International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(2) 2026 pp. 1885-1891
https://doi.org/10.70917/ijcisim-2026-5107
Article

Naïve Bayes Classification Based Dynamic Gaussian Regression Modal To Forecast Weather with Accurate Domination Number

T.M. Selvarajan[1], R. Subramoniam[2]

[1]Associate Professor, Department of Mathematics, Noorul Islam Centre for Higher Education, Kumaracoil, Thuckalay-629180, India, selvarajan1967@gmail.com
[2]Assistant Professor, Department of Mathematics, Lekshmipuram College of Arts and Science, Lekshmipuram, Neyyoor – 629802.

Abstract: Weather forecasting is the application of science and technology to forecast the local weather conditions. It's one of the hardest problems in the world. Gaussian process regression is a widely used technique for improving dynamical system models and addressing their errors because of its cutting-edge estimate performance combined with strict and non-conservative uncertainty constraints. The smallest forcing number of at least dominating set of F is its forced dominance number (F(x,y)). We examine the applicationof priors over functions for Gaussian processes, that enable precise execution for given hyperparameter values, the predictive Bayesian analysisthrough matrix operations. A latent space with modest dimensionswith related dynamics along with a mapconnecting the dormant and observingspacesmake up a GPDM.We use Gaussian Process (GP) priors for the dynamics and the observation mappings to marginalize out the model parameters in closed-form.. As an result, A Adaptablesystem nonparametric model which takes model ambiguityinto account is produced.The target data is categorized using the Naive Bayes learning technique.IT took a hypothetical dataset with Temperature Stamp that details the ideal weather for forecasting. Each tuple determines whether the weather conditions are fit ("Yes") or unfit ("No"). Based on the temperature fit or unfit and the humid condition classification is made whether the day is sunny or rainy. It uses probability to forecast a data point's category.
Keywords: NA

1. Introduction

GP regression has been applied to dynamic systems with temporal dependencies and control objectives due to its versatility in modeling nonlinear, unpredictable processes. In order to enable interpretable and probabilistic frameworks for industrial and real-time applications, research investigates incorporating prior knowledge, enforcing physical constraints such as monotonicity, modeling temporal dynamics, and propagating uncertainty for system identification, prediction, and control design. The forced independent dominance number of graphs under specificoperations in binary, including lexicographic product of two graphs, join, and corona. Every diagramunder consideration is undirected, finite, non-trivial, and has loops and numerous edges. for terminology related to graph theory. Predictive control, which optimizes the control signal while accounting for variance information, makes use of the additional data offered by the model of a Gaussian process. Thecontrol of the pH mechanism benchmark are used to illustrate the predictive control theory.Machine learning frequently equates to data-driven parameter estimation. These parameters, like the weights of a neural network, are sometimes many and difficult to understand. However, in contrast, Gaussian processes provide a means of directly infer the high-level characteristics of functions that might suit our data. By simply defining a Gaussian distribution over the function values that could fit our data, Gaussian processes make it simple to include these characteristics into our model.

2. Literature Review

Regression using Gaussian processes is an effective method for modeling, actively investigating, and utilizing unknown functions.An effective non-parametric Bayesian approach to regression problems, That's possible to apply Gaussian process regression in order toboth exploration [1] exploitation situations.An AI-driven Real-Time Analytics Engine (RAE), an intrusion detection system (AIDS), and a Smart Load Balancer (SLB)are the three primary components of the proposed architecture. To improve the system's flexibility, security, and efficiency, these components employ cutting-edge machine learning methods such as autoReinforcement Learning (RL), and Support Vector Machines (SVMs). ncoders [2], Support Vector Machines (SVMs), and Reinforcement Learning (RL).Many scalable GPs have been proposed to increase scalability without sacrificing desired prediction quality. They haven't yet been thoroughly examined and examined, though, so neither academia nor business can fully comprehend them. Given the rise in data quantity, the GP community's evaluation of scalable[3] GPs is both pertinent and crucial.

