Flu Trend Prediction - Regression Random Forest with GP leaves Algorithm and its Applications

Anonymous Author(s)
Affiliation
Address
email

Abstract

Random forest makes use of data ensembles to generalize the regression output, which has been proved to be more precise. The technique has been successful applied to social networks, medicine and games. Another machine technique is Gaussian Process that operates on the distribution of function and provides regression for flexible non-linear functions. It provides both expectation and confidence in the unknown data point. Convenient and flexible as GP is, it is not enough to model various piecewise functions in practice. Inspired by the regression forest, we are interested in using random forest with GP regression leaves to investigate piecewise models and give the prediction. Quick review of all of these algorithms will be presented. The new algorithm details will be derived. Its applications in flu trend prediction and geology measurement regression will also be simulated.

1 Introduction

Machine learning gives a smart and automatic way to predict the trend with a math measurement of its probability. Prediction helps people look for the optimum, simulate real scenarios and so on. Machine learning techniques take advantage of the fact the everything in the nature is related and smooth in some way, which provides insight into objects in another geographic position and time point. Significant leaps have been made in human society thanks to machine learning algorithms. Application fields extend from social network, more sophisticated game devices like Kinect, various mobile terminal Apps to medical detections and social phenomena predictions. ”Big Data” has been widely used to make a much more smart life for human beings. For example, in terms of the recent hot topic of bird flu burst as spring comes, trend predictions are being conducted every year. Google collects the related search keys such as ”headache” ,”fever”, ”runny nose”, analyze them and give a forecast if a flu breaks out. Based on the data of the flu diagnose cases sampled every week in the previous ten years in Canada, we are interested in designing the model of regression random forest based on Gaussian process to do a data based flu burst analysis prediction.

Gaussian process is stochastic process that operates over the distribution of functions. It is used conveniently to specify continuous and flexible regression function values. The kernel plays an important role in interpreting properties of data after Gaussian Process. Suitable kernel model with good parameters offer us a significantly better understanding of the data set. Dynamic Gaussian models for human motion[6], 3D people tracking[5] are all popular application of the GP.

Random forest is an ensemble learning algorithm developed by Leo. Breiman. It consists of a bunch of decision trees that can be used for classification and regression. The hierarchy structure of decision tree allows for capturing rules among the data features to give a precise classification or regression. Further, many decision tree algorithms has been developed and various applications have been conducted using this tool. For example, [1] adopted decision tree to solve land cover mapping problems via remote sensing. [2] uses decision tree algorithm for management of Parkison’s disease treatment guidelines. Further, alternative or improved algorithms of decision tree, for instance [3] have been developed. Random forest adopts bagging technique to combine and average a bunch of trees and has a more precise classification or regression effect. It is tailed for large amounts of data with deep dimensions. Application of random forest in medicine, multimedia and predictions are also popular and powerful, for example in [4].

An combination of Gaussian Process with decision tree and random forest, or alternatively, training a decision tree and random forest with GP leaves to do prediction is a new idea beyond what text-book interpreted. Prediction using random forest with GP leaves is a better algorithm for big data regression. We dedicate to figure out this algorithm and do several regression using this algorithm.

In this paper, a fundamental review of how Gaussian Process, decision tree and random forest functions will be introduced. Gaussian kernel width effects will be interpreted in detail and a maximum likelihood approach of optimizing kernel parameters will be conducted and illustrated. Algorithm of constructing random forest with GP leaves will be provided. In the experiment section, several simulation results will be shown. A comparison of single GP, decision GP and further, random forest with GP leaves will be provided. A prediction of flu trend using the new algorithm will be made. In addition, another regression experiment of geology measurement will also be conducted. Finally, a conclusion and future work will be presented.

2 Gaussian Process

Gaussian process is regression over functions and provides expectation and confidence of prediction points according to known sample points. We have the general expression like:

$$f(x)\ GP(m(x), k(x, x'))$$

where

$$m(x) = E(f(x)), k = E[(f(x) - m(x))(f(x') - m(x'))^T]$$

,k represents the kernel function.

