# Concept of Vapnik–Chervonenkis (VC) Dimension in Today's Artificial Intelligence Technology Era

Adegoolu Paul M.[1], C. H. Ugwuishiwu Ph.D [2], D. U. Ebem Ph.D [3]
[1]PhD student, University of Nigeria, Nsukka
[2,3]Lecturer, University of Nigeria, Nsukka

*Abstract*—**Artificial intelligence (AI) systems increasingly influence decisions in healthcare, finance, education, public administration, manufacturing, and digital communication. Their usefulness depends not only on fitting training data but also on generalising reliably to previously unseen examples. The Vapnik–Chervonenkis (VC) dimension is a foundational concept in statistical learning theory that characterises the expressive capacity of a hypothesis class through its ability to shatter finite sets of data points. This conceptual review explains the mathematical meaning of VC dimension, its relationship to empirical risk minimisation, uniform convergence, Probably Approximately Correct (PAC) learning, and sample complexity, and its continuing relevance to contemporary AI. It then considers how VC theory informs the analysis of classical classifiers, neural networks, deep learning, and foundation-model systems, while distinguishing theoretical capacity bounds from practical predictive performance. The review argues that VC dimension remains valuable as a language for reasoning about model complexity and data requirements, but is insufficient on its own to explain the generalisation behaviour of modern overparameterised systems. Distribution shift, optimisation, representation learning, data quality, regularisation, and deployment context must also be considered. The article recommends using VC dimension alongside complementary tools such as Rademacher complexity, margin-based analysis, stability, calibration, robustness evaluation, and out-of-distribution testing. The central conclusion is that VC theory is not an obsolete relic of early machine learning; rather, it is a foundational component of a broader theoretical and practical framework for building dependable AI.**

*Index Terms*—**artificial intelligence; VC dimension; statistical learning theory; generalisation; machine learning; deep learning; PAC learning; model complexity; trustworthy AI.**

## I. INTRODUCTION

Artificial intelligence has moved from a specialised research area into a general-purpose technology used to classify images, forecast demand, detect fraud, support clinical decisions, translate languages, generate text, and assist scientific discovery. These applications share a central challenge: a model must perform well not only on the examples used during training but also on new examples encountered in practice. This ability is commonly called generalisation. A model may achieve very low training error and nevertheless fail when the input population, operating environment, or task changes.

The challenge of generalisation motivated the development of statistical learning theory. Within this tradition, the Vapnik–Chervonenkis dimension, usually abbreviated as VC dimension, measures the expressive capacity of a class of binary classifiers. The concept is associated with the work of Vapnik and Chervonenkis (1971), and its relationship to learnability was developed further in work connecting VC theory to the Probably Approximately Correct (PAC) framework (Blumer et al., 1989; Valiant, 1984).

VC dimension is important because model complexity creates a trade-off. A hypothesis class that is too restricted may fail to represent the underlying relationship, producing high approximation error. A highly flexible class may fit training observations closely but can require more evidence to establish that its learned decision rule generalises. VC theory provides formal tools for describing this capacity–data relationship. It does not, however, claim that a single complexity number fully predicts every model's real-world behaviour.

The contemporary AI era makes this distinction especially important. Deep neural networks may contain millions or billions of parameters and can often fit training data almost perfectly, yet they sometimes generalise effectively when trained with suitable architectures, data, optimisation procedures, and regularisation. Large language models and other foundation models introduce further issues, including pretraining on large heterogeneous corpora, adaptation to downstream tasks, data contamination, prompt sensitivity, and distribution shift. The theoretical question is therefore not simply whether a model has high capacity, but how capacity interacts with the learning algorithm, available data, inductive biases, and deployment conditions.

This article provides a conceptual review of VC dimension and its implications for modern AI. It explains the underlying mathematics, discusses applications and limitations, and proposes a practical framework for using VC theory as one component of responsible model development.

## II. AIM AND REVIEW QUESTIONS

The aim of this article is to clarify the continuing relevance of VC dimension to AI technology and to identify the limits of applying classical VC results directly to modern deep learning and generative AI.

