Results

5.2 Dose differences

5.3 Time differences

5.4 Understanding/Confidence in Results

5.5 Other results

Something about other results… blah blah blah.

Suggested IR (Units/Kg/HR)	**Patient 1** 19.7		**Patient 2** 13.3		**Patient 3** 15.9		**Patient 4** 14.4		**Patient 5** 17.5	
Participant	**IR 1**	**IR 2**	**IR 1**	**IR 2**	**IR 1**	**IR 2**	**IR 1**	**IR 2**	**IR 1**	**IR 2**
1	18	25	18	19	18	20	18	26	20	24
2	18	19.8	18	13.9	13	16	12	14.7	13	17.6
3	18	20.3	80	14.1	18	16.4	18	15	18	18
4	12	12	15	15	12	15.9	15	15	12	17.3
5	18	19.6	18	13.7	12	15.9	18	14.5	12	17.4
6	18	18	18	18	13	13	12	14.8	0	0
7	18	19.6	18	13.6	17	16	15	15	18	18
8	18	19.7	18	13.8	13	16	12	14.7	18	18
Min Dose	12	12	15	13.6	12	13	12	14.5	0	0
Max Dose	18	25	80	19	18	20	18	26	20	24
Standard Deviation	2	4	22	2	3	2	3	4	6	7
Mean Dose	17	19	25	15	15	16	15	16	14	16
Median Dose	18	20	18	14	13	16	15	15	16	18

Table 1: My caption

	Patient 1		Patient 2		Patient 3		Patient 4		Patient 5		
Participant	**T1**	**T2**	**T1**	**T2**	**T1**	**T2**	**T1**	**T2**	**T1**	**T2**	**Total Time**
1	67	38	21	29	30	37	38	27	23	44	5 mins, 54 secs
2	136	42	49	45	53	33	98	26	112	19	10 mins, 13 secs
3	38	38	32	34	52	34	68	238	45	28	10 mins, 7 secs
4	23	69	50	53	237	79	96	32	86	72	13 mins, 17 secs
5	11	8	87	15	24	25	9	18	54	7	4 mins, 18 secs
6	55	60	25	84	38	21	85	46	51	22	8 mins, 7 secs
7	70	86	45	58	442	51	95	36	94	36	16 mins, 53 secs
8	135	83	54	40	29	19	29	18	44	50	8 mins, 21 secs
Min Time	11	8	21	15	24	19	9	18	23	7	
Max Time	136	86	87	84	442	79	98	238	112	72	
Standard Deviation	47	26	21	21	150	20	35	74	30	20	
Mean Time	67	53	45	45	113	37	65	55	64	35	9 mins, 39 secs
Median Time	61	51	47	42.5	45	33.5	76.5	29.5	52.5	32	7 mins, 51 secs

Table 2: Timing Results

6 Retrospective / Next Steps

6.1 Shortcomings in the Survey

- errors

- lack of data
- better explanation of how the model/logistic regression works. explain what features are being used in the prediction and what aren't. suggest a potential user interface that gives weights to things e.g. The patients suggested dose deviates from a baseline dose curve due to the patients ethnicity and creatinine and mean SOFA scores.

6.2 Places for Future Research

how did the doses differ in part 1 and part 2 of the survey.

6.3 Implementation Differences in future version

- transfer model to device so it can do queries more quickly
- instead of static data source have real medical database
- easily interchangeable models
- better analytics

7 Conclusion

Summary of work that was done, then summary of results. Then summary of places for future work.

Appendix 1: Some extra stuff

Participant	Patient 1	Patient 2	Patient 3	Patient 4	Patient 5
1	80	80	60	80	60
2	80	60	0	40	0
3	5000	5000	80	80	80
4	5000	6000	3500	6000	5400
5	80	80	0	80	0
6	80	80	0	44.5	0
7	80	80	0	60	80
8	80	80	0	60	80

Table 3: Bolus Information

- api definition
- link to test software
- link to source code git repos
- screenshots and links to alternative design for testing aPTT over time
- links to data sources/notebook of documentation
- extended results from survey.

  .. tables with stats like ave, std deviation etc for each patient 1-10 ..-

Appendix 2: Some extra stuff

This could be a list of papers by the author for example Also tutorials/people/stack overflow that was helpful. tom's markdown -> latex

References

Ghassemi, M.M. et al., 2014. A data-driven approach to optimized medication dosing: A focus on heparin. *Intensive care medicine*, 40(9), pp.1332–1339.

Saeed, M. et al., 2011. Multiparameter intelligent monitoring in intensive care iI (mIMIC-iI): A public-access intensive care unit database. *Critical Care Medicine*, 39, pp.952–960.