# ACCURATE POSTERIOR PROBABILITY ESTIMATES FOR CHANNEL EQUALIZATION USING GAUSSIAN PROCESSES FOR CLASSIFICATION

*Fernando Pérez-Cruz**

DTSC University Carlos III in Madrid.
Avda. de la Universidad 30. 28911,
Leganés (Madrid), Spain.
`fernando@tsc.uc3m.es`

*Pablo Martinez-Olmos, Juan José Murillo-Fuentes*†

DTSC University of Sevilla.
Paseo de los Descubrimientos s/n,
41092 Sevilla, Spain.
`murillo@us.es`

## ABSTRACT

In this paper we propose to use Gaussian processes for classification (GPC) for solving the channel equalization problem. GPC provides not only accurate decisions as other nonlinear machine learning tools do, i.e. support vector machines or neural networks, but it also assigns posterior probabilities to each one of its output. This is a significant advantage of GPC with respect to other machine learning tools for channel equalization, because the channel decoder benefits from its soft outputs to provide significantly better error correcting capabilities for the entire communication system. As, for previous schemes, the channel decoder had to rely on the hard decisions given by the equalizer, because the output of these methods cannot be transformed into posterior probabilities. We show that the GCP equalizer is able to estimate posterior probabilities accurately in a variety of real digital communications channel.

***Index Terms*—** Gaussian processes, Equalizers, Nonlinear estimation

## 1. INTRODUCTION

Channel equalization is a major issue in digital communications because the channel affects the transmitted sequence with linear and nonlinear distortions. Inter-symbol interference (ISI), which accounts for the linear distortion, consists in spreading the received symbol's energy through several time intervals. ISI occurs as a consequence of both multipath and the limited bandwidth of the channel. Besides, the channel cannot be considered linear due to the presence of nonlinear devices such as amplifiers and converters [1]. Channel equalization minimizes these distortions to recover the transmitted sequence. In wireless communications, the channel is time variant and thus it is necessary to send a training sequence in every burst. Hence, short training sequences are a prerequisite for achieving good spectral efficiency.

Traditionally, equalization of linear channels has been considered equivalent to inverse filtering, where a linear transversal equalizer (LTE) is used to invert the channel response and its parameters are usually adjusted using a minimum mean square error (MMSE) criterion. These solutions are simple but suboptimal. The optimal solution is based on maximum likelihood sequence estimation. However, its complexity grows exponentially with the dimension of the channel impulsive response (Viterbi Algorithm), and it introduces an unknown delay. Alternatively, neural networks (NNs) can be used to approximate the optimal solution at a lower computational cost. Several NNs schemes have been proposed to address this problem with varying degrees of success, such as the multi-layered perceptron (MLP) [2], radial basis functions (RBFs) [3], recurrent RBFs [4], self-organizing feature maps [5], wavelet neural networks [6], Kernel Adaline (KA) [7], support vector machines (SVMs) [8] and Gaussian Processes for Regression (GPR) [9]. Such structures usually outperform the LTE, especially when non-minimum phase channels are encountered. They can also compensate for nonlinearities in the channel. The major drawback of such schemes is the need for long training sequences to achieve optimal equalization.

In modern digital communication systems the bits are transmitted through a channel encoder to introduce redundancy in the bit sequence. The channel decoder exploits this redundancy to reduce the probability of error produced by the channel. The channel decoder can work either using Hamming distance, if only a sequence of bits is provided by the equalizer, or using the Euclidean distance, if a posterior probability for each bit is available at the equalizer's output. The former is known as *hard* decoding and the latter as *soft* decoding. The soft decoding is able to achieve significantly lower bit error rates for a given signal to noise ratio, because the equalizer provides information about how likely it is that each bit is in error. While the hard decoder has to be equally wary of all the received bits, as it has no information about which ones could be in error.

*This work has been partially supported by the Spanish MEC (TIC2006-13514-C01-01/TCM).

†This work has been partially supported by the Spanish MEC (TEC2006-13514-C02-02/TCM).

