BayPrAnoMeta: Bayesian Proto-MAML for Few-Shot Industrial Image Anomaly Detection

Anonymous authors
Paper under double-blind review

Abstract

Industrial image anomaly detection is a challenging problem owing to extreme class imbalance and the scarcity of labeled defective samples, particularly in few-shot settings. We propose BayPrAnoMeta, a Bayesian generalization of Proto-MAML for few-shot industrial image anomaly detection. Unlike existing Proto-MAML approaches that rely on deterministic class prototypes and distance based adaptation, BayPrAnoMeta replaces prototypes with task-specific probabilistic normality models and performs inner loop adaptation via a Bayesian posterior predictive likelihood. We model normal support embeddings with a Normal–Inverse–Wishart (NIW) prior, producing a Student-t predictive distribution that enables uncertainty-aware, heavy-tailed anomaly scoring and is essential for robustness in extreme few-shot settings. We further extend BayPrAnoMeta to a federated meta-learning framework with supervised contrastive regularization for heterogeneous industrial clients and prove convergence to stationary points of the resulting nonconvex objective. Experiments on the MVTec AD and VisA datasets show that the proposed method consistently achieves significant AUROC improvements over MAML, Proto-MAML, and PatchCore based methods in few-shot anomaly detection settings.

1 Introduction

Modern manufacturing requires automated anomaly detection even for minor defects, such as scratches, deformations, and contamination, which can significantly hamper product quality and result in expensive operational failures (Bergmann et al., 2019; Ruff et al., 2018). Despite its importance, anomaly detection in an industrial setting is difficult for several reasons. First, the nature of anomalies (rare and visually diverse) within datasets dominated by normal samples makes large-scale supervised training unfeasible (Pang et al., 2021). Moreover, significant domain shifts, which originate from different machines, sensors, and production sites, obstruct the generalization of models (Li et al., 2025). Privacy and data-sharing limitations are frequently a result of legal and intellectual property issues, which, in turn, hinder the aggregation of visual inspection data across different manufacturing facilities (Kairouz et al., 2021). Together, these factors demonstrate why classical supervised learning approaches are inadequate and motivate the development of few-shot, uncertainty-aware anomaly detection methods.

A prominent line of work in industrial anomaly detection focuses on nearest-neighbor and representation-based methods that model normality using only defect-free samples. Approaches such as PatchCore (Roth et al., 2022) and Reverse Distillation (Tien et al., 2023) achieve strong performance in fully unsupervised settings by leveraging powerful pretrained feature extractors. However, such methods rely on fixed representations and distance-based inference, which limits their ability to adapt to new object categories, configurations, or operating conditions when only a small number of labeled samples are available. In practical industrial environments, limited labeled anomalies or configuration-specific feedback are often available, motivating methods that can explicitly adapt to new tasks from few examples. Meta-learning provides a natural framework for this setting by enabling rapid task-specific adaptation (Hospedales et al., 2021). In particular, episodic meta-learning methods such as Prototypical Networks (Snell et al., 2017) and Model-Agnostic Meta-Learning (MAML) (Finn et al., 2017) align well with industrial scenarios where only a few labeled samples are available per object type or configuration. Nevertheless, standard meta-learning approaches are

primarily discriminative and do not provide calibrated uncertainty estimates, which are crucial for reliable decision-making in anomaly detection under extreme data scarcity (Malinin & Gales, 2018).

To address these challenges more rigorously, we introduce BayPrAnoMeta, a Bayesian meta-learning approach for few-shot industrial anomaly detection, and extend it to federated settings with contrastive representation learning. Our main contributions are as follows: ***(i)*** we propose BayPrAnoMeta, a Bayesian extension of Proto-MAML, where support embeddings are modeled using a Normal-Inverse-Wishart (NIW) prior. By marginalizing the latent mean and covariance parameters, the model yields a multivariate Student-t posterior predictive distribution (Murphy, 2012) to support robust heavy-tailed modeling suitable for anomalies (Gordon et al., 2018) and likelihood-based anomaly scoring in extreme few-shot regimes; ***(ii)*** we develop a federated Bayesian meta-learning (McMahan et al., 2017; Chen et al., 2018) formulation of BayPrAnoMeta to account for client heterogeneity and non i.i.d. data distributions, enabling scalable and privacy-preserving training across distributed industrial environments; ***(iii)*** we incorporate supervised contrastive regularization (Khosla et al., 2020) within this federated framework to improve representation quality under client heterogeneity.

2 Related Work

Early research on industrial anomaly detection focuses on unsupervised learning or one-class learning approaches where standard datasets such as MVTec AD (Bergmann et al., 2019) have been used for empirical evaluation. Furthermore, Deep One-Class Classification (Ruff et al., 2018) and subsequent surveys (Pang et al., 2021) focus on core challenges such as class imbalance, heterogeneous defect patterns, and limited labeled anomaly samples. Due to their reliance on large training datasets and their difficulty handling unseen anomaly types, classical anomaly detection methods have limited applicability.

Meta-learning provides a general framework for rapid adaptation under limited supervision. Metric-based approaches such as Prototypical Networks (Snell et al., 2017), Matching Networks (Vinyals et al., 2016), Relation Networks (Sung et al., 2018) learn embedding spaces where few-shot classification can be performed via distance-based inference whereas gradient based methods such as MAML (Finn et al., 2017) and its variants (Reptile, MetaSGD) enable fast adaptation through parameter initialization optimized across tasks. (Hospedales et al., 2021) summarize these developments and their applicability across several domains. Meta-learning provides a promising alternative to conventional supervised learning when the number of clean samples is limited. However, most of the approaches heavily rely on discriminative objectives and standard techniques provide point estimates rather than uncertainty aware predictions, which limits their application in the industrial systems.

In anomaly detection, uncertainty estimation plays a critical role, where models must be able to detect deviations rather than only classifying known categories. Several probabilistic approaches such as Prior Networks (Malinin & Gales, 2018) and Bayesian neural networks provide mechanisms for capturing this uncertainty. Representative methods include probabilistic MAML, hierarchical Bayesian meta-learning (Grant et al., 2018), and variational inference based meta learners. Gordon et al. (2018) show that probabilistic inference can be integrated into meta-learning, facilitating closed form posterior updates. Classical Bayesian models support robust modeling under few-shot settings. Specifically, placing a Normal-Inverse-Wishart (NIW) prior on class-conditional embeddings with a multivariate Student-t posterior distribution offers heavy-tailed robustness desirable for anomaly detection (Murphy, 2012) which we adopt as the foundation in our paper.

Contrastive learning has been increasingly adopted in anomaly detection to improve the structure of learned feature representations by using label information to shape embedding geometry more explicitly (Khosla et al., 2020). In the context of industrial anomaly detection, several methods exploit contrastive principles to encourage separation between normal and defective samples. For instance, recent methods such as PaDiM (Defard et al., 2021), and OCR-GAN (Liang et al., 2023) utilize a contrastive approach to shaping features, though they lack explicit supervision across several types of anomalies. However, these approaches typically operate in centralized settings and do not explicitly address non-i.i.d. data distributions or client-level heterogeneity.

Recent studies like FedAvg by McMahan et al. (2017) and comprehensive reviews like Kairouz et al. (2021) highlight the main considerations in Federated Learning (FL) design (e.g., communication efficiency, client heterogeneity, and statistical bias, among others). Federated meta-learning integrates FL with rapid task adaptation. While early work by Chen et al. (2018) focuses meta-learning formulations in decentralized settings, Fallah et al. (2020) explores personalized federated learning methods inspired by MAML, including Per-FedAvg, which allows client-level personalization through local adaptation.

Existing anomaly detection methods excel in isolated settings but rarely address few-shot adaptation, uncertainty estimation, task-specific representation learning, and privacy constraints simultaneously. To address this gap in a unified manner, we propose BayPrAnoMeta, a Bayesian extension of Proto-MAML for few-shot industrial anomaly detection, and further extend it to a federated learning framework with supervised contrastive learning for privacy-preserving anomaly detection across heterogeneous industrial clients. Figure 1 illustrates the resulting architecture, comprising a shared embedding network, task-specific Bayesian normality models in the embedding space, and a federated meta-learning loop across heterogeneous clients.

The rest of the paper is organized as follows. We present our methodology and algorithm in Section 3, with theoretical guarantees in Section 4. Section 5 describes the experimental setup along with data preparation, implementation details and evaluation protocols. Experimental results and discussion are presented in Section 6. We analyze training and validation loss curves of the proposed framework in section 7. Section 8 and 9 provide error analysis and runtime analysis of the proposed framework respectively and section 10 concludes the paper. Theory of classical Proto-MAML, training and testing algorithms of BayPrAnoMeta, proofs of results, formula to compute optimal F1 score and additional tables of runtime analysis are included in the Appendices. We further provide the additional experimental results on MVTec AD and VisA dataset in the 'Supplementary Materials'. Code and results for BayPrAnoMeta are available at `https://anonymous.4open.science/r/BayPrAnoMeta-Bayesian-Proto-MAML-5CF8`.

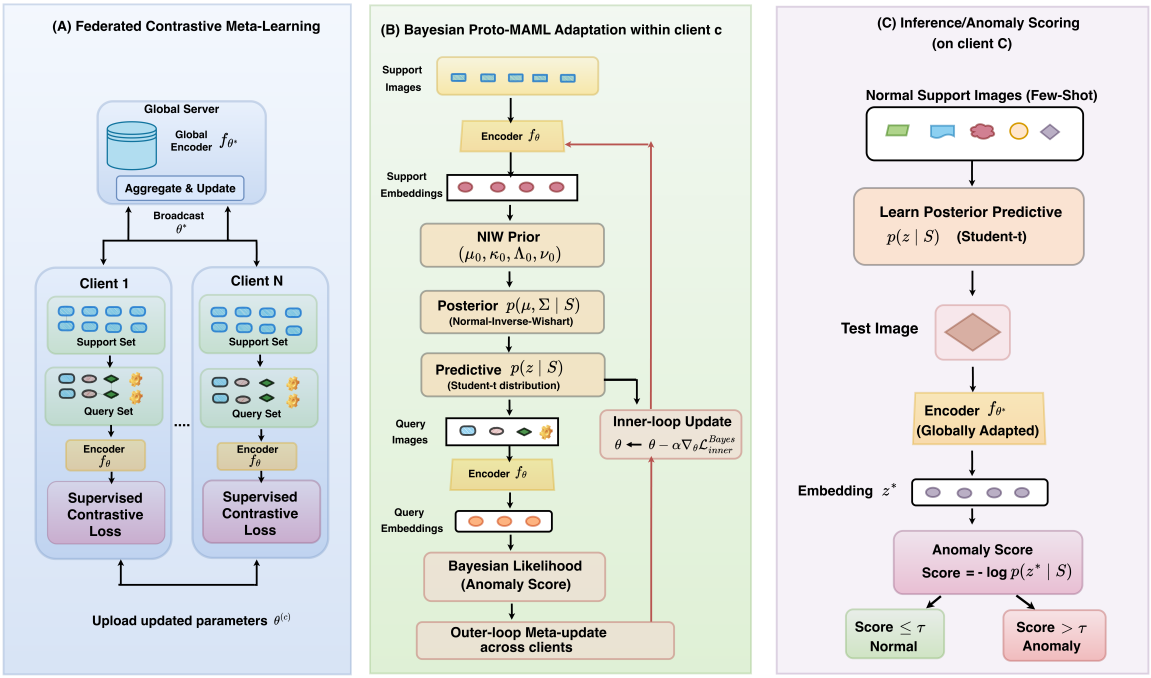


Figure 1: **Overview of the Framework:** Federated Contrastive BayPrAnoMeta framework. (A) Federated contrastive meta-learning across clients and the global server. (B) Bayesian Proto-MAML adaptation within a client using support and query sets, Normal-Inverse-Wishart priors, posterior predictive modeling, and Bayesian likelihood-based anomaly scoring. (C) Inference and anomaly scoring using few-shot normal support images and posterior predictive likelihoods.

3 Methodology

3.1 BayPrAnoMeta: Proposed Bayesian Proto-MAML for few-shot anomaly detection

Problem formulation Let $\mathcal{X}$ denote the space of input images and let $f_\theta : \mathcal{X} \to \mathbb{R}^d$ be a neural embedding function parameterized by θ. The data is assumed to arise from multiple industrial object categories, where each category defines a task $\mathcal{T} \sim p(\mathcal{T})$, with $p(\mathcal{T})$ denoting the distribution over anomaly detection tasks. For each task $\mathcal{T}$, learning proceeds episodically via an episode $\mathcal{E}_\mathcal{T} = (\mathcal{S}_\mathcal{T}, \mathcal{Q}_\mathcal{T})$, where the support set $\mathcal{S}_\mathcal{T} = \{x_i\}_{i=1}^K$ consists exclusively of normal samples, and the query set $\mathcal{Q}_\mathcal{T} = \{(x_j, y_j)\}$ contains both normal and anomalous samples, with labels $y_j \in \{0, 1\}$. Each input image $x \in \mathcal{X}$ is mapped to a latent representation $z = f_\theta(x)$.