The review is guided by four questions:

- What does VC dimension measure, and how is it defined mathematically?
- How does VC dimension relate to generalisation, sample complexity, and PAC learnability?
- In what ways can VC theory inform the design and evaluation of contemporary AI systems?
- What are the limitations of VC dimension, and how should it be combined with modern evaluation methods?

## III. REVIEW APPROACH

This is a conceptual narrative review rather than a systematic review or an empirical experiment. It synthesises foundational publications in statistical learning theory and selected scholarly discussions of neural-network generalisation. The emphasis is on explaining core concepts and drawing cautious implications for current AI practice. No new dataset, experiment, or statistical result is claimed. A journal submission should adapt this section to the target journal's expectations and, if a systematic review is required, add reproducible databases, search strings, inclusion criteria, screening procedures, and a study-selection flow diagram.

## IV. THEORETICAL FOUNDATIONS OF VC DIMENSION

### 4.1 Hypothesis classes and classifiers

Let X be an input space and let H be a hypothesis class containing functions h that map inputs to binary labels, conventionally $\{0, 1\}$ or $\{-1, +1\}$. A learning algorithm receives a sample of labelled observations and selects a hypothesis from H, or from a related parameterised family. Examples of hypothesis classes include linear decision boundaries, decision trees of bounded depth, and neural networks with specified architectures and activation functions.

## V. SHATTERING

A finite set of points $S = \{x_1, \ldots, x_m\}$ is shattered by H if every one of the $2^m$ possible binary label assignments on those points can be realised by some hypothesis in H. Shattering is therefore a stringent test of expressive power: the class must be able to produce every possible dichotomy of the selected points, not merely classify one particular labelling correctly.

## VI. FORMAL DEFINITION

The VC dimension of H, written VCdim(H), is the size of the largest finite set that H can shatter. If sets of arbitrarily large finite size can be shattered, the VC dimension is infinite. Formally:

$$\mathrm{VCdim}(H) = \max \{|S|: S \text{ is shattered by } H\},$$

with $\mathrm{VCdim}(H) = \infty$ when no finite maximum exists. The definition concerns the hypothesis class, not the performance of a single trained model. Two models trained on the same dataset can behave differently because of their optimisation procedures, initialisation, regularisation, or learned parameters even when their underlying hypothesis class is the same.

## VII. ILLUSTRATIVE EXAMPLES

- Threshold classifiers on the real line: a class of one-dimensional threshold functions has VC dimension 1 under the standard threshold definition, because it can shatter one point but cannot realise every labelling of two ordered points.
- Affine linear classifiers in d-dimensional Euclidean space: the standard class of linear classifiers with a bias term has VC dimension d + 1, under the usual non-degenerate setting.
- Axis-aligned rectangles in the plane: this class has VC dimension 4.
- Neural networks: their VC dimension depends on architecture, parameterisation, activation functions, and assumptions. Capacity can grow substantially with network size, but a parameter count alone should not be treated as a universal, exact estimate for every architecture.

These examples illustrate that VC dimension measures combinatorial flexibility rather than semantic understanding, computational cost, fairness, robustness, or the quality of a model's predictions.

## VIII. VC DIMENSION, GENERALISATION, AND SAMPLE COMPLEXITY

Empirical and true risk:

Let D be an unknown data-generating distribution over inputs and labels. For a hypothesis h, its true risk R(h) is the probability of an incorrect prediction on a fresh draw from D. Given a sample S of n observations, its empirical risk $\hat{R}S(h)$ is the fraction of training observations it misclassifies. Generalisation analysis asks when low empirical risk is evidence of low true risk.

For a sufficiently rich hypothesis class, some hypotheses may fit the training sample by exploiting accidental patterns. VC theory bounds the number of distinct labelings the class can impose on a finite sample through its growth function. The Sauer–Shelah lemma shows that, when VC dimension is d and n is sufficiently large relative to d, the number of distinct dichotomies grows polynomially rather than exponentially in n. This combinatorial control supports uniform-convergence results.

Generalisation bounds:

A representative VC-style result states, informally, that with probability at least $1 - \delta$ over an independently sampled training set, the difference between empirical and true risk can be bounded by a complexity term that decreases as sample size increases and increases with VC dimension and confidence requirements. Exact constants and logarithmic terms vary by theorem, assumptions, and whether the bound is distribution-free, one-sided, or two-sided.

