Today, I will present the Traumatrix project, which consists in improving triage and ressouces allocation for Trauma patients. This project can be considered as a **comprehensive case study** illustrating the many steps and challenges involved **in bringing a machine learning model from development to clinical deployment**. Today, the Traumatrix system is on its way to being deployed **across 16 regions in France and approximately 150 hospitals**. And as you can imagine, I can already give you a preview of the main message: t**he model itself represents only a small fraction of the overall effort.**

Before starting, I would like to briefly introduce my team, **Premedical**. We are a team of 25 researchers in statistics, biostatistics, AI, and clinicians. We cover different stages of the healthcare data lifecycle. From data storage, through federated learning approaches to learn from decentralized hospitals, to data analysis, we develop methods to address key challenges such as missing data in multimodal settings. We also specialize in policy learning methods, aiming at recommending the right treatment for the right patient at the right time, while ensuring data privacy, which is essential given the sensitivity of healthcare data. Our applications span several clinical domains, from fertility to neurodegenerative diseases, and today **I will present a collaboration with ICU clinicians working on trauma patients.**

Trauma patients are individuals with injuries that pose an immediate threat to life or limb. Major trauma includes conditions such as traumatic brain injury or hemorrhagic shock resulting from road traffic accidents, falls, or penetrating injuries. **It is one of the leading causes of death and disability worldwide, particularly among young adults.** These patients require highly **specialized care in dedicated trauma centers, especially Level I trauma centers**. **The earlier the appropriate resources—from blood transfusion to emergency surgery—are available, the greater the patient's chances of survival and functional recovery**. We need to keep in mind that the trauma patient's pathway is long, from the accident scene to hospital admission and then through rehabilitation, with the ultimate goal of returning to an independent life. However, managing trauma patients is extremely challenging. It involves multiple stakeholders, communication is often difficult, and Routing decisions must be made early, with incomplete information before complete diagnosis and the decisions must be made in a highly stressful environment **where every minute counts.**

Studies have shown that even experienced clinicians may miss critical conditions—for example, internal bleeding in a patient presenting with severe head trauma. Fatigue, particularly during overnight shifts, further increases the risk of errors. As a result, some patients are not transported to the appropriate trauma center, l**eading to substantial human costs.** Moreover, **uncertainty may encourage clinicians to adopt a precautionary approach** by referring more patients to trauma centers. Although this reduces the risk of under-triage, it may substantially increase **healthcare costs** by unnecessarily activating specialized trauma teams and consuming limited hospital resources.

The goal is therefore to develop machine learning models that improve pre-hospital triage by anticipating patients' needs before a complete diagnosis is available, directing them to the appropriate trauma center before arrival, and allowing critical resources to be mobilized in advance—for example, preparing blood products or alerting surgical teams

before the patient reaches the hospital. Almost ten years ago, I met Tobias Gauss, the ICU physician leading the Traumatrix project, over a coffee. **He introduced me to this clinical challenge,**

And he explained that the goal was to predict two key outcomes in order to improve trauma triage: the risk of hemorrhagic shock and the need for emergency neurosurgery, both defined according to established clinical criteria. The main constraint was that the **predictions had to rely exclusively on pre-hospital variables**—that is, information available at the accident scene or in the ambulance before the patient reached the hospital.

From a machine learning perspective, this initially appeared to be a fairly standard prediction problem: two binary outcomes, tabular data, and a relatively large dataset. More specifically, we used a trauma registry comprising approximately 65,000 patients collected between 2011 and 2026, with around 300 prehospital variables and approximately 4,000 new patients enrolled each year. The task is a binary classification problem. We selected two tree-based models, XGBoost and probability forest (from the grf package), which is consistent with the literature showing that tree-based methods remain state-of-the-art for tabular clinical data. **We also chose these models because, as I will show later, they naturally allow us to quantify several sources of uncertainty. We compared their performance with a simpler baseline, logistic regression, which already achieved excellent predictive performance**, as well as with recent tabular foundation models out of scientific curiosity.

