文章编号：1000-8608(2016)03-0279-06

面向大规模类不平衡数据的变分高斯过程分类算法

马 彪[1]，周 瑜[2]，贺建军*[1,2]

（1.大连民族大学 信息与通信工程学院，辽宁 大连 116600；
2.大连理工大学 电子信息与电气工程学部，辽宁 大连 116024）

摘要： 变分高斯过程分类器是最近提出的一种较有效的面向大规模数据的快速核分类算法，其在处理类不平衡问题时，对少数类样本的预测精度通常会较低.针对此问题，通过在似然函数中引入指数权重系数和构造包含相同数目正负类样本的诱导子集解决原始算法的分类面向少数类偏移的问题，建立了一种可以有效处理大规模类不平衡问题的改进变分高斯过程分类算法.在10个大规模UCI数据集上的实验结果表明，改进算法在类不平衡问题上的精度较原始算法得到大幅提高.

关键词： 类不平衡问题；高斯过程；变分推理；大规模数据分类

中图分类号： TP391 **文献标识码：** A **doi：** 10.7511/dllgxb201603009

0 引 言

高斯过程模型具有完全的贝叶斯公式化表示、可自适应地获得模型参数、易实现等优点，是继支持向量机之后又一种受到人们广泛关注的核机器学习方法，被广泛地用于回归[1]、分类[2]、聚类[3]、关系学习[4]、强化学习[5]、多示例多标记学习[6]等各种机器学习问题中.在处理分类问题时，定义的似然函数通常是一个S形函数，因此并不能直接得到潜变量函数的后验概率的分析表达式.为此，人们尝试利用各种方法来得到后验概率的近似表达式，如Laplace(LA)方法[2]、expectation propagation(EP)方法[7]、Kullback-Leibler(KL)散度最小化方法[8]、variational bounding(VB)方法[9]、mean field方法[10]等，虽然通过不断改进，目前已经能取得较高的预测精度，但是由于相关算法的计算复杂度通常是$O(n^3)$(其中n为样本数目)，只能处理数据规模比较小的问题.为了降低算法的计算复杂度，人们提出了构建稀疏模型的问题解决策略，代表性的算法有IVM[11]、FIFC[12]和KLSP[13]，它们的计算复杂度为$O(nm^2)$，其中m为选择的诱导变量的个数，通常m的取值达到数百时算法就可以取得较好的效果.

IVM、FIFC和KLSP等这些面向大规模数据的高斯过程分类算法，在模型构建时很少考虑数据的类不平衡性对算法性能的影响，因此当处理正负样本比例比较悬殊的问题时，对少数类样本的预测精度通常都很低[14].针对以上问题，本文尝试建立一种面向大规模类不平衡问题的高斯过程分类算法.KLSP是一种基于变分原理建立的算法，与其他两种算法相比，它不仅具有较高的精度，而且还具有避免过拟合、可以用分布式或者随机优化的方法对模型进行求解等优点.通过分析KLSP算法发现，造成对少数类样本预测精度比较低的主要原因在似然函数定义和诱导集合选择这两个环节，因此本文将通过改进KLSP算法的这两个环节来建立可以有效处理大规模类不平衡问题的高斯过程分类算法.

1 类不平衡变分高斯过程模型的基本思想与主要模块

设$\boldsymbol{X}$为样本的特征空间，$\boldsymbol{S}=\{(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,$

收稿日期： 2016-01-05； **修回日期：** 2016-03-06.
基金项目： 国家自然科学基金资助项目(61503058,61374170)；辽宁省自然科学基金资助项目(2015020084,2015020099)；辽宁省教育厅科学技术研究项目(L2014540,L2015127)；中央高校基本科研业务费专项资金资助项目(DC201501055,DC201501060201).
作者简介： 马 彪(1962-)，男，工程师，E-mail：mab996@sina.com；贺建军*(1983-)，男，博士，讲师，E-mail：jianjunhe@live.com.

$y_2),\cdots,(\boldsymbol{x}_n,y_n)\}$为包含 n 个样本的训练集,其中 $\boldsymbol{x}_i\in\boldsymbol{X}$ 和 $y_i\in\{+1,-1\}$分别是第 i 个样本的特征向量和类别标记. 为便于描述,下面将用 $\boldsymbol{D}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n\}$和 $\boldsymbol{Y}=(y_1\quad y_2\quad\cdots\quad y_n)^{\mathrm{T}}$ 分别表示 $\boldsymbol{S}$ 中所有训练样本的特征向量构成的集合和类别标记构成的向量. 构建分类算法实质上就是利用训练集在特征空间中建立一个能正确输出待预测样本 $\boldsymbol{x}_*$ 的真实标记的函数 $f(\boldsymbol{x})$. 传统高斯过程二分类模型的基本思想是在服从某个高斯过程分布的潜变量函数簇中寻找最有可能输出待预测样本 $\boldsymbol{x}_*$ 的真实标记的函数,要实现该目的通常包括定义潜变量函数的先验概率、定义似然函数、计算潜变量函数的后验概率和计算预测概率 4 个核心模块. 变分高斯过程模型与传统高斯过程模型一样,也包括这 4 个模块,二者主要的不同在后验概率的计算上. 传统高斯过程模型是利用潜变量函数的先验概率和似然函数推导其后验概率的;而变分高斯过程模型是利用一组中间诱导变量来计算潜变量函数的后验概率分布的,这样可以有效降低算法的计算复杂度.

