Risk-Sensitive Loss Functions for Distribution-Free, Tail-Risk-Aware Neural Network Training

1 Context

Following the Snell paper, I am looking for loss functions that are:

- **Distribution-free:** They make minimal assumptions on the data distribution, focusing on empirical or worst-case characteristics rather than a specific parametric form (like normality).
- **Tail-risk-aware:** They emphasize the worst-case or tail outcomes (the largest losses or errors), ensuring the model is penalized for those rare large errors.
- **Differentiable (or at least amenable to gradient-based optimization):** So that we can train neural networks with them using gradient descent or similar methods.

We could try adapt two pretty simple ones and then three more complex ones (derived below):

1. Value-at-Risk (VaR)
2. Conditional Value-at-Risk (CVaR), also known as Expected Shortfall (ES)
3. A **distributionally robust CVaR** objective (modeling an adversarial worst-case distribution within some ambiguity set, leading to an *entropic* risk measure)
4. **Entropic Value-at-Risk (EVaR)**, a smooth tail-risk measure derived from exponential moments
5. An **Extreme-Value-Theory-based Expected Shortfall (EVT-ES)** approach that leverages statistical extreme value theory to account for ultra-rare events beyond the observed data.

Current 5 candidates

1. **VaR loss (level α)**

$$L_{\mathrm{VaR},\alpha}(\theta,t)=\frac{1}{n}\sum_{i=1}^{n}\rho_\alpha\big(\ell_i(\theta)-t\big),\quad \rho_\alpha(z)=\begin{cases}\alpha\, z & (z\ge 0),\\ (\alpha-1)\, z & (z<0)\end{cases}$$

2. **CVaR loss (level α)**

$$L_{\mathrm{CVaR},\alpha}(\theta,t)=t+\frac{1}{(1-\alpha)n}\sum_{i=1}^{n}\big[\ell_i(\theta)-t\big]_+$$

3. **KL-robust CVaR (robustness η)**

$$L_{\text{DR-CVaR},\eta}(\theta)=\inf_{s>0}\Big\{\frac{1}{s}\ln\Big[\frac{1}{n}\sum_{i=1}^{n}e^{s\ell_i(\theta)}\Big]+\frac{\eta}{s}\Big\}$$

4. **EntropicVaR (EVaR, level α; set $\eta=-\ln(1-\alpha)$)**

$$L_{\mathrm{EVaR},\alpha}(\theta)=\inf_{s>0}\frac{1}{s}\ln\Big[\frac{1}{n(1-\alpha)}\sum_{i=1}^{n}e^{s\ell_i(\theta)}\Big]$$

5. **EVT-based Expected Shortfall (threshold u)**

$${}^{\mathrm{EVT}}_{\alpha}=u+\frac{\hat\beta}{\hat\xi}\Big[\big(\tfrac{1-\alpha}{\hat p_u}\big)^{-\hat\xi}-1\Big]$$

$$L_{\text{EVT-ES},\alpha}(\theta)=\frac{{}^{\mathrm{EVT}}_{\alpha}}{1-\hat\xi}+\frac{\hat\beta-\hat\xi\, u}{1-\hat\xi}\qquad(\hat\xi<1)$$

Distributionally Robust CVaR Loss (KL-Divergence Robustness)

My idea of distributional robustness is to prepare the model not just for the distribution of losses seen in the training data, but for a whole set of possible distributions that could occur. In other words, we don't trust the training distribution completely—there might be uncertainty in it or the true distribution might differ (say the future test data might have heavier tails or slightly shifted probabilities). We therefore consider an adversary ("Nature") that can perturb the distribution of outcomes within some allowed bounds, and we optimize for the worst-case loss under those perturbations. This yields a loss function that effectively says: "Minimize the worst-case CVaR (or mean loss) among all distributions close to the empirical distribution."

The "bounds" on distribution change are usually specified by a divergence or distance in probability space. Instead of Wasserstein (no analytical solution), I went for a KL-divergence ball: we allow Nature to pick any distribution Q that is within a certain KL divergence δ from the empirical distribution P (given by the training data). The model then minimizes the worst-case CVaR under any such Q. Intuitively, this means we are accounting for model mis-specification or sampling error: Nature could reweight the data in an adversarial way, especially putting more weight on cases where our model performs badly, up to a limit (the KL-bound).

