Bruno Rodrigues Cabral

Processos Gaussianos para Aprendizado Supervisionado

Florianópolis, Santa Catarina
2021

Bruno Rodrigues Cabral

Processos Gaussianos para Aprendizado Supervisionado

Trabalho Conclusão do Curso de Graduação em Matemática e Computação Científica do Centro de Ciências Físicas e Matemáticas da Universidade Federal de Santa Catarina como requisito para a obtenção do título de Bacharel em Matemática e Computação Científica.

Universidade Federal de Santa Catarina

Orientador: Prof. Dr. Leonardo Koller Sacht

Florianópolis, Santa Catarina
2021

Bruno Rodrigues Cabral

Processos Gaussianos para Aprendizado Supervisionado

Este de Trabalho Conclusão de Curso foi julgado adequado para obtenção do Título de "Bacharel em Matemática e Computação Científica" e aprovado em sua forma final pelo curso de Matemática e Computação Científica.

Florianópolis, Santa Catarina, 2021.

Prof. Dr. Silvia Martini de Holanda
Coordenadora do Curso

Banca Examinadora:

Prof. Dr. Leonardo Koller Sacht
Orientador
Universidade Federal de Santa Catarina

Prof. Dr. Edson Cilos Vargas Júnior
Universidade Federal de Santa Catarina

Prof. Dr. Francisco Itamarati Secolo Ganacim
Universidade Tecnológica Federal do Paraná

Florianópolis, Santa Catarina
2021

Resumo

Este trabalho investiga o uso de processos gaussianos para a resolução de problemas de aprendizado supervisionado, método que vem aumentando em popularidade nos últimos anos na área de aprendizado de máquinas e vem se mostrando uma estratégia competitiva e viável. Primeiramente, tais processos são definidos e suas propriedades estudadas, com um enfoque no papel de suas funções de covariância. Com isso, seguindo uma abordagem Bayesiana, esse processos são aplicados para a resolução de problemas de regressão e classificação. Infelizmente, na maioria dos casos, a distribuição posterior obtida está computacionalmente indisponível, o que leva ao estudo de métodos de aproximação para tal distribuição. Alguns métodos são discutidos, sendo esses o método de *expectation propagation*, o método de aproximação de Laplace e o uso de *Markov chain Monte Carlo*. Um enfoque maior é dado para o método de *expectation propagation*. São apresentados algoritmos para a implementação de tais métodos, com destaque para o *nested expectation propagation* como um algoritmo para classificação por processos gaussianos seguindo o método de *expectation propagation*. As estratégias, métodos e algoritmos apresentados são então testados e os resultados obtidos apontam para o uso de processos gaussianos para a resolução de problemas de aprendizagem supervisionada ser uma estratégia viável e com bom desempenho.

Palavras-chaves: Processos gaussianos, aprendizado supervisionado, função de covariância, expectation propagation, nested expectation propagation.

Abstract

In this work, we investigate the use of Gaussian processes for supervised learning problems, a method that has been increasing in popularity in recent years in the area of machine learning and has proven to be a competitive and viable strategy. First, such processes are defined and their properties studied, with a focus on the role of their covariance functions. Thus, following a Bayesian approach, these processes are used to model regression and classification problems. Unfortunately, in most cases, the posterior distribution obtained is computationally unavailable, which leads to the study of approximation methods for such distribution. Some methods are discussed, these being the expectation propagation method, Laplace approximation method and the use of Markov chain Monte Carlo. A greater focus is given to the expectation propagation method. Algorithms are presented for the implementation of such methods, with emphasis on the nested expectation propagation as an algorithm for classification by Gaussian processes following the expectation propagation method. The strategies, methods and algorithms presented are then tested and the results obtained point towards the use of Gaussian processes for supervised learning problems as a viable strategy with good performance.

Keywords: Gaussian process, supervised learning, covariance function, expectation propagation, nested expectation propagation

Lista de abreviaturas e siglas

pdf	Função densidade de probabilidade, do inglês *Probability density function*;
pmf	Função massa de probabilidade, do inglês *Probability mass function*;
cdf	Função distribuição acumulada, do inglês *Cumulative distribution function*;

Lista de símbolos

n	Representação genérica para o número de inputs em um dataset;
D	Representação genérica para a dimensão do espaço dos inputs;
log	Função logarítmica de base e (caso nenhuma base seja especificada);
exp	Função exponencial de base e;
$\|\cdot\|$	Norma euclidiana;
$< \cdot, \cdot >$	Produto interno euclidiano;
$\mathbb{E}$	Valor esperado;
$\mathbb{E}_F$	Valor esperado de acordo com a distribuição F;
$var(X)$	Variância da variável aleatória X ou matriz de covariância do vetor aleatório X;
$cov(X, Y)$	Covariância entre a variável aleatória X e Y;
$\sim$	Distribuído de acordo com;
$\propto$	Proporcional a;
$\mathbb{P}$	Medida de probabilidade;
$\mathbb{P}$-a.s.	Convergência quase certa em relação a medida de probabilidade;
$N(\mu, \Sigma)$	Distribuição gaussiana com valor esperado μ e matriz de covariância Σ;
$\phi(x\|\mu, \Sigma)$	Densidade de uma distribuição gaussiana com valor esperado μ e matriz de covariância Σ;
$\Phi(x\|\mu, \Sigma)$	Função distribuição acumulada de $N(\mu, \Sigma)$;
$\Phi(x)$	Função distribuição acumulada de $N(0, 1)$;
diag(x)	Matriz diagonal com as entradas do vetor x na diagonal;
$\otimes$	Produto de Kronecker;
tr(X)	Traço da matrix X;
Cholesky(X)	Matrix triangular inferior L da fatoração Cholesky de X ($X = LL^T$);

Sumário

1 Introdução

Temos como objetivo aqui apresentar o uso de processos gaussianos para a resolução de problemas de aprendizado supervisionado, técnica que vem aumentando em popularidade nos últimos anos na área de aprendizado de máquinas graças ao trabalho de Rasmussen e Williams em *"Gaussian Processes for Machine Learning"* (WILLIAMS; RASMUSSEN, 2006) e vem se mostrando uma estratégia competitiva e viável.

Como (HASTIE; TIBSHIRANI; FRIEDMAN, 2009) define, em um problema de aprendizado supervisionado temos um conjunto de dados $(x_i, y_i)_{i=1}^n$, onde os pontos $(x_i)_{i=1}^n$ são chamados de inputs (do inglês para "valores de entrada") e $(y_i)_{i=1}^n$ de outputs (do inglês para "valores de saída"), e assumimos que cada x_i está relacionado com y_i de alguma forma. Nosso objetivo é definir um modelo para prever o output y^* dado um novo input x^* com base nos dados presentes, assim "aprendendo" a relação entre as duas variáveis. Chamamos de "supervisionado" por causa da presença dos outputs, que guiam o processo de aprendizagem. Os problemas de aprendizagem supervisionados são divididos em classificação, quando os outputs são dados de uma forma discreta, por exemplo, $y_i \in \{1, 2, \ldots, C\}$, e em regressão, onde os outputs são dados de uma forma contínua, por exemplo, $y_i \in \mathbb{R}^C$.

De um ponto de vista estatístico, uma possível modelagem para o problema de aprendizagem supervisionado seria assumirmos inicialmente que cada y_i é uma observação de um vetor aleatório $Y(x_i)$. Nosso objetivo se resumiria em inferir um processo estocástico $\{Y(x)\}_{x\in\mathcal{X}}$, onde $\mathcal{X}$ é o conjunto de possíveis inputs, do nosso conjunto de dados de forma que cada $Y(x^*)$ modela a distribuição dos possíveis outputs do input x^*. Essa é uma ideia central para resolver um problema de aprendizagem supervisionado por processos gaussianos. De uma forma mais geral, abordamos rapidamente tópicos de inferência estatística no capítulo 2.

Por outro lado, chamamos de um processo gaussiano qualquer processo estocástico $\{f(x)\}_{x\in\mathcal{X}}$, indexado por $\mathcal{X}$, onde qualquer coleção finita dos vetores aleatórios (com C entradas) $f(x)$ segue uma distribuição conjunta gaussiana. De forma similar às distribuições gaussianas, a distribuição de um processo gaussiano é definida pela sua função de valor médio $m : \mathcal{X} \to \mathbb{R}^C$ e pela sua função de covariância $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{C\times C}$:

$$m(x) = \mathbb{E}[f(x)];$$

$$k(x, y) = cov(f(x), f(y)) = \mathbb{E}[(f(x) - m(x))(f(y) - m(y))^T].$$

Abordamos mais profundamente os processos gaussianos no capítulo 3.

As funções de covariância são de grande importância para a estrutura de tais processos e ditam várias propriedades e, por isso, uma discussão mais aprofundada delas é interessante, assim como uma biblioteca de tais funções e operações que podem ser usadas para criar novas. Essas discussões são feitas também no capítulo 3.

Assim como qualquer processo estocástico, podemos entender essa família de vetores aleatórios como uma função aleatória de $\mathcal{X}$ para $\mathbb{R}^C$. É interessante observar que não há restrição para $\mathcal{X}$,

o que resulta em não limitarmos a forma que os inputs x_i podem tomar quando utilizamos um método por processos gaussianos para problemas de aprendizagem supervisionado.

Os métodos por processos gaussianos englobam a versão probabilística de alguns dos métodos mais conhecidos (mais especificamente, a versão Bayesiana com priors normais) como, por exemplo, a regressão linear (ou método dos mínimos quadrados), splines e, sob algumas condições, redes neurais, como discute (NEAL, 1996) quando mostra que algumas redes neurais convergem para um processo gaussiano conforme o número de nós de suas camadas tende a infinito. Além disso, eles estão relacionados com outros métodos importantes, como, por exemplo, métodos por máquina de vetores de suporte (ou SVM, do inglês *"support vector machine"*). Ainda mais, alguns desses métodos podem ser mais facilmente manipulados e interpretados quando trabalhados como processos gaussianos, como no caso das redes neurais.

Trabalhando dentro de um paradigma Bayesiano, que por si só já traz uma série de vantagens em termo de inferência estatística, como, por exemplo, para a seleção de modelos e para a regularização de parâmetros, temos como resultado um modelo probabilístico para os possíveis outputs de um novo input, que carrega muito mais informação do que um resultado "determinístico", como normalmente os algoritmos para aprendizado de máquina são apresentados. Entretanto, em alguns casos, a implementação computacional de tais métodos pode não ser tão simples e métodos para uma aproximação da distribuição posterior são necessários. Apresentamos alguns desses métodos, dando mais enfoque para o método de expectation propagation, onde discutimos em detalhes e apresentamos o algoritmo nested expectation propagation para classificação por processos gaussianos no caso onde temos mais de duas classes, desenvolvido por (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013), no capítulo 4. Outros métodos de aproximação são apresentados no capítulo 5.

Por fim, no capítulo 6, realizamos experimentos numéricos com os diferentes métodos apresentados por processos gaussianos e diferentes funções de covariância e parâmetros para avaliar e estudar o desempenho de tais métodos e a ação das diferentes escolhas possíveis.

Para a realização de tal trabalho, assumimos como já conhecidos os conceitos de análise matemática como apresentados em *"Curso de Análise, vol.1"* (LIMA, 2017) e *"Curso de Análise, vol.2"* (LIMA, 2015), conceitos de álgebra linear, como os apresentados em *"Linear Algebra and Its Applications"* (STRANG, 1988), conceitos de álgebra linear computacional, como os apresentados em *"Matrix Computations"* (GOLUB; LOAN, 2013) e conceitos de probabilidade, como os apresentados em *"Probability with martingales"* (WILLIAMS, 1991).

2 Paradigmas de Inferência Estatística

Nessa seção vamos abordar rapidamente e um pouco informalmente dois grandes paradigmas para inferência estatística: Maximum likelihood e inferência Bayesiana. Apesar da inferência Bayesiana ser mais abrangente e não sofrer de problemas que a maximum likelihood sofre, a inferência por maximum likelihood exibe resultados equivalentes sob certas condições (como, por exemplo, regularização) e provém uma outra interpretação, além de, algumas vezes, requerer menos poder computacional. Em algumas modelagens, ambos os paradigmas são aplicados em partes diferentes da mesma.

Vamos usar como exemplo o uso desses paradigmas para estimar parâmetros de um modelo paramétrico. Todavia, esses paradigmas são mais abrangentes e podem ser usados em diversos outros cenários, como seleção de modelos e de variáveis, entre outros. No caso da análise Bayesiana, fazemos uma breve discussão sobre uma abordagem para seleção de modelos neste paradigma na seção 2.2.3.

Historicamente, inferência por maximum likelihood era a ferramenta dominante na estatística da primeira metade do século XIX, e só pela década 60, com o acesso aos computadores, que tivemos poder o suficiente para métodos mais computacionalmente intensos serem atrativos, como métodos Bayesianos mais sofisticados. Cada um desses panoramas está relacionado com uma linha de pensamento estatístico para a interpretação da probabilidade. Maximum likelihood está mais ligado com um pensamento frequentista, que interpretava a probabilidade como uma *chance* ou *frequência* e que experimentos para coleta de novos dados podem ser repetidos quantas vezes necessários, assim sempre tendo uma quantidade suficiente de dados. A inferência Bayesiana por sua vez, como aponta o nome, está ligado a um pensamento Bayesiano, que interpreta a probabilidade como uma *medida de incerteza* que pode ser alterada, ou atualizada, sob a luz de novas informações.

2.1 Inferência por Maximum Likelihood

Tome $z_1, z_2, \ldots, z_n$ como observações independentes de uma variável aleatória Z. Para começar nossa modelagem, vamos supor que Z tem uma pdf/pmf dada por $p_Z(z)$. Definimos a *função likelihood* como

$$L_n(Z) = \prod_{i=1}^{n} p_Z(z_i).$$

É importante também definir a *função log-likelihood* como

$$\ell_n(Z) = \log L_n(Z) = \sum_{i=1}^{n} \log p_Z(z_i).$$

Para questões de formalismo matemático, assumiremos que L_n e ℓ_n estão bem definidas e são tão bem comportadas quanto queremos durante todo o resto dessa seção.

No caso onde Z é uma variável discreta (e, portanto, $p_Z(z) = \mathbb{P}(Z = z)$ é uma pmf), temos que $L_n(Z)$ é a probabilidade de observarmos o conjunto de dados em análise. Essa é uma interpretação intuitiva no caso de Z ser uma variável absolutamente contínua, já que formalmente a probabilidade de observarmos um ponto em específico é zero para qualquer ponto neste caso, i.e., $L_n(Z)$ não é formalmente a probabilidade de observarmos os dados neste caso.

Essa interpretação intuitiva ajuda a mostrar que o objetivo da inferência por maximum likelihood é achar uma pdf/pmf que maximize o valor de $\ell_n(Z)$. Os termos "maximum" e "maximizar" foram usados aqui de uma forma solta e ficarão mais claros ao longo dessa seção.

Para dar uma ideia da validade deste método e explorar mais suas propriedades, precisamos falar de uma ferramente importante para inferência estatística e teoria da informação: A divergência de Kullback-Leibler.

2.1.1 Divergência de Kullback-Leibler

Definição 1. Seja P e Q duas distribuições com pdf (ou pmf) dadas por p e q respectivamente. A *divergência de Kullback-Leibler* (para variáveis aleatórias absolutamente contínuas ou discretas) de Q para P é definida como

$$\begin{aligned} D_{KL}(P\|Q) &= \mathbb{E}_P\left[\log\left(\frac{p(x)}{q(x)}\right)\right] = \int \log\left(\frac{p(x)}{q(x)}\right) dP \\ &= \mathbb{E}_P[\log(p(x))] - \mathbb{E}_P[\log(q(x))] \end{aligned}$$

Propriedades. Algumas propriedades da divergência de Kullback-Leibler:

1. $D_{KL}(P\|Q) = 0 \iff P = Q$;
2. $D_{KL}(P\|Q) \geq 0$, $\forall P, Q$;
3. A divergência de Kullback-Leibler é invariante a transformações de coordenadas em x.

A demostração dessas propriedades pode ser vista em (KULLBACK, 1997, pg. 14).

Mesmo a divergência não sendo uma métrica, já que não é simétrica e não respeita a desigualdade triangular, uma interpretação de $D_{KL}(P\|Q)$ é ser a "distância" da distribuição Q para a distribuição P do ponto de vista de P. Na teoria da informação, essa divergência também é conhecida como *entropia relativa* e também tem a interpretação de quantificar a informação de P relativo a Q (um pouco mais desse ponto de vista é abordado no apêndice A.5).

Assuma agora que $Z \sim F_0$, i.e., que F_0 é a verdadeira distribuição de Z. Seja também F_p a distribuição provinda de uma pdf/pmf da forma $p_Z(z)$ como apresentado anteriormente. Desta forma,

$$D_{KL}(F_0\|F_p) = \mathbb{E}_{F_0}[\log(p_0(z))] - \mathbb{E}_{F_0}[\log(p_Z(z))].$$

Observe que $\mathbb{E}_{F_0}[\log(p_0(z))]$ é constante em relação a p e portanto minimizar a divergência do nosso modelo à distribuição verdadeira é equivalente a maximizar $\mathbb{E}_{F_0}[\log(p_Z(z))]$.

Temos então, pela lei forte dos grandes números, para $(z_i)_{i=1}^{\infty}$ uma sequência de observações independentes de Z, que

$$\begin{aligned}\mathbb{E}_{F_0}[\log(p_Z(z_i))] &= \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\log(p_Z(z))\ \mathbb{P}\text{-a.s.}\\ &= \lim_{n\to\infty}\frac{1}{n}\ell_n(Z)\ \mathbb{P}\text{-a.s..}\end{aligned}$$

Assim, assintoticamente, maximizar a função log-likelihood é minimizar a divergência, isto é, para um número suficientemente grande de pontos, maximizar a função log-likelihood é aproximar o nosso modelo da distribuição real.

2.1.2 Estimativa por Maximum Likelihood

Assuma agora que nosso modelo da distribuição de Z segue um modelo paramétrico, i.e., a pdf/pmf de Z é da forma $p_z(z) = p_{Z|\theta}(z|\theta)$ com $\theta \in \mathbb{R}^d$, i.e., a pdf/pmf de Z depende de d parâmetros. Esses parâmetros são inicialmente desconhecidos e nosso objetivo é estimá-los.

Neste caso, escrevemos a função likelihood como

$$L_n(\theta|Z) = \prod_{i=1}^{n} p_{Z|\theta}(z_i|\theta),$$

e a função log-likelihood como

$$\ell_n(\theta|Z) = \log L_n(\theta|Z) = \sum_{i=1}^{n}\log p_{Z|\theta}(z_i|\theta),$$

o que são vistas como uma função em θ dado um conjunto de dados e o modelo para $p_{Z|\theta}(z|\theta)$.

O método de estimativa por *maximum likelihood* consiste em estimar o valor dos parâmetros em θ que melhor explicam os dados maximizando a função de log-likelihood, isto é, assumindo que $\ell(\theta|Z)$ tem um único maximador global, temos que o estimado de θ por maximum likelihood é dado por

$$\hat{\theta}_{ML} = \arg\max_{\theta}\ell_n(\theta|Z).$$

O estimador de maximum likelihood possui algumas propriedades importantes. Uma delas é o fato de $\hat{\theta}_{ML}$ ser consistente, i.e., $\hat{\theta}_{ML} \to \theta^*$ em probabilidade quando $n \to \infty$, onde θ^* são os parâmetros que melhor aproximam nosso modelo da distribuição real, i.e., minimiza a divergência. Isso também mostra que $var(\hat{\theta}_{ML}) \to 0$ em probabilidade quando $n \to \infty$ e que $\hat{\theta}_{ML}$ é assintoticamente *unbiased*, i.e., $Bias(\hat{\theta}_{ML}) = \mathbb{E}[\hat{\theta}_{ML} - \theta^*] \to 0$ em probabilidade quando $n \to \infty$. Temos também que $\hat{\theta}_{ML}$ é equivariente em relação a transformações nos dados.

Outra propriedade é que, pela cota de Cramér-Rao (veja (COX, 2006, pg. 100, pg. 164)), o estimador por maximum likelihood tem aproximadamente a menor variância entre estimadores *unbiased* para valores grandes o suficientes de n.

Perceba que todas as explicações e propriedades apresentadas são dependentes de termos um número grande de dados disponível. Este é o principal problema do método de maximum likelihood: para valores pequenos de n o método tende a dar *overfitting* nos dados. Para corrigir isso, inclui-se um termo de regularização à função log-likelihood antes de maximizá-la. Técnicas desse tipo são conhecidas como *penalized maximum likelihood*.

2.1.3 Fisher Information

A função likelihood pode nos informar mais do que apenas uma estimativa pontual. Por enquanto, o método de maximum likelihood nos dá apenas um ponto, mas não nos informa nada sobre a distribuição deste (uma vez que, assumindo que calculamos ele a partir de observações de uma variável aleatória, o mesmo também é estocástico). Para isso, definimos a *observed Fisher information* como

$$\mathcal{J}_n(\theta) = -\frac{\partial^2 \ell_n}{\partial \theta^2}(\theta|Z).$$

Com isso, pode-se aproximar a distribuição de $\hat{\theta}_{ML}$ assintoticamente como

$$\hat{\theta}_{ML} \overset{a}{\sim} N(\theta^*, [\mathcal{J}_n(\hat{\theta}_{ML})]^{-1}).$$

Uma demostração pode ser encontrado em *principles of statistical inference* (COX, 2006, pg. 100).

Com isso, podemos, numa vizinhança de $\hat{\theta}_{ML}$, analisar sua distribuição. Um exemplo do que se pode ser feito é construir intervalos de confiança para o estimador. Observe mais uma vez a importância de um grande número de dados para esta estimativa.

2.1.4 Observações não-independentes

No caso mais mais geral onde $z_1, z_2, \ldots, z_n$ são observações não independentes de $Z_1, Z_2, \ldots, Z_n$, temos que a função likelihood é

$$L_n(Z) = p_{Z_1, Z_2, \ldots, Z_n}(z_1, z_2, \ldots, z_n),$$

onde o termo à direita da igualdade é a pdf/pmf conjunta de $Z_1, Z_2, \ldots, Z_n$. As definições para a função log-likelihood seguem da mesma forma.

Seja $\{p_n(x_1, \ldots, x_n|\theta)\}_{n=1}^{\infty}$ uma família de pdf/pmf dependente dos parâmetros θ com o intuito de modelar as distribuições conjuntas de $Z_1, Z_2, \ldots$, no sentido que p_n modela $p_{Z_1, Z_2, \ldots, Z_n}$. Neste cenário, temos $\ell_n(\theta|Z) = \log(L_n(\theta|Z)) = \log(p_n(z_1, \ldots, z_n|\theta))$ e $\hat{\theta}_{ML}$ é definido da mesma forma.

Essa forma mais generalizada da estimativa por maximum likelihood mantém todas as propriedades e resultados apresentados para o caso independente, como discutido em (COX, 2006).

2.1.5 Exemplo: Distribuição Normal

Suponha que temos um conjunto de observações unidimensionais $(x_i)_{i=1}^n$ que assumimos serem independentes e seguir uma distribuição normal $N(\mu, \sigma^2)$ com valor médio $\mu \in \mathbb{R}$ e variância $\sigma^2 > 0$ desconhecidos. Nosso objetivo aqui é achar o estimador por maximum likelihood para estes parâmetros.

Temos que a função log-likelihood para esse modelo é

$$\begin{aligned}\ell_n(\mu,\sigma^2|X) &= \sum_{i=1}^{n}\log(\phi(x_i|\mu,\sigma^2)) = \sum_{i=1}^{n}\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}\right)\\ &= \sum_{i=1}^{n}-\frac{1}{2}\log(2\pi)-\frac{1}{2}\log(\sigma^2)-\frac{1}{2\sigma^2}(x_i-\mu)^2\\ &= -\frac{n}{2}\log(2\pi)-\frac{n}{2}\log(\sigma^2)-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\end{aligned}$$

Para acharmos os estimadores por maximum likelihood, temos que maximizar essa função $\ell_n(\mu,\sigma^2|X)$. Para isso, vamos investigar os seus pontos críticos. Observe que

$$\frac{\partial\ell_n}{\partial\mu}(\mu,\sigma^2|X) = \frac{1}{2\sigma^2}\sum_{i=1}^{n}2(x_i-\mu) = \frac{1}{\sigma^2}\sum_{i=1}^{n}x_i-\frac{n}{\sigma^2}\mu$$

e

$$\frac{\partial\ell_n}{\partial\sigma^2}(\mu,\sigma^2|X) = -\frac{n}{2\sigma^2}+\frac{1}{2(\sigma^2)^2}\sum_{i=1}^{n}(x_i-\mu)^2,$$

Portanto,

$$\frac{\partial\ell_n}{\partial\mu}(\mu,\sigma^2|X)=0 \Rightarrow \mu = \frac{1}{n}\sum_{i=1}^{n}x_i = \overline{X}$$

e

$$\frac{\partial\ell_n}{\partial\sigma^2}(\mu,\sigma^2|X)=0 \Rightarrow \sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i-\mu)^2,$$

Assim,

$$\nabla\ell_n(\mu,\sigma^2|X)=0 \Rightarrow \hat{\mu}_{ML}=\overline{X};\ \widehat{\sigma^2}_{ML} = \frac{1}{n}\sum_{i=1}^{n}(x_i-\overline{X})^2.$$

Observe também que

$$-\nabla^2\ell_n(\mu,\sigma^2|X) = \begin{bmatrix}\frac{n}{\sigma^2} & \frac{n}{(\sigma^2)^2}(\overline{X}-\mu)\\ \frac{n}{(\sigma^2)^2}(\overline{X}-\mu) & -\frac{n}{2(\sigma^2)^2}+\frac{1}{(\sigma^2)^3}\sum_{i=1}^{n}(x_i-\mu)^2\end{bmatrix}$$

E portanto temos que a *observed Fisher information* para nosso candidato a estimador é

$$\mathcal{J}_n(\hat{\mu}_{ML},\widehat{\sigma^2}_{ML}) = \begin{bmatrix}\frac{n}{\widehat{\sigma^2}_{ML}} & 0\\ 0 & \frac{n}{2(\widehat{\sigma^2}_{ML})^2}\end{bmatrix},$$

o que é uma matriz definida positiva (já que é diagonal com entradas positivas), mostrando que

$$\hat{\mu}_{ML} = \frac{1}{n}\sum_{i=1}^{n}x_i = \overline{X};$$

$$\widehat{\sigma^2}_{ML} = \frac{1}{n}\sum_{i=1}^{n}(x_i-\overline{X})^2;$$

são, de fato, os estimadores por maximum likelihood para o valor esperado e variância para uma distribuição normal.

2.2 Inferência Bayesiana

De um ponto de vista Bayesiano, uma distribuição de probabilidade representa a nossa incerteza perante os possíveis acontecimentos. Desta forma, se inicialmente acreditávamos que um evento segue uma distribuição $\mathbb{P}_Y(y)$, podemos reajustar nossa crença a luz de novas observações $\mathcal{D}$ relacionadas a esse evento. Esse reajuste se dá pela fórmula de Bayes:

$$\mathbb{P}_{Y|\mathcal{D}}(y|\mathcal{D}) = \frac{\mathbb{P}_{\mathcal{D}|Y}(\mathcal{D}|y)\mathbb{P}_Y(y)}{\mathbb{P}_{\mathcal{D}}(\mathcal{D})}.$$

Essa é a base para inferência Bayesiana. Primeiro, assumimos uma distribuição *a priori* $\mathbb{P}_Y(y)$ (normalmente chamada de *prior* na literatura) do evento em análise, e a *likelihood* $\mathbb{P}_{\mathcal{D}|Y}(\mathcal{D}|y)$, que representa a nossa incerteza de observar os dados $\mathcal{D}$ dado o evento y. Com isso, temos a *marginal likelihood* $\mathbb{P}_{\mathcal{D}}(\mathcal{D}) = \mathbb{E}_{\mathbb{P}_Y}[\mathbb{P}_{\mathcal{D}|Y}(\mathcal{D}|y)] = \displaystyle\int \mathbb{P}_{\mathcal{D}|Y}(\mathcal{D}|y)\, d\mathbb{P}_Y$ e podemos calcular a distribuição *a posteriori* $\mathbb{P}_{Y|\mathcal{D}}(y|\mathcal{D})$ (normalmente chamada de *posterior*). Esse passo é conhecido como *atualização Bayesiana ou update Bayesiano* e podemos representá-lo pelo diagrama

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Marginal likelihood}}.$$

Observe que esse panorama é mais abrangente do que o apresentado anteriormente, uma vez que não assumimos nada sobre nossas escolhas de prior e likelihood e nem independência nos dados. A escolha de um prior não nos limita e vamos comentar mais tarde o porquê e abordar quais vantagens ele nos traz. Paro o caso onde temos uma pdf ou pmf para cada uma desses distribuições, fazemos a atualização diretamente com estas:

$$p_{Y|\mathcal{D}}(y|\mathcal{D}) = \frac{p_{\mathcal{D}|Y}(\mathcal{D}|y)p_Y(y)}{p_{\mathcal{D}}(\mathcal{D})}.$$

A atualização Bayesiana segue o *princípio de mínima informação (ou princípio de máxima entropia)*, isto é, dada uma distribuição prior P_0, a distribuição P apropriada para representar o novo estado de informação minimiza $D_{KL}(P\|P_0)$ respeitando as novas evidências apresentadas. A interpretação desse princípio é que a nova distribuição é o mais parecido possível com a distribuição inicial levando em consideração os novos dados apresentados, assim introduzindo apenas as informações que não podem ser descartadas sob a luz das novas informações. Uma análise um pouco mais aprofundada sobre esse princípio pode ser encontrada em *Bayesian conditionalisation and the principle of minimum information* (WILLIAMS, 1980).

2.2.1 Estimadores Bayesianos

Voltamos agora para o caso onde temos $z_1, z_2, \ldots, z_n$ observações de uma variável aleatória Z que assumimos seguir um modelo com parâmetros $\theta \in \mathbb{R}^d$ para sua pdf/pmf $p(z|\theta)$. Definimos também uma distribuição prior $p(\theta)$ para os parâmetros θ. Com isso, estimamos a distribuição posterior

$$p(\theta|Z) = \frac{p(z_1, \ldots, z_n|\theta)p(\theta)}{\displaystyle\int p(z_1, \ldots, z_n|\theta)p(\theta)\, d\theta}.$$

A distribuição posterior inteira é nosso estimado, e podemos usar ela de diversas maneiras para escolher um estimador pontual para θ.

Uma estratégia para escolher um estimador pontual é definir inicialmente uma função de perda $\mathcal{L}(\theta, \hat{\theta})$ que mede o quanto se perde se escolhemos $\hat{\theta}$ no caso de θ ser o valor verdadeiro. Definimos então a função risco como $R(\hat{\theta}) = \mathbb{E}_{\theta|Z}[\mathcal{L}(\theta, \hat{\theta})]$ e escolhemos nosso estimador de forma a minimizar a função risco. Chamamos de *estimador Bayesiano* qualquer estimador que possa ser formulado dessa forma.

Por exemplo, Uma função de perda comum é um erro quadrático $\mathcal{L}(\theta, \hat{\theta}) = \|\hat{\theta} - \theta\|^2$, o que nos dá como função risco conhecida como *mean squared error*:

$$\begin{aligned} \mathrm{MSE}(\hat{\theta}) &= \mathbb{E}_{\theta|Z}[\|\hat{\theta} - \theta\|^2] = \mathbb{E}_{\theta|Z}[\|(\hat{\theta} - \mathbb{E}_{\theta|Z}[\theta]) - (\theta - \mathbb{E}_{\theta|Z}[\theta])\|^2] \\ &= \mathbb{E}_{\theta|Z}[\|\theta - \mathbb{E}_{\theta|Z}[\theta]\|^2] + \|\mathbb{E}_{\theta|Z}[\hat{\theta} - \theta]\|^2 \\ &= \mathrm{tr}(var(\theta|Z)) + \|\hat{\theta} - \mathbb{E}_{\theta|Z}[\theta]\|^2 \end{aligned}$$

Que é minimizada pelo valor esperado. Portanto, temos que o estimador por *minimum mean squared error* é:

$$\hat{\theta}_{posterior} = \mathbb{E}_{\theta|Z}[\theta] = \int \theta p(\theta|Z)\, d\theta. \tag{2.1}$$

Aqui podemos enfrentar um dos principais problemas quando usando métodos Bayesianos: Expectativas e integrais analiticamente indisponíveis. Isso pode acontecer já com a marginal likelihood $p(Z) = \int p(Z|\theta)p(\theta)\, d\theta$, o que faria a distribuição posterior analiticamente indisponível, e consequentemente $\int \theta p(\theta|Z)\, d\theta$ também. Para isso, existem, por exemplo, os chamados *prior conjugados* de uma likelihood, que proporcionam que a distribuição posteior seja da mesma família que o prior, apenas alterando os parâmetros, mantendo assim a atualização analiticamente computável. Nos casos onde isso não acontece, precisamos recorrer a técnicas para integração numérica ou métodos de Monte Carlo. Atualmente, o uso de técnicas sofisticadas de *Markov chain Monte Carlo* (MCMC) tem se mostrado essencial para o funcionamento e desempenho de métodos por inferência Bayesiana.

2.2.2 Estimador MAP e ligação com o método de maximum likelihood

Ainda no cenário de estimação de parâmetros, outro estimador que podemos escolher é conhecido por estimador *maximum a posterior* (MAP), que nada mais é do que uma moda da distribuição posterior, i.e., um maximizador da pdf/pmf posterior. Intuitivamente, estamos escolhendo o parâmetro com maior "chance" de ser. Nesse caso,

$$\hat{\theta}_{MAP} \in \arg\max_{\theta} \{p(\theta|Z)\}.$$

Observe que

$$\log p(\theta|Z) = \log(p(Z|\theta)) + \log(p(\theta)) - \log\left(\int p(Z|\theta)p(\theta)\, d\theta\right) = \ell_n(\theta|Z) + \lambda(\theta|Z),$$

o que mostra que o estimador MAP é equivalente a um estimador por penalised maximum likelihood com uma função de penalização λ, e que no caso onde $p(\theta)$ é constante (e portanto

nosso prior é uniforme) temos o estimador por maximum likelihood de volta. Isso mostra que podemos ver a estimativa por maximum likelihood como um caso especial desse estimador. Geralmente, estimadores MAP não são estimadores Bayesianos (isto é, não minimizam uma função de risco como definida anteriormente), sendo assim, eles não representa nenhum papel de destaque numa análise Bayesiana.

2.2.3 Fator de Bayes

Suponha que temos modelos $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_n$ para a distribuição dos dados de um dataset D e queremos eleger um desses como o melhor modelo que descreve os dados. Para isso, definimos uma distribuição prior $\mathbb{P}(\mathcal{M}_i)$ sobre os modelos. Com isso, temos, pela atualização Bayesiana,

$$\frac{\mathbb{P}(\mathcal{M}_i|D)}{\mathbb{P}(\mathcal{M}_j|D)} = \frac{\frac{\mathbb{P}(D|\mathcal{M}_i)\mathbb{P}(\mathcal{M}_i)}{\mathbb{P}(D)}}{\frac{\mathbb{P}(D|\mathcal{M}_j)\mathbb{P}(\mathcal{M}_j)}{\mathbb{P}(D)}} = \frac{\mathbb{P}(D|\mathcal{M}_i)}{\mathbb{P}(D|\mathcal{M}_j)}\frac{\mathbb{P}(\mathcal{M}_i)}{\mathbb{P}(\mathcal{M}_j)},$$

isto é, a razão entre as probabilidades posterior de dois modelos é proporcional a razão entre suas probabilidades prior, e o fator de proporcionalidade é dado pela razão entre as marginal likelihoods de D em cada um dos modelos. Definimos o *fator de Bayes* entre os modelos $\mathcal{M}_i$ e $\mathcal{M}_j$ como esse fator, isto é,

$$\mathcal{B}(\mathcal{M}_i, \mathcal{M}_j) = \frac{\mathbb{P}(D|\mathcal{M}_i)}{\mathbb{P}(D|\mathcal{M}_j)}.$$

Observe que $\mathcal{B}(\mathcal{M}_i, \mathcal{M}_j) > 1$ pode ter a interpretação que os dados em D dão mais suporte para a hipótese que o modelo $\mathcal{M}_i$ descreve melhor a distribuição destes do que para a hipótese que $\mathcal{M}_j$ descreve melhor, e o oposto para quando $\mathcal{B}(\mathcal{M}_i, \mathcal{M}_j) < 1$. Desta forma estamos quantificando o quanto o dataset D favorece escolhermos um modelo em contraste a outro. Esse argumento pode ser usado para justificar escolher o modelo $\mathcal{M}_i$ que recebe o maior "suporte" de D em relação a cada outro modelo individualmente, isto é, o modelo com a maior marginal likelihood $\mathbb{P}(D|\mathcal{M}_i)$.

É interessante notar que no cenário onde escolhemos uma distribuição prior $\mathbb{P}(\mathcal{M}_i)$ uniforme entre os modelos considerados (como no caso onde não queremos favorecer nenhum modelo em relação aos outros), temos que o fator de Bayes é igual à razão entre as probabilidades posterior entre os modelos, o que nos dá ainda mais suporte para a escolha do modelo $\mathcal{M}_i$.

2.2.4 Exemplo: Likelihood Normal

Suponha que temos um conjunto de observações unidimensionais e independentes $(x_i)_{i=1}^n$ que assumimos seguir uma distribuição normal $N(\mu, \sigma^2)$ com valor médio $\mu \in \mathbb{R}$ e variância $\sigma^2 > 0$. Faremos três análises sobre esse caso: A primeira assumiremos que o valor de μ é desconhecido, mas o valor de σ^2 é conhecido; A segunda assumiremos que o valor de σ^2 é desconhecido, mas o valor de μ é conhecido; A terceira assumiremos que ambos os valores são desconhecidos e usaremos os dois casos anteriores para construir um modelo. Observe como a escolha de um prior faz parte do modelo e os paralelos que podemos traçar com os estimadores encontrados aqui e os estimadores por maximum likelihood.

