

This is an Open Access document downloaded from ORCA, Cardiff University's institutional repository: <https://orca.cardiff.ac.uk/id/eprint/189541/>

This is the author's version of a work that was submitted to / accepted for publication.

Citation for final published version:

Balinsky, Alexander D., Krzeminski, Dominik and Balinsky, Alexander 2026. Conformal prediction for privacy-preserving machine learning: uncertainty quantification on encrypted data. *Applied Mathematics & Information Sciences* 20 (5) , pp. 1261-1268. 10.18576/amis/200511

Publishers page: <http://dx.doi.org/10.18576/amis/200511>

Please note:

Changes made as a result of publishing processes such as copy-editing, formatting and page numbers may not be reflected in this version. For the definitive version of this publication, please refer to the published source. You are advised to consult the publisher's version if you wish to cite this paper.

This version is being made available in accordance with publisher policies. See <http://orca.cf.ac.uk/policies.html> for usage policies. Copyright and moral rights for publications made available in ORCA are retained by the copyright holders.



Conformal Prediction for Privacy-Preserving Machine Learning: Uncertainty Quantification on Encrypted Data

Alexander D. Balinsky¹, Dominik Krzemiński² and Alexander A. Balinsky³

¹Department of Computing, Imperial College London, London, UK

²Cohere Labs Community

³School of Mathematics, Cardiff University, Cardiff, UK

Corresponding author: BalinskyA@cardiff.ac.uk

Abstract

We study whether Conformal Prediction (CP), a distribution-free framework for quantifying the uncertainty of machine-learning predictions, can be applied directly on encrypted data, without ever decrypting it. The motivation is privacy-preserving learning: ideally a model should produce calibrated, statistically valid prediction sets while the underlying data remains hidden. Our key observation is theoretical: because a fixed-key deterministic cipher is a single, fixed transformation applied identically to every record, it preserves *exchangeability*, the only assumption CP requires. CP guarantees therefore transfer to the encrypted domain unchanged. We test this on AES-encrypted MNIST and compare the classical p -value conformal predictor against the more recent e -value (BB) predictor of Balinsky and Balinsky. Empirically, a simple feed-forward network trained only on deterministically encrypted images still reaches 33.58% test accuracy, far above the near-chance 11.35% obtained when each image is encrypted with its own key (which destroys the consistent plaintext-to-ciphertext mapping and reduces learning to chance). At a nominal miscoverage of $\alpha = 0.4$ (a 60% coverage target), the e -value predictor is highly conservative: its guarantee of at least 60% coverage is met with a realized coverage of 97.46% (4873/5000), at the cost of large prediction sets; the p -value predictor produces much smaller sets but realizes only 59.4% coverage. These results demonstrate the feasibility of uncertainty quantification on encrypted data.

Keywords: conformal prediction, e -test statistics, exchangeability, encryption, privacy-preserving machine learning

1 Introduction

Modern machine-learning (ML) systems are increasingly deployed in high-stakes settings, e.g. health-care, finance, autonomous control, where a simple point prediction is not enough: we need to know *how much to trust it*. Conformal Prediction (CP) answers this need by turning any predictive model into one that outputs a *prediction set* guaranteed, with a user-chosen probability, to contain the true label [1]. Its central appeal is that this guarantee is *distribution-free*: it holds for any data-generating distribution P , requiring only that the data be *exchangeable* (a notion we define precisely in Section 2). This makes CP attractive precisely when distribution is unavailable [2, 3].

The main tool of CP is the *nonconformity measure*, a function that scores how strange a candidate (input, label) pair looks relative to a reference set. Ranking these scores and calibrating a threshold produces sets with the required coverage. CP has been developed for regression, classification, anomaly detection and time-series forecasting [4], and scaled to large data through

the *split* (inductive) construction [5], with applications across medicine, finance and safe autonomy [6, 7, 8].

A more recent development replaces the p -value with an e -value. The conformal e -prediction and *BB-predictor* of Balinsky and Balinsky [9] relax exchangeability to the weaker condition of cycle invariance, while Gauthier et al. show that e -value CP can give tighter Type-I-error control and adapt better in online or adversarial regimes. We build directly on the BB-predictor in this paper.

Learning under encryption. We ask a question that, to our knowledge, has not been posed in the CP literature: *can CP operate when the data is encrypted and never decrypted?* We model encryption as a map from a *message space* to a *ciphertext space* controlled by a *key*. Using one fixed key for every record makes the map *deterministic*: identical inputs always produce identical ciphertexts. This single fact has two consequences that drive the paper. First (Section 3), a fixed deterministic map preserves exchangeability, so every CP guarantee survives encryption untouched. Second, that same determinism is what lets a model find structure in data that looks, to the eye, like noise.

Contributions.

