Before Petrophysical Machine Learning

Petrophysical machine learning treats missing data as a preprocessing afterthought, a line of code that drops or fills the gaps so training can begin. We argue the opposite: that the most consequential modelling decisions are taken before any model exists, in a diagnostic pass that profiles, visualises, and reasons about missingness, and that skipping that pass is the most common and most expensive mistake in subsurface ML. A well-log table is missing data in three distinct registers at once. At the column level, the public Norwegian-Sea teaching slice (118 wells, 22 electrical-measurement columns, profiled with the missingno library by McDonald) shows a handful of near-complete primary curves above a long tail of sparse specialist tools, and the keep-or-drop decision on each column changes what the model can even learn. At the corpus level, a scanned regulatory archive is missing labels wholesale: the public Texas Railroad Commission collection holds 136,771 raster TIF files against only 7,781 digital LAS files, so at most a few percent of the imagery can ever be paired into supervised examples, which is the entire reason a 20,000-log synthetic corpus had to be manufactured. At the pipeline level, a missingness decision taken or skipped upstream propagates silently through the 80/20 train/validation split to the validation metric, where our multiclass model recorded a background IoU of 0.94 against a curve-1 mask IoU of just 0.26. We set out a diagnostic protocol with three steps, profile then visualise then decide, walk three interactive exhibits that argue each register of the problem, and state the operating rule we now apply: no corpus enters training until its nullity has been profiled, its label coverage has been mapped, and the consequences of every drop, fill, or synthesis have been reasoned about in the open. This document is distinct from our companion whitepaper on curve-extraction evaluation; that one grades the deliverable after the model runs, this one disciplines the data before it does.

The EarthScan Team
Research, engineering, and field

MISSING-DATA / DATA-DIAGNOSTICS / MISSINGNO / DATA-PROFILING / IMPUTATION / WELL-LOGS

CONTENTS

> The model you eventually train is downstream of a decision nobody scheduled: whether to look at the data first. The teams that look spend half an hour. The teams that skip it spend a month wondering why the loss will not move.

I

THE CASE

The discipline that has to precede the model

Executive summary

A petrophysical machine-learning project is usually narrated as a modelling story. The data exists, the preprocessing is a footnote, and the interesting work begins at the architecture. We have run enough of these to believe the narration is backwards. The decisions that most determine whether a subsurface model works are taken before any model exists, in a diagnostic pass over the data that profiles its gaps, visualises where they cluster, and reasons explicitly about what to keep, what to drop, and what to fill. That pass is short, unglamorous, and routinely skipped, and skipping it is the most common and most expensive mistake we see in the field.

This whitepaper is the argument for the pass, made concrete on real data at three scales. At the column scale, the public 118-well Norwegian-Sea teaching slice carries 22 electrical-measurement columns whose completeness ranges from near-total to almost absent, and the keep-or-drop call on each one changes what a model can learn [1][2]. At the corpus scale, a scanned regulatory archive is missing labels in bulk: the public Texas Railroad Commission collection holds 136,771 raster TIF files against only 7,781 digital LAS files, which caps the supervisable fraction at a few percent and is the reason we manufactured a 20,000-log synthetic corpus instead of hand-labelling. At the pipeline scale, a missingness decision taken upstream travels silently through the 80/20 train/validation split into the validation metric, where our multiclass segmentation model recorded a background IoU of 0.94 against a curve-1 mask IoU of just 0.26. Three exhibits below let you operate each scale directly. The operating rule we land on is one sentence: no corpus enters training until its nullity has been profiled, its label coverage has been mapped, and the consequence of every gap has been reasoned about in the open.

Profile the nullity

- Count present versus missing for every column, not just the dataset overall
- Separate the easy primary curves from the sparse specialist long tail
- On the 118-well slice this is 22 columns to characterise individually
- The output is a per-column missing share, the input to every later decision

Visualise where gaps cluster

- A nullity matrix shows whether gaps are scattered or aligned across wells
- Aligned gaps signal a structural cause, a tool that was never run, not random loss
- Co-missing columns reveal which features rise and fall together
- Seeing the pattern is what tells you the mechanism, which decides the remedy

Decide before you train

- Keep, drop, or fill each column, on the evidence, with the reason recorded
- Map label coverage: how many examples can be supervised at all
- Set the train/validation split knowing what each fold actually contains
- Only now does the first epoch run, against a corpus you understand

A note on scope. We have a companion whitepaper that grades a digitiser on the curve it emits rather than the mask it passes through, an evaluation argument that lives after the model runs. This document is its mirror image and deliberately upstream of it: it disciplines

the data before the model exists. The two share a project and a vocabulary but answer different questions, and neither substitutes for the other.