At a new point, the mean and deviation of the value is predicted by the GP posterior.

$$p(f|D) = \frac{p(D|f)p(f)}{D}$$

in which D represents the sampling point. Using only limited sample point to predict the new entries' expectation and confidence using Bayes Rule is the core technique of GP. We are interested in finding the mathematic expression of new data's mean and deviation via the observed ones. The training set is denoted by $D = (x_i, f_i), i = 1 : N$, where $f_i = f(x_i)$. Given the test set X_* of size $N_* \times D$ and the noise $\epsilon\, \mathcal{N}(0, \sigma_y^2)$, the posterior function output f_* follows this distribution:

$$\begin{pmatrix} y \\ f_* \end{pmatrix} \sim \mathcal{N}\big(0, \begin{pmatrix} K_y & K_* \\ K_*^T & K_{**} \end{pmatrix}\big)$$

where y is the noisy GP regression prediction function.$K = \kappa(X, X)$ is $N \times N$, $K_* = \kappa(X, X_*)$ is$N \times N_*$, and $K_{**} = \kappa(X_*, X_*)$ is $N_* \times N_*$ The posterior

$$p(f_*|X_*, X, y) = \mathcal{N}(f_*|\mu_*, \Sigma_*)$$

where $\mu_* = K_*^T K_y^{-1} y$ and $\Sigma_* = K_{**} - K_*^T K_y^{-1} K_*$ Note that the function values are normalized with mean zero. Given a bunch of data in practice, normalization is necessary to do Gaussian Process.

2.1 Effect of Kernel Parameters

As we have discussed in the expression of GP before, covariance of function values are given by kernel functions and their polynomials.

$$\kappa(x, x') = \sigma_f^2 \exp(-\frac{1}{2l^2}(x - x')^2)$$

Given the observed data, parameters in kernel directly effect the performance of GP prediction. Adjusting kernel width parameters in order to improve performance in terms of the preciseness of

GP prediction is a worthy investment. We conduct the minus-log-likelihood algorithm to fit the parameters of the kernel width. Likelihood is a function of the training set.

$$p(f|X, \mu, \Sigma) = |2\pi\Sigma|^{-1/2} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}$$

here μ, Σ are kernel polynomials with kernel parameters. By differentiating the target function $-logp(f|X, \mu, \Sigma)$, setting first derivation to zero and solve the equation, optimal parameters in terms of the training set are obtained. Differential entropy of both the original parameters and optimal parameters are calculated as a standard to evaluate the performance. Fig1. is an simple example illustrating the comparison of GP performance before and after optimization. Optimal values of width of kernel are provided.

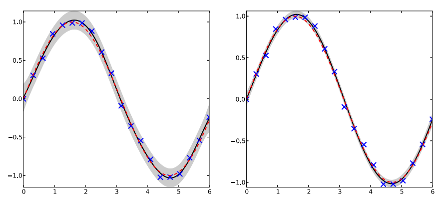


Figure 1: A comparison of kernel parameters.Defaul pamameter ($l = 1.0, \sigma_f^2 = 1.0, \sigma_y^2 = 0.1$) differential entropy 1.16. Optimized paramater ($l = 1.92, \sigma_f^2 = 1.20, \sigma_y^2 = 0.04$)differential entropy -15.3

3 Random forest with GP leaves

Ra It is discovered that the ensembles trees with slight differences make much higher preciseness than a single tree.GP regression over a random forest not only provides expectation and deviation of new testing data, but also generates a fairly accurate result with higher probability. The natural property of random forest makes the random forest with GP leaf algorithm perfect when doing regression with complex features, or multi-dimension data sets.

3.1 Decision tree, Classification and Regression tree

Decision tree is a widely used tool in machine learning to make decisions. From face detection, text filtering, to photograph classification, decision trees make classification problem tractable. Common use of decision tree are classification and regression. The tree ingredients include nodes and edges. Nodes are where the data stream flow apart and the edges are destination of data. Therefore, each node provides the function of telling apart the feature of the data and each leaf corresponds to a decision of what the data is. Once a decision tree is set up, new data point answers questions at each node and falls from root node all the way down to a leaf. The leaf node the data falls to gives the classification or a distribution of this data. Prediction model for the tree t is

$$p_t(c|v)$$

where v is the testing data and c is the class label. Or, in the case of regression tree, is a continuous variable which makes a posterior over the desired interval. Tree nodes split the data by examing a particular feature of the data and compare it with a threshold given by the tree node. Therefore, constructing the decision tree model is equivalent to optimizing each split node j such that

$$\theta_j^* = argmaxI_j$$

Before introducing the GP leaf based random forest, we may first review the decision tree training algorithm, which is a technique for splitting complex problems into a hierarchy of simpler ones and giving prediction for a new point in a view of probability. Table 1 gives the algorithm.