- We give a short, self-contained proof (Lemma 1) that any fixed deterministic encryption preserves exchangeability, and hence the validity of both p -value and e -value CP, on encrypted data.
- We reproduce, with full proof for self-containedness, the e -value BB-predictor of Balinsky and Balinsky [9] (Theorem 1) and explain its intuition.
- On AES-encrypted MNIST we demonstrate that learning and valid CP are both feasible in the encrypted domain, and we quantify a clear compactness–reliability trade-off between the p - and e -value predictors.

2 Background and Related Work

2.1 Conformal prediction in one page

Let observations be pairs $z = (x, y)$ with x an input and $y \in \mathcal{Y}$ a label. Throughout we write $\alpha \in (0, 1)$ for the *miscoverage level*; the target coverage is $1 - \alpha$.

Exchangeability. A sequence of random variables $L_1, \dots, L_{n+1}$ is *exchangeable* if its joint distribution is invariant under relabelling of the indices,

$$(L_1, \dots, L_{n+1}) \stackrel{d}{=} (L_{\sigma(1)}, \dots, L_{\sigma(n+1)})$$

for every permutation σ , where $\stackrel{d}{=}$ denotes equality of joint distribution. Intuitively, the order in which the points arrive carries no information. This is strictly weaker than the i.i.d. assumption (i.i.d. implies exchangeable, but not conversely) and is the only assumption standard CP needs.

Nonconformity measure. A nonconformity measure assigns each observation a score L_i measuring how poorly it fits the rest of the data: small L_i means typical, large L_i means strange. In supervised learning a natural choice is the model’s loss on (x_i, y_i) : a correct, confident prediction yields a small loss, an incorrect one a large loss.

Inductive (split) CP. Inductive CP separates the data into a training set (to fit the model), a *calibration set* (to learn the score threshold), and a test set. The threshold is read off the empirical distribution of calibration scores; this makes CP cheap and is the variant we use.

2.2 From p -values to e -values

Classical CP is built on a p -value: the rank of the new score among the others (Lemma 2). An e -value is a different testing approach: a non-negative statistic E with $\mathbb{E}(E) \leq 1$. Markov's inequality then gives, for any $\alpha > 0$,

$$P(E \geq \frac{1}{\alpha}) \leq \alpha.$$

Thus large e -values are rare under the null; e.g. at $\alpha = 0.05$ the event $E \geq 20$ has probability at most 5%.

2.3 Privacy-preserving learning and encryption

A standard route to confidential ML is to compute on data that has been transformed so the raw values are hidden. Our transformation is the Advanced Encryption Standard (AES) [10] run with a fixed symmetric key, making it deterministic. We stress at the outset that determinism is a double-edged sword, it enables learning but weakens confidentiality, and we return to this in Section 3.4. Unlike homomorphic-encryption approaches, which preserve algebraic structure so computation can proceed on ciphertexts by design, we make no structural assumption on the cipher: our guarantees follow purely from exchangeability.

3 Theoretical Framework

This section contains the paper's conceptual core. We first show that encryption is invisible to CP (Section 3.1); we then state the two predictors we compare (Sections 3.2–3.3); finally we discuss the security cost (Section 3.4).

3.1 Exchangeability survives deterministic encryption

Lemma 1 (Invariance under a fixed map). *Let $\varphi : \mathcal{Z} \rightarrow \mathcal{Z}'$ be any fixed measurable map and let $Z_1, \dots, Z_{n+1}$ be exchangeable. Then $\varphi(Z_1), \dots, \varphi(Z_{n+1})$ are exchangeable.*

Proof. Define $g(z_1, \dots, z_{n+1}) = (\varphi(z_1), \dots, \varphi(z_{n+1}))$, which applies φ coordinatewise and therefore commutes with any permutation σ :

$$g(z_{\sigma(1)}, \dots, z_{\sigma(n+1)}) = (\varphi(z_{\sigma(1)}), \dots, \varphi(z_{\sigma(n+1)})).$$

Since $(Z_1, \dots, Z_{n+1}) \stackrel{d}{=} (Z_{\sigma(1)}, \dots, Z_{\sigma(n+1)})$ and g is a fixed measurable function, applying g to both sides preserves equality in distribution. Hence

$$(\varphi(Z_1), \dots, \varphi(Z_{n+1})) \stackrel{d}{=} (\varphi(Z_{\sigma(1)}), \dots, \varphi(Z_{\sigma(n+1)}))$$

for every σ , which is exactly exchangeability. $\square$

Remark 1. AES with a fixed key and fixed initialization vector is precisely such a fixed φ . Consequently both CP predictors below remain exactly valid when applied to encrypted records: no re-derivation is needed. This is the sense in which CP does not notice encryption. The same argument fails for *per-record* keys, where each Z_i is hit by a different map and the common distribution, and with it learnability, is destroyed.