The previously mentioned nonlinear procedures for channel equalization only provide bitwise decisions and cannot produce the posterior probability for each received bit. These pattern recognition methods are trained to reach minimum probability of error. Therefore, their soft-outputs cannot be interpreted as posterior probabilities and they have to be threshold before being fed into the channel decoder. Hence, the procedures above can only be decoded using a hard decoder.

The support vector machine soft output can be transformed into posterior probabilities using the so-called Platt's method [10]. This method passes the SVM output through a sigmoid to obtain the posterior probabilities. Platt's method is not very principled, as Platt himself explains in his paper, but it typically provides good predictions. However, in some cases its probability predictions are not accurate, as shown in [11], and therefore it cannot be said a priori how good its fits will be. In this paper we propose to use a novel nonlinear tool for channel equalization known as Gaussian processes for classification (GPC). This Bayesian tool provides the posterior probability for each decision allowing the use of a soft decoder. This property makes it unique among the nonlinear methods for channel equalization. In a previous paper [9], we proposed to use Gaussian processes for regression (GPR) to solve the channel equalization problem. The GPR solution has the advantage of being analytical, which makes it easy to compute, while to obtain the GPC solutions we need to use approximations for its posterior distribution, which can be time consuming. However, the GPR solution cannot be interpreted as posterior probabilities for each output bit, because it assumes it is solving a regression problem with Gaussian noise. And that is not the case for channel equalization, which can be cast as a classification problem.

The remaining of the paper is organised as follows. In Section 2, we present the Gaussian processes for classification. We review the channel equalization problem as a pattern recognition task in Section 3. Section 4 shows, by means of computer experiments, that GPC matches the performance of other nonlinear methods for channel equalization, while being able to provide good posterior estimates for each bit. We conclude in Section 5 with some remarks and further work.

## 2. GAUSSIAN PROCESSES FOR CLASSIFICATION

Gaussian Processes for Classification is a supervised machine learning tool for predicting the posterior probability of the label, $y^*$, given an $d$-dimensional input vector, $\mathbf{x}^*$, and a training data set ($\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$, $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{\pm 1\}$):

$$p(y^* = 1|\mathbf{x}^*, \mathcal{D}) \tag{1}$$

For the training set, we will assume that the $n$ training patterns have been i.i.d. sampled from $p(\mathbf{x}, y)$ and that the probability of each sample can be computed using a latent function $f(\mathbf{x}_i) = f_i$:

$$p(y_i|f(\mathbf{x}_i)) = \Phi(y_i f(\mathbf{x}_i)) \tag{2}$$

where $\Phi(\cdot)$ is typically a sigmoid or a Probit [12]. Due to independence, we can express the joint distribution as:

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^n p(y_i|f(\mathbf{x}_i)) = \prod_{i=1}^n \Phi(y_i f(\mathbf{x}_i)) \tag{3}$$

The GPC assumes that, given the input samples $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ the latent function $\mathbf{f} = [f_1, \ldots, f_n]$, its prior probability is a multidimensional Gaussian density function (i.e., a Gaussian process) $p(\mathbf{f}|\mathbf{X}) = \mathcal{N}(\mathbf{0}, \mathbf{K})$, where the covariance matrix $\mathbf{K}$ is a positive definite kernel function. Therefore we can compute its posterior using Bayes rule:

$$p(\mathbf{f}|\mathcal{D}) = \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X})}{p(\mathcal{D})} \tag{4}$$

This posterior is non-Gaussian because the likelihood in (3) is not Gaussian.

The latent variable at a test point $f(\mathbf{x}^*)$ is:

$$p(f^*|\mathcal{D}, x^*) = \int p(f^*|\mathbf{f}, \mathbf{X}, x^*)p(\mathbf{f}|\mathcal{D})d\mathbf{f} \tag{5}$$

and the predictive class becomes:

$$p(y^*|\mathcal{D}, x^*) = \int p(y^*|f^*)p(f^*|\mathcal{D}, x^*)df^* \tag{6}$$

These integrals are intractable to compute. But the posterior in (4) is typically unimodal and a Gaussian approximation is likely to approximate it well. The two standard approximation for GPC are either Laplace or EP [12]. As shown in [11] the EP approximation does a better job describing posterior probabilities and this is the one we will be using in this paper.