The entropy of a data set represents the amount of disorder within the disorder. By subtracting the entropy at each child leaf, the remaining amount of entropy, known as information gain, measures

Table 1: Algorithm1-Decision tree with GP leaves algorithm

training a decision tree with GP leaves
while $treedepth < dandleafdata > n$
determine the feature set F we want to try
for k= 1 to K choose k from distinct values of feature f (k=1,2,...K)
a. split data from f=k, do kernel width optimization
b. calculate information gain $I_k = H(S_k) - \sum_{j=L,R} \frac{\lvert S_k^j \rvert}{\lvert S_k \rvert} H(S_k^j)$
choose k with the largest information gain
do iteration for chiled node until stop criteria satisfies

Table 2: Algorithm2-Random Forest with GP leaves algorithm

constructing random forest with GP leaves
for t= 1 to N
a. bootstrapping sampling data for each tree t from the training set
b. grow a tree T_t for the bootstrapped data following the algorithm1
output ensemble of trees

how much uncertainty the split removes from the original disorder. In the application of classification, the entropy indicates data labels and their consensus. In the application of regression, the entropy represents how well the data fit into the regression model or in a cluster, or in the same height, depending on desired regression targets. An in tree regression with GP leaves, the entropy is the differential entropy over the data on the node assuming that split has been made and GP leaf has been generated,

$$p(y|x, \mu, \Sigma) = |2\pi\Sigma|^{-1/2} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}$$

$$h_i(p) = \int p(y_i) log p(y_i) \mathrm{d}x = \frac{1}{2} log(2\pi e)^N |\Sigma|$$

given that GP actually is gaussian distribution over functions. Here the index i means the ith node of the tree.Σ is the covariance matrix represented with kernel. Here we cite it again:

$$\Sigma = \begin{pmatrix} K_y & K_* \\ K_*^T & K_{**} \end{pmatrix}$$

$K = \kappa(X, X)$ is $N \times N$, $K_* = \kappa(X, X_*)$ is$N \times N_*$, and $K_{**} = \kappa(X_*, X_*)$ is $N_* \times N_*$ That is to say, when we are computing the node data entropy and processing the node splitting in GP tree regression, we are actually dealing with kernels, which is a function of X's and three parameters. Moreover, Gaussian kernel is only a typical kernel model of fair performance; other forms of kernels can also be tested. In this paper, we only take Gaussian kernel into account, as we have already investigated in section 2. Since GP tree regression algorithm is super related to, and significantly inflected by kernel parameters, we dedicated to optimize the kernel parameters before figuring out the split feature and threshold in an internal node. Optimizing the kernel parameters before applying the regression of new data points removes the effect of redundant entropy brought by the kernel parameters. We choose the best split point in each node such that entropy at each leaf node arrives at a minimum. Therefore, the prediction of a new point will achieve a lower uncertainty, or high probability.

3.2 Random Forest with GP leaves

Bootstrapping technique conducts sampling with replacement of the training data and impose the bagging assembles on each tree. Predictions of each tree are averaged together to produce the generalized classification or regression result, which is of much higher accuracy. Generalization is the core of the random forest with bagging compared with the decision tree. Accendental points will not taken into account by the random forest. Random forest is more suitable for data with large dimensions. Also, it can also distribute the computation over different devices, therefore, decrease the burden in large scale data sets.The algorithm is shown in Table 2.

4 Experiment

4.1 A comparison of single GP and random forest with GP leaves

The most conventional GP is perfect when predicting smooth curves such as a sin function. But it fades in trying much more complex data sets, with faults and maybe with high dimensions. The reason is that the core idea of GP is constructing the predictions of unknown function values based on the continuity of the arguments. When the arguments fail in continuity, the Gaussian Process can only produce smooth curves, which are not what we want.

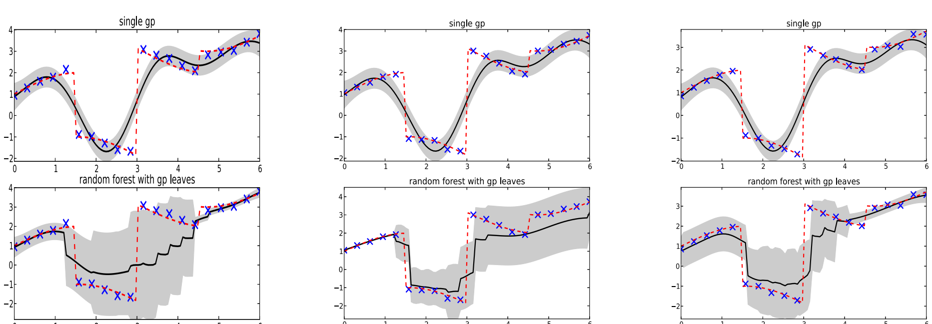
The first experiment shows an intuitive comparison of single gp and random forest gp with 1-D training and test data. According to the experiment result, even though single gp always return a nice smooth regression, it misses a lot of true data point especially at the edges. Random forest obviously is stronger at capturing the fault edges. Over-fitting performance of the random forest gp is within expectation, because of the small training and test set and low dimension. Further, the Beta version experiment gives us an experience that random forest parameters play an important role in the regression performance.