Assumimos agora que o valor de μ é desconhecido, mas o valor de σ^2 é conhecido. Tomamos como prior $\mu \sim N\left(\hat{\mu}_{prior}, \frac{\sigma^2}{m}\right)$ com $\hat{\mu}_{prior} \in \mathbb{R}$ e $m > 0$ real fixo. Chamamos esse modelo de modelo Normal-Normal para deixar claro que atribuímos uma likelihood normal e um prior conjugado normal para o valor médio.

Observe que, pela atualização Bayesiana, sendo $\phi(x|\mu, \sigma^2)$ a densidade da distribuição normal definida no apêndice A.1, temos que

$$\begin{aligned}
p(\mu|X) &\propto p(X|\mu)p(\mu) \\
&\propto \left(\prod_{i=1}^{n} \phi(x_i|\mu, \sigma^2)\right) \cdot \phi\left(\mu \,\middle|\, \hat{\mu}_{prior}, \frac{\sigma^2}{m}\right) \\
&\propto \exp\left(-\frac{(\mu - \hat{\mu}_{prior})^2}{2\frac{\sigma^2}{m}} - \sum_{i=1}^{n} \frac{(x_i - \mu)^2}{2\sigma^2}\right) \\
&\propto \exp\left(-\frac{(m+n)\mu^2 - 2(n\overline{X} + m\hat{\mu}_{prior})\mu + m\hat{\mu}_{prior}^2 + \sum_{i=1}^{n} x_i^2}{2\sigma^2}\right) \\
&\propto \exp\left(-\frac{\left(\mu - \left(\frac{n}{m+n}\overline{X} + \frac{m}{m+n}\hat{\mu}_{prior}\right)\right)^2}{2\frac{\sigma^2}{m+n}}\right).
\end{aligned}$$

Definindo $\lambda = \dfrac{m}{m+n}$, temos que a distribuição posterior é

$$\mu|X \sim N\left(\lambda\hat{\mu}_{prior} + (1-\lambda)\hat{\mu}_{ML}, \frac{\sigma^2}{m+n}\right).$$

Veja que aqui não precisamos calcular a constante de normalização pois ela é unicamente definida pela função proporcional a pdf posterior. Perceba também que aqui $\hat{\mu}_{posterior}$ é uma combinação convexa do nosso prior com o estimador por maximum likelihood, e que quando $n \to \infty$, temos que $\lambda \to 0$ e recuperamos $\hat{\mu}_{ML}$ com todas as suas propriedades assintóticas.

No caso de valor médio conhecido e variância desconhecida, temos o modelo Inverse-gamma-Normal que assume uma likelihood normal $N(\mu, \sigma^2)$ e um prior conjugado inverse-gamma $\sigma^2 \sim IG(\frac{m}{2} + 1, \frac{m}{2}\widehat{\sigma^2}_{prior})$, para $m > 0$ e $\widehat{\sigma^2}_{prior} > 0$.

Uma distribuição inverse-gamma $IG(\alpha, \beta)$, $\alpha > 0$ e $\beta > 0$, tem como pdf

$$p_{IG}(x|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} e^{-\frac{\beta}{x}},$$

onde $\Gamma(z) = \displaystyle\int_0^\infty x^{z-1}e^{-x}\,dx$ é a função gamma. Temos também que o valor esperado para a distribuição inverse-gamma é $\dfrac{\beta}{\alpha - 1}$ se $\alpha > 1$.

Dessa forma, temos pela atualização Bayesiana,

$$\begin{aligned}
p(\sigma^2|X) &\propto p(x|\mu, \sigma^2)p(\sigma^2) \\
&\propto \left(\prod_{i=1}^{n} \phi(x_i|\mu, \sigma^2)\right) \cdot p_{IG}\left(\sigma^2 \,\middle|\, \frac{m}{2} + 1, \frac{m}{2}\widehat{\sigma^2}_{prior}\right) \\
&\propto (\sigma^2)^{-\frac{n}{2}} (\sigma^2)^{-(\frac{m}{2}+1)-1} \exp\left(-\frac{\frac{m}{2}\widehat{\sigma^2}_{prior}}{\sigma^2} - \sum_{i=1}^{n} \frac{(x_i - \mu)^2}{2\sigma^2}\right) \\
&\propto (\sigma^2)^{-(\frac{m+n}{2}+1)-1} \exp\left(-\frac{\frac{m}{2}\widehat{\sigma^2}_{prior} + \frac{n}{2}\widehat{\sigma^2}_{ML}}{\sigma^2}\right)
\end{aligned}$$

Onde $\widehat{\sigma^2}_{ML} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2$. O que nos dá como distribuição posterior

$$\sigma^2|X \sim IG\left(\frac{m+n}{2} + 1, \frac{m}{2}\widehat{\sigma^2}_{prior} + \frac{n}{2}\widehat{\sigma^2}_{ML}\right).$$

Definindo $\lambda = \frac{m}{m+n}$, temos novamente, calculando o valor esperado para nossa distribuição posterior,

$$\widehat{\sigma^2}_{posterior} = \lambda\widehat{\sigma^2}_{prior} + (1-\lambda)\widehat{\sigma^2}_{ML}.$$

Os mesmos comentários feitos para o estimador do caso anterior podem ser feitos aqui.

Agora, para o caso onde tanto o valor médio quanto a variância são desconhecidos, podemos definir que $\mu|(m,\sigma^2) \sim N\left(\hat{\mu}_{prior}, \frac{\sigma^2}{m}\right)$ e $\sigma^2 \sim IG(\frac{s}{2}+1, \frac{s}{2}\widehat{\sigma^2}_{prior})$ para $\hat{\mu}_{prior} \in \mathbb{R}$ e $\widehat{\sigma^2}_{prior}, m, s > 0$, o que nos dá como prior de (μ, σ^2) uma distribuição Normal-inverse-gamma

$$NIG\left(\hat{\mu}_{prior}, m, \frac{s}{2}+1, \frac{s}{2}\widehat{\sigma^2}_{prior}\right),$$

que tem densidade

$$p_{NIG}(\mu,\sigma^2) = \phi\left(\mu\left|\hat{\mu}_{prior}, \frac{\sigma^2}{m}\right.\right) p_{IG}\left(\sigma^2\left|\frac{s}{2}+1, \frac{s}{2}\widehat{\sigma^2}_{prior}\right.\right).$$

E portanto vamos ter como distribuição posterior

$$(\mu,\sigma^2)|X \sim NIG\left(\hat{\mu}_{posterior}, m+n, \frac{s+n}{2}+1, \frac{s+n}{2}\widehat{\sigma^2}_{prior} + \frac{n}{2}\widehat{\sigma^2}_{ML} + \frac{mn}{m+n}\frac{(\hat{\mu}_{ML}-\hat{\mu}_{prior})^2}{2}\right),$$

onde $\widehat{\sigma^2}_{ML} = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \overline{X}\right)^2$. Assim,

$$\hat{\mu}_{posterior} = \frac{m}{n+m}\hat{\mu}_{prior} + \left(1 - \frac{m}{n+m}\right)\hat{\mu}_{ML};$$

$$\widehat{\sigma^2}_{posterior} = \frac{s}{n+s}\widehat{\sigma^2}_{prior} + \left(1-\frac{s}{n+s}\right)\widehat{\sigma^2}_{ML} + \left(1-\frac{s}{n+s}\right)\frac{m}{m+n}(\hat{\mu}_{ML}-\hat{\mu}_{prior})^2.$$

Apesar de $\hat{\mu}_{posterior}$ ter se mantido o mesmo, $\widehat{\sigma^2}_{posterior}$ agora tem um termo extra que se parece com alguma forma de resíduo entre $\hat{\mu}_{prior}$ e o estimador por maximum likelihood. Entretanto, observe que ainda temos que quando $n \to \infty$ temos $(\hat{\mu}_{posterior}, \widehat{\sigma^2}_{posterior}) \to (\hat{\mu}_{ML}, \widehat{\sigma^2}_{ML})$ e todas as propriedades assintóticas.

3 Processos Gaussianos

3.1 Definição

Nessa seção apresentaremos as definições e os fundamentos para processos gaussianos e seus usos em problemas de regressão e classificação. Uma das grandes referências na área é *Gaussian processes for machine learning* (WILLIAMS; RASMUSSEN, 2006), o qual trouxe o uso de tais técnicas para o cenário convencional do aprendizado de máquina como um método viável e eficiente, e formando uma base para os estudos seguintes sobre o tema.

Vamos começar com uma definição central:

Definição 2. Um processo gaussiano (unidimensional) é uma família de variáveis aleatórias $\{f(x)\}_{x\in\mathcal{X}}$, indexada pelo conjunto $\mathcal{X}$, tal que qualquer coleção finita de tais tem uma distribuição conjunta gaussiana.

Como uma distribuição gaussiana é unicamente definida pelo seu valor esperado e variância, um processo gaussiano é definido especificando todos os valores esperados e variâncias, i.e., a *função de valor médio* $m : \mathcal{X} \to \mathbb{R}$ e a *função de covariância* $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ definidas a seguir caracterizam completamente a distribuição de tais processos:

$$\begin{aligned} m(x) &= \mathbb{E}[f(x)]; \\ k(x,x') &= cov(f(x), f(x')) = \mathbb{E}[(f(x) - m(x))(f(x') - m(x'))]. \end{aligned}$$

Portanto, podemos representar um processo gaussiano pela notação

$$f \sim \mathcal{GP}(m, k).$$

A existência de tais processos é garantida pelo teorema da extensão de Kolmogorov (ver (TAO, 2011, pg. 235)), o qual a propriedade de marginalização das distribuições gaussianas (ver identidade (A.2) do apêndice A.1) é suficiente para satisfazer as condições do teorema.

Mesmo que processos estocásticos sejam comumente indexados por "tempo" (i.e., por {0, 1, 2, 3, . . . } ou $[0, \infty)$), isso não é o caso aqui. Na maioria das aplicações em aprendizado de máquinas, o nosso conjunto de índices será o mesmo que o espaço de inputs no nosso dataset (geralmente $\mathbb{R}^D$). Com essa notação, pode-se interpretar um processo estocástico como uma função aleatória de $\mathcal{X}$ para $\mathbb{R}$. Essa é uma ideia central para o uso de tais objetos em nossa modelagem.

É importante observar que a função de covariância k, diferente de m, não pode ser qualquer. k precisa ser simétrica positiva semidefinida, onde a simetria é definida como $k(x, y) = k(y, x)$, $\forall x, y \in \mathcal{X}$, e ser positiva semidefinida é definido como, para qualquer coleção finita de pontos X de $\mathcal{X}$, a matriz $K(X, X)$ (definida em (3.2)) ser positiva semidefinida. Discutimos mais sobre as funções de covariância na seção 3.4.

Um exemplo comum e muito usado de k, conhecida como *exponencial quadrática (squared exponential ou SE)*, pode ser definida como

$$k_{SE}(x, y|\sigma^2, l_1, \ldots, l_D) = \sigma^2 \exp\left(-\frac{1}{2}\sum_{i=1}^{D} l_i^{-2}(x^i - y^i)^2\right), \tag{3.1}$$

onde $\sigma^2, l_1, \ldots, l_D$ são chamados de *hiperparâmetros* da função de covariância, neste caso tendo a restrição de serem reais e positivos, e x^i, y^i representam a i-ésima entrada de x e y respectivamente.

Para um processo gaussiano $f \sim \mathcal{GP}(m, k)$ e $X = (x_i)_{i=1}^n$ e $Y = (y_i)_{i=1}^m$ coleções finitas de pontos de $\mathcal{X}$, definimos

$$K(X, Y) = \begin{bmatrix} k(x_1, y_1) & k(x_1, y_2) & \cdots & k(x_1, y_m) \\ k(x_2, y_1) & k(x_2, y_2) & \cdots & k(x_2, y_m) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_n, y_1) & k(x_n, y_2) & \cdots & k(x_n, y_m) \end{bmatrix} \tag{3.2}$$

como a matriz de covariância entre $f(X) = [f(x_i)]_{i=1}^n$ e $f(Y) = [f(y_i)]_{i=1}^m$. Definimos também $m(X)$ e $m(Y)$ de maneira similar. Teríamos então, por definição, que

$$\begin{bmatrix} f(X) \\ f(Y) \end{bmatrix} \sim N\left(\begin{bmatrix} m(X) \\ m(Y) \end{bmatrix}, \begin{bmatrix} K(X, X) & K(X, Y) \\ K(Y, X) & K(Y, Y) \end{bmatrix}\right).$$

Entretanto, em muitas ocasiões, apenas uma dimensão não é o suficiente para nossos objetivos, o que nos leva a expandir a nossa definição.

Definição 3. Um processo gaussiano C-dimensional é uma família de vetores aleatórios com C entradas $\{f(x)\}_{x\in\mathcal{X}}$, indexada pelo conjunto $\mathcal{X}$, tal que qualquer coleção finita de tais tem uma distribuição conjunta gaussiana.

Essa definição nos dá que $f(x)$, para qualquer $x \in \mathcal{X}$, tem uma distribuição normal C-dimensional, o que implica em cada entrada $f_i(x)$ seguir uma distribuição normal unidimensional, isto é, cada $f_1, f_2, \ldots, f_C$ ser um processo gaussiano (unidimensional) distinto.

Com base nisso e no intuito de caracterizar melhor esses processos, considere que $f_1 \sim \mathcal{GP}(m_1, k_{1,1})$, $f_2 \sim \mathcal{GP}(m_2, k_{2,2})$, …, $f_C \sim \mathcal{GP}(m_C, k_{C,C})$ e que $X = (x_i)_{i=1}^n$, $Y = (y_i)_{i=1}^m$ são

coleções finitas de pontos de $\mathcal{X}$. Definimos

$$f(X) = \begin{bmatrix} f_1(X) \\ f_2(X) \\ \vdots \\ f_C(X) \end{bmatrix} = \begin{bmatrix} f_1(x_1) \\ f_1(x_2) \\ \vdots \\ f_1(x_n) \\ f_2(x_1) \\ \vdots \\ f_2(x_n) \\ \vdots \\ f_C(x_1) \\ \vdots \\ f_C(x_n) \end{bmatrix}. \tag{3.3}$$

Definimos também $m^C : \mathcal{X} \to \mathbb{R}^C$ por

$$m^C(x_i) = \begin{bmatrix} m_1(x_i) \\ m_2(x_i) \\ \vdots \\ m_C(x_i) \end{bmatrix}$$

e $m^C(X)$ de maneira similar a $f(X)$. Temos também $k^C : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{C \times C}$ definido por

$$k^C(x_i, y_j) = \begin{bmatrix} k_{1,1}(x_i, y_j) & k_{1,2}(x_i, y_j) & \cdots & k_{1,C}(x_i, y_j) \\ k_{2,1}(x_i, y_j) & k_{2,2}(x_i, y_j) & \cdots & k_{2,C}(x_i, y_j) \\ \vdots & \vdots & \ddots & \vdots \\ k_{C,1}(x_i, y_j) & k_{C,2}x_i, y_j) & \cdots & k_{C,C}(x_i, y_j) \end{bmatrix},$$

e

$$K^C(X, Y) = \begin{bmatrix} K_{1,1}(X, Y) & K_{1,2}(X, Y) & \cdots & K_{1,C}(X, Y) \\ K_{2,1}(X, Y) & K_{2,2}(X, Y) & \cdots & K_{2,C}(X, Y) \\ \vdots & \vdots & \ddots & \vdots \\ K_{C,1}(X, Y) & k_{C,2}(X, Y) & \cdots & K_{C,C}(X, Y) \end{bmatrix}$$

onde $k_{a,b}(x_i, y_j) = cov(f_a(x_i), f_b(y_j))$ e $K_{a,b}(X, Y)$ segue a definição (3.2) com $k_{a,b}$ no lugar de k. Dessa forma, temos que

$$k^C(x, y) = cov(f(x), f(y)) = \mathbb{E}[(f(x) - m^C(x))(f(y) - m^C(y))^T].$$

Observe novamente que k^C tem que ser simétrica positiva semidefinida, onde simetria é definido como $k^C(x, y) = [k^C(y, x)]^T$, $\forall x, y \in \mathcal{X}$, e ser positiva semidefinida é definido como, para qualquer coleção finita de pontos Z de $\mathcal{X}$, a matriz $K^C(Z, Z)$ é positiva semidefinida.

Com isso, usamos a notação

$$f \sim \mathcal{GP}_C(m^C, k^C)$$

para denotar um processo gaussiano C-dimensional. Neste cenário, temos que, para uma coleção finita genérica X de pontos,

$$f(X) \sim N(m^C(X), K^C(X, X)). \tag{3.4}$$

Escolher uma função de valor médio m^C e, principalmente, as funções $k_{a,b}$ para $a \neq b$ tal que k^C seja positiva semidefinida pode não ser intuitivo. Nos usos que daremos para esse processos, não há motivos a priori que nos impeçam ou que apresente uma desvantagem em assumirmos que as funções componentes de f são independentes entre si, isto é, $k_{a,b}$ é a função nula se $a \neq b$. Sob essa hipótese, podemos representar k^C apenas como uma lista de funções de covariância $[k_1, k_2, \ldots, k_C]$, as funções de covariância de cada função componente de f. Para o problema de classificação, cabe ainda poder tomar a priori m^c como a função nula. Para não sobrecarregar a notação, denotaremos m^C, k^C e $K^C(X, X)$ apenas por m, k e $K(X, X)$ daqui em diante, já que nossas definições são consistentes para o caso unidimensional e para o caso mais geral.

3.2 Fundamentos para Regressão por Processos Gaussianos

Considere, para essa seção, que $(x_i, y_i)_{i=1}^n$ é um dataset com espaço de inputs $\mathcal{X}$ qualquer e espaço de outputs $\mathbb{R}^C$ e que não existem inputs repetidos. Interpretaremos esses pares como resultados de alguma relação entre os dois espaços. Estimar tal relação é nosso objetivo aqui. Tome também X como a coleção de pontos com os inputs, Y como o vetor com os outputs (rearranjando as entradas para seguir o mesmo estilo de indexação de (3.3)) e X^* como uma coleção finita qualquer de pontos de $\mathcal{X}$ na qual pretendemos estimar os valores relacionados Y^*.

Começamos nossa modelagem assumindo que cada output y_i pode ser visto como uma observação com ruído de uma função $f : \mathcal{X} \to \mathbb{R}^C$ aplicada em x_i. Seguindo uma abordagem Bayesiana para estimar tal função, precisamos definir um prior para a distribuição dos possíveis valores de $f(x)$ e uma likelihood $p(y(x)|f(x))$, que modela a distribuição desse ruído, para cada x no espaço de inputs. Observe que aqui $\mathcal{X}$ é um conjunto qualquer, e utilizando $f \sim \mathcal{GP}_C(m, k)$ como prior para f, com funções m e k apropriadas, trazemos nosso problema de um conjunto abstrato para um espaço euclidiano.

Escolhemos também uma likelihood gaussiana (possivelmente degenerada) $y(x)|f(x) = y|f, x \sim N(f(x), \Sigma)$, onde Σ é uma matriz semidefinida positiva e conhecida. Essas escolhas são equivalentes a assumir um modelo

$$y(x) = f(x) + \epsilon_x,$$

onde ϵ_x são identicamente distribuídos por $N(0, \Sigma)$ e são independente de f. Assumiremos também que ϵ_x são independentes entre si.

Encaixando o dataset neste modelo, temos $Y = f(X) + \epsilon_X$, onde ϵ_X é o vetor com os erros arranjados de forma correspondente a Y. Observe que, pela maneira que arrumamos os erros e

pela independência entre ele, temos

$$\epsilon_X \sim N(0, \Sigma \otimes I),$$

onde I é a matriz identidade (neste caso, $n \times n$) e $\otimes$ é o produto de Kronecker. Lembrando que

$$f(X) \sim N(m(X), K(X, X)),$$

temos, por causa da independência entre ϵ_X e f, que a marginal likelihood é dada por

$$Y|X \sim N(m(X), K(X, X) + (\Sigma \otimes I)). \tag{3.5}$$

Como discutido na seção 2.2.3, podemos utilizar a marginal likelihood (calculada aqui como a likelihood dos outputs seguindo essa distribuição (3.5)) para definir possíveis hiperparâmetros do nosso modelo.

Aproveitando a estrutura gaussiana da marginal likelihood e de $f(X^*)$, podemos achar a distribuição de $f(X^*)$ condicionado por $Y|X$ usando a fórmula apresentada no apêndice (A.3), observando que, pela independência entre f e ϵ_X, temos

$$cov(Y, f(X^*)) = cov(f(X), f(X^*)) = K(X, X^*).$$

Assim,

$$\begin{aligned} f(X^*)|Y, X \quad \sim \quad & N(m(X^*) + K(X^*, X)[K(X, X) + (\Sigma \otimes I)]^{-1}(Y - m(X)), \\ & K(X^*, X^*) - K(X^*, X)[K(X, X) + (\Sigma \otimes I)]^{-1}K(X, X^*)). \end{aligned} \tag{3.6}$$

Essa é uma equação central para regressão por processos gaussianos, já que é a distribuição posterior para $f(X^*)$. Com ela, podemos calcular a distribuição e estimadores para $Y^* = f(X^*) + \epsilon_{X^*}$, onde ϵ_{X^*} é o vetor de erros correspondente . No caso do estimador por *minimum mean squared error* (2.1), apresentado na seção anterior, temos

$$\hat{Y}^*_{posterior} = \mathbb{E}_{Y^*|Y,X}[Y^*] = m(X^*) + K(X^*, X)[K(X, X) + (\Sigma \otimes I))]^{-1}(Y - m(X))$$

Observe que a escolha de uma likelihood gaussiana proporcionou acharmos uma expressão analítica para a distribuição posterior. Para likelihoods mais gerais isso pode não ser o caso. Nestes cenários, é preciso seguir uma metodologia semelhante a que apresentaremos para problemas de classificação na seção 3.3.

3.2.1 Exemplo de Regressão por Processos Gaussianos

Para ilustrar a regressão por processos gaussianos, vamos considerar o exemplo artificial do dataset

$$(X, Y) = \{(1, -1), (3, 0.6), (4, 0)\}.$$

Assumimos aqui que cada par é da forma $(x, f(x))$ para uma função $f : [0, 5] \to \mathbb{R}$, i.e., não temos ruído nesse caso ($\Sigma = 0$). Nosso objetivo é estimar tal função.

Esses pontos foram definidos como amostras da função

$$f(x) = \frac{1}{15}x(x-2)(x-4)(x-6).$$

Como temos uma quantidade muito pequena de pontos (apenas 3), não é sensato esperar conseguir uma aproximação para ela no intervalo $[0, 5]$. Vamos aqui aplicar a teoria e observar o papel de cada parte que temos no modelo, não nos importando com a qualidade da previsão da distribuição posterior. Entretanto, apresentamos o gráfico da função f na figura 1.

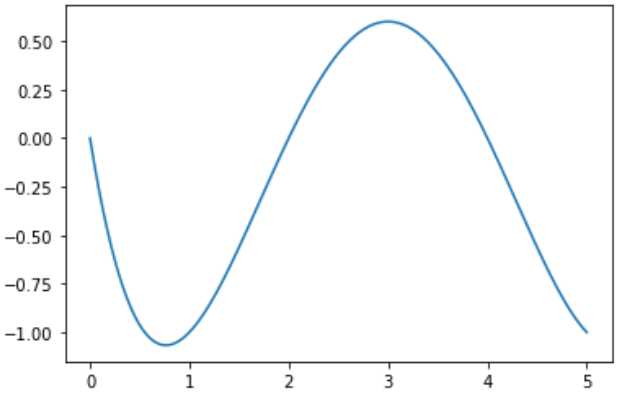


Figura 1 – Gráfico da função f.

Para começar nossa modelagem, precisamos definir nosso prior, i.e., definir a função de valor esperado m e a função de covariância k do processo gaussiano $\mathcal{GP}(m, k)$. Para k, vamos usar a função de covariância exponencial quadrática (definida em (3.1)) com hiperparâmetros $\sigma^2 = 1$ e $l_1 = 1$,

$$k_{SE}(x, y) = k_{SE}(x, y|1, 1) = \exp\left(-\frac{(x-y)^2}{2}\right).$$

Para esse exemplo, escolhemos a função de valor esperado como a função nula $m(x) = 0$,. Para ser consistente com o apresentado, tomamos a matriz Σ da likelihood do modelo como a matriz nula, i.e., $\Sigma = 0$.

Com isso, temos, por (3.6) que a distribuição posterior dos valores de $f(x)$, com $x \in [0, 5]$ e definindo $K = K(X, X)$ e $k^*(x) = K(X, x)$, é

$$f(x)|Y, X \sim N(k^*(x)^T K^{-1} Y,\ k_{SE}(x, x) - k^*(x)^T K^{-1} k^*(x)).$$

Para fins de esclarecer as definições, observe que

$$K = K(X, X) = \begin{bmatrix} k_{SE}(1,1) & k_{SE}(1,3) & k_{SE}(1,4) \\ k_{SE}(3,1) & k_{SE}(3,3) & k_{SE}(3,4) \\ k_{SE}(4,1) & k_{SE}(4,3) & k_{SE}(4,4) \end{bmatrix} = \begin{bmatrix} 1 & e^{-1} & e^{-\frac{3}{2}} \\ e^{-1} & 1 & e^{-\frac{1}{2}} \\ e^{-\frac{3}{2}} & e^{-\frac{1}{2}} & 1 \end{bmatrix},$$

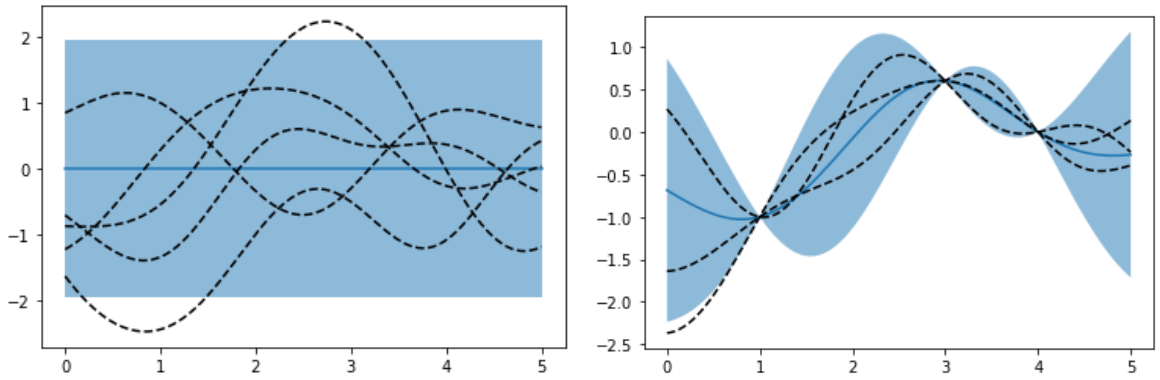
e

$$k^*(x) = K(X, x) = \begin{bmatrix} k_{SE}(1,x) \\ k_{SE}(3,x) \\ k_{SE}(4,x) \end{bmatrix} = \begin{bmatrix} e^{-\frac{(x-1)^2}{2}} \\ e^{-\frac{(x-3)^2}{2}} \\ e^{-\frac{(x-4)^2}{2}} \end{bmatrix}.$$

Portanto, a função de valor esperado da distribuição posterior é da forma

$$m|_{Y,X}(x) = \begin{bmatrix} e^{-\frac{(x-1)^2}{2}} & e^{-\frac{(x-3)^2}{2}} & e^{-\frac{(x-4)^2}{2}} \end{bmatrix} \begin{bmatrix} 1 & e^{-1} & e^{-\frac{3}{2}} \\ e^{-1} & 1 & e^{-\frac{1}{2}} \\ e^{-\frac{3}{2}} & e^{-\frac{1}{2}} & 1 \end{bmatrix}^{-1} \begin{bmatrix} -1 \\ 0.6 \\ 0 \end{bmatrix}$$

$$m|_{Y,X}(x) \approx -1.411e^{-\frac{(x-1)^2}{2}} + 1.468e^{-\frac{(x-3)^2}{2}} - 0.575e^{-\frac{(x-4)^2}{2}}.$$



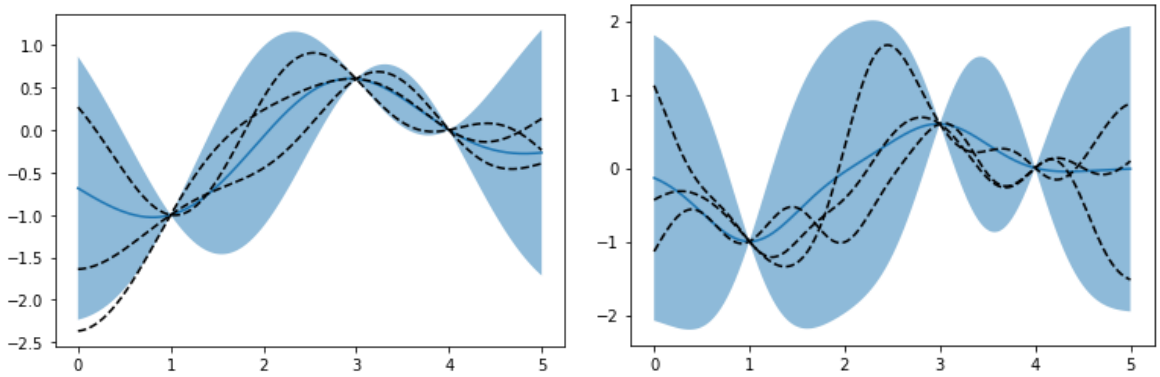
(a) Distribuição prior. (b) Distribuição posterior.

Figura 2 – Gráfico da função de valor esperado da distribuição (a) prior (b) posterior (linhas contínuas) com as regiões sombreadas representando os intervalos de confiança simétricos de 95% para cada valor $f(x)$ ao redor do valor esperado, com (a) 5 (b) 3 funções amostrais (linhas segmentadas).

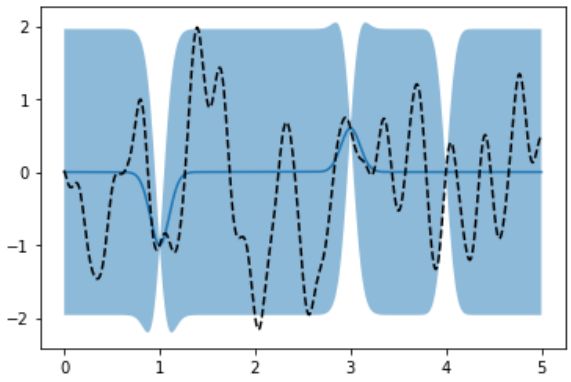
Na figura 2, ilustramos a distribuição prior e posterior. Observe na figura 2b como a incerteza do valor de $f(x)$ em cada input que temos colapsa para o output que temos (já que assumimos que não há ruído) e como a distribuição se adapta com essa informação. As funções amostrais foram geradas discretizando o intervalo e coletando amostras da distribuição normal multivariacional obtida por essa discretização seguindo o processo gaussiano considerado em cada caso.

As regiões sombreadas representam os intervalos de confiança de 95% ao redor do valor esperado de cada $f(x)$, para $x \in [0, 5]$, i.e., temos que a probabilidade do valor de $f(x)$ estar nessa região é de 95%. Veja que o quanto mais distante de um dos dados, mais largo é o intervalo de confiança, representando como o quanto mais longe dos dados, menos certeza temos do comportamento da função na região. Esse efeito fica ainda mais evidente nos extremos do intervalo $[0, 5]$.

Para ilustrar a importância da escolha dos hiperparâmetros, apresentamos na figura 3 os gráficos que ilustram a distribuição posterior de três modelos usando a função de covariância exponencial quadrática e com o hiperparâmetro $\sigma^2 = 1$, mas variando l_1 sobre os valores 1 (figura 3a), 0.5 (figura 3b) e 0.1 (figura 3c). Observe, pelas funções amostradas, que o valor de l_1 regula a frequência em que a função oscila, quanto menor o valor de l_1, mais frequentemente a função oscila. Não mostramos aqui, mas se pode observar da própria forma da função de covariância que o hiperparâmetro σ^2 regula a amplitude dos possíveis valores de $f(x)$, quanto maior o valor de σ^2, maior o intervalo dos prováveis valores.



(a) $l_1 = 1$. (b) $l_1 = 0.5$.



(c) $l_1 = 0.1$.

Figura 3 – Gráfico da função de valor esperado da distribuição posterior com hiperparâmetros $\sigma^2 = 1$ e (a) $l_1 = 1$ (b) $l_1 = 0.5$ (c) $l_1 = 0.1$ (linhas contínuas) com as regiões sombreadas representando os intervalos de confiança simétricos de 95% para cada valor $f(x)$ ao redor do valor esperado, com (a) 3 (b) 3 (c) 1 exemplo(s) de funções amostrais (linhas segmentadas).

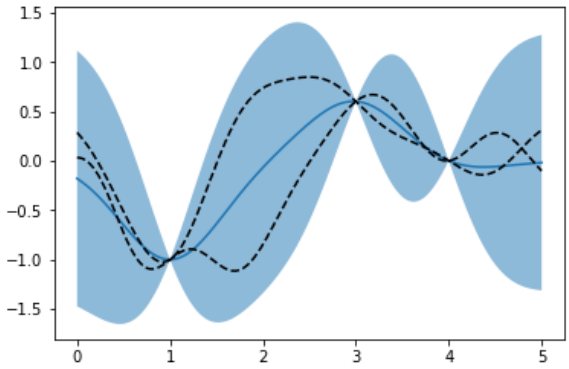


Figura 4 – Gráfico da função de valor esperado da distribuição posterior com maior marginal likelihood, com hiperparâmetros $\sigma^2 = 0.45$ e $l_1 = 0.54$, (linha contínua) com a região sombreada representando os intervalos de confiança simétricos de 95% para cada valor $f(x)$ ao redor do valor esperado, com 2 funções amostrais (linhas segmentadas).

Como discutido, podemos utilizar a equação (3.5) para calcular a marginal likelihood do modelo e escolher hiperparâmetros de modo a maximizar tal likelihood. Neste caso, um dos pares de hiperparâmetros que maximizam a marginal likelihood (com apenas duas casas decimais) são $\sigma^2 = 0.45$ e $l_1 = 0.54$. Ilustramos a distribuição posterior utilizando tais parâmetros na figura 4.

3.3 Fundamentos para Classificação por Processos Gaussianos

Considere, agora, que $(x_i, y_i)_{i=1}^n$ é um dataset com espaço de inputs $\mathcal{X}$ para o espaço de outputs $\{1, 2, \ldots, C\}$, isto é, cada x_i está relacionado com uma classe indicada pelo rótulo y_i. Tome X como a coleção de pontos com os inputs, Y o vetor com os rótulos e x^* um ponto qualquer de $\mathcal{X}$ o qual pretendemos estimar em qual classe pertence.

Nossa abordagem não vai ser estimar diretamente um rótulo y^* para um ponto x^*, mas sim estimar a probabilidade de x^* pertencer a cada uma das C classes. Se pensarmos na probabilidade de $y^* = j$, para algum $j \in \{1, 2, \ldots, C\}$, como uma função de x^*, i.e., $\pi_j(x^*) = \mathbb{P}(y^* = j|x^*)$, então nosso objetivo passa a ser uma regressão de $\mathcal{X}$ para $\mathbb{R}^C$ no intuito de estimar a função $\pi(x) = (\pi_1(x), \pi_2(x), \ldots, \pi_C(x))^T$, com as restrições de $\sum_{j=1}^C \pi_j(x) = 1$, $\forall x \in \mathcal{X}$ e $0 \leq \pi_j \leq 1$, $\forall j \in \{1, 2, \ldots, C\}$, uma vez que estamos trabalhando com uma distribuição discreta.

A ideia aqui é usar os processos gaussianos como uma função latente para essa predição. Primeiro, passaríamos do espaço dos inputs $\mathcal{X}$ para $\mathbb{R}^C$, levando x^* em $f^* = f(x^*)$ e, depois, com ajuda de uma função σ com domínio em $\mathbb{R}^C$ e que respeita as restrições impostas no parágrafo anterior, calcularíamos $\pi(x^*) = \sigma(f(x^*))$. Como não temos informações sobre $f(x^*)$ dado x^*, nós realizamos uma regressão por processos gaussianos para estimar uma distribuição para tais valores e depois calculamos a média entre todos os possíveis valores de f^* para estimar $\pi(x^*)$. Podemos justificar a escolha de um processo gaussiano com o fato de que as distribuições normais têm a maior entropia entre distribuições absolutamente contínuas com primeiro e segundo momento finitos, assim introduzindo a menor quantidade de informação ao nosso modelo, como discutido no apêndice A.5.

É interessante observar novamente que, graças aos processos gaussianos, transformamos nosso problema de classificar pontos de $\mathcal{X}$ em C classes, i.e., estimar uma função $\eta : \mathcal{X} \to \{1, 2, \ldots, C\}$, para um problema de regressão em espaços euclidianos.

Para melhor visualizar este procedimento, observe primeiramente que

$$\pi_j(x^*) = \mathbb{P}(y^* = j|x^*) = \int \mathbb{P}(y^* = j|f^*, x^*)p(f^*|x^*)\, df^* = \int p(j|f^*)p(f^*|x^*)\, df^*,$$

onde $p(f^*|x^*)$ é a pdf de $f(x^*)$, e $\mathbb{P}(y^* = j|f^*, x^*) = p(j|f^*)$ é a likelihood de y^* dado f^* para a classe i. Em nosso modelo, a primeira distribuição vai ser modelada por uma regressão por processos gaussianos, explicada em mais detalhes mais a diante, e $p(j|f^*)$ é modelada por σ_j.