3.2 The p -value predictor

Lemma 2 ([1]). *Suppose $L_1, \dots, L_{n+1}$ are exchangeable and set $U = \#\{i : L_i \geq L_{n+1}\}$, the number of scores at least as large as the last. Then $P\{U/(n+1) > \alpha\} \geq 1 - \alpha$ for all $\alpha \in [0, 1]$.*

In words, $U/(n+1)$ is the conformal p -value, and it is (super-)uniform, which is what makes the rank threshold valid. In inductive CP one keeps a label y in the prediction set whenever its score does not exceed the $\lceil (1 - \alpha)(n+1) \rceil$ -th smallest calibration score.

3.3 The e -value predictor (BB-predictor)

Theorem 1 ([9]). *Let $L_1, \dots, L_{n+1}$ be exchangeable non-negative random variables and set*

$$F = \frac{L_{n+1}}{(\sum_{j=1}^{n+1} L_j)/(n+1)}.$$

Then $\mathbb{E}(F) = 1$, and consequently $P\{F \geq 1/\alpha\} \leq \alpha$ for every $\alpha > 0$.

Proof. For $i = 1, \dots, n+1$ define $F_i = L_i/[(\sum_j L_j)/(n+1)]$. Exchangeability makes the L_i identically distributed, hence so are the F_i ; in particular they share a common mean, $\mathbb{E}(F_i) = \mathbb{E}(F)$. Summing, $\mathbb{E}(\sum_{i=1}^{n+1} F_i) = (n+1)\mathbb{E}(F)$. But by construction $\sum_{i=1}^{n+1} F_i = (n+1)$ identically (the denominator is the average of the numerators), so $(n+1)\mathbb{E}(F) = n+1$ and $\mathbb{E}(F) = 1$. Since $F \geq 0$ with unit mean, F is an e -value and Markov's inequality gives $P\{F \geq 1/\alpha\} \leq \alpha$. $\square$

Rearranging $F \geq 1/\alpha$ yields the operational form. Writing $\bar{L}_n = \frac{1}{n} \sum_{i=1}^n L_i$ for the calibration mean:

Corollary 1. *Under the hypotheses of Theorem 1, for every $\alpha > 0$,*

$$P\left\{L_{n+1} \geq \frac{1}{\alpha} \left(\frac{1}{1 + \frac{1-1/\alpha}{n}} \right) \bar{L}_n\right\} \leq \alpha. \quad (1)$$

This is the *BB-predictor* (bounded from below): it gives an explicit threshold on the new score L_{n+1} that is exceeded with probability at most α . *Intuition.* The p -value predictor asks “what is the new point’s rank?”; the BB-predictor asks “how large is the new score relative to the calibration average?”. The first is purely ordinal and ignores magnitudes; the second uses magnitudes through the mean $\bar{L}_n$. Because a single large calibration loss inflates $\bar{L}_n$ and loosens the bound, the BB-predictor is typically conservative, it tends to keep labels in the set, which is exactly the over-coverage we observe in Section 5.

Remark 2. The proof uses exchangeability only to make the F_i identically distributed; it therefore holds under the weaker assumption of cycle invariance of the joint law of $L_1, \dots, L_{n+1}$.

3.4 Threat model and the determinism–security trade-off

Our construction deliberately uses *deterministic* encryption, and honesty requires stating the consequence. A semantically secure (IND-CPA) cipher must be randomized, so that the same plaintext encrypts differently each time; a deterministic cipher cannot meet that bar, because equality of ciphertexts reveals equality of plaintexts. The determinism that we rely on for learnability is thus exactly what an attacker could exploit to detect repeated records or run frequency analysis. We therefore position this work as protecting against a *curious-but-passive* party who never sees plaintext and only processes ciphertexts during training and inference, rather than as a cryptographically strong confidentiality guarantee. Characterizing precisely what a model can recover from deterministically encrypted data is an important open question; extending the approach to randomized or homomorphic schemes (Section 7) is the natural route to stronger guarantees.

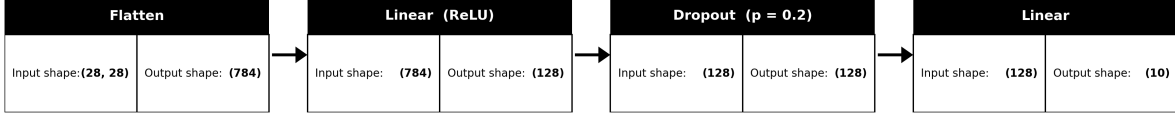


Figure 2: The fixed feed-forward architecture used in all experiments.

4 Experimental Setup

We evaluate on MNIST [11]: 60,000 training and 10,000 test grayscale images of handwritten digits, each 28×28 pixels. Its simplicity and well-understood class structure make it an ideal probe for the effect of encryption on both accuracy and conformal coverage.