A schematic expression is:

$R(h) \leq \hat{R}S(h) + O\left(\sqrt{((d \log(n/d) + \log(1/\delta))/n)}\right)$,

where d is the VC dimension, n is sample size, and $\delta$ is a failure-probability parameter. This expression is illustrative rather than a substitute for a theorem with specified assumptions and constants.

The practical interpretation is that a larger hypothesis class generally requires more data to obtain the same uniform statistical guarantee. The guarantee may be conservative, particularly for modern neural networks, and should not be mistaken for a precise forecast of test accuracy.

PAC learning:

The Probably Approximately Correct framework formalises learning in terms of accuracy and confidence. A learner is expected, with probability at least $1 - \delta$, to return a hypothesis whose error is at most $\varepsilon$, subject to the framework's assumptions. For binary classification under standard distribution-free settings, finite VC dimension characterises PAC learnability for broad classes of hypotheses. Sample-complexity bounds depend on VC dimension, the target accuracy $\varepsilon$, and confidence parameter $\delta$ (Blumer et al., 1989; Valiant, 1984).

The result is conceptually important: it connects the capacity of a model class to the amount of labelled data needed for reliable learning. It does not guarantee that an available optimisation algorithm will efficiently find a good hypothesis, that the training distribution matches deployment data, or that labels are accurate.

Structural risk minimization:

Structural risk minimisation (SRM) addresses the tension between fitting observed data and controlling model complexity by organising candidate hypotheses into nested or otherwise ordered classes. The learner

balances empirical performance against a complexity penalty. In practical systems, regularisation, early stopping, architecture selection, and validation-based model selection play related roles, although they are not all identical to classical SRM.

## IX. RELEVANCE TO ARTIFICIAL INTELLIGENCE TODAY

### A. Classical machine learning

In linear classification, support vector machines, and other conventional learning systems, VC dimension provides a principled way to discuss capacity and the amount of data needed to support generalisation. It can help explain why a more flexible decision-boundary class may fit complex patterns but also demands stronger controls against overfitting. For linear classifiers in d dimensions, the VC dimension result offers a transparent example of how feature-space dimensionality affects capacity.

### B. Deep neural networks

Deep neural networks can represent complicated nonlinear relationships through layered transformations. The expressive capacity of these networks depends on width, depth, activation functions, and parameter constraints. Classical VC bounds therefore provide useful worst-case insight, but often yield loose estimates when applied to large networks. In practice, generalisation also depends on optimisation dynamics, implicit biases of training algorithms, weight norms, margins, data augmentation, and the geometry of the learned representation.

A central modern observation is that overparameterized networks can fit training data while still achieving strong test performance. This phenomenon does not invalidate VC theory; rather, it shows that raw worst-case capacity is not a complete explanation of the behaviour of a particular training procedure. Analyses based on Rademacher complexity, algorithmic stability, PAC-Bayes bounds, compression, margins, and norm-based measures can sometimes provide complementary or more informative perspectives.

### C. Generative AI and foundation models

Large language models and multimodal foundation models are trained on broad corpora and then used through prompting, fine-tuning, retrieval augmentation, or other adaptation methods. A model's possible input-output behaviours form a complex function class, but its effective behaviour is shaped by pretraining data, architecture, tokenisation, optimisation, alignment procedures, and deployment settings. A single VC-dimension value is rarely a practical stand-alone measure of the reliability of a deployed foundation model.

VC-inspired reasoning remains relevant when comparing restricted adaptation strategies, studying classifier heads placed on pretrained representations, or asking how many labelled examples may be needed to learn a downstream decision rule. It is less direct as an explanation of open-ended language generation, where output spaces are enormous and evaluation involves qualities beyond binary classification. For generative AI, VC analysis should be combined with task-specific evaluations of factuality, calibration where meaningful, robustness, memorisation, data leakage, and harmful or biased outputs.