Tais funções que se encaixam na descrição de σ são chamadas de *response function*. Exemplos comuns para tais funções são a função *softmax*, a generalização multivariacional da função

logística, dada por

$$p(j|f) = \frac{\exp(f_j)}{\sum_{k=1}^{C} \exp(f_k)},$$

e a função *multinomial probit*, dada por

$$p(j|f) = \mathbb{E}_{u\sim N(0,1)}\left[\prod_{k=1,k\neq j}^{C} \Phi(u + f_j - f_k)\right] = \int_{-\infty}^{\infty}\left[\prod_{k=1,k\neq j}^{C} \Phi(u + f_j - f_k)\right] \phi(u|0,1)\, du, \quad (3.7)$$

onde o valor esperado é calculado em relação a variável auxiliar u que segue uma distribuição normal padrão, i.e., com valor esperado zero e variância 1, e $\Phi(x)$ é a função distribuição acumulada de uma distribuição normal padrão.

Com isso, dividimos a inferência nos seguintes passos: primeiro, calculamos a distribuição posterior de $f^*|X, Y$ por, onde $f(x^*) = f^*$ e $f(X) = f_X$,

$$p(f^*|X, Y, x^*) = \int p(f^*|X, x^*, f_X)p(f_X|X, Y)\, df_X,$$

onde $f^*|X, x^*, f_X$ segue a equação (3.6) com Σ igual a matriz nula e $Y = f_X$, já que se pode ver isto como uma regressão por processos gaussianos sem ruído, e depois "integramos fora" f_X (que não temos informações sobre) calculando a média entre os possíveis valores de f_X segundo sua distribuição posterior.

Observe também que, pela regra de Bayes e assumindo os outputs independentes,

$$p(f_X|X, Y) = \frac{p(Y|f_X, X)p(f_X|X)}{p(Y|X)} = \frac{p(f(X))}{p(Y|X)} \prod_{i=1}^{n} p(y_i|f(x_i)),$$

onde, já que assumimos um prior $\mathcal{GP}(m, k)$ para f, temos que $f(X)$ segue (3.4), $p(y_i|f(x_i))$ é a likelihood definida pela *response function* e $p(Y|X)$ é a marginal likelihood resultante do nosso modelo.

Com isso, passamos para o próximo passo onde usamos essa distribuição posterior para calcular um estimador para $\pi_j(x^*)$. Assim, por argumentos expostos anteriormente, temos

$$\hat{\pi}_j(x^*) = \int p(j|f^*)p(f^*|X, Y, x^*)\, df^*.$$

Dessa forma, calculamos a probabilidade de x^* pertencer em cada classe. Com isso, podemos escolher a classe mais provável como estimativa para o rótulo de x^*, mas podemos também analisar o quão confiante estamos nessa escolha. Por exemplo, no caso onde as duas classes mais prováveis tem suas probabilidades muito próximas, podemos interpretar que nosso modelo não está certo entre qual das duas eleger.

Por fim, podemos resumir a estratégia geral para resolver problemas de classificação por processos gaussianos, com função de valor médio m, função de covariância k e likelihood $p(j|f)$ definidas, nos seguintes passo:

1. Calcular a densidade posterior de f_X seguindo:

$$p(f_X|X, Y) = \frac{p(Y|f_X, X)p(f_X|X)}{p(Y|X)};$$

2. Calcular a densidade posterior de f^* seguindo:

$$p(f^*|X, Y, x^*) = \int p(f^*|X, x^*, f_X)p(f_X|X, Y)\, df_X;$$

3. Calcular o estimador de $\pi_j(x^*)$, para cada $j \in \{1, 2, \ldots, C\}$, seguindo:

$$\hat{\pi}_j(x^*) = \int p(j|f^*)p(f^*|X, Y, x^*)\, df^*.$$

Infelizmente, uma escolha adequada para a likelihood geralmente nos impossibilitará de conseguir uma expressão analítica para essas integrais. Para isso, técnicas de integração numéricas e algoritmos para aproximações são essenciais para o uso viável de tais modelos. Por esse motivo, apresentaremos no capítulo 4 um destes métodos de aproximação em mais detalhes e outros serão discutidos mais brevemente no capítulo 5.

3.4 Funções de covariância

Para um processo gaussiano $f \sim \mathcal{GP}_C(m, k)$, a função de covariância $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{C\times C}$, definida como

$$k(x, y) = cov(f(x), f(y)) = \mathbb{E}[(f(x) - m(x))(f(y) - m(y))^T],$$

tem uma importância significativa nas propriedades das funções amostrais que podem ser observadas desse processo, e por isso exercem um papel importante na modelagem de problemas quando usamos processos gaussianos.

Apresentaremos na seção 3.4.1 algumas dessas propriedades regidas pela função de covariância, como, por exemplo, a continuidade e diferenciabilidade de $f(x)$. Na seção 3.4.2 apresentaremos exemplos de classes de funções de covariância que temos na literatura, tendo assim uma biblioteca de opções iniciais para a função de covariância que pretendemos usar para modelar um problema, por exemplo. Além disso, na seção 3.4.3 apresentaremos formas de obter novas funções de covariância a partir de outras já pré-conhecidas, possibilitando a criação e adaptação de novas funções de covariância para diferentes situações. Finalmente, na seção 3.4.4 temos um exemplo do uso do apresentado aqui para modelar um problema. O material apresentado nas seções 3.4.2 e 3.4.3 foi retirado em sua maior parte do livro *Gaussian processes for machine learning* (WILLIAMS; RASMUSSEN, 2006, cap. 4), que aborda ainda em mais detalhes o presente aqui.

3.4.1 Continuidade e diferenciabilidade

Como apresentado na seção 3.1, para uma função $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{C\times C}$ ser uma função de covariância válida ela tem que ser simétrica positiva semidefinida, onde simetria é definido como $k^C(x, y) = [k^C(y, x)]^T$, $\forall x, y \in \mathcal{X}$, e ser positiva semidefinida é definido como, para qualquer coleção finita de pontos Z de $\mathcal{X}$, a matriz $K^C(Z, Z)$ é positiva semidefinida. Essa caraterização não é única para processos gaussianos e pode-se estendê-la para a função de

covariância de qualquer processo estocástico (definida de maneira análoga ao apresentado, assumindo que as covariâncias sempre existem), como apontado em (ADLER, 1981). Discussões para outras definições, propriedades e proposições não demonstradas citadas nessa seção podem ser encontradas na mesma referência.

Como queremos estudar propriedades como continuidade e diferenciabilidade das possíveis funções amostrais de f, precisamos primeiramente que tais conceitos façam sentido em $\mathcal{X}$. Para isso, vamos tomar $\mathcal{X}$ como um subconjunto de $\mathbb{R}^D$ pelo resto dessa seção, mas o mesmo vale para $\mathcal{X}$ sendo um subconjunto qualquer de outro espaço vetorial euclidiano. Como estamos trabalhando com processos estocásticos, precisamos fazer algumas definições antes.

Definição 4. Seja $(\Omega, \mathcal{F}, \mathbb{P})$ um espaço de probabilidade, $\{X_i : \Omega \to \mathbb{R}^C\}_{i=1}^{\infty}$ uma sequência de vetores aleatórios e $X : \Omega \to \mathbb{R}^C$ um vetor aleatório.

(i) Dizemos que a sequência $\{X_i\}_{i=1}^{\infty}$ converge para X quase certamente quando

$$\mathbb{P}\left\{\omega \in \Omega \,\middle|\, \lim_{i\to\infty} X_i(\omega) \to X(\omega)\right\} = 1.$$

Neste caso, dizemos que $X_i \to X$ $\mathbb{P}$-a.s. quando $i \to \infty$ (abreviação do inglês *"almost surely"*).

(ii) Dizemos que a sequência $\{X_i\}_{i=1}^{\infty}$ converge para X em média quadrática quando

$$\lim_{i\to\infty} \mathbb{E}[\|X_i - X\|^2] = 0.$$

Neste caso, dizemos que $X_i \overset{m.s.}{\to} X$ quando $i \to \infty$ (abreviação do inglês *"mean square"*).

(iii) Dizemos que a sequência $\{X_i\}_{i=1}^{\infty}$ converge para X em probabilidade quando, $\forall \epsilon > 0$,

$$\lim_{i\to\infty} \mathbb{P}\{\|X_i - X\| > \epsilon\} = 0.$$

Neste caso, dizemos que $X_i \overset{\mathbb{P}}{\to} X$ quando $i \to \infty$.

Isso nos leva às seguintes definições de continuidade para processos estocásticos com conjunto de índices $\mathcal{X} \subseteq \mathbb{R}^D$.

Definição 5. Seja $f = \{f(x)\}_{x\in\mathcal{X}}$ um processo estocástico C-dimensional com $\mathcal{X} \subseteq \mathbb{R}^D$, i.e., cada $f(x)$ é um vetor aleatório com C entradas, para cada $x \in \mathcal{X} \subseteq \mathbb{R}^D$.

(i) Dizemos que o processo f é contínuo quase certamente em $x \in \mathcal{X} \subseteq \mathbb{R}^D$ se, para toda sequência $\{x_i\}_{i=1}^{\infty} \subseteq \mathcal{X}$ tal que $\lim_{i\to\infty} x_i = x$, temos

$$f(x_i) \to f(x) \,\mathbb{P}\text{-a.s. quando } i \to \infty.$$

Dizemos que f é contínuo quase certamente em $A \subseteq \mathcal{X}$ se f é contínuo quase certamente em x, $\forall x \in A \subseteq \mathcal{X}$.

(ii) Dizemos que o processo f é contínuo em média quadrática em $x \in \mathcal{X} \subseteq \mathbb{R}^D$ se, para toda sequência $\{x_i\}_{i=1}^{\infty} \subseteq \mathcal{X}$ tal que $\lim_{i\to\infty} x_i = x$, temos

$$f(x_i) \overset{m.s.}{\to} f(x) \text{ quando } i \to \infty.$$

Dizemos que f é contínuo em média quadrática em $A \subseteq \mathcal{X}$ se f é contínuo em média quadrática em x, $\forall x \in A \subseteq \mathcal{X}$.

(iii) Dizemos que o processo f é contínuo em probabilidade em $x \in \mathcal{X} \subseteq \mathbb{R}^D$ se, para toda sequência $\{x_i\}_{i=1}^{\infty} \subseteq \mathcal{X}$ tal que $\lim_{i\to\infty} x_i = x$, temos

$$f(x_i) \overset{\mathbb{P}}{\to} f(x) \text{ quando } i \to \infty.$$

Dizemos que f é contínuo em probabilidade em $A \subseteq \mathcal{X}$ se f é contínuo em probabilidade em x, $\forall x \in A \subseteq \mathcal{X}$.

Vamos discutir quais propriedades e interpretações cada tipo de continuidade nos dá. É importante lembrar que como a convergência quase certa de uma sequência de vetores aleatórios implica em convergência em probabilidade da mesma sequência e que a convergência em média quadrática também implica na convergência em probabilidade, temos que se um processo é contínuo quase certamente ou contínuo em média quadrática, ele também é contínuo em probabilidade, tendo assim suas propriedades e interpretações também. Entretanto, convergência em média quadrática não implica em convergência quase certa, e a recíproca também não é válida.

No caso de um processo f ser contínuo quase certamente, temos que qualquer função amostral tem probabilidade 1 de ser contínua em x, $\forall x \in \mathcal{X}$. É importante ressaltar a diferença entre essa condição e f ter probabilidade 1 de ser contínua em $\mathcal{X}$, já que

$$\{f \text{ contínua em } \mathcal{X}\} = \bigcap_{x\in\mathcal{X}} \{f \text{ contínua em } x\},$$

e como $\mathcal{X}$ pode ser não-enumerável, não temos garantia nem de que esse conjunto seja mensurável. Mesmo assim, a continuidade quase certamente é o tipo mais forte de continuidade para processos estocásticos entre os apresentados aqui no sentido de ser o que chega mais perto de garantir continuidade das funções amostrais. No caso dos processos gaussianos, apresentaremos no teorema 2 um critério para garantir que funções amostrais de f têm probabilidade 1 de ser contínuas em blocos compactos.

No caso de um processo f contínuo em média quadrática, temos que a convergência em média quadrática implica em

$$\lim_{s\to x} \mathbb{E}[f(s)] = \mathbb{E}[f(x)], \forall x \in \mathcal{X},$$

isto é, a função de valor médio é contínua. Além disso, no caso onde seguimos uma abordagem Bayesiana para encaixar o dataset $(x_i, y_i)_{i=1}^n$ em um modelo da forma $g(f(x)) = Y(x)$, onde g

é uma função determinística e cada y_i é uma observações de $Y(x_i)$, que por sua vez têm sua densidade definida pela likelihood $p(y|f(x))$ que assumimos ser limitada e contínua em $f(x)$ quando fixamos um valor para y, temos

$$\lim_{s \to x} \mathbb{E}[f(s)|\{g(f(x_i)) = y_i\}_{i=1}^n] = \mathbb{E}[f(x)|\{g(f(x_i)) = y_i\}_{i=i}^n],$$

i.e., a função de valor médio da distribuição posterior de f é contínua.

Para um processo f contínuo em probabilidade, temos que a convergência em probabilidade implica em toda sequência $\{f(x_n)\}_{n=1}^{\infty}$ com $x_n \to x$ quando $n \to \infty$ ter uma subsequência que converge quase certamente para $f(x)$, para qualquer $x \in \mathcal{X}$ fixo, i.e., para toda função amostral $\hat{f}$, tem-se que qualquer sequência $\{\hat{f}(x_n)\}_{n=1}^{\infty}$ com $x_n \to x$ quando $n \to \infty$ tem, com probabilidade 1, uma subsequência que converge para $\hat{f}(x)$. É interessante observar que essa propriedade é válida para cada x individualmente e não necessariamente vale para todos os pontos de $\mathcal{X}$ simultaneamente pela mesma justificativa feita no caso onde f é contínuo quase certamente.

Pelos resultados a seguir, vamos relacionar esses tipos de continuidade com propriedades da função de covariância.

Teorema 1. Um processo estocástico f é contínuo em média quadrática em $x \in \mathcal{X}$ se, e somente se, sua função de valor médio, m, é contínua em x e o traço da função de covariância, $\text{tr}(k)$, é contínuo em (x, x).

A demonstração desse teorema está no apêndice A.2.

Para o caso específico de um processo gaussiano, temos um critério para verificar a possível continuidade das funções amostrais em blocos compactos da forma $I = \prod_{i=1}^{D} [a_i, b_i]$, $a_i < b_i \in \mathbb{R}$.

Teorema 2. Seja $f \sim \mathcal{GP}_C(m, k)$ definido em um bloco compacto $I \subseteq \mathbb{R}^D$ com função de valor esperado m e função de covariância k ambas contínuas em I. Se existem $\epsilon > 0$ e $E > 0$ tal que

$$\text{tr}\left(k(s,s) + k(t,t) - 2k(s,t)\right) + \|m(s) - m(t)\|^2 \leq \frac{E}{|\log \|s - t\| \,|^{1+\epsilon}},$$

para qualquer $s, t \in I$, então as funções amostrais de f são contínuas em I com probabilidade 1.

Esse teorema é um corolário direto do teorema 3.4.1 de (ADLER, 1981, pg. 60) junto a observações feitas no apêndice A.2.

Voltando a atenção agora para a diferenciabilidade de um processo estocástico, vamos apresentar a definição de um tipo de diferenciabilidade.

Definição 6. Seja $f = \{f(x)\}_{x \in \mathcal{X}}$ um processo estocástico C-dimensional com $\mathcal{X} \subseteq \mathbb{R}^D$ aberto. Dizemos que f tem uma derivada parcial na coordenada i em média quadrática em x se existe um vetor aleatório $\dfrac{\partial f}{\partial x_i}(x)$ tal que, para toda sequência $\{h_n\}_{n=1}^{\infty} \subseteq \mathbb{R}$ tal que $\lim_{n \to \infty} h_n = 0$, temos

$$\frac{f(x + h_n e_i) - f(x)}{h_n} \xrightarrow{m.s.} \frac{\partial f}{\partial x_i}(x) \text{ quando } i \to \infty.$$

Dizemos que f é diferenciável em média quadrática em $A \subseteq \mathcal{X}$ se f tem uma derivada parcial na coordenada i em média quadrática que é contínua em média quadrática em x, $\forall x \in A \subseteq \mathcal{X}$ e $\forall i \in \{1, 2, \ldots, D\}$.

Pode-se provar que qualquer vetor aleatório $\dfrac{\partial f}{\partial x_i}(x)$ que se encaixe na definição é idêntico quase certamente, o que implica que todos compartilham uma única distribuição, e temos também que qualquer processo que siga essa distribuição também vai satisfazer a definição. Por causa disso, quando nos referirmos ao vetor aleatório $\dfrac{\partial f}{\partial x_i}(x)$, estamos nos referindo a qualquer vetor aleatório que segue essa única distribuição.

Podemos então relacionar a diferenciabilidade de uma processo com sua função de valor esperado e sua função de covariância pelo seguinte teorema.

Teorema 3. Seja $f = \{f(x)\}_{x \in \mathcal{X}}$ um processo estocástico C-dimensional com $\mathcal{X} \subseteq \mathbb{R}^D$ aberto e com função de valor médio m e função de covariância k.

Se as derivadas $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x, y)$ existem e são contínuas em $\mathcal{X} \times \mathcal{X}$ para cada $i \in \{1, 2, \ldots, D\}$ e m é continuamente diferenciável em $\mathcal{X}$, então f é diferenciável em média quadrática em $\mathcal{X}$. Além disso, cada processo $\dfrac{\partial f}{\partial x_i}$ tem sua função de valor médio dada por $\dfrac{\partial m}{\partial x_i}(x)$ e a sua função de covariância dada por $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x, y)$.

E no caso específico de processos gaussianos, temos o seguinte teorema.

Teorema 4. Seja $f \sim \mathcal{GP}_C(m, k)$ definido em $\mathcal{X} \subseteq \mathbb{R}^D$ aberto tal que as derivadas $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x, y)$ existem e são contínuas em $\mathcal{X} \times \mathcal{X}$ para cada $i \in \{1, 2, \ldots, D\}$ e m é continuamente diferenciável em $\mathcal{X}$. Assim, f é diferenciável em média quadrada e cada $\dfrac{\partial f}{\partial x_i}$ também é um processo gaussiano dado por

$$\frac{\partial f}{\partial x_i} \sim \mathcal{GP}_C\left(\frac{\partial m}{\partial x_i}, \frac{\partial^2 k}{\partial x_i \partial y_i}\right).$$

A demostração de ambos teoremas pode ser encontrada no apêndice A.2. Uma implicação direta é que se a função de covariância for $2N$ vezes continuamente diferenciável e a função de valor médio é N vezes continuamente diferenciável, então f vai ser N vezes diferenciável em média quadrática.

3.4.2 Exemplos de Funções de Covariância

Montamos nessa seção uma biblioteca de funções de covariância. Em sua maioria, apresentamos famílias de funções, indexadas por parâmetros reais, onde o espaço de inputs é $\mathbb{R}^D$. Mais ainda, na seção 3.4.3 apresentamos formas de utilizar as funções aqui apresentadas para criar novas funções de covariância para outros espaços de inputs e/ou adaptá-las para outras situações. Todas as funções apresentadas aqui têm como contradomínio $\mathbb{R}$, por causa das observações apresentadas ao final da seção 3.1.

Dizemos que uma função de covariância $k(x, y)$ é *estacionária* quando é uma função de $x - y$. Funções de covariância estacionárias são invariantes a translações no espaço dos inputs e acabam por assumir que a função $f(x)$ tem a mesma suavidade e varia na mesma velocidade por todo o seu domínio, como aponta (PACIOREK; SCHERVISH, 2003). Essas propriedades podem ou não ser desejadas e devem ser consideradas em uma modelagem. Dizemos ainda que uma função de covariância $k(x, y)$ é *isotrópica* quando é uma função de $\|x - y\|$, sendo uma propriedade mais forte que a anterior, já que temos agora a função de covariância não só invariante a translações, como a rotações e reflexões também.

Como apontaremos na seção 3.4.3, qualquer função de covariância pode ser multiplicada por uma constante positiva ou somada a uma constante positiva e continuar sendo uma função de covariância válida. Muitas vezes, tais constantes são encaradas como parâmetros de tais funções. Decidimos aqui omitir essas constantes das expressões apresentadas.

Caso não mencionadas, as demonstrações que tais funções são de fato funções de covariâncias válidas podem ser encontradas em (WILLIAMS; RASMUSSEN, 2006, cap. 4).

Funções de covariância exponencial quadrática:

Essa família de funções de covariância isotrópicas tem como espaço de inputs $\mathbb{R}^D$ e tem um parâmetro real $l > 0$. É dada pela expressão

$$k(x, y) = \exp\left(-\frac{1}{2l}\|x - y\|^2\right).$$

Uma versão estacionária dessa família, parametrizada agora por uma matriz simétrica positiva definida Σ, é dada por

$$k(x, y) = \exp\left(-\frac{1}{2}(x - y)^T\Sigma^{-1}(x - y)\right).$$

É interessante observar que quando Σ é diagonal, recuperamos a definição (3.1).

É importante observar que essa função é infinitamente diferenciável em ambas as coordenadas. Pode-se mostrar também que existem constantes tais que o enunciado do teorema 2 da seção 3.4.1 é satisfeito independente do parâmetro escolhido, i.e., um processo com uma dessas funções de covariância e uma função de valor médio nula tem suas funções amostrais contínuas com probabilidade 1 em um domínio compacto.

Temos ainda uma versão não-estacionária. Sejam $l_1(x), l_2(x), \ldots, l_D(x)$ funções reais e positivas, i.e., $l_i(x) > 0$ para qualquer x e $i = 1, 2, \ldots, D$. Temos então a expressão

$$k(x, y) = \prod_{d=1}^{D}\left(\frac{2l_d(x)l_d(y)}{l_d^2(x) + l_d^2(y)}\right)^{\frac{1}{2}} \exp\left(-\sum_{d=1}^{D}\frac{(x_d - y_d)^2}{l_d^2(x) + l_d^2(y)}\right).$$

Funções de covariância de Matérn:

Essa família de funções de covariância isotrópicas tem como espaço de inputs $\mathbb{R}^D$ e tem dois parâmetros reais positivos, $l > 0$ e $\nu > 0$. É dada pela expressão

$$k(x, y) = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}\,\|x - y\|}{l}\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}\,\|x - y\|}{l}\right),$$

onde $\Gamma(z) = \int_0^\infty x^{z-1}e^{-x}\,dx$ é a função gama e K_ν é a função modificada de Bessel de parâmetro ν (ABRAMOWITZ; STEGUN, 1964, seção 9.6).

Pode-se mostrar que quando $\nu \to \infty$, recuperamos a função de covariância exponencial quadrática isotrópica com parâmetro l. Temos também que um processo f com uma dessas funções de covariância é k-vezes diferenciável em média quadrada se, e só se, $k < \nu$. Para valores específicos de ν, ressaltamos $\nu = \frac{1}{2}, \frac{3}{2}, \frac{5}{2}$:

$$
\begin{aligned}
k_{\nu=\frac{1}{2}}(x,y) &= \exp\left(-\frac{\|x-y\|}{l}\right);\\
k_{\nu=\frac{3}{2}}(x,y) &= \exp\left(-\frac{\sqrt{3}}{l}\|x-y\|\right)\left(1+\frac{\sqrt{3}}{l}\|x-y\|\right);\\
k_{\nu=\frac{5}{2}}(x,y) &= \exp\left(-\frac{\sqrt{5}}{l}\|x-y\|\right)\left(1+\frac{\sqrt{5}}{l}\|x-y\|+\frac{5}{2l^2}\|x-y\|^2\right).
\end{aligned}
$$

Funções de covariância γ-exponencial:

Essa família de funções de covariância isotrópicas tem como espaço de inputs $\mathbb{R}^D$ e tem dois parâmetros reais positivos, $l > 0$ e $0 < \gamma \leq 2$. É dada pela expressão

$$k(x,y) = \exp\left(-\left(\frac{\|x-y\|}{l}\right)^\gamma\right).$$

É interessante observar que um processo f com uma dessas funções de covariância não é diferenciável em média quadrada, a não ser no caso onde $\gamma = 2$, onde recuperamos a função de covariância exponencial quadrática com parâmetro $\frac{l^2}{2}$.

Funções de covariância racional quadrática:

Essa família de funções de covariância isotrópicas tem como espaço de inputs $\mathbb{R}^D$ e tem dois parâmetros reais positivos, $l > 0$ e $\alpha > 0$. É dada pela expressão

$$k(x,y) = \left(1+\frac{\|x-y\|^2}{2\alpha l}\right)^{-\alpha}.$$

É interessante observar que temos uma função infinitamente diferenciável e que quando $\alpha \to \infty$, recuperamos a função de covariância exponencial quadrática com parâmetro l.

Funções de covariância de rede neural:

Essa família de funções de covariância não estacionária tem como espaço de inputs $\mathbb{R}^D$ e tem um parâmetro, uma matriz $(D+1)\times(D+1)$ semipositiva definida Σ. É dada pela expressão

$$k(x,y) = \frac{2}{\pi}\sin^{-1}\left(\frac{2\tilde{x}^T\Sigma\tilde{y}}{\sqrt{(1+2\tilde{x}^T\Sigma\tilde{x})(1+2\tilde{y}^T\Sigma\tilde{y})}}\right),$$

onde $\tilde{x}^T = \begin{bmatrix}1 & x^T\end{bmatrix}$. É interessante observar que temos uma função infinitamente diferenciável.

O nome de tal função de covariância vem do fato de que uma rede neural Bayesiana com função de ativação da forma $h(x) = \text{erf}(u_0 + u^T x)$, onde $u_0 \in \mathbb{R}$, $u \in \mathbb{R}^D$ e $\text{erf}(z) = \frac{2}{\sqrt{\pi}}\int_0^z e^{-t^2}\,dt$, converge em distribuição para um processo gaussiano com essa função de covariância conforme

o número de nós cresce para infinito, como discute (NEAL, 1996) de forma mais geral e como mostra (WILLIAMS, 1998).

Funções de covariância polinomial:

Essa família de funções de covariância não estacionária tem como espaço de inputs $\mathbb{R}^D$ e tem três parâmetros, uma matriz simétrica semipositiva definida Σ, $\sigma_0^2 \geq 0$ e $p \in \{1, 2, 3, \dots\}$. É dada pela expressão

$$k(x, y) = \left(x^T \Sigma y + \sigma_0^2\right)^p.$$

Como caso particular, podemos escolher $\Sigma = 0$, tornado k constante.

Segundo (WILLIAMS; RASMUSSEN, 2006, pg. 90), tais funções se mostraram eficientes para problemas de classificação em dimensões grandes quando os dados tem cada uma de suas entradas em $[-1, 1]$.

É interessante observar que para o caso onde realizamos uma regressão por processos gaussianos segundo (3.6) com m identicamente nula e utilizando essa função de covariância com os hiperparâmetros $p = 1$ e $\Sigma = \Omega$, temos que a função de valor médio posterior é uma função polinomial de primeiro grau, já que

$$\begin{aligned} m_{posterior}(x_*) &= K(x_*, X)[K(X, X) + (\Sigma \otimes I))]^{-1} Y \\ = \sum_{i=1}^{n} \alpha_i k(x_*, x_i) &= x_*^T \left(\sum_{i=1}^{n} \alpha_i \Omega x_i\right) + \left(\sum_{i=1}^{n} \sigma_0^2 \alpha_i\right), \\ &= x_*^T \beta + \beta_0, \end{aligned}$$

onde α_i é a $i-$ésima entrada de $[K(X, X) + (\Sigma \otimes I))]^{-1} Y$. Em outras palavras, obtemos uma regressão linear para o dataset seguindo uma abordagem Bayesiana ao escolher essa função de covariância com $p = 1$.

Funções de covariância polinomial por partes com suporte compacto:

Essa família de funções de covariância isotrópicas tem como espaço de inputs $\mathbb{R}^D$ e um parâmetro $q \in \{0, 1, 2, 3, \dots\}$. Apontamos a expressão para alguns casos:

$$\begin{aligned} k_{q=0}(x, y) &= \max((1 - \|x - y\|)^j, 0); \\ k_{q=1}(x, y) &= \max((1 - \|x - y\|)^{j+1}, 0)\left((j + 1)\|x - y\| + 1\right); \\ k_{q=2}(x, y) &= \max((1 - \|x - y\|)^{j+2}, 0)\left((j^2 + 4j + 3)\|x - y\|^2 + (3j + 6)\|x - y\| + 3\right); \\ k_{q=3}(x, y) &= \max((1 - \|x - y\|)^{j+3}, 0)\left((j^3 + 9j^2 + 23j + 15)\|x - y\|^3 + (6j^2 + 36j + 45)\|x - y\|^2\right. \\ &\qquad \left. + (15j + 45)\|x - y\| + 15\right); \end{aligned}$$

onde $j = \left\lfloor \frac{D}{2} \right\rfloor + q + 1$ (observe a dependência da dimensão D do espaço de inputs). Pode-se achar as funções para os demais valores de q em (WENDLAND, 2004, pg. 128). Um processo f com uma dessas funções de covariância é $q-$vezes diferenciável em média quadrada.

A propriedade mais relevante aqui é o suporte compacto da função de covariância, já que esta gera matrizes de covariância esparsas, o que pode trazer vantagens computacionais consideráveis. Isso é relevante, por exemplo, para o caso de datasets com um número exorbitante de dados.

Função de covariância trigonométrica com suporte compacto:

Essa função de covariância isotrópica tem como espaço de inputs $\mathbb{R}^D$ e é dada pela expressão

$$k(x,y) = \begin{cases} \frac{2+\cos(2\pi\|x-y\|)}{3}(1-\|x-y\|) + \frac{1}{2\pi}\sin(2\pi\|x-y\|), & \text{se } \|x-y\|<1; \\ 0, & \text{se } \|x-y\|\geq 1. \end{cases}$$

Um processo f com uma dessas funções de covariância é duas vezes diferenciável em média quadrada. Importante observar que apesar do nome que damos aqui para essa função, ela não é uma função de covariância periódica. A demonstração que essa é uma função de covariância válida e sua derivação pode ser vista em (MELKUMYAN; RAMOS, 2009).

Assim como na função anterior, o suporte compacto é de grande relevância, já que gera matrizes de covariância esparsas, o que traz vantagens computacionais quando considerado na implementação. Isso é importante, por exemplo, para a inferência em datasets muito grandes.

Funções de covariância periódicas:

Essa família de funções de covariância isotrópicas tem como espaço de inputs $\mathbb{R}$ e tem um parâmetro $l > 0$. É dada pela expressão

$$k(x,y) = \exp\left(-\frac{2\sin^2\left(\frac{x-y}{2}\right)}{l}\right).$$

É interessante observar que temos uma função infinitamente diferenciável e que é periódica com um período de 2π quando observada como uma função de $r = x - y$.

Função de covariância de Cauchy:

Essa função de covariância não estacionária tem como espaço de inputs $(0,\infty]$ e é dada pela expressão

$$k(x,y) = \frac{1}{x+y}.$$

A justificativa que essa é uma função de covariância válida pode ser encontrada em (FIEDLER, 2010).

Função de covariância do processo de Wiener:

O processo de Wiener, também conhecido como movimento browniano, é um dos processos gaussianos mais conhecidos. Tendo como função de valor médio a função nula e uma função de covariância não estacionária para $x, y \in [0,\infty)$ dada por

$$k(x,y) = \min(x,y).$$

Pelo teorema 2 da seção 3.4.1, temos com probabilidade 1 que as funções amostrais do processo de Wiener são contínuas, pois podemos tomar $\epsilon = 1$, $E = 1$ e $I = [n, n+1]$ para cada $n \in \{0,1,2,\dots\}$. Entretanto, temos que essa função de covariância não é diferenciável, e portanto nossa teoria não se aplica aqui. Excepcionalmente, um resultado muito conhecido é que para qualquer $x \in [0,\infty)$, as funções amostrais do processo de Wiener têm probabilidade 1 de não serem diferenciáveis em x. Para uma discussão mais detalhada sobre esse processo, apontamos para (KARATZAS; SHREVE, 2014).

3.4.3 Operações com Funções de Covariância

Nessa seção apresentaremos jeitos de utilizar as funções de covariância apresentadas na seção anterior para criar novas funções de covariância, introduzir parâmetros ou outras modificações que podem ser feitas para adaptar os exemplos apresentados para diversas situações. Consideramos o contradomínio de todas as funções consideradas como $\mathbb{R}$. Caso não mencionadas, demonstrações para essas propriedades podem ser vistas em (WILLIAMS; RASMUSSEN, 2006, pg. 95).

Soma:

Sejam $k_1(x, y)$ e $k_2(x, y)$ funções de covariância, então $k(x, y) = k_1(x, y) + k_2(x, y)$ é uma função de covariância válida.

Como um caso particular, podemos adicionar uma constante a qualquer função de covariância. Tal constante normalmente é interpretada como um parâmetro da nova função de covariância.

Produto:

Sejam $k_1(x, y)$ e $k_2(x, y)$ funções de covariância, então $k(x, y) = k_1(x, y)k_2(x, y)$ é uma função de covariância válida. Sejam também $k_1(x, y)$ uma função de covariância e $a(x)$ uma função qualquer, então $k(x, y) = a(x)k(x, y)a(y)$ é uma função de covariância válida.

Como casos particulares, temos que se $k(x, y)$ é uma função de covariância, então $(k(x, y))^p$ é uma função de covariância válida para qualquer $p \in \mathbb{N}^*$. Temos também que se $a(x) = a$ é constante, então $a^2k(x, y)$ é uma função de covariância válida, i.e., podemos multiplicar uma função de covariância por qualquer constante positiva (normalmente encaramos essa constante como um parâmetro da nova função). Por fim, assumindo que $k_1(x, x) \neq 0$ para todo x, podemos tomar $a(x) = (k_1(x, x))^{-\frac{1}{2}}$, e assim

$$k(x, y) = \frac{k_1(x, y)}{\sqrt{k_1(x, x)k_1(y, y)}}$$

é uma função de covariância válida. Observe que dessa forma temos $k(x, x) = 1, \forall x$.

Convolução:

Sejam $k_1(x, y)$ uma função de covariância e $h(x, y)$ uma função com mesmo domínio que k_1. Então $k(x, y) = \int h(x, u)k_1(u, v)h(y, v)\, du\, dv$ é uma função de covariância válida.

Diferenciação:

Seja $k(x, y)$ uma função de covariância com espaço de inputs $\mathcal{X} \subseteq \mathbb{R}^D$ e duas vezes continuamente diferenciável. Pelo teorema 4 da seção 3.4.1, temos que $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x, y)$ é uma função de covariância válida.

Soma direta e produto cartesiano:

Seja $k_1(x, y)$ uma função de covariância para um espaço de inputs $\mathcal{X}_1$ e $k_2(x, y)$ uma função de covariância para um espaço de inputs $\mathcal{X}_2$. Então $k(x_1+x_2, y_1+x_2) = k_1(x_1, y_1)+k_2(x_2, y_2)$ é uma função de covariância válida para a soma direta $\mathcal{X}_1 \oplus \mathcal{X}_2$. Também temos que $k((x_1, x_2), (y_1, y_2)) = k_1(x_1, y_1)k_2(x_2, y_2)$ é uma função de covariância válida para o produto cartesiano $\mathcal{X}_1 \times \mathcal{X}_2$.

Como caso particular, podemos criar uma função de covariância com domínio em $\mathbb{R}^D$ escolhendo funções de covariância $k_1, k_2, \ldots, k_D$ com domínio em $\mathbb{R}$ e definindo $k(x, y) = \prod_{i=1}^{D} k_i(x_i, y_i)$, onde x_i e y_i são as $i-$ésimas coordenadas de x e y respectivamente.

***Warping*:**

Sejam $k_1(x, y)$ uma função de covariância para um espaço de inputs $\mathcal{X}_1$ e $\alpha : \mathcal{X}_2 \to \mathcal{X}_1$ uma função qualquer. Então $k(x, y) = k_1(\alpha(x), \alpha(y))$ é uma função de covariância válida para $\mathcal{X}_2$.

Como um caso particular, tomamos $\alpha(x) = \hat{k}(x, x)$, onde $\hat{k}$ é uma função de covariância tal que $\hat{k}(x, x) \neq 0$. Escolhemos $k_1(x, y)$ como a função de covariância de Cauchy. Assim,

$$k(x, y) = \frac{1}{\hat{k}(x, x) + \hat{k}(y, y)}$$

é uma função de covariância válida. Utilizando as propriedades de produtos apresentados anteriormente, temos que

$$k(x, y) = \frac{2\hat{k}(x, y)}{\hat{k}(x, x) + \hat{k}(y, y)}$$

é uma função de covariância válida. Observe que novamente temos $k(x, x) = 1, \forall x$.

Transformando uma função isotrópica em uma não estacionária:

Seja $k_I(x, y)$ uma função de covariância isotrópica com $x, y \in \mathbb{R}^D$. Neste caso, existe $\hat{k}_I : \mathbb{R} \to \mathbb{R}$ tal que $k_1(x, y) = \hat{k}_I(r)$, onde $r = \|x - y\|$. Seja $\Sigma(x)$ uma função tal que, para cada $x \in \mathbb{R}^D$, $\Sigma(x)$ é uma matriz $D \times D$ positiva definida. Então

$$k(x, y) = 2^{\frac{D}{2}} \frac{\det(\Sigma(x))^{\frac{1}{4}} \det(\Sigma(y))^{\frac{1}{4}}}{\det(\Sigma(x) + \Sigma(y))^{\frac{1}{2}}} \hat{k}_I\left(\sqrt{Q(x, y)}\right);$$

$$Q(x, y) = (x - y)\left(\frac{\Sigma(x) + \Sigma(y)}{2}\right)^{-1}(x - y),$$

é uma função de covariância válida e, se $\Sigma(x)$ não for constante, não estacionária.