Derivation

Let $\ell(\theta; x)$ be the loss of model θ on a data sample or scenario x. The empirical distribution P places mass $\frac{1}{n}$ on each observed sample x_i (for $i = 1, \ldots, n$). We consider a set of plausible distributions

$$\mathcal{U} = \{Q \ll P \mid D_{\mathrm{KL}}(Q \,\|\, P) \leq \eta\},$$

where $D_{\mathrm{KL}}(Q||P) = \sum_{i=1}^{n} q_i \ln(q_i p_i^{-1})$ with $p_i = 1/n$ and $q_i = Q(x_i)$ (since any $Q \ll P$ means Q can only allocate mass to the support of P). Here η is a divergence budget that controls how far Q can stray from P in terms of relative entropy.

We then define the robust loss for model θ as the worst-case expected loss under $Q \in \mathcal{U}$:

$$L_{\text{robust}}(\theta) = \sup_{Q \in \mathcal{U}} \mathbb{E}_{x \sim Q}[\ell(\theta; x)].$$

Consider:

$$\sup_{Q : D_{KL}(Q||P) \leq \eta} \sum_{i=1}^{n} q_i \ell_i,$$

where $\ell_i = \ell(\theta; x_i)$ are the losses on each sample. This is a finite-dimensional convex optimization in q_i (the adversary's problem, given θ fixed for the moment). The Lagrange dual of the KL constraint problem is derived using the fact that:

$$D_{\mathrm{KL}}(Q||P) \leq \eta \iff \sum_i q_i \ln(q_i n) \leq \eta,$$

with $p_i = 1/n$. The dual (from the perspective of the adversary maximizing) can be derived via the moment-generating function (MGF). In fact, a known result (Donsker-Varadhan variational formula) states:

$$\sup_{Q : D_{\mathrm{KL}}(Q||P) \leq \eta} \mathbb{E}_Q[\ell] = \inf_{t > 0} \left\{ \frac{1}{t} \ln \mathbb{E}_P[e^{t\ell}] + \frac{\eta}{t} \right\}.$$

This result comes from the convex conjugate of the KL divergence. In particular:

1. The unconstrained supremum (without KL limit) of $tE_Q[\ell] - D_{\mathrm{KL}}(Q||P)$ over Q yields $\ln E_P[e^{t\ell}]$ (this is a known variational representation of KL: $\sup_Q\{t \sum q_i \ell_i - \sum q_i \ln(q_i n)\} = \ln \frac{1}{n} \sum \exp(t\ell_i)$).

2. Imposing $D_{\mathrm{KL}} \leq \eta$ is like a Lagrange multiplier: for a given $t > 0$, if t is the multiplier for the KL constraint, one gets $tE_Q[\ell] \leq \ln E_P[e^{t\ell}] + t\eta$.

   Rearranged: $E_Q[\ell] \leq \frac{1}{t}[\ln E_P(e^{t\ell}) + \eta]$ for all such Q. The worst-case expectation equals this bound when the inequality is tight.

3. So the smallest upper bound on $E_Q[\ell]$ is found by minimizing over $t > 0$: $\inf_{t>0} \frac{1}{t}(\ln E_P[e^{t\ell}] + \eta)$.

Coming back, we have:

$$L_{\text{robust}}(\theta) = \inf_{t > 0} \left\{ \frac{1}{t} \ln \left(\frac{1}{n} \sum_{i=1}^{n} e^{t\ell_i(\theta)} \right) + \frac{\eta}{t} \right\}.$$

Connecting this to CVaR for some particular α, we have to find some reasonable value for the divergence budget (η).

I was thinking that we could do $\eta = -\ln\alpha$ because if $\eta = -\ln\alpha$, the condition $D_{\text{KL}}(Q||P) \leq -\ln\alpha$ roughly means the adversary has enough budget to concentrate nearly all probability on a subset of data.

In fact, if α is like 0.95, $-\ln\alpha \approx 0.0513$. That budget allows a moderate reweight. The theorem (as found in literature) actually states:

$$\sup_{Q:D_{KL}(Q||P)\leq -\ln\alpha} E_Q[\ell] =_\alpha (P,\ell),$$

the entropic VaR at confidence α. We will discuss EVaR shortly; essentially this expression equals:

$$\inf_{t>0} \frac{1}{t}\Big(\ln \sum_i \frac{1}{n} e^{t\ell_i} - \ln\alpha \Big).$$

Setting $\eta = -\ln\alpha$ yields exactly that form. And indeed, that is recognized as the definition of EVaR (Entropic VaR), which is a specific coherent risk measure dominating CVaR.