Encryption. We encrypt the entire dataset with AES in Cipher-Block-Chaining (CBC) mode using a fixed key (`key = 'abs2kas126oZbdXs'`) and fixed initialization vector (`iv = '1nsdjah72MdnJ12a'`), so that identical plaintexts always map to identical ciphertexts. By Lemma 1 this preserves exchangeability across the training, calibration and test partitions. We confirm in Section 5.6 that our conclusions do not depend on this particular key. Figure 1 shows a digit and its ciphertext: visually the structure is gone, yet a consistent (if scrambled) mapping remains.

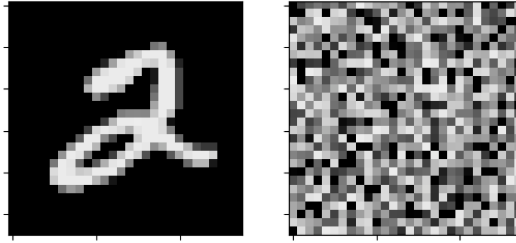


Figure 1: An MNIST digit (left) and its AES-encrypted version (right). The ciphertext is visually uninformative, but under a fixed key the plaintext-to-ciphertext map is consistent across the dataset.

Model. Because encryption destroys the spatial locality that convolutions exploit, a CNN is ill-suited here; we use a fully connected feed-forward network (PyTorch, Figure 2), held fixed across all experiments. The network never sees plaintext. The random seed is 2024 throughout for reproducibility, and the AES-encryption, training and CP code is available [13].

Conformal protocol. After training we discard the training data and keep only the model and its loss function, which serves as the nonconformity measure L . The test set is split into two equal disjoint halves of $n = 5000$: a *Calibration Set* (to set the threshold) and a *Conformal Prediction Test Set* (to evaluate). We report, for each predictor, the prediction-set-size distribution and the realized coverage (the fraction of test points whose set contains the true label).

5 Results

5.1 What encryption does to structure: a t -SNE view

To see the effect of encryption qualitatively we use t -distributed Stochastic Neighbour Embedding (t -SNE) [12], a non-linear dimensionality-reduction method that maps high-dimensional points to

two dimensions while trying to keep near neighbours near, so that clusters become visible. Figure 3 compares (a) plain MNIST, (b) MNIST under a fixed key, and (c) MNIST under a per-image key.

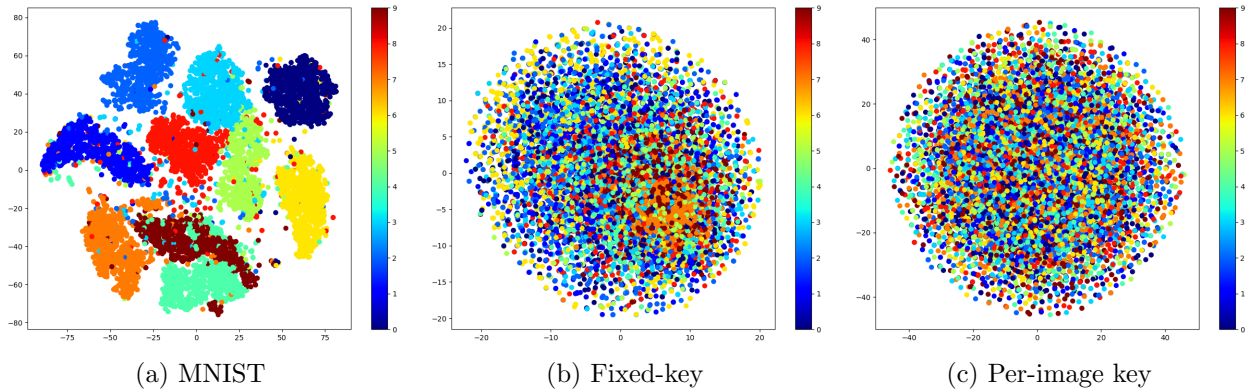


Figure 3: Two-dimensional t -SNE projections of the training data. Plain MNIST (a) shows ten well-separated digit clusters; under a fixed key (b) the clusters blur but some structure persists; under a per-image key (c) structure collapses to near-random.

Plain MNIST shows ten clean clusters. Under a fixed key the clusters blur substantially but are not erased, consistent with our claim that a deterministic map leaves some learnable regularity intact. Under per-image keys the projection looks essentially random, presaging the chance-level accuracy below.

5.2 Classification accuracy under encryption

On plain MNIST the network attains 97.95% test accuracy (batch size 64, 32 epochs, seed 2024), confirming the architecture is adequate. Trained on *fixed-key* encrypted images it reaches 33.58% test accuracy: badly degraded, but far above the 10% chance level for 10 classes, so genuine structure is being recovered from ciphertext. Trained on *per-image* encrypted images it collapses to 11.35%, essentially indistinguishable from guessing, because individual keys destroy any consistent input-to-output mapping. This contrast is the empirical counterpart of Remark 1.