4.2 Flu trend regression–2004 to 2012, Canada

We are interested in applying the derived algorithm into flu trend regression. Data set is from the CanadaFlueTrend website. The two-dimension data set records the number of diagnosed cases throughout the ten provinces in the past eight years. Half of the data set is used for training and another half used for testing. We can see from the simulation results that single gp show weakness in the edge regression when the true function is piecewise. It is meaningful in doing regression for the flu trend since gaussian process can also be used as prediction. If we can get the number of key words from search engines or social networks, combined with the flu curve of previous years, we can hopefully forecast whether large scale flu is happening and how severe it is.

4.3 geology application–seawater measurement regression

The third experiment we conducted is the seawater measurement regression. Data set is from UBC ocean research lab. The data set is in three dimensions-oxygen content, temperature and salinity of a seawater field. There are correlations among these three features of the data. Gaussian Process regression is operated over temperature and salinity of the data such that we can predict the corre-



(a) 5 trees, 3 depth, 3 minimun leaf data (b) 10 trees, 2 depth, 3 minimun leaf data (c) 10 trees, 2 depth, 4 minimun leaf data

Figure 2: A comparison of single GP and random forest with GP leaves,above: single gp, below:random forest with gp leaves

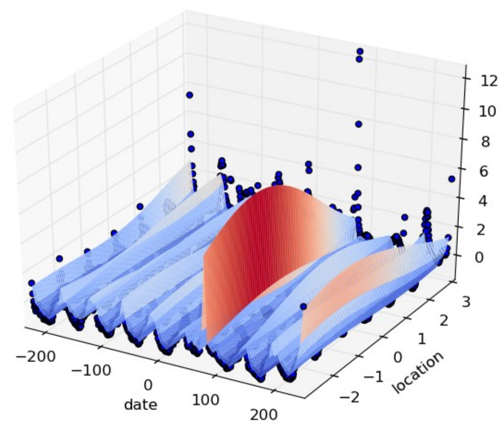


Figure 3: Gaussian Process regression of flu trend. Time: from 2004 to 2012. Location: Canada provinces. Function value: the cases of diagnosed flu per week. All data normalized.

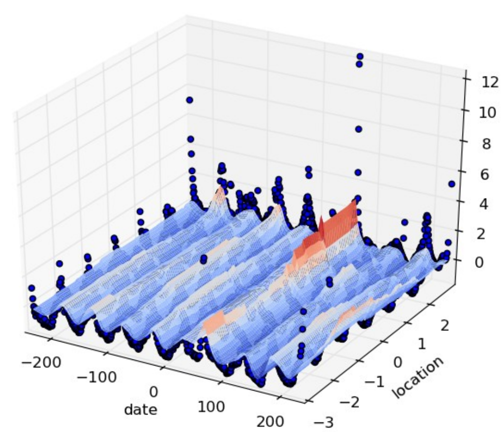


Figure 4: Random forest regression of flu trend. Tree: 15 trees. 4 depth. 10 minimum leaf data Time: from 2004 to 2012. Location: Canada provinces. Function value: the cases of diagnosed flu per week. All data normalized.

sponding oxygen content given the temperature and salinity features of the data. Fig.6 and Fig.7 show the experiment results for single and random forest regression trees respectively. Scatters are training data and surface plot is the regression.Half data are training set and the other half are testing set. The training and testing data are randomly chosen, as we can see that their MSR are close. Too much fluctuations are seen in the single gp case. Wheresas in the version of random forest, the algorithm removes the influces of the "edge", regards them as the child gp, does gaussian process in splitted domain and output an average. Therefore, a much more smooth and real prediction can be made by the random forest regression.