II

WHY NOW

Why the diagnostic pass keeps getting skipped

The problem is structural, not lazy

It would be comfortable to say teams skip data diagnostics out of haste, and sometimes they do. But the skip is structural. The tooling, the tutorials, and the incentives all point at the model. A new architecture is a publishable result; a careful nullity profile is not. A training run produces a loss curve you can watch; a profiling pass produces a table you have to read. And the modern framework makes it frictionless to feed a dataframe straight into a model, gaps and all, so the path of least resistance runs directly past the one step that would have caught the problem.

The cost lands later and looks like something else. A model that will not converge gets blamed on the learning rate. A validation metric that sits stubbornly low gets blamed on the architecture. A feature that should matter and does not gets blamed on the loss function. A surprising share of these are missing-data problems wearing a modelling costume: a column that is 80 percent absent contributing pure noise, a fold that drew the wells where a key tool was never run, a label set so sparse the model never saw enough of the hard class to learn it. The literature has been consistent on this point for decades; the missingness taxonomy exists precisely because the mechanism behind a gap dictates what you may legitimately do about it, and you cannot know the mechanism without looking [3]. A recent survey of missing-data handling in machine learning reaches the same conclusion from the applied side: detection comes first, and the choice between deletion and imputation is only defensible once the pattern is understood [5].

Subsurface data makes the problem unavoidable

Generic tabular data lets you get away with the skip more often than it should. Subsurface data does not, because well-log tables are missing data in a way that is both severe and informative. Curves go missing for reasons: a tool was not on the string for that well, a section was washed out, a service company logged a different suite, a scan lost a track. The gaps are rarely completely at random, which means the lazy remedies, drop the rows, fill with the mean, are not just suboptimal but can be actively misleading [3][4]. And because the same physical quantity is logged under different tool names across decades and operators, even counting the missing values correctly requires care before the profile is trustworthy.

That severity is also why subsurface data rewards the diagnostic pass so richly. The gaps carry signal. A column that is missing in exactly the older wells is telling you something about acquisition history. Two columns that are always missing together are telling you they came from one tool. Reading the nullity pattern is reading the acquisition story of the archive, and that story is what tells you whether a gap is safe to fill, safe to drop, or a warning that the feature is unusable [6]. The rest of this document walks the three registers in which that story has to be read, each with an instrument you can operate.

III

REGISTER ONE

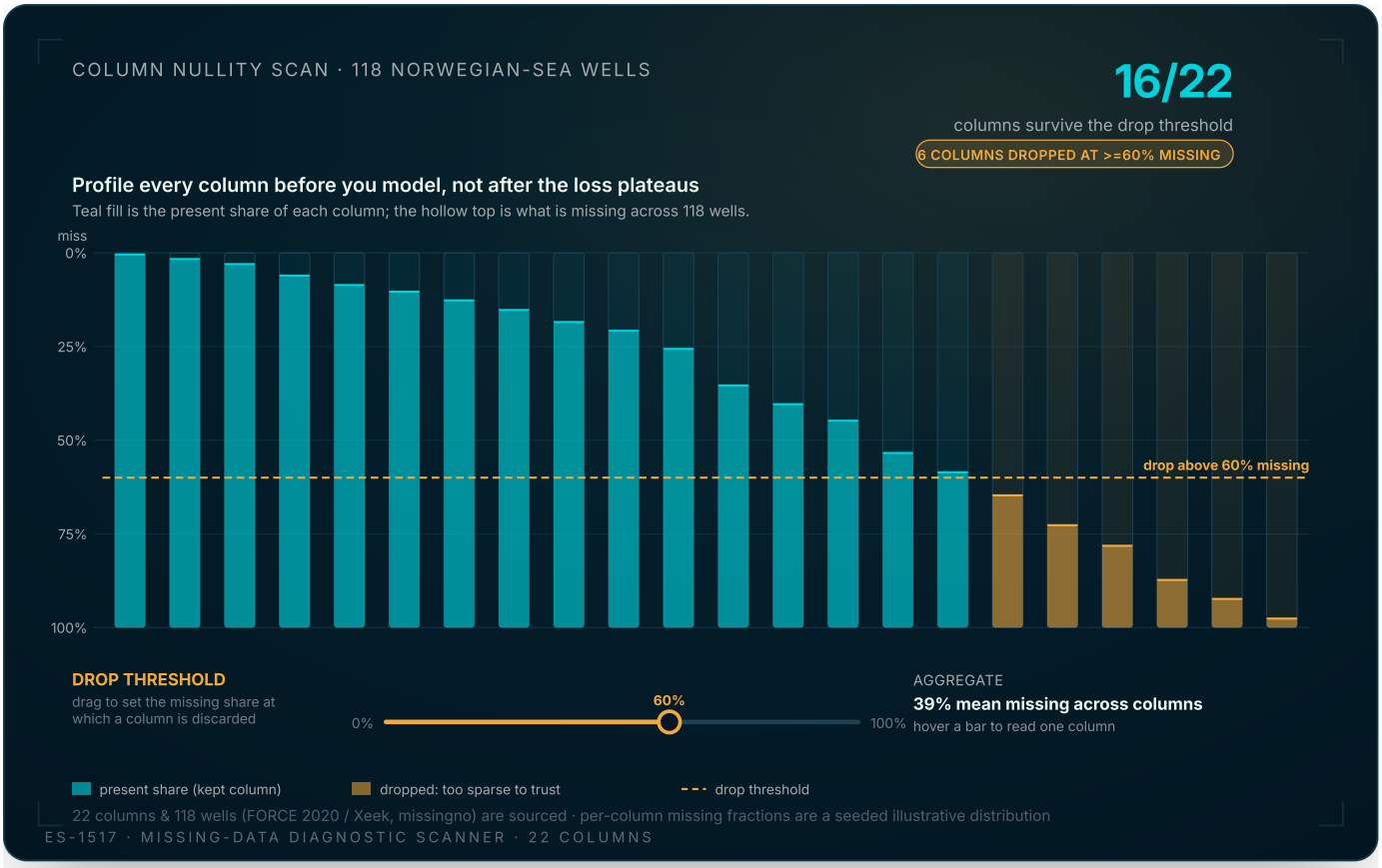
Column nullity: the keep-or-drop decision

