

Introducing the CZAR Loss: A Tailored Objective Function for Financial Log-Return Predictions

Joel Pfeffer¹, J. M. Diederik Kruijssen¹, Florian Stecker¹ & Steven N. Longmore^{1,2}

¹Allora Foundation, ²Liverpool John Moores University

Abstract

In quantitative finance, standard regression losses are misaligned with the economics of return prediction. As the conditional mean of financial log-returns is close to zero, symmetric losses such as the mean squared and mean absolute errors make the constant zero forecast a near-optimal solution, penalizing models with genuine but noisy directional skill. This applies both during training, where predictions shrink toward zero, and during evaluation, where trivial forecasters can lead loss-based rankings. Under a Gaussian linear prediction model, we show that *all* symmetric monotonic losses share a universal breakeven directional accuracy against the zero predictor, which rises sharply and becomes unobtainable as the prediction noise approaches the standard deviation of the returns. We introduce the CZAR (Composite Zero-Agnostic Return) loss function, a piecewise quadratic loss built around five requirements derived from this analysis: convexity in the prediction, asymmetry oriented by the direction of the true return that vanishes at zero, near-linear penalization of undershoots and wrong-direction predictions, divergence for large errors, and an adaptive loss floor for evaluation. CZAR is provably convex in the prediction at fixed true value, has closed-form gradient and Hessian suitable for custom objectives in gradient-boosted libraries, and its four hyperparameters reduce to a single choice through correlated defaults. In idealized tests, the minimum directional accuracy required for a CZAR-evaluated forecaster to outperform the zero predictor under mean log loss remains near the 50% chance level, whereas the corresponding threshold for symmetric losses rises sharply with prediction noise. This advantage persists under heavy-tailed return distributions. In a LightGBM experiment on intraday (15-minute and 1-hour) BTC log-returns, CZAR-trained models reduce the ‘zero-returns bias’ of the L1 and L2 baselines and improve long–short performance and directional accuracy on large-magnitude returns.

1 Introduction

Predicting financial log-returns is an unusually hostile regression problem. The predictable component of returns is small relative to the noise, return distributions are heavy-tailed (Blattberg & Gonedes, 1974; Gopikrishnan et al., 1999; Cont, 2001), and at most horizons the conditional mean return is close to zero. Machine-learning models, most prominently gradient-boosted decision trees (Chen & Guestrin, 2016; Ke et al., 2017) and deep networks (Gu et al., 2020), are nevertheless routinely trained on return targets using generic regression losses such as the mean squared error (MSE) or mean absolute error (MAE). These point-error objectives are the default in general forecasting practice (Makridakis et al., 2020, 2022), yet they are designed for settings where the target carries substantial extractable signal. The choice of loss function matters twice over. During training, its gradient and curvature determine what the model learns; during evaluation, sample-averaged losses drive hyperparameter tuning, early stopping, and model selection. A loss function misaligned with the actual prediction objective therefore corrupts both the models that are trained and the way they are compared.

For predicting log-returns, symmetric losses are misaligned in a specific and severe way, which we refer to as a *zero-returns bias*. Symmetric losses are minimized in expectation by central tendencies of the conditional return distribution (the conditional mean for MSE, the conditional median for MAE; Gneiting 2011), and for log-returns these are close to zero. A constant zero prediction therefore achieves a near-minimal expected loss, while a model with genuine directional skill but realistic prediction noise is penalized for the noise more than it is rewarded for the signal. As we show in §2, under a Gaussian linear prediction model a forecaster with directional accuracy well above chance can incur a *higher* MAE and MSE than the trivial zero predictor, and the directional accuracy required to break even against zero returns follows a universal curve shared by *all* symmetric monotonic losses, rising sharply as the prediction noise approaches the scale of the returns themselves. This problem is not confined to MSE and MAE. The conditional expectation is the optimal predictor for every loss in the Bregman family (Banerjee et al., 2005), so any such loss makes near-zero prediction an attractor whenever the conditional mean return is near zero. Practically, the predictable component of returns recovered by machine-learning models is small, with out-of-sample R^2 values of at most a few percent (e.g. Gu et al., 2020; Leippold et al., 2022). A model trained with a symmetric loss responds by shrinking its predictions toward zero, a degenerate solution that minimizes training loss while destroying the forecast as a directional signal. However, directional information is precisely what is economically valuable. Forecast profitability is tied to directional accuracy rather than to conventional error measures (Leitch & Tanner, 1991; Granger & Pesaran, 2000), directional accuracy is a standard evaluation criterion with well-developed tests (Pesaran & Timmermann, 1992), and even statistically weak return predictability can be economically significant (Kandel & Stambaugh, 1996).

Asymmetric loss functions are the natural remedy, and they have a long history. Granger (1969) showed that a non-quadratic cost of error renders the optimal forecast biased away from the conditional mean; Zellner (1986) introduced

the LINEX loss; Christoffersen & Diebold (1997) derived the optimal (time-varying) bias under asymmetric loss and conditional heteroskedasticity; and Elliott et al. (2005) and Patton & Timmermann (2007) developed estimation and testing under flexible families of asymmetric loss. On the training side, the pinball loss of quantile regression (Koenker & Bassett, 1978) and the asymmetric squared loss of expectile regression (Newey & Powell, 1987) are the canonical asymmetric objectives, and the Huber loss (Huber, 1964) the canonical robust one. None of these resolve the zero-returns bias, for two reasons. First, their asymmetry is fixed and oriented by the sign of the prediction error. They uniformly prefer over- (or under-) prediction everywhere, which merely relocates the optimal predictor from the conditional mean to a fixed quantile or expectile of the conditional distribution (Gneiting, 2011). This results in a uniform bias rather than a removal of the attractor, and one that imposes a directional preference even when the true return is zero, where none is justified. What return prediction requires is instead asymmetry oriented by the *direction of the true return*. Overshooting a correctly-signed prediction is nearly harmless, while undershooting or predicting the wrong direction is not, and the asymmetry should vanish as the true return approaches zero and strengthen as its magnitude grows. Second, as we show in §2, a constant asymmetry strong enough to materially lower the breakeven directional accuracy introduces pathologies of its own. Under log-averaged evaluation (ranking by the mean of the logarithm of the losses, equivalently their geometric mean, an established choice for aggregating positive error measures that span orders of magnitude, as it limits domination by large-magnitude samples (Fleming & Wallace, 1986; Armstrong & Collopy, 1992; Hyndman & Koehler, 2006)) it can push the breakeven below 50%, so that a model that is systematically *wrong* about direction outranks the zero predictor.

A complementary strand of work abandons statistical losses altogether in favor of economic objectives. This strand includes trading systems trained by direct maximization of the Sharpe ratio (Moody & Saffell, 2001; Zhang et al., 2020), decision-focused learning that trains the predictor on the downstream optimization objective (Elmachetoub & Grigas, 2022), no-arbitrage moment conditions used as adversarial losses (Chen et al., 2024), and bespoke asymmetric or direction-aware losses for asset-return prediction (Dessain, 2023; Michańkó et al., 2024). This literature confirms that replacing symmetric statistical losses with finance-aware objectives improves trading-relevant performance, but the resulting objectives are typically non-convex, tied to a particular portfolio construction, or lacking the analytical gradient and Hessian required of a custom objective in gradient-boosted libraries (Chen & Guestrin, 2016; Ke et al., 2017). What is missing is a general-purpose loss for return prediction that is convex in the prediction, directly trainable, and constructed explicitly so that neither the zero predictor nor a wrong-direction predictor can outrank a genuinely informative model.

In this paper we introduce the CZAR (Composite Zero-Agnostic Return) loss function, which is designed to fill this gap. Our contributions are as follows. (i) From an analysis of symmetric and asymmetric losses under a Gaussian linear prediction model, we derive five requirements that a loss function adequate for return prediction must satisfy: convexity in the prediction, direction-oriented asymmetry that is symmetric at a zero true return, near-linear penalization of undershoots and wrong-direction predictions, divergence for large errors, and an adaptive loss floor near the mean return (§2). (ii) We construct CZAR, a piecewise quadratic loss with an asymmetry that adapts to the magnitude of the true return and an evaluation-time loss floor, satisfying all five requirements, and reduce its four hyperparameters to effectively one through correlated defaults (§3). (iii) We prove that CZAR is convex in the prediction at fixed true value (§3.3). (iv) We derive the analytical gradient and Hessian, making CZAR directly usable as a training objective in gradient-boosted frameworks (§3.3, with a reference implementation in Appendix A). (v) In idealized tests, we show that CZAR’s log-loss breakeven directional accuracy remains near the 50% chance level as prediction noise increases, whereas the corresponding thresholds for symmetric and constant-asymmetry loss functions rise sharply. This advantage persists under heavy-tailed return distributions, and rankings by CZAR loss track directional skill rather than prediction variance. We then evaluate CZAR in a LightGBM experiment on intraday (15-minute and 1-hour) BTC log-returns, where CZAR-trained models reduce the prediction shrinkage of L1- and L2-trained baselines and improve long–short performance and directional accuracy on large-magnitude returns (§4). §5 discusses implications and limitations.

2 Background

As a starting point for evaluating loss functions for returns predictions, we consider a simple case where the underlying true returns follow a Gaussian distribution with mean at zero and scale of unity, $y = \mathcal{N}(\mu = 0, \sigma = 1)$. Predicted returns then follow the format of Gaussian linear predictions, $\hat{y} = \rho y + \mathcal{N}(0, \sigma_N^2)$, where ρ is the signal coefficient and σ_N is the noise scale, relative to the true distribution (i.e. for a fixed true return y , the prediction is drawn from a second Gaussian, whose additive noise is independent of y). Note that ρ is the regression slope, not the Pearson correlation between $\hat{y}$ and y ; the latter is $\rho / \sqrt{\rho^2 + \sigma_N^2}$. Example distributions are shown in Figure 1 for increasing signal coefficient and noise scales. In the following, we refer to the single case $\rho = 0, \sigma_N = 0$ (so that $\hat{y} \equiv 0$) as the ‘zero-returns’ model and the case $\rho = 0, \sigma_N = 1$ (a pure-noise predictor whose marginal distribution matches the true returns PDF) as the ‘same-PDF’ model. The remaining $\rho = 0, \sigma_N > 0$ panels in Figure 1 are pure-noise predictors, $\hat{y} = \sigma_N \varepsilon$, not zero-returns. In Figure 1, the directional accuracy, signal-to-noise ratio (which directly correlates with directional accuracy in this case) and mean loss values for MAE and MSE loss functions are indicated in each panel. For $\rho = 0$ with $\sigma_N > 0$ (pure noise), directional accuracy is $\approx 50\%$ (within some error due to sampling); for the zero-returns predictor ($\rho = 0, \sigma_N = 0$) every prediction is exactly zero and is reported as a directional miss, giving DA = 0 in the top-left panel. For $\rho > 0$ directional accuracy decreases with increasing σ_N . A key point to note is that both MAE and MSE have lower losses for the zero-returns model

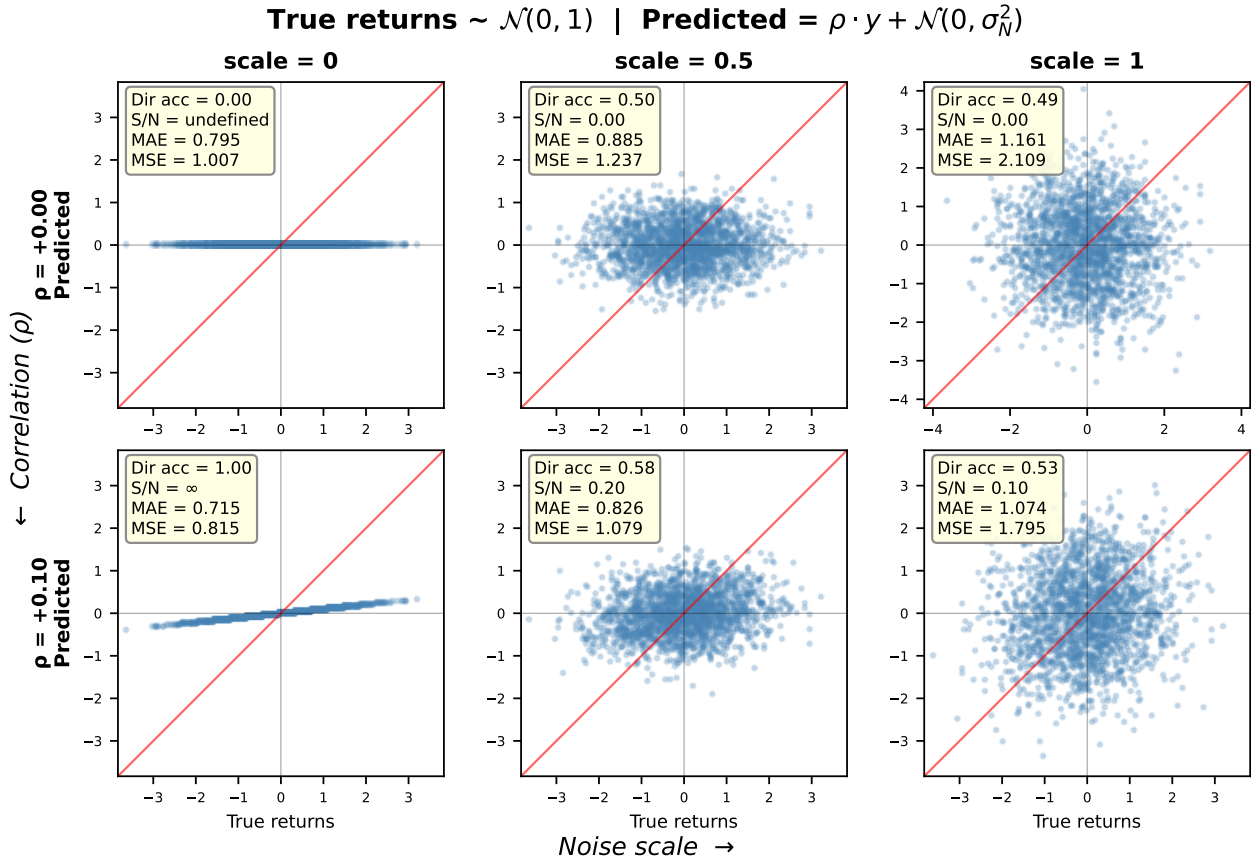


Figure 1: Scatter plots of predicted vs. true returns under the Gaussian linear model $\hat{y} = \rho y + \mathcal{N}(0, \sigma_N^2)$ for signal coefficient $\rho \in \{0, 0.1\}$ (rows) and noise scale $\sigma_N \in \{0, 0.5, 1\}$ (columns). The red diagonal marks perfect prediction. Directional accuracy (DA), signal-to-noise ratio, mean absolute error, and mean squared error are annotated in each panel. The top-left panel ($\rho = 0, \sigma_N = 0$) is the zero-returns predictor with $\hat{y} \equiv 0$, collapsing all predictions to the horizontal axis. The remaining $\rho = 0$ panels are pure-noise predictors $\hat{y} = \sigma_N \epsilon$. Despite zero DA, the zero-returns predictor achieves lower MAE (0.795) and MSE (1.007) than the signal-carrying predictor at $\rho = 0.1, \sigma_N = 0.5$ (MAE 0.826, MSE 1.079), illustrating the zero-returns attractor problem.

than for the signal-carrying model with $\rho = 0.1$ and $\sigma_N = 0.5$; i.e., noise in the predictions is more important than the signal itself. When evaluating the performance of models (e.g. when training and optimizing machine learning models to predict returns), the zero-returns case therefore becomes an attractor solution.

