NarxScore and ORS Walkthrough Script

Slide 16:

Alright! I am now going take the reins and shift the focus of the rest of this presentation to highlighting a couple of scores in the application: the narcotic score contained in the suite of NarxScores and the ORS score. But before getting started, I just wanna say feel free to jump in and ask questions along the way. I prefer an informal and conversational presentation style. Just makes me feel more comfortable in my own skin.

Okay, so peer underneath the hood of both – I will walk through the logic of how we created the Narcotic Score and show a manual calculation, step by step. Plus, I will also walk through all of the PDMP features that comprise the current ORS model, which allows us to easily explain and communicate the robustness of the score because the underlying features cover a wider range of scenarios compared to the original model as well as consider more factors such as chronic pain.

In other words, the updated feature set addresses the Explainable pillar of the Fair AI framework (which is kind of a big deal nowadays).

Slide 17:

All Narx Scores are calculated using peer-reviewed, literature based PDMP features. Since we are focusing on the Narcotic Score component for this demo, I have listed the salient PDMP features here on the slide. Each feature is aggregated across four windows of time, shown in the upper right corner of the slide. Super high-level TL;DR here: a Narx Score is calculated as a weighted average of scaled values, which we will get into in a moment. A 50% weighting is applied to the milligram equivalencies with the remaining factors making up the other 50%. This type of weighting results in several relationships. In other words, if we think of milligram equivalency as consumption and the combination of providers, pharmacies, and overlaps collectively as behaviors, we can create the score categories listed in the table.

It is important to understand that there are several different patterns of use that can result in the same score. So, it is always necessary to look at the actual PDMP data to determine what use patterns exist that have resulted in the Narx Score presented.

Slide 18:

Just to keep things super simple, let's focus on some simulated data for a single patient. Here we see some raw, aggregated counts for each of the PDMP features.

Slide 19:

Next, we convert these values to scaled (i.e., relative) values using percentile ranking. Epidemiologists love this stuff, so it is near and dear to my heart. In other words, we find it compelling to examine how each aggregated value compares in terms of percentile-rank across each window of time to the rest of the patient's PDMP population. So, normalization is the aim of this step.

Slide 20:

Next, we average the percentile scores across time. Focusing on the top row of the table, we can see that 85 + 76 + 84 + 64 / 4 = 77. We then repeat the same process for the remaining features.

Slide 21:

Then, we calculate the weighted average of all of the scores (shown in green). Quick note. Dividing by the sum of the weights is necessary to normalize the weighted average. Here, the sum of the weights is 8. Casting our memory back a few slides, please recall that the weights used to calculate the average reflect the importance or significance of each feature, but they may not necessarily add up to 1.0 or 100%. By dividing the sum of the weighted features by the sum of the weights, we are essentially calculating the weighted average as a percentage of the total weight. This normalization ensures that the resulting average is on a scale of 0 to 1, or 0% to 100%, which makes it easier to compare the average across different datasets or to interpret it in a meaningful way. Without normalization, the weighted average could be misleading, especially if the weights are not proportional to each other or if the sum of weights is significantly different from the expected value of 1.0 or 100%.

Slide 22:

Finally, we concatenate the number of active prescriptions onto the end of weighted average as a third digit to arrive at the final three-digit Narcotic Score. To determine the number of active prescriptions, we add the day's supply to the filled at date. If the current date falls within those two bookends, then we consider the prescription to be 'active.'

Slide 23:

Here we can see how this process is represented as a simple equation, where each feature is given a variable and the letter 't' represents each window of time.

Slide 24:

Next up is the ORS. And, referring back to the pillars of FairAI, we can see some of these explainable features here. Don't worry about trying to get a handle on all of this right now. We will take a look at them in more detail in just a few slides. These features (including MOUD history) had the most explanatory power in correlating unintentional overdose death in our training data. Quick note: MOUDs are formerly known as MATs.

Slide 25:

Just to be super transparent on which features are new to the current model that were not included in the original model, (next slide please) we have incorporated patient age, gender, the number of days a patient had an opioid and sedative in their possession at the same time, a flag that captures chronic opioid use (which, in the absence of diagnosis codes, we use as a proxy for identifying patients with chronic pain) and a flag that captures Medications used for Opioid Use Disorder (MOUD).

Slide 26:

As we make our way through the next 5 slides, keep in mind that these features are listed in descending order of importance in terms of how much weight they confer to a patient's score calculation. So, a history of MOUD causes the largest swing in scores. This is what we've observed in the training data. Moreover, we can justify this empirical evidence with what has been published in the peer reviewed literature. This is true for each feature, and I have included citations to each peer-reviewed study in the presentation.