Profiling every column, not the dataset

The first register is the one the missingno tutorial makes legible: column-level nullity on a clean, vector well-log table [1]. The public Norwegian-Sea slice is the right object to reason on because it is public, model-ready, and exactly the shape a petrophysicist trusts: 118 wells, depth-indexed, with 22 electrical-measurement columns whose names are the standard FORCE 2020 vocabulary [2]. It is the dataset a team would actually start from, and its missingness is representative of the genre.

What a column profile reveals is a shape, not a single number. A handful of primary curves, the depth index, the gamma-ray, the bulk density and neutron porosity, the principal resistivities, are present in nearly every well, because they are the curves you run on everything. Below them sits a long tail of specialist measurements, the shear sonic, the micro-resistivities, the mud weight, that were run only where a particular question justified them and are absent everywhere else. The dataset-level missing percentage averages these two populations into a number that describes neither. The per-column profile separates them, and it is the per-column profile that the keep-or-drop decision actually needs.

The instrument below puts that decision in your hands. Each of the 22 columns is a bar whose teal fill is its present share; drag the drop-threshold lever and every column too sparse to clear it turns orange and drops out, while the headline counts how many survive. The point of operating it is to feel how sensitive the surviving feature set is to a threshold most teams never state out loud. Set the bar at a tolerant level and you keep almost everything, including columns so sparse they contribute noise. Set it strictly and you keep a clean, complete feature set that is also a smaller one. There is no universal right answer; there is only a decision, and the discipline is making it on the evidence rather than by default.



A column-by-column nullity scan over the 22 electrical-measurement columns of the public Norwegian-Sea well-log slice (Xeek / FORCE 2020, 118 wells, profiled with the missingno library by McDonald). Each column is a vertical bar: the teal fill is the present fraction, the hollow top is what is missing across the wells. Drag the drop-threshold lever and every column whose missing share crosses it turns orange and is counted as discarded, while the headline reports how many of the 22 survive. This is the decision the diagnostic discipline forces, settled by looking before a single epoch runs: which columns are complete enough to feed a model and which are too sparse to trust. The 22 columns, the 118 wells, and the missingno-profiled FORCE 2020 / Xeek slice are sourced; the per-column missing fractions are a seeded illustrative distribution spanning the realistic range a well-log table shows, not the tutorial's exact per-column audit.

The threshold is a modelling choice disguised as a data-cleaning step. A column kept at 70 percent missing will be imputed for most of its values, which means the model is largely learning from a fill rule rather than from measurement, and whether that is acceptable depends entirely on the mechanism behind the gap [3][4]. A column dropped at 40 percent missing throws away real signal in the wells that did carry it. Neither call is free, and the only wrong move is to make it implicitly by feeding the raw table to the model and letting the framework decide. We treat the threshold as a logged decision with a stated rationale, the same way we would treat a hyperparameter, because that is what it is.

There is a second reading the per-column bar chart does not give you, and it is the one that most often changes a decision: how columns are missing together. The missingno toolkit pairs its nullity bar chart with a correlation heatmap and a dendrogram precisely so this co-missingness is visible, and on a well-log table it is rarely an accident [1]. When the shear sonic and a micro-resistivity are blank in exactly the same wells, they were almost certainly absent because one logging run covered both; when a porosity curve is present wherever the density curve is, they came off the same pass. Reading those clusters tells you which gaps are independent, where filling one column from its neighbours is defensible because a correlated, present column carries the information, and where it is circular because the only columns that could inform the fill are missing in the same wells. A keep-or-drop call made on the single-column missing share alone can be reversed once the co-missingness is on the screen, which is why the visualise step is not a nicety layered on top of the profile but a distinct source of evidence that the count cannot supply.