1.1 定义先验概率

假设潜变量函数 $f(\boldsymbol{x})$服从如下零均值高斯过程分布:

$$f(\boldsymbol{x})\sim GP(\boldsymbol{0},k(\boldsymbol{x},\boldsymbol{x}')) \tag{1}$$

其中 $k(\boldsymbol{x},\boldsymbol{x}')$表示协方差函数,本文将采用如下的协方差函数:

$$k(\boldsymbol{x},\boldsymbol{x}')=\alpha\mathrm{e}^{-\|\boldsymbol{x}-\boldsymbol{x}'\|^2/\beta} \tag{2}$$

假设 $\boldsymbol{U}\subset\boldsymbol{D}$ 是选取的用于生成诱导变量的训练样本子集,$\boldsymbol{x}_*\in\boldsymbol{X}$ 为待预测样本,则可以根据式(1)得到 $f(\boldsymbol{x})$在 $\boldsymbol{D}$、$\boldsymbol{U}$ 和 $\boldsymbol{x}_*$ 上的函数值 $\boldsymbol{F}_{\boldsymbol{D}}$、$\boldsymbol{F}_{\boldsymbol{U}}$ 和 f_* 的如下先验概率分布和条件先验概率分布:

$$\begin{aligned}
&p(\boldsymbol{F}_{\boldsymbol{D}}\mid\boldsymbol{D})=N(\boldsymbol{F}_{\boldsymbol{D}}\mid\boldsymbol{0},\boldsymbol{K}_{\boldsymbol{D}})\\
&p(\boldsymbol{F}_{\boldsymbol{U}}\mid\boldsymbol{D})=N(\boldsymbol{F}_{\boldsymbol{U}}\mid\boldsymbol{0},\boldsymbol{K}_{\boldsymbol{U}})\\
&p(\boldsymbol{F}_{\boldsymbol{D}}\mid\boldsymbol{F}_{\boldsymbol{U}},\boldsymbol{D})=N(\boldsymbol{F}_{\boldsymbol{D}}\mid\boldsymbol{K}_{\boldsymbol{DU}}\boldsymbol{K}_{\boldsymbol{U}}^{-1}\boldsymbol{F}_{\boldsymbol{U}},\\
&\qquad\qquad\boldsymbol{K}_{\boldsymbol{D}}-\boldsymbol{K}_{\boldsymbol{DU}}\boldsymbol{K}_{\boldsymbol{U}}^{-1}\boldsymbol{K}_{\boldsymbol{DU}}^{\mathrm{T}})\\
&p(f_*\mid\boldsymbol{F}_{\boldsymbol{U}},\boldsymbol{D},\boldsymbol{x}_*)=N(f_*\mid\boldsymbol{K}_{*\boldsymbol{U}}\boldsymbol{K}_{\boldsymbol{U}}^{-1}\boldsymbol{F}_{\boldsymbol{U}},\\
&\qquad\qquad K_{**}-\boldsymbol{K}_{*\boldsymbol{U}}\boldsymbol{K}_{\boldsymbol{U}}^{-1}\boldsymbol{K}_{*\boldsymbol{U}}^{\mathrm{T}})
\end{aligned} \tag{3}$$

式中:$\boldsymbol{F}_{\boldsymbol{D}}$ 是 $f(\boldsymbol{x})$在样本集 $\boldsymbol{D}$ 上的函数值构成的列向量,即 $\boldsymbol{F}_{\boldsymbol{D}}=(f_1\quad f_2\quad\cdots\quad f_n)^{\mathrm{T}}$,$f_i=f(\boldsymbol{x}_i)$;$\boldsymbol{K}_{\boldsymbol{D}}$ 是协方差函数 $k(\boldsymbol{x},\boldsymbol{x}')$在 $\boldsymbol{D}$ 上的值构成的矩阵,第 i 行第 j 列上的元素是 $k(\boldsymbol{x}_i,\boldsymbol{x}_j)$;$\boldsymbol{F}_{\boldsymbol{U}}$ 是 $f(\boldsymbol{x})$在诱导集 $\boldsymbol{U}$ 上的函数值构成的列向量;$\boldsymbol{K}_{\boldsymbol{U}}$ 是协方差函数 $k(\boldsymbol{x},\boldsymbol{x}')$在 $\boldsymbol{U}$ 上的值构成的矩阵;$\boldsymbol{K}_{\boldsymbol{DU}}$表示 $\boldsymbol{D}$ 和 $\boldsymbol{U}$ 中样本之间的协方差函数值构成的矩阵;f_* 表示 $f(\boldsymbol{x})$在待预测样本 $\boldsymbol{x}_*$ 上的值,即 $f_*=f(\boldsymbol{x}_*)$;$\boldsymbol{K}_{*\boldsymbol{U}}$表示 $\boldsymbol{x}_*$ 和 $\boldsymbol{U}$ 中样本之间的协方差函数值构成的行向量;$K_{**}=k(\boldsymbol{x}_*,\boldsymbol{x}_*)$.