### D. High-stakes and resource-constrained applications

In healthcare, credit decisions, educational assessment, agriculture, public services, and industrial monitoring, the cost of generalisation failure can be high. VC theory encourages developers to ask whether the hypothesis class is too flexible for the available labelled data. However, a low-complexity model is not automatically fair, safe, or well calibrated. Such systems also require subgroup analysis, error-cost analysis, data governance, privacy protection, human oversight, and monitoring after deployment.

For organizations with limited data or computing resources, a simpler model with transparent assumptions may be preferable to a highly flexible model that cannot be adequately validated. This is a decision about the complete system including data quality, interpretability, operational constraints, and risk not about VC dimension alone.

## X. BENEFITS AND LIMITATIONS

### A. Benefits

- Provides a mathematically defined measure of the expressive capacity of a hypothesis class.
- Links model capacity to sample complexity and distribution-free generalisation guarantees.

- Supports reasoning about overfitting and the balance between model flexibility and available data.
- Offers a common theoretical language for comparing classes of classifiers and understanding learnability.
- Provides a foundation for later complexity measures and statistical learning theory.

B. Limitations

- Worst-case bounds may be loose and may not reflect the behaviour of a particular trained model.
- The VC dimension of a hypothesis class does not, by itself, describe the effect of a particular optimiser, initialisation, regularisation scheme, or training trajectory.
- Classical results usually rely on assumptions such as independent and identically distributed observations; real deployment can involve temporal dependence, domain shift, and changing user behaviour.
- VC dimension does not directly measure calibration, robustness to adversarial inputs, fairness, privacy, interpretability, or factual reliability.
- For generative and structured-output systems, binary-classification VC intuition may not map cleanly onto the full deployment task.

Consequently, VC dimension should be understood as one theoretical lens, not as a single score that certifies an AI system as reliable.

C. Practical Framework for AI Developers

The following workflow translates the theoretical discussion into a practical model-development process.

- Define the task and hypothesis class. Specify the prediction target, input representation, loss function, and model family before interpreting capacity.
- Estimate capacity where feasible. Use known VC-dimension results for standard classes, or clearly label estimates and assumptions for complex architectures.
- Relate capacity to data availability. Consider the number, diversity, representativeness, and label quality of observations, rather than relying on sample count alone.
- Control model selection. Use regularisation, validation, early stopping, suitable architecture choices, and data augmentation where justified.
- Evaluate on untouched data. Use a test set separated from model development and guard against leakage, duplicate records, and contamination.
- Test distribution shift. Evaluate performance across time, geography, devices, demographic groups where appropriate, and other deployment-relevant conditions.
- Use complementary complexity and reliability measures. Consider margins, Rademacher complexity, stability, calibration, uncertainty estimates, robustness tests, and subgroup metrics as appropriate.
- Monitor the deployed system. Track drift, errors, user feedback, and changes in data-generating conditions, with a defined procedure for reassessment and retraining.

This workflow treats VC theory as a guide to questions and assumptions rather than as a replacement for empirical evaluation.

## XI. RESEARCH GAPS AND FUTURE DIRECTIONS

Several directions merit further research. First, researchers should investigate when VC-style bounds can be sharpened for modern architectures and training procedures. Second, theoretical capacity measures need to be related more systematically to observed performance under distribution shift. Third, the analysis of foundation models should distinguish the capacity of the pretrained model from the effective complexity of a downstream task after prompting, fine-tuning, or retrieval augmentation. Fourth, researchers should examine how sample complexity changes when labels are noisy, data are imbalanced, or examples are generated synthetically. Fifth, theoretical learning guarantees should be integrated with system-level assessments of safety, fairness, privacy, calibration, and operational robustness.

For empirical follow-up, a useful study could compare a small number of model classes on clearly specified datasets, reporting training and test errors, sample-size curves, calibration, and out-of-distribution performance. Any such study should distinguish

empirical findings from theoretical claims and document datasets, preprocessing, hyperparameters, uncertainty intervals, and reproducible code.

## XII. DISCUSSION

The enduring value of VC dimension lies in its conceptual precision. It explains why a learning problem cannot be assessed solely by training accuracy and why the flexibility of a hypothesis class matters in relation to available evidence. It also makes clear that reliable learning is a statistical claim about unseen observations, not merely a computational achievement.