Bayesian modeling of normal embeddings For a task $\mathcal{T}$, we assume embeddings of normal samples follow a Gaussian distribution $z \mid (\mu, \Sigma) \sim \mathcal{N}(\mu, \Sigma)$, with a conjugate Normal-Inverse-Wishart (NIW) prior $(\mu, \Sigma) \sim \mathrm{NIW}(\mu_0, \kappa_0, \Lambda_0, \nu_0)$. Given support embeddings $z_i = f_\theta(x_i)$, denoting $\bar{z} = \sum_{i=1}^K z_i/K$, $S_z = \sum_{i=1}^K (z_i - \bar{z})(z_i - \bar{z})^\top$, we get a NIW posterior with parameters

$$\begin{aligned} \kappa_n &= \kappa_0 + K, \quad \nu_n = \nu_0 + K, \quad \mu_n = \frac{\kappa_0 \mu_0 + K\bar{z}}{\kappa_n}, \\ \Lambda_n &= \Lambda_0 + S_z + \frac{\kappa_0 K}{\kappa_n}(\bar{z} - \mu_0)(\bar{z} - \mu_0)^\top. \end{aligned} \tag{1}$$

Marginalizing μ and Σ yields the posterior predictive distribution

$$p_0(z \mid \mathcal{S}_\mathcal{T}) = \text{Student-}t(z \,|\, \mu_n, a_n \Lambda_n, \nu_n - d + 1)\,, \tag{2}$$

where $a_n = (\kappa_n + 1)/\{\kappa_n(\nu_n - d + 1)\}$ (Murphy, 2007). We denote the predictive scale matrix by $\Sigma_n := a_n \Lambda_n$. The Bayesian inner loop replaces the deterministic prototype in Proto-MAML with a distributional model of the normal class in embedding space. Support embeddings are modeled as Gaussian with unknown mean and covariance under an NIW prior, yielding a posterior that combines empirical statistics with prior regularization, which is crucial in few-shot regimes. Instead of using a plug-in estimate, we marginalize the parameters to obtain a Student-t predictive distribution. The heavier tails arise from parameter uncertainty and lead to more conservative predictions in low-data regions. Thus, the inner loop can be viewed as uncertainty-aware prototype learning, where both location and dispersion are inferred jointly, resulting in more stable and robust anomaly scoring.

Bayesian inner-loop adaptation Task-specific adaptation is performed by maximizing the Bayesian likelihood of support samples,

$$\mathcal{L}_{\text{inner}}^{\text{Bayes}}(\theta; \mathcal{S}_\mathcal{T}) = -\sum_{x_i \in \mathcal{S}_\mathcal{T}} \log p_0(f_\theta(x_i) \mid \mathcal{S}_\mathcal{T}), \tag{3}$$

yielding adapted parameters $\theta_\mathcal{T} = \theta - \alpha \nabla_\theta \mathcal{L}_{\text{inner}}^{\text{Bayes}}(\theta; \mathcal{S}_\mathcal{T})$.

Anomaly reference model To model anomalous embeddings, we introduce a task-independent background distribution

$$p_1(z) = \text{Student-}t(z \mid 0, \sigma^2 I, \nu_a), \tag{4}$$

which assigns non-negligible likelihood to out-of-distribution embeddings and remains fixed across tasks. The anomaly reference model is fixed across tasks and is used both in the supervised query loss during meta-training and in the likelihood-ratio anomaly score at test time.

Outer-loop Bayesian meta-objective Given the task-adapted parameters $\theta_\mathcal{T}$, query samples are embedded as $z_j = f_{\theta_\mathcal{T}}(x_j)$. For each query pair $(x_j, y_j) \in \mathcal{Q}_\mathcal{T}$, the supervised Bayesian query loss is defined as

$$\mathcal{L}_{\text{query}}^{\text{Bayes}} = -\left[(1 - y_j) \log p_0(z_j \mid \mathcal{S}_\mathcal{T}) + y_j \log p_1(z_j)\right]. \tag{5}$$

The meta-learning objective minimizes the expected query loss across tasks,

$$\min_{\theta} \ \mathbb{E}_{\mathcal{T}\sim p(\mathcal{T})}\left[\sum_{(x_j,y_j)\in\mathcal{Q}_{\mathcal{T}}} \mathcal{L}^{\text{Bayes}}_{\text{query}}(\theta_{\mathcal{T}})\right],$$

with gradients propagated through the Bayesian inner-loop updates, and the meta-parameters optimized via gradient descent with meta-learning rate β:

$$\theta \leftarrow \theta - \beta\nabla_{\theta}\mathbb{E}_{\mathcal{T}\sim p(\mathcal{T})}\left[\sum_{(x_j,y_j)\in\mathcal{Q}_{\mathcal{T}}} \mathcal{L}^{\text{Bayes}}_{\text{query}}(\theta_{\mathcal{T}})\right].$$

Testing and anomaly scoring At test time, a previously unseen task $\mathcal{T}$ is provided with a small support set $\mathcal{S}_{\mathcal{T}}$ containing only normal samples. The posterior predictive distribution $p_0(\cdot \mid \mathcal{S}_{\mathcal{T}})$ is computed using the closed-form NIW updates, and the embedding network is adapted via the Bayesian inner loop to obtain $\theta_{\mathcal{T}}$. For a test sample x^*, the anomaly score is defined using a likelihood-ratio criterion,

$$s(x^*) = \log p_1(f_{\theta_{\mathcal{T}}}(x^*)) - \log p_0(f_{\theta_{\mathcal{T}}}(x^*) \mid \mathcal{S}_{\mathcal{T}}).$$

Samples with larger scores are considered more likely to be anomalous. This likelihood-based scoring provides calibrated, uncertainty-aware anomaly detection without requiring distance normalization. We emphasize that θ parameterizes the deterministic embedding network, while we introduce task-specific latent parameters (μ, Σ) as inferred in closed form from support embeddings.

A key difficulty in few-shot anomaly detection is that K can be much smaller than the embedding dimension d. In such regimes, likelihood-based objectives that rely on *empirical* covariance estimates may become ill-posed (e.g. singular covariance when $K < d$), leading to undefined log-likelihoods and unstable gradients. In the proposition below, we show that the NIW prior used in BayPrAnoMeta guarantees that the Bayesian predictive model remains well-defined and numerically stable for *any* support size $K \geq 1$, including $K < d$.

Proposition 3.1. *Assume $\Lambda_0 \succ 0$ and $\nu_0 > d-1$. Then for every $K \geq 1$, every support set $\mathcal{S}$ of size K, and every parameter θ, the posterior predictive density $p_0(\cdot \mid \mathcal{S})$ is a proper multivariate Student-t distribution with (i) strictly positive definite scale matrix $\Sigma_n \succ 0$, and (ii) finite log-density $\log p_0(z \mid \mathcal{S})$ for all $z \in \mathbb{R}^d$. In particular, the predictive model remains well-defined even when $K < d$.*

We provide the proof of the above proposition in the Appendix.

3.2 Federated Bayesian Proto-MAML for few-shot anomaly detection with supervised contrastive loss

In reality, data are generated across multiple clients such as machines, production lines, or factories, operating under heterogeneous conditions. These clients cannot share data due to privacy policies and communication constraints, which motivates the federated learning paradigm, where model training is collaborative while data remain local. While Section 3.1 introduces BayPrAnoMeta, a Bayesian Proto-MAML framework for few-shot anomaly detection in a centralized meta-learning setting, we now extend this formulation to a federated learning scenario, which enables privacy-preserving, uncertainty-aware anomaly detection across distributed and heterogeneous clients. Compared to the centralized BayPrAnoMeta formulation in Section 3.1, the proposed federated extension introduces the following: ***(i)*** Training tasks are distributed across multiple clients, each holding local data that cannot be shared. Each client independently performs Bayesian inner-loop adaptation, and only model updates are communicated to the server; and ***(ii)*** While the Bayesian inner-loop adaptation remains identical to Section 3.1, the outer-loop objective is changed to incorporate a supervised contrastive regularization term in addition to the Bayesian query likelihood to mitigate representation drift caused by heterogeneous client data, which encourages globally consistent and discriminative embeddings.

Federated episodic learning setup We consider a federated learning setting with C clients, indexed by $c \in \{1, \ldots, C\}$. Each client corresponds to an independent data source and retains its data locally. Learning

proceeds episodically. At each communication round, client c samples a task $\mathcal{T}_c \sim p(\mathcal{T})$ from its local data distribution and constructs an episode $\mathcal{E}_c = (\mathcal{S}_c, \mathcal{Q}_c)$, where $\mathcal{S}_c$ denotes a small support set of normal samples and Q_c denotes a query set containing labeled normal and anomalous samples.[1]

Bayesian client-side inner-loop adaptation Given the global meta-parameters θ broadcast by the server, each client performs task-specific adaptation using the Bayesian Proto-MAML inner loop introduced in Section 3.1. For a support set S_c, the Bayesian inner loss $\mathcal{L}_{\text{inner}}^{\text{Bayes}}(\theta; \mathcal{S}_c)$ is defined as in Eq. (3). The adapted parameters are obtained via the one-step update $\theta'_c = U(\theta; \mathcal{S}_c) = \theta - \alpha \nabla_\theta \mathcal{L}_{\text{inner}}^{\text{Bayes}}(\theta; \mathcal{S}_c)$.

Episode-level outer-loop objective After inner-loop adaptation, each client evaluates an episode-level outer loss on its query set $\mathcal{Q}_c$ using the adapted parameters θ'_c. Following Section 3.1, the Bayesian query likelihood loss $\mathcal{L}_{\text{query}}^{\text{Bayes}}(\theta'_c; \mathcal{S}_c, \mathcal{Q}_c)$ is computed using the Student-t posterior predictive for normal samples and a fixed anomaly reference distribution. To improve representation quality across heterogeneous clients, we augment this objective with a supervised contrastive regularization term $\mathcal{L}_{\text{supcon}}(\theta'_c; \mathcal{Q}_c)$. The resulting episode-level loss is

$$\mathcal{L}(\theta; \mathcal{E}_c) = \mathcal{L}_{\text{query}}^{\text{Bayes}}(\theta'_c; \mathcal{S}_c, \mathcal{Q}_c) + \lambda \, \mathcal{L}_{\text{supcon}}(\theta'_c; \mathcal{Q}_c), \tag{6}$$

where $\lambda > 0$ controls the strength of the contrastive regularization.

Supervised contrastive loss. Let $z_i = f_{\theta'_c}(x_i)$ denote the adapted embedding of a query sample $(x_i, y_i) \in Q_c$, with label $y_i \in \{0, 1\}$ indicating normal or anomalous class. For each anchor i, define the positive index set $P(i) = \{\, j \neq i \mid y_j = y_i \,\}$. The supervised contrastive loss is given by

$$\mathcal{L}_{\text{supcon}} = \sum_i \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(z_i^\top z_p / \tau)}{\sum_{k \neq i} \exp(z_i^\top z_k / \tau)}, \tag{7}$$

where τ is the temperature parameter.

Client and global objectives Each client c induces an episode distribution $\mathcal{D}_c$ through its local task and episode sampling procedure. The client-level objective is defined as

$$F_c(\theta) \;:=\; \mathbb{E}_{\mathcal{E}_c \sim \mathcal{D}_c}\big[\mathcal{L}(\theta; \mathcal{E}_c)\big]. \tag{8}$$

The global federated meta-objective is the average of client objectives,

$$F(\theta) \;:=\; \frac{1}{C} \sum_{c=1}^{C} F_c(\theta). \tag{9}$$

Federated optimization and server-side aggregation At round r, the server has θ^{r-1} and selects a subset $\mathcal{C}_r \subseteq \mathcal{C}$. Each selected client draws an episode $\mathcal{E}_c^r \sim \mathcal{D}_c$ and forms the stochastic meta-gradient $g_c^r := \nabla_{\theta_c} \mathcal{L}(\theta^{r-1}; \mathcal{E}_c)$. The server then aggregates

$$\bar{g}^{\,r} = \frac{1}{|\mathcal{C}_r|} \sum_{c \in \mathcal{C}_r} g_c^r, \qquad \theta^r = \theta^{r-1} - \eta \, \bar{g}^{\,r}, \tag{10}$$

where $\eta := \gamma\beta$ (server aggregation $\times$ meta step size). This procedure is analyzed theoretically in the Appendix.

Test-time anomaly detection At test time, no contrastive regularization is used. For a previously unseen task, Bayesian inner-loop adaptation is performed using a small support set of normal samples, and anomaly scores are computed using the likelihood-ratio criterion defined in Section 3.1.