This series of models can be used to calculate a ‘breakeven’ directional accuracy, which occurs when the expectation values are identical, i.e. $\mathbb{E}[\mathcal{L}(y, \hat{y})] = \mathbb{E}[\mathcal{L}(y, 0)]$. That is, for a given parameter (e.g. noise scale) what is the directional accuracy needed to achieve the same loss as the zero-returns model. We show this for the MAE and MSE loss functions in Figure 2, with breakeven directional accuracy as a function of noise scale. Critically, for a Gaussian true returns distribution, monotonic loss functions and Gaussian linear prediction models, symmetric loss functions follow a universal breakeven function, $DA = 0.5 + \arctan(\rho_{\min} \sigma / \sigma_N) / \pi$ with $\rho_{\min} = 1 - \sqrt{1 - (\sigma_N / \sigma)^2}$, valid for $0 \leq \sigma_N \leq 1$ (both for evaluation with mean losses and the mean logarithm of the losses). We derive this result in Appendix B. At a noise scale $\sigma_N > 1$ (with $\sigma = 1$), Gaussian linear models cannot achieve a lower loss than the zero-returns model, regardless of their directional accuracy.

What can improve the breakeven point is using asymmetric functions, which reward overestimates relative to underestimates. This is shown by the dash-dotted colored lines in Figure 2, which use a constant α term to downweight losses for overestimates (analogous to quantile or LinLin loss functions). For the MSE loss we set α^2 scaling for equivalency with the MAE loss. For evaluation with mean (linear) losses (left panels in the figure), smaller α lowers the gradient in the σ_N -breakeven plane, but the breakeven points always remain $> 50\%$. The minimal breakeven point of course occurs when $\alpha = 0$, i.e. there is no punishment for overshoots in the correct direction. What is clear from comparing the MAE and MSE panels is that, for evaluation of mean losses, a linear term on the undershooting side is preferred over a quadratic term, since MAE with $\alpha = 0$ achieves far lower breakeven directional accuracy than MSE.

Taking the mean log loss¹ values for evaluation creates concave loss functions, which changes how asymmetric func-

¹Throughout, “mean log loss” (and “log loss” more generally) denotes log-averaged evaluation, i.e. the mean of the logarithm of the per-sample losses or equivalently the logarithm of their geometric mean. This is unrelated to the cross-entropy loss of classification, which is also commonly called

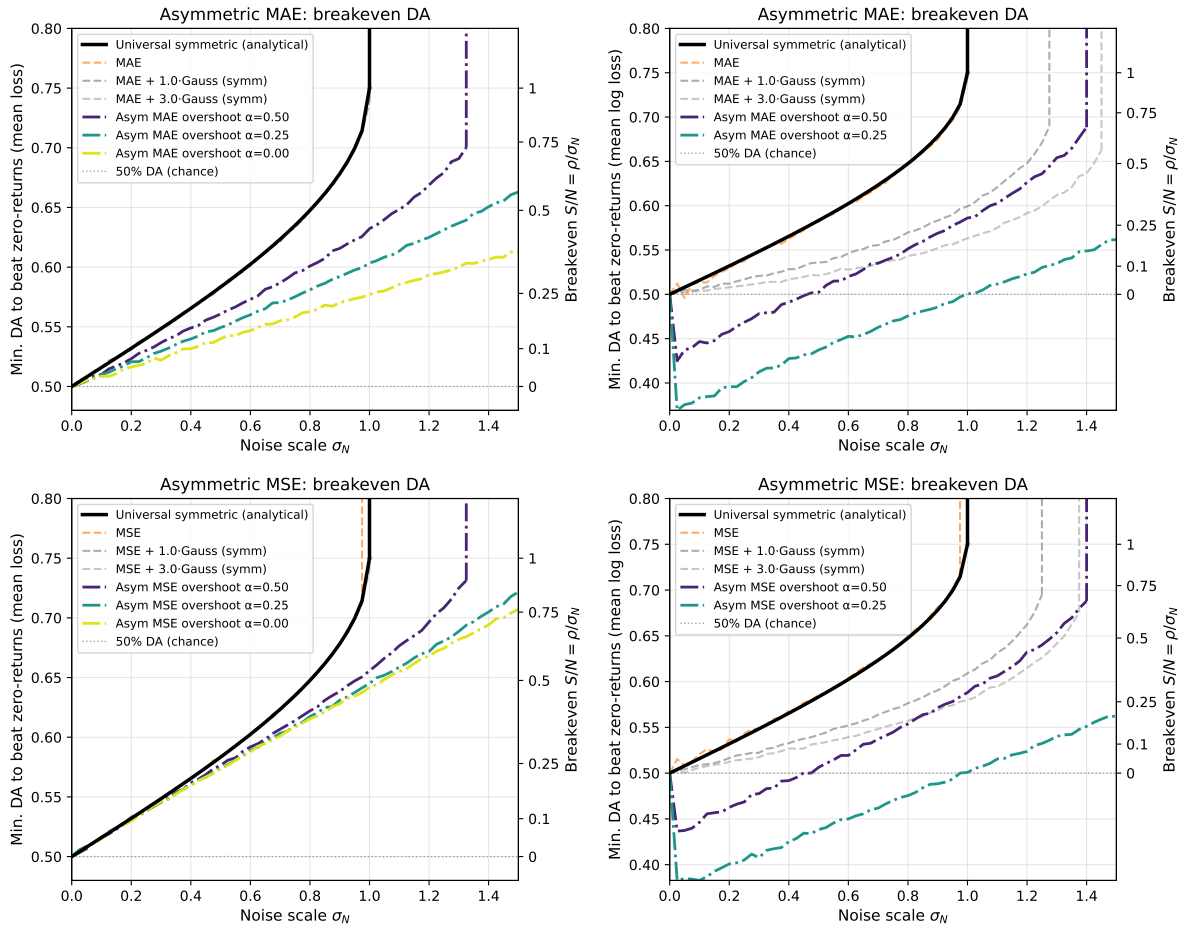


Figure 2: Breakeven directional accuracy vs. noise scale σ_N for asymmetric MAE (top row) and asymmetric MSE (bottom row), evaluated under mean linear loss (left) and mean log loss (right). Black solid lines show the universal symmetric analytical breakeven, orange dashed lines show symmetric MAE or MSE baseline, gray dashed lines show symmetric loss with an additive Gaussian floor ($\lambda \in \{1, 3\}$), and colored dash-dotted lines show asymmetric loss downweighting overshoots by factor $\alpha \in \{0, 0.25, 0.5\}$ (linear panels) or $\alpha \in \{0.25, 0.5\}$ (log panels). In all panels, the relations become vertical when Gaussian linear models cannot achieve lower losses than the zero-returns model. Left panels: For linear losses, asymmetry reduces the breakeven DA while remaining $> 50\%$ for all σ_N . A linear undershooting term (MAE, $\alpha = 0$, upper left panel) achieves far lower breakeven than a quadratic one (MSE, lower left panel), and the α sensitivity is negligible for MSE. Right panels: For log losses, asymmetry pushes the breakeven below 50% at small σ_N , meaning a model predicting consistently in the wrong direction could achieve a lower mean log loss than the zero-returns predictor, a clearly undesirable property. The Gaussian floor instead keeps the breakeven $\geq 50\%$ while reducing it at intermediate σ_N .

tions affect breakeven directional accuracy. For evaluation with mean log losses, the asymmetry term α acts as a vertical scaling. In this case, for small noise scales the breakeven point can occur at $< 50\%$; i.e. models could be optimized to *incorrectly* predict returns, which obviously should be avoided. For mean log losses, reduction in breakeven directional accuracy can be achieved by adding a ‘floor’ term to the loss function that peaks at $y = 0$ and decreases with increasing $|y|$. We show this as the gray dashed lines in Figure 2, which adds a floor following a scaled Gaussian, $\lambda \exp(-y^2)$, to the symmetric MAE loss. This achieves a similar reduction in breakeven directional accuracy for mean log losses as asymmetry does for mean losses. The floor term has no effect on mean loss since it is independent of the predicted values and cancels from the breakeven inequality.

These tests, together with the demands of gradient-based training (a custom objective in gradient-boosted libraries must supply a well-behaved gradient and Hessian, which favors convexity in the prediction) and the need to adequately penalize extreme outliers, motivate five requirements for a loss function designed for return prediction. Such a loss should:

- (i) be convex in the prediction $\hat{y}$ at fixed true value y (joint convexity is not required), making it suitable as a training objective for gradient-boosted models;
- (ii) be asymmetric, so that correct-direction overshoots are punished less than undershoots or wrong-direction predictions for the same fixed error, $|\hat{y} - y|$, while becoming symmetric when the true return is zero ($y = 0$);

¹‘log loss’.

- (iii) have a near-linear loss term for undershoots and wrong-direction predictions;
- (iv) diverge as the prediction error grows, so that $\mathcal{L} \rightarrow \infty$ for large errors and extreme outliers are adequately penalized; and
- (v) have an adaptive loss floor that raises the loss for samples whose true returns lie near the distribution mean, thereby reducing the relative weight of these samples under log-averaged evaluation.

3 CZAR loss function

The CZAR (Composite Zero-Agnostic Return) loss function is convex in the prediction $\hat{y}$ at fixed true value y (piecewise quadratic on either side of a region boundary at $\hat{z} = z$), designed around a single asymmetry: a prediction that overshoots the true return in the correct direction should be penalized less than one that undershoots or points in the wrong direction. This section presents the core formulation. Appendix C describes an optional pseudo-Huber smoothing extension for undershooting (Region A); in preliminary experiments it did not improve performance, and it is not used in any result reported in this paper.

3.1 Notation

Given a true return y and predicted return $\hat{y}$ with known distributional mean μ and standard deviation σ^2 , define standardized z -score variables

$$z = \frac{y - \mu}{\sigma}, \quad \hat{z} = \frac{\hat{y} - \mu}{\sigma}, \quad (1)$$

and let $\Delta z = |\hat{z} - z|$ denote the unsigned prediction error in standardized units. The key adaptive quantity is the asymmetry factor

$$\beta_{\text{eff}}(|z|) = \frac{1}{1 + \beta|z|} \in (0, 1], \quad (2)$$

which equals 1 at $z = 0$ and decays to 0 as $|z| \rightarrow \infty$, parameterized by $\beta \geq 0$. In the base loss below it multiplies the quadratic overshoot penalty, while its complement $1 - \beta_{\text{eff}}$ weights the linear undershoot term. Its two limits implement requirements (ii)–(iii) of §2. At $z = 0$ the loss is exactly symmetric, and as $|z|$ grows the asymmetry strengthens, so overshooting a large true move in the correct direction becomes nearly free while undershooting it is penalized at full linear weight.

3.2 Base loss

The base loss splits into two regions determined by whether the prediction overshoots the true value:

Region A ($\text{sign}(\hat{z}) \neq \text{sign}(z)$ or $|\hat{z}| \leq |z|$): prediction does not overshoot; either wrong direction or correct direction but undershooting.

Region B ($\text{sign}(\hat{z}) = \text{sign}(z)$ and $|\hat{z}| > |z|$): prediction overshoots; correct direction with magnitude exceeding $|z|$.

$$\mathcal{L}_{\text{base}}(z, \hat{z}) = \begin{cases} (1 - \beta_{\text{eff}}) \Delta z + \frac{\alpha}{2} \Delta z^2 & \text{(Region A),} \\ \beta_{\text{eff}} \frac{\alpha}{2} \Delta z^2 & \text{(Region B),} \end{cases} \quad (3)$$

where $\alpha > 0$ is the quadratic penalty scale, applied uniformly across both regions. This base loss satisfies the first four requirements from §2. At $z = 0$, $\beta_{\text{eff}} = 1$ and both regions collapse to the symmetric quadratic $\mathcal{L}_{\text{base}}|_{z=0} = \frac{\alpha}{2} \Delta z^2$ (with $\text{sign}(0) = 0$, every prediction formally falls in Region A, but the distinction is immaterial since the two expressions coincide); as $|z|$ grows, β_{eff} decays and the asymmetry between regions increases. The quadratic terms in both regions guarantee convexity in $\hat{y}$ (at fixed y) and unbounded growth for large prediction errors. We accordingly restrict the CZAR domain to $\alpha > 0$. At $\alpha = 0$ Region B is identically zero (no penalty for overshoot in the correct direction), and Region A reduces to a pure MAE-like $(1 - \beta_{\text{eff}}) \Delta z$, so the unbounded-growth requirement from §2 fails on the overshoot side and the loss admits a one-parameter degenerate family of zero-loss predictions. The correlated defaults compound the degeneracy because $\beta^*(0) = 0$ and $C^*(0) = 0$ collapse both the asymmetry and the loss floor, leaving the base loss flat in $\hat{y}$. We therefore treat $\alpha \rightarrow 0$ only as a limiting or diagnostic regime (e.g. the small- α entries in Figure 4 and Table 2 below stay strictly positive at $\alpha \geq 10^{-3}$), not as a CZAR operating point.

Figure 3 shows the sensitivity of the loss profile for a fixed true value $z = 2$. Here, α controls the overall quadratic scale uniformly across both regions, while β governs the asymmetry by setting how quickly β_{eff} decays with $|z|$.

² μ and σ are not fitted parameters of the loss but externally supplied properties of the target (true-return) distribution, treated as known constants for each sample. In practice they are estimated from recent history of the realized returns and the same values are used during training and evaluation. In our experiments (§4.2) σ is a 100-candle rolling standard deviation of realized returns and $\mu = 0$, the natural choice for log-returns at short horizons where the mean is negligible relative to the volatility. However, we retain μ in the notation for generality.