There are several issues that we have not described in detail in this short introduction for GPC due to lack of space. Details on how the kernel is selected and trained, why GPC is a good model for classification problems and why it approximates the posterior probabilities. This information, together with deeper insight about Gaussian processes for machine learning, is perfectly described in [12]

## 3. CHANNEL EQUALIZATION

In Figure 1 we depict a simple band-based model to describe a dispersive communication channel, yet typically used [13]. The transmitted signal $b(t)$ is a BPSK modulated signal, i.e., an independent and equiprobable sequence of symbols with values $\{-1, +1\}$, and $n(t)$ represents additive white Gaussian noise (AWGN). The linear time-invariant impulse response for the channel is given by:

$$C(z) = \sum_{i=0}^{n_c-1} c_i z^{-i} \tag{7}$$

where $n_c$ denotes the channel length. A linear transversal equalizer (LTE) is typically used to invert the channel response for recovering the transmitted signals [14]. The LTE

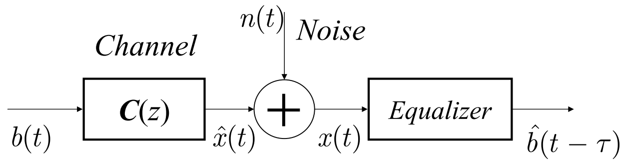


**Fig. 1**. Simple discrete-time transmission channel model.

linearly combines the last $m$ samples from the channel to estimate the transmitted bit, i.e.:

$$\hat{b}(t-\tau) = \text{sign}\left(\mathbf{w}^\top \mathbf{x}(t)\right) \quad (8)$$

where $\mathbf{x}(t) = [x(t), x(t-1), \ldots, x(t-m+1)]$ and $\tau$ is a delay to ensure that enough information is available in the LTE to estimate the transmitted bit. The weight vector $\mathbf{w}$ is typically computed applying the MMSE criterion [14], as it offers a trade-off between minimising the effects of the dispersive nature of the channel and its noise. Nonlinear channel equalizers transform $\mathbf{x}(t)$ prior to computing the weight vector to obtain solutions that achieve minimum BER.

## 4. EXPERIMENTS

In this section we include some experimental results of channel equalization for a linear channel model. We actually illustrate that the novel approach based on GPC does not only provide optimal results but also accurate predictions for posterior probabilities. We have first used a simple channel model that will actually allow us depict the percentile curves for the optimum decision function and the one provided by the GPC equalizer. The channel model is given by:

$$C(z) = 1 + 0.5z^{-1} \quad (9)$$

In Figure 2a we have depicted the $BER$ versus the $E_b/N_0$ when 100 and 800 samples have been used for training a two tap equalizer. We have limited ourselves to low signal-to-noise ratio since we are only interested in high probability of error at the equalizer's output. The resulting $BER$ will be further reduced after the channel decoder. The channel decoder has not been simulated in this paper, but any state-of-the-art procedure as Low-Density Parity-Check (LDPC) codes or Repeat and Accumulate (RA) codes [15] will provide $BER$ bellow $10^{-5}$ for $E_b/N_0$ as low as 1 or 2dB. Also, we have not compared the $BER$ of the GPC equalizer with other nonlinear method as it performs as the GPR equalizer in [9], in which a full comparison is made with state-of-the-art channel equalization methods. In any case, we are not as much interested in the $BER$ performance as we are on the predicted probabilities given by the GPC equalizer, as accurate prediction for these posterior probabilities allows optimal performance for the channel decoder. In Figure 2b we have depicted the posterior probabilities of the optimum classifier