[1]Each client samples tasks locally; hence, in the federated setting, we index episodes directly by the client c and write $\mathcal{E}_c = (\mathcal{S}_c, \mathcal{Q}_c)$, with the underlying task index left implicit.

Algorithm 1 Federated BayPrAnoMeta with Supervised Contrastive Learning

Input: Client set $\mathcal{C}$; embedding network f_θ; NIW prior $(\mu_0, \kappa_0, \Lambda_0, \nu_0)$; anomaly reference model $p_1(z)$; temperature τ; regularization coefficient λ; inner-loop step size α; meta step size β; server aggregation rate γ; number of rounds R
Output: Global meta-parameters θ^R
Initialize: θ^0
for $r = 1$ **to** R **do**
  Server broadcasts θ^{r-1}
  for all clients $c \in \mathcal{C}_r$ **in parallel do**
    $\theta_c \leftarrow \theta^{r-1}$
    Sample an episode $\mathcal{E}_c = (\mathcal{S}_c, \mathcal{Q}_c)$
    Bayesian inner loop:
    Compute embeddings $z_i = f_{\theta_c}(x_i)$ for $x_i \in \mathcal{S}_c$
    Compute NIW posterior parameters $(\mu_{n,c}, \Lambda_{n,c}, \kappa_{n,c}, \nu_{n,c})$
    Define Student-t posterior predictive $p_{0,\theta_c}(\cdot \mid \mathcal{S}_c)$
    $\mathcal{L}^{\text{Bayes}}_{\text{inner}}(\theta_c; \mathcal{S}_c) = -\sum_{x_i \in S_c} \log p_{0,\theta_c}\big(f_{\theta_c}(x_i) \mid \mathcal{S}_c\big)$
    $\theta'_c \leftarrow U(\theta_c; \mathcal{S}_c) = \theta_c - \alpha \nabla_{\theta_c} \mathcal{L}^{\text{Bayes}}_{\text{inner}}(\theta_c; \mathcal{S}_c)$
    Outer loop:
    Compute adapted query embeddings $z_j = f_{\theta'_c}(x_j)$ for $(x_j, y_j) \in \mathcal{Q}_c$
    Compute $\mathcal{L}^{\text{Bayes}}_{\text{query}}(\theta'_c; \mathcal{S}_c, \mathcal{Q}_c)$ as in (5)
    Define $P(i) = \{j \neq i : \ y_j = y_i\}$
    Compute $\mathcal{L}_{\text{supcon}}(\theta'_c; \mathcal{Q}_c)$ as in (7)
    $\mathcal{L}(\theta^{r-1}; \mathcal{E}_c) = \mathcal{L}^{\text{Bayes}}_{\text{query}}(\theta'_c; \mathcal{S}_c, \mathcal{Q}_c) + \lambda\, \mathcal{L}_{\text{supcon}}(\theta'_c; \mathcal{Q}_c)$
    $g^r_c \leftarrow \nabla_{\theta_c} \mathcal{L}(\theta^{r-1}; \mathcal{E}_c)$
    Send g^r_c to server
  end for
  Server aggregation:
  $\bar{g}^{\,r} = \frac{1}{|\mathcal{C}_r|} \sum_{c \in \mathcal{C}_r} g^r_c, \ \eta \leftarrow \gamma\beta, \ \theta^r = \theta^{r-1} - \eta\, \bar{g}^{\,r}$
end for
Test-time:
Given a new episode $\mathcal{E} = (\mathcal{S}, \mathcal{Q})$, compute $\theta' = U(\theta^R; \mathcal{S})$
For a test sample x^*, output anomaly score
$s(x^*) = \log p_1(f_{\theta'}(x^*)) - \log p_{0,\theta'}(f_{\theta'}(x^*) \mid \mathcal{S})$

To conduct this experiment, we consider the object categories (like, *Bottle*, *Cable*, among others) of the MVTec AD dataset (Bergmann et al., 2019) as individual clients of the federated set-up. Here, we assume each client as an independent anomaly detection task. Each client corresponds to a distinct industrial object category and performs task-specific Bayesian Proto-MAML adaptation and supervised contrastive representation learning exclusively on its local data, which is never shared with the server. The client datasets are strongly non-i.i.d., differing in visual structure, anomaly subtypes, and class imbalance, and the server does not have access to a unified global objective or synchronized mini-batches. Instead, clients independently compute meta-gradients arising from episodic, task-conditioned objectives, and only these updates are communicated to the server, which performs aggregation without observing client losses, labels, or contrastive pairs. This separation between local task adaptation and global aggregation, combined with heterogeneous client objectives and privacy-preserving update exchange, fundamentally distinguishes our approach from distributed training and aligns it with the defining principles of federated learning. Moreover, client heterogeneity is explicitly modeled in our federated setup, where each client corresponds to a distinct object category with different structures, defect types, and class imbalance, yielding a strongly non-i.i.d. setting. Robustness arises from two mechanisms: (i) task-specific Bayesian adaptation, which models client-level normality from limited support data, and (ii) shared meta-representation learning across clients. In addition, contrastive regularization helps align representations under heterogeneous distributions, with more noticeable benefits when client variability is high. The algorithm of this setting is given in Algorithm 1, along with the proposed BayPrAnoMeta training and testing algorithm (Algorithms 2 and 3 respectively) in the Appendix.

In the next section, we provide an optimization-theoretic convergence guarantee for the federated training procedure in Algorithm 1. Our goal is to establish convergence to *stationary points* of the (nonconvex) global federated Bayesian meta-objective under standard smoothness, variance, and heterogeneity assumptions, and to make explicit the additional regularity conditions required by the NIW–Student-t predictive likelihood.

4 Theoretical Guarantees

Assumptions. The following conditions are standard in nonconvex stochastic/federated optimization and are stated here to make the guarantee precise. The expressions of the client objective and meta objective are as in (8) and (9) respectively.

Assumption 4.1 (Smoothness of client objectives)**.** Each F_c is L-smooth: for all θ, θ',

$$\|\nabla F_c(\theta) - \nabla F_c(\theta')\| \le L\|\theta - \theta'\|.$$

Consequently, F is also L-smooth.

Assumption 4.2 (Unbiased stochastic meta-gradients and bounded variance)**.** For each round r and selected client $c \in \mathcal{C}_r$,

$$\mathbb{E}\big[g_c^r \mid \theta^r\big] = \nabla F_c(\theta^r), \quad \mathbb{E}\big[\|g_c^r - \nabla F_c(\theta^r)\|^2 \mid \theta^r\big] \le \sigma^2.$$

Assumption 4.3 (Bounded client heterogeneity)**.** There exists $\zeta \ge 0$ such that for all θ,

$$\frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}} \big\|\nabla F_c(\theta) - \nabla F(\theta)\big\|^2 \le \zeta^2.$$

Assumption 4.4 (Predictive regularity (NIW–Student-t well-posedness))**.** There exists $\varepsilon > 0$ such that the predictive scale matrices satisfy, for all θ and all support sets $\mathcal{S}$ with $|\mathcal{S}| = K$,

$$\Sigma_n \succeq \varepsilon I_d.$$

Under the NIW prior with $\Lambda_0 \succ 0$ and $\nu_0 > d - 1$, the predictive scale Σ_n is strictly positive definite for all $K \ge 1$ (Proposition 3.1), so Assumption 4.4 holds automatically. Additionally, this prevents singular covariance estimates when $K < d$ (few-shot regime), and it ensures the Student-t log-likelihood has bounded curvature.

Lemma 4.5 (Smoothness of Student-t negative log-likelihood in the embedding)**.** *Fix $\nu > 0$, $\mu \in \mathbb{R}^d$, and a symmetric $\Sigma \succeq \varepsilon I_d$. Let*

$$\ell(z) \;:=\; -\log \text{Student-}t(z \mid \mu, \Sigma, \nu), \qquad z \in \mathbb{R}^d.$$

Then ℓ is twice differentiable and has a globally bounded Hessian on any bounded set $\{z : \|z - \mu\| \le R_z\}$. In particular, for all $\|z - \mu\| \le R_z$,

$$\begin{aligned}\|\nabla^2 \ell(z)\| &\le \frac{\nu + d}{\nu}\,\|\Sigma^{-1}\| + \frac{2(\nu + d)}{\nu^2}\,\|\Sigma^{-1}\|^2\, R_z^2 \\ &\le \frac{\nu + d}{\nu}\,\frac{1}{\varepsilon} + \frac{2(\nu + d)}{\nu^2}\,\frac{R_z^2}{\varepsilon^2}.\end{aligned}$$

Consequently, $\nabla \ell$ is Lipschitz on $\{z : \|z - \mu\| \le R_z\}$, with Lipschitz constant equal to the bound above.

Implication for BayPrAnoMeta. Under Assumption 4.4 and bounded embeddings within episodes, Lemma 4.5 implies the Student-t likelihood terms in (2)–(5) are smooth in the embedding z. Together with standard smoothness of f_θ (bounded Jacobian/Hessian on the parameter domain traversed by training), this yields smoothness of the composed losses in θ, supporting Assumption 4.1.

We now state a standard nonconvex convergence guarantee for the federated update (10). The result is stated in terms of stationarity of the global objective F.

Theorem 4.6 (Nonconvex convergence of Federated BayPrAnoMeta)**.** *Suppose Assumptions 4.1–4.4 hold. Let $\{\theta^r\}_{r\ge 0}$ be generated by* (10)*, where at each round the client subset $\mathcal{C}_r$ is sampled uniformly without bias (or $\mathcal{C}_r = \mathcal{C}$). Assume $\eta \le 1/L$. Then, for any $R \ge 1$,*

$$\begin{aligned}\frac{1}{R}\sum_{r=0}^{R-1} \mathbb{E}\big[\|\nabla F(\theta^r)\|^2\big] \le\; & \frac{2\big(F(\theta^0) - F^\star\big)}{\eta R} \\ & +\; L\eta\left(\frac{\sigma^2}{|\mathcal{C}_r|} + \zeta^2\right).\end{aligned} \tag{11}$$

where $F^\star := \inf_\theta F(\theta)$*. In particular, choosing* $\eta = \Theta(R^{-1/2})$ *yields*

$$\min_{0\le r\le R-1} \mathbb{E}\|\nabla F(\theta^r)\|^2 = \mathcal{O}\left(\frac{1}{\sqrt{R}}\right) + \mathcal{O}\left(\frac{\sigma^2}{|\mathcal{C}_r|}\right) + \mathcal{O}(\zeta^2).$$

Theorem 4.6 establishes that Federated BayPrAnoMeta converges to an ε-stationary point of the global objective F at the standard nonconvex stochastic rate, up to two unavoidable error sources: (i) episode-level stochasticity σ^2, reduced by larger client participation $|\mathcal{C}_r|$, and (ii) non-IID client heterogeneity ζ^2. Also, the regularity of the predictive scale matrix is precisely what permits differentiable inner-loop adaptation and the application of standard nonconvex federated optimization arguments to our Bayesian meta-learning objective. The novelty lies not in the convergence rate per se, but in showing that Bayesian inner-loop adaptation with heavy-tailed predictive likelihoods satisfies the smoothness conditions required for federated meta-optimization.

5 Experimental Setup

Implementation details All methods are implemented in PyTorch (Paszke et al., 2019), using the `higher` library for differentiable inner-loop optimization. Experiments are conducted on CPU.

Dataset and task definition We evaluate all methods on the MVTec Anomaly Detection (MVTec AD) dataset (Bergmann et al., 2019), which contains high-resolution industrial images from 15 object categories (e.g., *Bottle, Cable, Capsule, Grid*, among others). Each object category is treated as an independent meta-learning task $\mathcal{T}$, following standard practice in few-shot learning. Each category consists of: *(i).* a set of normal images provided under `train/good` and `test/good`, and *(ii).* multiple anomaly subtypes organized as subdirectories under `test/`. All images are resized to 224×224 pixels and normalized using ImageNet mean and standard deviation. We additionally evaluate our methods on the VisA (Zou et al., 2022) benchmark, which contains 12 industrial object categories (e.g., *candle, macaroni1, fryum* etc.).

Episodic few-shot formulation Learning proceeds episodically. For each task $\mathcal{T}$, an episode $\mathcal{E}_\mathcal{T} = (\mathcal{S}_\mathcal{T}, \mathcal{Q}_\mathcal{T})$ is constructed as follows: *(i.)* **Support set $\mathcal{S}_\mathcal{T}$:** $K = 5$ normal samples only, and *(ii.)* **Query set $\mathcal{Q}_\mathcal{T}$:** $Q_N = 12$ normal samples and $Q_A = 4$ anomalous samples. Query labels are binary, with $y = 0$ denoting normal samples and $y = 1$ denoting anomalous samples. This episodic construction is used consistently during training, validation, and testing.