However, let's not jump ahead too far. For now, our derivation has given us a new loss function:

$$L_{\text{robust}}(\theta) = \inf_{t>0} \Big\{ \frac{1}{t} \ln \frac{1}{n} \sum_{i=1}^{n} e^{t\ell_i(\theta)} + \frac{\eta}{t} \Big\}.$$

This loss function is implicitly differentiable (it's a pointwise infimum of smooth functions; one could differentiate under the infimum by selecting the minimizing t). In practice, one can treat t as an additional parameter to optimize along with θ.

The interpretation is: t controls how "aggressive" the adversary weighting is. If t is large, the term $\frac{1}{n}\sum e^{t\ell_i}$ heavily emphasizes the largest ℓ_i (since $e^{t\ell}$ amplifies large ℓ exponentially), effectively approximating a worst-case. If t is small, it's closer to the average loss.

The $\frac{\eta}{t}$ term penalizes large t though, preventing t from going to infinity (which would focus only on the single largest loss). Specifically, if $\eta = -\ln\alpha$, then $\frac{\eta}{t}$ penalizes large t. The optimum t^* will balance focusing on tail vs the penalty. It can be shown that $t^* = \arg\inf$ corresponds to a situation where the KL constraint is exactly met ($D_{KL}(Q^*||P) = \eta$) and the worst-case Q^* is obtained by

$$q_i^* \propto p_i \exp(t^*\ell_i),$$

i.e. an exponential tilt of the empirical distribution.

To connect to robust CVaR: if we specifically want the worst-case CVaR at level α, we would incorporate $(1-\alpha)$ fraction explicitly. Another approach: CVaR itself can be written as an optimization:

$$_\alpha(Z) = \sup_{\substack{Q:Q\ll P \\ Q(A)\leq \frac{1}{1-\alpha}P(A)\ \forall \text{ measurable } A}} E_Q[Z].$$

This is a representation of CVaR as a worst-case mean over distributions Q that are "dominated" by P in a certain way (no subset of outcomes can get more than $1/(1-\alpha)$ times its original probability — effectively an ∞-norm bound on likelihood ratios). If one relaxes that to a softer KL bound, we get EVaR. So indeed: - Robust CVaR with a KL ambiguity yields EVaR, which is an entropic version of CVaR.

Thus, the derivation shows that a distributionally robust approach transforms the loss function into an **entropic loss**: an exponential averaging of losses, with a parameter controlling tail emphasis.

Properties:

1. Conservatism Tuning (η or α): The level of robustness η (or equivalently the confidence level $\alpha = e^{-\eta}$ if using that substitution) determines how pessimistic the loss is. $\eta = 0$ means no distribution shift allowed, so $Q = P$ and L_{robust} reduces to the standard average loss. As η increases, the worst-case distribution can deviate more, leading to a larger loss value (the model sees more worst-case). In the limit $\eta \to \infty$, the adversary could concentrate all probability on the absolute worst sample, so L_{robust} would approach the maximum loss (worst-case scenario planning).

2. Smoothness: The robust loss is typically smooth (differentiable) w.r.t. θ, because it involves $\ln \sum \exp(t\ell_i)$, which is a smooth approximation of the max. This is a big advantage for gradient-based training.

3. Convexity in loss space: $L_{\text{robust}}(\theta)$ as a function of the loss values $\{\ell_i\}$ is convex (since $\ln \sum \exp$ is convex in the vector (ℓ_i)). If $\ell(\theta; x)$ is convex in θ, then the robust objective remains convex in θ (plus the additional $1/t$ factor stuff which does not harm convexity in θ because t is separate).

4. Distributional uncertainty handling: This approach explicitly accounts for the fact that the empirical distribution might have errors. It provides probabilistic guarantees: one can show that with high confidence, the true expected loss is bounded by $L_{\text{robust}}(\theta)$ if η is chosen according to sample size (this ties into concentration inequalities and such).