Tobias approached me because he knew that my research focused on missing data, and their database contained a substantial amount of missing values. My initial objective was therefore to develop predictive models that could effectively handle missing covariates in this setting. Here is a bar plot showing the percentage of missing values for a subset of features, and as you can see, it varies from 0 to nearly 80 percent for some features. Of course, it is not possible to delete observations with missing values, because it can lead to some bias if you delete a subpopulation. But in any case, it will lead to deleting almost all of your data as soon as you are in medium dimensions. As an example, consider a case with d features, where each feature has a 0.01 probability of being missing. With d=5 features, complete-case analysis will lead to keeping almost 95 percent of your data, while with d=300, it will lead to keeping less than 5 percent.

Another specificity, quite unique to Traumabase, is that missing values are encoded using several different labels because clinicians originally recorded the reason why a measurement was unavailable. For example, **ND** stands for "*Non Disponible"*: the measurement was simply not collected. But there is also **IMP**, meaning that the patient's condition was so critical that the measurement could not be performed. In this case, the missing value is likely to carry important clinical information, as it may indicate an extremely low or extremely high underlying value. Finally, some entries are labeled **Not Applicable**. These are not truly missing values in the statistical sense, because they do not conceal an unobserved measurement. Rather, the variable simply does not apply in that particular clinical situation—for example, the variable blood pressure is NA when there is a heart attack. Or when there are nested questions. For instance, we may have a treatment variable, **osmotherapy**, with two options, Yes or No, and another variable for the **type of osmotherapy**, either SSH or Mannitol. When the first variable, Osmotherapy, is No, the second variable is NA. Therefore, we cannot treat all missing values in the same way. We

first leverage the available clinical information to logically impute non-applicable values or appropriately recode these variables. For instance, we can set the heart rate to 0 in the case of cardiac arrest. Or, we can create a single variable with three categories instead of two variables: **Osmotherapy No, Osmotherapy Yes — SSH, and Osmotherapy Yes — Mannitol. For the remaining missing values, we rely on the theoretical framework that we developed at the time. Interestingly, this theory turned out to be very simple: in a predictive setting, imputing missing values with a constant is actually Bayes-optimal under very weak assumptions**. **This result greatly simplifies the practical implementation of missing data handling.** For the probability forest, we used a method specifically designed to handle missing data in forests, implemented in the GRF package, which is also very similar in practice to constant imputation.

This conclusion was not immediately obvious to me. For years, I had been explaining that mean imputation was one of the worst things one could do. And if you look at this simulated example, it is clear why: mean imputation distorts both the marginal and joint distributions of the data.

However, this reasoning comes from an inferential perspective. **The appropriate way to handle missing data depends fundamentally on the objective of the analysis. Different goals require different strategies**: estimating parameters as accurately as possible, recovering the underlying data distribution, or making the best possible predictions. We created a website, **Rmistatic**, gathering many resources on missing values, data analysis pipelines, lectures, tutorials, and so on. The goal is to give users a key to enter the field and better understand how to deal with missing data. Few years ago, we have listed more than 150 different implementations, which illustrates the fact that there is no single way to deal with missing values.

When we consider the predictive setting, the problem is actually not trivial. More precisely, let's denote by X the complete features, and by X~ the features with missing values. We can see that X~ does not live in a continuous space anymore, but rather in a semi-discrete space, which makes the learning problem more complicated. We denote by M the mask, or indicator matrix, which codes for the missing values: 1 when a value is missing and 0 when it is observed. Our goal is to find the prediction function that minimizes the expected risk. The optimal prediction function is known as the Bayes predictor. The difficulty is that the prediction function takes X~ as input. So the Bayes rule, which is the conditional expectation of Y given X~, essentially amounts to having a different prediction rule for each possible missing-data pattern. In my example, we have only 2 missing-data patterns out of the 8 possible patterns, so we can quickly face a combinatorial issue.