Without making any assumptions about a cosmological model, [4] Gaussian processes offer a way to extract cosmological information from data. Using a geometric model and kinematic variables cosmologya probe, we do cosmography (tracking the momentdevelopment of the expansion of the universe) in a model-independent way.The deceleration and Hubble parameters in relation to redshift. Subject to some statistical warnings, the removal of theserelationships is accomplishedevaluated in contrast to models that have features ondifferentscales of coherence.Gaussian processes are the main method used to create emulators. In order to verify and evaluate the suitability of an emulator of a Gaussian process [5]. a stand-inregarding the simulator, this work offers various diagnostics. For certain test data, referred to as validation data, which is determined by a subset of simulator runs thatutilizedto construct an emulator, these Comparisons between simulator outputs serve as the foundation for diagnostics.and Gaussian process emulator outputs. A GPDM consists of a latent space in low dimensions with related dynamics and a mathat links the hidden area to a [6] space for observation. Using Gaussian Process (GP) priors for the dynamics and the observation mappings, we marginalize out the model parameters in closed form.The non-parametric result of this dynamical system model in which takes model ambiguity into account.Gaussian process regression networks (GPRN) combine the non-parametric [7] Gaussian processes' adaptabilitywith the frameworkcharacteristicsBuilding neural networks using Bayesian. Input-dependent length-scales and amplitudes, correlations between many response variables that are depending on input and noise, and heavy-tailed predictive distributions are all supported by this model. For this model, derive Effective Markov chain with variational BayesMonte Carlo inference methods.Two approaches have been explored on several difficult situations and have yielded good results: optimization [8] anderaging across hyperparameters using Hybrid Monte Carlo. A deep GP s a deeply held belief system. constructed using Gaussian mapping of processes. Output of a multivariate GP is used to model the data. Another GP then controls the Gaussian process's inputs. The GP latent variable model (GP-LVM) or a typical GPare equivalent to a single layer model.Stochastic [9] gradient descent is commonly used to optimize deep belief networks on comparatively large data sets.The input locations are taken to be noise-free in

normal Gaussian Process analysis. We employ For every input point, a local linear expansionto make calculations tractable. This makes it possible to rewritethe input noise as output noise in proportion to theGP posterior mean's squared gradient. The data is used to infer the input noise variances as additional hyperparameters. They are trained using the normtechnique of optimizing the marginal probability in conjunction using additional hyperparameters. An iterative method for training alternates between calculating the posterior gradient and optimizing the[10] hyperparameters. For test points with a Gaussian distribution, analytic predictive moments can then be determined. We demonstrate that our model outperforms existing approaches by comparing it to others across a variety of regression issues.Recent advances in the use of statistical and Bayesian probability concepts to expert systems. We demonstrate how qualitative and quantitative knowledge can be represented within a directed graphical model, commonly referred to as a belief network in this context, using a real, relatively difficult medical scenario. The original subjective quantitative inputs can be updated using Bayesian [11] statistical approaches, and we offer a set of diagnostics to find discrepancies between the data and the previous definition.Good was the initialBayesian to include uncertainty in the prior distribution clearly into his statistical theory. In fact, there has never been a better explanation of the robust Bayesian [12] position in terms of both philosophy and practice. The formula represents the forcing correct domination number, which is assumed to be the minimum of all −sets in. The general characteristics of this idea are examined. A few frequent [13] graphs are established using the coercionprecise number of dominance. Every pair of integers is shown to have a connected graph G.A group G⊆ D is referred as a linkeddominant collection of a linkedsimple graph D = (E, V) if its < G> is connected. Connected dominance number is whatlowestA dominating set's cardinalitythat is connected. In the symbol since it is γc(G). If D is the only[14] minimum linked dominating setof V contains G, then a group D⊆ T is referred to be a group that is required for T.The number that is forced ,are G(F), is a maximum on all symmetric matrices with a sparsity pattern have a maximum nullityrepresented by the simple graph F. G(F) < m − 1, where mis the order of the graph, and μ ≤ F(G), where μ is the smallest degree, are straightforward lower also[15] upper bounds. This thesis improves the minimal degree lower bound when G has a girth of at least 5.If the subgraph brought created by a dominant group of a diagram's verticesmust form zero forcing group, It is described as a¥emph{dom−forcing group}. The dom−forcingquantity of the graph, represented as, is the lowest cardinality of such a [16] collection. The dom−forcing graph's number is investigated within this article. It also explores the exact calculation of for some well−known graphs.If each vertex ina graph Dis controlled byprecisely a single vertex in adominating group D, then D is perfect. We investigate the presence and creation of PDSs in graph families that result from parallel computer connectivity networks. De Bruijn graphs, cube−connected cycles, cube−connected routes, trees, hypercubes, meshes, tori, dags, and series−parallel graphs are some of these.We demonstrate the [17]presence of a PDS in an unlimited number of situationsfor de

Bruijn graphs, cube-connected cycles, and higher dimensional meshes and tori,although our characterization, It's not comprehensive. The different types of domination numbers of a graph are studied in [18, 19,20].