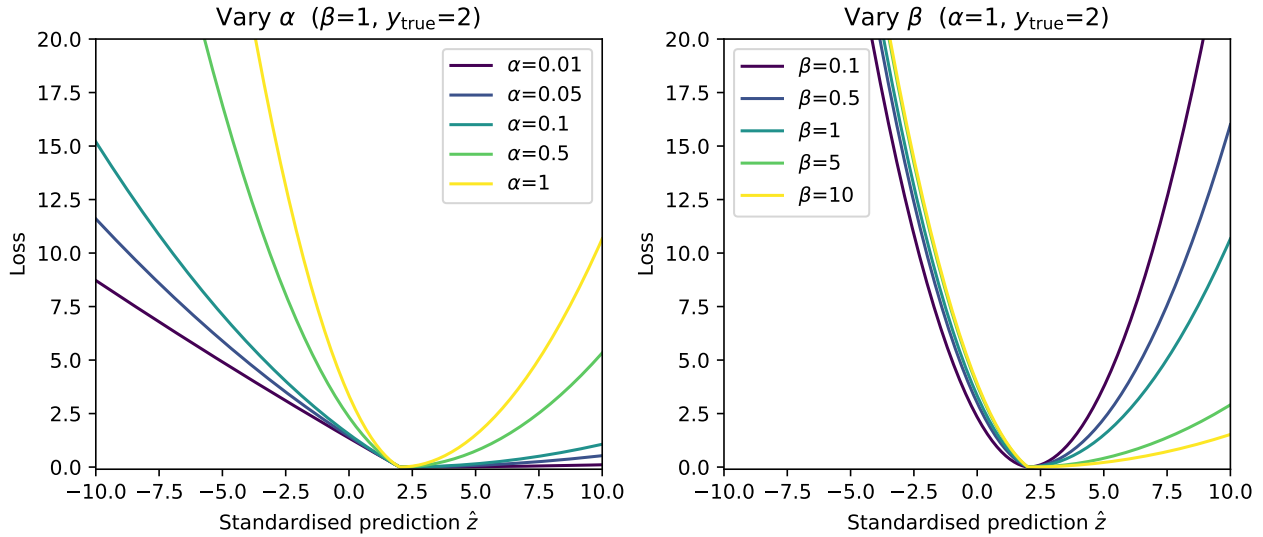


Figure 3: $\mathcal{L}_{\text{CZAR}}$ as a function of standardized prediction $\hat{z}$ for fixed true value $z = 2$, showing sensitivity to the two primary hyperparameters. The left panel shows varying α at fixed $\beta = 1$. Larger α steepens the quadratic penalty in both regions. The right panel shows varying β at fixed $\alpha = 1$. Larger β suppresses Region B (overshoot, $\hat{z} > z$), making the asymmetry more pronounced.

3.3 Gradient and Hessian

Let $s_{\Delta} = \text{sign}(\hat{z} - z)$, set to 0 at $\hat{z} = z$. Differentiating Equation (3) with respect to $\hat{y} = \mu + \sigma\hat{z}$:

$$\frac{\partial \mathcal{L}_{\text{base}}}{\partial \hat{y}} = \frac{s_{\Delta}}{\sigma} \begin{cases} (1 - \beta_{\text{eff}}) + \alpha \Delta z & \text{(Region A),} \\ \beta_{\text{eff}} \alpha \Delta z & \text{(Region B).} \end{cases} \quad (4)$$

The second derivative is

$$\frac{\partial^2 \mathcal{L}_{\text{base}}}{\partial \hat{y}^2} = \frac{1}{\sigma^2} \begin{cases} \alpha & \text{(Region A),} \\ \beta_{\text{eff}} \alpha & \text{(Region B).} \end{cases} \quad (5)$$

The MAE term in Region A contributes no curvature, so the Region A Hessian is simply α/σ^2 . At $\hat{z} = z$ with $z \neq 0$ the gradient has a step discontinuity of magnitude $(1 - \beta_{\text{eff}})/\sigma$. The Region A side-limit is $-\text{sign}(z)(1 - \beta_{\text{eff}})/\sigma$ and the Region B side-limit is 0. For $z > 0$, Region A lies left of the region boundary, so the gradient jumps from $-(1 - \beta_{\text{eff}})/\sigma$ up to 0 as $\hat{z}$ increases through z . For $z < 0$ the sides reverse and the gradient jumps from 0 up to $+(1 - \beta_{\text{eff}})/\sigma$. In both signs the one-sided jump $\partial^+ \mathcal{L}/\partial \hat{y} - \partial^- \mathcal{L}/\partial \hat{y} = (1 - \beta_{\text{eff}})/\sigma \geq 0$ is upward. This concentrates the reward signal near the true value. The loss pulls strongly toward z from the Region A side, while overshoot is penalized only quadratically.

For each fixed z , $\mathcal{L}_{\text{base}}(z, \cdot)$ is convex in $\hat{y}$ (equivalently in $\hat{z}$). On each open side of the region boundary the derivative $\partial \mathcal{L}_{\text{base}}/\partial \hat{y}$ is affine in $\hat{z}$ with non-negative slope (the Hessian is α/σ^2 on Region A and $\beta_{\text{eff}}\alpha/\sigma^2$ on Region B, both ≥ 0), and at $\hat{z} = z$ the left and right one-sided derivatives satisfy $\partial^+ \mathcal{L}/\partial \hat{y} \geq \partial^- \mathcal{L}/\partial \hat{y}$ (upward jump $(1 - \beta_{\text{eff}})/\sigma \geq 0$ established above). The subdifferential $\partial \mathcal{L}_{\text{base}}(z, \cdot)$ is therefore monotone non-decreasing in $\hat{y}$, which is sufficient for convexity in $\hat{y}$ at fixed y . We also note that joint convexity in $(y, \hat{y})$ is not claimed and does not hold, since β_{eff} depends on $|z|$.

The gradient and Newton step are shown in Figure 4. The step discontinuity is visible for all $z \neq 0$, while at $z = 0$, $\beta_{\text{eff}} = 1$ and the gradient is continuous and linear throughout. At fixed β , varying α would change the post-step slope while leaving the step height unchanged, since the step arises from the MAE term rather than the quadratic. Under the correlated default $\beta^*(\alpha)$, however, smaller α also reduces β , so the step height $1 - \beta_{\text{eff}}(|z|)$ shrinks alongside the slope, visible as the much smaller steps for the dashed ($\alpha = 0.01$, $\beta \approx 0.32$) curves compared with the solid ($\alpha = 1$, $\beta \approx 31.2$) curves. The Newton step (right panel) behaves differently in the two regions. In Region B the shared factor $\beta_{\text{eff}}\alpha$ cancels between gradient and Hessian, so the step is exactly σ , $\Delta z = \hat{y} - y$, independent of α and β_{eff} ; i.e. the overshoot damping present in the gradient leaves the Newton step unchanged. In Region A the same $\sigma, \Delta z$ is augmented by a constant $\sigma(1 - \beta_{\text{eff}})/\alpha$ from the MAE term, which grows as $|z|$ increases and α decreases, so the undershoot side takes the largest Newton steps for large true values.

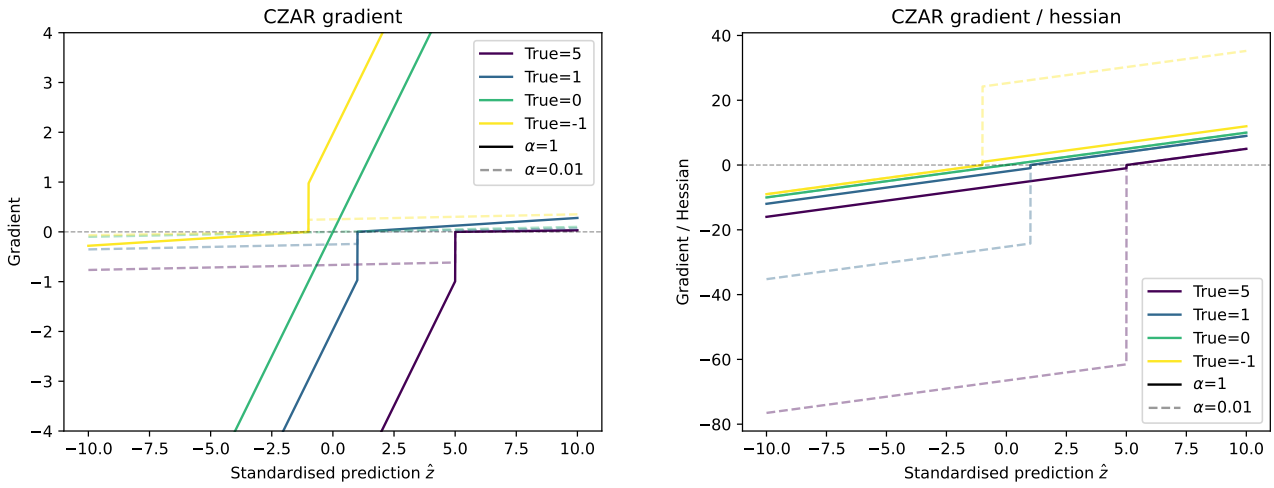


Figure 4: Gradient and Newton step of $\mathcal{L}_{\text{CZAR}}$ for standardized true values $z \in \{-1, 0, 1, 5\}$. The left panel shows the gradient $\sigma \partial \mathcal{L}_{\text{CZAR}} / \partial \hat{y}$ vs. standardized prediction $\hat{z}$. Each curve has a step discontinuity at $\hat{z} = z$ of magnitude $1 - \beta_{\text{eff}}(|z|)$. The gradient approaches $-\text{sign}(z)(1 - \beta_{\text{eff}})$ from Region A and 0 from Region B, so it always jumps upward as $\hat{z}$ increases through z . For $z > 0$ (e.g. $z = 1, 5$) Region A lies to the left of the region boundary and the gradient jumps from $-(1 - \beta_{\text{eff}})$ up to 0. For $z < 0$ (e.g. $z = -1$) Region A lies to the right and the jump runs from 0 up to $+(1 - \beta_{\text{eff}})$. For large $|z|$ the step grows as $\beta_{\text{eff}} \rightarrow 0$, while for $z = 0$ ($\beta_{\text{eff}} = 1$) the gradient is continuous and linear (MSE-like), with no discontinuity. The right panel shows the Newton step $(\partial \mathcal{L} / \partial \hat{y}) / (\partial^2 \mathcal{L} / \partial \hat{y}^2)$ vs. $\hat{z}$. In Region B the factor $\beta_{\text{eff}}\alpha$ is shared by the gradient and Hessian and cancels in the ratio, leaving a Newton step of $\sigma, \Delta z = \hat{y} - y$ that is independent of α and β_{eff} (i.e. the overshoot damping seen in the loss and gradient does not carry over to the Newton step). In Region A the constant Hessian (α/σ^2) gives a Newton step $\sigma, \Delta z + \sigma(1 - \beta_{\text{eff}})/\alpha$, i.e. the MSE step plus a constant offset from the MAE term that grows as $|z|$ increases and as α shrinks. In both panels, solid lines show $\alpha = 1$ and dashed lines show $\alpha = 0.01$. Both use the correlated asymmetry term $\beta = \beta^*(\alpha)$ from Equation (10) and floor level $C = C^*(\alpha)$ from Equation (11) (with $\tau = 0.5$ and $\mu = 0$).

3.4 Adaptive loss floor

The base loss of a zero prediction approaches zero as $|z| \rightarrow 0$, so samples with true returns near zero contribute very little to the loss, allowing a zero-returns predictor to achieve a deceptively low mean log loss. Therefore, to satisfy requirement (v) of §2, we add a floor that raises the minimum loss at small true returns. The floor equals C at $z = 0$ and decays to zero as $|z|$ grows.

Specifically, writing the zero-prediction base loss as

$$\begin{aligned} \mathcal{L}_0(z) &= \mathcal{L}_{\text{base}}(z, 0) \\ &= (1 - \beta_{\text{eff}})|z| + \frac{\alpha}{2} z^2 \end{aligned} \quad (6)$$

(a zero prediction never overshoots, so always falls in Region A with $\Delta z = |z|$), the increase required to raise $\mathcal{L}_0(z)$ to at least C is $\max(C - \mathcal{L}_0(z), 0)$. To avoid unintended behavior (see Appendix D), we implement a smoothed version of the rectification function. Define the smooth hinge

$$\text{hinge}(x, \tau) = \frac{1}{2} \left(x + \sqrt{x^2 + \tau^2} \right), \quad (7)$$

which approximates $\max(x, 0)$ for small $\tau > 0$. The loss floor then becomes

$$\mathcal{L}_{\text{floor}}(z) = C \cdot \frac{\text{hinge}(C - \mathcal{L}_0(z), \tau)}{\text{hinge}(C, \tau)}. \quad (8)$$

The denominator removes the upward bias of the hinge, so the floor equals exactly C at $z = 0$ (where $\mathcal{L}_0 = 0$) and approaches zero smoothly as $\mathcal{L}_0$ grows beyond C . The total loss at $\hat{z} = 0$ is thus pinned to $\approx \max(\mathcal{L}_0, C)$, non-decreasing in $|z|$ because $\mathcal{L}_0$ is strictly increasing.

Since $\mathcal{L}_{\text{floor}}$ does not depend on $\hat{z}$, it contributes nothing to the gradient or Hessian and is invisible to gradient-based training. The pressure away from the zero attractor in model training is thus carried entirely by the Region A asymmetry of $\mathcal{L}_{\text{base}}$ (for any $z \neq 0$, Equation (4) gives a gradient signed away from zero). The role of the loss floor is instead in evaluation, ranking, and model selection. It prevents a zero-returns predictor from achieving a misleadingly low mean log loss and, with the correlated default $C^*(\alpha)$, keeps the breakeven directional accuracy at or above 50% across noise scales (Appendix D).

Parameter	Role	Domain	Default
α	Quadratic regularization strength	$\alpha > 0$	1
β	Asymmetry decay rate	$\beta \geq 0$	$\beta^*(\alpha)$, eq. (10)
C	Floor target level	$C \geq 0$	$C^*(\alpha)$, eq. (11)
τ	Hinge smoothing width	$\tau > 0$	0.5

Table 1: CZAR hyperparameters. At $\beta = 0$, $\beta_{\text{eff}} = 1$ for all z and both regions are purely quadratic (MSE-like). Larger β increases the asymmetry between Region A (MAE+quadratic) and Region B (discounted quadratic) for returns with large $|z|$.

3.5 Total loss

The total CZAR loss therefore becomes

$$\mathcal{L}_{\text{CZAR}}(y, \hat{y}) = \mathcal{L}_{\text{base}}(z, \hat{z}) + \mathcal{L}_{\text{floor}}(z), \quad (9)$$

where the raw pair $(y, \hat{y})$ is converted to $(z, \hat{z})$ through Equation (1) and the properties of the target distribution (μ, σ) . An example Python implementation for the loss function is described in Appendix A.

The loss function has four hyperparameters (α, β, C, τ) , which are summarized in Table 1. However, the parameters α, β and C are not independent in practice. For each value of α , there is an optimal $\beta^*(\alpha)$ that minimizes the breakeven directional accuracy (as defined in §2). Sweeping both parameters over the breakeven DA at $\sigma_N = 1$ yields optimal pairs well described by a two-term power-law

$$\beta^*(\alpha) = 4.2 \alpha^{0.56} + 27.0 \alpha^{2.17}, \quad (10)$$

on the calibrated range $10^{-4} \leq \alpha \leq 0.7$ (at larger α the breakeven basin in β becomes too shallow to fit, see Appendix D). At the reference value $\alpha = 1$, Equation (10) gives $\beta^* \approx 31.2$. The implementation uses $\beta^*(\alpha)$ as the default when β is not specified explicitly. We show the full optimization results in Appendix D.