However, we proved the following theorem: **if we learn on mean-imputed training data, use the same mean to impute the test set, and then predict, this is Bayes-optimal when the missing data are MAR and the learning algorithm is universally consistent.** In other words, with an infinite amount of data, the algorithm reaches the Bayes rate, which is the best possible error. The intuition is that the missing values can actually be seen as a code: a specific value for categorical variables, and a constant for numerical variables. The learner can detect this code during training and recognize it at test time. This is why it is important to use the **same constant** for imputation: otherwise, we introduce a distributional shift between the training and test sets. The learner can therefore **recover the missing-data**

pattern from the imputed data, and implicitly adapt its prediction to that pattern. Of course, this is an asymptotic result: it requires a lot of data and a universally consistent learner. But it provides a theoretical justification for a practice that is already widely used in practice.

We have extended this result to deterministic imputation (function of observed values such as regression but not stochastic imputation). We then define the impute then regress procedure with g is a learner, phi is an imputation, and we can show that for all missing data mechanisms (MAR, MCAR, MNAR) & almost all imputation functions, impute then regress is bayes optimal, which states that asymptotically imputing well is not needed for prediction. Of course this is asymptotic and for finite sample, the best can be to jointly learn imputation and regression.

At this stage, we had developed predictive models that could handle missing values in the covariates. However, when looking more carefully at the outcome we wanted to predict, I realized that this was not simply a standard binary outcome problem. **The reason is that competing events can occur**—in this case, mortality. For example, the label "neurosurgery: yes or no" c**an be negative simply because the patient died before receiving neurosurgery.** When death occurs after 24 hours, since we are interested in early interventions, it is reasonable to consider that the patient did not require neurosurgery. **However, when death occurs within the first 24 hours, the interpretation becomes much less clear: is this truly a negative label?** The answer is not obvious. Several strategies can be considered to address what is effectively **a missing outcome problem**. One possible solution is reweighting: individuals with observed outcomes can be weighted so that they better represent individuals whose outcomes are not observed. After a careful investigation, we chose instead to exclude these patients from the training dataset. **We found no systematic differences between patients with observed and non-observed outcomes. We also retrained the predictive models using several strategies to handle these missing outcomes, and the results were remarkably consistent—not only in terms of overall performance, but also at the individual prediction level.**

As I mentioned earlier, the prediction models had to rely exclusively on pre-hospital variables. **We also had an important constraint regarding the number of variables that could be collected**. **Indeed, for deployment, emergency dispatchers cannot realistically fill in an application requiring hundreds of inputs.** We therefore had to limit the model to **approximately ten variables**. There are many methods that can identify, among the 300 available pre-hospital variables, the most predictive ones. However, the real challenge **was not how to select the most predictive variables, but rather which variables should be considered appropriate to use**. Indeed, not all available variables play the same role. Some variables are direct clinical measurements, such as blood pressure, **while others are intervention or treatment variable**s, such as intubation. **The latter are not intrinsic patient characteristics but decisions made by physicians.** These variables turned out to be highly predictive of our outcomes, as they are also strong markers of severity. Another important consideration: **trauma systems are highly context-dependent, and each healthcare system has its own organization and clinical practices**. Some interventions may be very specific **to the French trauma system**. As a consequence, models relying on such variables may have limited generalizability, which is a **frequent concern among clinicians and in medical publications**. However, I believe this argument needs to be nuanced. **What can be generalized is not necessarily a single**

universal model, but rather a methodology and a framework that can be adapted to each healthcare system. Since our objective was to deploy this model within the French trauma system, we decided to retain these variables in the final models.

With only around ten variables, we obtained the following performances using three different models: XGBoost, a probabilistic random forest, and logistic regression (using an SAEM approach to handle missing values). We considered it essential to compare our approach with simple and well-established baselines, as these remain important references in clinical applications. The clinicians defined a key requirement for deployment: **the false negative rate had to remain below 15%, as missing a patient requiring critical care could have severe consequences.**