No caso onde temos $\Sigma(x) = \Sigma$ constante, então $k(x, y) = \hat{k}_i\left(\sqrt{(x - y)\Sigma^{-1}(x - y)}\right)$ é uma função de covariância válida e estacionária.

A demonstração para essa afirmação pode ser vista em (PACIOREK, 2003).

3.4.4 Exemplo: SSIM como uma função de covariância

Nessa seção vamos utilizar o apresentado na seção 3.4 para mostrar que a função SSIM usada na área de processamento de imagens é uma função de covariância válida e uma candidata em modelagens que os inputs são imagens. Ressaltamos que não temos conhecimento de tal desenvolvimento atualmente na literatura e isso pode ter consequências para o uso de métodos por processos gaussianos na área de processamentos de imagens.

SSIM é um acrônimo para *structural similarity index*, foi originalmente proposto como um índice para medir a similaridade entre duas imagens em (WANG et al., 2004) e vem apresentando bons resultados na prática desde então.

Tomando duas imagens de mesmas resoluções x e y na forma de vetores com N entradas não negativas, (WANG et al., 2004) define o SSIM entre x e y como

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)},$$

onde

$$\mu_x = \sum_{i=1}^{N} w_i x_i,$$

$$\sigma_{xy} = \sum_{i=1}^{N} w_i (x_i - \mu_x)(y_i - \mu_y),$$

$\sigma_x^2 = \sigma_{xx}$, $C_1, C_2 > 0$ são constantes e os pesos w_i tais que $\sum_{i=1}^{N} w_1 = 1$ e $w_i \geq 0, \forall i$.

Vamos reescrever tais expressões. Para isso, considere w como o vetor de pesos, i.e., o vetor onde, para cada i, a $i-$ésima entrada é w_i. Assim,

$$\mu_x = x^T w;$$

$$\mu_x \mu_y = x^T w w^T y;$$

$$\begin{aligned}\sigma_{xy} &= \sum_{i=1}^{N} w_i (x_i - \mu_x)(y_i - \mu_y) = \sum_{i=1}^{N} w_i x_i y_i - \mu_x \sum_{i=1}^{N} w_i y_i - \mu_y \sum_{i=1}^{N} w_i x_i + \sum_{i=1}^{N} w_i \mu_x \mu_y \\ &= x^T \text{diag}(w) y - \mu_x \mu_y - \mu_x \mu_y + \mu_x \mu_y = x^T \text{diag}(w) y - x^T w w^T y \\ &= x^T \left(\text{diag}(w) - w w^T\right) y.\end{aligned}$$

Note que ambas as matrizes ww^T e $\text{diag}(w) - ww^T$ são simétricas. Além disso, elas são semipositivas definidas, já que se $z \in \mathbb{R}^N$, então temos

$$0 \leq (z^T w)^2 = z^T w w^T z,$$

e, pela desigualdade de Jensen,

$$z^T w w^T z = \left(\sum_{i=1}^{N} w_i z_i \right)^2 \leq \sum_{i=1}^{N} w_i z_i^2 = z^T \text{diag}(w) z$$

$$\Rightarrow 0 \leq z^T \left(\text{diag}(w) - w w^T\right) z.$$

Com isso, temos que as funções

$$\hat{k}_1(x, y) = \mu_x \mu_y + \frac{C_1}{2} = x^T w w^T y + \frac{C_1}{2}$$

e

$$\hat{k}_2(x, y) = \sigma_{xy} + \frac{C_2}{2} = x^T \left(\text{diag}(w) - w w^T\right) y + \frac{C_1}{2}$$

são funções de covariância polinomiais com parâmetros $\Sigma = ww^T$, $\sigma_0^2 = \frac{C_1}{2}$, $p = 1$ e $\Sigma = \text{diag}(w) - ww^T$, $\sigma_0^2 = \frac{C_2}{2}$, $p = 1$ respectivamente. Observe que como $C_1, C_2 > 0$, então $k_1(x, x) \neq 0$ e $k_2(x, x) \neq 0$. Assim, pelo caso particular discutido na propriedade de *warping* da seção 3.4.3, temos que

$$k_1(x, y) = \frac{2\hat{k}_1(x, y)}{\hat{k}_1(x, x) + \hat{k}_1(y, y)} = \frac{(2\mu_x \mu_y + C_1)}{(\mu_x^2 + \mu_y^2 + C_1)}$$

e

$$k_2(x,y) = \frac{2\hat{k}_2(x,y)}{\hat{k}_2(x,x) + \hat{k}_2(y,y)} = \frac{(2\sigma_{xy} + C_2)}{(\sigma_x^2 + \sigma_y^2 + C_2)}$$

são funções de covariância válidas.

Por fim, como o produto de funções de covariância é uma função de covariância, temos que

$$\text{SSIM}(x,y) = k_1(x,y)k_2(x,y)$$

é uma função de covariância válida.

Ainda mais, se desejado, devido as propriedades de soma e produto apresentados, podemos introduzir mais dois parâmetros: $\alpha > 0$, que controla a amplitude da função, e $\beta \geq 0$, que modela a incerteza sobre os dados observados. Tendo assim

$$k_{SSIM}(x,y|\alpha,\beta,C_1,C_2,w) = \alpha \cdot \text{SSIM}(x,y|C_1,C_2,w) + \beta$$

como uma possível escolha de função de covariância.

Em aplicações práticas, é comum utilizar a SSIM em subseções equivalentes das imagens x e y, chamadas aqui de janelas, e fazer uma média desses valores. Veja que para uma escolha apropriada de pesos w, temos que a SSIM entre duas janelas equivalentes é uma função de covariância válida. Dessa forma, escolhendo M pares de janelas equivalentes de x e y, denotadas por $(x^{(1)}, y^{(1)}), \ldots, (x^{(M)}, y^{(M)})$, temos que

$$k_{MSSIM}(x,y|\alpha,\{C_1^{(j)}\},\{C_2^{(j)}\},\{w^{(j)}\}) = \alpha_0 + \sum_{j=1}^{M} \alpha_j \text{SSIM}(x^{(j)}, y^{(j)}|C_1^{(j)}, C_2^{(j)}, w^{(i)})$$

é uma função de covariância válida, onde $\alpha \in \mathbb{R}^{M+1}$, $\alpha_0 \geq 0$ e $\alpha_j, C_1^{(j)}, C_2^{(j)} > 0$ e $w^{(i)}$ é um vetor de pesos adequados para as janelas $(x^{(j)}, y^{(j)})$ conforme definido anteriormente, $\forall j \in \{1, \ldots, M\}$.

A forma apresentada de k_{MSSIM} tem um grande número de parâmetros e escolhas que se pode tomar. Na prática, costuma-se escolher as janelas como todos os sub-blocos $m \times m$ entorno de cada pixel correspondente entre as imagens e tomar C_1, C_2 e os pesos w iguais para todas as janelas (talvez adaptando w para os blocos que estão nas bordas). Uma estratégia razoável seria fixar C_1, C_2 e as escolhas de w e deixar α, o vetor de pesos (não normalizado) dos pares de janelas, livre para ser escolhido de forma a maximizar a marginal likelihood do modelo.

No caso de termos diferentes escolhas de distribuições de pesos w, podemos usar o discutido na seção 2.2.3 para justificar a comparação das marginal likelihoods de cada modelo com distribuições de pesos distintas, com α de cada modelo já escolhido de forma a maximizar a marginal likelihood correspondente, e escolher o que apresenta a maior marginal likelihood.

4 Expectation Propagation para Classificação por Processos Gaussianos

4.1 Método Geral

O método de expectation propagation (MINKA, 2001b) como uma ferramenta para aproximações de distribuições é mais geral, e aqui mostraremos um uso específico deste para classificação por processos gaussianos.

O modelo que seguiremos toma como prior $f \sim \mathcal{GP}_C(0, [k_1, \ldots, k_C])$, isto é, tomamos a função de valor médio como a função nula e que cada função componente é a priori independente das outras, como discutido anteriormente. Mudanças nessas escolhas de funções podem ser adaptadas do exposto aqui. Para a likelihood, por hora tome qualquer $p(j|f)$ apropriada. Daqui em diante, denotaremos $f(X)$ apenas por f e $f(x_i)$ for f_i.

Nosso objetivo com esse método é achar uma aproximação normal $q(f|X,Y)$ para a densidade posterior $p(f|X,Y)$. Escolhemos aproximar por uma distribuição normal pois, como visto no caso apresentado na seção sobre princípios para regressão por processos gaussianos, conseguimos achar uma expressão analítica para a distribuição posterior.

Precisamos de um critério para definir tal aproximação. Um critério apresentado anteriormente seria minimizar a divergência de Kullback-Leibler de $Q(f|X,Y)$ para $P(f|X,Y)$, $D_{KL}(P(f|X,Y)||Q(f|X,Y))$, onde estas são as distribuições com pdf $q(f|X,Y)$ e $p(f|X,Y)$ respectivamente (em contrapartida, métodos que visam minimizar $D_{KL}(Q(f|X,Y)||P(f|X,Y))$ são conhecidos por *Variational Bayesian*). Infelizmente, minimizar tal divergência pode ser inviável ou ter um custo proibitivo. Por isso, no intuito de se aproximar desse mínimo, o método de expectation propagation propõe aproveitar a forma da distribuição posterior $p(f|X,Y)$:

$$p(f|X,Y) = \frac{p(f|X)}{p(Y|X)} \prod_{i=1}^{n} p(y_i|f_i),$$

e alcançar $q(f|X,Y)$ fazendo aproximações locais t_i de cada likelihood $p(y_i|f_i)$ em função de f_i por meio de distribuições gaussianas não-normalizadas com parâmetros locais $\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i$:

$$p(y_i|f_i) \approx t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i) = \tilde{Z}_i \phi(f_i|\tilde{\mu}_i, \tilde{\Sigma}_i),$$

onde $\phi(f_i|\tilde{\mu}_i, \tilde{\Sigma}_i)$ é a pdf de $N(\tilde{\mu}_i, \tilde{\Sigma}_i)$.

O parâmetro $\tilde{Z}_i$ seria uma aproximação de $\int p(y_i|f_i)\, df_i$, uma vez que, como uma função de f_i, $p(y_i|f_i)$ não tem necessariamente integral igual a um. Para $\tilde{\mu}_i$ e $\tilde{\Sigma}_i$, o método de expectation propagation propõem um processo iterativo atualizando os parâmetros locais sequencialmente, onde, em cada passo, os parâmetros locais eleitos são atualizados de forma a minimizar uma divergência entre distribuições marginais para f_i envolvendo o prior $p(f|X)$, as outras aproximações locais e a likelihood exata $p(y_i|f_i)$, até a convergência de todos os parâmetros para um ponto fixo.

Para explicar em mais detalhes estes passos, precisamos primeiramente fazer algumas construções.

Para começar, montamos o produto das aproximações locais (independentes entre si) como

$$\prod_{i=1}^{n} t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i) = \phi(f|\tilde{\mu}, \tilde{\Sigma}) \prod_{i=1}^{n} \tilde{Z}_i,$$

onde $\tilde{\mu}$ é o vetor contendo os $\tilde{\mu}_i$ arranjados de acordo com (3.3) e $\tilde{\Sigma} = \sum_{i=1}^{n}(\tilde{\Sigma}_i \otimes e_i e_i^T)$, onde e_i é o i-ésimo vetor da base canônica de $\mathbb{R}^n$ e $\otimes$ é o produto de Kronecker, devido a como f está arranjado.

Portanto, nossa aproximação normal para a distribuição posterior é

$$q(f|X,Y) = \frac{1}{Z_{EP}} p(f|X) \prod_{i=1}^{n} t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i) = \frac{1}{Z_{EP}} p(f|X)\phi(f|\tilde{\mu}, \tilde{\Sigma}) \prod_{i=1}^{n} \tilde{Z}_i = \phi(f|\mu, \Sigma), \quad (4.1)$$

onde Z_{EP} é a aproximação deste algoritmo para a marginal likelihood (desenvolvemos mais sobre isso na seção 4.4). Com o auxílio do apêndice (A.4), e lembrando que $f|X \sim N(0, K(X,X) = K)$, temos que μ e Σ devem seguir

$$\Sigma = (K^{-1} + \tilde{\Sigma}^{-1})^{-1} \quad \text{e} \quad \mu = \Sigma\tilde{\Sigma}^{-1}\tilde{\mu}. \quad (4.2)$$

Com isso, definimos a distribuição marginal para f_i de $q(f|X,Y)$ por

$$q(f_i|X,Y) = \int q(f|X,Y) \prod_{j=1, j\neq i}^{n} df_j = \phi(f_i|\mu_i, \Sigma_i), \quad (4.3)$$

onde μ_i e Σ_i são o subvetor e a submatriz correspondentes de μ e Σ para f_i respectivamente, como mostrado em (A.2).

Definimos agora a distribuição *cavity* $q_{-i}(f_i)$ para f_i, que exclui a aproximação local para a likelihood de f_i e fixa todas as outras, como se deixasse um "buraco" no lugar onde devia estar a likelihood de f_i, daí o nome *cavity* (do inglês para cavidade):

$$q_{-i}(f_i) \propto \int p(f|X) \prod_{j=1, j\neq i}^{n} t_j(f_j|\tilde{Z}_j, \tilde{\mu}_j, \tilde{\Sigma}_j)\, df_j. \quad (4.4)$$

Juntando as equações (4.3) e (4.4), temos que

$$q_{-i}(f_i) t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i) \propto q(f_i|X,Y). \quad (4.5)$$

O que nos dá, novamente pelo apêndice (A.4), que

$$q_{-i}(f_i) = \phi(f_i|\mu_{-i}, \Sigma_{-i}), \quad \text{com} \quad \Sigma_{-i} = (\Sigma_i^{-1} - \tilde{\Sigma}_i^{-1})^{-1} \quad \text{e} \quad \mu_{-i} = \Sigma_{-i}(\Sigma_i^{-1}\mu_i - \tilde{\Sigma}_i^{-1}\tilde{\mu}_i). \quad (4.6)$$

No intuito de dar um passo no método proposto e caminhar para a convergência, construímos agora uma nova aproximação normal não-normalizada $\hat{q}_i(f_i)$ para aproximar o produto dessa distribuição *cavity* com a likelihood exata $p(y_i|f_i)$ da seguinte forma:

$$\hat{q}_i(f_i) = \hat{Z}_i \phi(\hat{\mu}_i, \hat{\Sigma}_i) \approx q_{-i}(f_i) p(y_i|f_i).$$

O parâmetro $\hat{Z}_i$ é novamente escolhido como uma aproximação da constante de normalização $Z_i = \int q_{-i}(f_i)p(y_i|f_i)\,df_i$. Já os parâmetros $\hat{\mu}_i$ e $\hat{\Sigma}_i$ são escolhidos de forma a minimizar $D_{KL}(Q_{-i}P_i||N(\hat{\mu}_i, \hat{\Sigma}_i))$, onde $Q_{-i}P_i$ é um distribuição sobre f_i com pdf igual a $\frac{1}{Z_i}q_{-i}(f_i)p(y_i|f_i)$. Como demostrado no apêndice A.4, a divergência é minimizada ao igualarmos o primeiro e segundo momento da aproximação normal com os de $Q_{-i}P_i$, isto é, temos que os valores ideais que minimizam a divergência são

$$\hat{\Sigma}_i = var_{Q_{-i}P_i}(f_i) \quad \text{e} \quad \hat{\mu}_i = \mathbb{E}_{Q_{-i}P_i}[f_i]. \tag{4.7}$$

Estes momentos podem ser calculados com mais facilidade devido a q_{-i} ser a pdf de uma distribuição normal. Por exemplo, podemos reescrever

$$\mathbb{E}_{Q_{-i}P_i}[f_i] = \int f_i q_{-i}(f_i)p(y_i|f_i)\,df_i = \int f_i p(y_i|f_i)\phi(f_i|\mu_{-i}, \Sigma_{-i})\,df_i = \mathbb{E}_{N(\mu_{-i},\Sigma_{-i})}[f_i p(y_i|f_i)]$$

e o cálculo de valores esperados com respeito a distribuições normais pode ser manuseado com mais facilidade, visto que há diversos métodos para lidar com tais na literatura (discutimos mais sobre tais métodos na seção 4.5.1). O mesmo pode ser feito para $\hat{Z}_i$ e para $var_{Q_{-i}P_i}(f_i)$.

Como incorporamos a likelihood exata $p(y_i|f_i)$ em $\hat{q}_i$, podemos considerar essa como uma aproximação melhor para a distribuição marginal posterior exata de f_i do que a $q(f_i|X, Y)$ usada. Por isso, atualizamos os parâmetros locais de t_i de forma a satisfazer a seguinte versão modificada da equação (4.5):

$$q_{-i}(f_i)t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i) \propto \hat{q}_i(f_i). \tag{4.8}$$

O que, novamente pelo apêndice (A.4), nos dá as equações

$$\tilde{\Sigma}_i = \left(\hat{\Sigma}_i^{-1} - \Sigma_{-i}^{-1}\right)^{-1} \quad \text{e} \quad \tilde{\mu}_i = \tilde{\Sigma}_i\left(\hat{\Sigma}_i^{-1}\hat{\mu}_i - \Sigma_{-i}^{-1}\mu_{-i}\right), \tag{4.9}$$

e

$$\tilde{Z}_i = \hat{Z}_i(2\pi)^{\frac{C}{2}} \det\left(\Sigma_{-i} + \tilde{\Sigma}_i\right)^{\frac{1}{2}} \exp\left(\frac{1}{2}(\mu_{-i} - \tilde{\mu}_i)^T\left(\Sigma_{-i} + \tilde{\Sigma}_i\right)^{-1}(\mu_{-i} - \tilde{\mu}_i)\right). \tag{4.10}$$

Portanto, podemos separar o algoritmo iterativo proposto pelo método de expectation propagation neste caso nos seguintes passos:

1. Fixar um $i \in \{1, 2, \ldots, n\}$;
2. Calcule os parâmetros de $q_{-i}(f_i)$ segundo (4.6);
3. Calcule os parâmetros de $\hat{q}_i(f_i)$ segundo (4.7);
4. Atualize os parâmetros locais de t_i segundo (4.9);
5. Atualize os parâmetros de $q(f|X, Y)$ segundo (4.2);
6. Fixe um novo valor para i e repita os passos de 2 a 5 até a convergência de todos os parâmetros locais para um ponto fixo.

Apesar de não especificado, o ideal seria escolher o valor de i de forma sequencial, passando por todos os outros valores possíveis antes de repetir um. Os parâmetros $\tilde{Z}_i$, $\hat{Z}_i$ e Z_{EP} não precisam ser atualizados em cada iteração e podem ser deixados para serem calculados após a convergência do método, se desejados.

4.2 Convergência

No caso onde procuramos uma aproximação normal de $p(f|X,Y)$, como estamos aqui, é garantida a existência de pelo menos um ponto fixo para o algoritmo apresentado (veja (MINKA, 2001a)). Mas essa existência não é garantia de unicidade nem de convergência global do algoritmo. As escolhas de valores iniciais para os parâmetros locais, de função de covariância (e seus possíveis hiperparâmetros) e de likelihood têm um papel importante na convergência.

Uma observação relevante é que precisamos que as likelihoods $p(y_i|f_i)$ e a distribuição posterior $p(f|X,Y)$ possam ser bem aproximadas por distribuições gaussianas. Para demonstrar isso e avaliar o que estamos alcançando com este método, observe que

$$D_{KL}(P(f|X,Y)||Q(f|X,Y)) = \int \log\left(\frac{\frac{p(f|X)}{p(Y|X)}\prod_{i=1}^{n} p(y_i|f_i)}{\frac{p(f|X)}{Z_{EP}}\prod_{i=1}^{n} t_i(f_i|\tilde{Z}_i,\tilde{\mu}_i,\tilde{\Sigma}_i)}\right) p(f|X,Y)\,df$$

$$= \sum_{i=1}^{n}\int \log\left(\frac{p(y_i|f_i)}{t_i(f_i|\tilde{Z}_i,\tilde{\mu}_i,\tilde{\Sigma}_i)}\right) p(f|X,Y)\,df + \log\left(\frac{Z_{EP}}{p(Y|X)}\right)$$

$$= \sum_{i=1}^{n}\int \log\left(\frac{Z_i}{\hat{Z}_i}\frac{\frac{1}{Z_i}q_{-i}(f_i)p(y_i|f_i)}{\frac{1}{\hat{Z}_i}q_{-i}(f_i)t_i(f_i|\tilde{Z}_i,\tilde{\mu}_i,\tilde{\Sigma}_i)}\right)\frac{1}{Z_i}q_{-i}(f_i)p(y_i|f_i)\frac{\frac{C_i}{p(Y|X)}p_{-i}(f_i)p(y_i|f_i)}{\frac{1}{Z_i}q_{-i}(f_i)p(y_i|f_i)}\,df_i$$

$$+\log\left(\frac{Z_{EP}}{p(Y|X)}\right),$$

onde $p_{-i}(f_i) \propto \int p(f|X)\prod_{j=1,j\neq i}^{n} p(y_i|f_i)\,df_j$ e C_i é a constante de normalização de p_{-i}.

Supondo que estamos em um ponto fixo do método e que ele aproxima bem a distribuição posterior, temos que $\frac{C_i}{p(Y|X)}p_{-i}(f_i)p(y_i|f_i) \approx \frac{1}{Z_i}q_{-i}(f_i)p(y_i|f_i)$, $p(Y|X) \approx Z_{EP}$ e $Z_i \approx \hat{Z}_i$, e portanto

$$D_{KL}(P(f|X,Y)||Q(f|X,Y)) \approx \sum_{i=1}^{n} D_{KL}(Q_{-i}P_i||N(\hat{\mu}_i,\hat{\Sigma}_i)).$$

Como estamos em um ponto fixo, todos os parâmetros locais satisfazem as equação (4.7) simultaneamente, e consequentemente todas as divergências $D_{KL}(Q_{-i}P_i||N(\hat{\mu}_i,\hat{\Sigma}_i))$ são simultaneamente minimizadas. Portanto, estamos aproximadamente minimizando a divergência

de $Q(f|X,Y)$ para $P(f|X,Y)$ ao usar o ponto fixo conquistado convergindo o método de expectation propagation.

No caso de múltiplos pontos fixos, pode-se encarar cada um como um modelo distinto e usar alguma estratégia para seleção de modelos para eleger um.

4.3 Estimando Outputs de Novos Dados

Considere agora que queremos estimar o output de um input novo x^*. Pelo apresentado na seção 3.3, queremos ter a distribuição de $f^*|X,Y,x^*$. Pelas propriedades (A.2) e (A.4), temos que se $f|X,Y$ segue uma distribuição normal, então $f^*|X,Y,x^*$ também segue uma distribuição normal. Para isso, utilizamos a aproximação normal $p(f|X,Y) \approx q(f|X,Y) = \phi(f|\mu,\Sigma)$.

Observe que para um caso genérico de classificação por processos gaussianos temos, denotando $k_* = K(X,x^*)$,

$$\begin{aligned}\mathbb{E}_{f^*|X,Y,x^*}[f^*] &= \mathbb{E}_{f|X,Y}\left[\mathbb{E}_{f^*|X,x^*,f}[f^*]\right] = \mathbb{E}_{f|X,Y}\left[k_*^T K^{-1} f\right] \\ &= k_*^T K^{-1}\mathbb{E}_{f|X,Y}[f],\end{aligned}$$

e

$$\begin{aligned}var_{f^*|X,Y,x^*}(f^*) &= \mathbb{E}_{f|X,Y}[var_{f^*|X,x^*,f}(f^*)] + var_{f|X,Y}(\mathbb{E}_{f^*|X,x^*,f}[f^*]) \\ &= k(x^*,x^*) - k_*^T K^{-1} k_* + var_{f|X,Y}(k_*^T K^{-1} f) \\ &= k(x^*,x^*) - k_*^T K^{-1} k_* + k_*^T K^{-1} var_{f|X,Y}(f) K^{-1} k_*,\end{aligned}$$

lembrando que $\mathbb{E}_{f^*|X,x^*,f}[f^*] = k_*^T K^{-1} f$ e $var_{f^*|X,x^*,f}(f^*) = k(x^*,x^*) - k_*^T K^{-1} k_*$ pela equação (3.6).

Aplicando a aproximação $q(f|X,Y)$, temos, usando a igualdade (A.5), que

$$\begin{aligned}\mathbb{E}_{f^*|X,Y,x^*}[f^*] &= k_*^T K^{-1}\mu = k_*^T K^{-1}(K^{-1} + \tilde{\Sigma}^{-1})^{-1}\tilde{\Sigma}^{-1}\tilde{\mu} \\ &= k_*^T (K + \tilde{\Sigma})^{-1}\tilde{\mu},\end{aligned}$$

e

$$\begin{aligned}var_{f^*|X,Y,x^*}(f^*) &= k(x^*,x^*) - k_*^T(K^{-1} - K^{-1}(K^{-1} + \tilde{\Sigma}^{-1})^{-1}K^{-1})k_* \\ &= k(x^*,x^*) - k_*^T(K + \tilde{\Sigma})^{-1}k_*.\end{aligned}$$

Concluímos então que utilizando a aproximação normal encontrada pelo método de expectation propagation temos

$$p(f^*|X,Y,x^*) \approx q(f^*|X,Y,x^*) = \phi(f^*|k_*^T(K+\tilde{\Sigma})^{-1}\tilde{\mu},\ k(x^*,x^*) - k_*^T(K+\tilde{\Sigma})^{-1}k_*).$$

Essa distribuição é usada para calcular estimadores para $\pi_j(x^*)$ que, como discutido no final da seção 3.3, interpretamos como a probabilidade da x^* pertencer a classe j. Estes

estimadores são então usados como critério para classificar x^*. Calculamos os estimadores $\hat{\pi}_1(x^*), \hat{\pi}_2(x^*), \ldots, \hat{\pi}_C(x^*)$ segundo

$$\hat{\pi}_j(x^*) = \int p(j|f^*)q(f^*|X,Y,x^*)df^*. \tag{4.11}$$

Essa integral por sua vez pode ser calculado por alguma técnica de integração numérica ou integração de Monte Carlo, caso não possua uma expressão analítica (discutimos mais sobre essas técnicas na seção 4.5.1).

4.4 Marginal Likelihood

Como já comentado, temos que Z_{EP} é a aproximação da marginal likelihood $p(Y|X)$. Essa aproximação pode ser útil, já que a marginal likelihood tem usos na análise Bayesiana, a principal que vamos usar aqui sendo para a seleção de modelos usando o fator de Bayes, como apresentado na seção 2.2.3. Esse pensamento nos leva também a desejar escolher os hiperparâmetros da função de covariância tal que a marginal likelihood seja maximizada. Para isso, apresentamos também uma fórmula para as derivadas parciais de marginal log-likelihood em relação a esses hiperparâmetros.

Para uma expressão para Z_{EP}, lembre primeiramente que pela igualdade (4.1) temos

$$\begin{aligned} q(f|X,Y) &= \frac{1}{Z_{EP}} p(f|X) \prod_{i=1}^{n} t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i) \\ &= \frac{1}{Z_{EP}} \phi(f|0,K)\phi(f|\tilde{\mu},\tilde{\Sigma}) \prod_{i=1}^{n} \tilde{Z}_i = \phi(f|\mu,\Sigma)\frac{Z}{Z_{EP}} \prod_{i=1}^{n} \tilde{Z}_i, \end{aligned}$$

onde Z é, pela propriedade (A.4), dado por

$$Z = (2\pi)^{-\frac{nC}{2}} \det\left(K + \tilde{\Sigma}\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\,\tilde{\mu}^T \left(K + \tilde{\Sigma}\right)^{-1} \tilde{\mu}\right). \tag{4.12}$$

Como $q(f|X,Y)$ e $\phi(f|\mu,\Sigma)$ são distribuições, temos que ambas têm integral igual a um, portanto

$$\int q(f|X,Y)\, df = \int \phi(f|\mu,\Sigma)\frac{Z}{Z_{EP}} \prod_{i=1}^{n} \tilde{Z}_i\, df$$

$$Z_{EP} = Z \prod_{i=1}^{n} \tilde{Z}_i$$

$$\log Z_{EP} = -\frac{1}{2}\log\det\left(K + \tilde{\Sigma}\right) - \frac{1}{2}\,\tilde{\mu}^T \left(K + \tilde{\Sigma}\right)^{-1} \tilde{\mu} - \frac{nC}{2}\log(2\pi) + \sum_{i=1}^{n} \log \tilde{Z}_i. \tag{4.13}$$

Utilizando a equação (4.10) para $\tilde{Z}_i$, temos que

$$\begin{aligned} \log Z_{EP} &= -\frac{1}{2}\log\det\left(K + \tilde{\Sigma}\right) - \frac{1}{2}\,\tilde{\mu}^T \left(K + \tilde{\Sigma}\right)^{-1} \tilde{\mu} + \frac{1}{2}\sum_{i=1}^{n} \log\det\left(\Sigma_{-i} + \tilde{\Sigma}_i\right) \\ &+ \frac{1}{2}\sum_{i=1}^{n} (\mu_{-i} - \tilde{\mu}_i)^T \left(\Sigma_{-i} + \tilde{\Sigma}_i\right)^{-1} (\mu_{-i} - \tilde{\mu}_i) + \sum_{i=1}^{n} \log \hat{Z}_i. \end{aligned}$$

Pelas igualdades (4.2), (4.6), podemos aplicar o lema 3 do apêndice A.3, assim reescrevendo a expressão para $\log Z_{EP}$ como

$$\begin{aligned}\log Z_{EP} &= -\frac{1}{2}\tilde{\mu}^T\tilde{\Sigma}^{-1}\tilde{\mu} + \frac{1}{2}\mu^T\Sigma^{-1}\mu - \frac{1}{2}\log\det\left(\tilde{\Sigma}^{-\frac{1}{2}}K\tilde{\Sigma}^{-\frac{1}{2}} + I\right) - \frac{1}{2}\log\det(\tilde{\Sigma}) \\ &+\frac{1}{2}\sum_{i=1}^{n}\left(\log\det(\Sigma_{-i}) + \log\det(\tilde{\Sigma}_i) - \log\det(\Sigma_i)\right) \\ &+\frac{1}{2}\sum_{i=1}^{n}\left(\tilde{\mu}_i^T\tilde{\Sigma}_i^{-1}\tilde{\mu}_i + \mu_{-i}^T\Sigma_{-i}^{-1}\mu_{-i} - \mu_i^T\Sigma_i^{-1}\mu_i\right) + \sum_{i=1}^{n}\log\hat{Z}_i.\end{aligned}$$

Como $\tilde{\Sigma}$ é a matriz bloco diagonal com blocos $\tilde{\Sigma}_i$ permutada de forma simétrica, com $\tilde{\mu}$ sendo os valores de cada $\tilde{\mu}_i$ ordenados de forma correspondente, e também que $\tilde{\Sigma}$ e cada $\tilde{\Sigma}_i$ é positiva definida, temos que $\tilde{\mu}^T\tilde{\Sigma}^{-1}\tilde{\mu} = \sum_{i=1}^{n}\tilde{\mu}_i^T\tilde{\Sigma}_i^{-1}\tilde{\mu}_i$ e $\log\det(\tilde{\Sigma}) = \sum_{i=1}^{n}\log\det(\tilde{\Sigma}_i)$. Lembrando também que $\mu = \Sigma\tilde{\Sigma}^{-1}\tilde{\mu}$, chegamos na forma final para a expressão da aproximação da marginal log-likelihood pelo método de expectation propagation aqui apresentado:

$$\begin{aligned}\log Z_{EP} &= \frac{1}{2}\mu^T\tilde{\Sigma}^{-1}\tilde{\mu} - \frac{1}{2}\log\det\left(\tilde{\Sigma}^{-\frac{1}{2}}K\tilde{\Sigma}^{-\frac{1}{2}} + I\right) + \frac{1}{2}\sum_{i=1}^{n}\left(\mu_{-i}^T\Sigma_{-i}^{-1}\mu_{-i} + \log\det\left(\Sigma_{-i}\right)\right) \\ &-\frac{1}{2}\sum_{i=1}^{n}\left(\mu_i^T\Sigma_i^{-1}\mu_i + \log\det\left(\Sigma_i\right)\right) + \sum_{i=1}^{n}\log\hat{Z}_i,\end{aligned} \tag{4.14}$$

como apontado em (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013).

No caso de a função de covariância ter hiperparâmetros, o trabalho de (SEEGER, 2005) mostra que apenas termos diretamente dependentes dos hiperparâmetros são relevantes para as derivadas parciais em relação a tais (neste caso, o termo Z definido em (4.12)), já que as contribuições indiretas dos outros termos se cancelam (em outra palavras, podemos considerar os parâmetros locais $\tilde{Z}_i$ como constantes), supondo que estamos em um ponto fixo do método de expectation propagation. Neste caso, para o $j-$ésimo hiperparâmetro θ_j, temos, usando a igualdade (4.13),

$$\begin{aligned}\frac{\partial[\log Z_{EP}]}{\partial\theta_j} &= \frac{\partial}{\partial\theta_j}\left[-\frac{1}{2}\log\det\left(K+\tilde{\Sigma}\right) - \frac{1}{2}\tilde{\mu}^T\left(K+\tilde{\Sigma}\right)^{-1}\tilde{\mu}\right] \\ &= -\frac{1}{2}\mathrm{tr}\left(\left(K+\tilde{\Sigma}\right)^{-1}\frac{\partial K}{\partial\theta_j}\right) + \frac{1}{2}\tilde{\mu}^T\left(K+\tilde{\Sigma}\right)^{-1}\frac{\partial K}{\partial\theta_j}\left(K+\tilde{\Sigma}\right)^{-1}\tilde{\mu} \\ &= \frac{1}{2}\mathrm{tr}\left(\left(EE^T - \left(K+\tilde{\Sigma}\right)^{-1}\right)\frac{\partial K}{\partial\theta_j}\right),\end{aligned}$$

onde $E = \left(K+\tilde{\Sigma}\right)^{-1}\tilde{\mu}$.

Essas derivadas parciais podem ser usadas para achar hiperparâmetros que maximizam localmente a marginal log-likelihood do modelo por meio de algum método de otimização, como por exemplo um método de descida de gradiente ou gradiente conjugado (NOCEDAL; WRIGHT, 2006).

4.5 Algoritmo Nested Expectation Propagation

Nessa seção, apresentaremos um algoritmo para classificação por processos gaussianos usando o método de expectation propagation com uma likelihood predeterminada. O algoritmo apresentado aqui é oriundo do artigo *Nested Expectation Propagation for Gaussian Process Classification with a Multinomial Probit Likelihood* (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013). Pode-se também entender esse algoritmo como uma generalização para o caso multi-classes do algoritmo para classificação binária por expectation propagation apresentado em *Gaussian processes for machine learning* (WILLIAMS; RASMUSSEN, 2006, pg. 52).

Na seção 4.1 foi discutido o procedimento geral para o método de expectation propagation com uma likelihood qualquer. Definiremos agora uma likelihood especifica e apresentaremos métodos para o cálculo dos parâmetros $\hat{\Sigma}_i$ e $\hat{\mu}_i$ dada essa likelihood. Apresentaremos também detalhes de implementação e reuniremos tudo em pseudo-códigos mais adiante. O mesmo será feito nas seções 4.5.1 e 4.5.2, onde apresentaremos as especifidades do método com a likelihood escolhida e detalhes de implementação para a inferência de novos outputs e o cálculo da marginal likelihood, como discutido nas seções 4.3 e 4.4 respectivamente.

O modelo que seguiremos toma, além do prior $f \sim \mathcal{GP}_C(0, [k_1, \ldots, k_C])$, a função *multinomial probit*, definida em (3.7), como modelo para a likelihood $p(j|f)$. A escolha de tal função para a likelihood deve-se ao seu formato em específico, que é aproveitado para a implementação do algoritmo aqui apresentado, como ficará evidente ao decorrer dessa seção.

Para aplicar o método, precisamos conseguir estimar os parâmetros de $\hat{q}_i(f_i)$ segundo (4.7). Para isso, observe a forma da pdf de $Q_{-i}P_i$ neste caso:

$$\begin{aligned}
\frac{1}{Z_i} q_{-i}(f_i) p(y_i|f_i) &= \frac{1}{Z_i} \phi(f_i|\mu_{-i}, \Sigma_{-i}) \mathbb{E}_{u\sim N(0,1)} \left[\prod_{k=1, k\neq y_i}^{C} \Phi(u + f_i^{y_i} - f_i^k) \right] \\
&= \int \frac{1}{Z_i} \phi(f_i|\mu_{-i}, \Sigma_{-i}) \phi(u|0,1) \prod_{k=1, k\neq y_i}^{C} \Phi(u + f_i^{y_i} - f_i^k)\, du \\
&= \int \frac{1}{Z_i} \phi\left(\left[\begin{matrix} f_i \\ u \end{matrix} \right] \middle| \left[\begin{matrix} \mu_{-i} \\ 0 \end{matrix} \right], \left[\begin{matrix} \Sigma_{-i} & 0 \\ 0 & 1 \end{matrix} \right] \right) \prod_{k=1, k\neq y_i}^{C} \Phi(u + f_i^{y_i} - f_i^k)\, du,
\end{aligned}$$

onde f_i^j representa a j-ésima entrada de f_i e $\Phi(x)$ é a função de distribuição acumulada de uma distribuição normal padrão. Denominando $w_i = [f_i \; u]^T$ e $b_{j,k} = e_{C+1} + e_j - e_k$, temos que podemos reescrever a distribuição estendida que montamos dentro da integral como

$$q_{w_i}(w_i) = \frac{1}{Z_i} \phi(w_i|\mu_{w_i}, \Sigma_{w_i}) \prod_{k=1, k\neq y_i}^{C} \Phi(w_i^T b_{y_i,k}). \tag{4.15}$$

Observe que se q_{w_i} seguisse a forma de uma densidade gaussiana, poderíamos integrar sobre u usando a propriedade de marginalização (A.2), onde conseguiríamos outra distribuição normal e obteríamos seus momentos de forma direta, o qual é o nosso objetivo aqui. Isso nos motiva a achar uma aproximação normal para q_{w_i}.

$$\hat{q}_{w_i} = \phi(w_i|\hat{\mu}_{w_i}, \hat{\Sigma}_{w_i}) \approx q_{w_i}(w_i).$$

Para alcançar tal aproximação, utilizamos algoritmo por expectation propagation apresentado em (WILLIAMS; RASMUSSEN, 2006, pg. 52) com as alterações apresentadas em (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013).