> **THE RULE WE APPLY AT THE COLUMN LEVEL**
>
> Profile every column individually before training. State a drop threshold and the reason for it. For every column kept above a meaningful missing share, name the imputation method and check that the missingness mechanism makes that method valid. A column's fate is a recorded decision, never a framework default.

IV

REGISTER TWO

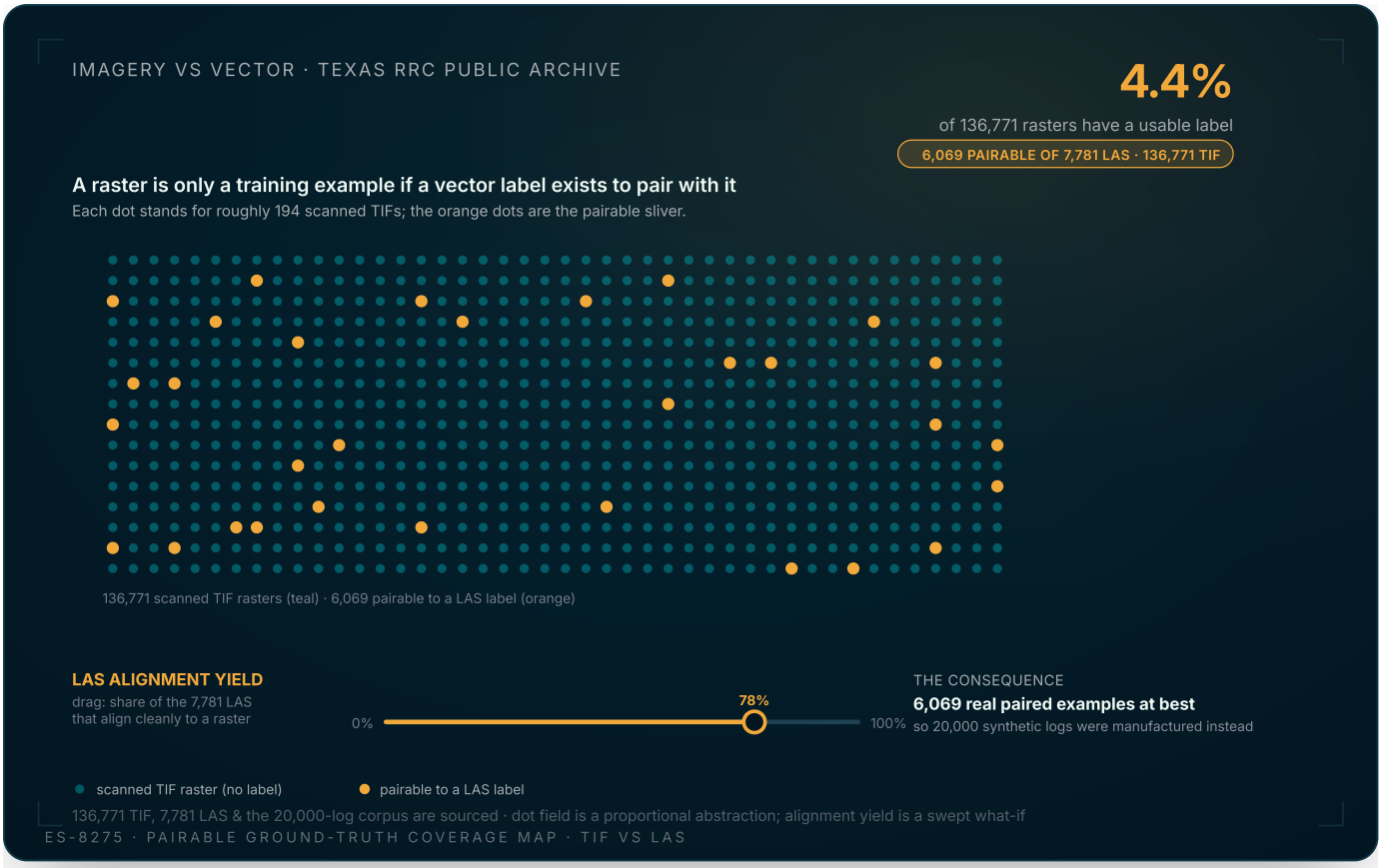
Label coverage: the missing data is the labels

When the gap is the whole supervisory signal

The second register is the one that does not appear in any single table, because the missing data is not cells in a dataframe; it is labels for an entire archive. This is the regime a scanned regulatory collection lives in, and it is where the diagnostic pass stops being about cleaning and becomes about whether supervised learning is even possible.

The public Texas Railroad Commission archive is the case we worked. It holds 136,771 scanned TIF raster images of paper well logs against 7,781 digital LAS files, an imagery-to-vector ratio of roughly 17.6 to 1. A raster TIF carries the curve a petrophysicist drew but no machine-readable values; a LAS file carries the values. To train a digitiser, which learns to turn the raster into the values, you need both for the same well: the image as input, the LAS as the label. The LAS count is therefore a hard ceiling on the supervisable fraction of the archive, and even that ceiling is optimistic, because not every LAS aligns cleanly to a matching TIF.