...sensitivity analysis for inclusion of MOUD hx flag

Slide 32

Just want to make a quick note here. The primary focus of our patient simulations today is on the score below the dial, though I have tried to simulate Narx Scores for Narcotics appropriately. I left the score for sedatives as 0 to keep the simulations as simple as possible and I left the score for stimulants as 0 because neither ORSv1 nor ORSv2 takes stimulant use into account.

What we're going to do for the next 2 slides is establish two 'base' patients, one male and one female. I've chosen to use the age of 45 years in all simulations for two reasons. 1): so that we can hold age constant and 2: this age group, along with folks aged 25 to 44 have somewhat comparable rates of death due to unintentional overdose (see citation from slide 5). And then, for the remaining 6 slides, we are going to examine the impact of some of the top features on the scores for our base patients.

Just a quick note on the effect of self-reported gender. The literature reports significantly higher drug overdose rates in males compared to females. The citation for this can also be found on slide 5. That's also what we have observed in our training data. For those of you doing the math to figure out how this relates to scores, it's a difference of 67. You'll see that pattern throughout the simulations when comparing the scores for males and females for the PDMP features that we will be looking at.

Slide 34:

Now, let's take a look at how the top PDMP feature in our training data (history of MOUD use) affects ORS.

This is the same 45-year-old self-reported female that we started with. The only thing that has changed is that we've simulated a history of MOUD use. This increased the base score from 316 to 446 – a swing of 130.

Slide 35:

And here we have the same 45-year-old self-reported male that we started with. Again, the only thing that has changed is that we've simulated a history of MOUD use. This increased the base score from 383 to 513 – a swing again of 130.

Slide 36:

Now let's take a look at how the next PDMP feature from a weighting perspective (i.e., Number of high-risk scripts) affects ORS.

Recall that a risk script is receiving a prescription for fentanyl patches, methadone, extended-release morphine formulations, or any oxycontin prescription or other opioid prescription with a daily MME > 120.

This is again the same 45-year-old female that we started with. We've removed the history of MOUD use and instead have simulated the effect of having 2 high-risk scripts in the most recent year. This increased the base score from 316 to 397 – a swing of 81.

Slide 37:

This too is the same 45-year-old male that we started with. Similarly, we've removed the history of MOUD use and instead simulated the effect of having 2 high-risk scripts. This increased the base score from 383 to 464 – the same swing of 81.

Slide 38:

So far, our simulations have only covered the top 2 PDMP features correlating with unintentional overdose death. We could keep going for the remaining features listed in the UI but I fear that we'd be too close on time and wouldn't have time for Q & A.

For a better sense of completeness, however, I can summarize what we would see in those simulations as follows: those deltas in scores that we just observed for the first two PDMP features get smaller and smaller as we make our way through the rest.

Let's end our simulations by examining the effect of a somewhat less correlated, but very salient feature that we try to account for in our model – chronic opioid use.

As mentioned, we identify this pattern of use by flagging continuous, nonoverlapping dispensations for opioids whose total days supply spans at least 80 days out of any given 90-day period in the most recent year. We realize this is not a perfect measure, nor a one-size-fits all approach, and we continually research ways to identify patterns of use that are associated with the treatment of chronic pain. That being said, we do think that this is a step in the right direction because when we are able to identify this particular pattern of use in our data, we have observed that it is negatively associated with scores. Meaning that this pattern is not associated with unintentional overdose death and thus decreases scores.

For our base 45-year-old female, having this pattern of use reduced the original score by 2 (316 to 314).

Slide 39:

For our base 45-year-old male, having this pattern of use reduce also reduced the original score by 2 (383 to 381).

Appendix:

Pillars of Fair AI

- Explainability
- Fairness
- Robustness
- Transparency
- Privacy

The “below average,” “average”, and “above average” text indicator for ORS within the tile is based on the distribution of scores obtained from a large PDMP patient population in 2020/2021 (i.e., the reference PDMP population). The below and above average thresholds currently represent the scores at the 25th and 75th percentiles from the reference PDMP population. For example, if a patient’s score is less than the score at the 25th percentile from the reference PDMP population, it is categorized as below average. Currently, this score is set to 260. If a patient’s score is greater than the score at the 75th percentile from the reference PDMP population, it is categorized as above average. If a patient from *any* s tate’s PDMP population has a score greater than the score at the 75th percentile for the Southeast PDMP population, it is categorized as above average. Currently, this score is set to 380.