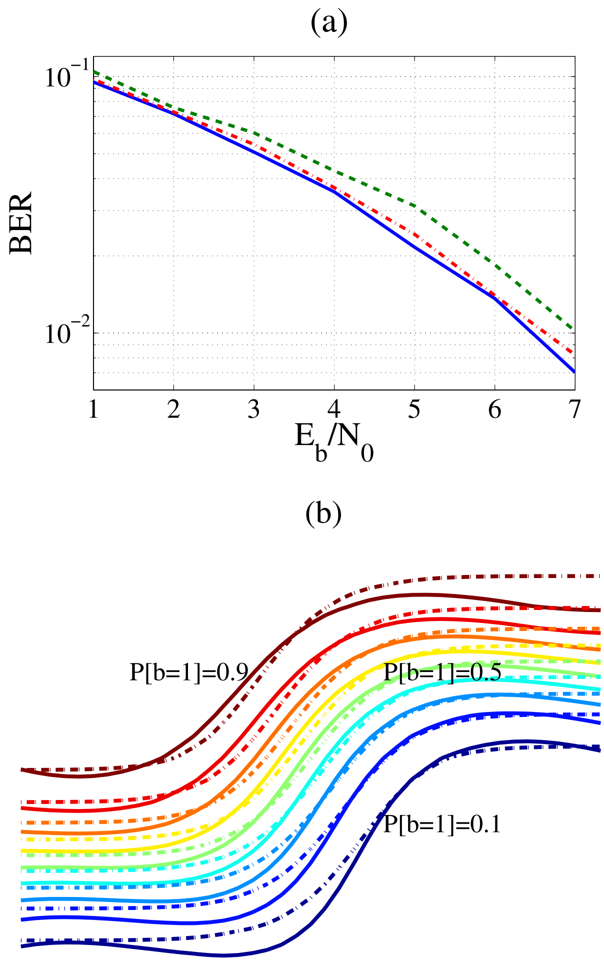


**Fig. 2**. In (a), $BER$ vs. $E_b/N_0$ for the channel in (9). The solid line represents the optimal $BER$. The dashed and dash-dotted lines, respectively, represent the $BER$ for the GPC equalizer with 100 and 800 training examples. In (b), we represent the percentile curves for a signal to noise ratio of 2dB for both the optimum decision function (solid) and the GPC equalizer trained with 800 examples (dashed).

(solid lines) and the predicted probabilities of the GPC equalizer (dashed lines). These curves are almost identical, which explain the perfect fit between the GPC equalizer and the optimum classifier.

We have shown in Figure 3-6 the calibration curves for the GPC equalizer compared to the optimum posterior probabilities for $10^4$ test samples. We have shown these plots for training sequences with 100, 200, 400 and 800 samples. In the horizontal axis we report the true posterior probability and in the vertical axis the probability predicted by the GPC equalizer. In these plots we can see that, as we increase the number of training samples, the points concentrate on the diagonal axis, which represents the perfect fit between GPC equalizer and the optimum. It is important to notice that although the

$BER$ does not improve when we move from 100 to 800 training examples, as it can be seen in Figure 2a, the calibration curves vary significantly as the number of training examples increases. The GCP equalizer is able to find the 50% percentile accurately with few training example, which explains the $BER$ performance. It is also understandable that we will need more training samples to provide with accurate posterior probabilities predictions compared to only predicting the right label, as we are asking for a better description about the probability of each received bit. Hence, the longer the training sequence is the better the matched between the predicted and optimum posterior probabilities will be.