Data preparation To evaluate generalization to unseen anomaly types, we adopt the following data preparation protocol at the category level for MVTec AD dataset. For each object category: one anomaly subtype is held out entirely for testing, and all remaining anomaly subtypes are used during training. Specifically: *(i).* **Training:** normal samples from `train/good` and `test/good`, and anomalous samples from all but the held-out subtype. and *(ii).* **Testing:** normal samples from `test/good`, and anomalous samples exclusively from the held-out subtype. Categories with fewer than two anomaly subtypes (e.g., toothbrush) are excluded from our experiments. This data preparation ensures that anomaly types encountered at test time are strictly unseen during training.

In VisA dataset, anomaly subtypes are not present. For each object category, anomaly images are randomly split into training and testing subsets, approximately 20% anomaly images are held out exclusively for evaluation, and the remaining anomaly images used during training. All normal images are retained in both splits. We keep all other experimental protocols for this dataset identical to those used for the MVTec AD experimental set-up.

5.1 Model architecture

All methods share a common embedding network $f_\theta : \mathcal{X} \to \mathbb{R}^{128}$.

Backbone We use a ResNet-18 architecture initialized with ImageNet-pretrained weights. The final classification layer is removed, the first max-pooling layer is replaced with average pooling to preserve spatial detail, and global average pooling produces a 512-dimensional feature vector.

Projection head The backbone output is passed through a projection head consisting of a linear layer mapping $512 \to 128$, followed by a ReLU activation and Layer Normalization. The resulting 128-dimensional embedding is used for prototype estimation, Bayesian modeling, contrastive learning, and anomaly scoring.

Meta-learning compatibility The entire network, including both backbone and projection head, is optimized within a meta-learning framework. Gradients are propagated through task-specific inner-loop updates using differentiable optimization, enabling end-to-end meta-learning without task-specific classifier heads.

To ensure fair comparison, the same architecture is used unchanged across all baselines and proposed methods.

Optimization and hyperparameters Unless stated otherwise, all experiments use the hyperparameters listed in Table 1. Meta-optimization is performed using the Adam optimizer, and all random seeds are fixed to 42 for reproducibility.

Table 1: Optimization and training hyperparameters

Hyperparameter	**Value**
Inner-loop learning rate (α)	5×10^{-4}
Inner-loop steps	1
Meta-learning rate (β)	1×10^{-4}
Episodes per epoch	50
Validation episodes per epoch	20
Training epochs / federated rounds	50
Server aggregation rate (γ)	1.0
Contrastive temperature (τ)	0.07
Regularization coefficient (λ)	0.1
NIW prior mean (μ_0)	$\mathbf{0} \in \mathbb{R}^{128}$
NIW prior mean strength (κ_0)	0.01
NIW prior scale matrix (Λ_0)	I_{128}
NIW prior degrees of freedom (ν_0)	$128 + 2$
Anomaly reference mean (μ_a)	$\mathbf{0} \in \mathbb{R}^{128}$
Anomaly reference covariance (Σ_a)	$100 \cdot I_{128}$
Anomaly reference degrees of freedom (ν_a)	2

5.2 Training and Validation

At each epoch, training loss is computed as the average meta-loss over 50 training episodes. Validation loss is computed over 20 validation episodes sampled from held-out data. Training and validation loss curves are reported to assess convergence and overfitting. The same set-up is used for centralized and federated model training.

Evaluation metrics At test time, anomaly scores are computed at the episode level and aggregated across episodes, i.e. All metrics are reported as mean $\pm$ standard error over 300 randomly sampled test episodes. We report: Area Under the ROC (AUROC), Area Under the Precision-Recall (AUPRC), and F1-scores based on optimal threshold selected from the anomaly scores. Additionally, we obtain ROC curves, precision-recall curves, and score histograms, which we do not report in this paper for brevity. Details on computation of optimal threshold based F1-score are given in the Appendix (Section D).

Table 2: AUROC (%) across MVTec AD object categories for several baselines.

Category	PatchCore-25	Classical MAML	Classical Proto-MAML	**BayPrAnoMeta**	% Gain
Bottle	$\mathbf{100.0 \pm 0.00}$	27.5 ± 0.90	98.0 ± 0.14	98.4 ± 0.13	-1.60%
Cable	50.8 ± 0.79	58.4 ± 0.83	95.7 ± 0.30	$\mathbf{96.9 \pm 0.23}$	$+1.25\%$
Capsule	$\mathbf{68.9 \pm 0.56}$	51.6 ± 0.99	61.8 ± 0.86	67.9 ± 0.76	-1.45%
Carpet	47.6 ± 0.63	75.8 ± 0.90	89.5 ± 0.57	$\mathbf{93.7 \pm 0.45}$	$+4.69\%$
Grid	42.6 ± 0.80	62.6 ± 0.69	57.7 ± 0.87	$\mathbf{67.9 \pm 0.79}$	$+8.47\%$
Hazelnut	43.7 ± 0.57	61.5 ± 0.98	66.7 ± 0.81	$\mathbf{75.0 \pm 0.74}$	$+12.44\%$
Leather	36.5 ± 0.67	44.4 ± 1.15	85.5 ± 0.60	$\mathbf{89.5 \pm 0.50}$	$+4.68\%$
Metal Nut	53.2 ± 0.63	38.1 ± 0.94	84.4 ± 0.75	$\mathbf{85.8 \pm 0.67}$	$+1.66\%$
Pill	44.6 ± 0.54	60.1 ± 1.06	60.9 ± 0.85	$\mathbf{65.8 \pm 0.80}$	$+8.05\%$
Screw	56.0 ± 0.55	45.9 ± 0.78	50.1 ± 0.87	$\mathbf{57.6 \pm 0.76}$	$+2.86\%$
Tile	53.0 ± 0.60	95.6 ± 0.35	98.4 ± 0.13	$\mathbf{99.5 \pm 0.07}$	$+1.12\%$
Transistor	63.8 ± 0.97	64.1 ± 0.96	81.0 ± 0.61	$\mathbf{84.4 \pm 0.57}$	$+4.20\%$
Wood	54.3 ± 0.90	64.8 ± 0.86	99.2 ± 0.10	$\mathbf{99.6 \pm 0.05}$	$+0.40\%$
Zipper	65.4 ± 0.63	59.8 ± 0.96	93.7 ± 0.38	$\mathbf{95.0 \pm 0.32}$	$+1.39\%$

Table 3: AUROC (%) across VisA object categories for several baselines.

Category	PatchCore-25	Classical MAML	Classical Proto-MAML	**BayPrAnoMeta**	% Gain
candle	$\mathbf{100.00 \pm 0.00}$	70.50 ± 0.60	84.39 ± 0.63	84.72 ± 0.65	-15.28%
capsules	61.44 ± 0.92	44.36 ± 0.24	$\mathbf{76.43 \pm 0.51}$	76.12 ± 0.59	-0.41%
cashew	46.59 ± 1.31	$\mathbf{89.45 \pm 0.35}$	84.17 ± 0.48	85.96 ± 0.47	-3.90%
chewinggum	41.06 ± 0.86	49.24 ± 0.24	69.28 ± 0.51	$\mathbf{72.53 \pm 0.53}$	$+4.69\%$
fryum	37.53 ± 0.73	40.37 ± 0.43	$\mathbf{74.24 \pm 0.78}$	73.78 ± 0.74	-0.62%
macaroni1	34.70 ± 0.97	41.16 ± 0.34	61.85 ± 0.77	$\mathbf{65.36 \pm 0.75}$	$+5.68\%$
macaroni2	$\mathbf{91.77 \pm 0.25}$	54.47 ± 0.55	47.36 ± 0.72	54.53 ± 0.81	-40.58%
pcb1	64.25 ± 0.85	$\mathbf{96.86 \pm 0.19}$	83.81 ± 1.10	81.13 ± 0.97	-16.24%
pcb2	47.64 ± 1.16	36.48 ± 0.27	63.00 ± 0.83	$\mathbf{65.94 \pm 0.82}$	$+4.67\%$
pcb3	38.88 ± 1.44	50.17 ± 0.58	$\mathbf{88.36 \pm 0.39}$	87.62 ± 0.46	-0.84%
pcb4	41.56 ± 1.12	63.89 ± 0.57	92.49 ± 0.39	$\mathbf{92.90 \pm 0.42}$	$+0.44\%$
pipe_fryum	43.03 ± 1.46	28.76 ± 0.30	87.03 ± 0.41	$\mathbf{89.74 \pm 0.40}$	$+3.11\%$

6 Results and Discussion

Table 2 compares the proposed BayPrAnoMeta (Bayesian Proto-MAML) with several baselines: PatchCore (Roth et al., 2022), classical MAML (Finn et al., 2017), and Proto-MAML (Alfatemi et al., 2025) for MVTec AD dataset. To ensure a fair architectural comparison, all methods operate under a shared representation constraint. Specifically, PatchCore is adapted to use the same 128-dimensional ResNet-18 projection head as BayPrAnoMeta, rather than its original multi-scale, high-dimensional feature hierarchy. Patch representations are obtained via dense feature extraction using this shared encoder, followed by k-center subsampling (25%) for efficiency and nearest-neighbor scoring. This constrained version intentionally omits multi-layer feature aggregation, which is known to benefit PatchCore, and therefore may reduce its absolute performance. However, this design ensures that performance differences reflect inference strategy—Bayesian meta-learning and uncertainty-aware scoring—rather than representation capacity or feature specialization. PatchCore is trained exclusively on normal samples from the held-out test split, following the same episodic protocol used for the meta-learning baselines.

PatchCore exhibits strong performance on visually homogeneous and texture-simple objects such as *Bottle*, where anomalies are highly separable at the local patch level. However, its performance degrades on structurally complex categories including *Carpet*, *Grid*, *Leather*, and *Hazelnut*; where local patch statistics alone are less effective at capturing higher-level semantic or global shape deviations. In contrast, BayPrAnoMeta

Table 4: AUROC (%) comparison across MVTec AD object categories for ablation studies.

Category	Contrastive BayPrAnoMeta	Federated BayPrAnoMeta	Federated Contrastive BayPrAnoMeta
Bottle	49.0 ± 1.94	88.7 ± 0.73	$\mathbf{88.9 \pm 0.75}$
Cable	$\mathbf{93.6 \pm 0.45}$	59.4 ± 1.02	58.8 ± 1.03
Capsule	55.4 ± 1.00	$\mathbf{68.9 \pm 0.95}$	63.8 ± 1.07
Carpet	$\mathbf{53.0 \pm 1.21}$	45.4 ± 1.32	52.3 ± 1.22
Grid	$\mathbf{65.3 \pm 0.88}$	59.7 ± 0.97	59.1 ± 0.90
Hazelnut	75.1 ± 0.80	80.9 ± 0.84	$\mathbf{82.4 \pm 0.79}$
Leather	47.2 ± 1.69	55.9 ± 1.09	$\mathbf{59.7 \pm 1.13}$
Metal Nut	$\mathbf{76.2 \pm 0.96}$	70.8 ± 0.83	72.0 ± 0.78
Pill	55.2 ± 1.08	$\mathbf{57.8 \pm 0.98}$	51.9 ± 0.98
Screw	54.9 ± 0.81	$\mathbf{63.1 \pm 1.16}$	46.1 ± 1.17
Tile	39.4 ± 1.61	$\mathbf{56.9 \pm 1.08}$	52.2 ± 1.14
Transistor	$\mathbf{67.8 \pm 0.94}$	41.8 ± 1.08	43.1 ± 1.10
Wood	62.6 ± 1.44	67.8 ± 1.63	$\mathbf{75.7 \pm 1.42}$
Zipper	71.3 ± 1.19	72.9 ± 1.20	$\mathbf{84.8 \pm 0.92}$

consistently outperforms PatchCore across most object categories, particularly those involving global shape irregularities or high intra-class variability. PatchCore results are reported at 25% coreset, with additional configurations (1%, 10%, 50%, 100%) provided in the 'Supplementary Materials'. These results indicate that while PatchCore remains a strong baseline, BayPrAnoMeta offers a more robust, uncertainty-aware, and task-adaptive approach to few-shot anomaly detection.

Classical MAML performs poorly across object categories (Table 2), likely due to its reliance on discriminative gradient-based adaptation. Incorporating prototypical networks into MAML reframes task adaptation as a metric-learning problem, enabling the learning of compact representations of normal samples. This hybrid Proto-MAML formulation significantly improves anomaly detection performance across heterogeneous object categories, emphasizing the role of representation-based adaptation in few-shot settings.

Building upon Proto-MAML, Table 2 shows that Bayesian Proto-MAML (BayPrAnoMeta) further improves detection accuracy by modeling normal embeddings probabilistically and using likelihood-based anomaly scoring. This probabilistic treatment is particularly effective for challenging object categories with high intra-class variability, leading to the best performance in 12 out of 14 categories. The gains are most pronounced on structurally complex objects such as *Grid* (+8.47 %), *Hazelnut* (+12.44 %), and *Pill* (+8.05 %); while performance remains comparable on visually simpler categories such as *Bottle* and *Capsule*. These results indicate that Bayesian uncertainty modeling is especially beneficial in challenging, high intra-class variability settings. The % Gain column reports the relative improvement of BayPrAnoMeta over the strongest competing baseline for each category.