At this stage, it becomes essential to quantify the uncertainty associated with these predictions. Several approaches can be considered for this purpose. We retained **probability forest** and **XGBoost** not only because of their predictive performance, but also because they naturally support different approaches to quantifying predictive uncertainty. For **probability forest (in GRF package)**, which I use extensively in classification settings, we can estimate both the probability that a patient will develop hemorrhagic shock and a **confidence interval** around this probability. Importantly, this confidence interval is **conditional on the patient's covariates** and comes with **asymptotic theoretical guarantees**. In practice, for each patient, we obtain both a predicted risk and an estimate of the uncertainty surrounding that prediction. For **XGBoost**, we instead relied on **conformal prediction**, one of the most widely used uncertainty quantification frameworks. Its popularity stems from its **finite-sample validity guarantees**. However, these guarantees are **marginal**, meaning they hold on average over the population rather than conditionally on individual patient characteristics. Another limitation of existing conformal methods is that they typically ignore uncertainty induced by **missing covariates**. As a result, the actual coverage can vary substantially across different missing-data patterns. Together with collaborators, a former student investigated this issue and developed methods to address it, although I will not go into those details today due to time constraints.

We decided to **combine these two models** and their two complementary approaches to uncertainty quantification, which rely on different assumptions. For GRF a prediction is considered positive when the entire confidence interval lies above the decision threshold, and negative when it lies entirely below. Otherwise, the prediction is considered uncertain. We then combined the outputs of the two models into a **five-level prediction scale**. When both models are uncertain, we return **Uncertain**. When both predict a negative outcome, we assign **High-confidence Negative**; when both predict a positive outcome, we assign **High-confidence Positive**. If one model predicts positive while the other is uncertain, we return **Low-confidence Positive**. Finally, if one predicts negative and the other is uncertain, we return **Low-confidence Negative**. This decision framework was developed in close collaboration with our clinical partners.

Once such an uncertainty scale is defined, the next question becomes **how to evaluate it**. Unlike a standard binary classifier, performance now depends on **how clinicians choose to interpret the different confidence levels**. To address this, we introduced three evaluation scenarios. The first, **Safety**, corresponds to **a risk-averse clinician**: every uncertain prediction is treated as positive to avoid missing patients requiring neurosurgery. The

second, **L2**, evaluates performance only on resolved cases—that is, positive and negative predictions—while leaving uncertain cases to clinician judgment. Finally, **L3** evaluates the algorithm only when it is highly confident, considering only high-confidence positive and high-confidence negative predictions.

The results are quite encouraging. In the Safety scenario, although the model was never explicitly optimized to achieve a false negative rate below 15%, it nevertheless satisfies this clinical requirement and even achieves a lower false negative rate than either individual model. As expected, this comes at the cost of a higher false positive rate, reflecting a more conservative strategy. At the opposite extreme, when we focus only on high-confidence predictions, the algorithm makes remarkably few mistakes. Of course, this also means that clinicians remain responsible for the more ambiguous cases. This just gives few ideas on what can happen and much more could be done. An interesting question, which we have not yet investigated, is whether the cases that are easy for the algorithm are also the ones clinicians find straightforward.

We also performed a **feature omission trial** to evaluate the robustness of our five-level prediction system to missing input variables. In real-world deployment, missing data are inevitable. **As more variables are removed, the model becomes increasingly likely to return an Uncertain** prediction rather than a confident recommendation. This is actually a desirable behavior, as uncertainty increases when less information is available. Importantly, some confident positive predictions are still maintained, which also contributes to patient safety.

At this stage, we felt that we had thoroughly stress-tested the models. We evaluated many sources of uncertainty—different predictive models, missing-data strategies, variable selection procedures, competing-risk assumptions, and many additional sensitivity analyses that I have not described today. More importantly, we explicitly attempted to quantify **when we should trust the model and when we should not**.

However, before deploying such a system, a more fundamental question arises. Rather than focusing only on predictive metrics, **we should ask whether the model is actually likely to improve patient care**. First, we should remember that our outcome is **not an objective outcome**. **The label "Neurosurgery: Yes" or "No" reflects a decision made by a clinician.** Consequently, **our models learn to reproduce historical clinical decisions**, not whether neurosurgery was truly the optimal treatment. Our clinical collaborators were comfortable with this objective because, in practice, the model can be viewed as reproducing the judgment of an average experienced clinician—potentially more consistent than an exhausted physician making decisions at three o'clock in the morning. Nevertheless, the model is fundamentally learning the **average historical clinical judgment** encoded in the registry, together with all of its strengths and biases. We therefore have to keep in mind that **high predictive accuracy does not necessarily imply clinical optimality**. Ideally, we **should reason causally from the outset** and ask a different question: ***which patients would actually benefit from neurosurgery?***

This next slide simply illustrates that change in perspective. **Instead of predicting whether a patient historically received neurosurgery**, we now compare the patient's expected outcome under alternative treatment decisions. In this framework, **the previous prediction**

target becomes the treatment, while the outcome of interest becomes something clinically meaningful, such as 28-day survival.