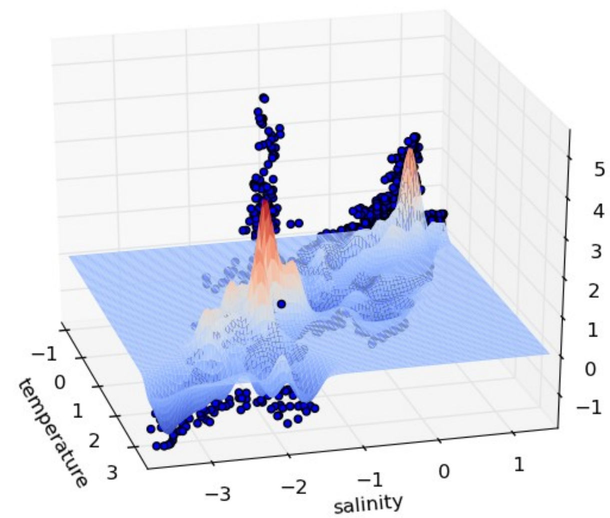


Figure 5: Single tree regression of seawater measurement. training MSE 412.55. test MSE 643.28. All data normalized.

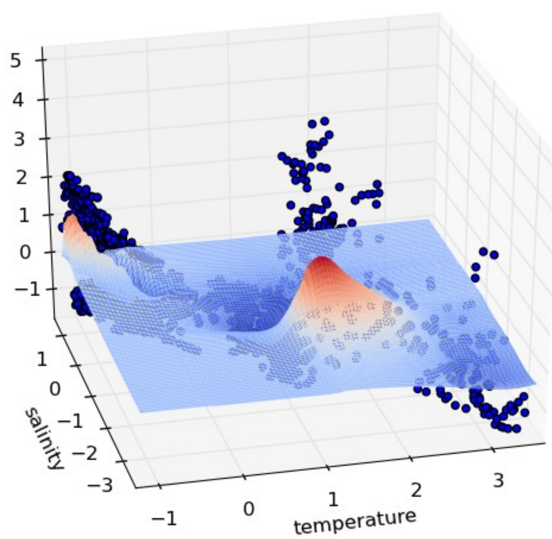


Figure 6: Random forest regression of seawater measurement. data size 2710. train data size= testing data size= 1355. 40 trees. 10 depth. 20 minimum leaf data. training MSE 599.63. test MSE 561.75. All data normalized.

5 Conclusion

Gaussian Process examines the relations in the desired variables and provides both expectation and confidence of the function production. But the performance evaluation significantly fluctuates with different kernel width parameters. The kernel influences are explored and its parameter widths are optimized in order to achieve a best matching between the prediction and the true function. But single GP is still not enough when faced with high dimension data sets or piecewise data sets, which are very common in real world. Driven by the motivation of processing the prediction this kind of data sets, regression trees with GP leaves are constructed and furthermore, expanded to random forest in order to have a better generalization and prediction. In training the decision tree, differential entropy of GP leaves are computed out as a reference of information gain. Kernel parameters are optimized before making predictions and therefore, quite amount of redundancy of the node data entropy and its negative influence on fairly splitting the node is wipes out. We genuinely provide the algorithm of training the GP tree and its further expansion towards random forest with GP leaves. In performance evaluation, we see the regression performances of both single gp and random forest gp in two applications. It is shown that when data set becomes large random forest presents the advantage in the regression of edges. Considering the fact that gp is an ideal basis in doing Bayesian Optimization, futher work include Byesian optimizaiton using random forest with GP leaves.

References

[1] Friedl, Mark A., and Carla E. Brodley. ”Decision tree classification of land cover from remotely sensed data.” Remote sensing of environment 61.3 (1997): 399-409.

[2] Olanow, C. Warren, and William C. Koller. ”An algorithm (decision tree) for the management of Parkinson’s disease Treatment guidelines.” Neurology 50.3 Suppl 3 (1998): S1-S1.

[3] Olaru, Cristina, and Louis Wehenkel. ”A complete fuzzy decision tree technique.” Fuzzy sets and systems 138.2 (2003): 221-254.

[4] Prasad, Anantha M., Louis R. Iverson, and Andy Liaw. ”Newer classification and regression tree techniques: bagging and random forests for ecological prediction.” Ecosystems 9.2 (2006): 181-199.

[5] Urtasun, Raquel, David J. Fleet, and Pascal Fua. ”3D people tracking with Gaussian process dynamical models.” Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. Vol. 1. IEEE, 2006.

[6] Wang, Jack M., David J. Fleet, and Aaron Hertzmann. ”Gaussian process dynamical models for human motion.” Pattern Analysis and Machine Intelligence, IEEE Transactions on 30.2 (2008): 283-298.