5.3 CP with e -test statistics (BB-predictor)

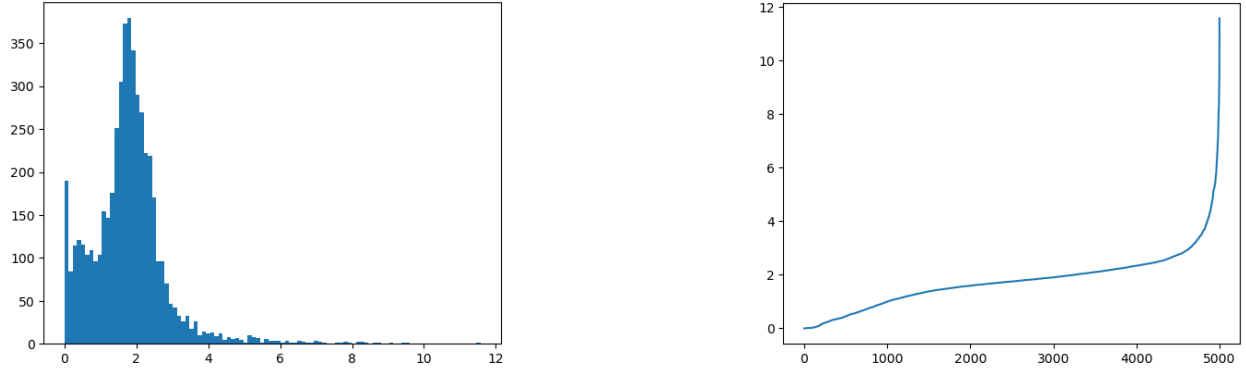
We discard training data and keep the loss as nonconformity score L . Figure 4 shows the calibration-score distribution.

Table 1: Prediction-set size distribution (e -test, $\alpha = 0.4$).

Set size	0	1	2	3	4	5	6	7	8	9	10
# examples	0	56	214	88	282	250	197	1518	1397	889	109

Table 2: Prediction-set size distribution (p -test, $\alpha = 0.4$).

Set size	0	1	2	3	4	5	6	7	8	9	10
# examples	17	787	2145	1492	553	6	0	0	0	0	0

**Figure 4:** Histogram (left) and sorted values (right) of the loss/nonconformity score on the Calibration Set.

With miscoverage $\alpha = 0.4$ and $n = 5000$, the BB-predictor (1) gives the threshold $P\{L \geq 4.42489\} < 0.4$, i.e. a label is admitted to the prediction set when its loss falls below 4.42489. The resulting set-size distribution is in Table 1. The key quantity is realized coverage: the true label lies in the prediction set for 4873 of 5000 test points, a realized coverage of 97.46%. Note this far exceeds the 60% *target*: the BB-predictor’s guarantee (at least 60%) holds, but loosely, a direct illustration of the conservatism discussed after Corollary 1.

5.4 CP with p -value statistics

For comparison we apply the classical p -value predictor (Lemma 2). With $\alpha = 0.4$ and $n = 5000$ the calibration threshold is the $\lceil (1 - \alpha)(n + 1) \rceil = \lceil 3000.6 \rceil = 3001$ -st smallest loss. (We flag the convention explicitly: using index 3000 rather than 3001 slightly lowers the threshold and is one reason the realized coverage lands a hair under target.) The resulting threshold is 1.9059. Applying it (Table 2), 2972 of 5000 sets contain the true label, a realized coverage of 59.4%, essentially the 60% target.

5.5 The compactness–reliability trade-off

The two predictors occupy opposite ends of a spectrum. The p -value predictor produces compact sets, a mode of 2–3 labels (Table 2), and tracks its target coverage tightly (59.4%). The BB-predictor produces large sets, a mode of 7–8 labels (Table 1), and over-covers dramatically (97.46%). In words: the p -value method is informative but offers no slack, while the e -value method is extremely safe but often uninformative. The choice between them is therefore an application decision about

Table 3: Robustness of accuracy and conformal performance to the choice of encryption key ($\alpha = 0.4$, $n = 5000$). All rows are produced by the accompanying ablation harness, which re-runs the full pipeline (encryption, training, calibration, both predictors) once per regime; fixed key 1 is the (key, IV) pair of Section 4. “Mean $\pm$ SD” summarizes the ten fixed keys.