Before introducing policy learning, I would like to mention another important issue concerning **model evaluation**. **If we deploy a predictive model and continue evaluating it using the same outcome that it predicts, we may incorrectly conclude that the model improves performance simply because it changes clinical practice**. This phenomenon is known as a **self-fulfilling prediction**. For example, if the model predicts that a patient does not require neurosurgery, the patient may be transferred to a lower-level trauma center where neurosurgery is unavailable. Unsurprisingly, the observed outcome will then be "No neurosurgery," apparently confirming the model's prediction—even though the prediction itself influenced the outcome. The model appears highly accurate simply because it helped create the outcome it was evaluated against. Fortunately, **our clinical trial avoids this issue.** Rather than evaluating predictions against the neurosurgery decision itself, we assess their impact on under-triage and over-triage, which are independent clinical outcomes.

Let us now return to **policy learning**. The objective is to identify, for each patient, the “best treatment”. More precisely, we are looking for treatments that maximize the **policy value**, that is, the expected outcome if everyone were treated according to that policy. In our setting, this corresponds to identifying the treatment strategy expected to maximize 28-day survival. Under the standard assumptions of causal inference—consistency, positivity, and no unmeasured confounding—a large number of policy learning methods have been proposed, including policy trees and CATE-based approaches such as X-, R-, and DR-learners. Ironically, t**his abundance of methods can itself become a barrier to clinical adoption**, because **different methods often recommend different treatments** for the same patient (despite having values that are significantly non-different from each other). On this slide, I show both the proportion of patients recommended for neurosurgery by four representative methods and several examples of individual recommendations where they disagree.

Perhaps more importantly, all of these approaches return **a single treatment recommendation**. They provide no indication of the uncertainty associated with that choice. Yet two treatments may lead to statistically indistinguishable outcomes, in which case recommending only one is arbitrary. Even worse, when there is **no treatment effect**, forcing the algorithm to choose one treatment makes little sense. This issue becomes even more problematic in settings involving multiple treatment options rather than a simple binary decision. Surprisingly, until very recently, there was no principled framework to communicate this type of uncertainty. Together with Mathieu Even, Uri Shalit, and Antoine Chambaz, our PhD student Laura Fuentes, recently proposed **set-valued policy learning**, which explicitly returns the set of treatments that are statistically compatible with being optimal, instead of forcing a single arbitrary recommendation. I think this provides a much more natural way to communicate uncertainty and should facilitate the clinical adoption of policy learning methods. Another advantage of returning a **set of possible treatments** is that clinicians can incorporate additional criteria that may not be available to the model, such as choosing the treatment with the **fewest side effects**, the **lowest cost**, or other patient- or context-specific considerations. This fits within the paradigm of learning to defer.

We therefore applied our **set-valued policy learning** approach, again performing sensitivity analyses across different methodological choices. We compared its recommendations—identifying patients who would benefit from neurosurgery to maximize survival—with the predictions of our models, which do not use survival as an outcome. In a way, we use this as an **audit of our predictive models**. The results are reassuring: the two approaches agree on the most extreme cases. Patients predicted to benefit from neurosurgery by the policy are generally also classified as **"Neurosurgery"** by our models.

As I mentioned, the project is now moving into the deployment phase. More specifically, the predictive models will be made available to the SAMU dispatch centers. When an accident occurs, t**he first responders at the scene contact the SAMU dispatch center** and provide information about the patient. **The dispatcher then fills in the interface, reviews the model results, and makes the final orientation decision.** We will then follow the patient's outcome.