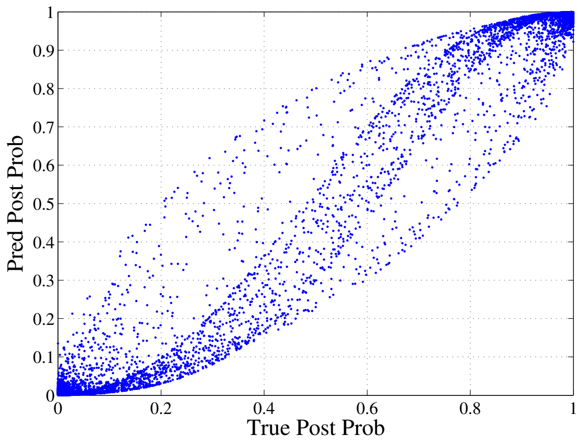


**Fig. 3**. Calibration curve for a 2dB signal to noise ratio channel for 100 training samples

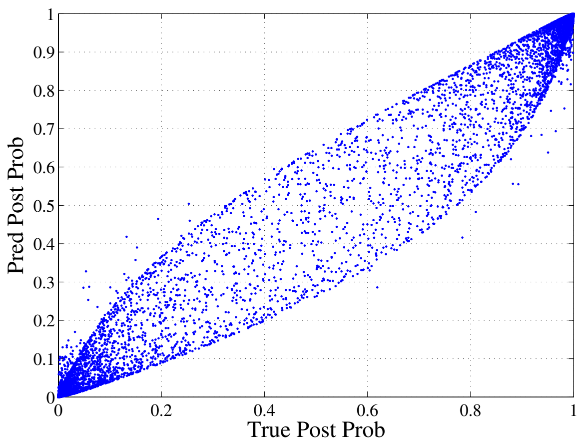


**Fig. 4**. Calibration curve for a 2dB signal to noise ratio channel for 200 training samples.

Finally, we propose to use a more realistic channel model

$$C(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2} \tag{10}$$

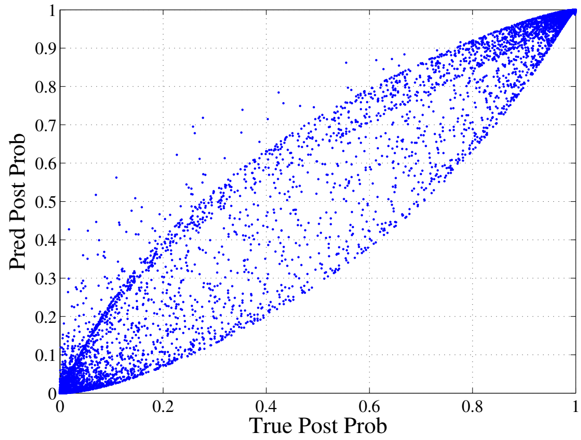


**Fig. 5**. Calibration curve for a 2dB signal to noise ratio channel for 400 training samples.

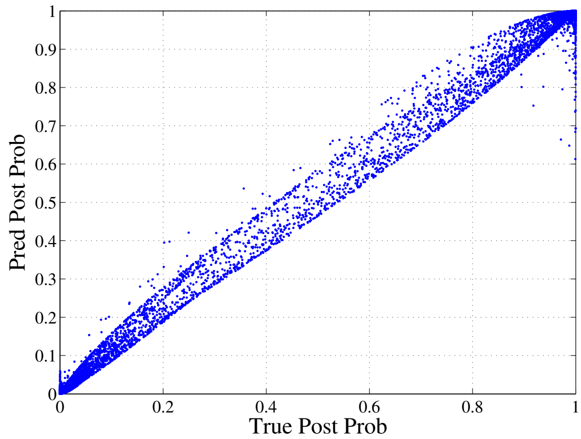


**Fig. 6**. Calibration curve for a 2dB signal to noise ratio channel for 8100 training samples

as proposed in [7] to model radiocommunication channels. For this channel we report in Figure 7 the calibration curves for 800 training samples with a four-tap unit-delay equalizer. In this more realistic scenario the GPC classifier performs as well as it did in the previous example and the reported probabilities are very close to each other, so they can be used in a soft channel decoder to significantly reduce the $BER$ at its output.

To train the GPC we have used Carl Rasmussen software at *http://www.gaussianprocess.org* and we have used a Gaussian kernel with isotropic covariance matrix.