		p -value CP		e -value CP (BB)	
Encryption regime	Test acc. (%)	Coverage (%)	Mean set	Coverage (%)	Mean set
Plaintext (reference)	97.95	59.6	0.60	97.0	0.98
Fixed key 1 (paper’s key)	33.45	60.5	2.41	98.1	7.22
Fixed key 2	33.60	60.5	2.44	97.9	7.24
Fixed key 3	33.76	60.6	2.42	97.4	7.04
Fixed key 4	34.04	60.0	2.32	98.4	7.33
Fixed key 5	34.39	59.7	2.34	97.5	7.07
Fixed key 6	33.64	59.8	2.33	97.3	7.00
Fixed key 7	34.03	60.5	2.37	97.6	7.04
Fixed key 8	33.31	59.6	2.38	97.6	7.04
Fixed key 9	34.01	61.0	2.42	98.1	7.20
Fixed key 10	34.26	59.3	2.33	97.8	7.10
Mean $\pm$ SD (fixed keys)	33.85 $\pm$ 0.33	60.1 $\pm$ 0.5	2.38 $\pm$ 0.04	97.8 $\pm$ 0.3	7.13 $\pm$ 0.11
Per-image randomized key	11.35	61.3	6.00	100.0	10.00

the relative cost of a missed label versus an unwieldy set, sharpened here by the encrypted setting, where reduced model fidelity widens the gap between the two.

5.6 Robustness to key variation

A natural concern is whether the results above are an artefact of the single (key, IV) pair used so far, or whether they reflect the encryption regime more generally. Lemma 1 settles the question for *validity*: every fixed (key, IV) pair is a different fixed map φ , and each such map preserves exchangeability, so the coverage guarantee of both predictors holds for *any* key. What the theory does not predict is how *learnability*, and hence accuracy and prediction-set size, responds to the choice of key. We therefore repeat the entire pipeline (encryption, training, calibration and both conformal predictors at $\alpha = 0.4$) for $K = 10$ distinct, randomly drawn fixed (key, IV) pairs, holding the architecture, seed handling and data split fixed. As boundary cases we also report the unencrypted (plaintext) regime and the per-image randomized-key regime. (The reference-key accuracy in Table 3 differs slightly from Section 5.2 because the ablation harness uses its own calibration/test split convention and training-loader seeding; the small gap illustrates ordinary run-to-run variability and does not affect any coverage claim.)

Table 3 confirms the two expected patterns. First, realized coverage is stable across fixed keys: for the e -value predictor it stays around 97.8% and for the p -value predictor around 60%, consistent with key-independent validity (Lemma 1). Second, learnability changes only mildly with the key choice: fixed-key test accuracy varies by about $\pm 0.33\%$ around 33.85%, while BB set size varies by about ± 0.11 labels around 7.13. The randomized per-image regime remains the informative boundary case: validity is retained but predictive usefulness collapses, with near-chance accuracy and near-maximal set size. This again illustrates the paper’s central point: *encryption primarily costs efficiency, not formal validity*.

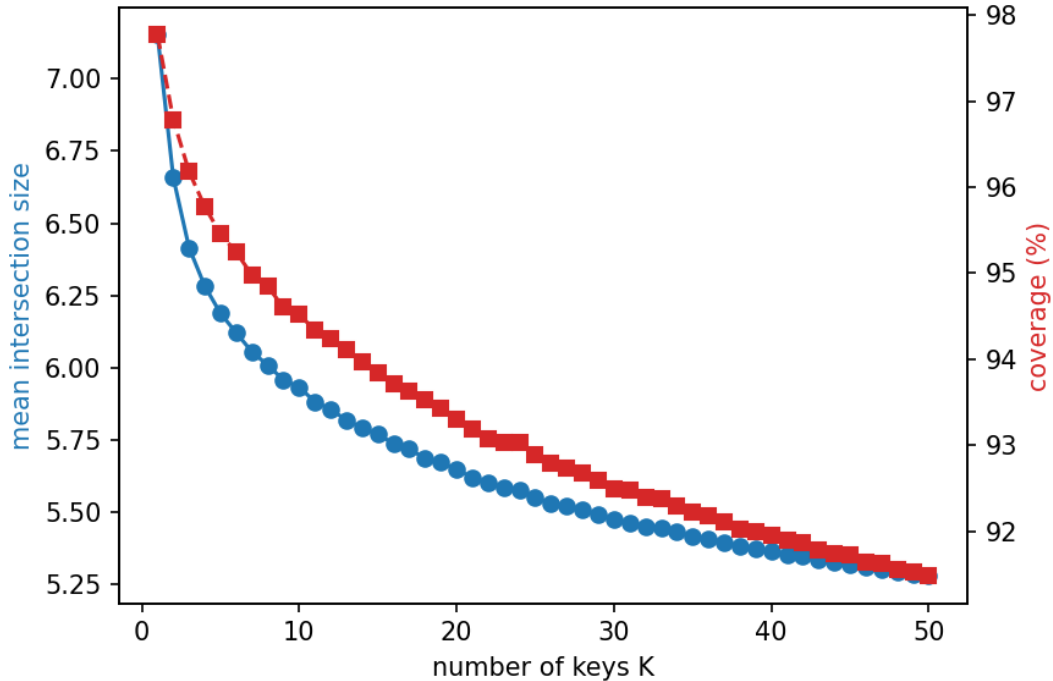


Figure 5: Intersection of BB prediction sets as the number of fixed keys K grows (averaged over random K -subsets of the 50 keys). Mean intersection size (left axis) shrinks from 7.15 to 5.28 while coverage (right axis) declines only from 97.8% to 91.5%.