The coverage map below makes the ceiling tangible. The dot field is the imagery archive at proportional scale, each dot a block of scanned rasters; the orange sliver is the fraction that could be paired to a vector label, and the alignment-yield lever lets you sweep from the optimistic case, every LAS pairs, toward the realistic one. Watch how thin the sliver stays even at full optimism. That sliver, a few percent of the imagery at best, is not a number on a slide; it is the design constraint that reshaped the entire project.



The supervised-learning question underneath any missing-data audit on a scanned archive: a raster file can only be a training example if a matching vector file exists to supply its label. The public Texas RRC archive holds 136,771 scanned TIF rasters against only 7,781 digital LAS files, so even in the best case the LAS count caps the paired set at roughly 5.7 percent of the imagery, and once imperfect alignment is allowed the truly pairable sliver shrinks further. Each dot stands for a block of about 194 scanned rasters; drag the alignment-yield lever from optimistic toward pessimistic and the orange pairable sliver contracts while the readout reports the paired count and its share. That sliver is the whole reason a 20,000-log synthetic corpus had to be manufactured rather than hand-labelled. The TIF and LAS counts and the synthetic-corpus size are sourced engagement and Texas RRC figures; the dot field is a proportional abstraction and the alignment yield is a user-swept what-if, capped at the true LAS ceiling.

We confronted exactly this map and drew the only conclusion it allows: you cannot hand-label your way out of a 17.6-to-1 deficit, and the pairable real examples are too few to train a robust segmentation model on their own. So we manufactured the labels we could not pair, generating a synthetic corpus of 20,000 procedurally drawn logs whose ground truth is known by construction because we drew it. The synthetic corpus is not a workaround bolted on after a failed training run; it is the direct, planned consequence of a coverage diagnostic done before training. Reading the label-coverage map first is what turned an impossible supervised problem into a tractable one, and a team that skipped the map would have discovered the deficit the expensive way, by trying to train on the sliver and watching it fail to generalise [6].

> **“The coverage map is not a status report on the data. It is the document that decides whether you label, synthesise, or abandon the supervised framing entirely.”**

— FROM THE ENGAGEMENT'S DATA-READINESS REVIEW

V

REGISTER THREE

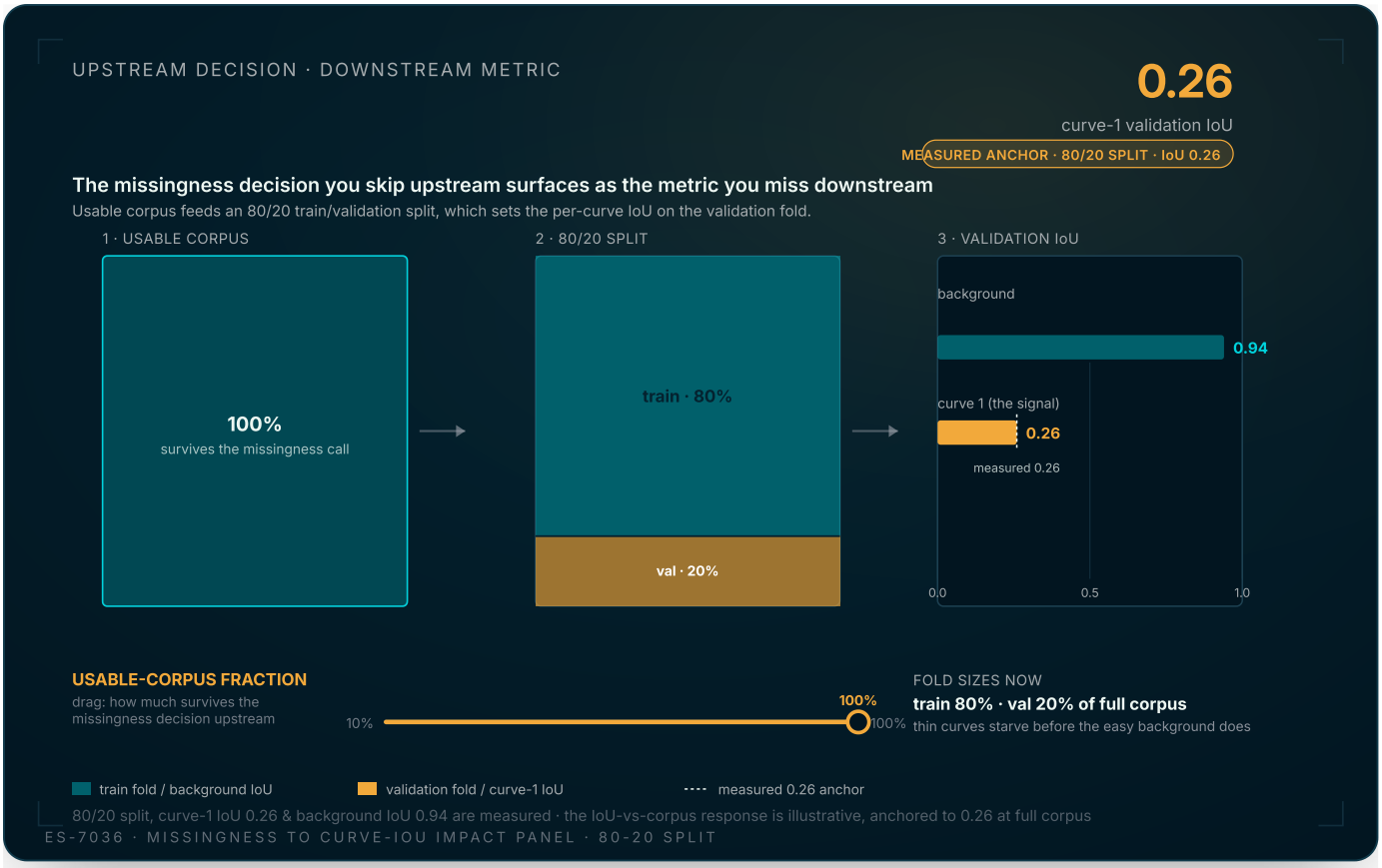
Propagation: how an upstream gap becomes a downstream metric