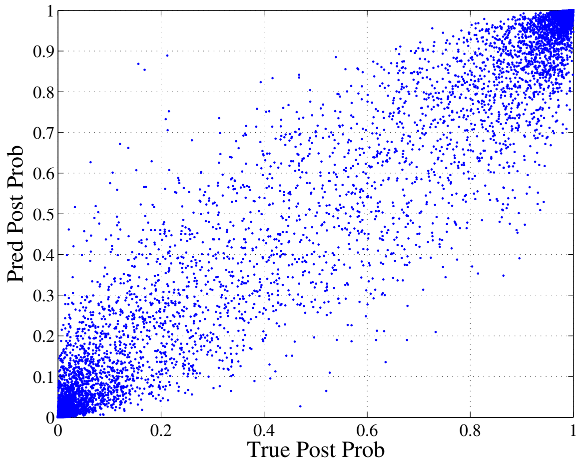


**Fig. 7**. Calibration curve for a 1dB signal to noise ratio with 800 training examples for the channel in (10).

## 5. CONCLUSION AND FURTHER WORK

In this paper we have shown how Gaussian processes for classification can be used for solving the channel equalization problem. This nonlinear tool performs as well as other neural network and kernel methods. Moreover it provides extra information for each one of its decisions, i.e. the posterior probability of being in the correct class. This information can be used by the channel decoder to significantly reduce the BER for low signal to noise ratio. This characteristic is not shared by the other nonlinear machine learning tools, as they can only provide hard decisions as outputs. We have shown that, as the number of samples increases, the predicted probabilities tend to the true posterior probabilities.

We have not checked how the channel decoder will perform with these estimated probabilities and which will be the final performance of the complete communication system. This has been left as a further work, as it will not only depend on the quality of the predicted probabilities, but also on the error correcting code. We need to study further the interaction between the GPC equalizer and the used error correcting codes to find synergies that can be exploited to improve the BER at the decoder's output, which is the final goal when describing a communication system.

## 6. REFERENCES


[1] R. Parisi, E.D. Di Caudio, G. Orlandi, and B.D. Rao, "Fast adaptive digital equalization by recurent neural networks," *IEEE TSP*, vol. 45, 1997.

[2] G.J. Gibson, S. Siu, and C.F.N. Cowan, "The application of nonlinear structures to the reconstruction of binary signals," *IEEE TSP*, vol. 39, 1991.

[3] S. Chen, G.J. Gibson, C.F.N. Cowan, and P.M. Grant, "Reconstruction of binary signals using an adaptive radial-basis-function equalizer," *Signal Proc.*, vol. 22, 1991.

[4] J. Cid-Sueiro, A. Artés-Rodríguez, and A.R. Figueiras-Vidal, "Recurrent radial basis function networks for optimal symbol-by-symbol equalization," *Signal Proc.*, vol. 40, 1994.

[5] T. Kohonen, K. Raivio, O. Simula, O. Venta, and J. Henriksson, "Combining linear equalization and self-organizing adaptation in dynamic discrete-signal detection," in *Proc. IJCNN*, San Diego, CA, June 1990.

[6] P.R. Chang and B.C. Wang, "Adaptive decision feedback equalization for digital satellites channels using multilayer neural networks," *IEEE JSAC*, vol. 13, 1995.

[7] B. Mitchinson and R.F. Harrison, "Digital communications channel equalization using the kernel adaline," *IEEE TCom*, vol. 50, 2002.

[8] F. Pérez-Cruz, A. Navia-Vázquez, P.L. Alarcón-Diana, and A. Artés-Rodríguez, "SVC-based equalizer for burst TDMA transmissions," *Signal Proc.*, vol. 81, 2001.

[9] S. Caro, F. Perez-Cruz, and J.J. Murillo-Fuentes, "Gaussian processes for regression in channel equalization," in *Proc. EUSIPCO'06*, Rhodes, Greece, Sept. 2006.

[10] J.C. Platt, "Probabilities for sv machines," in *Advances in Large Margin Classifiers*, A.J. Smola, P. L. Bartlett, B. Schölkopf, and D. Schuurmans, Eds., M.I.T. Press, 2000.

[11] M. Kuss and C.E. Rasmussen, "Assessing approximate inference for binary gaussian process classification," *JMLR*, vol. 6, 2006.

[12] C.E. Rasmussen and C.K.I. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, 2006.

[13] B. Mulgrew, S. Chen and P.M. Grant, "A clustering technique for digital communications channel equalization using radial basis function networks," *IEEE TNN*, vol. 4, 1993.

[14] J.G. Proakis, *Digital Communications*, McGraw-Hill, 1995.

[15] D.J.C. MacKay, *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, 2003.