Similarly, an optimal floor level $C^*(\alpha)$ can be found by minimizing the log-loss³ breakeven directional accuracy at fixed τ , subject to the constraint that it remains $\geq 50\%$. The optimal values are well-described by an asymmetric Hill-bell model with a linear correction

$$C^*(\alpha) = A h(\alpha)^k [1 - h(\alpha)] + (R + m\alpha) h(\alpha), \quad h(\alpha) = \frac{\alpha^p}{t^p + \alpha^p}, \quad (11)$$

over the range $10^{-5} \leq \alpha \leq 1$ with $(A, t, p, k, R, m) = (7.90, 4.59 \times 10^{-3}, 0.657, 0.684, 2.25, -0.218)$. This form satisfies $C^*(0) = 0$, peaks near $\alpha \sim 10^{-2}$, and decays slowly at larger α . The implementation uses $C^*(\alpha)$ as the default when C is not specified explicitly; at $\alpha = 1$ this gives $C^* \approx 2.19$. The hinge width τ is relatively insensitive to α and β ; we adopt $\tau = 0.5$ as the default. As with $\beta^*(\alpha)$, the fit in Equation (11) is intended for the working range $\alpha \lesssim 1$ (Appendix D). Extrapolating well beyond that range, the linear correction causes $C^*(\alpha)$ to become negative near $\alpha \approx 10$, outside the domain $C \geq 0$ of Table 1. The reference implementation (Appendix A) therefore disables the loss floor whenever $C \leq 0$.

Figure 5 shows $\mathcal{L}_{\text{CZAR}}$ as a function of standardized prediction $\hat{z}$ for several true values, at two parameter sets each using the correlated default $\beta^*(\alpha)$. Each curve is asymmetric about its minimum at $\hat{z} = z$. For large $|z|$, Region A (undershoot and wrong-direction predictions) is nearly linear, i.e. $\beta_{\text{eff}} \rightarrow 0$ and the MAE-like term $(1 - \beta_{\text{eff}})\Delta z$ dominates. Region B (overshoot) is instead near-flat for large $|z|$. The discounted quadratic $\beta_{\text{eff}} \frac{\alpha}{2} \Delta z^2$ contributes negligible penalty once β_{eff} is small, so overshooting a large true return is almost free. At $z = 0$, the two regions collapse to a single MSE-like parabola; the loss floor then provides the bulk of the loss, pinning the zero-prediction loss to the target level C . Smaller α (dashed) scales down the overall loss without qualitatively changing the shape.

To see the full shape of the loss surface rather than slices through it, Figure 6 plots $\mathcal{L}_{\text{CZAR}}$ over the $(y, \hat{y})$ plane alongside MSE as a reference symmetric baseline. MSE is symmetric about the diagonal $\hat{y} = y$ and depends only on the residual magnitude $(\hat{y} - y)^2$. CZAR breaks this symmetry along the orthogonal direction. Predictions that share the sign of the true return but overshoot in magnitude (off-diagonal points in the upper-right and lower-left quadrants) incur only a small additional penalty above the diagonal, whereas undershoots and wrong-direction predictions (off-diagonal points in the opposite-sign quadrants) are penalized much more heavily. The loss floor seen as the elevated minimum of the $z = 0$ curve in Figure 5 appears here as a bright horizontal band of elevated loss along $y = 0$ in the CZAR maps. This raises the sample-averaged (ranking) loss of a near-zero predictor, since predictions in this strip cannot bring their loss arbitrarily close to zero; it does not act on per-sample gradients, which depend only on $\mathcal{L}_{\text{base}}$ (§3.4). The corresponding training-time suppression of $\hat{z} \approx 0$ is therefore carried by the Region A asymmetry of $\mathcal{L}_{\text{base}}$, whose gradient is signed away from zero whenever $z \neq 0$.

³Recall that constant terms in loss functions do not affect the breakeven directional accuracy for linear losses, see §2.

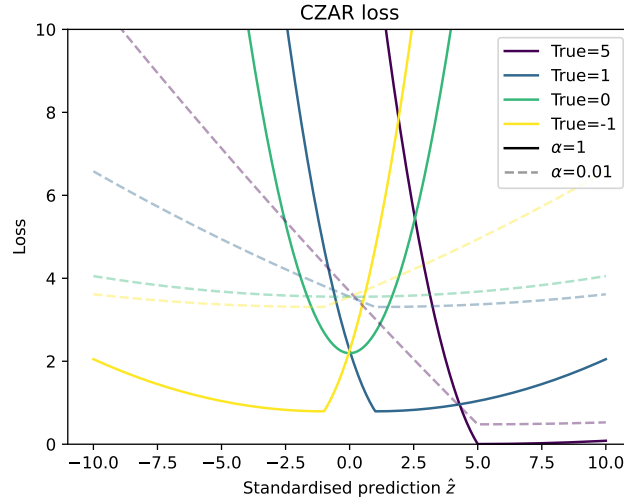


Figure 5: $\mathcal{L}_{\text{CZAR}}$ as a function of standardized prediction $\hat{z}$ for several standardized true values z (default parameters, $\mu = 0$) and $\alpha = 1$ (solid lines) and 0.01 (dashed lines). Each curve attains its minimum at $\hat{z} = z$. For large $|z|$, Region A (undershooting or wrong direction) adopts a near-linear profile while Region B (overshoot) remains quadratic, reflecting the asymmetry encoded by β_{eff} . The loss floor raises the minimum loss for small $|z|$, visible as the elevated minimum of the $z = 0$ curve.

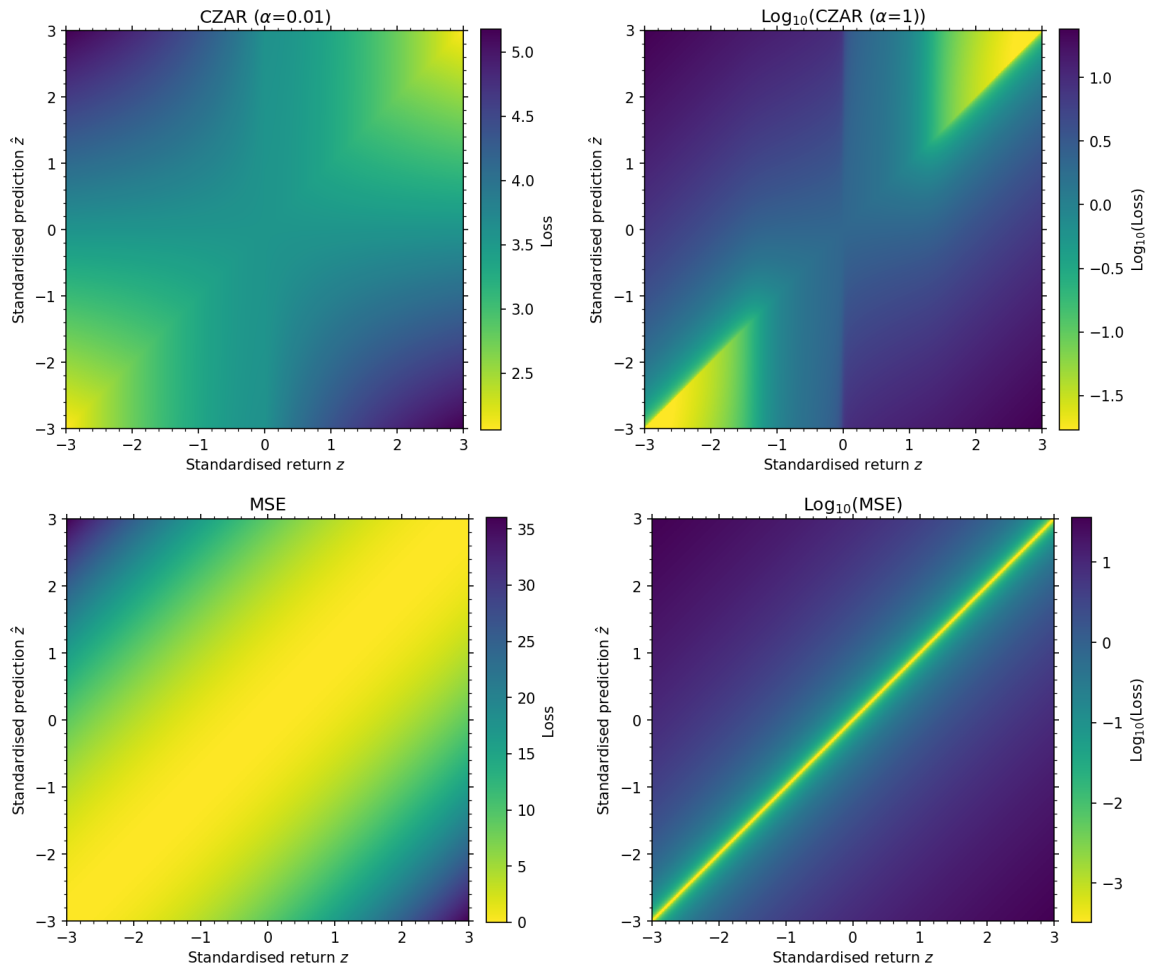


Figure 6: Loss in the $(z, \hat{z})$ plane for CZAR (top) and MSE (bottom). Left panels show loss on a linear scale, as seen by a gradient-based optimizer; right panels show $\log_{10}(\text{loss})$, more representative of the dynamic range used when ranking models. The CZAR linear panel uses $\alpha = 0.01$ (where the MAE-like undershoot regime is most visible on a linear scale) while the CZAR log panel uses $\alpha = 1$ (where the dynamic range is best resolved on a log scale); both use the correlated defaults $\beta^*(\alpha)$, $C^*(\alpha)$, and $\tau = 0.5$.

4 Evaluation

4.1 Idealized tests

We revisit the Gaussian linear testbed of §2, $y \sim \mathcal{N}(0, 1)$ with $\hat{y} = \rho y + \sigma_N \varepsilon$, and ask two questions of CZAR: how does the breakeven directional accuracy compare to the symmetric and constant-asymmetry envelopes derived in §2; and does the ordering induced by CZAR over a broad (ρ, σ_N) sweep correctly track the prediction-quality metrics that matter downstream? Throughout this subsection, CZAR uses the correlated defaults $\beta^*(\alpha)$ and $C^*(\alpha)$ from Equations (10) and (11), with $\tau = 0.5$. Loss expectations are estimated by Monte Carlo over 5×10^4 samples per (ρ, σ_N) pair.

Figure 7 repeats the breakeven analysis of §2 for CZAR. Under mean log loss, CZAR tracks the 50% chance line to within $\lesssim 5$ pp across the full $\sigma_N \in [0, 1.5]$ range, a substantial improvement over both the universal symmetric envelope (which rises sharply as $\sigma_N \rightarrow 1$) and the lower bound achievable with a purely asymmetric MAE loss ($\alpha = 0$ overshoot reward). Under mean linear loss, CZAR's breakeven is intermediate but still improves on the asymmetric-MAE envelope at every noise scale. In the Gaussian linear testbed, the combination of asymmetry and the adaptive loss floor substantially reduces the zero-returns breakeven directional-accuracy threshold, narrowing the regime in which a model with genuine but noisy directional skill ranks below the zero-returns predictor, a regime that persists even for the best-tuned symmetric or constant asymmetry losses in Figure 2.

The Gaussian linear testbed is a deliberately friendly baseline; real return distributions are heavier-tailed, with empirically documented tail indices in the range $\nu \sim 3\text{--}5$ (Blattberg & Gonedes, 1974; Cont, 2001; Gopikrishnan et al., 1999). Figure 8 repeats the breakeven sweep with y and the noise term ε for $\hat{y}$ drawn from unit-variance Student- t distributions at $\nu = 5$ (typical daily equity) and $\nu = 3$ (heavy-tail stress test, for which the kurtosis is infinite). Under linear loss, both CZAR and MSE breakeven curves are empirically close across these unit-variance distributions, and the CZAR advantage over MSE is preserved across all three. Under log loss, heavier tails raise the breakeven DA for both losses, but CZAR retains a substantial margin over MSE at every σ_N and at every tail index considered. The mild degradation of CZAR's log-loss breakeven under heavier tails is at least partly an artifact of our defaults. $C^*(\alpha)$ and τ were calibrated against the Gaussian linear testbed, so re-optimizing these parameters against a heavy-tailed reference distribution should recover much of the Gaussian-case performance for problems where heavy tails are expected. The combination of asymmetry and adaptive loss floor therefore does not rely on Gaussian tails to deliver its advantage.

The breakeven analysis collapses each (ρ, σ_N) pair to a single scalar (the minimum DA needed to beat zero returns). Figure 9 unfolds this view by comparing the mean log-loss against two prediction-quality metrics (directional accuracy, top, and log aspect ratio $\log_{10}(\sigma_{\text{pred}}/\sigma_{\text{true}})$, bottom) across a wide sweep $\rho \in [-1.5, 1.5]$, $\sigma_N \in [0, 1.5]$. The dashed vertical line in each panel is the log-loss of the zero-returns predictor, so models to its left rank above the zero-returns baseline. Ideally, all models with positive signal should have lower losses than the zero-returns predictor. Two failure modes of the symmetric losses (MAE, MSE, Huber) are visible.

Directional-accuracy failure mode (top panels). For each symmetric loss, the high- $|\rho|$, high- σ_N branch (models with $\text{DA} \approx 0.6\text{--}0.7$) sits entirely to the right of the zero-returns line. Any ranking based on these losses therefore prefers the trivial constant predictor to a noisy model with real directional skill. Under CZAR, the asymmetric overshoot discount lowers the loss of directionally correct ($\rho > 0$) noisy predictions, moving this branch leftward across the zero-returns line, while the negative-signal ($\rho < 0$) branch remains in the penalized region; informative noisy models are then correctly preferred. Beyond this shift in position, the $\rho > 0$ branch also collapses onto a single DA-log-loss track. Different (ρ, σ_N) pairs with the same DA land at essentially the same CZAR loss, so ranking by mean CZAR loss tracks ranking by DA directly. Symmetric losses fail this alignment because they spread the same DA values across a wide log-loss band, reflecting residual magnitude rather than directional correctness.

Aspect-ratio degeneracy (bottom panels). At fixed ρ , increasing σ_N moves $\hat{y} = \rho y + \sigma_N \varepsilon$ to higher $\log_{10}$ AR as the noise contribution to $\sigma_{\hat{y}}$ grows. Under the symmetric losses, this sweep also raises the log-loss. A fixed- ρ sequence thus traces a diagonal in the panel, so noise is directly punished at fixed signal. A consequence is that a model with no directional skill can reduce its loss simply by shrinking the magnitude of its predictions, since this reduces the average residual magnitude; the well-known regression-to-the-mean attractor. Under CZAR, the same fixed- ρ sequence runs nearly vertically (at least for $\rho \lesssim 0.5$). The mean log-loss is essentially independent of σ_N and is set by the signal level alone. Ranking by mean CZAR loss therefore tracks the directional signal rather than the prediction's overall variance, decoupling rank from prediction noise and magnitude.