At the same time, modern AI demonstrates that capacity cannot be interpreted in isolation. An expressive model may generalise well because its training process favours particular solutions, because the data contain strong regularities, or because learned representations make a downstream task easier. Conversely, a model with apparently manageable capacity may fail under distribution shift or biased sampling. The appropriate conclusion is not that capacity is irrelevant, but that it must be considered alongside data, algorithm, objective, and deployment environment.

For generative AI, the distinction between model-class capacity and deployed-system reliability is particularly important. A foundation model can generate plausible outputs without guaranteeing factual correctness. VC theory does not directly certify such outputs. Nevertheless, the broader statistical-learning perspective encourages explicit assumptions, careful separation of training and evaluation data, and a disciplined account of what evidence supports claims of generalisation.

## XIII. CONCLUSION

The Vapnik–Chervonenkis dimension remains a foundational concept for understanding machine learning in today's artificial intelligence era. By measuring the largest finite set that a hypothesis class can shatter, it formalises one aspect of model expressive capacity and provides a basis for results about uniform convergence, PAC learnability, and sample complexity. Its relevance extends from classical classifiers to selected analyses of neural networks and downstream learning tasks involving foundation models.

However, VC dimension is not a universal predictor of real-world AI performance. Modern deep learning and generative systems require a broader analysis that includes optimisation, regularisation, data quality, distribution shift, robustness, calibration, fairness, privacy, and deployment monitoring. Developers and researchers should therefore use VC theory alongside empirical testing and complementary theoretical tools. In this combined role, VC dimension continues to contribute to the development of AI systems that are not only powerful, but also better understood and more dependable.

## ACKNOWLEDGMENT

I appreciate the author of Life for given me the ability to start my journey into publication. Thanks to my Lecturers for the guide into this journey.

## REFERENCES

[1]. A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, "Learnability and the Vapnik–Chervonenkis Dimension," Journal of the ACM, vol. 36, no. 4, pp. 929–965, 1989, doi: 10.1145/76359.76371.

[2]. L. Devroye, L. Györfi, and G. Lugosi, A Probabilistic Theory of Pattern Recognition. Springer, 1996.

[3]. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016. Available: https://www.deeplearningbook.org/

[4]. M. J. Kearns and U. V. Vazirani, An Introduction to Computational Learning Theory. MIT Press, 1994.

[5]. S. Shalev-Shwartz and S. Ben-David, Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, 2014, doi: 10.1017/CBO9781107298019.

[6]. L. G. Valiant, "A Theory of the Learnable," Communications of the ACM, vol. 27, no. 11, pp. 1134–1142, 1984, doi: 10.1145/1968.1972.

[7]. V. N. Vapnik, Statistical Learning Theory. Wiley, 1998.

[8]. V. N. Vapnik and A. Ya. Chervonenkis, "On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities," Theory of

Probability and Its Applications, vol. 16, no. 2, pp. 264–280, 1971, doi: 10.1137/1116025.

[9]. V. N. Vapnik, The Nature of Statistical Learning Theory. Springer, 1995.

[10]. M. Anthony and P. L. Bartlett, Neural Network Learning: Theoretical Foundations. Cambridge University Press, 1999, doi: 10.1017/CBO9780511624216.

[11]. S. Arora, R. Ge, B. Neyshabur, and Y. Zhang, "Stronger Generalization Bounds for Deep Nets via a Compression Approach," in Proceedings of the 35th International Conference on Machine Learning, 2018, pp. 266–274. Available: https://proceedings.mlr.press/v80/arora18b.html

[12]. M. Belkin, D. Hsu, S. Ma, and S. Mandal, "Reconciling Modern Machine-Learning Practice and the Classical Bias–Variance Trade-Off," Proceedings of the National Academy of Sciences, vol. 116, no. 32, pp. 15849–15854, 2019, doi: 10.1073/pnas.1903070116.

[13]. C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding Deep Learning Requires Rethinking Generalization," in Proceedings of the International Conference on Learning Representations (ICLR), 2017. Available: https://arxiv.org/abs/1611.03530