Começamos usando a transformação $g_i = B_i^T w_i$, onde $B_i = [b_{y_i,k}]_{k=1,k\neq y_i}^{C}$, obtendo

$$q_{g_i}(g_i) = \frac{1}{Z_i}\phi(g_i|B_i^T\mu_{w_i}, B_i^T\Sigma_{w_i}B_i)\prod_{k=1}^{C-1}\Phi(g_i^k).$$

Nessa forma, podemos aplicar o algoritmo considerando $\phi(g_i|B_i^T\mu_{w_i}, B_i^T\Sigma_{w_i}B_i)$ como prior e a função *probit* $\Phi(g_i^k)$ como likelihood. Neste caso, conseguimos uma fórmula fechada para calcular os parâmetros (4.7)(WILLIAMS; RASMUSSEN, 2006, pg. 52). Para obter melhores resultados numéricos e simplificar as equações, reparametrizamos as aproximações locais de $\Phi(g_i^k)$ usando os parâmetros naturais da distribuição normal (ver A.1), i.e.,

$$\Phi(g_i^k) \approx t_{i,k}(g_i^k|\tilde{Z}_{i,k}, \beta_{i,k}, \alpha_{i,k}) = \tilde{Z}_{i,k}\phi(g_i^k|\alpha_{i,k}^{-1}\beta_{i,k}, \alpha_{i,k}^{-1})$$

com $\alpha_{i,k}$ sendo a precisão e $\beta_{i,k}$ sendo a locação. Portanto, temos que os passos para estimar essa aproximação, usando fórmulas derivadas em (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013) e em (WILLIAMS; RASMUSSEN, 2006, pg. 58) para atualizar os parâmetros nos passos 4 e 6, são:

1. Fixar $k \in \{1, 2, \ldots, C\}$ com $k \neq y_i$;

2. Calcular os parâmetros da distribuição marginal de g_i^k,

$$\sigma_{i,k}^2 = b_{y_i,k}^T\hat{\Sigma}_{w_i}b_{y_i,k} \quad \text{e} \quad \mu_{i,k} = b_{y_i,k}^T\hat{\mu}_{w_i};$$

3. Calcular os parâmetros da distribuição *cavity*,

$$\sigma_{-i,k}^2 = ((\sigma_{i,k}^2)^{-1} - \alpha_{i,k})^{-1}, \quad \text{e} \quad \mu_{-i,k} = \sigma_{-i,k}^2((\sigma_{i,k}^2)^{-1}\mu_{i,k} - \beta_{i,k});$$

4. Calcular os parâmetros (4.7),

$$\hat{\mu}_{i,k} = \mu_{-i,k} + \frac{\sigma_{-i,k}^2\phi(z_{i,k}|0,1)}{\Phi(z_{i,k})\sqrt{1+\sigma_{-i,k}^2}} \quad \text{e} \quad \hat{\sigma}_{i,k}^2 = \sigma_{-i,k}^2 - \frac{(\sigma_{-i,k}^2)^2\phi(z_{i,k}|0,1)}{\Phi(z_{i,k})(1+\sigma_{-i,k}^2)}\left(z_{i,k} + \frac{\phi(z_{i,k}|0,1)}{\Phi(z_{i,k})}\right);$$

   onde $z_{i,k} = \dfrac{\mu_{-i,k}}{\sqrt{1+\sigma_{-i,k}^2}}$. Temos também que $\hat{Z}_{i,k} = \Phi(z_{i,k})$;

5. Atualizar os parâmetros locais,

$$\alpha_{i,k}^{new} = (\hat{\sigma}_{i,k}^2)^{-1} - (\sigma_{-i,k}^2)^{-1}, \quad \text{e} \quad \beta_{i,k}^{new} = (\hat{\sigma}_{i,k}^2)^{-1}\hat{\mu}_{i,k} - (\sigma_{-i,k}^2)^{-1}\mu_{-i,k},$$

   calculando também $\Delta\alpha_{i,k} = \alpha_{i,k}^{new} - \alpha_{i,k}$ e $\Delta\beta_{i,k} = \beta_{i,k}^{new} - \beta_{i,k}$;

6. Atualizar os parâmetros de $\hat{q}_{w_i}(w_i)$,

$$\hat{\Sigma}_{w_i}^{new} = \hat{\Sigma}_{w_i} - \left(\frac{\Delta\alpha_{i,k}}{1+\Delta\alpha_{i,k}\sigma_{i,k}^2}\right) s_{i,k}s_{i,k}^T \quad \text{e} \quad \hat{\mu}_{w_i}^{new} = \hat{\mu}_{w_i} + \left(\frac{\Delta\beta_{i,k} - \Delta\alpha_{i,k}\mu_{i,k}}{1+\Delta\alpha_{i,k}\sigma_{i,k}^2}\right) s_{i,k},$$

onde $s_{i,k} = \hat{\Sigma}_{w_i} b_{y_i,k}$;

7. Repetir os passos de 1 a 6 até a convergência de todos os parâmetros locais para um ponto fixo.

Com isso podemos montar um pseudo-código para o loop interno (quando estamos estimando $\hat{q}_{w_i}(w_i)$) do algoritmo *Nested Expectation Propagation*, exposto no Algoritmo 1, alterando algumas fórmulas para atualizar os parâmetros de forma mais eficiente e introduzindo variáveis de suporte para aproveitar valores que se repetem entre as fórmulas.

Algoritmo 1: Nested Expectation Propagation - Loop interno

Input: y_i e Valores iniciais para $\hat{\Sigma}_{w_i}$, $\hat{\mu}_{w_i}$, α_i, β_i

repita

  para $k \in \{1, 2, \dots C\}$, $k \neq y_i$, **faça**

    Sendo $b_{j,k}$ definido como para (4.15)

    $\sigma_{i,k}^2 = b_{y_i,k}^T \hat{\Sigma}_{w_i} b_{y_i,k}, \quad \mu_{i,k} = b_{y_i,k}^T \hat{\mu}_{w_i}$

    $\tau_{-i,k} = (\sigma_{i,k}^2)^{-1} - \alpha_{i,k}, \quad \nu_{-i,k} = (\sigma_{i,k}^2)^{-1}\mu_{i,k} - \beta_{i,k}$

    $\tau_{i,k}^{supp} = \sqrt{\tau_{-i,k}(1+\tau_{-i,k})}, \quad z_{i,k} = \frac{\nu_{-i,k}}{\tau_{i,k}^{supp}}, \quad \rho = \frac{\phi(z_{i,k}|0,1)}{\Phi(z_{i,k})\tau_{i,k}^{supp}}$

    $\hat{\tau}_{i,k} = \left(\tau_{-i,k}^{-1} - \rho^2 - \frac{z_{i,k}\rho}{\tau_{i,k}^{supp}}\right)^{-1}, \quad \hat{\nu}_{i,k} = \left(\frac{\nu_{-i,k}}{\tau_{-i,k}} + \rho\right)\hat{\tau}_{i,k}$

    $\Delta\alpha_{i,k} = \hat{\tau}_{i,k} - \tau_{-i,k} - \alpha_{i,k}, \quad \alpha_{i,k} = \hat{\tau}_{i,k} - \tau_{-i,k}$

    $\Delta\beta_{i,k} = \hat{\nu}_{i,k} - \nu_{-i,k} - \beta_{i,k}, \quad \beta_{i,k} = \hat{\nu}_{i,k} - \nu_{-i,k}$

    $s_{i,k} = \hat{\Sigma}_{w_i} b_{y_i,k}$

    $\hat{\Sigma}_{w_i} = \hat{\Sigma}_{w_i} - \left(\frac{\Delta\alpha_{i,k}}{1+\Delta\alpha_{i,k}\sigma_{i,k}^2}\right) s_{i,k}s_{i,k}^T, \quad \hat{\mu}_{w_i} = \hat{\mu}_{w_i} + \left(\frac{\Delta\beta_{i,k} - \Delta\alpha_{i,k}\mu_{i,k}}{1+\Delta\alpha_{i,k}\sigma_{i,k}^2}\right) s_{i,k}$

até α_i, β_i *convergirem*

retorna α_i, β_i

Novamente, para obter resultados numéricos melhores e simplificar as equações, reparametrizamos as aproximações locais $t_i(f_i|\tilde{Z}_i, \tilde{\mu}_i, \tilde{\Sigma}_i)$ usando os parâmetros naturais para as distribuições normais (ver A.1) de forma que temos $t_i(f_i|\tilde{Z}_i, \tilde{\nu}_i, \tilde{\mathcal{T}}_i) = \tilde{Z}_i\phi(f_i|\tilde{\mathcal{T}}_i^{-1}\tilde{\nu}_i, \tilde{\mathcal{T}}_i^{-1})$ daqui em diante.

O trabalho de (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013) também apresenta fórmulas para atualizar os parâmetros $\tilde{\nu}_i$ e $\tilde{\mathcal{T}}_i$ diretamente dos valores atualizados de α_i, β_i, presentes no Algoritmo 2. Ele também argumenta que experimentalmente se observa uma convergência mais rápida se incorporados os valores de $\alpha_{i,k}$ e $\beta_{i,k}$ da iteração anterior do loop externo (quando

estamos estimando $q(f|X,Y)$) como valores iniciais para a loop interno. Para isso, os valores iniciais de $\hat{\Sigma}_{w_i}$ e $\hat{\mu}_{w_i}$ têm que estar de acordo com os valores de Σ_{w_i}, μ_{w_i}, $\alpha_{i,k}$ e $\beta_{i,k}$ segundo, pela propriedade (A.4),

$$\hat{\Sigma}_{w_i} = \left(\Sigma_{w_i}^{-1} + B_i \text{diag}(\tilde{\alpha}_i) B_i^T\right)^{-1} \quad \text{e} \quad \hat{\mu}_{w_i} = \hat{\Sigma}_{w_i}\left(\Sigma_{w_i}^{-1}\mu_{w_i} + B_i\tilde{\beta}_i\right), \tag{4.16}$$

onde $\tilde{\alpha}_i$ e $\tilde{\beta}_i$ são vetores com $C-1$ entradas contendo os valores $\alpha_{i,k}$ e $\beta_{i,k}$ ordenados em ordem crescente dos índices $k \in \{1,2,\ldots,C\}$, $k \neq y_i$. Definimos também α_i e β_i como vetores com C entradas onde $\alpha_i^k = \alpha_{i,k}$ e $\beta_i^k = \beta_{i,k}$, para $k \neq y_i$, e $\alpha_i^{y_i} = 1$ e $\beta_i^{y_i} = 0$.

Utilizando o lema de inversão de matrizes (A.5), reescrevemos a equação (4.16) como

$$\hat{\Sigma}_{w_i} = \Sigma_{w_i} - \Sigma_{w_i} B_i (B_i^T \Sigma_{w_i} B_i + \text{diag}(\tilde{\alpha}_i)^{-1})^{-1} B_i^T \Sigma_{w_i}$$

$$\hat{\Sigma}_{w_i} = \Sigma_{w_i} - \Sigma_{w_i} B_i \left(B_i^T \Sigma_{w_i} B_i + \text{diag}\left(\tilde{\alpha}_i^{-\frac{1}{2}}\right)\text{diag}\left(\tilde{\alpha}_i^{-\frac{1}{2}}\right)\right)^{-1} B_i^T \Sigma_{w_i}$$

$$\hat{\Sigma}_{w_i} = \Sigma_{w_i} - \Sigma_{w_i} B_i \left(\text{diag}\left(\tilde{\alpha}_i^{-\frac{1}{2}}\right)\left(\text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right) B_i^T \Sigma_{w_i} B_i \text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right) + I\right)\text{diag}\left(\tilde{\alpha}_i^{-\frac{1}{2}}\right)\right)^{-1} B_i^T \Sigma_{w_i}$$

$$\hat{\Sigma}_{w_i} = \Sigma_{w_i} - \Sigma_{w_i} B_i \text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right) J^{-1} \text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right) B_i^T \Sigma_{w_i},$$

onde

$$J = \left(\text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right) B_i^T \Sigma_{w_i} B_i \text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right) + I\right).$$

Fatoramos $\text{diag}(\tilde{\alpha}_i)$ para conseguir J, uma matriz simétrica positiva definida com os autovalores limitados inferiormente por um, o que garante estabilidade numérica da fórmula e permite usarmos a fatoração Cholesky de tal matriz para aproveitar a simetria da expressão e diminuir o custo computacional, como utilizado no algoritmo 2. Para $\hat{\mu}_{w_i}$, temos

$$\hat{\mu}_{w_i} = \mu_{w_i} - \Sigma_{w_i} B_i \text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right) J^{-1} \text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right) B_i^T \mu_{w_i} + \hat{\Sigma}_{w_i} B_i \tilde{\beta}_i.$$

Precisamos agora de alguma forma de atualizar Σ de forma mais eficiente depois de atualizar os parâmetros locais $\tilde{\mathcal{T}}_i$. A equação (4.2) nos informa que

$$\Sigma = \left(K^{-1} + \tilde{\mathcal{T}}\right)^{-1},$$

onde $\tilde{\mathcal{T}} = \sum_{i=1}^n (\tilde{\mathcal{T}}_i \otimes e_i e_i^T)$ (equivalentemente vale que $\tilde{\mathcal{T}} = \tilde{\Sigma}^{-1}$). Infelizmente, K geralmente é mal condicionada e portanto inverter ela não é numericamente estável e algo a ser evitado. Por isso, usamos o lema de inversão de matrizes (A.5) para obter

$$\Sigma = K - K\left(K + \tilde{\mathcal{T}}^{-1}\right)^{-1} K.$$

Reescrevemos

$$\left(K + \tilde{\mathcal{T}}^{-1}\right)^{-1} = \left(\tilde{\mathcal{T}}^{-\frac{1}{2}}\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)\tilde{\mathcal{T}}^{-\frac{1}{2}}\right)^{-1} = \tilde{\mathcal{T}}^{\frac{1}{2}}\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}},$$

onde $\tilde{\mathcal{T}}^{\frac{1}{2}}$ é a matriz simétrica positiva definida tal que seu quadrado é $\tilde{\mathcal{T}}$. Desta forma temos novamente uma matriz simétrica positiva definida com os autovalores limitados inferiormente por um, o que garante novamente a estabilidade numérica e permite usar o fator Cholesky

$L = \text{Cholesky}\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)$ para aproveitar a simetria da expressão e diminuir o custo computacional na forma de

$$K\left(K + \tilde{\mathcal{T}}^{-1}\right)^{-1} K = K\tilde{\mathcal{T}}^{\frac{1}{2}}\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)^{-1} \tilde{\mathcal{T}}^{\frac{1}{2}} K = K\tilde{\mathcal{T}}^{\frac{1}{2}}\left(LL^T\right)^{-1} \tilde{\mathcal{T}}^{\frac{1}{2}} K$$

$$= K\tilde{\mathcal{T}}^{\frac{1}{2}}\left(L^{-1}\right)^T L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}} K = \left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}} K\right)^T \left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}} K\right),$$

já que assim só é preciso calcular uma vez a matriz $\left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}} K\right)$.

Portanto, temos que nossa equação para a atualização de Σ é

$$\Sigma^{new} = K - \left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}} K\right)^T \left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}} K\right).$$

Para μ, temos, pela equação (4.2), que atualizamos esse parâmetro pela equação

$$\mu^{new} = \Sigma^{new}\tilde{\nu}.$$

Como essas atualizações são muito custosa computacionalmente, optamos por atualizar os valores de μ e Σ depois de percorrer todos os pontos do dataset.

Finalmente, para os parâmetros da distribuição *cavity* (4.6), temos, novamente usando (A.5), que

$$\Sigma_{-i} = \Sigma_i - \Sigma_i(\Sigma_i - \tilde{\mathcal{T}}_i^{-1})^{-1}\Sigma_i = \Sigma_i - \Sigma_i(\tilde{\mathcal{T}}_i\Sigma_i - I)^{-1}\tilde{\mathcal{T}}_i\Sigma_i,$$

e

$$\mu_{-i} = \mu_i - \Sigma_i(\tilde{\mathcal{T}}_i\Sigma_i - I)^{-1}\tilde{\mathcal{T}}_i\mu_i - \Sigma_{-i}\tilde{\nu}_i.$$

É importante observar que $\tilde{\mathcal{T}}_i\Sigma_i - I$ não é simétrica e não é necessariamente positiva definida. Inclusive, não podemos seguir o mesmo modelo apresentado nas equações anteriores pois não podemos garantir que $\tilde{\mathcal{T}}_i^{\frac{1}{2}}\Sigma_i\tilde{\mathcal{T}}_i^{\frac{1}{2}} - I$ ou $I - \tilde{\mathcal{T}}_i^{\frac{1}{2}}\Sigma_i\tilde{\mathcal{T}}_i^{\frac{1}{2}}$ sejam positivas definidas. A opção que escolhemos seguir é de resolver os sistemas lineares $(\tilde{\mathcal{T}}_i\Sigma_i - I)X = \tilde{\mathcal{T}}_i$ para obter $X = (\tilde{\mathcal{T}}_i\Sigma_i - I)^{-1}\tilde{\mathcal{T}}_i$.

Introduzindo variáveis de suporte para aproveitar aproveitar expressões repetidas nas fórmulas apresentadas, reunimos todos os passos e montamos um pseudo-código para o algoritmo *Nested Expectation Propagation* no Algoritmo 2.

Nos casos testados ao longo deste trabalho se mostrou benéfico adicionar uma constante de regularização a cada $\tilde{\mathcal{T}}_i$ cada vez que os atualizamos, por exemplo algo como λI com λ pequeno (algo como $\lambda = 10^{-10}$), aumentando a estabilidade numérica. É importante ter em mente que essa constante precisa que ser uma matriz simétrica positiva definida para manter as propriedades de $\tilde{\mathcal{T}}_i$.

4.5.1 Estimando Outputs

A seção 4.3 apresenta como estimar outputs de novos inputs com a aproximação normal calculada pelo método. Para a implementação das equações apresentadas, é importante primeiramente observar novamente que é mais benéfico numericamente reescrevermos

$$M = \left(K + \tilde{\Sigma}\right)^{-1} = \left(K + \tilde{\mathcal{T}}^{-1}\right)^{-1} = \tilde{\mathcal{T}}^{\frac{1}{2}}\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}},$$

Algoritmo 2: Nested Expectation Propagation

Input: K (matriz de covariância), Y (outputs do dataset), *likelihood* =Falso (Definir como verdeiro caso o cálculo da log-marginal likelihood seja desejado)

$\tilde{\nu} = 0$, $\tilde{\mathcal{T}}_i = 0$, $\Sigma = K$, $\mu = 0$, $\alpha_i = 0$, $\beta_i = 0$, $\forall i \in \{1, 2, \ldots, n\}$

$\alpha_{i,y_i} = 1$, $\forall i \in \{1, 2, \ldots, n\}$

repita

  para $i \in \{1, 2, \ldots, n\}$ **faça**

    $V = \Sigma_i(\tilde{\mathcal{T}}_i\Sigma_i - I)^{-1}\tilde{\mathcal{T}}_i$

    $\Sigma_{-i} = \Sigma_i - V\Sigma_i$

    $\mu_{-i} = \mu_i - V\mu_i - \Sigma_{-i}\tilde{\nu}_i$

    Montar Σ_{w_i} e μ_{w_i} de acordo com (4.15) e B_i, $\tilde{\alpha}_i$ e $\tilde{\beta}_i$ como definido para (4.16)

    $B_{\alpha_i} = B_i \text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right)$

    $L = \text{Cholesky}\left(I + B_{\alpha_i}^T \Sigma_{w_i} B_{\alpha_i}\right)$

    $V = L^{-1}\left(B_{\alpha_i}^T \Sigma_{w_i}\right)$

    $\hat{\Sigma}_{w_i} = \Sigma_{w_i} - V^T V$

    $\hat{\mu}_{w_i} = \mu_{w_i} - V^T\left(L^{-1}B_{\alpha_i}^T \mu_{w_i}\right) + \hat{\Sigma}_{w_i} B_i \tilde{\beta}_i$

    $[\alpha_i, \beta_i]$ =Loop interno(y_i, $\hat{\Sigma}_{w_i}$, $\hat{\mu}_{w_i}$, α_i, β_i) (Algoritmo 1)

    $\tilde{\mathcal{T}}_i = \text{diag}\left(\alpha_i\right) - (\text{sum}(\alpha_i))^{-1}\alpha_i\alpha_i^T$

    $\tilde{\nu}_i = \dfrac{\text{sum}(\beta_i)}{\text{sum}(\alpha_i)}\alpha_i - \beta_i$

  Montar $\tilde{\mathcal{T}} = \sum_{i=1}^{n}(\tilde{\mathcal{T}}_i \otimes e_i e_i^T)$, onde e_i é o $i-$ésimo vetor da base canônica de $\mathbb{R}^C$

  $L = \text{Cholesky}\left(\tilde{\mathcal{T}}^{\frac{1}{2}}\Sigma\tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)$

  $V = L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}}\Sigma$

  $\Sigma = \Sigma - V^T V$

  $\mu = \Sigma\tilde{\nu}$

até $\tilde{\nu}$, $\tilde{\mathcal{T}}_i$ *convergirem,* $\forall i \in \{1, 2, \ldots, n\}$

se *likelihood=Verdadeiro* **então**

  $\log Z_{EP}$=Log-marginal likelihood$(K, \Sigma, \mu, \tilde{\nu}, \tilde{\mathcal{T}}_i, \alpha_i, \beta, \forall i \in \{1, \ldots, n\})$ (Algoritmo 4)

  retorna $\log Z_{EP}$, $\tilde{\nu}$, $\tilde{\mathcal{T}}_i$, $\forall i \in \{1, 2, \ldots, n\}$

retorna $\tilde{\nu}$, $\tilde{\mathcal{T}}_i$, $\forall i \in \{1, 2, \ldots, n\}$

pois assim temos uma matriz positiva definida com os autovalores limitados inferiormente por um, o que garante estabilidade numérica e permite usarmos o fator Cholesky $L = \text{Cholesky}\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)$ para aproveitar a simetria da expressão e diminuir o custo computacional. Assim temos

$$\begin{aligned} M &= \tilde{\mathcal{T}}^{\frac{1}{2}} \left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)^{-1} \tilde{\mathcal{T}}^{\frac{1}{2}} = \tilde{\mathcal{T}}^{\frac{1}{2}} \left(LL^T\right)^{-1} \tilde{\mathcal{T}}^{\frac{1}{2}} = \tilde{\mathcal{T}}^{\frac{1}{2}} \left(L^{-1}\right)^T L^{-1} \tilde{\mathcal{T}}^{\frac{1}{2}} \\ &= \left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}}\right)^T \left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}}\right) \end{aligned}.$$

Com isso, no caso da variância de $f^*|X, Y, x^*$, escrevemos ela como

$$\Sigma^* = var_{f^*|X,Y,x^*}(f^*) = k(x^*, x^*) - k_*^T M k_*,$$

o que é computacionalmente barato para cada x^* considerando que já calculamos previamente a matriz M.

Já para $E = \left(K + \tilde{\Sigma}\right)^{-1} \tilde{\mu}$, podemos reescrever-lo, com ajuda da identidade (A.5), como

$$\begin{aligned} \left(K + \tilde{\Sigma}\right)^{-1} \tilde{\mu} &= \left(K + \tilde{\mathcal{T}}^{-1}\right)^{-1} \tilde{\mathcal{T}}^{-1} \tilde{\nu} = \left(\tilde{\mathcal{T}} - \tilde{\mathcal{T}} \left(K^{-1} + \tilde{\mathcal{T}}\right)^{-1} \tilde{\mathcal{T}}\right) \tilde{\mathcal{T}}^{-1} \tilde{\nu} \\ &= \tilde{\nu} - \left(K^{-1}\tilde{\mathcal{T}}^{-1} + I\right)^{-1} \tilde{\nu} = \tilde{\nu} - \left(K + \tilde{\mathcal{T}}^{-1}\right)^{-1} K\tilde{\nu} \\ &= \tilde{\nu} - \left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}}\right)^T \left(L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}}\right) K\tilde{\nu} = \tilde{\nu} - MK\tilde{\nu}, \end{aligned}$$

o que pode ser obtido de forma direta considerando que já temos K e M calculadas. Neste caso, temos que o valor esperado de $f^*|X, Y, x^*$ toma a forma de

$$\mu^* = \mathbb{E}_{f^*|X,Y,x^*}[f^*] = k_*^T E.$$

Para estimar $\hat{\pi}_j(x^*)$ (4.11), utilizaremos um método de integração de Monte Carlo por causa da dimensão do domínio de integração, uma vez que no caso onde temos três classes (o menor número de classes para o qual o algoritmo apresentado aqui é relevante) temos que o domínio de integração tem dimensão quatro, já que

$$\begin{aligned} \hat{\pi}_j(x^*) &= \int p(j|f^*)\phi(f^*|\mu^*, \Sigma^*)df^* \\ &= \iiiint_{\mathbb{R}^4} \left(\prod_{k=1,k\neq j}^{3} \Phi(u + f_*^j - f_*^k)\right) \phi\left(f^*|\mu^*, \Sigma^*\right) \phi(u|0,1)\, du\, df_*^1\, df_*^2\, df_*^3, \end{aligned}$$

onde f_*^k é a k-ésima entrada de f^*. Nestes casos, métodos de Monte Carlo se apresentam mais atrativos que métodos de integração numérica por quadratura, até mesmo em relação a quadraturas específicas para valores esperados de distribuições normais, como a quadratura de Gauss–Hermite, uma vez que se precisa de malhas com muitos pontos para conseguir uma boa aproximação, o que é computacionalmente muito custoso ou até mesmo inviável em dimensões maiores.

O método de integração de Monte Carlo na sua formulação mais básica para estimar $\theta = \mathbb{E}_P[f(X)]$ consiste em gerar um conjunto de n_0 amostras independentes $\{x_i\}_{i=1}^{n_0}$ igualmente distribuídas de acordo com P e definir

$$\hat{\theta}_{n_0} = \frac{1}{n_0} \sum_{i=1}^{n_0} f(x_i)$$

como o estimador de θ. É interessante observar que $\hat{\theta}_{n_0} \to \theta$ $\mathbb{P}$-a.s. quando $n_0 \to \infty$ pela lei forte dos grandes números, e que temos também que $var_P(\hat{\theta}_{n_0}) = \dfrac{var_P(f(X))}{n_0}$. Uma discussão mais detalhada do método e demostrações para resultados apresentados podem ser encontradas em (RIZZO, 2019, pg. 119).

Para diminuir a variância do método e conseguir uma melhor aproximação com o mesmo número de pontos, utilizaremos variáveis de controle. Essa técnica consiste em escolher funções $g_1, \ldots, g_c$ e constantes $\beta_0, \ldots, \beta_c$ tal que conseguimos (ou é mais conveniente) calcular $\mathbb{E}_P[g_i(X)]$ para cada $i = 1, \ldots, c$. Com isso, reescrevemos θ como

$$\begin{aligned}\theta = \mathbb{E}_P[f(X)] &= \mathbb{E}_P\left[f(X) - \beta_0 - \sum_{k=1}^{c} \beta_k g_k(X) + \beta_0 + \sum_{k=1}^{c} \beta_k g_k(X)\right] \\ &= \mathbb{E}_P\left[f(X) - \beta_0 - \sum_{k=1}^{c} \beta_k g_k(X)\right] + \beta_0 + \sum_{k=1}^{c} \beta_k \mathbb{E}_P[g_k(X)].\end{aligned}$$

E, dessa forma, temos que o estimador $\hat{\theta}_{n_0}$ para um conjunto de n_0 amostras independentes $\{x_i\}_{i=1}^{n_0}$ igualmente distribuídas de acordo com P tem a forma de

$$\hat{\theta}_{n_0} = \frac{1}{n_0} \sum_{i=1}^{n_0} \left(f(x_i) - \beta_0 - \sum_{k=1}^{c} \beta_k g_k(x_i) \right) + \beta_0 + \sum_{k=1}^{c} \beta_k \mathbb{E}_P[g_k(X)].$$

Para alcançar o objetivo de diminuir a variância de $\hat{\theta}_{n_0}$, queremos escolher $g_1, \ldots, g_c$ e $\beta_0, \ldots, \beta_c$ tal que

$$var_P\left(f(X) - \beta_0 - \sum_{k=1}^{c} \beta_k g_k(X) \right) < var_P\left(f(X)\right).$$

Para tentar alcançar isso, escolhemos definir $\beta_0, \ldots, \beta_c$ como os coeficientes da regressão linear com os inputs $\{(g_1(x_i), \ldots, g_c(x_i))\}_{i=1}^{n_0}$ e os outputs $\{f(x_i)\}_{i=1}^{n_0}$.

Para $\beta_0, \ldots, \beta_c$ definidos dessa forma, temos

$$\sum_{i=1}^{n_0} \left(f(x_i) - \beta_0 - \sum_{k=1}^{c} \beta_k g_k(x_i) \right) = 0,$$

e portanto nosso estimador simplifica para

$$\hat{\theta}_n = \beta_0 + \sum_{k=1}^{c} \beta_k \mathbb{E}_P[g_k(X)].$$

Temos também que

$$var_P(\hat{\theta}_{n_0}) \approx \frac{RSS}{n_0(n_0 - 1 - c)},$$

onde

$$RSS = \sum_{i=1}^{n_0} \left(f(x_i) - \beta_0 - \sum_{k=1}^{c} \beta_k g_k(x_i) \right)^2.$$

Uma discussão mais detalhada de tal técnica e demostrações para resultados apresentados podem ser encontrados em (RIZZO, 2019, pg. 132).

Para $\hat{\pi}_j(x^*)$, utilizaremos como variáveis de controle as funções $g_k(f^*, u) = \Phi(u + f_*^j - f_*^k)$, $k = 1, \ldots, C$, $k \neq j$. Fazemos essa escolha pela relação entre essas funções e $p(j|f^*)$, o que nos faz esperar um valor baixo para RSS, e porque, como demonstrado em (WILLIAMS; RASMUSSEN, 2006, pg. 74), temos uma fórmula fechada para o valor esperado de $\Phi(u + f_*^j - f_*^k)$, dado por

$$\begin{aligned}\mathbb{E}_{N(w^*|\mu_w^*,\Sigma_w^*)}\left[\Phi(u + f_*^j - f_*^k)\right] &= \mathbb{E}_{N(w^*|\mu_w^*,\Sigma_w^*)}\left[\Phi(b_{j,k}^T w^*)\right] = \mathbb{E}_{N\left(g^*\middle|b_{j,k}^T\mu_w^*,\, b_{j,k}^T\Sigma_w^* b_{j,k}\right)}\left[\Phi(g^*)\right] \\ &= \Phi\left(\frac{b_{j,k}^T\mu_w^*}{\sqrt{1 + \left(b_{j,k}^T\Sigma_w^* b_{j,k}\right)}}\right),\end{aligned}$$

onde $w^* = \begin{bmatrix} f^* \\ u \end{bmatrix}$, $\mu_w^* = \begin{bmatrix} \mu^* \\ 0 \end{bmatrix}$, $\Sigma_w^* = \begin{bmatrix} \Sigma^* & 0 \\ 0 & 1 \end{bmatrix}$ e $g^* = b_{j,k}^T w^*$.

É interessante observar que estamos usando $C - 1$ variáveis de controle, e portanto, quando definimos β_k, $k = 0, \ldots, C$, $k \neq j$, como discutido anteriormente, temos que

$$var_{N(w^*|\mu_w^*,\Sigma_w^*)}\left(\hat{\pi}_j(x^*)\right) \approx \frac{RSS}{n_0(n_0 - C)}.$$

Nos casos testados ao longo deste trabalho, o uso dessas variáveis de controle com β_k, $k = 0, \ldots, C$, $k \neq j$, definidos como discutido se mostrou, no geral, efetivo em diminuir suficientemente a variância de maneira que é atrativo usar as variáveis de controle, mesmo com o custo computacional de quando utilizamos n_0 amostras ser consideravelmente maior em comparação a quando não as utilizamos.

Reunimos o discutido nessa seção no algoritmo 3, onde apresentamos um algoritmo para estimar cada uma das probabilidades $\hat{\pi}_j(x^*)$ (4.11) de x^* pertencer a classe j em nosso modelo.

4.5.2 Estimando a Marginal Likelihood

Na seção 4.4, foi apresentado o panorama geral para a aproximação da marginal likelihood e seu gradiente em relação aos hiperparâmetros da função de covariância, que então pode ser usado para achar hiperparâmetros que maximizam a marginal likelihood.

Para implementarmos o cálculo da marginal likelihood do nosso algoritmo, precisamos do valor de $\hat{Z}_i$. Como estamos usando o método de expactation propagation para aproximar as likelihoods, temos que $\hat{Z}_i$ segue a mesma linha de pensamento apresentada na seção 4.4, e portanto, como apontado por (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013), temos

$$\begin{aligned}\log \hat{Z}_i &= \frac{1}{2}\hat{\mu}_{w_i}^T \hat{\Sigma}_{w_i}^{-1}\hat{\mu}_{w_i} + \frac{1}{2}\log\det(\hat{\Sigma}_{w_i}) - \frac{1}{2}\mu_{w_i}^T\Sigma_{w_i}^{-1}\mu_{w_i} - \frac{1}{2}\log\det(\Sigma_{w_i}) + \sum_{k=1,k\neq y_i}^{C}\log\Phi(z_{i,k}) \\ &+ \frac{1}{2}\sum_{k=1,k\neq y_i}^{C}\left((\sigma_{-i,k}^2)^{-1}\mu_{-i,k}^2 + \log(\sigma_{-i,k}^2)\right) - \frac{1}{2}\sum_{k=1,k\neq y_i}^{C}\left((\sigma_{i,k}^2)^{-1}\mu_{i,k}^2 + \log(\sigma_{i,k}^2)\right).\end{aligned}$$

Algoritmo 3: Nested Expectation Propagation - Estimando outputs

Input: X^* (conjunto de novos inputs), k (funções de covariância), K (matriz de covariância), X (inputs do dataset), $\tilde{\nu}$, $\tilde{\mathcal{T}}_i$, $\forall i \in \{1, 2, \ldots, n\}$ (parâmetros locais), n_0 (número de amostras para a integração de Monte Carlo)

Montar $\tilde{\mathcal{T}} = \sum_{i=1}^{n} (\tilde{\mathcal{T}}_i \otimes e_i e_i^T)$, onde e_i é o $i-$ésimo vetor da base canônica de $\mathbb{R}^C$

$L = \text{Cholesky}\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)$

$V = L^{-1}\tilde{\mathcal{T}}^{\frac{1}{2}}$

$M = V^T V$

$E = \tilde{\nu} - MK\tilde{\nu}$

para $x^* \in X^*$ **faça**

  $k_* = K(X, x^*)$

  $\Sigma^* = k(x^*, x^*) - k_*^T M k_*$

  $\mu^* = k_*^T E$

  Gerar amostras $\{f_i\}_{i=1}^{n_0}$ de $N(\mu^*, \Sigma^*)$ e $\{u_i\}_{i=1}^{n_0}$ de $N(0, 1)$

  para $j \in \{1, 2, \ldots, C\}$ **faça**

    $$F_j = \left\{ \prod_{k=1, k\neq j}^{C} \Phi(u_i + f_i^j - f_i^k) \right\}_{i=1}^{n_0}$$

    para $k \in \{1, 2, \ldots, C\}, k \neq j$ **faça**

      $G_{j,k} = \{\Phi(u_i + f_i^j - f_i^k)\}_{i=1}^{n_0}$

      Sendo $b_{j,k}$ definido como para (4.15)

      $$z_{j,k}^* = \frac{b_{j,k}^T \mu_w^*}{\sqrt{1 + \left(b_{j,k}^T \Sigma_w^* b_{j,k}\right)}}$$

      $$E_{g_{j,k}} = b_{j,k}^T \mu_w^* + \frac{\left(b_{j,k}^T \Sigma_w^* b_{j,k}\right) \phi(z_{j,k}^*)}{\Phi(z_{j,k}^*)\sqrt{1 + \left(b_{j,k}^T \Sigma_w^* b_{j,k}\right)}}$$

    Definir $\{\beta_k\}_{k=0, k\neq j}^{C}$ como os coeficientes da regressão linear entre $\{G_{j,k}\}_{k=1,k\neq j}^{C}$ e F_j

    $$\hat{\pi}_j(x^*) = \beta_0 + \sum_{k=1, k\neq j}^{C} \beta_k E_{g_{j,k}}$$

    $$RSS = \sum_{i=1}^{n_0} \left(F_{j,i} - \beta_0 - \sum_{k=1, k\neq j}^{C} \beta_k G_{j,k,i} \right)^2$$

    $$\widehat{var}(\hat{\pi}_j(x^*)) = \frac{RSS}{n_0(n_0 - C)}$$

retorna $(\hat{\pi}_j(x^*))_{j=1}^{C}, (\widehat{var}(\hat{\pi}_j(x^*)))_{j=1}^{C}, \forall x^* \in X^*$

Observando que $\mu_{w_i}^T \Sigma_{w_i}^{-1} \mu_{w_i} = \mu_{-i}^T \Sigma_{-i}^{-1} \mu_{-i}$ e $\det(\Sigma_{w_i}) = \det(\Sigma_{-i})$, a expressão para $\log Z_{EP}$ (4.14) apresenta na seção 4.4 quando adicionada a expressão para $\log \hat{Z}_i$ simplifica para

$$\begin{aligned}\log Z_{EP} &= \frac{1}{2}\mu^T \tilde{\nu} - \frac{1}{2}\log\det\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right) + \frac{1}{2}\sum_{i=1}^{n}\left(\hat{\mu}_{w_i}^T \hat{\Sigma}_{w_i}^{-1}\hat{\mu}_{w_i} + \log\det(\hat{\Sigma}_{w_i})\right) \\ &\quad -\frac{1}{2}\sum_{i=1}^{n}\left(\mu_i^T \Sigma_i^{-1}\mu_i + \log\det(\Sigma_i)\right) + \sum_{i=1}^{n}\left(\sum_{k=1,k\neq y_i}^{C}\frac{1}{2}\left((\sigma_{-i,k}^2)^{-1}\mu_{-i,k}^2 + \log(\sigma_{-i,k}^2)\right)\right. \\ &\quad \left. -\frac{1}{2}\sum_{k=1,k\neq y_i}^{C}\left((\sigma_{i,k}^2)^{-1}\mu_{i,k}^2 + \log(\sigma_{i,k}^2)\right) + \sum_{k=1,k\neq y_i}^{C}\log\Phi(z_{i,k})\right).\end{aligned}$$

Com isso, escrevemos o algoritmo 4 para calcular $\log Z_{EP}$. Devido às fórmulas apresentadas, o melhor lugar para esse algoritmo é apos a convergência dos parâmetros locais no algoritmo 2, e por isso incluímos ele como um output opcional. Aproveitamos também o fato que Σ_i, $\hat{\Sigma}_{w_i}$ e $\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I$ são matrizes positivas definidas para calcular do logaritmo de seus determinantes, como apresentado.