ORS FAQ

- How did we get to percentage weightings (aka history of MOUD use having a ~40% influence)?

        - Let's take a step back and start with the model itself – it's a logistic regression model, which at a super high level correlates a bunch of variables (aka features) with a binary outcome. In this case, the outcome is a coroner adjudicated death due to unintentional overdose. So, for each feature that we include in the model, we get as output its relative importance/correlation with the outcome. This is represented by the logistic model as a beta coefficient. Which is hard to understand without a background in stats. Percentages are a lot easier to understand. So, I summed all of the beta coefficients (call this the denominator of the percentage) and divided each individual coefficient (call each of these the numerator) by the sum of all coefficients to arrive at the individual percentages. It's crude and the units are different, but the meaning is basically the same.
        - Are they modified in real-time?
            - No. After the model has been trained, the features that comprise the model and their relative contributions do not change. The only scenario in which they would is if we retrained the model with different data (same features, though) and noticed that the features were shifting around in terms of importance, or weight. Which is a great segue into the next question.
- How often are the contributing factors validated?
    - If the same PDMP data (aka features) are fed to the logistic regression model and the beta coefficients are re-examined and don't change, then we do not retrain the model. That means it's a pretty good model
        - The features (aka contributing factors) have been validated once since the initial model build
            - The original version used decedent data from a Midwest state, spanning the years 2013 to 2016)
            - And we validated the model with more recent decedent data from a different state (spanning 2017 through 2023).
                - The coefficients didn't change significantly, so we didn't retrain the model. Makes perfect sense.
    - We are now working with another state to obtain decedent data for another validation run. But to answer your question more directly – we retrain as often as folks let us have access to their decedent data
- How can we ensure that certain patients are not impacted based on race, socioeconomic status, age? And how can we be sure this information is used and validated appropriately?
    - Totally understand what you mean by this. Machine learning is inherently viewed as a form of statistical discrimination because models are designed to detect patterns based on input data (i.e., training data). Thus, statistical bias can always be present when the input data contains some form of unfairness/underrepresentation or bias. Because bias can espouse many forms, there is no universal definition of fairness that applies in all machine learning contexts. Each context can require different fairness and bias mitigation techniques. Bamboo Health has adopted a framework constructed of key pillars for principled machine learning practices: Fairness, Transparency & Explainability,

Robustness, Accountability, and Ethics. We draw from these pillars to identify, measure and address potential bias in our models.

- So, to answer your question more directly - when we detect any sort of imbalance across categories of demographic attributes, we apply the SMOTE technique. SMOTE stands for synthetic minority over-sampling technique. And rather than get into how it works at a super technical level, imagine instead that you're trying to teach a computer to recognize two types of flowers: roses and daisies. But you have a problem - you have 100 pictures of roses, but only 10 pictures of daisies. This imbalance can make it hard for the computer to learn about daisies properly. SMOTE is like a clever technique to help with this problem. Here's how it works:
    - It looks at your small group of daisies.
    - It picks one daisy picture.
    - It then finds the most similar daisy pictures to that one.
    - It creates a new, synthetic daisy picture that's a mix of the original and one of its similar neighbors.
    - It does this multiple times until you have about as many daisy pictures as you do rose pictures.
- These new, synthetic daisy pictures aren't exact copies of your original ones. They're like "inspired-by" versions that capture the essence of what makes a daisy look like a daisy. By doing this, SMOTE helps balance out your dataset. Now your computer has a fair chance to learn about both types of flowers equally well. So, this technique is super useful in many real-world situations where you're trying to teach computers about rare events or underrepresented groups in your data.
- Now, let’s go back to the second part of the question for how we can be sure this information is used and validated appropriately. A couple of years ago, I developed a Fair AI policy which includes the pillars mentioned a few minutes ago, as well as standard practices for machine learning modeling (e.g., how to do train/test splits, scaling, resampling techniques such as SMOTE, as well as interpretation of model performance statistics). So, if I or anyone else on the team gets hit by a bus or other object with sufficient mass to terminate our current conscious experience, there is a policy in place that encapsulates our practices.

- How many users are reaching out/what types of discrepancies are being raised related to ORS?
    - Since the newer version of the ORS model went live for all of our customers, we have only had a handful of folks reach out (i.e., specifically 3), all with the same question.
        - My patient has no controlled substance history, yet they have a score.
            - Gender
            - Age
    - What is our review process?

- Since there are not very many folks reaching out, we review their commentary on a quarterly basis and provide detailed explanations to any questions they've posited.