So I think the biggest issues with this is if the training distribution is actually exactly representative and not heavy-tailed, an overly robust training might underfit (for example, if truly everything is Gaussian and mild, focusing on imaginary heavier tails could hurt fit). Also, if tail events are so rare that even an adversary within a reasonable η can't simulate them (like if none or one sample exists of a kind), then robust training has limited effect beyond perhaps upweighting that one sample. In extremely heavy-tailed real distributions where the mean is infinite, one could set η but then $\ln E[e^{t\ell}]$ might keep growing with t without bound, implying infinite robust risk — which signals that under such conditions, no finite robustness can control the risk (similar to EVT's message: if tail index ≥ 1, you can't have finite expectation of tail).

Entropic Value-at-Risk (EVaR) Loss Function

Intuitively, EVaR says: “Consider using an exponential weighting for losses; what loss level can we guarantee at confidence α using Markov’s inequality (Chernoff bound)?”

In narrative terms, EVaR provides an upper bound on CVaR. It’s the worst-case CVaR among all distributions in a certain class, which makes it always at least as large as CVaR for the same α. So it is more conservative (hence safer) but not overly so: it is the *tightest possible upper bound* that is still coherent. When used as a loss function, EVaR will prioritize tail events even more than CVaR does, but in a differentiable way. It’s like applying a “soft-max” to the losses to focus on the worst ones.

Mathematical Derivation:

The formal definition of ${}_\alpha(Z)$ for $\alpha = 1 - p$ (so p is the tail probability, e.g. $p = 0.05$ for $\alpha = 0.95$) is:

$${}_\alpha(Z) = \inf_{t>0} \frac{1}{t} \ln\left(\frac{1}{\alpha}\mathbb{E}[e^{tZ}]\right).$$

Note: some conventions define α as the tail probability, but here we stick to α as the confidence level, so $(1-\alpha)$ is the tail. So we could also write ${}_\alpha(Z) = \inf_{t>0} \frac{1}{t} \ln\left(\frac{E[e^{tZ}]}{1-\alpha}\right)$ with slight reparameterization. For clarity: - If $\alpha = 0.95$, $1-\alpha = 0.05$. Then $1/(1-\alpha) = 20$. So $\mathrm{EVaR}_{0.95}(Z) = \inf_{t>0} \frac{1}{t} \ln(20 \cdot E[e^{tZ}])$.

We derive this from Chernoff/Markov inequality: For any $t > 0$, $\Pr(Z \geq c) \leq \frac{E[e^{tZ}]}{e^{tc}}$. Set $\Pr(Z \geq c) = 1 - \alpha$. We want the smallest c such that $\Pr(Z \geq c) \leq 1 - \alpha$. Using the bound:

$$1 - \alpha \leq \frac{E[e^{tZ}]}{e^{tc}} \iff c \leq \frac{1}{t} \ln\left(\frac{E[e^{tZ}]}{1-\alpha}\right).$$

This holds for each $t > 0$. To get the tightest (smallest) such bound on c, we minimize the right-hand side over t. That minimum value is exactly ${}_\alpha(Z)$. At the optimal t, the inequality via Chernoff becomes an equality bound in a sense of optimization (though not in actual probability, in the worst-case distribution scenario it matches).

Thus EVaR is derived as a variational bound on VaR/CVaR using exponential moments.

Now, we already encountered EVaR in the previous section: indeed ${}_\alpha(Z) = \sup_{Q:D_{KL}(Q||P)\leq -\ln\alpha} E_Q[Z]$. But here, we see it also as:

$${}_\alpha(Z) = \inf_{t>0} \frac{1}{t} \left(\ln E[e^{tZ}] - \ln\alpha\right).$$

If we have samples, $E[e^{tZ}]$ is approximated by $\frac{1}{n}\sum_i e^{tz_i}$. In training, one can implement EVaR by computing that quantity. Notice it is essentially the same as the robust formula with $\eta = -\ln\alpha$:

$${}_\alpha(Z) = \inf_{t>0} \left\{ \frac{1}{t} \ln \frac{1}{n} \sum_{i=1}^{n} e^{tz_i} + \frac{-\ln\alpha}{t} \right\}.$$

This is identical to earlier robust expression. The minimizing t^* in EVaR formula can be interpreted as the “risk aversion” parameter that one would put in an exponential utility or an entropic risk measure. Indeed, entropic risk measure often is defined as $\rho(X) = \frac{1}{\lambda} \ln E[e^{\lambda X}]$ (without the $-\ln\alpha$ part). EVaR adds that $\frac{1}{\lambda}\ln(1/(1-\alpha))$ piece and then optimizes λ. In some sense, t^* will be large for distributions with heavy tails (the optimizer tries to focus on tail by raising t until the $\frac{\eta}{t}$ penalty stops it).