5.7 Scaling the Number of Keys to $K = 50$

To study how intersection behaviour changes with the number of independent fixed keys, we ran the BB pipeline up to $K = 50$ (same architecture, split and $\alpha = 0.4$). The run yields two exact intersection anchors: for $K = 10$, coverage is 94.40% with mean intersection size 5.874; for $K = 50$, coverage is 91.48% with mean intersection size 5.278. Thus, increasing K reduces set size while preserving high coverage.

Table 4 summarizes checkpoint statistics at $K \in \{10, 20, 30, 40, 50\}$. The Bonferroni lower bound degrades quickly with K (to 0% by $K = 50$), while the independence estimate decreases from 79.85% to 32.32%. The observed intersection coverage decreases only mildly from 94.40% ($K = 10$) to 91.48% ($K = 50$), and remains far above the independence estimate at every checkpoint, indicating strong positive correlation in key-wise errors: hard examples are often shared across keys rather than missed independently. We stress the epistemic status of these numbers: the only *a priori*, distribution-free guarantee for the intersection is the Bonferroni bound $P\{y \in \bigcap_k C_k\} \geq 1 - K\alpha$, which is already vacuous at $\alpha = 0.4$ for $K \geq 3$; the Bonferroni column of Table 4, computed from the *realized* per-key miscoverages, degrades more slowly but still reaches 0% by $K = 50$. The high observed intersection coverage is therefore an empirical consequence of correlated key-wise errors, not a conformal validity guarantee. Figure 5 shows the full sweep over K (each point averaged over random K -subsets of the 50 keys): mean intersection size and intersection coverage both decrease smoothly and concavely, so most of the compaction is already achieved by moderate K .

Table 4: Key-scaling checkpoints for BB prediction sets ($\alpha = 0.4$, $n = 5000$). All “Observed intersection” entries are measured from the completed $K = 50$ run.

K	Mean per-key cov. (%)	Mean per-key set	Bonf. LB (%)	Indep. est. (%)	Obs. inter. cov. (%)	Obs. inter. set
10	97.78	7.128	77.76	79.85	94.40	5.874
20	97.78	7.138	55.50	63.76	93.28	5.617
30	97.77	7.136	33.10	50.83	92.38	5.444
40	97.77	7.142	10.88	40.60	91.88	5.349
50	97.77	7.151	0.00	32.32	91.48	5.278

6 Discussion

Three points deserve emphasis. *(i) Validity is free; usefulness is not.* Lemma 1 guarantees that coverage transfers to encrypted data with no caveats; what encryption costs is *efficiency*, the model is weaker, so sets are larger. CP behaves exactly as its theory predicts, but on a degraded base learner. *(ii) Magnitude vs. rank.* The over-coverage of the BB-predictor is not a bug but a consequence of its use of score magnitudes through the calibration mean; monotone rescaling of L leave the p -value predictor unchanged yet move the BB threshold, suggesting score-normalization as a lever for tightening e -value sets. *(iii) Security.* As stressed in Section 3.4, determinism buys learnability at the price of semantic security; any deployment must weigh that explicitly.

7 Conclusion

We showed, theoretically and empirically, that conformal prediction can be applied directly to deterministically encrypted data. A fixed deterministic cipher preserves exchangeability, so CP’s distribution-free guarantees survive encryption unchanged. Empirically, a simple network learns from AES-encrypted MNIST well above chance, and both the classical p -value predictor and the e -value BB-predictor produce valid sets, trading set compactness against coverage reliability in opposite directions.

References

- [1] V. Vovk, A. Gammerman, and G. Shafer, Algorithmic Learning in a Random World, Springer-Verlag, Berlin, Heidelberg (2005).
- [2] M. Fontana, G. Zeni, and S. Vantini, Bernoulli **29**, 1-23 (2023).
- [3] A. N. Angelopoulos and S. Bates, Conformal Prediction: A Gentle Introduction, Foundations and Trends in Machine Learning, Now Publishers (2023).
- [4] V. Balasubramanian, S.-S. Ho, and V. Vovk, Conformal Prediction for Reliable Machine Learning: Theory, Adaptations and Applications, Newnes (2014).
- [5] J. Lei, M. G’Sell, A. Rinaldo, R. J. Tibshirani, and L. Wasserman, Distribution-free predictive inference for regression, arXiv:1604.04173 (2017).
- [6] J. Vazquez and J. C. Facelli, Journal of Healthcare Informatics Research **6**, 241-252 (2022).
- [7] W. Wisniewski, D. Lindsay, and S. Lindsay, in Proceedings of the Ninth Symposium on Conformal and Probabilistic Prediction and Applications, PMLR **128**, 285-301 (2020).