In addition to MVTec AD, we evaluated on the VisA dataset (Zou et al., 2022), which introduces different object categories and data distributions (Table 3). This dataset is also another widely adopted benchmark in the image anomaly detection literature and is frequently included in several self-supervised anomaly detection surveys (e.g. Hojjati et al. (2024)). Due to the presence of highly heterogeneous clients, the AUROC gain for VisA dataset is not as good as MVTec AD dataset, whereas the AUPRC gains in VisA dataset is substantial. BayPrAnoMeta shows consistent improvements over Proto-MAML across most categories, with notable gains such as +17.1% (macaroni2), +11.3% (macaroni1), +10.1% (pcb2), and +4.8% (chewinggum) in AUPRC (check 'Supplementary Materials') . While a few categories show marginal drops, the overall trend remains positive, indicating robust cross-dataset generalization. This is expected since the method operates on learned embeddings and models task-specific normality via a Bayesian formulation that is not tied to a specific dataset and is particularly effective in few-shot regimes. Full results are provided in the 'Supplementary Materials'.

Table 5: Effect of the contrastive loss in the federated setting for MVTec AD (AUROC).

Category	Federated BayPrAnoMeta	Federated Contrastive BayPrAnoMeta	% Gain
Bottle	88.7 ± 0.73	$\mathbf{88.9 \pm 0.75}$	+0.23%
Cable	$\mathbf{59.4 \pm 1.02}$	58.8 ± 1.03	−1.01%
Capsule	$\mathbf{68.9 \pm 0.95}$	63.8 ± 1.07	−7.40%
Carpet	45.4 ± 1.32	$\mathbf{52.3 \pm 1.22}$	+15.20%
Grid	$\mathbf{59.7 \pm 0.97}$	59.1 ± 0.90	−1.01%
Hazelnut	80.9 ± 0.84	$\mathbf{82.4 \pm 0.79}$	+1.85%
Leather	55.9 ± 1.09	$\mathbf{59.7 \pm 1.13}$	+6.80%
Metal Nut	70.8 ± 0.83	$\mathbf{72.0 \pm 0.78}$	+1.69%
Pill	$\mathbf{57.8 \pm 0.98}$	51.9 ± 0.98	−10.21%
Screw	$\mathbf{63.1 \pm 1.16}$	46.1 ± 1.17	−26.94%
Tile	$\mathbf{56.9 \pm 1.08}$	52.2 ± 1.14	−8.26%
Transistor	41.8 ± 1.08	$\mathbf{43.1 \pm 1.10}$	+3.11%
Wood	67.8 ± 1.63	$\mathbf{75.7 \pm 1.42}$	+11.65%
Zipper	72.9 ± 1.20	$\mathbf{84.8 \pm 0.92}$	+16.32%

Table 6: Effect of the contrastive loss in the federated setting for VisA (AUROC).

Category	Federated BayPrAnoMeta	Federated Contrastive BayPrAnoMeta	% Gain
candle	**41.30 ± 1.03**	36.10 ± 1.00	-12.59
capsules	46.31 ± 1.11	**47.37 ± 1.09**	+2.29
cashew	84.27 ± 0.75	**85.94 ± 0.69**	+1.98
chewinggum	74.90 ± 0.61	**76.42 ± 0.61**	+2.03
fryum	**48.66 ± 0.81**	47.04 ± 0.80	-3.33
macaroni1	**60.47 ± 1.09**	57.84 ± 1.11	-4.35
macaroni2	**55.60 ± 0.83**	54.60 ± 0.82	-1.80
pcb1	89.65 ± 0.51	**89.90 ± 0.49**	+0.28
pcb2	**86.10 ± 0.56**	85.87 ± 0.58	-0.27
pcb3	66.37 ± 0.79	**67.20 ± 0.83**	+1.25
pcb4	78.28 ± 0.65	**78.73 ± 0.68**	+0.58
pipe_fryum	**69.29 ± 0.85**	67.08 ± 0.79	-3.19

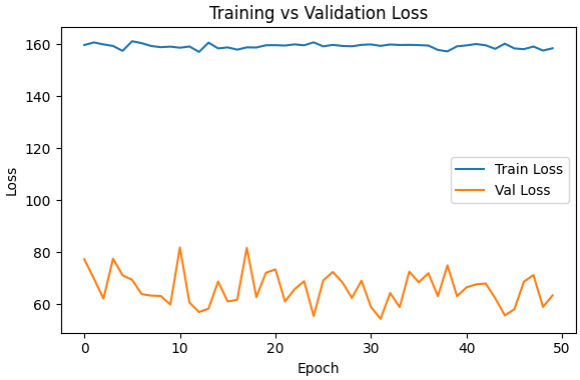
6.1 Ablation Experiment

Table 4 compares Contrastive BayPrAnoMeta, Federated BayPrAnoMeta, and Federated Contrastive BayPrAnoMeta. Incorporating supervised contrastive learning in the federated setting improves anomaly detection for several visually structured and texture-rich categories, including *Bottle*, *Carpet*, *Leather*, *Wood*, and *Zipper*. On the other hand, table 5 and 6 show the effect of contrastive learning for both datasets in the federated set-up. Notably, Federated Contrastive BayPrAnoMeta achieves substantial gains in AUROC on *Zipper* (+16.32 %), *Wood* (+11.65 %), and *Carpet* (+15.20 %) compared to its non-contrastive federated counterpart.

While improvements are not uniform across all categories (e.g., *Cable*, *Screw*, and *Pill*), this behavior is expected in few-shot anomaly detection, where contrastive objectives may over-regularize the embedding space when anomaly subtypes are scarce or visually similar. Overall, the results suggest that contrastive supervision acts as a representation stabilizer under client heterogeneity, leading to more robust anomaly scoring when supervision is distributed across heterogeneous clients.

7 Discussions on Training and Validation Loss Curves

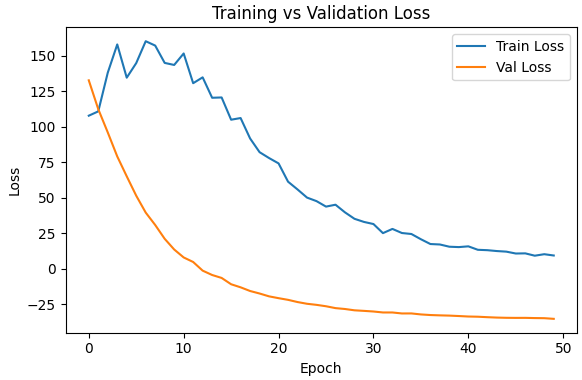
(a) Training vs Validation Loss (Classical Proto-MAML).



(b) Training vs Validation Loss (BayPrAnoMeta).

Figure 2: Comparison of training and validation loss curves of Classical Proto-MAML and BayPrAnoMeta (Bayesian Proto-MAML) for MVTec AD.

In this section we discuss training and validation loss curves to understand the optimization and convergence behavior of the proposed framework. From Figure 2, it is clear that BayPrAnoMeta exhibits higher training loss values because it optimizes a likelihood-based objective in a high-dimensional embedding space. However, the stable gap between training and validation loss, together with the evaluation metrics, suggests robust anomaly detection performance relative to Classical Proto-MAML. In episodic meta-learning, the training objective is evaluated over independently sampled tasks. Each episode induces its own support-set prototype (or Bayesian posterior), which causes the scale and difficulty of the loss to vary across episodes. As a result, optimization seeks a stable initialization that performs well on average across tasks, leading to convergence toward a stationary meta-loss rather than a monotonically decreasing training curve.



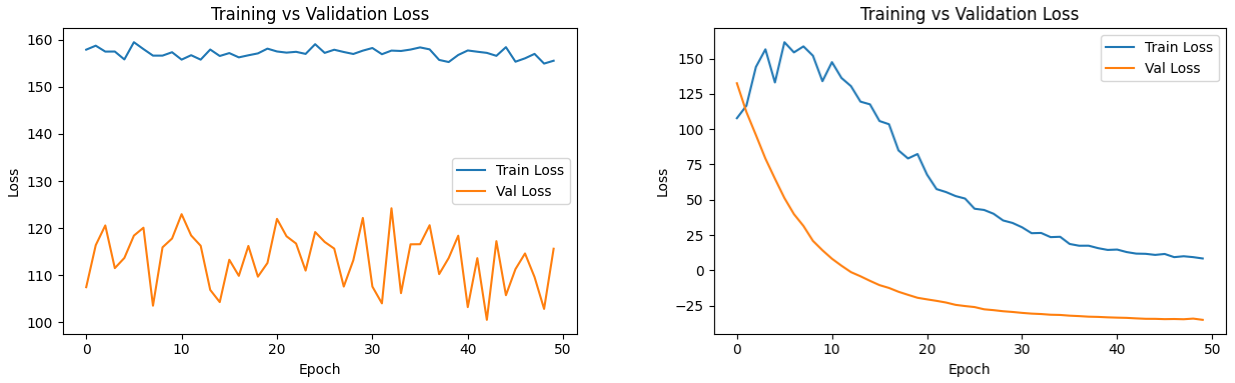
(a) Training vs Validation Loss (Federated BayPrAnoMeta).

(b) Training vs Validation Loss (Federated Contrastive BayPrAnoMeta).

Figure 3: Comparison of training and validation loss curves of Federated BayPrAnoMeta (Bayesian Proto-MAML) and Federated Contrastive BayPrAnoMeta (Bayesian Proto-MAML) for MVTec AD.

The loss curves in Figure 3 further emphasize the role of contrastive learning in the federated setting. As shown in 3a, training and validation losses decrease steadily, reflecting client-specific distribution shifts and limited inter-client feature alignment. In contrast, 3b exhibits a more consistent and smoother decline in both

training and validation loss, which validates improved optimization stability. The supervised contrastive loss acts as a cross-client regularizer, which aligns normal representations across clients despite non-i.i.d. data and episodic sampling. Thus, Figure 3 provides strong evidence supporting the inclusion of contrastive learning in the Federated BayPrAnoMeta framework, even when the performances are not uniform.



(a) Training vs Validation Loss (Centralized Contrastive BayPrAnoMeta).

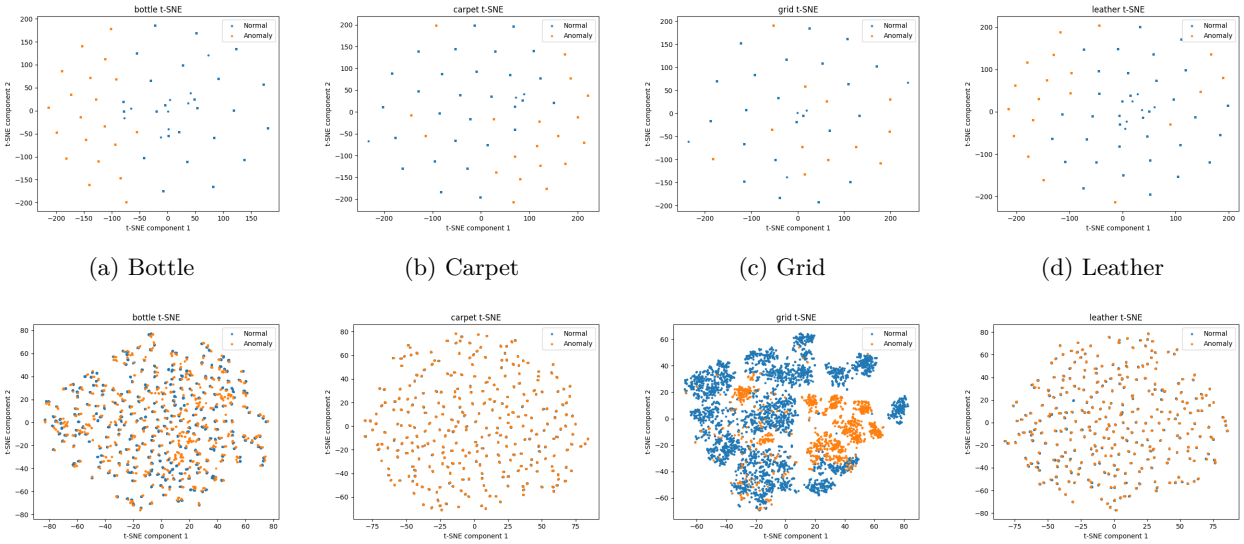
(b) Training vs Validation Loss (Federated Contrastive BayPrAnoMeta).