Para a implementação do cálculo do gradiente de $\log Z_{EP}$ apresentado na seção 4.4, temos as mesmos considerações feitas no começo da seção 4.5.1 para o cálculo de cada parte. Como comentado anteriormente, esse gradiente podem ser usado para achar hiperparâmetros que maximizam localmente a marginal log-likelihood do modelo por meio de algum método de otimização, como por exemplo um método de descida de gradiente ou gradiente conjugado (NOCEDAL; WRIGHT, 2006).

Algoritmo 4: Nested Expectation Propagation - Log-marginal likelihood

Input: K (matriz de covariância), Σ, μ (parâmetros da aproximação normal), $\tilde{\nu}$, $\tilde{\mathcal{T}}_i$ (parâmetros locais), α_i, β_i (parâmetros locais do loop interno), $\forall i \in \{1, 2, \ldots, n\}$

$\log Z_{EP} = 0$

para $i \in \{1, 2, \ldots, n\}$ **faça**

$V = \Sigma_i(\tilde{\mathcal{T}}_i\Sigma_i - I)^{-1}\tilde{\mathcal{T}}_i$

$\Sigma_{-i} = \Sigma_i - V\Sigma_i$

$\mu_{-i} = \mu_i - V\mu_i - \Sigma_{-i}\tilde{\nu}_i$

Montar Σ_{w_i} e μ_{w_i} de acordo com (4.15) e B_i, $\tilde{\alpha}_i$ e $\tilde{\beta}_i$ como definido para (4.16)

$B_{\alpha_i} = B_i \text{diag}\left(\tilde{\alpha}_i^{\frac{1}{2}}\right)$

$L = \text{Cholesky}\left(I + B_{\alpha_i}^T \Sigma_{w_i} B_{\alpha_i}\right)$

$V = L^{-1}\left(B_{\alpha_i}^T \Sigma_{w_i}\right)$

$\hat{\Sigma}_{w_i} = \Sigma_{w_i} - V^T V$

$\hat{\mu}_{w_i} = \mu_{w_i} - V^T\left(L^{-1}B_{\alpha_i}^T \mu_{w_i}\right) + \hat{\Sigma}_{w_i} B_i \tilde{\beta}_i$

para $k \in \{1, 2, \ldots C\}$, $k \neq y_i$, **faça**

Sendo $b_{j,k}$ definido como para (4.15)

$\sigma_{i,k}^2 = b_{y_i,k}^T \hat{\Sigma}_{w_i} b_{y_i,k}, \quad \mu_{i,k} = b_{y_i,k}^T \hat{\mu}_{w_i}$

$\sigma_{-i,k}^2 = \left((\sigma_{i,k}^2)^{-1} - \alpha_{i,k}\right)^{-1}, \quad \mu_{-i,k} = \sigma_{-i,k}^2((\sigma_{i,k}^2)^{-1}\mu_{i,k} - \beta_{i,k})$

$z_{i,k} = \dfrac{\mu_{-i,k}}{\sqrt{(1 + \sigma_{-i,k}^2)}}$

$\log Z_{EP} = \log Z_{EP} + \frac{1}{2}\left((\sigma_{-i,k}^2)^{-1}\mu_{-i,k}^2 + \log(\sigma_{-i,k}^2) - (\sigma_{i,k}^2)^{-1}\mu_{i,k}^2 - \log(\sigma_{i,k}^2)\right)$

$\log Z_{EP} = \log Z_{EP} + \log \Phi(z_{i,k})$

$M = \text{Cholesky}\left(\Sigma_i\right); \quad N = \text{Cholesky}\left(\hat{\Sigma}_{w_i}\right)$

$\log Z_{EP} = \log Z_{EP} + \dfrac{1}{2}\left(\hat{\mu}_{w_i}^T \hat{\Sigma}_{w_i}^{-1} \hat{\mu}_{w_i} - \mu_i^T \Sigma_i^{-1} \mu_i\right) + \sum_{j=1}^{n} \log(n_{j,j}) - \log(m_{j,j})$

Montar $\tilde{\mathcal{T}} = \sum_{i=1}^{n}(\tilde{\mathcal{T}}_i \otimes e_i e_i^T)$, onde e_i é o $i-$ésimo vetor da base canônica de $\mathbb{R}^C$

$L = \text{Cholesky}\left(\tilde{\mathcal{T}}^{\frac{1}{2}} K \tilde{\mathcal{T}}^{\frac{1}{2}} + I\right)$

$\log Z_{EP} = \log Z_{EP} + \frac{1}{2}\mu^T\tilde{\nu} - \sum_{j=i}^{n} \log(l_{j,j})$

retorna $\log Z_{EP}$

5 Outras Métodos de Aproximação para Regressão e Classificação por Processos Gaussianos

Como mencionado no capítulo 3, vamos discutir agora outros métodos para conseguir aproximações das distribuições posteriores em problemas de regressão e classificação por processos Gaussianos. Em contraposição ao método de expectation propagation, entraremos em menos detalhes dos métodos, mas apresentaremos pseudos-códigos para tais.

De forma geral, em relação ao desempenho dos dois métodos que serão apresentados nessa seção e do método de expectation propagation apresentado no capítulo 4, é esperado que a aproximação feita pelo método de expectation propagation é mais próxima à distribuição posterior exata do que a do método de aproximação de Laplace, entretanto o algoritmo da aproximação de Laplace é esperado que seja computacionalmente mais rápido do que o do expectation propagation. Por outro lado, temos que o método *Markov Chain Monte Carlo* (MCMC) converge para a distribuição posterior exata, mas em contrapartida é esperado ser o mais computacionalmente lento entre os métodos apresentados. Tais comportamentos podem ser observados em trabalhos como (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013) e (WILLIAMS; RASMUSSEN, 2006).

5.1 Aproximação de Laplace

O objetivo aqui é conseguir uma aproximação normal $q(f|X,Y)$ por meio de uma expansão de Taylor de segunda ordem de $\log p(f|X,Y)$ em um ponto de máximo.

Seja $\hat{f} = \arg\max_f p(f|X,Y)$. Então, $\hat{f}$ também é um ponto de máximo de $\log p(f|X,Y)$. Como mostra (NOCEDAL; WRIGHT, 2006, pg. 15), temos então que $\nabla_f \log p(f|X,Y) = 0$ e a matriz hessiana $\nabla_f^2 \log p(f|X,Y)$ é negativa definida (assumindo $\hat{f}$ maximizador local estrito), olhando $\log p(f|X,Y)$ como uma função de f.

Com isso, podemos fazer a seguinte expansão de Taylor

$$\begin{aligned}
\log p(f|X,Y) &= \log p(\hat{f}|X,Y) + (\nabla_f \log p(\hat{f}|X,Y))^T (f-\hat{f}) \\
&\quad + \frac{1}{2}(f-\hat{f})^T (\nabla_f^2 \log p(\hat{f}|X,Y))(f-\hat{f}) + R(f) \\
&= \log p(\hat{f}|X,Y) - \frac{1}{2}(f-\hat{f})^T (-\nabla_f^2 \log p(\hat{f}|X,Y))(f-\hat{f}) + R(f)
\end{aligned}$$

Assim, se tomarmos

$$Q(f|X,Y) = N(f|\hat{f}, (-\nabla_f^2 \log p(\hat{f}|X,Y))^{-1})$$

temos

$$\begin{aligned} p(f|X,Y) &= \exp\left(\log p(f|X,Y)\right) \\ &\approx \exp\left(\log p(\hat{f}|X,Y) - \frac{1}{2}(f-\hat{f})^T(-\nabla_f^2 \log p(\hat{f}|X,Y))(f-\hat{f})\right) \\ &\propto \exp\left(-\frac{1}{2}(f-\hat{f})^T(-\nabla_f^2 \log p(\hat{f}|X,Y))(f-\hat{f})\right) \\ &\propto q(f|X,Y) \end{aligned}$$

Observando $\log p(f|X,Y)$ mais detalhadamente e derivando em relação a f, temos, lembrando que $f|X \sim N(0,K)$,

$$\begin{aligned} \log p(f|X,Y) &= \log p(Y|f) + \log p(f|X) - \log p(Y|X) \\ &= \log p(Y|f) - \frac{nC}{2}\log(2\pi) - \frac{1}{2}\log\det(K) - \frac{1}{2}f^T K^{-1} f - \log p(Y|X) \\ \Rightarrow \nabla_f \log p(f|X,Y) &= \nabla_f \log p(Y|f) - K^{-1}f \\ \Rightarrow \nabla_f^2 \log p(f|X,Y) &= \nabla_f^2 \log p(Y|f) - K^{-1} = -W(f) - K^{-1} \end{aligned}$$

onde $W(f) = (-\nabla_f^2 \log p(Y|f))$. Observe também que como

$$\log p(Y|f) = \sum_{i=1}^{n} \log p(y_i|f_i),$$

temos

$$W(f) = \sum_{i=1}^{n} (-\nabla_{f_i}^2 \log p(y_i|f_i) \otimes e_i e_i^T),$$

i.e., podemos montar a matriz $W(f)$ assumindo que temos acesso as hessianas $\nabla_{f_i}^2 \log p(y_i|f_i), i \in \{1,2,\ldots,n\}$.

Para $\hat{f}$, podemos utilizar o método de Newton (NOCEDAL; WRIGHT, 2006, pg. 44) com tamanho de passo igual a 1 para minimizar $-\log p(f|X,Y)$, o que nos dá a iteração

$$\begin{aligned} f^{new} &= f - (\nabla_f^2 \log p(f|X,Y))^{-1}(\nabla_f \log p(f|X,Y)) \\ &= f + (W(f) + K^{-1})^{-1}(\nabla_f \log p(Y|f) - K^{-1}f) \\ &= (W(f) + K^{-1})^{-1}(\nabla_f \log p(Y|f) + (W(f) + K^{-1})f - K^{-1}f) \\ &= (W(f) + K^{-1})^{-1}(\nabla_f \log p(Y|f) + W(f)f). \end{aligned}$$

Caso $p(f|X,Y)$ seja uma função log-côncava, temos a garantia de convergência global do método de Newton para o único máximo global (caso exista). Para o caso geral, (NOCEDAL; WRIGHT, 2006, pg. 48) apresenta uma proposta de modificação para o método de Newton para obter uma convergência global.

Com essa aproximação normal, por argumentos similares aos apresentados na seção 4.3, temos que a aproximação para a distribuição posterior de f^* é da forma

$$Q(f^*|X,Y,x^*) \sim N(f^*|k_*^T(\nabla_f \log p(Y|\hat{f})),\ k(x^*,x^*) - k_*^T(K + W(\hat{f})^{-1})^{-1}k_*), \tag{5.1}$$

já que $\mathbb{E}_{f^*|X,Y,x^*}[f^*] = k_*^T K^{-1}\mathbb{E}_{f|X,Y}[f]$, e, como $\hat{f}$ é um máximo local, temos $\nabla_f \log p(\hat{f}|X,Y) = \nabla_f \log p(Y|\hat{f}) - K^{-1}\hat{f} = 0$. Com essa aproximação, podemos estimar os valores de $\hat{\pi}_j(x^*)$, para cada classe j, de um novo input x^* da mesma forma que (4.11).

Finalmente, (WILLIAMS; RASMUSSEN, 2006, pg. 48) apresenta uma expressão para a aproximação da marginal likelihood pelo método de aproximação de Laplace:

$$\log p(Y|X) \approx \log q(Y|X) = -\frac{1}{2}\hat{f}^T K^{-1}\hat{f} + \log p(Y|\hat{f}) - \frac{1}{2}\log\det\left(I + W(\hat{f})^{\frac{1}{2}} K W(\hat{f})^{\frac{1}{2}}\right).$$

A utilização do método de Newton dá à aproximação de Laplace uma garantia de convergência local com uma velocidade de convergência quadrática, sobre as hipóteses necessárias. Todavia, não temos estimativas para a qualidade dessa aproximação. Para efeitos de comparação, (RIIHIMÄKI; JYLÄNKI; VEHTARI, 2013) conclui de seus experimentos que as aproximações geradas pelo algoritmo *nested expectation propagation* se mostraram mais precisas que as aproximações de Laplace em aproximar a distribuição posterior $p(f|X,Y)$, mas ambos os métodos tiveram resultados similares em relação a acurácia das classes estimadas.

5.1.1 Algoritmo Aproximação de Laplace para Classificação

Nessa seção, apresentamos um algoritmo utilizando o método de aproximação de Laplace para problemas de classificação por processos gaussianos com duas ou mais classes, desenvolvido por (WILLIAMS; RASMUSSEN, 2006, pg. 48). Escolhemos aqui a função likelihood como a função *softmax*, dada por

$$p(j|f_i) = \pi_i^j = \frac{\exp\left(f_i^j\right)}{\sum_{k=1}^{D}\exp\left(f_i^k\right)}.$$

Observe que nesse caso temos

$$\frac{\partial}{\partial f_i^j}\left[\log p(y_i|f_i)\right] = \delta_{jy_i} - \pi_i^j$$

e

$$\frac{\partial^2}{\partial f_i^j \partial f_i^k}\left[\log p(y_i|f_i)\right] = -\pi_i^j\delta_{kj} + \pi_i^j\pi_i^k,$$

onde $\delta_{kj} = 1$ se $j = k$ e $\delta_{kj} = 0$ se $j \neq k$.

Definindo y e π como vetores com mesma dimensão que f com as entradas correspondentes a f_i^j sendo δ_{jy_i} em y e π_i^j em π, π_i sendo o subvetor com C entradas de π correspondente a f_i e π^j o subvetor com n entradas de π contendo apenas os valores correspondentes a classe j. Com isso, temos, pelo apresentado anteriormente, que

$$\nabla_f \log p(Y|f) = y - \pi$$

e

$$\nabla^2_{f_i} \log p(y_i|f_i) = -\mathrm{diag}(\pi_i) + \pi_i\pi_i^T.$$

Observe que, sendo u_k o $k-$ésimo vetor da base canônica de $\mathbb{R}^C$ e e_i o $i-$ésimo vetor da base canônica de $\mathbb{R}^n$,

$$\begin{aligned}\sum_{i=1}^{n}(\pi_i\pi_i^T \otimes e_ie_i^T) &= \sum_{i,j=1}^{n}(\pi_i\pi_i^T \otimes e_ie_i^Te_je_j^T)\\ &= \sum_{i,j=1}^{n}\sum_{k,l=1}^{C}\pi_i^k\pi_i^l(u_ku_l^T \otimes e_ie_i^Te_je_j^T)\\ &= \left(\sum_{i=1}^{n}\sum_{k=1}^{C}(u_k \otimes \pi_i^ke_ie_i^T)\right)\left(\sum_{j=1}^{n}\sum_{l=1}^{C}(u_l \otimes \pi_j^le_je_j^T)\right)^T\\ &= \left(\sum_{j=1}^{C}(u_j \otimes \mathrm{diag}(\pi^j))\right)\left(\sum_{j=1}^{C}(u_j \otimes \mathrm{diag}(\pi^j))\right)^T.\end{aligned}$$

Assim, denotando $\Pi = \sum_{j=1}^{C}(u_j \otimes \mathrm{diag}(\pi^j))$, i.e., Π é a matriz $nC \times n$ com as matrizes $\mathrm{diag}(\pi^j)$ empilhadas, temos

$$W(f) = \sum_{i=1}^{n}\left(-\nabla_{f_i}^2 \log p(y_i|f_i) \otimes e_ie_i^T\right) = \mathrm{diag}(\pi) - \Pi\Pi^T.$$

Portanto

$$\nabla_f \log p(f|X,Y) = -K^{-1}f + y - \pi \tag{5.2}$$

e

$$\nabla_f^2 \log p(f|X,Y) = -K^{-1} - W(f) = -K^{-1} - \mathrm{diag}(\pi) + \Pi\Pi^T. \tag{5.3}$$

É interessante observar que $-\nabla_{f_i}^2 \log p(y_i|f_i)$ é semipositiva definida, $\forall i$, pelo já demonstrado na seção 3.4.4, e portanto $W(f)$ é semipositiva definida também. Assim, $p(f|X,Y)$ é log-côncava e temos a garantia de todas as propriedades para o método de Newton comentadas na seção anterior. Temos também que $\hat{f} = K(y - \pi(\hat{f}))$.

Em particular para a iteração do método de Newton, temos, como mostra (WILLIAMS; RASMUSSEN, 2006, pg. 52),

$$(K^{-1} + W(f))^{-1} = K - K(K + W(f)^{-1})^{-1}K,$$

onde $(K + W(f)^{-1})^{-1}$ é uma notação (já que neste caso temos $W(f)$ singular) para a matriz

$$(K + W(f)^{-1})^{-1} = E - ER\left(\sum_{j=i}^{C}E_j\right)^{-1}R^TE,$$

e $E = (K + \mathrm{diag}(\pi)^{-1})^{-1} = \mathrm{diag}(\pi)^{\frac{1}{2}}(I + \mathrm{diag}(\pi)^{\frac{1}{2}}K\mathrm{diag}(\pi)^{\frac{1}{2}})^{-1}\mathrm{diag}(\pi)^{\frac{1}{2}}$ é uma matriz diagonal por blocos, $\sum_{j=i}^{C}E_j$ é a soma desses blocos de E e $R = \mathrm{diag}(\pi)^{-1}\Pi$ é a matriz $nC \times n$ com I_n empilhada C vezes.

Por fim, para o cálculo da marginal likelihood, temos, usando as identidades (A.5) e (A.6) e definindo $D = \text{diag}(\pi)$,

$$\begin{aligned}
\det\left(I + W(f)^{\frac{1}{2}} K W(f)^{\frac{1}{2}}\right) &= \det\left(I + KW(f)\right) = \det\left(K(K^{-1} + W(f)\right) \\
&= \det(K)\det\left(K^{-1} + D - \Pi\Pi^T\right) \\
&= \det(K)\det(K^{-1} + D)\det(I)\det(I + \Pi^T(K^{-1} + D)^{-1}\Pi) \\
&= \det(I + KD)\det(I - \Pi^T(D^{-1} - D^{-1}(K + D^{-1})^{-1}D^{-1})\Pi) \\
&= \det(I + D^{\frac{1}{2}} K D^{\frac{1}{2}})\det(I - \Pi^T D^{-1}\Pi + R^T E R) \\
&= \det\left(I + D^{\frac{1}{2}} K D^{\frac{1}{2}}\right)\det\left(\sum_{j=i}^{C} E_j\right)
\end{aligned}$$

Juntando o apresentado nessa seção, montamos um pseudo-código do algoritmo para estimar $\hat{f}$ e a marginal likelihood do método de aproximação de Laplace com a likelihood *softmax* para problemas de classificação no algoritmo 5. Montamos também no algoritmo 6 um pseudo-código para o algoritmo que vai estimar os valores de $\hat{\pi}_j(x^*)$, para cada classe j, de um novo input x^* como definido em (4.11) usando (5.1). Optamos por utilizar a formulação mais básica do método de integração de Monte Carlo, como discutido na seção 4.5.1, para aproximar (4.11).

Algoritmo 5: Aproximação de Laplace para Classificação

Input: K (matriz de covariância), Y (outputs do dataset), *likelihood* =Falso (Definir como verdeiro caso o cálculo da log-marginal likelihood seja desejado)

Montar y como definido para (5.2)
$f = 0$
repita
  Calcular π e Π em função de f como definido para (5.2) e (5.3)
  para $j \in \{1, 2, \ldots, C\}$ **faça**
    $L = Cholesky(I + \text{diag}(\pi^j)^{\frac{1}{2}} K_j \text{diag}(\pi^j)^{\frac{1}{2}})$
    $E_j = \text{diag}(\pi^j)^{\frac{1}{2}}(L^{-T}(L^{-1}\text{diag}(\pi^j)^{\frac{1}{2}}))$
    $z_j = \sum_{i=1}^{n} \log l_{ii}$
  $M = Cholesky(\sum_{j=1}^{C} E_j)$
  $b = (\text{diag}(\pi) - \Pi\Pi^T)f + y - \pi$
  $c = EKb$
  $a = b - c + ER(M^{-T}(M^{-1}R^T c))$
  $f = Ka$
até f *convergir*
se *likelihood=Verdadeiro* **então**
  $\log q(Y|X) = -\frac{1}{2}a^T f + y^T f - \sum_{i=1}^{n} \log\left(\sum_{j=1}^{C} \exp(f_j^i)\right) - \sum_{j=1}^{C} z_j - \sum_{i=1}^{n} \log m_{ii}$
  retorna $f, \log q(Y|X)$
retorna f

5.2 *Markov Chain Monte Carlo*

Os métodos anteriormente apresentados têm o propósito de aproximar a distribuição posterior por uma distribuição normal e os algoritmos apresentados fixam uma função likelihood pré-

Algoritmo 6: Aproximação de Laplace para Classificação - Estimando Outputs

Input: X^* (conjunto de novos inputs), k (funções de covariância), K (matriz de covariância), X (inputs do dataset), Y (outputs do dataset), $\hat{f}$ (Moda de $P(f|Y,X)$), n_0 (número de amostras para a integração de Monte Carlo)

Calcular π, Π e y em função de $\hat{f}$ como definido para (5.2) e (5.3)
para $j \in \{1, 2, \ldots, C\}$ **faça**
  $L = Cholesky(I + \text{diag}(\pi^j)^{\frac{1}{2}} K_j \text{diag}(\pi^j)^{\frac{1}{2}})$
  $E_j = \text{diag}(\pi^j)^{\frac{1}{2}}(L^{-T}(L^{-1}\text{diag}(\pi^j)^{\frac{1}{2}}))$
$M = Cholesky(\sum_{j=1}^{C} E_j)$
$V = M^{-1}(R^T E)$
$\hat{\Sigma} = E - V^T V$
$\hat{\mu} = (y - \pi)$
para $x^* \in X^*$ **faça**
  $k_* = K(X, x^*)$
  $\Sigma^* = k(x^*, x^*) - k_*^T \hat{\Sigma} k_*$
  $\mu^* = k_*^T \hat{\mu}$
  Gerar amostras $\{f_i\}_{i=1}^{n_0}$ de $N(\mu^*, \Sigma^*)$
  para $j \in \{1, 2, \ldots, C\}$ **faça**
    $$\hat{\pi}_j(x^*) = \frac{1}{n_0} \sum_{i=1}^{n_0} p(j|f_i)$$
retorna $(\hat{\pi}_j(x^*))_{j=1}^{C}$, $\forall x^* \in X^*$

escolhida. Entretanto, gostaríamos de ter a opção de uma aproximação da distribuição posterior exata com uma função de likelihood qualquer. Para isso, apresentamos o método de aproximação *Markov Chain Monte Carlo* (MCMC).

Já discutimos um pouco sobre métodos de Monte Carlo na seção 4.5.1. Se conseguíssemos gerar observações independentes de $p(f|X,Y)$, poderíamos estimar $p(f^*|X,Y,x^*)$ e por sua vez $\hat{\pi}_j(x^*)$. Entretanto, como gerar essas observações? Como queremos apresentar um único método que cubra qualquer escolha de likelihood, apresentamos o método MCMC, que consiste em usar uma cadeia de Markov para gerar observações da distribuições desejada e usar essas para o método de Monte Carlo.

Uma cadeia de Markov em tempo discreto é um processo estocástico $\{X_n\}_{n=0}^{\infty}$ tal que

$$\mathbb{P}\{X_{n+1} \in A | X_0 = x_0, \ldots X_n = x_n\} = \mathbb{P}\{X_{n+1} \in A | X_n = x_n\} = P(x_n, A, n),$$

onde $A \subseteq E$, $x_0, \ldots, x_{n+1} \in E$ e E é chamado de conjunto de estados da cadeia, isto é, a distribuição de X_{n+1} depende apenas do passo anterior X_n da cadeia. Dizemos que a cadeia é homogênea se ela não depende do tempo n, i.e., $P(x, A, n) = P(x, A, 0) = P(x, A)$ para qualquer A e x. Neste caso, temos que, de forma recursiva,

$$\mathbb{P}\{X_n \in A | X_0 = x\} = \int_E P^{n-1}(x, y) P(y, A) dy = P^n(x, A)$$

Por fim, para uma distribuição π em E, dizemos que π é a distribuição estacionária, ou

distribuição invariante, de uma cadeia homogênea se, $\forall A \subseteq E$,

$$\int_A d\pi = \int_E P(x, A)\, d\pi.$$

Como apontam os resultados mostrados em (GILKS; RICHARDSON; SPIEGELHALTER, 1995, Cap. 4), temos que se $\{X_n\}_{n=0}^{\infty}$ é uma cadeia de Markov homogênea, irredutível, aperiódica (essas definições podem ser vista em (GILKS; RICHARDSON; SPIEGELHALTER, 1995, pg. 62,65)) e com uma distribuição estacionária π, então

$$\lim_{n\to\infty} P^n(x, A) = \pi(A),$$

para todo $A \subseteq E$ e $\mathbb{P}_\pi-$quase todo x.

E mais ainda, se temos uma cadeia de Markov homogênea, irredutível e com uma distribuição estacionária π e $f : E \to \mathbb{R}$ é uma função tal que $\mathbb{E}_\pi[|f(x)|] < \infty$, então

$$P(x, \{\bar{f}_n \overset{n\to\infty}{\longrightarrow} \mathbb{E}_\pi[f(x)]\}) = 1,$$

para $\mathbb{P}_\pi-$quase todo x, onde

$$\bar{f}_n = \frac{1}{n+1} \sum_{i=0}^{n} f(X_i).$$

Desses resultados temos que se temos uma sequência $\{x_n\}_{n=0}^{\infty}$ de observação dos estados da cadeia de Markov $\{X_i\}_{i=0}^{\infty}$, i.e., x_n é uma observação de X_n, então essas amostras dependentes (e por isso a teoria anteriormente apresentada para o método de Monte Carlo não se aplica diretamente) são tais que, para valores suficientemente grande de n, x_n são aproximadamente amostras de π. Além disso, temos, sob as hipóteses colocada, a garantia de convergência do método MCMC para o valor verdadeiro, diferenciando este dos outros métodos apresentados.

O método MCMC para estimar $\theta = \mathbb{E}_P[f(x)]$ se resume em criar uma cadeia de Markov tal que se encaixe nas hipóteses dos resultados referenciados e que a distribuição P é a distribuição estacionária desta. Então reunimos as amostras $\{x_n\}_{n=0}^{\infty}$ de forma que x_n é o estado observado de X_n e calculamos o estimador

$$\hat{\theta} = \frac{1}{n-m} \sum_{i=m+1}^{n} f(X_i)$$

para o valor esperado desejado. m é conhecido como o período de *burn-in*, isto é, descartamos os primeiros m estados observados da cadeia. A ideia por trás desse período é esperar a distribuição da cadeia se aproximar da distribuição estacionária para calcular $\hat{\theta}$ utilizando amostras de uma distribuição mais próxima da desejada.

5.2.1 Algoritmo de Metropolis–Hastings

Como descrito na seção anterior, queremos estimar $\theta = \mathbb{E}_P[f(x)]$ pelo método MCMC. Como mostra (GILKS; RICHARDSON; SPIEGELHALTER, 1995, pg. 5), um dos métodos mais usados para criar uma cadeia de Makov com distribuição estacionária P, com densidade p, é conhecido com algoritmo de Metropolis–Hastings.

Para iniciar o algoritmo, precisamos de uma distribuição de proposta $Q|Y$ (condicionada a Y), com densidade $q(x|y)$. Um exemplo seria tomar $Q|Y \sim N(Y, \Sigma)$, i.e., $Q|Y$ é uma distribuição normal com valor médio Y e uma variância constante Σ. Com $Q|Y$ e um ponto inicial $X_0 = x_0$, o algoritmo segue os seguintes passos

1. Gerar uma amostra y de $Q|X_n$;
2. Calcular
$$\alpha(x_n, y) = \min\left(1, \frac{p(y)q(x_n|y)}{p(x_n)q(y|x_n)}\right);$$
3. Gerar uma amostra u de $U(0,1)$ (distribuição uniforme no intervalo $(0,1)$);
4. Se $u \leq \alpha(x_n, y)$, então $X_{n+1} = y$. Caso contrário, $X_{n+1} = x_n$;
5. Voltar ao passo 1 e repetir até ter o número desejado de amostras.

Uma demonstração que P é a distribuição estacionária da cadeia gerada pode ser vista em (GILKS; RICHARDSON; SPIEGELHALTER, 1995, pg. 7).

É interessante observar que não precisamos da constante de normalização de P (nem de $Q|Y$) para calcular $\alpha(x, y)$, já que, por exemplo, se p é proporcional a f, temos que $p(x) = \frac{f(x)}{\int f(y)\,dy}$ e, portanto, $\frac{p(y)}{p(x)} = \frac{f(y)}{f(x)}$.

Veja a importância da escolha de distribuição de proposta $Q|Y$. Precisamos escolher tal que conseguimos gerar amostras dela, mas, ao mesmo tempo, queremos $Q|Y$ parecida com P, já que assim esperamos valores maiores para $\alpha(x_n, y)$ e portanto mais pontos serão aceitos, o que esperamos gerar uma convergência mais rápida das distribuições.

Existem diversas estratégias para escolher a distribuição de proposta, como apresentado em (GILKS; RICHARDSON; SPIEGELHALTER, 1995, Cap. 1), mas vamos apenas discutir uma delas aqui. O algoritmo de Metropolis consiste em tomar a distribuição de proposta com densidade simétrica no sentido de que $q(x|y) = q(y|x)$, para qualquer x e y. Neste caso, temos

$$\alpha(x, y) = \min\left(1, \frac{p(y)}{p(x)}\right).$$

Como exemplo, temos novamente $Q|Y \sim N(Y, \Sigma)$, já que $q(x|y) = \phi(x|y, \Sigma) = \phi(y|x, \Sigma) = q(y|x)$.

5.2.2 Algoritmo MCMC para Classificação por Processos Gaussianos

Vamos agora aplicar o método MCMC para o problema de Classificação por Processos Gaussianos apresentado na seção 3.3.

Primeiro iremos aplicar o método para gerar amostras $\{\hat{f}_i\}_{i=0}^{N_0}$ de f aproximadamente distribuídas de acordo com a distribuição posterior, conforme apresentado no algoritmo 7. Escolhemos a distribuição de proposta como $Q|Y \sim N(Y, \Sigma)$, onde deixamos Σ como um input do algoritmo. Uma recomendação para o algoritmo 7 seria utilizar o método de aproximação de Laplace ou expectation propagation antes e escolher Σ como um múltiplo da aproximação da

matriz de covariância posterior encontrada (possivelmente com uma constante de normalização) e utilizar a aproximação do valor esperado como ponto inicial.

Por fim, lembrando que

$$p(f|X,Y) = \frac{p(Y|f)p(f|X)}{p(Y|X)} = \frac{p(f|X)}{p(Y|X)} \prod_{i=1}^{n} p(y_i|f_i),$$

temos, para o cálculo de α do algoritmo de Metropolis, que

$$\frac{p(f_a|X,Y)}{p(f_b|X,Y)} = \frac{\phi(f_a|0,K)}{\phi(f_b|0,K)} \left(\prod_{i=1}^{n} \frac{p(y_i|f_a)}{p(y_i|f_b)} \right) = \left(\prod_{i=1}^{n} \frac{p(y_i|f_a)}{p(y_i|f_b)} \right) \exp\left(\frac{1}{2} \left(f_b^T K^{-1} f_b - f_a^T K^{-1} f_a \right) \right),$$

i.e., não precisamos do valor da marginal likelihood $p(Y|X)$ (que a princípio não temos acesso) para gera tais amostras.

Com essa amostras, podemos aproximar a distribuição posterior $f|X,Y$, estimando sua cdf, quantis e outras propriedades e medidas que podem ser usadas para avaliar a qualidade das aproximações de $f|X,Y$ geradas pelos outros métodos apresentados.

Pode-se aplicar o método novamente, já que com as amostras geradas podemos estimar

$$p(f^*|X,Y,x^*) = \int p(f^*|X,x^*,f)p(f|X,Y)\, df = \mathbb{E}_{f|X,Y}[p(f^*|X,x^*,f)]$$

por

$$\hat{p}(f^*|X,Y,x^*) = \frac{1}{N_0 - m} \sum_{i=m+1}^{N_0} p(f^*|X,x^*,\hat{f}_i),$$

e com isso gerar amostras de f^* para estimar

$$\hat{\pi}_j(x^*) = \int p(j|f^*)p(f^*|X,Y,x^*)\, df^* = \mathbb{E}_{f^*|X,Y,x^*}[p(j|f^*)],$$

lembrando que, por (3.6) e tomando Σ como um múltiplo da identidade,

$$p(f^*|X,x^*,f) = \phi(f^*|k_x^T(K+\lambda I)^{-1}f,\ k(x^*,x^*) - k_x^T(K+\lambda I)^{-1}k_x),$$

onde a constante de regularização λ é escolhida de forma a melhorar o condicionamento de K, caso necessário.

Entretanto, essa estratégia requer muitos pontos para conseguirmos boas aproximações de $\hat{\pi}_j(x^*)$, já que as amostrar de f^* dependem de termos boas aproximações de $p(f^*|X,Y,x^*)$, que por sua vez é estimado a partir de amostras de dimensão nC. As grandes dimensões dos suportes das distribuições que estamos amostrando e o fato do erro de uma aproximação ser carregado para a próxima são fatores importantes para evidenciar a necessidade de um número consideravelmente grande de amostras que precisam ser geradas e, consequentemente, o grande custo computacional de realizar toda a inferência por esse método.

Por causa disso, restringiremos o uso do método MCMC apenas para a comparação das aproximações da distribuição posterior de f pelos outros métodos apresentados com a distribuição posterior exata no experimento computacional relevante do capítulo 6. Isto é, não usaremos o método MCMC para classificar novos inputs, mas sim para avaliar os outros métodos de aproximação apresentados.

Algoritmo 7: Aproximação MCMC para Classificação - Amostrando

Input: K (matriz de covariância), Y (outputs do dataset), $p(j|f)$ (função likelihood), x_0 (ponto inicial para a cadeia de Markov), Σ (matriz de variância da distribuição de proposta), N (número de amostras desejadas), λ (constante de regularização)

$L = Cholesky(\Sigma)$
$M = Cholesky(K + \lambda I_{nC})$
Gerar $\{v_i\}_{i=0}^{N-1}$ amostras de $N(0, I_{nC})$
Gerar $\{u_i\}_{i=0}^{N-1}$ amostras de $U(0, 1)$
$l_0 = \prod_{j=1}^{n} p(y_j|x_0)$
$s_0 = M^{-1}x_0$
para $i \in \{0, 1, \ldots, N-1\}$ **faça**
    $v_i^* = Lv_i + x_i$
    $l_i^* = \prod_{j=1}^{n} p(y_j|v_i^*)$
    $s_i^* = M^{-1}v_i$
    $\alpha(v_i^*, x_i) = \left(\frac{l_i^*}{l_i}\right) \exp\left(\frac{1}{2}\left(s_i^T s_i - (s_i^*)^T s_i^*\right)\right)$
    se $u_i \leq \alpha(v_i^*, x_i)$ **então**
        $x_{i+1} = v_i^*$
        $l_{i+1} = l_i^*$
        $s_{i+1} = s_i^*$
    senão
        $x_{i+1} = x_i$
        $l_{i+1} = l_i$
        $s_{i+1} = s_i$
retorna $\{x_i\}_{i=0}^{N}$

6 Experimentos Numéricos

Aplicaremos os métodos discutidos nos capítulos anteriores em problemas de classificação e regressão. Todos os experimentos foram realizados em Python.

Primeiramente, temos um problema de regressão criado artificialmente na seção 6.1. Nosso objetivo aqui é observar o impacto de diferentes escolhas de função de covariância na distribuição posterior e na qualidade da predição para a regressão.

Já na seção 6.2, temos um problema de classificação clássico utilizando o dataset *Iris*. Nosso objetivo é comparar o desempenho dos algoritmos apresentados para classificação por processos gaussianos.