4.2 Model training

In this section, we turn from idealized Gaussian predictions to a gradient-boosted model trained on real data. We test a LightGBM regressor predicting BTC log-returns at two intraday horizons (15-minute and 1-hour candles), with CZAR as the training objective through its analytical gradient and Hessian (§3.3). This experiment is intended as a controlled illustration of how the choice of training loss affects what a model learns, not as a search for the best attainable forecaster; the absolute numbers matter far less than the comparison across losses. The feature set is intentionally restricted to a basic, easy-to-reproduce set, so the question is which loss extracts the most usable signal from the simplified set, rather

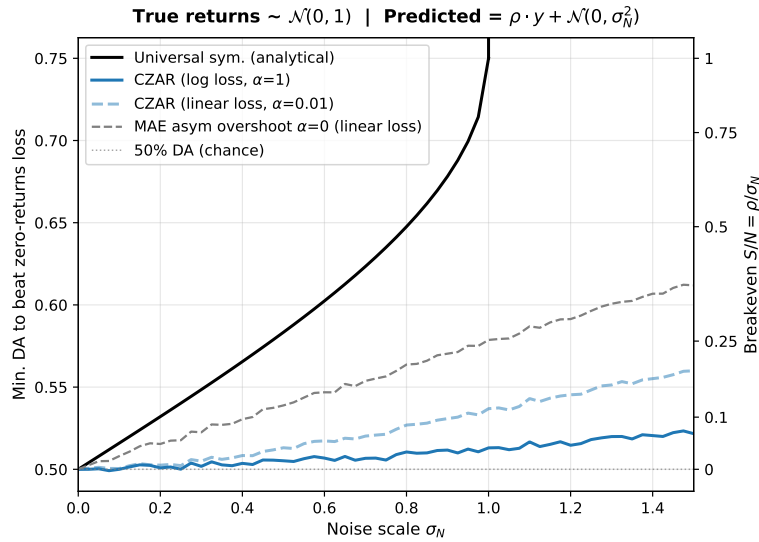


Figure 7: Breakeven directional accuracy vs. noise scale σ_N under the Gaussian linear prediction model. The left axis shows the minimum DA required to beat the zero-returns predictor; the right axis shows the equivalent breakeven signal-to-noise ρ/σ_N . The black solid line shows universal analytical breakeven for symmetric monotonic losses, $DA = 0.5 + \arctan(\rho_{\min}/\sigma_N)/\pi$ with $\rho_{\min} = 1 - \sqrt{1 - \sigma_N^2}$; the gray dashed line shows the lower envelope achievable with the asymmetric MAE loss ($\alpha = 0$) under mean linear loss (reproduced from Figure 2); the light blue dashed line shows CZAR under mean linear loss ($\alpha = 0.01$); and the dark blue solid shows CZAR under mean log loss ($\alpha = 1$). Both CZAR curves use the correlated defaults $\beta^*(\alpha)$, $C^*(\alpha)$, and $\tau = 0.5$.

than whether the features are competitive in absolute terms. The learning rate is also calibrated per loss and then held fixed (as described below), factoring out the strong dependence of the effective step size on the loss function.

For the CZAR loss we sweep α from 0.005 to 1 and additionally report a CZAR-tuned variant (where α is treated as an Optuna parameter, but using fixed $\alpha = 1$ for evaluation to keep the loss scale constant). The CZAR loss variants are compared to two symmetric baseline loss functions (L1, L2). Table 2 summarizes the held-out test set at each horizon across the prediction-quality metrics introduced in §4: directional accuracy (DA) over the full sample and restricted to large-magnitude truths (those with $|z| > 1\sigma$, and separately those outside the interquartile range), log aspect ratio $\log_{10}(\sigma_{\hat{y}}/\sigma_y)$, Pearson correlation R_P , rank correlation (IC), and a naive (annualized) Sharpe ratio⁴.

Inputs are one-minute BTC/USD OHLCV candles sourced from Tiingo⁵ (retrieved on 2026-04-30), aggregated to the prediction horizon. The target is the next aggregated-candle log return $\log(c_{t+1}/c_t)$. For the CZAR objective and evaluation, returns are standardized by a trailing volatility σ , computed using a 100-candle rolling standard deviation of single-candle realized returns (minimum 50 candles). We use a single most-recent split at each horizon of 20 000 candles for training (with the last 2 000 held out as Optuna inner validation), a one-candle gap, then a 2 000-candle test window. Features are computed exclusively from past candles: single-candle log returns and within-candle realized volatilities at the six most recent lags; candle-geometry features (the log high–low range, the open-to-close return, the upper and lower wick lengths, and the position of the close within the candle range); the log gap between the close and the OHLC4 typical price together with its 5- and 15-candle rolling means; volume features (log volume and two log volume-change ratios over horizon-relative windows); and sin/cos encodings of time-of-day and day-of-week (from UTC timestamps).

Each entry in Table 2 is independently tuned with Optuna (TPE sampler, 300 trials per model), minimizing the sample mean of the logarithm of the per-sample values of the model’s own training loss, $\langle \log |\mathcal{L}| \rangle$. The search is joint over the LightGBM hyperparameters `num_leaves` $\in [15, 127]$, `min_child_samples` $\in [20, 300]$, `subsample` and `colsample_bytree` $\in [0.6, 1]$, and the L^1 and L^2 leaf regularization strengths $\in [10^{-8}, 10]$ (log-scaled), with a fixed random seed (42) so that each per-row Optuna study is deterministic. The training window is fixed at 20 000 candles. Rather than tuning the number of trees, each fit early-stops after 100 rounds with no inner-validation gain (capped at 5 000, though in practice it is typically < 200), and the stopped iteration count sets the final tree count. The learning rate requires care because different losses produce Newton steps of very different magnitude. A fixed rate would place L2 and small- α CZAR at very different points on the convergence curve. We calibrate a per-loss scale s_{loss} by fitting a one-tree LightGBM with `learning_rate` = 1 for each loss, measuring the standard deviation of its first-step predictions, and dividing by the L2 reference; the learning rate is then fixed at $0.05/s_{\text{loss}}$, anchored to a base L2 rate of 0.05. For the

⁴Each period the strategy takes a unit long position when $\hat{y} > 0$ and a unit short position when $\hat{y} < 0$, earning the realized return $\text{sign}(\hat{y})y$. The reported figure is the ratio of the mean of $\text{sign}(\hat{y})y$ to its standard deviation, annualized by $\sqrt{T}$ (the annualization factor is given below). We call it *naive* because it uses only the sign of the prediction (discarding magnitude and confidence), stays fully invested and rebalances every candle, and excludes transaction costs, slippage, and position sizing.

⁵<https://www.tiingo.com/>

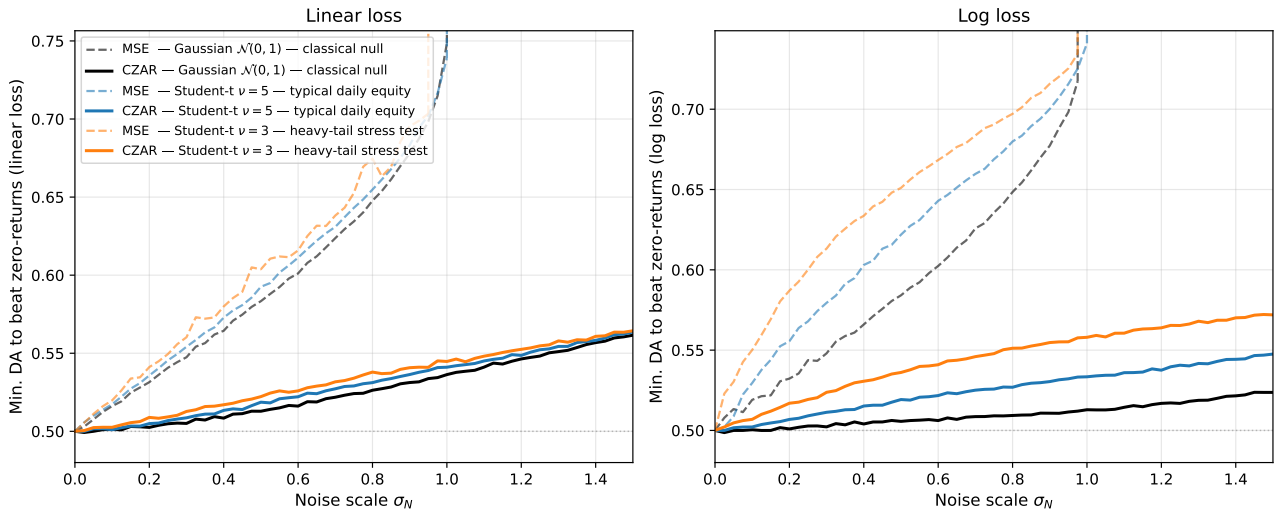


Figure 8: Breakeven directional accuracy vs. noise scale σ_N under Gaussian (black lines) and heavy-tailed Student- t truth distributions (blue lines for $\nu = 5$ and orange lines for $\nu = 3$), all normalized to unit variance. The left panel shows mean linear loss. The right panel shows mean log loss. In both panels, solid lines show the CZAR loss with the correlated defaults $\beta^*(\alpha)$, $C^*(\alpha)$, $\tau = 0.5$ ($\alpha = 0.01$ for linear loss, $\alpha = 1$ for log loss), and dashed lines show the MSE loss. Under linear loss the breakeven curves are close to distribution-invariant. Under log loss heavier tails raise the breakeven DA for both losses, but CZAR retains its margin over MSE at every σ_N and tail index.

CZAR-tuned row, $\alpha \in [10^{-3}, 1]$ is sampled on a log grid as an additional Optuna parameter and the scale is interpolated between calibrated grid points. Custom losses are passed to LightGBM through its `objective` hook (using the closed-form gradient and Hessian from §3.3) and the Hessian is clipped from below at 10^{-6} to guard against degenerate Newton steps when β_{eff} is very small. The naive Sharpe ratios are annualized by $\sqrt{525\,960/\ell}$, where ℓ is the prediction horizon in minutes and 525 960 is the number of one-minute intervals per year, inferred from the median candle interval; this gives factors of ≈ 187 at the 15-minute horizon and ≈ 94 at the 1-hour horizon. The $\sqrt{T}$ scaling assumes serially uncorrelated returns and should be read as a reporting convention rather than an estimate of attainable annual performance (Lo, 2002).

The qualitative picture matches the idealized analysis of §4.1. The two symmetric losses exhibit the aspect-ratio degeneracy. At both horizons their predictions are shrunk roughly an order of magnitude below the truth ($\log_{10} \text{AR} \approx -0.9$ to -1.2), their full-sample DA hovers near the chance line, and their directional accuracy on large-magnitude moves falls at or below 50% ($\text{DA}_{1\sigma} \approx 0.47$ – 0.50). In contrast, CZAR keeps the aspect ratio within a factor of a few of unity across the whole α sweep and lifts $\text{DA}_{1\sigma}$ above chance for every setting (≈ 0.50 – 0.56), while raising R_P and IC and generally improving the naive Sharpe.

However, the improvement for CZAR is not uniform across every metric. Given a binomial standard error of ≈ 1.1 percentage points (pp) on the 2000-candle test window, full-sample DA is generally statistically indistinguishable from L1 and L2. Instead, the improvement in CZAR tends to occur for larger moves ($\text{DA}_{1\sigma}$ and DA_{IQR}), which also help to drive the several-fold gain in R_P and higher naive Sharpe. The conditional accuracies $\text{DA}_{1\sigma}$ and DA_{IQR} are computed on tail subsets (roughly a third and a half of the sample) and so carry larger standard errors (≈ 2 and ≈ 1.6 pp), but the sign of the gap (symmetric losses at or below chance, CZAR above) is consistent across the two horizons and both conditioning rules.

Every fixed CZAR setting clears both symmetric losses (L1 and L2) on the naive Sharpe at the 1-hour horizon, and all but one ($\alpha = 0.01$) do so at the 15-minute horizon. The tuned variant lands at a small α at both horizons (≈ 0.054 and ≈ 0.0045) and delivers the best 15-minute $\text{DA}_{1\sigma}$ (0.561) while keeping the aspect ratio well-behaved. The log aspect ratio itself varies systematically with α . It grows as α decreases, from ≈ -0.82 at $\alpha = 1$ to ≈ -0.46 at $\alpha = 0.005$ at the 15-minute horizon (and the same trend, more noisily, at the 1-hour horizon), because a smaller α produces larger Newton steps in the undershoot/wrong-direction region and so pushes predictions further from zero (Figure 4).

A final check isolates where the improvement comes from. The metrics in Table 2 tune and early-stop each model on its own log-loss (the CZAR log-loss for the CZAR rows, the symmetric log-loss for L1 and L2). Instead, we repeat the 15-minute experiment selecting *every* model, including the CZAR-trained ones, on the symmetric log-L1 validation loss instead; the boosting gradients are unchanged, so only the model-selection and early-stopping criterion differs. In this test, the CZAR models are pulled back onto the aspect-ratio degeneracy. Their log aspect ratios collapse from the -0.4 to -0.8 range into the -1.0 to -1.4 range of the symmetric losses (predicted spreads shrink from roughly a quarter to a tenth of the true scale), and $\text{DA}_{1\sigma}$ and DA_{IQR} are reduced to $\approx 50\%$. Training with CZAR is thus necessary but not sufficient (a symmetric selection criterion silently reintroduces the shrinkage attractor even when the gradients are CZAR's), so the gains reported above rely on using CZAR as both the training objective and the model-selection loss.