Figure 4: Comparison of training and validation loss curves of Centralized Contrastive BayPrAnoMeta (Bayesian Proto-MAML) and Federated Contrastive Supervised BayPrAnoMeta (Bayesian Proto-MAML) for MVTec AD.

Figure 4 illustrates the distinct behaviors of centralized and federated BayPrAnoMeta frameworks through their loss curves. The centralized Contrastive BayPrAnoMeta framework shows a nearly flat training loss and a highly unstable validation loss, suggesting ineffective optimization and poor generalization. Conversely, the Federated Contrastive BayPrAnoMeta pipeline exhibits a steadily decreasing training loss and a rapidly converging validation loss, which is indicative of stable meta-optimization and enhanced generalization across tasks.

8 Error Analysis

We analyze the learned embedding spaces using two-dimensional t-SNE projections of query embeddings to better understand the behavior of different models beyond performance metrics reported in Section 6. These visualizations would provide qualitative insights into class separability, representation collapse, and failure modes in few-shot anomaly detection.



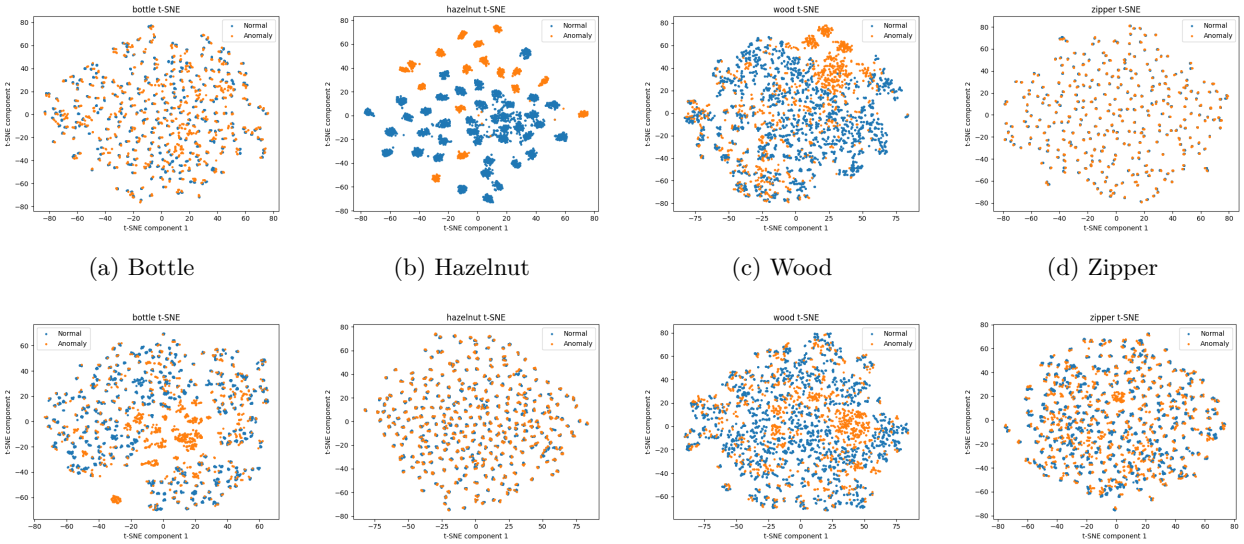
(a) Bottle (b) Carpet (c) Grid (d) Leather

(e) Bottle (f) Carpet (g) Grid (h) Leather

Figure 5: t-SNE visualizations for MVTec AD. Upper (a) - (d): BayPrAnoMeta (Centralized). Lower (e) - (h): Contrastive BayPrAnoMeta (Centralized).

For Contrastive BayPrAnoMeta (Centralized), Figure 5 reveals that t-SNE embeddings show evidence of contrastive collapse, where normal and anomalous query embeddings cluster too tightly or align along a low-variance manifold. This collapse is particularly evident in object categories such as *Bottle, Carpet, Grid*, and *Leather*; where contrastive learning aggressively enforces intra-class compactness based on limited support samples.

Under our data preparation protocol, the contrastive objective is trained with incomplete class information, since the support set contains only normal samples. Consequently, the contrastive loss can unintentionally pull normal and unseen anomaly embeddings closer together in the feature space, which can reduce class separability. This effect explains the observed drop in anomaly detection performance, and why Contrastive BayPrAnoMeta (Centralized) can underperform its non-contrastive counterpart despite stronger representation regularization.



(a) Bottle (b) Hazelnut (c) Wood (d) Zipper

(e) Bottle (f) Hazelnut (g) Wood (h) Zipper

Figure 6: t-SNE visualizations for MVTec AD. Upper (a) - (d): Contrastive BayPrAnoMeta (Centralized). Lower (e) - (h): Federated Contrastive BayPrAnoMeta.

The t-SNE visualizations of Figure 6 provide direct insight into the failure modes observed in Table 4. In the Contrastive BayPrAnoMeta (Centralized) setting, the t-SNE plots consistently show strong embedding collapse, where query normal samples and query anomalies overlap heavily in the latent space. In several object categories (e.g., *Bottle, Wood, Zipper*), anomaly embeddings are tightly clustered within or adjacent to normal clusters. This geometric overlap explains the near-random AUROC (Table 4).

In contrast, in the Federated Contrastive BayPrAnoMeta t-SNE embeddings exhibit greater inter-class dispersion and reduced collapse across most objects. While perfect separation is still not achieved, anomaly embeddings tend to form looser, partially separated regions away from the dominant normal manifold. This structural improvement is consistent with the gains reported in Table 4, particularly for clients *Bottle, Hazelnut, Wood*, and *Zipper*. The federated setting exposes the global model to heterogeneous normality distributions across clients to regularize contrastive objective, which prevents excessive contraction of the embedding space.

Overall, the t-SNE analysis (Figure 5 and Figure 6) confirms that contrastive learning is not flawed, but highly sensitive to supervision completeness and data heterogeneity. While centralized contrastive learning amplifies embedding collapse under weak anomaly semantics, federated learning takes care of this issue by diversifying the learned representation space, which justifies the inclusion of contrastive loss in the Federated BayPrAnoMeta framework despite its weaker centralized performance.

(a) Bottle (b) Hazelnut (c) Wood (d) Zipper

(e) Bottle (f) Hazelnut (g) Wood (h) Zipper

Figure 7: Misclassified Images of Federated Contrastive BayPrAnoMeta (MVTec AD dataset). Upper (a) - (d): False Positive images of clients *Bottle, Hazelnut, Wood, Zipper*. Lower (e) - (h): False Negative images of clients *Bottle, Hazelnut, Wood, Zipper*.

Figure 7 shows representative misclassified images for the *Bottle, Hazelnut, Wood,* and *Zipper* clients, and these errors are consistent with the contrastive embedding behavior observed in the corresponding t-SNE visualizations (Figure 6). For clients *Bottle* and *Hazelnut*, false positives predominantly occur in visually normal samples that exhibit specular highlights, subtle surface texture variations, or mild color inconsistencies.

In our proposed framework, we characterize normal embeddings by task specific Normal-Inverse-Wishart (NIW) distribution and heavy tailed distribution like Student-t distribution is used for anomaly detection. Localized intensity variations in such images cause the support embeddings to scatter more widely. This reduces the likelihood of visually valid normal samples under the normal model, $p_0(z \mid S_\tau)$, but these samples still receive a non-negligible likelihood under the fixed, heavy-tailed anomaly reference distribution, $p_1(z)$. This imbalance directly inflates the anomaly score, $s(x) = \log p_1(z) - \log p_0(z \mid S_\tau)$, leading to an increased rate of false positives. Conversely, false negatives for *Bottle* and *Hazelnut* arise when defects such as small dents, internal contaminations, or minor surface irregularities induce only weak deviations from the learned normal embedding distribution, resulting in insufficient contrast against the anomaly model.

For client Wood, errors are primarily driven by its highly heterogeneous texture. False positives occur when strong grain variations or illumination changes increase intra-class embedding dispersion, which mainly causes normal samples to fall outside the high-density region of the learned normal manifold. At the same time, genuine anomalies such as fine cracks or knots often preserve the overall texture statistics, leading to false negatives because these samples remain well explained by the posterior predictive distribution. In the case of client Zipper, false positives are commonly associated with specular reflections or slight misalignments of the metallic teeth, which perturb embeddings without corresponding to true defects. False negatives typically correspond to subtle, shape-preserving defects such as minor tooth deformations or stitching inconsistencies. Due to partial embedding collapse, these anomalies retain a high likelihood under $p_0(z \mid S_\tau)$, resulting in missed detections.

Table 7: Theoretical Training Complexity.

Method	Runtime Complexity	Memory Complexity	Main Extra Cost
MAML	$\mathcal{O}((TS+Q)P)$	$\mathcal{O}(P+TSP)$	Differentiable inner-loop adaptation
ProtoMAML	$\mathcal{O}((TS+Q)P+(S+Q)d) \approx \mathcal{O}((TS+Q)P)$	$\mathcal{O}(P+TSP+(S+Q)d)$	Prototype mean + distance scoring
BayPrAnoMeta	$\mathcal{O}((TS+Q)P+(S+Q)d^2)$	$\mathcal{O}(P+TSP+(S+Q)d+d^2)$	NIW posterior + Student-t likelihood
Contrastive BayPrAnoMeta	$\mathcal{O}((TS+Q)P+(S+Q)d^2+Q^2d)$	$\mathcal{O}(P+TSP+(S+Q)d+d^2+Q^2)$	Bayesian scoring + contrastive similarity matrix
Federated BayPrAnoMeta	$\mathcal{O}(CE[(TS+Q)P+(S+Q)d^2])$	$\mathcal{O}(P+TSP+(S+Q)d+d^2)$ + aggregation buffers	Client-wise adaptation + server aggregation
Federated Contrastive BayPrAnoMeta	$\mathcal{O}(CE[(TS+Q)P+(S+Q)d^2+Q^2d])$	$\mathcal{O}(P+TSP+(S+Q)d+d^2+Q^2)$ + aggregation buffers	Federated Bayesian + contrastive computation
PatchCore (25% coreset)	$\mathcal{O}(MP+\rho M^2d)$	$\mathcal{O}(Md)$	Patch embedding extraction + coreset

Table 8: Empirical Training-time computational cost for centralized and federated models.

Method	**# Params**	**Avg Time / Epoch (s)**	**Peak RAM (GB)**
MAML	11.18M	18.90	2.47
Proto-MAML	11.24M	15.68	2.00
BayPrAnoMeta	**11.24M**	**15.61**	**1.92**
Contrastive BayPrAnoMeta	11.24M	18.88	2.40
Federated BayPrAnoMeta	11.24M	52.34	3.55
Federated Contrastive BayPrAnoMeta	11.24M	52.38	3.40

9 Runtime Analysis

In addition to predictive performance, computational efficiency is very critical for few-shot anomaly detection techniques, specifically in the federated settings where the training is distributed across multiple clients. To assess the scalability of the proposed approach, we report the theoretical and empirical training and evaluation complexity all considered methods. The following notations are used throughout:

Let P denotes the number of trainable parameters, S the support set size, Q the query set size, d the embedding dimension, T the number of inner-loop adaptation steps, C the number of federated clients, E the number of local episodes per client, M the number of extracted patch embeddings, and ρ the coreset fraction.

To quantify practical overhead, we performed short run CPU profiling on the same set-up used in our experiments. However, we preserved the original computation per-task for each method.

For the centralized meta-learning models, viz. MAML, Proto-MAML, BayPrAnoMeta and Contrastive BayPrAnoMeta, training time, parameter count, average clock time and peak RAM were measured over 3 epochs, where we used 10 training episodes and 5 validation episodes per epoch. For the federated models, the same quantities were measured over 3 federated rounds with 3 episodes per client per round, while preserving client-wise local adaptation and server aggregation. For PatchCore, which does not use gradient-based episodic training, we profiled the complete feature extraction and memory-bank construction pipeline across all clients using 25% coreset selection. We further like to clarify that here peak RAM refers to the maximum CPU memory footprint reached at any single point during execution (i.e. the heaviest episode), rather than memory accumulated across all episodes or epochs. We also profiled the evaluation-time cost, where we used 20 episodes per client at the time of evaluation. Table 7 and 10 show complexity of theoretical training and theoretical evaluation respectively, whereas table 8 shows empirical training-time computational cost for both centralized and federated models. On the other hand, we report empirical training-time computational cost for PatchCore in table 9, along with empirical evaluation complexity of all the considered methods in tables 11 and 12. We report the evaluation complexity comparison tables (i.e., tables 10, 11, and 12) in section E of the Appendices.

These tables confirm that BayPrAnoMeta has runtime comparable to Proto-MAML, with an additional $(S+Q)d^2$ cost from covariance operations. This does not affect the dominant cost when $P \gg d^2$, which is typical in deep models. Empirically, runtime remains similar, with moderate increase in memory due to covariance terms. Evaluation-time overhead is small in runtime and primarily reflected in memory. In federated settings, cost scales linearly with the number of clients, consistent with standard federated meta-learning. PatchCore is faster at inference due to nearest-neighbor lookup, but requires a memory bank and a separate feature extraction pipeline.