3. Proposed System

A novel machine learning technique based on Bayesian theory is called Gaussian process regression. It works well for handling complicated regression issues such nonlinearity, short sample sizes, and high dimensions. Fig.1 shows the proposedframe work Accurate Domination Number of a Graph with Dynamic Gaussian Regression Modal

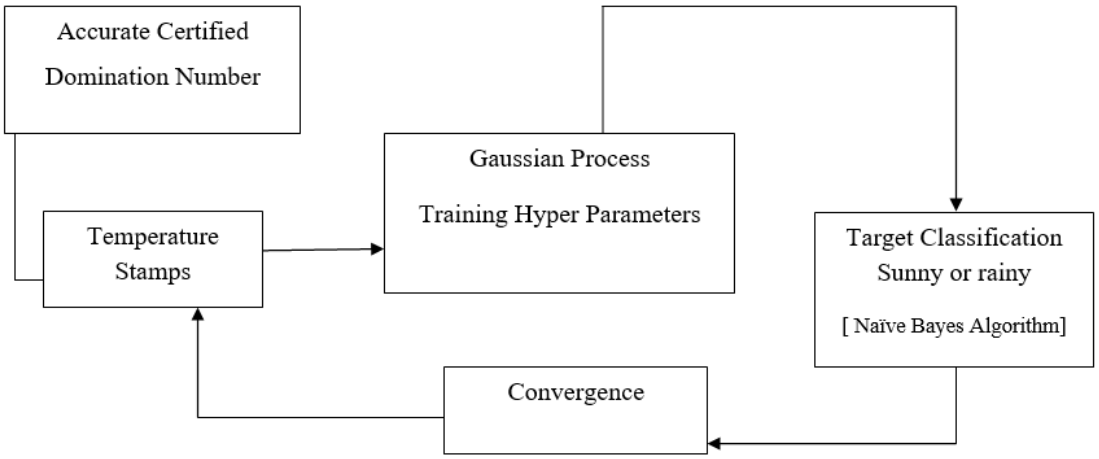

Fig 1 Gaussion process with training Hyper Parameters

3.1 Accurate Certified Domination Number

If D has at least two neighbors or none at all in$X - D$, then D is an accurate verified dominant group of F = (V, E). The minimal cardinality of a precisecertified the exact dominating setcertified number of dominance$\gamma acer$ (F) of F.

For instance 2.2 In the plot below, G(V1) = {1,2,3,4,5}, while {1,4,5,} met the precise certified criterion. Therefore, $\gamma acerG1 = 4$. Fig 2 shows the Forced accurate Domination Number

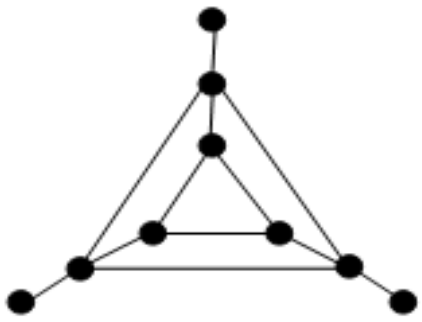

Fig. 2 The Forced Accurate Dominance Number Graph

3.2 Gaussian Process

The m + The outputs of the n training cases and the test case have a one-dimensional joint Gaussian distribution. gives the test case n's predicted distribution through conditioning on the training group's observed goals. Figure 1

shows this process in the scenario with single test and training poin. The mean and variance of the predictive distribution are usually Gaussian.

$$x(n) = k^t(n)k^{-1}t \quad (1)$$

$$\sigma^2_{y-1}=\text{C(n,n) -}l^t\text{ (n) }l^{-1}\text{ l(n) L(n)} \quad (2)$$

where l(n) = (C(n, n(1)), ... , C(n, n(n))f , L is the covariance matrix for the training examples t = (t(l),... , t(n)) and Mij = C(n(i), n(j)).T. y= (y(l), ... , y(n))T .