More specifically, the interface works as follows. Before seeing the model's predictions, we first ask the physician for their own assessment: do they think the patient is likely to develop hemorrhagic shock or require neurosurgery? We use five categories: very low, rather low, rather high, high, and uncertain. **The idea is to explicitly capture uncertainty in human judgment as we**ll. Choosing the right wording for these categories was actually quite challenging. The dispatcher then enters around ten variables, including age, heart rate, pupillary abnormalities, and so on. The model results are then displayed using a color code: red indicates a very high risk of hemorrhagic shock or a very high likelihood of requiring neurosurgery. This color coding corresponds to the matrix obtained by combining the two models. We then ask the physician for the patient's **desired orientation**, followed by the **actual orientation**. Indeed, even if the risk of hemorrhagic shock is very high and the patient would theoretically be best directed to a level-1 trauma center, practical constraints may lead to a different decision. For example, if there is no level-1 center nearby, it may be safer to stabilize the patient in a level-2 center rather than expose them to the risks of a very long transfer.

We will therefore launch a clinical trial next September across 16 regions in France. The objective is to evaluate the impact of this decision-support tool on under-triage and over-triage. It is a cluster cross-over randomized trial: eight dispatch centers will first be in the intervention group for nine months, followed by a washout period and then nine months in the control group. The other eight centers will follow the opposite sequence, starting in the control group and then switching to the intervention group. This trial will allow us to compare **Human + AI versus Human alone**, and we will also explore whether the effect of the AI alone can be estimated through counterfactual reasoning.

Of course, I have mainly presented the machine-learning component of the project, but deploying such a system requires much more. There are major regulatory constraints, including obtaining patient consent for the study, which is particularly challenging given the nature of the population and the severity of the accidents. We also need to make sure that there are no technical barriers, such as hospital firewalls, that could interfere with deployment. All dispatchers also had to be trained to use the tool and to handle different

practical situations. From the beginning, we tried to integrate social-science researchers into the project. Questionnaires were used during the interface design process, for example to make sure that data entry would not take too long not to disrupt the workflow. For the clinical trial, this collaboration has been strengthened: two researchers will be based in two dispatch centers to observe how dispatchers interact with the tool over time, and several ancillary studies will also be conducted.

To conclude this journey, I would like to highlight a few lessons for safe clinical AI. First, extensive **sensitivity analyses** are essential. We need to understand how methodological choices affect the outputs. I have already presented several of them, but we performed many additional analyses (audit of fairness, impact of some distributional shifts whether temporal or in population, assessing using one multi-ouctome model instead of two models, etc). Ultimately, this allowed us to be confident in the deployed model and to consider it stable: not only were the overall results stable, but even individual predictions changed very little across analyses. Quantifying uncertainty in the predictions was also crucial from the beginning, both for us and for the physicians. And we tried to **document and trace all of our methodological choices** as carefully as possible. This approach is, I think, very much in line with the principles of **Veridical Data Science** proposed by Bin Yu.

There are, however, several limitations worth mentioning. First, our process is still a **two-step procedure**: we train models on historical data and then study how they interact with physicians. Ideally, we would want to take this human-AI interaction into account from the very beginning when developing the models. Second, we need to be careful because our predictions may eventually **change the data themselves**. Future datasets may be influenced by the predictions and decisions generated by the system, bringing us into the paradigm of **performative learning**. Finally, perhaps the most fundamental limitation is that we are doing **prediction rather than causal inference**, and we may therefore question whether prediction is the right objective in the first place. However, causal approaches rely on strong assumptions and require extensive sensitivity analyses as well. In addition, clinicians rightly point out that we may not have measured all relevant confounders and policy learning methods were not from my point of view ready for deployment, not enough mature, with not uncertainty quantification. At the same time, clinicians believe that the tool could be useful in practice: it may reduce cognitive load and support physicians in difficult, stressful situations. Even the simple process of entering the relevant information may itself be useful. Testing this specific effect would ideally require a three-arm trial, which we did not design here. Our prospective trial will nevertheless allow us to evaluate whether the tool actually improves care, and the causal audit we performed was reassuring.

So, ultimately, as a colleague put it: **“It works” is no longer sufficient. The real question is whether AI improves healthcare.**