The decision you skip surfaces where you least expect it

The third register is the one that makes the case for diagnostics impossible to argue with, because it connects the upstream data decision to the downstream number a stakeholder actually reads. A missingness decision does not stay where it was made. It flows through the train/validation split into the validation metric, and by the time it surfaces there it no longer looks like a data problem at all.

Our multiclass segmentation stage is the worked example. The corpus that survived the upstream decisions was partitioned with an 80/20 train/validation split, and on the held-out fold the model recorded a background IoU of 0.94 and a curve-1 mask IoU of 0.26. Read naively, that spread looks like a model that finds empty space well and curves badly. Read with the data discipline in mind, it is partly a statement about what the corpus contained: the easy, dense background class had abundant signal in every example, while the thin curve class, sparse by nature and sensitive to how much usable data survived profiling, had far less to learn from. The upstream completeness of the corpus and the way the split allocated the hard class both shaped that 0.26, and neither is visible anywhere except in the metric where they finally land.

The impact panel below traces that propagation as three coupled stages: the usable corpus after a missingness decision, the 80/20 split applied to it, and the per-curve validation IoU that results. The real anchor is fixed, the measured 80/20 split and the 0.26 / 0.94 IoUs, and the lever lets you vary how much of the corpus survives upstream and watch the folds and a modelled curve-1 IoU respond. The thin curve starves faster than the easy background as the corpus shrinks, which is the whole point: the class that carries the signal is also the class most exposed to an upstream data decision, and the metric is where that exposure becomes visible, far too late to fix cheaply.



The downstream half of the diagnostic argument: a decision taken or skipped at the profiling stage propagates all the way to the validation metric. The panel reads left to right as three coupled stages: the corpus that survives a missingness decision, the 80/20 train/validation split applied to it, and the resulting per-curve IoU on the validation fold. The fixed real anchor is the engagement's measured multiclass outcome, an 80 percent training split that produced a curve-1 mask IoU of 0.26 against a background IoU of 0.94. Drag the lever to vary how much of the corpus survives upstream and the train fold, the validation fold, and a modelled curve-1 IoU all move with it, while the dashed marker pins the real 0.26 so the what-if is always read against ground truth. The thin curve starves faster than the easy background, which is exactly why the upstream skip is invisible until this last panel. The 80/20 split and the 0.26 / 0.94 IoUs are measured; the IoU-versus-corpus response is an illustrative monotone curve anchored to the real 0.26 at full corpus, not a fitted sensitivity.

The split itself deserves a sentence of its own, because it is the stage where an upstream gap can be amplified rather than merely passed through. A random 80/20 partition assumes the wells are interchangeable, and on a corpus with structured missingness they are not: if the wells that carry the hard class well happen to land mostly in the training fold, the validation number understates the model, and if they land mostly in validation, it overstates the difficulty. The diagnostic pass is what lets you stratify the split deliberately, so that the scarce, hard-to-learn signal is represented in both folds, rather than discovering after the run that the held-out number was an accident of how the partition fell. None of that is visible at training time. It is decided, knowingly or by default, at the moment the split is drawn, which is upstream of every epoch and downstream of the missingness profile.

This is the argument that closes the loop. The column profile and the coverage map are upstream documents; the IoU is a downstream number; and the line connecting them is invisible unless you drew it on purpose. A team that profiles, maps, and decides in the open can read a low curve-1 IoU correctly, as a partly-data signal that points at the corpus, and act on it. A team that skipped the diagnostic pass reads the same 0.26 as a model failure, reaches for the architecture, and changes the one thing that was not the problem.