Equations (1) and (2)'s matrix inversion step suggests that the algorithm's time complexity is O(n3).

3.3 Training Gaussion Process

The GPR methodology is a way to extrapolate (or interpolate) the data from a training set $\mathcal{D}$ = {m , n}. The vector of inputis x={mi|i=1,2,...,}, and the output vector is ni=f(mi), where f is an unknown function. The approach assumes that the data were taken from an underlying GP g(y) ∼$\mathcal{GP}$(μμ(y),k μ(y,y′)) with a certain covariance function L(y,y′) and mean μ(y), which is usually assumed to be zero.

This convargence operation can be freely specified; popular options, likeThe Matérn function , provide a number of free hyperparameters θ={θi}μ{j=1,2,……,n}{|i=1,2,...,m}, which influence the GP's properties, i.e., k(y,y′)=k(y,y′;θ:μ)

3.4 Bayesian analysis

A prior distribution over weights generates a prior distribution over functions in the Bayesian approach to neural networks. The posterior more functions that can be used for predictions is produced by combining this prior using a model of noisethat indicates likelihood of the witnessing he provided functionvalues of thetargetsy. Because the prior over functions for neural networks have a complicated structure, implementations must either approximate

We accomplish this by using The distribution that comes after throughout the scales using Bayesian framework. The distribution after the fact is q(v|yxlm, Xtn) ∝ p(yt|Xt, wm) if a Gaussian prior is applied to the weightsq(wl) = N (0, Σμ) and the Gaussian likelihood p(xt|Ylt, wl) = N (Y t w, μσ2 * I).

3.5 Naive Bayes Algorithm for Target Classification

The Naive Bayes Classifier, a simple probabilistic classifier with few parameters, is used to build machine learning models that outperform other classification algorithms in terms of prediction speed. Because it assumes that one of the model's featuresis independent out of the existence of It is a probabilistic classifier, which is an additional feature.In other words, each characteristic contributes to the predictions separately from the others. The classifier Naive Bayes's basic concept is the utilize The Bayes Theorem to categorize data in line with the likelihood of several classes based on the characteristics of the group.

3.6 Naive Bayes Terminology

This basic premise with Naive Bayes is feature independence which means it classify by assuming that one point in the group does not affect the other point in the group.

The Bayes Theorem offers a methodical approach to conditional probability reversal.

$$Probability\ P(Y) = \frac{p(x)\,.\,p(x)}{p(Y)}$$

Where, $P(Y)$and$p(x)$ -Posterior probability and Probability probability

$p(Y)$- Marginal likelihood

In the posterior, for every class x and select the most likely class

$$x = arg\ arg\ maxy\ p(Y)\ *\pi P(x|Y)$$

4. Results and Discussion

GP realizations with convergence functionsL1 and L2 along with temperature stamp T are shown in Fig 3.

A realization of the T with n points was developed and analyzed using L1 and L2 convergence functions to perform calculations for test model comparison.