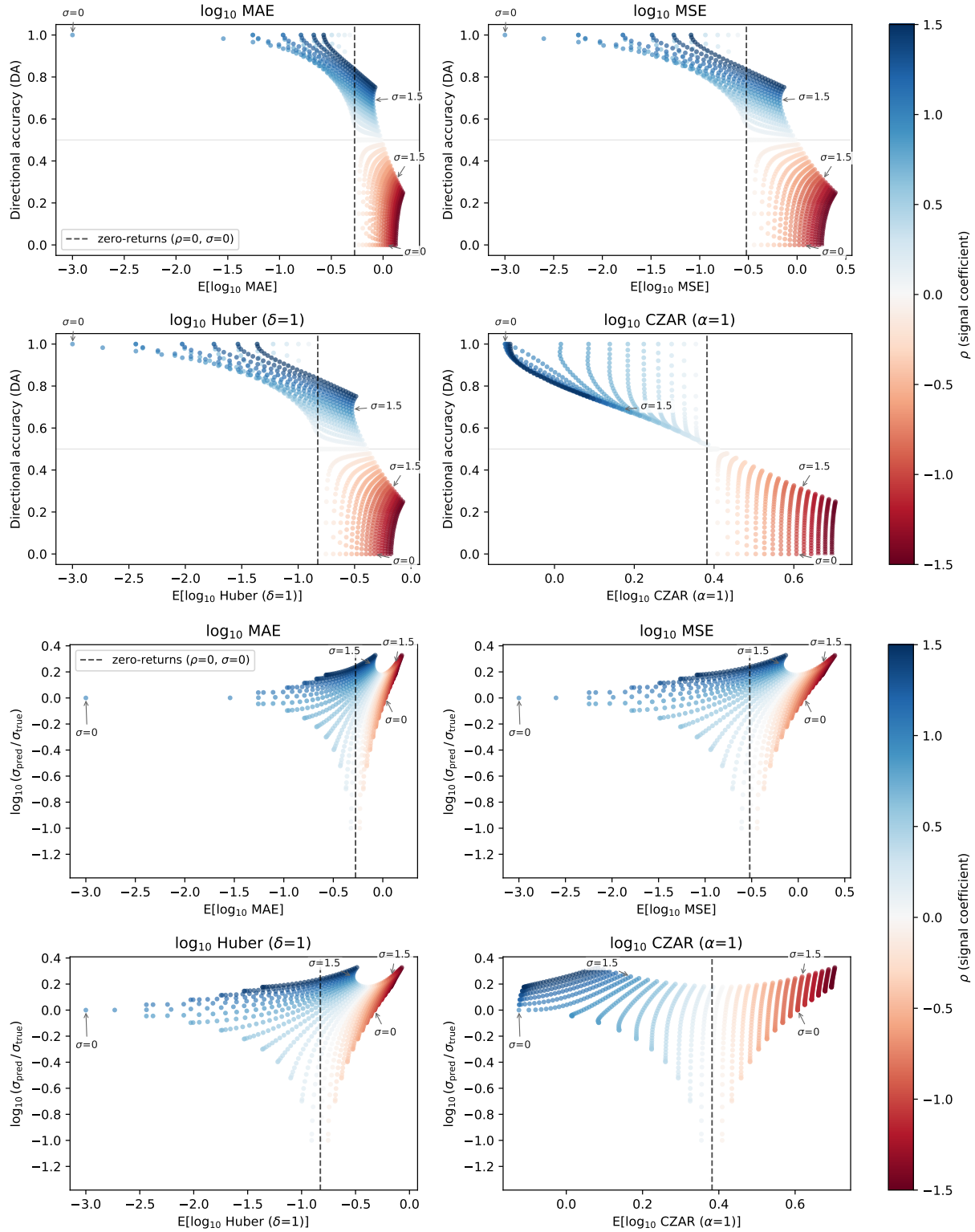


Figure 9: Sample-averaged $\log_{10}$ -loss vs. prediction-quality metric for the Gaussian linear model, swept over $\rho \in [-1.5, 1.5]$ and $\sigma_N \in [0, 1.5]$. Each point is one (ρ, σ_N) pair, with color encoding ρ (blue for positive, red for negative); a small floor $\epsilon = 10^{-3}$ is added before taking the log to avoid divergence at zero residuals. The floor is orders of magnitude below the zero-returns baseline of every loss shown and matters only for near-perfect predictors ($\rho \approx 1$, $\sigma_N \approx 0$), so the position of the dashed line and the orderings discussed in the text are unaffected by its value. The dashed vertical line in each panel marks the log-loss of the zero-returns predictor ($\rho = 0$, $\sigma_N = 0$); points to its left beat the zero-returns baseline. Columns compare MAE, MSE, Huber ($\delta = 1$) and CZAR ($\alpha = 1$ with the correlated defaults $\beta^*(\alpha)$, $C^*(\alpha)$, and $\tau = 0.5$). Top: directional accuracy (DA) on the vertical axis. Bottom: log aspect ratio $\log_{10}(\sigma_{\text{pred}}/\sigma_{\text{true}})$ on the vertical axis.

Training loss	DA	DA _{IQR}	DA _{1σ}	log ₁₀ AR	R _P	IC	Sharpe _{naive}
<i>BTC 15-minute</i>							
L1	0.510	0.487	0.473	−0.93	0.019	0.018	−4.97
L2	0.514	0.508	0.499	−1.18	0.016	0.019	0.96
CZAR ($\alpha = 0.005$)	0.531	0.523	0.537	−0.46	0.046	0.034	5.62
CZAR ($\alpha = 0.01$)	0.522	0.504	0.501	−0.44	0.017	0.033	−0.95
CZAR ($\alpha = 0.05$)	0.523	0.525	0.534	−0.59	0.033	0.050	3.95
CZAR ($\alpha = 0.1$)	0.513	0.514	0.511	−0.58	0.025	0.019	1.15
CZAR ($\alpha = 0.5$)	0.521	0.523	0.532	−0.71	0.033	0.029	4.76
CZAR ($\alpha = 1$)	0.514	0.514	0.527	−0.82	0.055	0.032	5.08
CZAR (tuned, $\alpha \approx 0.054$)	0.524	0.519	0.561	−0.60	0.039	0.056	7.09
<i>BTC 1-hour</i>							
L1	0.518	0.505	0.473	−0.96	0.019	0.053	−1.09
L2	0.517	0.513	0.485	−1.02	0.001	0.021	1.29
CZAR ($\alpha = 0.005$)	0.521	0.526	0.538	−0.41	0.038	0.046	2.67
CZAR ($\alpha = 0.01$)	0.526	0.528	0.524	−0.54	0.034	0.065	2.38
CZAR ($\alpha = 0.05$)	0.516	0.522	0.531	−0.49	0.034	0.058	2.86
CZAR ($\alpha = 0.1$)	0.526	0.546	0.550	−0.55	0.047	0.072	5.96
CZAR ($\alpha = 0.5$)	0.515	0.521	0.520	−0.71	0.044	0.059	3.89
CZAR ($\alpha = 1$)	0.512	0.523	0.529	−0.62	0.041	0.044	3.21
CZAR (tuned, $\alpha \approx 0.0045$)	0.519	0.525	0.520	−0.45	0.031	0.052	3.00

Table 2: Held-out metrics for LightGBM models trained with each loss on BTC log-returns, at the 15-minute (top) and 1-hour (bottom) horizons. DA_{1 σ} and DA_{IQR} denote directional accuracy restricted to truths with $|z| > 1\sigma$ and to truths outside the interquartile range, respectively. The symmetric losses L1 and L2 shrink their predictions well below the truth (log₁₀ AR ≈ -0.9 to -1.2 , i.e. roughly an order of magnitude too small) and their directional accuracy on large moves sits at or below the 50% chance line. Every CZAR variant keeps log₁₀ AR closer to zero, and often raise DA on large moves above chance, lift R_P and IC several-fold over the symmetric baselines, and generally improve the naive Sharpe. Full-sample DA differences are smaller and often within sampling error. Binomial standard errors on the 2000-candle test window are ≈ 1.1 percentage points on the full-sample DA, and ≈ 2 and ≈ 1.6 points on DA_{1 σ} and DA_{IQR}, which use only the roughly one-third and one-half of truths in the tails. The naive Sharpe ratios exclude costs (i.e. are not comparable to live trading) and are annualized from test windows of ≈ 21 days (15-minute) and ≈ 83 days (1-hour).

5 Discussion and conclusions

We have introduced CZAR, a composite zero-agnostic loss function for log-return prediction, motivated by a simple but consequential failure of standard regression losses: because the conditional mean of returns sits close to zero, symmetric losses make the constant zero forecast a near-optimal solution, an attractor during training and a misleading benchmark during evaluation. From an analysis of Gaussian linear prediction models we derived five requirements for an adequate returns loss (§2) and constructed a piecewise quadratic loss satisfying all of them (§3). CZAR is provably convex in the prediction at fixed true value, has a closed-form gradient and Hessian suitable for gradient-boosted libraries, and its four hyperparameters reduce in practice to a single choice of α through the correlated defaults (β^* and C^*) and the fixed hinge width (τ). In idealized tests its log-loss breakeven directional accuracy tracks the 50% chance line where all symmetric monotonic losses share a sharply-increasing universal envelope, an advantage that survives heavy-tailed truth distributions. The correlated defaults $\beta^*(\alpha)$ and $C^*(\alpha)$ were calibrated against the Gaussian linear testbed; under heavy-tailed truths the log-loss breakeven degrades mildly, and recalibrating against the target distribution should recover much of the gap (§4.1).

In a LightGBM experiment on intraday (15-minute and 1-hour) BTC log-returns, CZAR-trained models reduce the prediction shrinkage exhibited by the L1 and L2 baselines and improve the naive (cost-excluded) Sharpe ratio and the directional accuracy on large-magnitude returns (§4). The loss floor is invisible to per-sample gradients, so the training-time pressure away from the zero attractor rests entirely on the base asymmetry; the loss floor acts only through sample-averaged validation and ranking losses.

Beyond its role as a training objective, the sample-averaged CZAR log-loss is a candidate criterion for ranking or scoring forecasters. Two properties from §4.1 support this. First, across the (ρ, σ_N) sweep the positive-signal models collapse onto a single DA–loss track. Models with the same directional accuracy receive essentially the same CZAR loss regardless of how that accuracy is composed from signal and noise, so ranking by mean CZAR loss approximates ranking by directional skill rather than by residual magnitude. Second, the loss floor pins the loss of near-zero predictors to at least C , so trivial forecasters cannot lead a ranking, while wrong-direction forecasters remain penalized through the Region A asymmetry. These properties matter wherever sample-averaged losses carry decision weight, such as in conventional pipelines, for hyperparameter tuning, early stopping, and model selection (where selecting on a symmetric validation loss can silently reintroduce the zero-returns attractor even when training uses CZAR), and in decentralized inference networks such as Allora (Kruijssen et al., 2024), where topic-level losses determine the relative rewards of competing workers and

a symmetric criterion would actively subsidize zero-like inference strategies.

The evaluation results also carry a methodological warning that is independent of CZAR itself. Symmetric losses conflate two distinct properties of a forecaster, i.e. directional skill and prediction variance. A model can reduce its MAE or MSE either by becoming more accurate or simply by shrinking the magnitude of its predictions toward the unconditional mean; the L1- and L2-trained models in Table 2, whose predictions are shrunk roughly an order of magnitude below the truth while their directional accuracy on large moves sits at or below chance, are the practical case of a low-loss but directionally uninformative forecaster. Evaluation methodology should therefore treat the aspect ratio $\sigma_{\hat{y}}/\sigma_y$ as a first-class diagnostic alongside any loss-based leaderboard. It should also recognize the averaging convention as part of the loss design, because linear and log averaging induce different breakeven behavior, and asymmetries that are safe under one can become pathological under the other (§2). Loss-based rankings constructed from symmetric criteria, common in forecasting competitions and production model selection alike, will systematically favor shrunken or degenerate predictors over noisy but directionally informative ones.

Tests of the CZAR loss in this work are naturally non-exhaustive. However, the central message is nevertheless robust across both the idealized and the trained experiments: when the conditional mean of the target is near zero, the loss function is not a neutral implementation detail but the decisive design choice. CZAR offers a convex, trainable, and evaluation-consistent loss that removes the zero-returns attractor by construction, at the cost of a characterizable forecast bias. We argue this compromise is favorable wherever directional information is the economically relevant output.

References

- Armstrong, J. S. & Collopy, F. 1992, *International Journal of Forecasting*, 8, 69. [https://doi.org/10.1016/0169-2070\(92\)90008-W](https://doi.org/10.1016/0169-2070(92)90008-W)
- Banerjee, A., Guo, X., & Wang, H. 2005, *IEEE Transactions on Information Theory*, 51, 2664. <https://doi.org/10.1109/TIT.2005.850145>
- Blattberg, R. C. & Gonedes, N. J. 1974, *The Journal of Business*, 47, 244. <https://doi.org/10.1086/295634>
- Chen, L., Pelger, M., & Zhu, J. 2024, *Management Science*, 70, 714. <https://doi.org/10.1287/mnsc.2023.4695>
- Chen, T. & Guestrin, C. 2016, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. <https://doi.org/10.1145/2939672.2939785>
- Christoffersen, P. F. & Diebold, F. X. 1997, *Econometric Theory*, 13, 808. <https://doi.org/10.1017/S0266466600006277>
- Cont, R. 2001, *Quantitative Finance*, 1, 223. <https://doi.org/10.1080/713665670>
- Dessain, J. 2023, *Advances in Artificial Intelligence and Machine Learning*, 3, 1640. <https://doi.org/10.54364/aaiml.2023.1193>
- Elliott, G., Komunjer, I., & Timmermann, A. 2005, *The Review of Economic Studies*, 72, 1107. <https://doi.org/10.1111/0034-6527.00363>
- Elmachtoub, A. N. & Grigas, P. 2022, *Management Science*, 68, 9. <https://doi.org/10.1287/mnsc.2020.3922>
- Fleming, P. J. & Wallace, J. J. 1986, *Communications of the ACM*, 29, 218. <https://doi.org/10.1145/5666.5673>
- Gneiting, T. 2011, *Journal of the American Statistical Association*, 106, 746. <https://doi.org/10.1198/jasa.2011.r10138>
- Gopikrishnan, P., Plerou, V., Nunes Amaral, L. A., Meyer, M., & Stanley, H. E. 1999, *Phys. Rev. E*, 60, 5305. <https://doi.org/10.1103/PhysRevE.60.5305>
- Granger, C. W. J. 1969, *Journal of the Operational Research Society*, 20, 199. <https://doi.org/10.1057/jors.1969.52>
- Granger, C. W. J. & Pesaran, M. H. 2000, *Journal of Forecasting*, 19, 537. [https://doi.org/10.1002/1099-131X\(200012\)19:7<537::AID-FOR769>3.0.CO;2-G](https://doi.org/10.1002/1099-131X(200012)19:7<537::AID-FOR769>3.0.CO;2-G)
- Gu, S., Kelly, B., & Xiu, D. 2020, *The Review of Financial Studies*, 33, 2223. <https://doi.org/10.1093/rfs/hhaa009>
- Huber, P. J. 1964, *Annals of Mathematical Statistics*, 35, 73. <https://doi.org/10.1214/aoms/1177703732>
- Hyndman, R. J. & Koehler, A. B. 2006, *International Journal of Forecasting*, 22, 679. <https://doi.org/10.1016/j.ijforecast.2006.03.001>
- Kandel, S. & Stambaugh, R. F. 1996, *Journal of Finance*, 51, 385. <https://doi.org/10.1111/j.1540-6261.1996.tb02689.x>
- Ke, G., Meng, Q., Finley, T., et al. 2017, in *Advances in Neural Information Processing Systems*, ed. I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett, Vol. 30 (Curran Associates, Inc.), 3146–3154. <https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html>
- Koenker, R. & Bassett, G. 1978, *Econometrica*, 46, 33. <https://doi.org/10.2307/1913643>
- Kruijssen, J. M. D., Emmons, N., Peluso, K., et al. 2024, *Allora Decentralized Intelligence*, 1, 1. <https://doi.org/10.70235/allora.0x10001>
- Leippold, M., Wang, Q., & Zhou, W. 2022, *Journal of Financial Economics*, 145, 64. <https://doi.org/10.1016/j.jfineco.2021.08.017>
- Leitch, G. & Tanner, J. E. 1991, *American Economic Review*, 81, 580. <https://www.jstor.org/stable/2006520>
- Lo, A. W. 2002, *Financial Analysts Journal*, 58, 36. <https://doi.org/10.2469/faj.v58.n4.2453>

- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. 2020, International Journal of Forecasting, 36, 54, m4 Competition. <https://doi.org/10.1016/j.ijforecast.2019.04.014>
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. 2022, International Journal of Forecasting, 38, 1346, special Issue: M5 competition. <https://doi.org/10.1016/j.ijforecast.2021.11.013>
- Michańków, J., Sakowski, P., & Ślepaczuk, R. 2024, Journal of Computational Science, 81, 102375. <https://doi.org/10.1016/j.jocs.2024.102375>
- Moody, J. & Saffell, M. 2001, IEEE Transactions on Neural Networks, 12, 875. <https://doi.org/10.1109/72.935097>
- Newey, W. K. & Powell, J. L. 1987, Econometrica, 55, 819. <https://doi.org/10.2307/1911031>
- Patton, A. J. & Timmermann, A. 2007, Journal of Econometrics, 140, 884. <https://doi.org/10.1016/j.jeconom.2006.07.018>
- Pesaran, M. H. & Timmermann, A. 1992, Journal of Business & Economic Statistics, 10, 561. <https://doi.org/10.1080/07350015.1992.10509922>
- Zellner, A. 1986, Journal of the American Statistical Association, 81, 446. <https://doi.org/10.1080/01621459.1986.10478289>
- Zhang, Z., Zohren, S., & Roberts, S. 2020, Journal of Financial Data Science, 2, 8. <https://doi.org/10.3905/jfds.2020.1.042>

Appendices

A Reference implementation

The following Python listing implements the base CZAR loss (§3), gradient, and Hessian using NumPy. All functions operate on batches (`y_true`, `y_pred`, and `std` may be scalars or arrays of matching shape). The listing includes two numerical guards that are absent from the mathematical formulation of §3.4. The loss-floor normalization is protected against division by zero (`safe_norm`), and the loss-floor term is clipped at 10^{-12} to guard against floating-point underflow when logarithms of the total loss are taken for evaluation; neither affects the loss away from degenerate parameter values. The Hessian clip at 10^{-6} used in the LightGBM experiments (§4.2) is applied in the training wrapper, not in this reference implementation.

```

1  import numpy as np
2
3  ALPHA = 1.0    # quadratic regularization strength
4  TAU    = 0.5    # hinge smoothing width
5
6
7  def _correlated_beta(alpha):
8      # Two-term power-law fit to optimal (alpha, beta*) pairs from the breakeven
9      # DA heatmap; minimizes the DA required to beat a zero-returns predictor.
10     return 4.2 * alpha**0.56 + 27.0 * alpha**2.17
11
12
13  def _correlated_C(alpha):
14      # Asymmetric Hill-bell fit (with linear correction) to optimal C per alpha
15      # at fixed tau; keeps the log-loss breakeven DA at or above 50% across
16      # noise scales, and satisfies C*(0) = 0.
17      A, t, p, k, R, m = 7.9, 0.00459, 0.657, 0.684, 2.25, -0.218
18      h = alpha**p / (t**p + alpha**p)
19      return A * h**k * (1 - h) + (R + m * alpha) * h
20
21
22  def czar_loss(y_true, y_pred, std, mean=0,
23               alpha=ALPHA, beta=None, C=None, tau=TAU):
24      if beta is None:
25          beta = _correlated_beta(alpha)
26      if C is None:
27          C = _correlated_C(alpha)
28      # Standardize to zero mean, unit variance.
29      z = (y_true - mean) / std
30      z_hat = (y_pred - mean) / std
31
32      # Unsigned prediction error and true-value magnitude.
33      dz = np.abs(z_hat - z)
34      a = np.abs(z)

```

```

35
36     # s is the sign of the true value (convention: +1 when z = 0).
37     # u = s * z_hat projects the prediction onto the direction of the truth:
38     #   u > a means the prediction overshoots (Region B),
39     #   u <= a means it undershoots or is in the wrong direction (Region A).
40     s = np.where(z == 0, 1.0, np.sign(z))
41     u = s * z_hat
42
43     # Adaptive discount: b_eff -> 1 as a -> 0 (both regions become quadratic),
44     # b_eff -> 0 as a -> inf (Region B vanishes, Region A becomes MAE-like).
45     b_eff = 1.0 / (1.0 + beta * a)
46
47     # Region A: MAE + quadratic. Region B: discounted quadratic only.
48     L_A = (1 - b_eff) * dz + 0.5 * alpha * dz**2
49     L_B = b_eff * 0.5 * alpha * dz**2
50
51     # --- Loss floor ---
52     # L0 is the base loss when z_hat = 0 (always in Region A since u = 0 <= a).
53     # deficit = C - L0: positive when the zero-prediction loss is below the
54     # target C, meaning additional floor loss is required.
55     L0 = (1 - b_eff) * a + 0.5 * alpha * a**2
56     deficit = C - L0
57
58     # hinge(x, tau) ~ max(x, 0) for small tau (smooth rectification).
59     # Normalized by hinge(C, tau) so that L_floor = C when deficit = C
60     # (i.e. at a = 0, where L0 = 0) and L_floor -> 0 when L0 >> C.
61     hinge = 0.5 * (deficit + np.sqrt(deficit**2 + tau**2))
62     norm = 0.5 * (C + np.sqrt(C**2 + tau**2))
63     safe_norm = np.where(norm > 0, norm, 1.0)
64     L_floor = np.where(C > 0, np.maximum(C * hinge / safe_norm, 1e-12), 0.0)
65
66     return np.where(u > a, L_B, L_A) + L_floor
67
68
69 def czar_gradient(y_true, y_pred, std, mean=0,
70                  alpha=ALPHA, beta=None):
71     if beta is None:
72         beta = _correlated_beta(alpha)
73     # L_floor depends only on z (not z_hat), so it contributes no gradient.
74     z = (y_true - mean) / std
75     z_hat = (y_pred - mean) / std
76     err = z_hat - z # signed error
77     dz = np.abs(err)
78     a = np.abs(z)
79     s = np.where(z == 0, 1.0, np.sign(z))
80     u = s * z_hat
81
82     # s_err = sign of the error, used to direct the gradient.
83     # Set to 0 exactly at z_hat = z so the gradient is 0 at the minimum,
84     # consistent with Region B where G_B -> 0 as dz -> 0.
85     s_err = np.where(err == 0, 0.0, np.sign(err))
86     b_eff = 1.0 / (1.0 + beta * a)
87
88     # Region A gradient: magnitude (1 - b_eff) + alpha*dz, directed by s_err.
89     # Approaches -(1-b_eff)/std as z_hat -> z from below (the gradient step),
90     # then 0 from Region B, concentrating reward near the true value.
91     G_A = s_err * ((1 - b_eff) + alpha * dz)
92     G_B = s_err * b_eff * alpha * dz
93
94     return np.where(u > a, G_B, G_A) / std
95
96
97 def czar_hessian(y_true, y_pred, std, mean=0,
98                  alpha=ALPHA, beta=None):
99     if beta is None:
100         beta = _correlated_beta(alpha)

```

```

101     z      = (y_true - mean) / std
102     z_hat  = (y_pred - mean) / std
103     a      = np.abs(z)
104     s      = np.where(z == 0, 1.0, np.sign(z))
105     u      = s * z_hat
106     b_eff  = 1.0 / (1.0 + beta * a)
107
108     # Region A Hessian: the MAE term (1-b_eff)*dz has zero curvature,
109     # so only the quadratic term contributes -> constant alpha.
110     # Region B Hessian: b_eff * alpha, which shrinks toward 0 for large |z|.
111     # Both are non-negative, confirming convexity in y_pred at fixed y_true.
112     H_A = alpha
113     H_B = b_eff * alpha
114
115     return np.where(u > a, H_B, H_A) / std**2

```

B Breakeven directional accuracy

Prediction model and directional accuracy

We model predicted returns as a linear combination of signal and noise:

$$\hat{y} = \rho y + \sigma_N \xi, \quad y \sim \mathcal{N}(0, \sigma^2), \quad \xi \sim \mathcal{N}(0, 1) \text{ independent}, \quad (\text{B1})$$

where $\rho \in \mathbb{R}$ is the signal coefficient and $\sigma_N \geq 0$ is the noise scale. The directional accuracy (DA) is the probability of a correct-sign prediction:

$$\text{DA}(\rho, \sigma_N) = P(\text{sign}(\hat{y}) = \text{sign}(y)). \quad (\text{B2})$$

The pair $(y, \hat{y})$ is jointly bivariate Gaussian with zero mean, $\text{Cov}(y, \hat{y}) = \rho\sigma^2$, and correlation

$$r = \frac{\rho\sigma}{\sqrt{\rho^2\sigma^2 + \sigma_N^2}}. \quad (\text{B3})$$

The sign-concordance formula for zero-mean bivariate Gaussians gives $P(X > 0, Y > 0) = \frac{1}{4} + \frac{\arcsin r}{2\pi}$, so

$$\text{DA} = 2 P(y > 0, \hat{y} > 0) = \frac{1}{2} + \frac{\arcsin r}{\pi}. \quad (\text{B4})$$

Applying the identity $\arcsin(x/\sqrt{x^2+1}) = \arctan(x)$:

$$\text{DA}(\rho, \sigma_N) = \frac{1}{2} + \frac{1}{\pi} \arctan\left(\frac{\rho\sigma}{\sigma_N}\right). \quad (\text{B5})$$

The DA depends only on the signal-to-noise ratio $\text{SNR} = \rho\sigma/\sigma_N$, not on ρ and σ_N separately.

Breakeven conditions

A predictor *breaks even* against a constant-zero forecast when its expected loss equals the baseline loss, either under linear or log averaging:

$$\text{linear: } \mathbb{E}[\mathcal{L}(y, \hat{y})] = \mathbb{E}[\mathcal{L}(y, 0)], \quad (\text{B6})$$

$$\text{log: } \mathbb{E}[\log \mathcal{L}(y, \hat{y})] = \mathbb{E}[\log \mathcal{L}(y, 0)]. \quad (\text{B7})$$

For each noise scale σ_N , the minimum DA satisfying these conditions traces the breakeven curves in Figure 2.

Symmetric losses share a universal breakeven curve

Proposition 1. *For any symmetric loss $\mathcal{L}(y, \hat{y}) = g(|y - \hat{y}|)$ with g strictly increasing and positive, and with $\mathbb{E}[g(|e|)]$ and $\mathbb{E}[|\log g(|e|)|]$ finite for every centered Gaussian e , the linear and log breakeven conditions (B6)–(B7) are equivalent under model (B1), and both reduce to*

$$(1 - \rho)^2\sigma^2 + \sigma_N^2 = \sigma^2. \quad (\text{B8})$$

In particular, the condition does not depend on g .

Proof. The prediction error is $e = \hat{y} - y = (\rho - 1)y + \sigma_N \xi \sim \mathcal{N}(0, s^2)$ with $s^2 = (1 - \rho)^2 \sigma^2 + \sigma_N^2$. Writing $e = sZ$ with $Z \sim \mathcal{N}(0, 1)$, the loss becomes $g(s|Z|)$. Since g is strictly increasing, both

$$h_{\text{lin}}(s) = \mathbb{E}[g(s|Z|)] \quad \text{and} \quad h_{\text{log}}(s) = \mathbb{E}[\log g(s|Z|)]$$

are strictly monotone in s . Each breakeven condition therefore reduces to $s = \sigma$, which is (B8). $\square$

The function g cancels entirely. MSE ($g(r) = r^2$), MAE ($g(r) = r$), Huber loss, and any other symmetric loss meeting these conditions therefore all satisfy the same breakeven condition (B8). Solving Equation (B8) for the minimum signal coefficient gives $\rho_{\min} = 1 - \sqrt{1 - (\sigma_N/\sigma)^2}$, which substituted into Equation (B5) yields the universal breakeven curve quoted in §2, $\text{DA} = 0.5 + \arctan(\rho_{\min}\sigma/\sigma_N)/\pi$, valid for both linear and log averaging simultaneously.

Asymmetric losses can decouple the two breakevens

Proposition 1 establishes a hard lower bound: no symmetric loss function can achieve a breakeven curve below (B8) in either linear or log averaging. For a loss to improve on this bound it must be asymmetric, i.e. it must treat $\text{sign}(y) = \text{sign}(\hat{y})$ differently from $\text{sign}(y) \neq \text{sign}(\hat{y})$.

For an asymmetric loss, the distribution of $\mathcal{L}(y, \hat{y})$ changes shape (not only scale) as ρ varies, because the fraction of samples falling in the correct- and wrong-direction regions depends on ρ . The log and linear expectations then weight the distribution differently, and the two breakeven conditions are no longer equivalent.

CZAR exploits exactly this freedom by making its loss depend separately on $\text{sign}(z)$ and $\text{sign}(\hat{z})$, with different functional forms in each region. As visible in Figure 7, CZAR's log-loss breakeven lies substantially below the symmetric bound (B8), while its linear-loss breakeven tracks that bound more closely. The separation between the two breakevens, achievable only through asymmetry, is the central property that motivates the CZAR design.

C Pseudo-Huber smoothing

The core formulation of §3 uses a plain MAE distance in Region A. This produces a step discontinuity in the gradient at $\hat{z} = z$ of magnitude $(1 - \beta_{\text{eff}})/\sigma$ (§3.3). Setting $\varepsilon > 0$ replaces the MAE term with a pseudo-Huber distance, smoothing the step into a continuous zero-crossing. This appendix describes the modifications required.

Modified quantities

Introduce a smoothing scale $\varepsilon_{\text{eff}}(|z|) = \varepsilon|z|$ proportional to the true-value magnitude, and define the pseudo-Huber effective distance

$$\Delta z_{\text{eff}} = \sqrt{1 + \varepsilon^2} \left(\sqrt{\Delta z^2 + \varepsilon_{\text{eff}}^2} - \varepsilon_{\text{eff}} \right). \quad (\text{C1})$$

When $\varepsilon = 0$, $\Delta z_{\text{eff}} = \Delta z$ (MAE). For small errors ($\Delta z \ll \varepsilon_{\text{eff}}$), $\Delta z_{\text{eff}} \approx \sqrt{1 + \varepsilon^2} \Delta z^2 / (2\varepsilon|z|)$ (quadratic); for large errors ($\Delta z \gg \varepsilon_{\text{eff}}$), $\Delta z_{\text{eff}} \approx \sqrt{1 + \varepsilon^2} \Delta z$ (linear, with boosted slope). Because $\Delta z_{\text{eff}} < \Delta z$ for small errors at fixed $\varepsilon > 0$, an amplitude boost factor

$$b(|z|) = 1 + 2\varepsilon(1 - \beta_{\text{eff}}(|z|)) \geq 1 \quad (\text{C2})$$

is introduced to compensate. Without it, increasing ε would reduce the Region A loss for moderate errors relative to the $\varepsilon = 0$ case.

Modified base loss

Replace $(1 - \beta_{\text{eff}}) \Delta z$ in Region A of Equation (3) with $b(1 - \beta_{\text{eff}}) \Delta z_{\text{eff}}$:

$$\mathcal{L}_{\text{base}}^{(\varepsilon)}(z, \hat{z}) = \begin{cases} b(1 - \beta_{\text{eff}}) \Delta z_{\text{eff}} + \frac{\alpha}{2} \Delta z^2 & (\text{Region A}), \\ \beta_{\text{eff}} \frac{\alpha}{2} \Delta z^2 & (\text{Region B}). \end{cases} \quad (\text{C3})$$

Region B is unchanged.