From a training perspective, we have the formula; we can differentiate through it by either optimizing t as a free parameter or by computing gradient of the inner function at t^*.

Could approximate $\inf_{t>0}$ by picking an appropriate t. But since the infimum is a smooth convex function in t, one can simply have t learn via gradient (the combined objective is joint in θ and t and can be optimized).

Properties:

1. Coherent and Law-invariant:** EVaR satisfies coherence axioms and depends only on the distribution of Z. It's actually the **most conservative coherent risk measure under the same information that is still an upper bound of CVaR** (for a given α).
2. For very light-tailed distributions (almost Gaussian), EVaR and CVaR might be close. For heavy-tailed but finite exponential moment distributions, EVaR might be significantly larger than CVaR because exponential amplifies tail. However, EVaR has a key limitation: it requires $E[e^{tZ}] < \infty$ for all t in some interval $(0, t_0]$. If the distribution of Z has heavy tails (like a power-law), then as we noted, $E[e^{tZ}]$ might diverge for any $t > 0$. If for example Z is such that $\Pr(Z > z) \sim z^{-\gamma}$ for large z, then $E[e^{tZ}] = \infty$ for all $t > 0$ (since the tail doesn't have an exponential cutoff). In those cases, EVaR is effectively infinite (meaning you cannot bound that tail with any finite number at the given α because the exponential moment fails).
3. Should be easier computationally- EVaR's inner quantity $\ln \sum e^{tz_i}$ is like a LogSumExp, which is common and stable to compute (often implemented with shifting to avoid overflow). So computing EVaR for a batch is quite efficient. One just has to also perhaps do a line search or gradient step on t. In practice, one could treat t as a hyperparameter: for example, pick t such that the effective $\alpha = e^{-\eta} = e^{-tx}$ for some reference x. Alternatively, some cross-validation could find a good t.
4. but could be super intense computationally for certain distributions → for moderate tail distributions, EVaR might be only slightly larger than CVaR (so a mild safety margin). For very skewed distributions, EVaR can be significantly larger, thus more conservative. EVaR is arguably safer if you suspect that even CVaR might underestimate risk due to not capturing extremely rare possibilities.

Evaluation

Outline of main ideas: easier to plug into gradient training, penalizes outliers more, better theoretical properties

EVaR is arguably easier to plug into gradient-based training than CVaR, because you do not need to sort or select top losses explicitly; it uses a smooth exponential weighing of all losses. It's essentially a smooth approx to the maximum (with a bias). That means gradient flows through every example, but high-loss examples get larger gradient weights $\propto e^{t\ell}$.

Compared to CVaR which averages a fixed fraction of worst cases, EVaR (especially with large t at optimum) effectively puts even more weight on the very worst cases within that fraction. In fact, if one sample is extremely larger loss than the rest, an optimized t tends to focus on that one sample (bounded by the $\ln(1/\alpha)$ term to not go infinite). So EVaR training will really try to eliminate any extreme outlier by heavily penalizing it. This is beneficial if that outlier is a plausible scenario we want to guard against; it could be problematic if that outlier is a noise or a one-off error in data (the model might over-correct to fit one point).

EVaR's coherence and being an upper bound means that if one cares about mathematical guarantees, using EVaR ensures no nasty surprises like violation of subadditivity. It can be used to derive tail bounds etc. In a static training scenario, this is less of an issue, but if the professor cares about risk measure theory, EVaR is an elegant object.

As noted, if data suggest an extremely heavy tail (e.g., some samples have such high loss that any $E[e^{tZ}]$ appears huge), EVaR might push t to a very high value, effectively blowing up the loss!!! In an extreme case where heavy-tail implies infinite EVaR, the training objective is effectively saying "your risk is infinite" which means no model can bound it. For instance, if the model has unbounded loss possibilities (like not clipped), EVaR might not be finite. This signals the need for either constraining the model or acknowledging that the risk cannot be controlled.

Extreme-Value-Theory-Based Expected Shortfall (EVT-ES)

So all our loss functions (VaR, CVaR, robust CVaR, EVaR) use the empirical distribution of losses directly (or some transformation of it) to assess tail risk. They are distribution-free in the sense of not assuming a parametric distribution like normal, but they do assume that the tail risk can be assessed from observed data.