6.1 Problema Teste

O primeiro problema que vamos discutir é um exemplo artificial que construímos amostrando a função

$$f(x) = \sin\left(\frac{1+e^x}{5\pi}\right) \tag{6.1}$$

em $n = 11$ pontos igualmente espaçados no intervalo $[2.5, 5]$, com $x_1 = 2.5$ e $x_{11} = 5$. Adicionamos então um ruído aleatório $\epsilon \sim N(0, 10^{-3})$ em cada $f(x_i)$. Apresentamos o gráfico da função f e o gráfico dos pares de dados (x_i, y_i) na figura 5. O objetivo aqui é estimar a função f a partir do conjunto de dados gerado utilizando a regressão por processos gaussianos discutida na seção 3.2. Apresentamos o dataset completo no apêndice A.6.1.

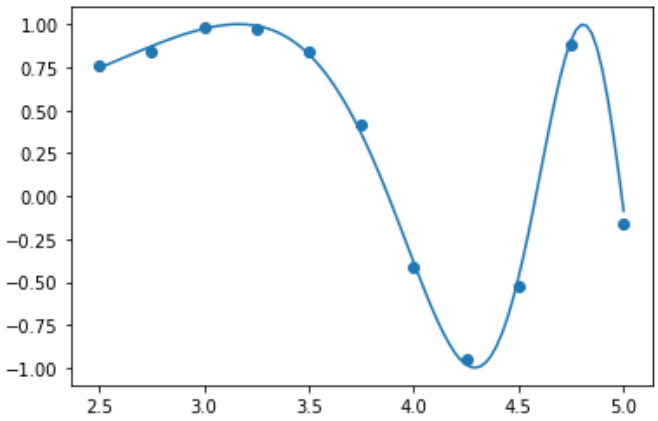


Figura 5 – Gráfico da função da função f definida em (6.1) (linha contínua) e o gráfico dos pares de dados (x_i, y_i) do problema (círculos).

Usaremos a fórmula (3.6) com a função de valor médio m nula e $\Sigma = 10^{-3}I$. Escolhemos esse valor para Σ pois sabemos qual é a distribuição do erro, esse é um cenário semelhante ao de termos os outputs como resultados de medições onde sabemos a precisão de tais. Vamos avaliar

o uso de quatro funções de covariância distintas: a função de covariância exponencial quadrática

$$k_{SE}(x, y|\ell, \sigma^2) = \sigma^2 \exp\left(-\frac{(x-y)^2}{2\ell^2}\right),$$

com $\ell, \sigma^2 > 0$. A função de covariância de rede neural

$$k_{NN}(x, y|\sigma_0, \sigma_1, \sigma^2) = \frac{2\sigma^2}{\pi} \sin^{-1}\left(\frac{2\sigma_0 + 2\sigma_1 xy}{\sqrt{(1 + 2\sigma_0 + 2\sigma_1 x^2)(1 + 2\sigma_0 + 2\sigma_1 y^2)}}\right),$$

com $\sigma_0, \sigma_1, \sigma^2 > 0$. A função de covariância polinomial com $p = 1$

$$k_{LIN}(x, y|\sigma^2, \sigma_0^2) = \sigma^2 xy + \sigma_0^2,$$

com $\sigma^2 > 0$ e $\sigma_0^2 \geq 0$. Por fim, a função de covariância trigonométrica de suporte compacto

$$k_{CST}(x, y|\ell, \sigma^2) = \frac{2 + \cos\left(2\pi\frac{|x-y|}{\ell}\right)}{3}\left(1 - \frac{|x-y|}{\ell}\right)\sigma^2 + \frac{\sigma^2}{2\pi}\sin\left(2\pi\frac{|x-y|}{\ell}\right)$$

se $|x - y| < \ell$ e

$$k_{CST}(x, y|\ell, \sigma^2) = 0$$

se $|x - y| \geq \ell$, com $\ell, \sigma^2 > 0$.

Para os hiperparâmetros de cada função de covariância, escolhemos tais de forma a maximizar a marginal likelihood de cada modelo. Lembre que, por (3.5), temos neste caso que

$$\log p(Y|X) = -\frac{1}{2}\left(\log\det\left(K + 10^{-3}I\right) + y^T\left(K + 10^{-3}I\right)^{-1} y + n\log(2\pi)\right),$$

onde y é o vetor com os outputs y_i em suas entradas. De maneira semelhante ao feito na seção 4.4, podemos calcular o gradiente de $\log p(Y|X)$ em relação aos hiperparâmetros, já que cada função de covariância apresentada é diferenciável em relação a cada um de seus hiperparâmetros. Usamos o discutido e a função *minimize* do pacote *scipy.optimize* no Python para determinar os hiperparâmetros. Apresentamos os valores encontrados na figura 6 com cinco casas decimais de precisão.

Devido a estarmos trabalhando com um dataset pequeno e todas as funções de covariância apresentadas terem tempos computacionais para uma avaliação semelhantes, decidimos não discutir o tempo computacional de cada escolha neste problema.

Apresentamos na figura 6 gráficos que ilustram as distribuições posteriores obtidas para cada função de covariância considerada, onde os gráficos são construídos como discutido na seção 3.2.1 com a adição do gráfico de f para comparação.

Montamos também a tabela 1, onde na primeira linha temos a distância entre a função f e a função de valor médio da distribuição posterior $m_{posterior}$ do modelo com a função de covariância correspondente à coluna (onde os hiperparâmetros são os apresentados na figura 6). Essa distância é dada por

$$\|f - m_{posterior}\|_2 = \sqrt{\int_{2.5}^{5} \left(f(x) - m_{posterior}(x)\right)^2 \, dx}. \tag{6.2}$$

Como nosso objetivo é estimar a função f e estamos tomando $m_{posterior}$ como o estimador pontual para o problema de regressão, podemos entender $\|f - m_{posterior}\|_2$ como uma medida do erro do nosso estimador. Já na segunda linha da tabela 1, temos o valor da log-marginal likelihood $\log p(Y|X)$ do modelo com a função de covariância correspondente à coluna.

Como discutido na seção 3.4.2 sobre as funções de covariância polinomiais, quando tomamos o hiperparâmetro $p = 1$, temos que a função de valor médio posterior $m_{posterior}$ é linear. Por causa disso, temos que o modelo onde utilizamos a função k_{LIN} fornece uma regressão linear para o problema. É interessante observar que a região de confiança da imagem 6c ilustra a região de confiança da regressão linear em si, e não para os valores da f verdadeira, já que escolher k_{LIN} como função de covariância é de certo modo equivalente a implicitamente "assumir" que a função f é linear. Isso explica a discrepância da log-marginal likelihood desse modelo para a dos demais casos. Como a região de confiança é relativamente pequena, podemos interpretar que o método está relativamente confiante que $m_{posterior}$ é a melhor função linear que descreve os dados segundo o modelo proposto.

Observando a tabela 1, vemos que o modelo utilizando k_{NN} obteve o menor erro. Esse resultado era esperado, uma vez que tanto k_{SE} e k_{CST} são funções de covariância estacionárias, e por isso k_{NN}, uma função de covariância não estacionária, consegue capturar melhor como a função f varia ao longo do intervalo.

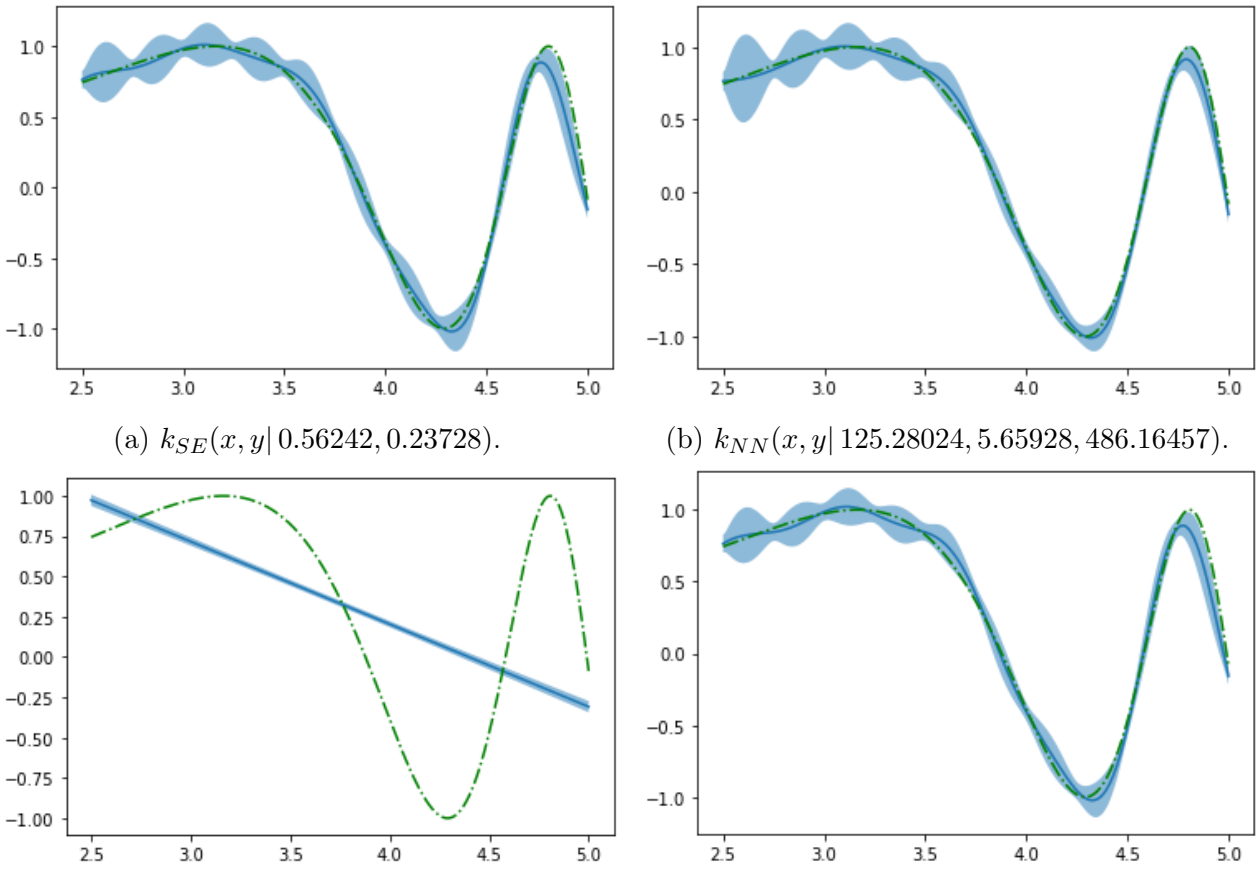
Por outro lado, o modelo utilizando k_{SE} obteve a maior marginal likelihood. Uma possível interpretação para o que observamos é que o modelo utilizando k_{NN} está sendo penalizado por ser mais complexo (ter uma função de covariância não estacionária com mais parâmetros que os outros), e por isso esse critério está preferindo um modelo menos complexo mas com resultados não tão distantes do ótimo encontrado.

Por fim, observamos que os modelos utilizando k_{SE} e k_{CST} obtiveram resultados semelhantes tanto em ambos os critérios apresentados quanto visualmente no apresentado na figura 6. Esse é um resultado interessante, pois temos que k_{CST} pode performar de forma relativamente semelhante a k_{SE} e ainda aproveitar do fato de ser uma função de covariância com suporte compacto, o que gera matrizes de covariância esparsas e de banda, propriedades que podem ser aproveitadas em dataset maiores.

6.2 *Iris Dataset*

Nessa seção vamos utilizar o dataset clássico *Iris*. Nele temos três classes diferentes de plantas iris: iris-Setosa, iris-Versicolor e iris-Virginica. Nosso objetivo é classificar cada espécime de acordo com quatro atributos: comprimento da sépala, largura da sépala, comprimento da pétala e largura da pétala. Os dados utilizados foram retirado do repositório *UCI Machine Learning* (DUA; GRAFF, 2017).

O dataset contém 150 pares de inputs e outputs. Separamos 20% (30 pares) para um conjunto de teste, e os restantes $n = 120$ pares compõem o conjunto de treinamento que vamos aplicar os métodos para classificação por processos gaussianos apresentados. Para mais detalhes



(a) $k_{SE}(x, y|\, 0.56242, 0.23728)$. (b) $k_{NN}(x, y|\, 125.28024, 5.65928, 486.16457)$.

(c) $k_{LIN}(x, y|\, 0.26275, 5.08217)$. (d) $k_{CST}(x, y|\, 0.83478, 0.55634)$.

Figura 6 – Gráfico da função de valor esperado da distribuição posterior com função de covariância (a) k_{SE} (b) k_{NN} (c) k_{LIN} (d) k_{CST} (linhas contínuas) com os hiperparâmetros indicados, onde as regiões sombreadas representando os intervalos de confiança simétricos de 95% para cada valor $f(x)$ ao redor do valor esperado, e o gráfico da função f definida em (6.1) (linha segmentada e pontilhada).

	k_{SE}	k_{NN}	k_{LIN}	k_{CST}
$\|f - m_{posterior}\|_2$	0.11468	0.06906	0.91885	0.11025
$\log p(Y\|X)$	-9.75609	-19.65281	-1591.79324	-9.80073

Tabela 1 – Tabela com as distâncias entre f e as funções de valor esperado posterior $m_{posterior}$, como definido em (6.2), na primeira linha e a log-marginal likelihood $\log p(Y|X)$ na segunda linha, onde cada coluna indica a função de covariância usada para a regressão (com os hiperparâmetros apresentados na figura 6).

dessa divisão, consulte o apêndice A.6.2.

Para ambos os modelos considerados, vamos escolher $[k_{SE}(x, y|\ell_1), k_{SE}(x, y|\ell_2), k_{SE}(x, y|\ell_3)]$ como o conjunto das funções de covariância de cada classe, onde

$$k_{SE}(x, y|\ell) = \exp\left(-\frac{\|x - y\|^2}{2\ell^2}\right),$$

i.e., cada classe tem k_{SE} como função de covariância com um hiperparâmetro ℓ distinto, totalizando três hiperparâmetros em cada um dos modelos.

Dois dos algoritmos apresentados serão utilizados: o algoritmo *nested expectation propagation* (NEP), apresentado na seção 4.5, e o algoritmo de aproximação de Laplace (LA) utilizando a likelihood softmax, apresentado na seção 5.1.1.

Em ambos os casos, os hiperparâmetros foram escolhidos de forma a maximizar a marginal likelihood de cada método. Assim como na seção anterior, utilizamos a função *minimize* do pacote *scipy.optimize* no Python para determinar os hiperparâmetros. Algo para se observar é que não temos a garantia de convergência para o algoritmo NEP, e, por causa disso, podem existir conjuntos de parâmetros onde o algoritmo falha em achar um ponto fixo. Isso pode ser um empecilho para determinar os hiperparâmetros. Um comportamento interessante observado é a convergência do algoritmo para um par de pontos de modo a alternar entre tais a cada iteração, ficando "preso" nesse ciclo. Apesar disso, fomos bem-sucedidos em determinar hiperparâmetros da forma a maximizar a marginal likelihood em ambos os modelos neste caso. Os hiperparâmetros utilizados foram $(\ell_1, \ell_2, \ell_3) = (1.06086403, 1.71082538, 1.73546152)$ para o algoritmo NEP e $(\ell_1, \ell_2, \ell_3) = (1.01290655, 1.66673504, 1.34826497)$ para o algoritmo LA.

Na tabela 2 se encontra o número de erros de classificação do conjunto de teste de cada método, onde o algoritmo 3 foi usado para estimar as probabilidades de cada input do conjunto de teste pertencer a cada uma das classes para o modelo usando o algoritmo NEP e o algoritmo 6 para o modelo usando o algoritmo LA. A partir destas probabilidades, definimos a classe de maior probabilidade como o output estimado. Temos também a aproximação da log-marginal likelihood $\log q(Y|X)$ de cada algoritmo (calculada usando o algoritmo 4 para o algoritmo NEP e o algoritmo 5 para o algoritmo LA), assim como a aproximação da log-marginal likelihood exata $\log p(Y|X)$ de cada modelo. Foi utilizado a formulação mais simples do método de Monte Carlo, apresentada na seção 4.5.1, para o cálculo das aproximações $\log p(Y|X)$, já que

$$p(Y|X) = \int p(Y|f)p(f|X)\, df = \mathbb{E}_{f\sim N(0,K)}[p(Y|f)].$$

Por fim, apresentamos os tempos computacionais médios dos algoritmos que estimam a aproximação posterior de cada modelo, i.e., do algoritmo 2 para NEP e o algoritmo 5 para LA. A tolerância escolhida para o algoritmo 2 foi de 10^{-5} e para o algoritmo 5 foi de 10^{-8}, e o tempo apresentado é a média do tempo computacional de sete execuções do código. Como ambos os métodos seguem os mesmos princípios para a classificação, escolhemos por não comparar tais tempos computacionais. Entretanto, apontamos que o uso das variáveis de controle, como feito no algoritmo 3, foi bem-sucedido em diminuir a variância dos estimados, necessitando assim de um número consideravelmente menor de amostras e reduzindo o tempo computacional

significativamente. Como tal método não foi implementado no algoritmo 6, temos que uma comparação direta dos tempos computacionais de cada algoritmo para classificação seria injusta.

	NEP	**LA**
Número de erros de classificação	0	0
$\log q(Y\|X)$	-38.46614	-45.01823
$\log p(Y\|X)$	-54.16229	-55.51399
Tempo computacional médio	7.22 s	0.856 s

Tabela 2 – Tabela com o número de erros de classificação do conjunto de teste de cada modelo na primeira linha, a aproximação da log-marginal likelihood $\log q(Y|X)$ de cada algoritmo na segunda linha, a aproximação da log-marginal likelihood exata $\log p(Y|X)$ de cada modelo na terceira linha e os tempos computacionais médios referentes aos algoritmos 2 e 5 medidos em segundos na quarta linha.

Analisando a tabela 2, vamos que ambos os métodos foram bem-sucedidos em classificar corretamente todos os inputs do conjunto de teste. Vemos também que tanto o algoritmo NEP quanto o modelo utilizando a likelihood *multinomial probit* apresentam uma marginal likelihood maior que o obtido para o algoritmo LA e para o modelo usando a likelihood *softmax*. Entretanto, temos que as marginal likelihoods exatas de cada modelo são próximas e que a aproximação da marginal likelihood do algoritmo LA está mais perto do valor exato do que a aproximação do algoritmo NEP.

Mais importante ainda, temos que o algoritmo LA foi consideravelmente computacionalmente mais rápido do que o algoritmo NEP. Portanto, se as métricas mais relevantes são o número de classificações corretas e o tempo computacional, o algoritmo LA apresentou um desempenho melhor que o do algoritmo NEP.

Com o intuito de avaliar a qualidade da aproximação gerada por cada algoritmo, utilizamos o método MCMC, discutido na seção 5.2, para gerar amostras da distribuição posterior de cada modelo utilizando o algoritmo 7.

Com essas amostras, queremos estimar inicialmente o desvio do valor esperado das aproximações, $\hat{f} = \mathbb{E}_{Q(f|X,Y)}[f]$, do valor esperado exato:

$$bias(\hat{f}) = \left\| \hat{f} - \mathbb{E}_{P(f|X,Y)}[f] \right\|.$$

Queremos estimar também o *mean squared error* (MSE) de $\hat{f}$ que, como discutido na seção 2.2.1, é dado por:

$$\mathrm{MSE}(\hat{f}) = \mathbb{E}_{P(f|X,Y)}\left[\left\| \hat{f} - f \right\|^2 \right] = \mathrm{tr}(var_{P(f|X,Y)}(f|X,Y)) + (bias(\hat{f}))^2.$$

Com isso, estimamos também $\mathrm{tr}(var_{P(f|X,Y)}(f|X,Y))$ e comparamos com o valor obtido pelas aproximações, $\mathrm{tr}(\hat{\Sigma})$ com $\hat{\Sigma} = var_{Q(f|X,Y)}(f)$. Os valores de tais aproximações podem ser encontrados na tabela 3.

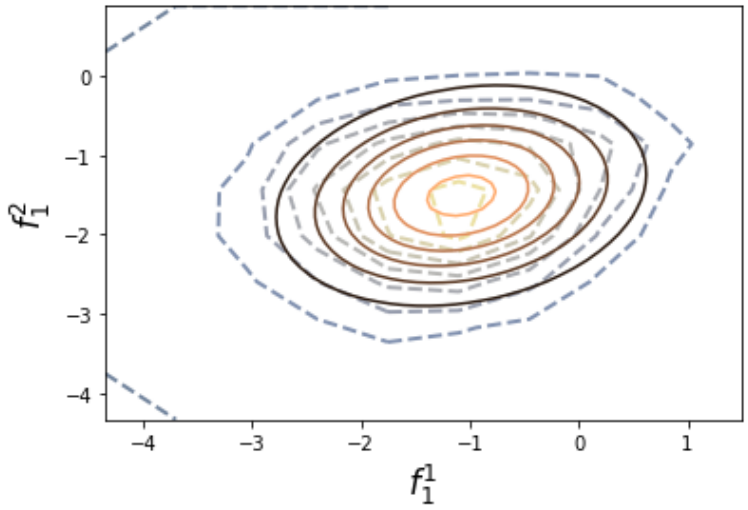
	NEP	**LA**
$bias(\hat{f})$	1.31149	1.98801
$\text{MSE}(\hat{f})$	195.53224	216.96302
$\text{tr}(var_{P(f\|X,Y)}(f))$	193.81222	213.01081
$\text{tr}(\hat{\Sigma})$	189.99493	206.66086

Tabela 3 – Tabela com os valores aproximados de *bias* e MSE do valor esperado da cada método $\hat{f} = \mathbb{E}_{Q(f|X,Y)}[f]$, assim como as aproximações para o traço da matriz de covariância exata $var_{P(f|X,Y)}(f|X,Y)$ e o traço da matriz de covariância de cada método $\hat{\Sigma} = var_{Q(f|X,Y)}(f)$.

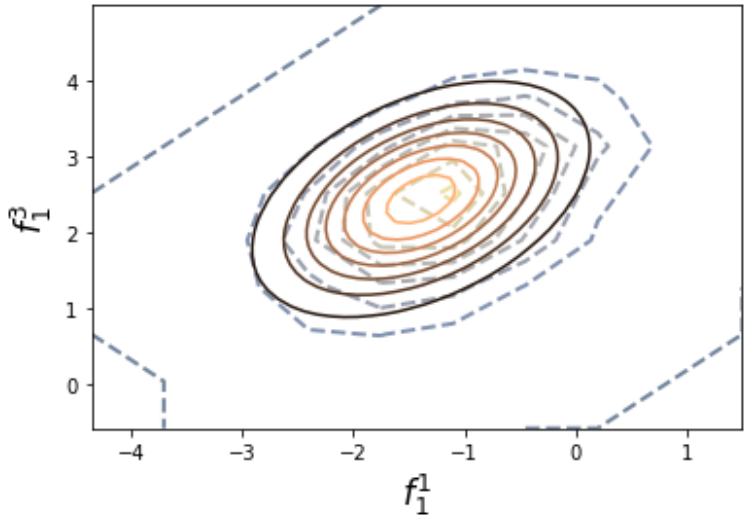
Com essas amostras, geramos também as imagens presentes nas figuras 7 e 8, que mostram a sobreposição das curvas de nível do histograma das amostras das variáveis latentes correspondentes ao primeiro input do conjunto de treinamento (f_1^1, f_1^2 e f_1^3) e das curvas de nível da densidade da aproximação obtida.

Observe que os valores aproximados pelos algoritmos na tabela 3 são razoavelmente próximos as aproximações dos valores exatos. Ainda mais, temos que ambos os métodos apresentaram uma performance semelhante nas medidas apresentadas. O mesmo se aplica para as curvas de nível apresentadas nas figuras 7 e 8.

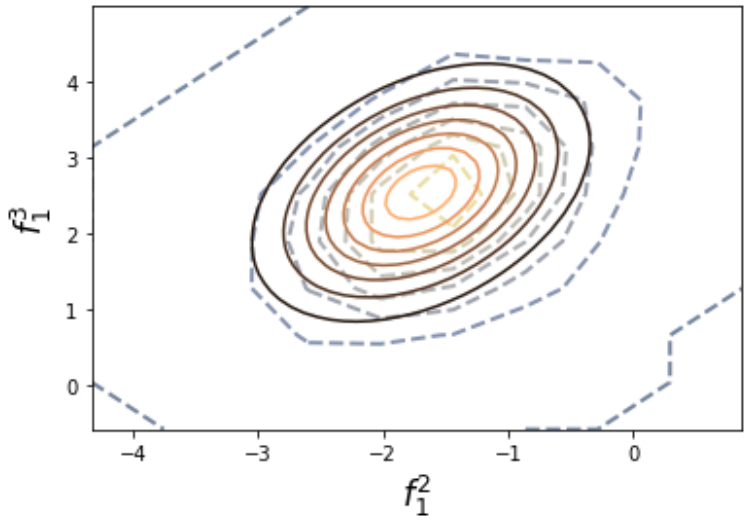
Em conclusão, ambos os algoritmos foram bem-sucedidos em resolver o problema de classificação e em gerar aproximações aceitáveis da distribuição posterior. Em contrapartida, apenas este experimento com as métricas apresentadas não é o suficiente para averiguar a qualidade da aproximação da distribuição posterior gerada pelos métodos de uma forma mais geral. Mais experimentos precisariam ser realizados nessa direção.



(a) Variáveis latente f_1^1 e f_1^2.

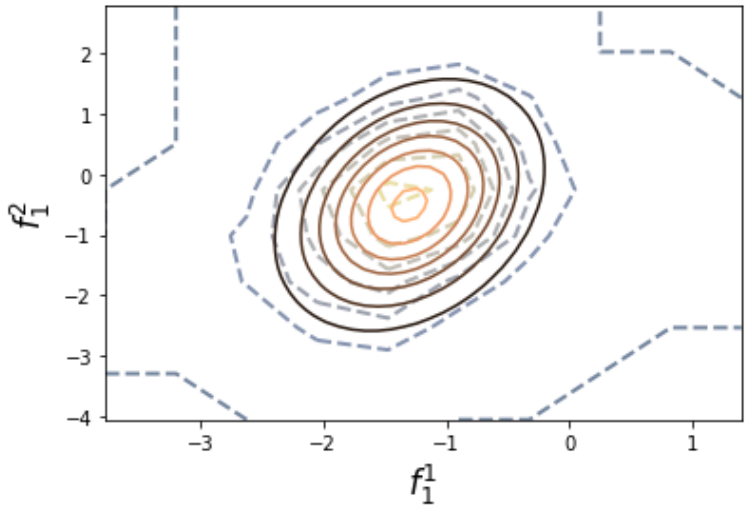


(b) Variáveis latente f_1^1 e f_1^3.

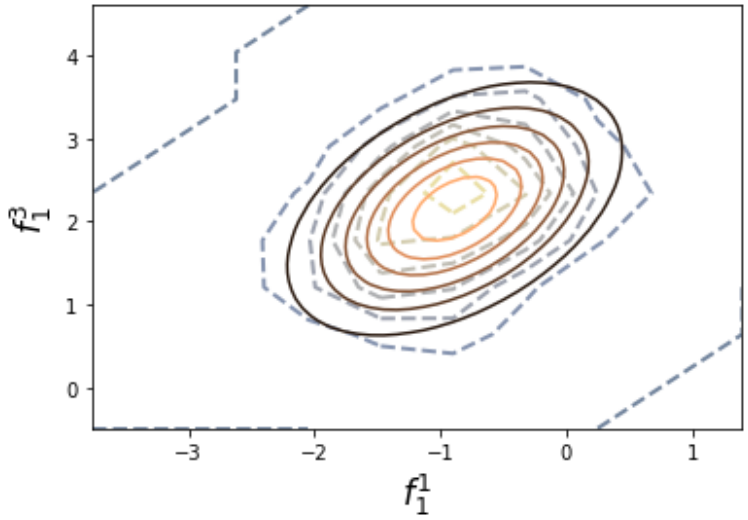


(c) Variáveis latente f_1^2 e f_1^3.

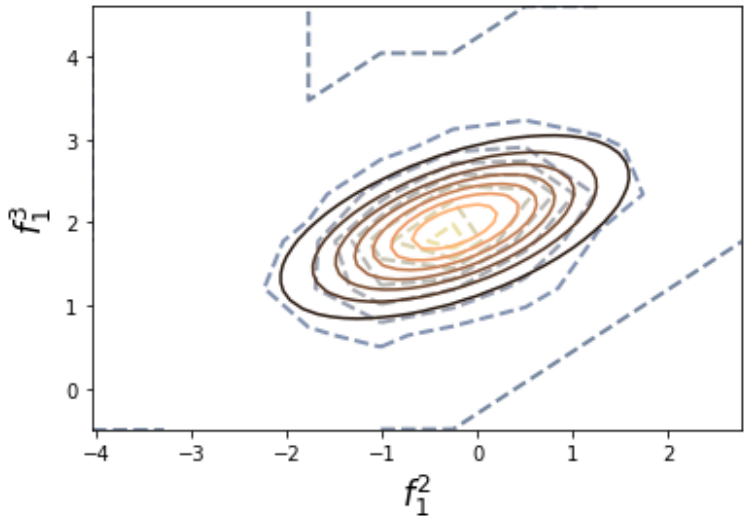
Figura 7 – Sobreposição das curvas de nível do histograma das amostras das variáveis latentes indicadas geradas pelo algoritmo 7 utilizando a likelihood *softmax* (linhas segmentadas) e das curvas de nível da densidade da aproximação obtida pelo algoritmo LA (linhas contínuas).



(a) Variáveis latente f_1^1 e f_1^2.



(b) Variáveis latente f_1^1 e f_1^3.



(c) Variáveis latente f_1^2 e f_1^3.

Figura 8 – Sobreposição das curvas de nível do histograma das amostras das variáveis latentes indicadas geradas pelo algoritmo 7 utilizando a likelihood *multinomial probit* (linhas segmentadas) e das curvas de nível da densidade da aproximação obtida pelo algoritmo NEP (linhas contínuas).

7 Conclusão

Como podemos ver nos experimentos realizados na seção 6, os métodos apresentados por processos gaussianos se mostraram viáveis e foram bem-sucedidos em resolver os problemas de aprendizado supervisionado, trazendo todos os benefícios de trabalhar dentro de uma abordagem Bayesiana e sua interpretabilidade.

Como pode-se observar na seção 6.1 e pelo discutido na seção 3.4, temos que as funções de covariância dos processos gaussianos não só ditam propriedades dos processos em si como também são de grande relevância para a modelagem quando seguindo métodos por processos gaussianos. Por isso, a biblioteca de funções de covariância apresentada na seção 3.4.2 e as operações com funções de covariância apresentadas na seção 3.4.3 são ferramentas de grande utilidade para tais modelagens. Temos também que, para certas funções de covariância, pode-se ver relações com os métodos por processos gaussianos e outros métodos conhecidos na literatura, como, por exemplo, o discutido para as funções de covariância polinomial e para as funções de covariância de rede neural na seção 3.4.2.

Tanto o método de aproximação expectation propagation, discutido na capítulo 4, quanto o método de aproximação de Laplace, discutido na capítulo 5, se mostraram viáveis e capazes de gerar aproximações da distribuição posterior suficientemente boas para a resolução de problemas de classificação. Entretanto, mais experimentos precisariam ser realizados para averiguar a qualidade de tais aproximações. Investigar tal aspecto desses métodos é relevante por causa que podemos, futuramente, aplicá-los para problemas de regressão com outras funções para a likelihood, como, por exemplo, com um erro seguindo uma distribuição exponencial ou *t-student*. Outro ponto a ser estudado é se há maneiras mais eficientes de se implementar o método de expectation propagation, reduzindo o tempo computacional.

Um ponto não abordado, mas não menos relevante, é como adaptar os métodos e estratégias apresentadas para os chamados problemas de *big data*, problemas onde o dataset tem um número exorbitante de pontos. Em alguns momentos do texto apontamos para o possível uso de funções de covariância com suporte compacto para obter uma matriz de covariância esparsa, aproveitando essa estrutura para adaptar os métodos propostos. Esta é apenas uma proposta de estratégia para atacar tais problemas. Vários outras estratégias são apresentadas e discutidas em (WILLIAMS; RASMUSSEN, 2006, Cap. 8). Ainda não temos resultados definitivos sobre esse tema, esta é uma área ativa de pesquisa.

Por fim, além de estudar o uso de diferentes funções para a likelihood, podemos estudar o uso de diferentes processos, seguindo diferentes distribuições. Isso é interessante para a modelagem de problemas onde outras distribuições seriam mais "adequadas" ou mais "precisas" naquele cenário.

Em Conclusão, os métodos por processos gaussianos se mostraram úteis e atrativos, uma área em desenvolvimento com muito potencial.

Referências

ABRAMOWITZ, M.; STEGUN, I. A. *Handbook of mathematical functions with formulas, graphs, and mathematical tables*. [S.l.]: US Government printing office, 1964. v. 55.

ADLER, R. J. *The Geometry of Random Fields*. [S.l.]: John Wiley Sons Inc, 1981. (Wiley Series in Probability and Statistics). ISBN 0471278440,9780471278443.

COX, D. R. *Principles of statistical inference*. [S.l.]: Cambridge university press, 2006.

DUA, D.; GRAFF, C. *UCI Machine Learning Repository*. 2017. Disponível em: <http://archive.ics.uci.edu/ml>.

FIEDLER, M. Notes on hilbert and cauchy matrices. *Linear algebra and its Applications*, North-Holland, v. 432, n. 1, p. 351–356, 2010.

GILKS, W.; RICHARDSON, S.; SPIEGELHALTER, D. *Markov Chain Monte Carlo in Practice*. Taylor & Francis, 1995. (Chapman & Hall/CRC Interdisciplinary Statistics). ISBN 9780412055515. Disponível em: <https://books.google.com.br/books?id=TRXrMWY_i2IC>.

GOLUB, G.; LOAN, C. V. *Matrix Computations*. [S.l.]: Johns Hopkins University Press, 2013. (Johns Hopkins Studies in the Mathematical Sciences).

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The elements of statistical learning: data mining, inference, and prediction*. [S.l.]: Springer Science & Business Media, 2009.

KARATZAS, I.; SHREVE, S. *Brownian motion and stochastic calculus*. [S.l.]: springer, 2014. v. 113.

KULLBACK, S. *Information theory and statistics*. [S.l.]: Courier Corporation, 1997.

LIMA, E. L. *Curso de Análise, vol. 2*. [S.l.]: IMPA, 2015.

LIMA, E. L. *Curso de Análise, vol. 1*. [S.l.]: IMPA, 2017.

MELKUMYAN, A.; RAMOS, F. A sparse covariance function for exact gaussian process inference in large datasets. In: *IJCAI*. [S.l.: s.n.], 2009. v. 9, p. 1936–1942.

MINKA, T. P. Expectation propagation for approximate bayesian inference. In: *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001. (UAI '01), p. 362–369. ISBN 1558608001.

MINKA, T. P. *A family of algorithms for approximate Bayesian inference*. Tese (Doutorado) — Massachusetts Institute of Technology, 2001.

NEAL, R. *Bayesian Learning for Neural Networks*. [S.l.]: Springer New York, 1996. v. 118. (Lecture Notes in Statistics, v. 118).

NOCEDAL, J.; WRIGHT, S. *Numerical optimization*. [S.l.]: Springer Science & Business Media, 2006.

PACIOREK, C. J. *Nonstationary Gaussian processes for regression and spatial modelling*. Tese (Doutorado) — Citeseer, 2003.

PACIOREK, C. J.; SCHERVISH, M. J. Nonstationary covariance functions for gaussian process regression. In: CITESEER. *NIPS*. [S.l.], 2003. p. 273–280.

PRESS, W. H. et al. *Numerical recipes 3rd edition: The art of scientific computing*. [S.l.]: Cambridge university press, 2007.

RIIHIMÄKI, J.; JYLÄNKI, P.; VEHTARI, A. Nested expectation propagation for gaussian process classification with a multinomial probit likelihood. *Journal of Machine Learning Research*, v. 14, n. Jan, p. 75–109, 2013.

RIZZO, M. L. *Statistical computing with R*. [S.l.]: CRC Press, 2019.

SEEGER, M. *Expectation propagation for exponential families*. [S.l.], 2005.

SHANNON, C. E. A mathematical theory of communication. *The Bell system technical journal*, Nokia Bell Labs, v. 27, n. 3, p. 379–423, 1948.

STRANG, G. *Linear Algebra and Its Applications*. [S.l.]: Harcourt, Brace, Jovanovich, Publishers, 1988. ISBN 9780155510050.

TAO, T. An introduction to measure theory. American Mathematical Society Providence, RI, 2011.

TONG, Y. L. *The multivariate normal distribution*. New York: Springer-Verlag, 1990. (Springer series in statistics). ISBN 0387970622.

WANG, Z. et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, IEEE, v. 13, n. 4, p. 600–612, 2004.

WENDLAND, H. *Scattered Data Approximation*. [S.l.]: Cambridge University Press, 2004. (Cambridge Monographs on Applied and Computational Mathematics).

WILLIAMS, C. K. Computation with infinite neural networks. *Neural Computation*, MIT Press, v. 10, n. 5, p. 1203–1216, 1998.

WILLIAMS, C. K.; RASMUSSEN, C. E. *Gaussian processes for machine learning*. [S.l.]: MIT press Cambridge, MA, 2006. v. 2.

WILLIAMS, D. *Probability with martingales*. [S.l.]: Cambridge university press, 1991.

WILLIAMS, P. M. Bayesian conditionalisation and the principle of minimum information. *The British Journal for the Philosophy of Science*, JSTOR, v. 31, n. 2, p. 131–144, 1980.