Modified gradient and Hessian

The Region A gradient becomes

$$\left. \frac{\partial \mathcal{L}_{\text{base}}^{(\varepsilon)}}{\partial \hat{y}} \right|_{\text{Region A}} = \frac{s \Delta}{\sigma} \left[b(1 - \beta_{\text{eff}}) \sqrt{1 + \varepsilon^2} \frac{\Delta z}{\sqrt{\Delta z^2 + \varepsilon_{\text{eff}}^2}} + \alpha \Delta z \right]. \quad (\text{C4})$$

The factor $\Delta z / \sqrt{\Delta z^2 + \varepsilon_{\text{eff}}^2} \rightarrow 0$ as $\Delta z \rightarrow 0$ (for $\varepsilon > 0$), so the gradient vanishes continuously at $\hat{z} = z$. The Region A Hessian gains a curvature term:

$$\left. \frac{\partial^2 \mathcal{L}_{\text{base}}^{(\varepsilon)}}{\partial \hat{y}^2} \right|_{\text{Region A}} = \frac{1}{\sigma^2} \left[b(1 - \beta_{\text{eff}}) \sqrt{1 + \varepsilon^2} \frac{\varepsilon_{\text{eff}}^2}{(\Delta z^2 + \varepsilon_{\text{eff}}^2)^{3/2}} + \alpha \right]. \quad (\text{C5})$$

This is maximized at $\Delta z = 0$ (where it equals $\frac{1}{\sigma^2} [b(1 - \beta_{\text{eff}}) \sqrt{1 + \varepsilon^2} / \varepsilon_{\text{eff}} + \alpha]$) and decays to α / σ^2 as $\Delta z \rightarrow \infty$, recovering the constant Hessian of the $\varepsilon = 0$ case. The Region B gradient and Hessian are unchanged.

Modified loss floor

At $\hat{z} = 0$, the pseudo-Huber effective distance evaluates to

$$\Delta z_{0,\text{eff}} = |z| (1 + \varepsilon^2 - \varepsilon \sqrt{1 + \varepsilon^2}), \quad (\text{C6})$$

obtained by substituting $\Delta z = |z|$ and $\varepsilon_{\text{eff}} = \varepsilon |z|$ into Equation (C1). The zero-prediction base loss in Equation (6) becomes

$$\mathcal{L}_0^{(\varepsilon)}(z) = b(1 - \beta_{\text{eff}}) \Delta z_{0,\text{eff}} + \frac{\alpha}{2} z^2, \quad (\text{C7})$$

which replaces $(1 - \beta_{\text{eff}}) |z|$ from the $\varepsilon = 0$ case. The loss-floor formulas (7)–(8) are otherwise unchanged, with the deficit redefined as $d(z) = C - \mathcal{L}_0^{(\varepsilon)}(z)$.

Convexity

The Region A Hessian (C5) satisfies $H_A \geq \alpha > 0$ and the Region B Hessian remains $\beta_{\text{eff}} \alpha > 0$ (since $\alpha > 0$ and $\beta_{\text{eff}} \in (0, 1]$). For $\varepsilon > 0$ the gradient now vanishes continuously from both sides at $\hat{z} = z$ (no step discontinuity), and (writing $s = \text{sign}(z)$) has sign $-s$ on the Region A side of the region boundary and $+s$ on the Region B side (i.e. it is negative for $\hat{z} < z$ and positive for $\hat{z} > z$ when $z > 0$, with the sides reversed when $z < 0$). Convexity in $\hat{y}$ at fixed y is preserved for all $\varepsilon \geq 0$. On each side of the region boundary the derivative is continuous and non-decreasing in $\hat{y}$, and any residual one-sided jump at $\hat{z} = z$ (≥ 0 at $\varepsilon = 0$, exactly 0 for $\varepsilon > 0$) is upward.

D Parameter optimization

The prediction model and breakeven directional accuracy are defined in §2. Here we apply them to the CZAR loss to determine the correlated defaults $\beta^*(\alpha)$ and $C^*(\alpha)$, along with the constant hinge width τ adopted across the working range.

The floor parameters C and τ are both optimized by the requirement that the log-loss breakeven DA stays at or above 50% at all noise scales (see §2). A breakeven below 50% would mean a model that predicts consistently in the wrong direction achieves a lower loss than a zero-returns predictor. However, C and τ act on the floor term in a degenerate way and cannot be optimized jointly from the breakeven condition alone. We therefore fix τ first from an independent criterion, and then optimize C at the chosen τ .

Optimizing β

The breakeven DA is evaluated numerically at noise scale $\sigma_N = 1$ over $N = 50,000$ samples from the Gaussian linear model ($y \sim \mathcal{N}(0, 1)$, $\hat{y} = \rho y + \sigma_N \xi$). Figure D1 shows the resulting breakeven DA as a joint function of α and β . For each α , the value $\beta^*(\alpha)$ that minimizes the breakeven DA is traced by the black curve. The loss surface is shallow near this optimum, i.e. neighboring (α, β) pairs achieve nearly identical breakeven DA, so the default is robust to the exact gridding choice for β .

The two-term power-law fit (Equation (10), red dashed curve) closely tracks the numerical optimum across more than three decades of α . We restrict the fit to $\alpha \leq 0.7$. Beyond this the breakeven basin in β becomes very shallow and the optimum drifts toward arbitrarily large values, so a single $\beta^*(\alpha)$ ceases to be a meaningful target and any sufficiently large β achieves consistently low breakeven DA. For practical use we recommend $\alpha \leq 1$, with β^* taken from Equation (10) (a mild extrapolation in $(0.7, 1]$). What is also clear from the figure is that the breakeven directional accuracy converges for $\alpha \lesssim 0.005$, and therefore cannot be improved by using arbitrarily small α values.

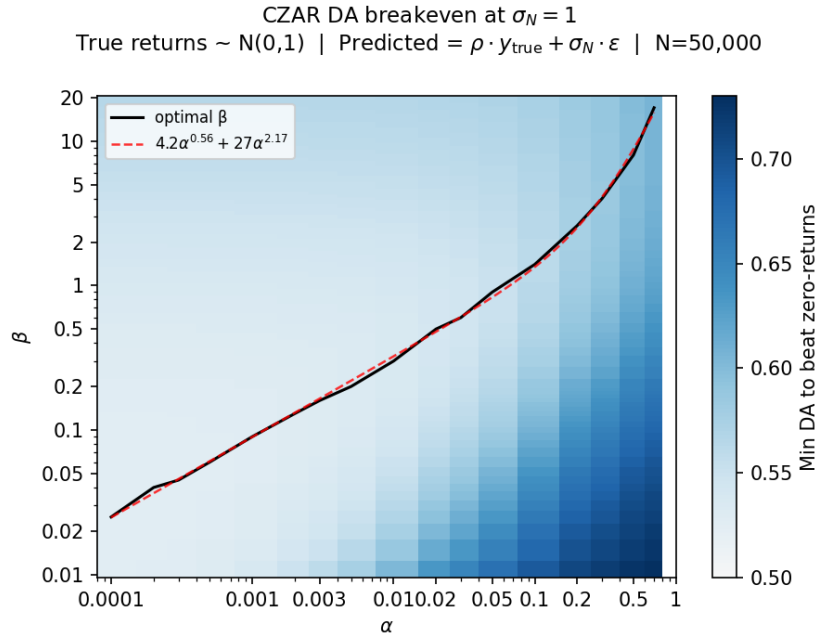


Figure D1: CZAR breakeven directional accuracy at $\sigma_N = 1$ as a function of α and β (both axes log-scale). Color indicates the minimum DA required to beat a zero-returns predictor under linear loss averaging; lower (redder) is better. The black curve marks the optimal $\beta^*(\alpha)$ and the red dashed curve the two-term power-law fit from Equation (10). The shallow basin around the optimum shows that the default is robust to moderate departures from β^* .

Optimizing τ

We fix τ by requiring that the expected log-loss $E_y[\log_{10} \mathcal{L}(y, \hat{y})]$, integrated over $y \sim \mathcal{N}(0,1)$, is monotonic in $\hat{z}$ for $\hat{z} \geq 0$ (and by symmetry for $\hat{z} \leq 0$). A non-monotonic profile would imply that some non-trivial predicted return is preferred by the loss in expectation, biasing the optimizer toward that scale.

Figure D2 sweeps $\tau \in [0.1, 1.0]$ at fixed $C = 2$ and $\beta = \beta^*(\alpha)$ for three representative α values. For each α there is a threshold $\tau_{\min}(\alpha)$ below which the curves develop shallow minima near $\hat{y} \approx \pm 2$ (dotted lines, excluded); for $\tau \geq \tau_{\min}(\alpha)$ the curves are monotonic (solid). The threshold grows with α . At $\alpha = 0.1$ values as low as $\tau \approx 0.1$ remain monotonic, whereas at $\alpha = 1$ we require $\tau \gtrsim 0.5$. We adopt $\tau = 0.5$ as a constant default, i.e. the smallest value that remains monotonic across the working range $\alpha \leq 1$. Smaller τ could in principle be used at small α , but the expected-loss profile is increasingly insensitive to τ there, so a single constant is preferred for simplicity.

Optimizing $C(\alpha)$

With τ fixed, C is determined by requiring the log-loss breakeven DA to equal 50% at a reference noise scale $\sigma_N = 0.1$. Operationally, for each α we evaluate the breakeven DA on a small σ_N window around 0.1, fit a line to those points, and bisect on C until the linear fit at $\sigma_N = 0.1$ equals 0.5. Targeting the fit rather than a single sample point suppresses binomial sampling noise (the 95% CI for DA = 0.5 at $N = 10^6$ is shown by the dashed gray lines in the left panel of Figure D3). The breakeven curves at the resulting $C^*(\alpha)$ cluster tightly around 0.5 across all σ_N , confirming that this single-point calibration produces a near-flat profile.

The right panel of Figure D3 shows the resulting $C^*(\alpha)$. The points rise from $C^* = 0$ at $\alpha = 0$ to a peak near $\alpha \sim 10^{-2}$, followed by a slow decline at larger α . We fit them with an asymmetric Hill-bell model with a linear correction, which captures both the zero limit and the high- α tail while remaining smooth across the transition. The fitted coefficients give Equation (11), repeated here for convenience:

$$C^*(\alpha) = A h(\alpha)^k [1 - h(\alpha)] + (R + m\alpha) h(\alpha), \quad h(\alpha) = \frac{\alpha^p}{t^p + \alpha^p},$$

with $(A, t, p, k, R, m) = (7.90, 4.59 \times 10^{-3}, 0.657, 0.684, 2.25, -0.218)$.

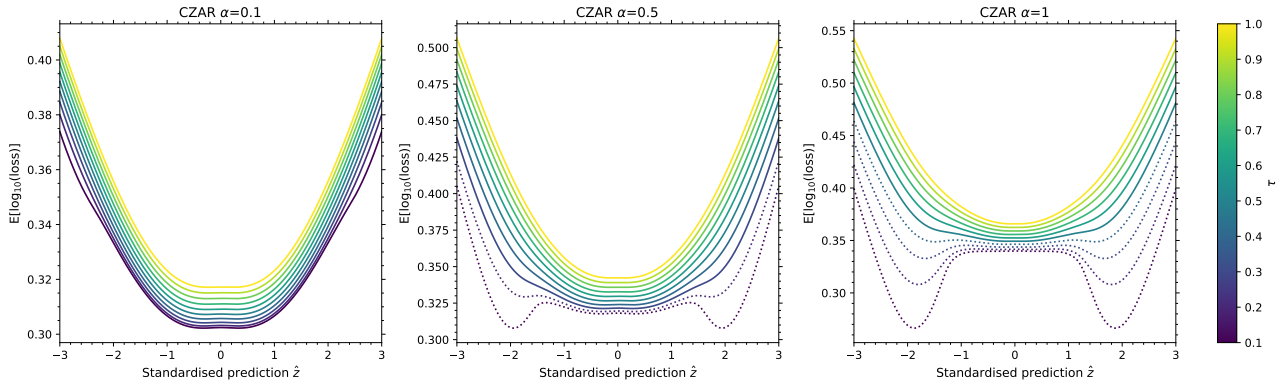


Figure D2: Expected log-loss $E_{y \sim \mathcal{N}(0,1)}[\log_{10} \mathcal{L}(y, \hat{y})]$ as a function of standardized prediction $\hat{z}$ for $\tau \in [0.1, 1.0]$ (color) at three α values, with $C = 2$ and $\beta = \beta^*(\alpha)$. Solid curves are monotonic in $|\hat{z}|$ and define admissible τ values; dotted curves develop spurious minima near $\hat{z} \approx \pm 2$ and are excluded. The smallest admissible τ grows with α ; $\tau = 0.5$ is admissible across $\alpha \leq 1$ and is adopted as the default.

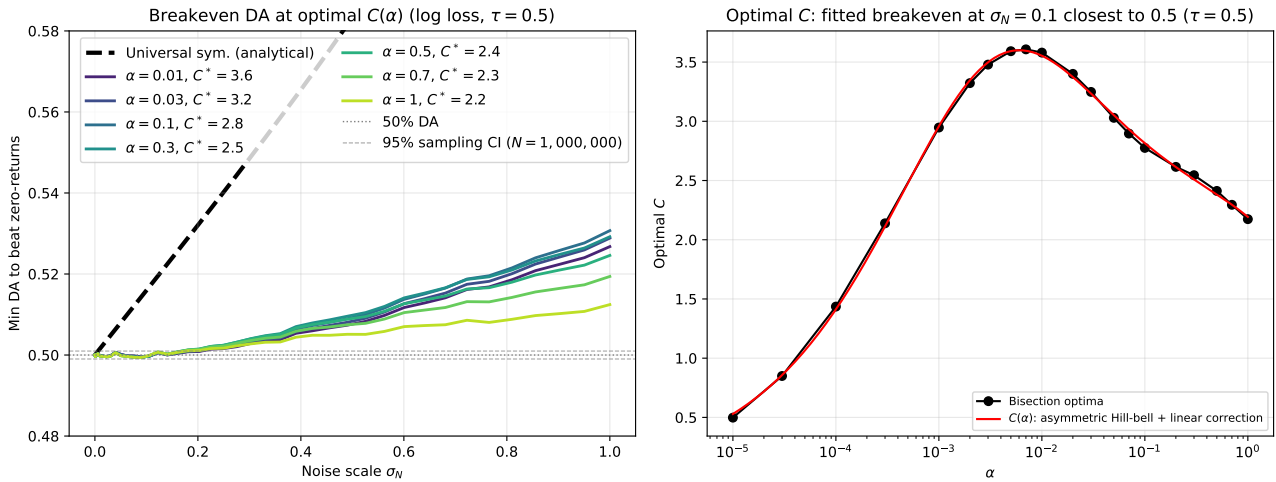


Figure D3: Optimization of the loss floor parameter C . The left panel shows log-loss breakeven DA versus σ_N at the optimized $C^*(\alpha)$ (color), with $\tau = 0.5$ and $\beta = \beta^*(\alpha)$. The universal symmetric (MSE) breakeven (thick dashed black) is shown for reference; the dashed gray lines bound the 95% binomial CI for DA = 0.5 at $N = 10^6$. The right panel shows optimal $C^*(\alpha)$ (black points) and the asymmetric Hill-bell fit of Equation (11) (red line). C^* is selected as the value for which a linear fit to the breakeven DA in a small window around $\sigma_N = 0.1$ equals 0.5.