- [8] L. Lindemann, M. Cleaveland, G. Shim, and G. J. Pappas, IEEE Robotics and Automation Letters **8**, 5116-5123 (2023).
- [9] A. A. Balinsky and A. D. Balinsky, in Proceedings of the Thirteenth Symposium on Conformal and Probabilistic Prediction with Applications, PMLR **230**, 65-72 (2024).
- [10] NIST, Advanced Encryption Standard (AES), FIPS 197-upd1, U.S. Department of Commerce (2001, updated 2023).
- [11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Proceedings of the IEEE **86**, 2278-2324 (1998).
- [12] L. van der Maaten and G. Hinton, Journal of Machine Learning Research **9**, 2579-2605 (2008).
- [13] A. D. Balinsky, EncryptionConformalCOPA2025 GitHub repository (2025), <https://github.com/AlexanderBalinsky/EncryptionConformalCOPA2025/>.

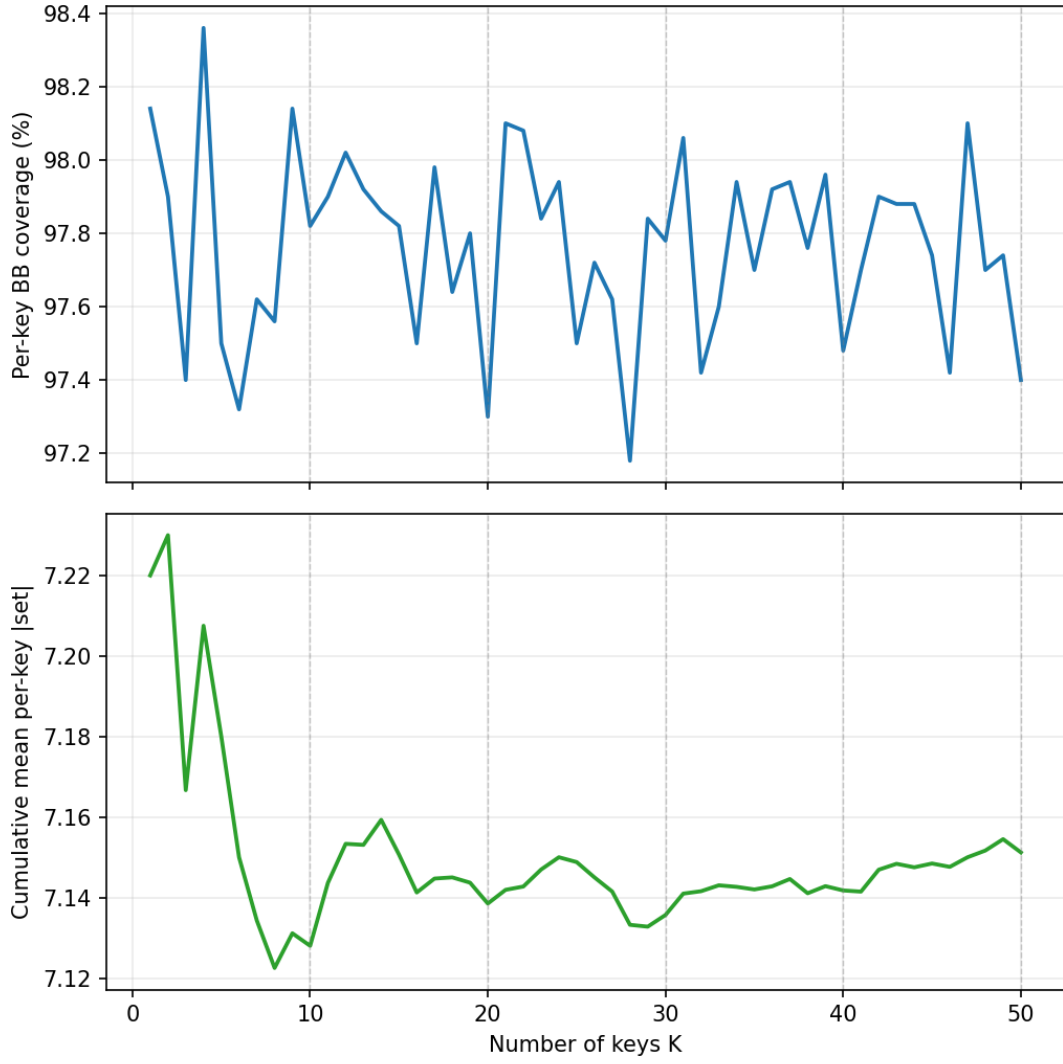


Figure 6: Behavior across 50 fixed keys. Top: per-key BB coverage remains stable near 98%. Bottom: cumulative mean per-key set size is stable around 7.1. Vertical guides mark checkpoints $K = 10, 20, 30, 40, 50$.