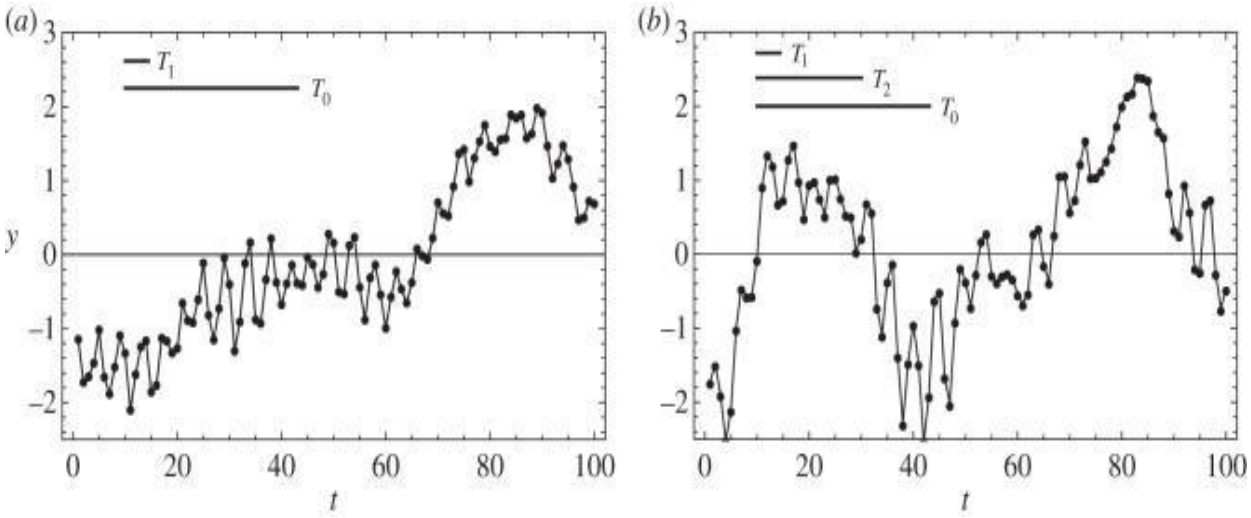

Fig. 3 GPs with convergence functions

The accurate domination number was deduced using a GP regression Fig 4. The target hyper-parameters is specified between H=6000to H=9000. These optimum values were determined asμ= 0.67 σ=0.45. The values of these parameters are important because it relate switching time scales and mean set-point, respectively. GP converged on hyper-parameters that are closely associated with the system's physical properties

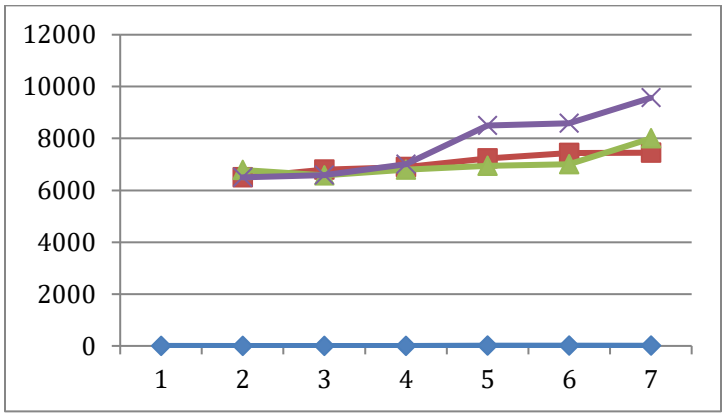

Fig.4 Target hyperparameter

The following table1 shows for Temperature stamp with Accurate Certified Domination Number

T0	Yes	No	P(Yes)	P(No)

False	4	3	2/8	3/5
True	5	2	4/5	3/4
T1	Yes	No	P(Yes)	P(No)
False	5	3	4/7	3/6
True	7	9	5/8	6/7
T2	Yes	No	P(Yes	P(No)
False	9	7	8/9	7/9
True	6	2	¼	6/9

The following tables Table 2 shows the classification result (Target Classifiaction) . Based on the Temperature Fit (Yes) or unfit (No) classification is made whether the day is sunny or rainy hot or normal.

Temperature				
	No	YES	P(NO)	P(Yes)
Mid	3	3	3/6	4/6
Cool	2	5	5/8	3/5
Hot	4	2	6/9	2/7
Humidity				
	No	Yes	P(No)	P(Yes)
Normal	4	2	6/8	7/9
Hot	5	7	3/7	2/5
Weather Forecast of a day				
	No	Yes	P(No)	P(Yes)
Sunny	5	4	4/7	2/6
Rainy	7	9	6/9	1/7

5. Conclusion

Gaussian regression process is a versatile technique formodeling, investigating, and using unknown functions. The proposed system concentrated on Gaussian process regression using a convergence and trained hyperparameter, Bayesian regression techniques is Gaussian regression process-equivalent parametrizationunder certain convergencepresumptions. The GP with Convergence variation is determined.Comparison is made with trained hyperparameter and Forced Accurate Domination number with time stamp.The proposed GPR provides probabilistic framework for regressionwhich means it offers estimations of prediction uncertainty. It is very adaptable and captures intricate correlations in the data. The target is classified by using Naïve Bayes algorithm which is computationally efficient and simple to implement when there are lot of points. Based on the given temperature condition this classification algorithm classifies the weather condition. Even with large amount of training group with categorical points it performs the weather prediction effectively.