Table 9: Empirical Training-time computational cost for PatchCore.

Method	# Params	Total Time (s)	Avg Time / Client (s)	Peak RAM (GB)
PatchCore (25% coreset)	11.24M	154.04	11.00	0.53

10 Conclusion

In this paper, we propose BayPrAnoMeta, a Bayesian extension of Proto-MAML for few-shot industrial anomaly detection. Our method replaces deterministic prototypes with Normal-Inverse-Wishart (NIW) prior to model class conditional embeddings. This change enables uncertainty-aware task adaptation and a principled, likelihood-based approach to anomaly scoring. We conduct extensive experiments on the MVTec AD and VisA dataset, which show that BayPrAnoMeta consistently outperforms both classical MAML and Proto-MAML across most object categories, demonstrating particular efficacy with challenging texture-based anomalies. Although contrastive learning degrades performance in centralized settings, we find that supervised contrastive regularization is beneficial in federated environments with strong client heterogeneity. This finding suggests that contrastive objectives primarily function to align client-specific representations during the aggregation process, rather than simply improving individual discrimination. In summary, our results establish Bayesian Proto-MAML as a robust meta-learning framework for industrial anomaly detection, which offers a principled methodology for combining Bayesian inference, meta-learning, and federated optimization in environments where data are scarce and highly heterogeneous.

References

Ali Alfatemi, Mohamed Rahouti, Zakirul Alam Bhuiyan, Aiman Solyman, and Mohammed Aledhari. Protomaml: A hybrid meta-learning approach integrating prototypical networks for data-efficient ddos attack detection. In *2025 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 1144–1149. IEEE, 2025. doi: https://doi.org/10.1109/IWCMC65282.2025.11059701.

Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9592–9600, 2019.

Fei Chen, Mi Luo, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. Federated meta-learning with fast convergence and efficient communication. *arXiv preprint arXiv:1802.07876*, 2018. doi: https://doi.org/10.48550/arXiv.1802.07876.

Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International conference on pattern recognition*, pp. 475–489. Springer, 2021. doi: https://doi.org/10.1007/978-3-030-68799-1_35.

Alireza Fallah, Aryan Mokhtari, and Asu Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. In *Advances in Neural Information Processing Systems*, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

Jonathan Gordon, John Bronskill, Matthias Bauer, Sebastian Nowozin, and Richard E Turner. Meta-learning probabilistic inference for prediction. *arXiv preprint arXiv:1805.09921*, 2018. doi: https://doi.org/10.48550/arXiv.1805.09921.

Erin Grant, Chelsea Finn, Sergey Levine, Trevor Darrell, and Thomas Griffiths. Recasting gradient-based meta-learning as hierarchical bayes. *arXiv preprint arXiv:1801.08930*, 2018. doi: https://doi.org/10.48550/arXiv.1801.08930.

Hadi Hojjati, Thi Kieu Khanh Ho, and Narges Armanfard. Self-supervised anomaly detection in computer vision and beyond: A survey and outlook. *Neural Networks*, 172:106106, 2024. doi: https://doi.org/10.1016/j.neunet.2024.106106.

T Hospedales, A Antoniou, P Micaelli, and A Storkey. Meta-learning in neural networks: A survey ieee transactions on pattern analysis and machine intelligence 1, 2021.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021. doi: https://doi.org/10.1561/2200000083.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Zhuo Li, Yuhao Yan, Xiangheng Wang, Yifei Ge, and Lin Meng. A survey of deep learning for industrial visual anomaly detection. *Artificial Intelligence Review*, 58(9):279, 2025.

Yufei Liang, Jiangning Zhang, Shiwei Zhao, Runze Wu, Yong Liu, and Shuwen Pan. Omni-frequency channel-selection representations for unsupervised anomaly detection. *IEEE Transactions on Image Processing*, 32:4327–4340, 2023. doi: https://doi.org/10.1109/TIP.2023.3293772.

Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. *Advances in neural information processing systems*, 31, 2018.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. PMLR, 2017.

Kevin P Murphy. Conjugate bayesian analysis of the gaussian distribution. *def*, 1(2σ2):16, 2007.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2):1–38, 2021. doi: https://doi.org/10.1145/3439950.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14318–14328, June 2022.

Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pp. 4393–4402. PMLR, 2018.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H.S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2018.

Tran Dinh Tien, Anh Tuan Nguyen, Nguyen Hoang Tran, Ta Duc Huy, Soan Duong, Chanh D Tr Nguyen, and Steven QH Truong. Revisiting reverse distillation for anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24511–24520, 2023.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, 2016.

Yang Zou, Jongheon Jeong, Latha Pemula, Dongqing Zhang, and Onkar Dabeer. Spot-the-difference self-supervised pre-training for anomaly detection and segmentation. In *European conference on computer vision*, pp. 392–408. Springer, 2022. doi: https://doi.org/10.1007/978-3-031-20056-4_23.

The appendices provide additional methodological, algorithmic, and theoretical details that complement the main text and are included to ensure clarity and reproducibility.

A Classical Proto-MAML for few-shot anomaly detection

Motivation In modern manufacturing environments, visual anomaly detection systems are expected to identify subtle defects such as scratches, dents, misalignments, and contaminations before they propagate downstream and cause costly failures. Unlike conventional classification problems, these defects are inherently rare, visually diverse, and often unknown at deployment time. As a result, collecting large, well-balanced, and exhaustively labeled datasets of all possible defect types is practically infeasible. This setting naturally violates the assumptions of classical supervised learning and motivates few-shot learning formulations.

Few-shot anomaly detection can be naturally framed as a meta-learning problem, where each task corresponds to learning a normality model from a small number of examples and detecting deviations from it. In prototypical networks, each task is represented by a prototype computed as the mean of support embeddings, and query samples are classified based on their distance from this prototype (Snell et al., 2017). When applied to anomaly detection, the support set typically contains only normal samples, and anomalies are identified as samples whose embeddings lie far from the normal prototype in the learned metric space. This paradigm has been widely adopted in industrial anomaly detection, particularly on datasets such as MVTec AD, due to its simplicity and effectiveness under limited supervision (Bergmann et al., 2019; Ruff et al., 2018; Pang et al., 2021).

Proto-MAML, a hybridization of prototypical networks and model-agnostic meta-learning (MAML), extends this idea by incorporating task-specific adaptation through gradient-based meta-learning (Finn et al., 2017). In this formulation, the embedding network is meta-trained such that a small number of gradient steps on the support set minimizes the dispersion of normal embeddings around their prototype. After adaptation, anomaly scores for query samples are computed using distance-based metrics in the adapted embedding space that enables rapid adaptation to new object categories.

Despite its success, classical Proto-MAML for anomaly detection has certain limitations. First, the prototype is treated as a point estimate, ignoring uncertainty arising from small support sets. Second, Euclidean distance–based scoring lacks probabilistic interpretability. Finally, discriminative objectives dominate the learning process, providing limited uncertainty quantification—an essential requirement in safety-critical industrial systems (Malinin & Gales, 2018).

These limitations motivate a probabilistic reformulation of Proto-MAML, where normality is modeled as a distribution rather than a point, and anomaly detection is performed using likelihood-based inference. In the following section, we introduce BayPrAnoMeta, a Bayesian extension of Proto-MAML that addresses these shortcomings while preserving its few-shot adaptability. We study few-shot industrial anomaly detection under a meta-learning framework.

Prototype estimation Given support embeddings $z_i = f_\theta(x_i)$, the task-specific prototype is computed as

$$c_\mathcal{T} = \frac{1}{K} \sum_{i=1}^{K} z_i.$$

Inner-loop adaptation Proto-MAML adapts the embedding network by minimizing the dispersion of support embeddings around the prototype:

$$\mathcal{L}_{\text{inner}}(\theta; \mathcal{S}_{\mathcal{T}}) = \frac{1}{K} \sum_{i=1}^{K} \|f_\theta(x_i) - c_{\mathcal{T}}\|_2^2 .$$

The task-adapted parameters are obtained via one or more gradient descent steps: $\theta_{\mathcal{T}} = \theta - \alpha \nabla_\theta \mathcal{L}_{\text{inner}}(\theta; \mathcal{S}_{\mathcal{T}})$.

Query scoring For a query sample x_j, the adapted embedding $z_j = f_{\theta_{\mathcal{T}}}(x_j)$ is compared to the prototype using squared Euclidean distance. The anomaly logit is defined as: $\ell_j = -\|z_j - c_{\mathcal{T}}\|_2^2$.

Outer-loop meta-objective Given labels $y_j \in \{0, 1\}$, the query loss is defined using binary cross-entropy with logits:

$$\mathcal{L}_{\text{query}} = \sum_j \text{BCEWithLogits}(\ell_j, y_j),$$

which is equivalent to

$$\mathcal{L}_{\text{query}} = -\sum_j \left[y_j \log \sigma(\ell_j) + (1 - y_j) \log(1 - \sigma(\ell_j)) \right],$$

where $\sigma(\cdot)$ denotes the sigmoid function. The meta-learning objective minimizes the expected query loss across tasks:

$$\min_\theta \ \mathbb{E}_{\mathcal{T} \sim p(\mathcal{T})} \left[\mathcal{L}_{\text{query}}(\theta_{\mathcal{T}}) \right],$$

with gradients propagated through the inner-loop update, and the meta-parameters updated using gradient descent with meta-learning rate β:

$$\theta \leftarrow \theta - \beta \nabla_\theta \mathbb{E}_{\mathcal{T} \sim p(\mathcal{T})} \left[\mathcal{L}_{\text{query}}(\theta_{\mathcal{T}}) \right].$$

Testing and anomaly detection At test time, for a previously unseen task $\mathcal{T}$, a support set $\mathcal{S}_{\mathcal{T}}$ consisting only of normal samples is provided. The prototype $c_{\mathcal{T}}$ is computed from the support embeddings, and the embedding network is adapted via the Proto-MAML inner loop to obtain $\theta_{\mathcal{T}}$. For a test sample x^*, the anomaly score is defined as

$$s(x^*) = \|f_{\theta_{\mathcal{T}}}(x^*) - c_{\mathcal{T}}\|_2^2 .$$

Samples with larger scores are considered more likely to be anomalous.

B BayPrAnoMeta Algorithm

This appendix provides detailed algorithm for the Meta-training and meta-testing of BayPrAnoMeta, introduced in Section 3.1.

Algorithm 2 BayPrAnoMeta (Bayesian Proto-MAML): Meta-Training

Input: Task distribution $p(\mathcal{T})$; embedding network f_θ; NIW prior $(\mu_0, \kappa_0, \Lambda_0, \nu_0)$; step sizes α, β
while not converged **do**
  Sample tasks $\mathcal{T} \sim p(\mathcal{T})$
  for all tasks $\mathcal{T}$ **do**
    Sample episode $(\mathcal{S}_\mathcal{T}, \mathcal{Q}_\mathcal{T})$
    Compute embeddings $z_i = f_\theta(x_i)$ for $x_i \in \mathcal{S}_\mathcal{T}$
    $\bar{z} = \frac{1}{K}\sum_i z_i$
    $S_z = \sum_i (z_i - \bar{z})(z_i - \bar{z})^\top$
    $\kappa_n = \kappa_0 + K$
    $\nu_n = \nu_0 + K$
    $\mu_n = \frac{\kappa_0 \mu_0 + K\bar{z}}{\kappa_n}$
    $\Lambda_n = \Lambda_0 + S_z + \frac{\kappa_0 K}{\kappa_n}(\bar{z} - \mu_0)(\bar{z} - \mu_0)^\top$
    Define $p_0(z \mid \mathcal{S}_\mathcal{T}) = \text{Student-}t(\mu_n, \Lambda_n, \nu_n)$
    $\mathcal{L}_{\text{inner}}^{\text{Bayes}} = -\sum_{x_i \in \mathcal{S}_\mathcal{T}} \log p_0(f_\theta(x_i))$
    $\theta_\mathcal{T} = \theta - \alpha \nabla_\theta \mathcal{L}_{\text{inner}}^{\text{Bayes}}$
    for all $(x_j, y_j) \in \mathcal{Q}_\mathcal{T}$ **do**
      $z_j = f_{\theta_\mathcal{T}}(x_j)$
      $\mathcal{L}_{\text{query}}^{\text{Bayes}} = -(1 - y_j)\log p_0(z_j) - y_j \log p_1(z_j)$
    end for
  end for
  $\theta \leftarrow \theta - \beta \nabla_\theta \sum_\mathcal{T} \sum \mathcal{L}_{\text{query}}^{\text{Bayes}}$
end while

Algorithm 3 BayPrAnoMeta (Bayesian Proto-MAML): Meta-Testing

Input: Initialization θ; support set $\mathcal{S}_\mathcal{T}$; test sample x^*; threshold τ
Compute embeddings $z_i = f_\theta(x_i)$ for $x_i \in \mathcal{S}_\mathcal{T}$
$\bar{z} = \frac{1}{K}\sum_i z_i$
$S_z = \sum_i (z_i - \bar{z})(z_i - \bar{z})^\top$
$\kappa_n = \kappa_0 + K$
$\nu_n = \nu_0 + K$
$\mu_n = \frac{\kappa_0 \mu_0 + K\bar{z}}{\kappa_n}$
$\Lambda_n = \Lambda_0 + S_z + \frac{\kappa_0 K}{\kappa_n}(\bar{z} - \mu_0)(\bar{z} - \mu_0)^\top$
Define $p_0(z \mid \mathcal{S}_\mathcal{T}) = \text{Student-}t(\mu_n, \Lambda_n, \nu_n)$
$\mathcal{L}_{\text{inner}}^{\text{Bayes}} = -\sum_{x_i \in \mathcal{S}_\mathcal{T}} \log p_0(f_\theta(x_i))$
$\theta_\mathcal{T} = \theta - \alpha \nabla_\theta \mathcal{L}_{\text{inner}}^{\text{Bayes}}$
$z^* = f_{\theta_\mathcal{T}}(x^*)$
$s(x^*) = \log p_1(z^*) - \log p_0(z^* \mid \mathcal{S}_\mathcal{T})$
if $s(x^*) > \tau$ **then**
  Anomaly
else
  Normal
end if

C Proofs of results

This appendix provides the proofs of the proposition on the well-posedness of the Student-t posterior predictive distribution as outlined in Proposition 3.1. We also provide the proofs of the smoothness lemma given in Lemma 4.5, and the main result on optimization-theoretic convergence guarantee for the federated training procedure in Algorithm 1 as given by Theorem 4.6.