VI

THE EVIDENCE

The numbers the discipline is built on

Results, stated against the three registers

The figures that anchor this whitepaper are deliberately drawn from one engagement and the public datasets it sat on, so they hang together rather than being assembled to flatter a point.

THE THREE REGISTERS IN NUMBERS

22

Electrical-measurement columns profiled on the 118-well slice

17.6:1

label deficit

Raster TIF to digital LAS ratio in the public archive

20,000

the response

Synthetic logs manufactured because labels could not be paired

0.26

vs 0.94 background

Curve-1 mask IoU on the 80/20 validation fold

At the column level, the public slice is 118 wells and 22 electrical-measurement columns, and the diagnostic deliverable is a per-column missing share that sorts the near-complete primary curves from the sparse specialist tail [1][2]. At the corpus level, the archive is 136,771 TIF rasters against 7,781 LAS files, a ratio of roughly 17.6 to 1, and the diagnostic deliverable is a coverage map whose pairable sliver justified manufacturing 20,000 synthetic logs. At the pipeline level, the 80/20 split produced a background IoU of 0.94 and a curve-1 IoU of 0.26, and the diagnostic deliverable is the ability to read that spread as partly a data signal rather than purely a model one. Three registers, three deliverables, one discipline.

What the numbers share is that none of them is recoverable after the fact. You cannot reconstruct a column's true missingness mechanism from a trained model; you cannot widen a label-coverage ceiling by training harder; and you cannot tell, from the IoU alone, how much of the 0.26 was the corpus and how much was the network. Every one of these is a question that can only be answered by the diagnostic pass, and only before training, which is precisely why the pass is not optional.

VII

THE METHOD

How we run the diagnostic pass

Methods deep-dive: profile, visualise, decide

The pass has three steps and they are always in this order, because each depends on the one before it.

The first step is to profile the nullity column by column. We compute, for every column, the count of present and missing values across all wells, and we do it only after normalising the many ways absence is written in legacy log files to a single machine-readable null, because a profile computed before that normalisation lies twice: it counts text sentinels as present and counts real readings that happen to equal a numeric sentinel as absent. The output is a per-column missing share, and it is the input to everything that follows [1].

The second step is to visualise where the gaps cluster, because the count alone cannot tell you the mechanism. A nullity matrix, columns across, wells down, present cells inked and missing cells blank, shows at a glance whether the gaps are scattered or aligned. Scattered gaps suggest something close to missing-at-random; gaps that line up across a block of wells suggest a structural cause, a tool that was never on the string for those wells, which is a different problem with a different remedy [3]. Co-missing columns, ones that are blank together, reveal features that came from a single tool and rise and fall as one. The visualisation is not decoration; it is the step that converts a count into a hypothesis about the mechanism, and the mechanism is what the law of missing data says you must know before you act [3][4].

The third step is to decide, in the open, with the reason recorded. For each column: keep it, drop it, or fill it, and if fill, with which method and on what justification that the mechanism makes valid [4][5]. For the corpus: map the label coverage and decide whether the supervisable fraction is sufficient, and if not, whether to label, synthesise, or reframe. For the split: set the train/validation partition knowing what each fold contains, so that the hard

class is not accidentally concentrated in one fold. Only when those decisions are made and logged does the first epoch run. The companion exhibits in this document are, in effect, the three steps made operable: a column profiler, a coverage map, and a propagation panel.

> ### WHY THE ORDER IS FIXED
>
> Profiling without visualisation gives you counts but not mechanisms. Visualisation without a recorded decision gives you insight that evaporates by the next sprint. Deciding without first profiling and visualising is guessing. The three steps are cheap individually and only valuable in sequence, which is why we never reorder them and never drop one to save time.

VIII

THE HORIZON

What the discipline buys the next engagement

What the discipline returns, and where we take it next

Installed as standing practice, the diagnostic pass changes the shape of a subsurface ML project in three concrete ways. It moves the most expensive failures forward in time, from the middle of a stalled training run to the cheap half-hour before training, where a column that should have been dropped or a coverage ceiling that should have been mapped is caught when it costs an afternoon rather than a month. It makes the model's eventual numbers interpretable, because a low metric can be traced back to a logged data decision rather than guessed at. And it produces an artefact, the profile, the map, the recorded decisions, that the next engagement inherits, so that the second project on an archive starts from the first project's understanding of it.

The roadmap from here is to make the pass harder to skip than to run. We are folding the three steps into the project's continuous-integration checks, so that a corpus arriving without a committed nullity profile and a label-coverage map fails the gate before any training job is scheduled, the same way untested code fails a build. We are extending the propagation panel into a standing sensitivity tool, so that the question of how much of a metric is corpus and how much is model can be asked routinely rather than reconstructed after a surprise. And we are treating the synthetic-corpus decision, the one the coverage map forced on us, as a first-class, documented response to a label deficit rather than an improvisation, so that the next archive with a 17.6-to-1 ratio meets a known playbook instead of a scramble.