REFERENCES

1. Schulz, Eric, Maarten Speekenbrink, and Andreas Krause. "A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions." *Journal of mathematical psychology* 85 (2018): 1-16.

2. Deringer, Volker L., Albert P. Bartók, Noam Bernstein, David M. Wilkins, Michele Ceriotti, and Gábor Csányi. "Gaussian process regression for materials and molecules." *Chemical reviews* 121, no. 16 (2021): 10073-10141.
3. Liu, Haitao, Yew-Soon Ong, Xiaobo Shen, and Jianfei Cai. "When Gaussian process meets big data: A review of scalable GPs." *IEEE transactions on neural networks and learning systems* 31, no. 11 (2020): 4405-4423.
4. Shafieloo, Arman, Alex G. Kim, and Eric V. Linder. "Gaussian process cosmography." *Physical Review D—Particles, Fields, Gravitation, and Cosmology* 85, no. 12 (2012): 123530.
5. Bastos, Leonardo S., and Anthony O'hagan. "Diagnostics for Gaussian process emulators." *Technometrics* 51, no. 4 (2009): 425-438.
6. Tran, Dustin, Rajesh Ranganath, and David M. Blei. "The variational Gaussian process." *arXiv preprint arXiv:1511.06499* (2015).
7. Wilson, Andrew Gordon, David A. Knowles, and Zoubin Ghahramani. "Gaussian process regression networks." *arXiv preprint arXiv:1110.4411* (2011).
8. Williams, Christopher, and Carl Rasmussen. "Gaussian processes for regression." *Advances in neural information processing systems* 8 (1995).
9. Damianou, Andreas, and Neil D. Lawrence. "Deep gaussian processes." In *Artificial intelligence and statistics*, pp. 207-215. PMLR, 2013.
10. McHutchon, Andrew, and Carl Rasmussen. "Gaussian process training with input noise." *Advances in neural information processing systems* 24 (2011).
11. Spiegelhalter, David J., A. Philip Dawid, Steffen L. Lauritzen, and Robert G. Cowell. "Bayesian analysis in expert systems." *Statistical science* (1993): 219-247.
12. [12] Berger, James O. "Robust Bayesian analysis: sensitivity to the prior." *Journal of statistical planning and inference*25, no. 3 (1990): 303-328.
13. Subramoniam, R. "THE FORCING ACCURATE DOMINATION NUMBER OF A GRAPH." *International Journal of Applied Mathematics* 38, no. 5 (2025): 1119-1127.
14. [14] Kavitha, S., S. Robinson Chellathurai, and J. John. "On the forcing connected domination number of a graph." *Journal of Discrete Mathematical Sciences and Cryptography*20, no. 3 (2017): 611-624.
15. Davila, Randy R. "Bounding the forcing number of a graph." PhD diss., 2015.
16. [16]Dominic, Charles. "Dom-forcing sets in graphs." *arXiv preprint arXiv:2411.00580*(2024).
17. Livingston, Marilynn, and Quentin F. Stout. *Perfect dominating sets*. Ann Arbor, MI, USA: University of Michigan, Computer Science and Engineering Division, Department of Electrical Engineering and Computer Science, 1990.
18. D. Ahila Jeyanthi and T. M. Selvarajan. "Enhanced Mesh Network Using Novel Forgotten Polynomial Algorithm for Pharmaceutical Design." Intelligent Automation and Soft Computing, 33(1), (2022), 669-680.
19. [19] T.M. Selvarajan and D. Ahila Jeyanthi, "Dominating sets and dominationpolynomial of fan related graphs," Journal of Advanced Research inDynamical and Control Systems, vol. 12, no. 3, pp. 233–243, 2020.
20. [20] G. Sheeba and T. M. Selvarajan, "Construction of an Energy-Efficient Detour Non-Split Dominating Set inWSN", Computers, Materials & Continua 2022,73(1), 689-700.https://doi.org/10.32604/cmc.2022.021781.