Proof of Proposition 3.1. Fix $K \geq 1$ and $\mathcal{S}$. Since $S_z \succeq 0$ and $(\bar{z}-\mu_0)(\bar{z}-\mu_0)^\top \succeq 0$, the update (1) implies

$$\Lambda_n = \Lambda_0 + \underbrace{S_z}_{\succeq 0} + \underbrace{\frac{\kappa_0 K}{\kappa_0 + K}(\bar{z}-\mu_0)(\bar{z}-\mu_0)^\top}_{\succeq 0} \succeq \Lambda_0 \succ 0.$$

Hence Λ_n is positive definite. Because $\kappa_n = \kappa_0 + K > 0$ and $\nu_{\text{pred}} = \nu_0 + K - d + 1 > 0$ by assumption, the scalar factor in Σ_n strictly positive. Therefore Σ_n is positive definite, proving (i).

A multivariate Student-t distribution with degrees of freedom $\nu_{\text{pred}} > 0$ and scale $\Sigma_n \succ 0$ is a proper density on $\mathbb{R}^d$, and its log-density is finite everywhere, proving (ii).

□

Corollary C.1. *The predictive scale is uniformly bounded below as*

$$\Sigma_n \succeq \frac{\kappa_n + 1}{\kappa_n(\nu_n - d + 1)}\,\Lambda_0 \;\succeq\; \frac{1}{\nu_0 + K - d + 1}\,\Lambda_0.$$

Proof. For the lower bound, since $\Lambda_n \succeq \Lambda_0$, multiplying by the positive scalar $\frac{\kappa_n+1}{\kappa_n(\nu_n-d+1)}$ yields

$$\Sigma_n = \frac{\kappa_n + 1}{\kappa_n(\nu_n - d + 1)}\,\Lambda_n \succeq \frac{\kappa_n + 1}{\kappa_n(\nu_n - d + 1)}\,\Lambda_0.$$

Finally, observe that $\frac{\kappa_n+1}{\kappa_n} \geq 1$, so

$$\frac{\kappa_n + 1}{\kappa_n(\nu_n - d + 1)} \geq \frac{1}{\nu_n - d + 1} = \frac{1}{\nu_0 + K - d + 1}.$$

□

Corollary C.2. *Under the conditions of Proposition 3.1, Assumption 4.4 holds with $\varepsilon = \lambda_{\min}(\Lambda_0)/(\nu_0 + K - d + 1)$. Also, the uniform lower bound in Corollary C.1 implies a uniform upper bound on the inverse scale:*

$$\|\Sigma_n^{-1}\| \leq \left(\frac{\nu_0 + K - d + 1}{\lambda_{\min}(\Lambda_0)}\right),$$

where $\lambda_{\min}(\Lambda_0) > 0$ is the minimum eigenvalue of Λ_0.

Combined with Lemma 4.5, this shows that curvature and gradient Lipschitz constants of the Bayesian likelihood terms remain controlled in the few-shot regime because the prior prevents degeneracy of the predictive scale matrix.

Proof of Lemma 4.5. Let $\delta(z) = (z-\mu)^\top \Sigma^{-1}(z-\mu)$. Up to an additive constant, the Student-t negative log-likelihood has the form

$$\ell(z) = \frac{\nu + d}{2}\log\Big(1 + \frac{1}{\nu}\delta(z)\Big) + \text{const}.$$

A direct calculation yields

$$\nabla \ell(z) = \frac{\nu + d}{\nu + \delta(z)}\,\Sigma^{-1}(z-\mu),$$

and

$$\nabla^2 \ell(z) = \frac{\nu + d}{\nu + \delta(z)}\,\Sigma^{-1} - \frac{2(\nu + d)}{(\nu + \delta(z))^2}\,\Sigma^{-1}(z-\mu)(z-\mu)^\top \Sigma^{-1}.$$

Since $\delta(z) \geq 0$, we have $(\nu + \delta(z))^{-1} \leq \nu^{-1}$ and $(\nu + \delta(z))^{-2} \leq \nu^{-2}$. On the set $\|z-\mu\| \leq R_z$,

$$\big\|\Sigma^{-1}(z-\mu)(z-\mu)^\top \Sigma^{-1}\big\| \leq \|\Sigma^{-1}\|^2\,\|z-\mu\|^2 \leq \|\Sigma^{-1}\|^2 R_z^2.$$

Combining these bounds gives the stated Hessian norm bound. The final inequality uses $\|\Sigma^{-1}\| \leq 1/\varepsilon$ when $\Sigma \succeq \varepsilon I_d$. □

Proof of Theorem 4.6. Because F is L-smooth (Assumption 4.1), for any update $\theta^{r+1} = \theta^r - \eta\bar{g}^r$ we have the descent inequality

$$F(\theta^{r+1}) \leq F(\theta^r) - \eta\langle\nabla F(\theta^r), \bar{g}^r\rangle + \frac{L\eta^2}{2}\|\bar{g}^r\|^2. \tag{12}$$

Take conditional expectation given θ^r. By Assumption 4.2 and unbiased client sampling, $\mathbb{E}[\bar{g}^r \mid \theta^r] = \nabla F(\theta^r)$, hence

$$\mathbb{E}\big[\langle\nabla F(\theta^r), \bar{g}^r\rangle \mid \theta^r\big] = \|\nabla F(\theta^r)\|^2.$$

Moreover,

$$\mathbb{E}\big[\|\bar{g}^r\|^2 \mid \theta^r\big] = \|\nabla F(\theta^r)\|^2 + \mathbb{E}\big[\|\bar{g}^r - \nabla F(\theta^r)\|^2 \mid \theta^r\big].$$

Decompose the deviation as

$$\bar{g}^r - \nabla F(\theta^r) = \underbrace{\frac{1}{|\mathcal{C}_r|}\sum_{c\in\mathcal{C}_r}\big(g_c^r - \nabla F_c(\theta^r)\big)}_{\text{stochastic episode noise}} + \underbrace{\frac{1}{|\mathcal{C}_r|}\sum_{c\in\mathcal{C}_r}\big(\nabla F_c(\theta^r) - \nabla F(\theta^r)\big)}_{\text{client heterogeneity}}.$$

Using $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$, Jensen's inequality, and Assumptions 4.2–4.3, we obtain

$$\mathbb{E}\big[\|\bar{g}^r - \nabla F(\theta^r)\|^2 \mid \theta^r\big] \leq \frac{\sigma^2}{|\mathcal{C}_r|} + \zeta^2.$$

Substituting these bounds into (12) and taking total expectation gives

$$\mathbb{E}[F(\theta^{r+1})] \leq \mathbb{E}[F(\theta^r)] - \eta\left(1 - \frac{L\eta}{2}\right)\mathbb{E}\|\nabla F(\theta^r)\|^2 + \frac{L\eta^2}{2}\left(\frac{\sigma^2}{|\mathcal{C}_r|} + \zeta^2\right).$$

For $\eta \leq 1/L$, we have $1 - \frac{L\eta}{2} \geq \frac{1}{2}$, hence

$$\frac{\eta}{2}\,\mathbb{E}\|\nabla F(\theta^r)\|^2 \leq \mathbb{E}[F(\theta^r)] - \mathbb{E}[F(\theta^{r+1})] + \frac{L\eta^2}{2}\left(\frac{\sigma^2}{|\mathcal{C}_r|} + \zeta^2\right).$$

Summing over $r = 0, \ldots, R-1$ telescopes:

$$\frac{\eta}{2}\sum_{r=0}^{R-1}\mathbb{E}\|\nabla F(\theta^r)\|^2 \leq F(\theta^0) - \mathbb{E}[F(\theta^R)] + \frac{L\eta^2 R}{2}\left(\frac{\sigma^2}{|\mathcal{C}_r|} + \zeta^2\right).$$

Lower bound $\mathbb{E}[F(\theta^R)] \geq F^\star$ and divide by $\eta R/2$ to obtain (11). □

D F1 Score Computation (Optimal Threshold)

For each query sample x_j, the model produces a continuous anomaly score $s_j \in \mathbb{R}$, where larger values indicate higher anomaly likelihood. A threshold u maps scores to binary predictions

$$\hat{y}_j(u) = \begin{cases} 1, & s_j \geq u, \\ 0, & \text{otherwise.} \end{cases}$$

For each candidate threshold u, precision and recall are computed, yielding

$$\mathrm{F1}(u) = \frac{2\,\mathrm{TP}(u)}{2\,\mathrm{TP}(u) + \mathrm{FP}(u) + \mathrm{FN}(u)}.$$

The optimal threshold is selected as $u^\star = \arg\max_u \mathrm{F1}(u)$, and the reported F1 score is $\mathrm{F1}^\star = \mathrm{F1}(u^\star)$, averaged across test episodes.

E Evaluation Complexity Comparison Tables (Runtime Analysis)

In this section, we present the comparisons of the theoretical and empirical evaluation complexities.

Table 10: Theoretical Evaluation Complexity.

Method	Runtime Complexity	Memory Complexity	Main Cost
MAML	$\mathcal{O}((TS+Q)P)$	$\mathcal{O}(P+TSP)$	Inner-loop adaptation
ProtoMAML	$\mathcal{O}((TS+Q)P)$	$\mathcal{O}(P+TSP+(S+Q)d)$	Prototype scoring
BayPrAnoMeta	$\mathcal{O}((TS+Q)P+(S+Q)d^2)$	$\mathcal{O}(P+TSP+(S+Q)d+d^2)$	Bayesian scoring
Contrastive BayPrAnoMeta	$\mathcal{O}((TS+Q)P+(S+Q)d^2+Q^2d)$	$\mathcal{O}(P+TSP+(S+Q)d+d^2+Q^2)$	Bayesian + contrastive
Federated BayPrAnoMeta	$\mathcal{O}(C[(TS+Q)P+(S+Q)d^2])$	Same as above	Per-client evaluation
Federated Contrastive BayPrAnoMeta	$\mathcal{O}(C[(TS+Q)P+(S+Q)d^2+Q^2d])$	Same as above	Federated + contrastive
PatchCore (25% coreset)	$\mathcal{O}(QP+QMd)$	$\mathcal{O}(Md)$	Patch extraction + nearest-neighbor search

Table 11: Empirical Evaluation Complexity Comparison.

Method	**# Params**	**Avg Time / Episode (s)**	**Peak RAM (GB)**
MAML	11.18M	0.645	1.24
ProtoMAML	11.24M	0.562	0.77
BayPrAnoMeta	**11.24M**	**0.640**	**1.80**
Contrastive BayPrAnoMeta	11.24M	0.642	2.21
Federated BayPrAnoMeta	11.24M	0.655	1.58
Federated Contrastive BayPrAnoMeta	11.24M	0.658	2.26

Table 12: Empirical Evaluation Complexity Comparison for PatchCore.

Method	**# Params**	**Total Time (s)**	**Avg Time (s)**	**Peak RAM (GB)**
PatchCore (25% coreset)	11.24M	30.39	0.043/image	1.23