A Apêndice

A.1 Distribuições Gaussianas

Um vetor aleatório $X \in \mathbb{R}^d$ segue uma distribuição gaussiana (ou normal) se sua distribuição tem uma densidade e esta é da forma

$$\phi(x|\mu,\Sigma) = (2\pi)^{-\frac{d}{2}} \det(\Sigma)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right), \tag{A.1}$$

onde $\mu \in \mathbb{R}^d$ é chamado de vetor de valor esperado e Σ é uma matriz $d \times d$ simétrica positiva definida chamada de matriz de covariância. Denotamos $N(\mu,\Sigma)$ como a distribuição induzida por essa densidade, e portanto falamos que $X \sim N(\mu,\Sigma)$. Definimos também $\Phi(x|\mu,\Sigma)$ como a cdf de $N(\mu,\Sigma)$.

Outra parametrização para as distribuições normais é definir $\mathcal{T} = \Sigma^{-1}$ (chamada de precisão) e $\nu = \Sigma^{-1}\mu$ (chamada de locação). Esses são conhecidos como os parâmetros naturais da distribuição normal.

Considere agora X e Y vetores aleatórios tais que,

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim N\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \begin{bmatrix} \Sigma_X & \Sigma_{XY} \\ (\Sigma_{XY})^T & \Sigma_Y \end{bmatrix}\right).$$

A propriedade de marginalização de distribuições gaussianas nos dá que

$$X \sim N(\mu_X, \Sigma_X), \tag{A.2}$$

e a propriedade de condicionamento de distribuições gaussianas nos dá que

$$X|Y \sim N\left(\mu_x + \Sigma_{XY}(\Sigma_Y)^{-1}(Y-\mu_Y),\ \Sigma_X - \Sigma_{XY}(\Sigma_Y)^{-1}(\Sigma_{XY})^T\right). \tag{A.3}$$

Temos também que o produto de duas densidades gaussianas é proporcional a outra densidade gaussiana, seguindo

$$\phi(x|a,A)\phi(x|b,B) = Z\phi(x|c,C),$$

$$\text{onde } C = (A^{-1}+B^{-1})^{-1},\ c = C(A^{-1}a + B^{-1}b), \tag{A.4}$$

$$Z = (2\pi)^{-\frac{d}{2}} \det(A+B)^{-\frac{1}{2}} \exp\left(-\tfrac{1}{2}(a-b)^T(A+B)^{-1}(a-b)\right).$$

Como referência para distribuições normais multivariadas, veja (TONG, 1990).

A.1.1 Distribuições Gaussianas Degeneradas

Para uma distribuição normal unidimensional $N(\mu,\sigma^2)$ com $\mu \in \mathbb{R}$ e $\sigma^2 > 0$, temos que quando σ^2 tende a zero, $N(\mu,\sigma^2)$ tende a distribuição de Dirac δ_μ, i.e., uma distribuição com

pmf $p(\mu) = \mathbb{P}(X = \mu) = 1$ e $p(x) = 0$ para qualquer outro valor real. Por isso, definimos a distribuição gaussiana degenerada unidimensional como $N(\mu, 0) \sim \delta_\mu$.

Considere agora $X \sim N(\mu, \Sigma)$, com $\mu \in \mathbb{R}^d$ e Σ matriz $d \times d$ simétrica positiva definida. Tome a decomposição em valores singulares de $\Sigma = USU^T$ e defina a transformação $U^T X = Y$, assim $Y \sim N(U^T\mu, S) = \prod_{i=1}^d N(Y_i|\mu_{y_i}, s_i)$, onde s_i é o i-ésimo valor singular de Σ, Y_i é a i-ésima entrada de Y e μ_{y_i} é a i-ésima estrada de $U^T\mu$. Observe que, como Y é uma transformação ortogonal de X, temos que $\mathbb{P}(X \in A) = \mathbb{P}(Y \in U^T(A))$, para qualquer conjunto mensurável A de $\mathbb{R}^d$.

A generalização para o caso degenerado onde Σ é uma matriz semipositiva definida de rank r é direta. Definimos a distribuição de $X \sim N(\mu, \Sigma)$ como seguindo, para qualquer conjunto mensurável A de $\mathbb{R}^d$ e seguindo a notação do parágrafo anterior,

$$\mathbb{P}(X \in A) = \int_{U^T(A)} \prod_{i=1}^{r} \phi(y_i|\mu_{y_i}, s_i)\, dy_1 \cdots dy_r d\delta_{\mu_{y_{r+1}}}(y_{r+1}) \cdots d\delta_{\mu_{y_d}}(y_d).$$

A.2 Relações entre a função de covariância e propriedades de um processos estocástico

Vamos aqui demonstrar os teoremas que foram deixados em aberto na seção 3.4.1. Relembre que para um processo estocástico C-dimensional $f = \{f(x)\}_{x\in\mathcal{X}}$, definimos a função de covariância $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{C\times C}$ como

$$k(x, y) = cov(f(x), f(y)) = \mathbb{E}[(f(x) - \mathbb{E}[f(x)])(f(y) - \mathbb{E}[f(y)])^T].$$

Teorema. Um processo estocástico f é contínuo em média quadrática em $x \in \mathcal{X}$ se, e somente se, sua função de valor médio, m, é contínua em x e o traço da função de covariância, $\mathrm{tr}(k)$, é contínuo em (x, x).

Demostração: Primeiramente, vamos assumir que $\mathbb{E}[f(x)] = 0, \forall x \in \mathcal{X}$. Nesse caso, observe que

$$\mathbb{E}\left[\|f(x)\|^2\right] = \mathbb{E}\left[f(x)^T f(x)\right] = \mathbb{E}\left[\mathrm{tr}\left(f(x)f(x)^T\right)\right] = \mathrm{tr}\left(\mathbb{E}\left[f(x)f(x)^T\right]\right) = \mathrm{tr}\left(k(x, x)\right),$$

e que f ser contínuo em média quadrática em $x \in \mathcal{X}$ pela definição dada é equivalente a

$$\lim_{h\to 0} \mathbb{E}\left[\|f(x + h) - f(x)\|^2\right] = 0,$$

e ainda mais, vale que

$$\lim_{(h,s)\to 0} \mathbb{E}\left[\|f(x + h) - f(x + s)\|^2\right] = 0.$$

Assuma agora que f é contínuo em média quadrática em $x \in \mathcal{X}$. Pela desigualdade de Minkowski (WILLIAMS, 1991, pg. 69), temos, para qualquer h tal que $x + h \in \mathcal{X}$,

$$\sqrt{\mathbb{E}\left[\|f(x + h) - f(x)\|^2\right]} \geq \left|\sqrt{\mathbb{E}\left[\|f(x + h)\|^2\right]} - \sqrt{\mathbb{E}\left[\|f(x)\|^2\right]}\right| \geq 0.$$

Assim

$$\lim_{h\to 0}\sqrt{\mathbb{E}\left[\|f(x+h)\|^2\right]} = \sqrt{\mathbb{E}\left[\|f(x)\|^2\right]}$$
$$\Rightarrow \lim_{h\to 0}\mathbb{E}\left[\|f(x+h)\|^2\right] = \mathbb{E}\left[\|f(x)\|^2\right]$$
$$\Rightarrow \lim_{h\to 0}\mathrm{tr}\left(k(x+h,x+h)\right) = \mathrm{tr}\left(k(x,x)\right).$$

Por outro lado, veja que, para qualquer h e s tal que $x+h, x+s \in \mathcal{X}$,

$$\mathbb{E}\left[\|f(x+h)-f(x+s)\|^2\right] = \mathbb{E}\left[\|f(x+h)\|^2 + \|f(x+s)\|^2 - 2 < f(x+h), f(x+s) >\right]$$
$$= \mathbb{E}\left[\|f(x+h)\|^2\right] + \mathbb{E}\left[\|f(x+s)\|^2\right] - 2\mathbb{E}\left[f(x+h)^T f(x+s)\right]$$
$$= \mathrm{tr}\left(k(x+h,x+h)\right) + \mathrm{tr}\left(k(x+s,x+s)\right) - 2\mathrm{tr}\left(\mathbb{E}\left[f(x+h)f(x+s)^T\right]\right)$$
$$= \mathrm{tr}\left(k(x+h,x+h)\right) + \mathrm{tr}\left(k(x+s,x+s)\right) - 2\mathrm{tr}\left(k(x+h,x+s)\right).$$

Fazendo $(h,s)\to 0$, temos

$$0 = 2\mathrm{tr}\left(k(x,x)\right) - 2\lim_{(h,s)\to 0}\mathrm{tr}\left(k(x+h,x+s)\right)$$
$$\lim_{(h,s)\to 0}\mathrm{tr}\left(k(x+h,x+s)\right) = \mathrm{tr}\left(k(x,x)\right).$$

i.e., $\mathrm{tr}\left(k\right)$ é contínuo (x,x).

Por outro lado, assuma que $\mathrm{tr}(k)$ é contínuo em (x,x). Pelo demostrado anteriormente, temos

$$\lim_{h\to 0}\mathbb{E}[\|f(x+h)-f(x)\|^2] = \lim_{h\to 0}\mathrm{tr}\left(k(x+h,x+h)\right) + \mathrm{tr}\left(k(x,x)\right) - 2\mathrm{tr}\left(k(x+h,x)\right)$$
$$= 2\mathrm{tr}\left(k(x,x)\right) - 2\mathrm{tr}\left(k(x,x)\right) = 0,$$

i.e., f é contínuo em média quadrada em x.

Vamos para o caso geral com função de valor médio m qualquer. Defina $\bar{f}(x) = f(x) - m(x), \forall x \in \mathcal{X}$, e veja que $\mathbb{E}[\bar{f}(x)] = 0, \forall x \in \mathcal{X}$, e

$$cov(\bar{f}(x),\bar{f}(y)) = \mathbb{E}\left[\bar{f}(x)\bar{f}(y)^T\right] = \mathbb{E}\left[(f(x)-m(x))(f(y)-m(y))^T\right] = k(x,y),$$

i.e., $\bar{f}$ e f tem a mesma função de covariância.

Assuma primeiramente que f é contínua em média quadrada em x. Portanto, analogamente ao feito no caso anterior, temos

$$\lim_{h\to 0}\mathbb{E}\left[\|f(x+h)\|^2\right] = \mathbb{E}\left[\|f(x)\|^2\right]$$
$$\Rightarrow \lim_{h\to 0}\mathbb{E}\left[f(x+h)\right] = \mathbb{E}\left[f(x)\right]$$
$$\Rightarrow \lim_{h\to 0} m(x+h) = m(x),$$

i.e., a função de valor médio é contínua em x.

Veja também que

$$\mathbb{E}\left[\left\|\bar{f}(x+h)-\bar{f}(x)\right\|^2\right] = \mathbb{E}\left[\|(f(x+h)-f(x)) - (m(x+h)-m(x))\|^2\right]$$

$$= \mathbb{E}\left[\|f(x+h)-f(x)\|^2 + \|m(x+h)-m(x)\|^2 - 2 < (f(x+h)-f(x)), (m(x+h)-m(x)) >\right]$$
$$= \mathbb{E}\left[\|f(x+h)-f(x)\|^2\right] - \|m(x+h)-m(x)\|^2,$$

e portanto, como m é contínua em x e f é contínuo em média quadrática em x, temos que $\bar{f}$ também é contínua em média quadrática em x. Como $\bar{f}$ e f têm a mesma função de covariância, temos pelo caso anterior que $\mathrm{tr}(k)$ é contínua em (x,x).

Finalmente, suponha que m é contínua em x e $\mathrm{tr}(k)$ é contínua em (x,x). Pelos casos anteriores, temos que $\bar{f}$ é contínua em média quadrática em x e que vale

$$\mathbb{E}\left[\left\|\bar{f}(x+h)-\bar{f}(x)\right\|^2\right] + \|m(x+h)-m(x)\|^2 = \mathbb{E}\left[\|f(x+h)-f(x)\|^2\right],$$

e portanto f é contínua em média quadrática em x. □

Teorema. Seja $f = \{f(x)\}_{x\in\mathcal{X}}$ um processo estocástico C-dimensional com $\mathcal{X} \subseteq \mathbb{R}^D$ aberto e com função de valor médio m e função de covariância k.

Se as derivadas $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x,y)$ existem e são contínuas em $\mathcal{X}\times\mathcal{X}$ para cada $i \in \{1,2,\ldots,D\}$ e m é continuamente diferenciável em $\mathcal{X}$, então f é diferenciável em média quadrática em $\mathcal{X}$. Além disso, cada processo $\dfrac{\partial f}{\partial x_i}$ tem sua função de valor médio dada por $\dfrac{\partial m}{\partial x_i}(x)$ e a sua função de covariância dada por $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x,y)$.

Demostração: Como a derivada $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x,y)$ existe em $\mathcal{X}\times\mathcal{X}$ para $i \in \{1,2,\ldots,D\}$, então temos que

$$\frac{\partial^2 k}{\partial x_i \partial y_i}(x,y) = \lim_{h\to 0}\frac{1}{h}\left[\frac{\partial k}{\partial x_i}(x, y+he_i) - \frac{\partial k}{\partial x_i}(x,y)\right]$$
$$= \lim_{(s,h)\to 0}\frac{1}{sh}\left[k(x+se_i, y+he_i) - k(x, y+he_i) - k(x+se_i, y) + k(x,y)\right],$$

o que nos leva a

$$\lim_{(s,h)\to 0} cov\left(\frac{f(x+se_i)-f(x)}{s}, \frac{f(y+he_i)-f(y)}{h}\right) =$$
$$= \lim_{(s,h)\to 0}\frac{1}{sh}\left[cov(f(x+se_i), f(y+he_i)) - cov(f(x), f(y+he_i)) \quad -cov(f(x+se_i), f(y)) \right.$$
$$\left. +cov(f(x), f(y))\right]$$
$$= \lim_{(s,h)\to 0}\frac{1}{sh}\left[k(x+se_i, y+he_i) - k(x, y+he_i) - k(x+se_i, y) + k(x,y)\right] = \frac{\partial^2 k}{\partial x_i \partial y_i}(x,y).$$

Para compactar a notação, vamos definir, para $x \in \mathcal{X}$ qualquer e $h > 0$,

$$f_x(h) = \frac{f(x+he_i)-f(x)}{h}$$

e

$$m_x(h) = \frac{m(x+he_i)-m(x)}{h}.$$

Assim, temos

$$\mathbb{E}\left[\|f_x(h)-f_x(s)\|^2\right] = \mathbb{E}\left[\|(f_x(h)-m_x(h)) - (f_x(s)-m_x(s)) + (m_x(h)-m_x(s))\|^2\right]$$

$$= \mathbb{E}\left[\|f_x(h) - m_x(h)\|^2 + \|f_x(s) - m_x(s)\|^2 \quad -2 < (f_x(h) - m_x(h)), (f_x(s) - m_x(s)) >\right] + \|m_x(h) - m_x(s)\|^2$$

$$= \text{tr}(cov(f_x(h), f_x(h)) + cov(f_x(h), f_x(h)) - 2cov(f_x(h), f_x(s))) + \|m_x(h) - m_x(s)\|^2$$

Como temos que m é diferenciável, temos $\lim_{(s,h)\to 0} \|m_x(h) - m_x(s)\|^2 = 0$ e, portanto,

$$\lim_{(s,h)\to 0} \mathbb{E}\left[\|f_x(h) - f_x(s)\|^2\right] = \text{tr}\left(\frac{\partial^2 k}{\partial x_i \partial y_i}(x,x) + \frac{\partial^2 k}{\partial x_i \partial y_i}(x,x) - 2\frac{\partial^2 k}{\partial x_i \partial y_i}(x,x)\right) = 0.$$

Esse limite nos mostra que para qualquer sequência $\{h_n\}_{n=1}^{\infty}$ com $\lim_{n\to\infty} h_n = 0$, temos que a sequência $\{f_x(h_n)\}_{n=1}^{\infty}$ é de Cauchy em L^2, o espaço das classes de equivalência de vetores aleatório com $\mathbb{E}\left[\|g\|^2\right] < \infty$ e idênticos quase certamente. Para uma definição mais precisa e mais propriedades, apontamos a referência (WILLIAMS, 1991, cap. 6). Precisamos apenas do fato que esse espaço é completo (WILLIAMS, 1991, pg. 65) para concluir que existe um vetor aleatório $\dfrac{\partial f}{\partial x_i}(x)$ tal que

$$\frac{f(x + he_i) - f(x)}{h} \xrightarrow{m.s.} \frac{\partial f}{\partial x_i}(x) \text{ quando } i \to \infty.$$

Veja que esse vetor não é único, já que, pela construção de L^2, $\{f_x(h_n)\}_{n=1}^{\infty}$ vai convergir para um vetor aleatório se, e somente se, ele é idêntico quase sempre a $\dfrac{\partial f}{\partial x_i}(x)$. Isso vai implicar em todos compartilharem uma única distribuição, e temos também que qualquer processo que siga essa distribuição também vai satisfazer a definição. Por causa disso, quando nos referirmos ao vetor aleatório $\dfrac{\partial f}{\partial x_i}(x)$, estamos nos referindo a qualquer vetor aleatório que segue essa única distribuição.

Finalmente, pela definição de diferenciabilidade em média quadrática, temos que

$$\mathbb{E}\left[\frac{\partial f}{\partial x_i}(x)\right] = \lim_{h\to 0} \mathbb{E}\left[f_x(h)\right] = \lim_{h\to 0} m_x(h) = \frac{\partial m}{\partial x_i}(x).$$

E também que

$$cov\left(\frac{\partial f}{\partial x_i}(x), \frac{\partial f}{\partial x_i}(y)\right) = \lim_{(s,h)\to 0} cov\left(f_x(s), f_y(h)\right) = \frac{\partial^2 k}{\partial x_i \partial y_i}(x,y).$$

□

Teorema. Seja $f \sim \mathcal{GP}_C(m,k)$ definido em $\mathcal{X} \subseteq \mathbb{R}^D$ aberto tal que as derivadas $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x,y)$ existem e são contínuas em $\mathcal{X} \times \mathcal{X}$ para cada $i \in \{1, 2, \ldots, D\}$ e m é continuamente diferenciável em $\mathcal{X}$. Assim, f é diferenciável em média quadrada e cada $\dfrac{\partial f}{\partial x_i}$ também é um processo gaussiano dado por

$$\frac{\partial f}{\partial x_i} \sim \mathcal{GP}_C\left(\frac{\partial m}{\partial x_i}, \frac{\partial^2 k}{\partial x_i \partial y_i}\right).$$

Demostração: Pelo teorema anterior, já temos que f é diferenciável em média quadrática e que $\dfrac{\partial f}{\partial x_i}$ tem sua função de valor médio dada por $\dfrac{\partial m}{\partial x_i}(x)$ e a sua função de covariância dada por $\dfrac{\partial^2 k}{\partial x_i \partial y_i}(x, y)$. Basta provar agora que $\dfrac{\partial f}{\partial x_i}$ é um processo gaussiano.

Pelas observações feitas na demostração anterior em relação a unicidade de $\dfrac{\partial f}{\partial x_i}(x)$, temos que se mostrarmos que um desses processos é gaussiano, então qualquer processo $\dfrac{\partial f}{\partial x_i}$ é gaussiano.

Para isso, observe primeiro que, para $x \in \mathcal{X}$ e $i \in \{1, 2, \ldots, D\}$ e seguindo a notação apresentada para $f_x(h)$ e $m_x(h)$, temos

$$f_x(h) \sim N\left(m_x(h), cov(f_x(h), f_x(h))\right).$$

Como a pdf de uma distribuição normal é contínua, temos que

$$\lim_{h\to 0} \phi(s|m_x(h), cov(f_x(h), f_x(h))) = \phi\left(s \left| \frac{\partial m}{\partial x_i}(x), \frac{\partial^2 k}{\partial x_i \partial y_i}(x, x)\right.\right), \forall s \in \mathbb{R}^D.$$

Como cada pdf é limitada com seu ponto de máxima sendo $m_x(h)$, temos também a convergência das cdf,

$$\lim_{h\to 0} \Phi(s|m_x(h), cov(f_x(h), f_x(h))) = \Phi\left(s \left| \frac{\partial m}{\partial x_i}(x), \frac{\partial^2 k}{\partial x_i \partial y_i}(x, x)\right.\right), \forall s \in \mathbb{R}^C.$$

Por outro lado, a convergência em média quadrática implica na convergência em distribuição, isto é, se $F_{\partial x_i}(s|x)$ é a cdf de $\dfrac{\partial f}{\partial x_i}(x)$, então

$$\lim_{h\to 0} \Phi(s|m_x(h), cov(f_x(h), f_x(h))) = F_{\partial x_i}(s|x), \forall s \in \mathbb{R}^C.$$

Portanto $F_{\partial x_i}(s|x) = \Phi\left(s \left| \dfrac{\partial m}{\partial x_i}(x), \dfrac{\partial^2 k}{\partial x_i \partial y_i}(x, x)\right.\right), \forall s \in \mathbb{R}^C.$

Isso prova que

$$\frac{\partial f}{\partial x_i}(x) \sim N\left(\frac{\partial m}{\partial x_i}(x), \frac{\partial^2 k}{\partial x_i \partial y_i}(x, x)\right).$$

Como já sabemos a função de covariância de $\dfrac{\partial f}{\partial x_i}$, temos, portanto, pelas propriedades das distribuições gaussianas, que

$$\frac{\partial f}{\partial x_i} \sim \mathcal{GP}_C\left(\frac{\partial m}{\partial x_i}, \frac{\partial^2 k}{\partial x_i \partial y_i}\right).$$

□

A.3 Identidades para Matrizes

Lema 1 (Lema de inversão de matrizes)**.** Considere as matrizes Z, $n \times n$ e invertível, W, $m \times m$ e invertível, U e V, ambas $n \times m$. Neste caso, temos que

$$(Z + UWV^T)^{-1} = Z^{-1} - Z^{-1}U(W^{-1} + V^T Z^{-1} U)^{-1} V^T Z^{-1}. \tag{A.5}$$

Essa identidade também é conhecida como fórmula de Woodbury, Sherman e Morrison (PRESS et al., 2007, pg.80).

Demostração: Observe que

$$(Z + UWV^T)(Z^{-1} - Z^{-1}U(W^{-1} + V^TZ^{-1}U)^{-1}V^TZ^{-1}) =$$

$$= I - U(W^{-1} + V^TZ^{-1}U)^{-1}V^TZ^{-1} + UWV^TZ^{-1} - UWV^TZ^{-1}U(W^{-1} + V^TZ^{-1}U)^{-1}V^TZ^{-1}$$

$$= I + UW(-W^{-1}(W^{-1} + V^TZ^{-1}U)^{-1} + I - V^TZ^{-1}U(W^{-1} + V^TZ^{-1}U)^{-1})V^TZ^{-1}$$

$$= I + UW(I - (W^{-1} + V^TZ^{-1}U)(W^{-1} + V^TZ^{-1}U)^{-1})V^TZ^{-1}$$

$$= I.$$

Isso é suficiente para provar a igualdade. □

Lema 2. Sob as mesmas hipóteses do lema anterior, temos também uma identidade para determinantes, dada por

$$\det\left(Z + UWV^T\right) = \det(Z)\det(W)\det\left(W^{-1} + V^TZ^{-1}U\right). \tag{A.6}$$

Demostração: Primeiramente, considere U e V matrizes $n \times m$ qualquer. Então, observe que

$$\begin{bmatrix} I_n & 0 \\ V^T & I_m \end{bmatrix}\begin{bmatrix} I_n + UV^T & U \\ 0 & I_m \end{bmatrix}\begin{bmatrix} I_n & 0 \\ -V^T & I_m \end{bmatrix} =$$

$$= \begin{bmatrix} I_n + UV^T & U \\ V^T + V^TUV^T & I_m + V^TU \end{bmatrix}\begin{bmatrix} I_n & 0 \\ -V^T & I_m \end{bmatrix}$$

$$= \begin{bmatrix} I_n + UV^T - UV^T & U \\ V^T + V^TUV^T - V^T - V^TUV^T & I_m + V^TU \end{bmatrix}$$

$$= \begin{bmatrix} I_n & U \\ 0 & I_m + V^TU \end{bmatrix}.$$

Dessa igualdade vemos que $\det(I_n + UV^T) = \det(I_m + V^TU)$.

Para o caso geral, usando a igualdade que acabamos de provar, temos que

$$\begin{aligned} \det(Z + UWV^T) &= \det(Z(I + (Z^{-1}U)(WV^T))) \\ &= \det(Z)\det(I + (Z^{-1}U)(WV^T)) \\ &= \det(Z)\det(I + (WV^T)(Z^{-1}U)) \\ &= \det(Z)\det(W)\det(W^{-1} + V^TZ^{-1}U). \end{aligned}$$

Provando a igualdade.

□

Lema 3. Suponha que A, B e C são matrizes $n \times n$ simétricas positivas definidas e a, b e c vetores em $\mathbb{R}^n$ tal que as igualdades

$$C^{-1} = A^{-1} + B^{-1} \tag{A.7}$$

e

$$C^{-1}c = A^{-1}a + B^{-1}b \tag{A.8}$$

são respeitadas. Então

$$(a-b)^T(A+B)^{-1}(a-b) = a^TA^{-1}a + b^TB^{-1}b - c^TC^{-1}c$$

e

$$\log\det(A+B) = \log\det(A) + \log\det(B) - \log\det(C).$$

Demostração: Inicialmente, observe que por (A.5) e pelas hipóteses (A.7) e (A.8) temos que

$$(A+B)^{-1} = B^{-1} - B^{-1}(A^{-1}+B^{-1})^{-1}B^{-1} = B^{-1} - B^{-1}CB^{-1},$$

e que

$$B^{-1}(a-b) = B^{-1}a - B^{-1}b = (C^{-1}-A^{-1})a - (C^{-1}c - A^{-1}a) = C^{-1}(a-c),$$

e também que

$$\begin{aligned}(a-c)^TC^{-1}(a-c) &= c^TC^{-1}c + a^TC^{-1}a - a^TC^{-1}c - c^TC^{-1}a \\ &= c^TC^{-1}c + a^T(A^{-1}+B^{-1})a - a^T(A^{-1}a + B^{-1}b) - (a^TA^{-1} + b^TB^{-1})a \\ &= c^TC^{-1}c - a^TA^{-1}a + a^TB^{-1}a - a^TB^{-1}b - b^TB^{-1}a.\end{aligned}$$

Finalmente, temos

$$\begin{aligned}(a-b)^T(A+B)^{-1}(a-b) &= (a-b)^T(B^{-1}-B^{-1}CB^{-1})(a-b) \\ &= (a-b)^TB^{-1}(a-b) - (B^{-1}(a-b))^TC(B^{-1}(a-b)) \\ &= (a-b)^TB^{-1}(a-b) - (C^{-1}(a-c))^TC(C^{-1}(a-c)) \\ &= (a-b)^TB^{-1}(a-b) - (a-c)^TC^{-1}(a-c) \\ &= b^TB^{-1}b + a^TB^{-1}a - a^TB^{-1}b - b^TB^{-1}a \\ &\quad -(c^TC^{-1}c - a^TA^{-1}a + a^TB^{-1}a - a^TB^{-1}b - b^TB^{-1}a) \\ &= a^TA^{-1}a + b^TB^{-1}b - c^TC^{-1}c.\end{aligned}$$

Provando a primeira igualdade.

Para a segunda igualdade, utilizando (A.6) e (A.7) temos

$$\begin{aligned}\log\det(A+B) &= \log(\det(A)\det(B)\det(A^{-1}+B^{-1})) \\ &= \log\det(A) + \log\det(B) + \log\det(C^{-1}) \\ &= \log\det(A) + \log\det(B) - \log\det(C).\end{aligned}$$

Provando a segunda igualdade. □

A.4 Minimizando a Divergência de Kullback-Leibler para uma Aproximação Normal

Queremos achar parâmetros $\mu \in \mathbb{R}^d$ e Σ, matriz $d \times d$ simétrica positiva definida, de forma a minimizar $D_{KL}(P||N(\mu,\Sigma))$, onde P é uma distribuição absolutamente contínua com pdf

$p(x)$ e tal qual seu primeiro e segundo momento existem e são finitos. Assumimos também que $var_P(X)$ é não-singular.

Observe primeiramente que

$$\log(\phi(x|\mu,\sigma)) = -\frac{1}{2}(d\log(2\pi) + \log(\det(\Sigma)) + (x-\mu)^T\Sigma^{-1}(x-\mu))$$

$$= -\frac{1}{2}(d\log(2\pi) + \log(\det(\Sigma)) + x^T\Sigma^{-1}x - 2\mu^T\Sigma^{-1}x + \mu^T\Sigma^{-1}\mu).$$

Portanto, podemos expandir a divergência como

$$2D_{KL}(P||N(\mu,\Sigma)) = 2\int \log(p(x))p(x)\,dx - 2\int \log(\phi(x|\mu,\Sigma))p(x)\,dx$$

$$= C + \log(\det(\Sigma)) + \mu^T\Sigma^{-1}\mu - 2\mu^T\Sigma^{-1}\mathbb{E}_P[X] + \int x^T\Sigma^{-1}xp(x)\,dx,$$

onde C é uma constante em relação a μ e Σ.

Derivando em função de μ, temos

$$\nabla_\mu[2D_{KL}(P||N(\mu,\Sigma))] = 2\Sigma^{-1}\mu - 2\Sigma^{-1}\mathbb{E}_P[X],$$

e portanto temos como ponto estacionário $\mu = \mathbb{E}_P[X]$. Derivando o gradiente para obter a matriz hessiana, como Σ^{-1} tem que ser positiva definida, temos que esse é um ponto de mínimo da função objetivo para qualquer Σ fixo dentro da restrição imposta.

Aplicando esse ponto à divergência, temos

$$2D_{KL}(P||N(\mathbb{E}_P[X],\Sigma)) = C + \log(\det(\Sigma)) + \int (x-\mathbb{E}_P[X])^T\Sigma^{-1}(x-\mathbb{E}_P[X])p(x)\,dx$$

$$= C + \mathrm{tr}(\log(\Sigma)) + \int \mathrm{tr}\left(\Sigma^{-1}(x-\mathbb{E}_P[X])(x-\mathbb{E}_P[X])^T p(x)\right)\,dx$$

$$= C + \mathrm{tr}(\log(\Sigma)) + \mathrm{tr}\left(\Sigma^{-1}\int (x-\mathbb{E}_P[X])(x-\mathbb{E}_P[X])^T p(x)\,dx\right)$$

$$= C + \mathrm{tr}(\log(\Sigma)) + \mathrm{tr}(\Sigma^{-1}var_P(X)).$$

Derivando agora em relação a Σ, temos

$$\nabla_\Sigma[2D_{KL}(P||N(\mathbb{E}_P[X],\Sigma))] = \Sigma^{-1} - \Sigma^{-1}var_P(X)\Sigma^{-1}.$$

Igualando a zero a procura de pontos estacionários, temos

$$0 = \Sigma^{-1} - \Sigma^{-1}var_P(X)\Sigma^{-1} \Rightarrow var_P(X)\Sigma^{-1} = I$$

$$\Sigma = var_P(X).$$

Desta forma, temos que a distribuição normal $N(\mathbb{E}_P[X], var_P(X))$ minimiza a divergência.

A.5 Entropia

Definimos a entropia de uma distribuição P com pdf/pmf $p(x)$ como

$$H(P) = \mathbb{E}_P[-\log(p(x))] = \int -\log(p(x))\, dP.$$

No caso de uma variável discreta, temos que a entropia é minimizada e igual a zero para uma variável "determinística" (quando existe um x tal que $\mathbb{P}(X = x) = 1$) e maximizada para distribuições uniformes (quando todos os eventos são igualmente provável). Esse é só um exemplo de como esse valor pode ser entendido como uma medida de quanta "incerteza" está presente em uma distribuição, ou o quanto não informativa ela é. Esse conceito foi introduzido por C. E. Shannon em *A Mathematical Theory of Communication* (SHANNON, 1948), onde se pode achar uma discussão mais intensa sobre a motivação dessa definição, resultados e aplicações em tópicos de teoria da informação.

Podemos definir também a entropia cruzada de Q (com pdf/pmf $q(x)$) em relação a P como

$$H_P(Q) = \mathbb{E}_P[-\log(q(x))] = \int -\log(q(x))\, dP.$$

Com isso, podemos decompor a divergência de Kullback-Leibler de Q para P, conhecida também por entropia relativa de P em relação a Q, como

$$D_{KL}(P||Q) = \int \log\left(\frac{p(x)}{q(x)}\right)\, dP = H_P(Q) - H(P),$$

o que deixa ainda mais evidente a não simetria e o papel em que cada distribuição assume, i.e., que a divergência mede o quão "distante" Q está de P do ponto de vista de P, ou quanta informação se perde a escolher a distribuição Q no lugar de P (como quando escolhemos Q para aproximar uma distribuição rela P).

Como exemplo, observe a fórmula da entropia de uma distribuição normal d-dimensional

$$\begin{aligned}
H(N(\mu,\Sigma)) &= \int -\log(\phi(x|\mu,\Sigma))\phi(x|\mu,\Sigma)\, dx \\
&= \frac{1}{2}\log\left((2\pi)^d \det(\Sigma)\right) + \frac{1}{2}\int (x-\mu)^T\Sigma^{-1}(x-\mu)\phi(x|\mu,\Sigma)\, dx \\
&= \frac{1}{2}\log\left((2\pi)^d \det(\Sigma)\right) + \frac{1}{2}\operatorname{tr}\left(\Sigma^{-1}\int (x-\mu)(x-\mu)^T\phi(x|\mu,\Sigma)\, dx\right) \\
&= \frac{1}{2}\log\left((2\pi)^d \det(\Sigma)\right) + \frac{1}{2}\operatorname{tr}(\Sigma^{-1}\Sigma) = \frac{1}{2}\log\left((2\pi)^d \det(\Sigma)\right) + \frac{d}{2} \\
&= \frac{1}{2}\log\left((2\pi e)^d \det(\Sigma)\right).
\end{aligned}$$

Por outro lado, observe que na seção A.4 demostramos que para uma distribuição P absolutamente contínua e com primeiro e segundo momento finitos, digamos $\mu = \mathbb{E}_P[X]$ e $\Sigma = var_P(X)$, a entropia relativa de P em relação a uma distribuição normal é minimizada igualando o valor esperado e matriz de covariância. Assim, seguindo do apresentado na seção A.4,

$$D_{KL}(P||N(\mu,\Sigma)) = \frac{1}{2}\left(\log\left((2\pi)^d\right) + \log(\det(\Sigma)) + \operatorname{tr}(\Sigma^{-1}var_P(X))\right) - H(P)$$

$$= \frac{1}{2}\log\left((2\pi e)^d \det(\Sigma)\right) - H(P) = H(N(\mu,\Sigma)) - H(P),$$

e como a divergência é não-negativa, temos

$$D_{KL}(P||N(\mu,\Sigma)) = H(N(\mu,\Sigma)) - H(P) \geq 0$$

$$\Rightarrow H(N(\mu,\Sigma)) \geq H(P),$$

isto é, a distribuição normal tem a maior entropia entre as distribuições absolutamente contínuas com primeiro e segundo momento finitos. Entendendo a entropia como uma medida de o quão não informativa uma distribuição é, pode-se usar esse fato para justificar a escolha de um prior gaussiano em modelos Bayesiano com essas restrições para o prior, já que assim introduzimos a menor quantidade de informação a mais possível no sistema antes de introduzir os dados.

A.6 Mais Informações dos Datasets Usadas

A.6.1 Problema Teste

Os inputs utilizados foram

$$X = [2.5,\ 2.75\,,3,\ 3.25\,,3.5,\ 3.75,\ 4,\ 4.25,\ 4.5,\ 4.75,\ 5]$$

e os outputs foram, com 16 casas decimais,

$$\begin{aligned} Y \ = \ & [0.7644575612952016,\ 0.8446143587765195,\ 0.986221976378661,\ 0.9700513454438474,\\ & 0.8442823528773645,\ 0.4114259081377641,\ -0.4089374446370663,\ -0.9544157861580462,\\ & -0.5216629218071086,\ 0.8797014079024436,\ -0.16003857209092667] \end{aligned}$$

A.6.2 *Iris Dataset*

Os dados utilizados foram retirado do repositório *UCI Machine Learning* (DUA; GRAFF, 2017), onde o dataset pode ser encontrado pelo nome *Iris*. Utilizamos o vetor de índices *shuffle* para reordenar os dados (os dados estão indexados de 0 a 149).

$$\begin{aligned} \textit{shuffle} = \ & [19, 131, 37, 22, 18, 89, 46, 41, 123, 111, 97, 144, 92, 31, 109, 7, 49, 102, 61, 121, 16,\\ & 27, 147, 141, 48, 35, 88, 50, 72, 52, 91, 43, 59, 26, 149, 23, 9, 136, 33, 57, 4, 82, 99, 24,\\ & 36, 64, 125, 146, 12, 62, 38, 128, 63, 53, 75, 86, 107, 60, 119, 44, 51, 126, 90, 110, 100,\\ & 93, 45, 115, 113, 143, 87, 10, 21, 98, 1, 56, 13, 106, 84, 124, 120, 77, 108, 116, 127, 130,\\ & 28, 69, 11, 138, 70, 3, 79, 32, 71, 96, 139, 39, 145, 104, 85, 65, 135, 40, 134, 2, 132, 15,\\ & 140, 78, 129, 122, 25, 74, 5, 148, 14, 47, 66, 83, 114, 81, 118, 117, 29, 137, 80, 55, 103,\\ & 20, 54, 34, 42, 142, 112, 17, 94, 8, 133, 73, 76, 6, 58, 68, 101, 105, 67, 30, 95, 0] \end{aligned}$$

Os primeiros 120 pontos compõem o conjunto de treinamento, enquanto os últimos 30 formam o conjunto de teste.