WHAT TO CARRY INTO THE NEXT PROJECT

1. The most consequential modelling decisions are taken **before any model exists**, in a diagnostic pass that profiles, visualises, and decides on missingness. Skipping it is the most common and most expensive mistake in subsurface ML.
2. A well-log table is missing data in three registers at once: **column** nullity (22 columns on the 118-well slice), **corpus** label coverage (136,771 TIF against 7,781 LAS, ~17.6:1), and **pipeline** propagation (the 80/20 split into a curve-1 IoU of 0.26 against 0.94 background).
3. Profile every column individually, not the dataset average; a drop threshold is a logged modelling decision, and an imputed column is only valid if the missingness mechanism allows the fill method.
4. Label coverage can be the missing data: the ~17.6:1 raster-to-vector deficit capped the supervisable fraction at a few percent and is the documented reason a **20,000-log synthetic corpus** was manufactured rather than hand-labelled.
5. An upstream missingness decision surfaces downstream as the validation metric, where it no longer looks like a data problem. Reading the 0.26 correctly requires having drawn the line from data to metric on purpose, before training.

GLOSSARY

IoU

Intersection-over-Union on a segmentation mask: correctly predicted foreground pixels divided by the union of predicted and actual foreground. Our multiclass model scored 0.94 on the easy background class and 0.26 on the thin curve-1 class, a spread the upstream data discipline directly shapes.

LAS

Log ASCII Standard, the canonical text format for digital well-log curves. A LAS file is a model-ready vector log: depth-indexed numeric curves. In a scanned archive, the LAS files are the only material that can supply supervised labels for the raster images.

MCAR / MAR / MNAR

The missingness taxonomy. Missing Completely At Random: the gap is unrelated to anything. Missing At Random: the gap depends on observed variables. Missing Not At Random: the gap depends on the missing value itself. The mechanism dictates which remedies are valid, which is why you diagnose before you decide.

missingno

A Python library for visualising missing data as matrices, bar charts, heatmaps, and dendrograms. It is the standard tool for making the nullity of a tabular dataset legible at a glance before any model is fit. McDonald's tutorial applies it to the 118-well Norwegian-Sea slice.

Nullity

The pattern of present versus absent values in a dataset. A nullity profile is the per-column count of how many cells carry a real measurement and how many are empty. It is the first thing a missing-data diagnostic produces and the thing every later decision rests on.

TIF	A scanned raster image of a paper well log. Pixels, not numbers. A TIF carries the curve a petrophysicist drew decades ago but no machine-readable values, so on its own it cannot be a supervised training example.
Train/validation split	The partition of a corpus into the fold a model learns from and the held-out fold it is scored on. Our multiclass training used an 80/20 split. What each fold contains is set entirely by the upstream missingness and coverage decisions, which is why those decisions show up in the validation metric.

The shortest defence of all this is that the data has the first and the last word, and the only choice a team gets is whether to hear the first word before the model speaks or to wait for the last one after it has failed. We choose to listen early, and every figure in this document is what early listening sounded like on one real archive.

References

1 McDonald, A. (2021). *Using the missingno Python library to identify and visualise missing data prior to machine learning*. Towards Data Science. https://towardsdatascience.com/using-the-missingno-python-library-to-identify-and-visualise-missing-data-prior-to-machine-learning-34c8c5b5f009

2 Bormann, P., Aursand, P., Dilib, F., Manral, S., Dischington, P. (2020). *FORCE 2020 Well log and lithofacies dataset for machine learning competition*. FORCE / Zenodo record 4351156. https://zenodo.org/records/4351156

3 Little, R. J. A., Rubin, D. B. (2019). *Statistical Analysis with Missing Data, 3rd ed.* Wiley. https://onlinelibrary.wiley.com/doi/book/10.1002/9781119482260

4 van Buuren, S. (2018). *Flexible Imputation of Missing Data, 2nd ed.* CRC Press. https://stefvanbuuren.name/fimd/

5 Emmanuel, T., Maupong, T., Mpoeleng, D., Semong, T., Mphago, B., Tabona, O. (2021). *A survey on missing data in machine learning*. Journal of Big Data 8(140). https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00516-9

6 Koroteev, D., Tekic, Z. (2021). *Artificial intelligence in oil and gas upstream: Trends, challenges, and scenarios for the future*. Energy and AI. https://www.sciencedirect.com/science/article/pii/S2666546820300033

Get the full whitepaper

This page is the long-form summary. The complete whitepaper adds the full per-column nullity profile of the 118-well slice, the worked normalisation that has to precede an honest count, the label-coverage arithmetic that sized the 20,000-log synthetic corpus, and the propagation analysis behind the 0.26 curve-1 IoU under the 80/20 split.

Missing-Data Diagnostics Before Petrophysical Machine Learning

Published August 2023. Generated 2026-07-22.
https://www.earthscan.io/whitepapers/missing-data-diagnostics-before-petrophysical-ml