But I feel like with small datasets or extremely high confidence levels (like 99.9% or 99.99%), we may have very few observations in the tail (think Liberation day, 3 8-10 sigma days in a row). Even with bigger data, the far tail can be noisy. I found that Extreme Value Theory (EVT) could help because it provides a framework to model the tail of distributions using asymptotic theory, typically the Generalized Pareto Distribution (GPD) for exceedances over a high threshold.

The EVT-based ES approach is basically: "Fit a model to the tail of the loss distribution, so that you can extrapolate beyond the observed range and estimate extremely high quantiles or tail expectations." It is not a single closed-form risk measure like CVaR or EVaR, but rather a method to improve the estimation of CVaR (and VaR) at high levels by using statistical tail modeling. In the context of training, one might incorporate EVT by:

1. First, choosing a high threshold u (for example, the 95th percentile of losses observed so far).
2. Assume that losses beyond u follow a GPD with some parameters (ξ, β) (xi is shape parameter controlling tail heaviness, β is scale).
3. Estimate those parameters from the data exceeding u.
4. Use the fitted tail distribution to compute CVaR at the desired level α (even if α is very high, we can extrapolate). - Use that as part of the loss objective or constraint for training.

Nevertheless, as a "loss function" approach, one could imagine adjusting the loss such that the worst few losses are modeled with a GPD and ensuring the model's parameters minimize the estimated extreme CVaR. Narratively, EVT-ES is about acknowledging that "we may not have seen the worst-case in our data, so let's use extreme value theory to predict what unseen worst-cases could be and train against those as well."

Mathematical Derivation For any loss variable X and high threshold u, the Pickands–Balkema–de Haan theorem states that the excess $Y = X - u \,|\, X > u$ converges to a *Generalised Pareto Distribution* $\mathrm{GPD}(\xi, \beta)$,

$$F_Y(y) = 1 - \left(1 + \tfrac{\xi y}{\beta}\right)^{-1/\xi}, \quad y \geq 0$$

(with exponential tail when $\xi = 0$).

1. **Fit the tail.** Let k of n losses exceed u; set $\hat{p}_u = k/n \approx \Pr(X > u)$ and estimate $(\hat{\xi}, \hat{\beta})$ from those k points (e.g. maximum likelihood).
2. **Extrapolate VaR.** For confidence α such that $1 - \alpha < \hat{p}_u$ (i.e. the α-quantile lies *above* u),

   $$\mathrm{VaR}_\alpha(X) \;\approx\; u + \frac{\hat{\beta}}{\hat{\xi}}\Big[\big(\tfrac{1-\alpha}{\hat{p}_u}\big)^{-\hat{\xi}} - 1\Big], \quad (\hat{\xi} \neq 0),$$

   or $\mathrm{VaR}_\alpha \approx u + \hat{\beta}\ln\!\big(\tfrac{1-\alpha}{\hat{p}_u}\big)$ when $\hat{\xi} = 0$.
3. **Extrapolate ES (CVaR).** Provided $\hat{\xi} < 1$ (finite mean in the tail),

   $$ES_\alpha(X) \;\approx\; \frac{{}_{\alpha}(X)}{1-\hat{\xi}} + \frac{\hat{\beta} - \hat{\xi}\,u}{1-\hat{\xi}}.$$

Key points.

- ξ is the *tail index*: $\xi > 0$ implies heavy Pareto tails, $\xi < 0$ implies a finite upper bound.
- Setting a threshold too low *biases* (GPD not yet valid); too high *inflates variance* (few exceedances).
- If $\hat{\xi} \geq 1$, the tail mean diverges and $ES_\alpha = \infty$—a red-flag for unmanageable risk.

Assume a distribution 97% N(0,1) + 3% N(8,1) (so a mixture with a heavy-ish tail due to the 3% "crash" scenario). Empirical 99.9% CVaR was not computable (no sample that extreme in one dataset). EVT fit (with threshold at 95th percentile) gave $\hat{\xi} = 0.22$ (some moderate tail) and predicted ${}_{99.9\%} \approx 5.92, ES_{99.9\%} \approx 7.55$. The actual simulation might confirm those are reasonable. So the EVT method gave numbers where empirical method gave none.

For a neural network, we could simulate many runs, fit EVT to the worst errors, and thus estimate extremely rare failure probabilities and severities.