1.2 定义似然函数

与传统高斯过程分类模型一样,$f(\boldsymbol{x})$在 $\boldsymbol{x}_i$ 上的值 f_i 的似然函数 $p(y_i\mid f_i)$可以定义为

$$p(y_i\mid f_i)=\frac{1}{1+\mathrm{e}^{-y_if_i}} \tag{4}$$

对于各个类别样本数据比较均衡的问题,联合似然函数 $p(\boldsymbol{Y}\mid\boldsymbol{F}_{\boldsymbol{D}})$可以定义为所有 $p(y_i\mid f_i)(i=1,2,\cdots,n)$的乘积. 但是对于类不平衡问题,这样的定义意味着错分一个少数类样本的代价与错分一个多数类样本的代价一样,从而使得算法的分类面偏向少数类一侧. 本文将在联合似然函数中的正负类样本对应的似然函数上引入不同的权重系数,使得错分少数类样本的代价大于错分多数类样本的代价,最终实现改善少数类样本预测精度的目的. 联合似然函数的具体定义如下:

$$p(\boldsymbol{Y}\mid\boldsymbol{F}_{\boldsymbol{D}})=\prod_{i=1}^{n}(p(y_i\mid f_i))^{r_i} \tag{5}$$

其中权重系数 r_i 的取值为

$$r_i=\begin{cases}\dfrac{n}{2n_+}; & y_i=+1\\[2ex] \dfrac{n}{2n_-}; & y_i=-1\end{cases}$$

n_+ 和 n_- 分别表示训练集中正负样本的个数. 从式(5)可以看出,少数类样本对应的权重系数要大于 1,而多数类样本对应的权重系数要小于 1,两类样本数目越悬殊,权重的差距也越大,由于所有正(负)类样本的权重系数之和是 $n/2$,这在一定程度上保证了联合似然函数中正负类样本在整体上具有一样的话语权,因此可以避免分类面的偏移.

1.3 计算后验概率

由于利用贝叶斯公式 $p(\boldsymbol{F}_{\boldsymbol{D}}\mid\boldsymbol{D},\boldsymbol{Y})=\dfrac{p(\boldsymbol{Y}\mid\boldsymbol{F}_{\boldsymbol{D}})p(\boldsymbol{F}_{\boldsymbol{D}}\mid\boldsymbol{D})}{\int p(\boldsymbol{Y}\mid\boldsymbol{F}_{\boldsymbol{D}})p(\boldsymbol{F}_{\boldsymbol{D}}\mid\boldsymbol{D})\mathrm{d}\boldsymbol{F}_{\boldsymbol{D}}}$ 计算 $p(\boldsymbol{F}_{\boldsymbol{D}}\mid\boldsymbol{D},\boldsymbol{Y})$需要较高的计算复杂度,变分高斯过程模型将利用诱导变量 $\boldsymbol{F}_{\boldsymbol{U}}$ 的后验概率 $p(\boldsymbol{F}_{\boldsymbol{U}}\mid\boldsymbol{D},\boldsymbol{Y})$来得到

$p(\boldsymbol{F}_D|\boldsymbol{D},\boldsymbol{Y})$，即

$$p(\boldsymbol{F}_D|\boldsymbol{D},\boldsymbol{Y})=\int p(\boldsymbol{F}_D|\boldsymbol{F}_U,\boldsymbol{D})p(\boldsymbol{F}_U|\boldsymbol{D},\boldsymbol{Y})\mathrm{d}\boldsymbol{F}_U\approx\int p(\boldsymbol{F}_D|\boldsymbol{F}_U,\boldsymbol{D})q(\boldsymbol{F}_U)\mathrm{d}\boldsymbol{F}_U \tag{6}$$

其中 $q(\boldsymbol{F}_U)=N(\boldsymbol{F}_U|\boldsymbol{\mu},\boldsymbol{\Sigma})$ 是 $p(\boldsymbol{F}_U|\boldsymbol{D},\boldsymbol{Y})$ 的一个高斯逼近，参数 $\boldsymbol{\mu}$、$\boldsymbol{\Sigma}$ 可以通过最大化边缘似然函数 $p(\boldsymbol{Y}|\boldsymbol{D})$ 的如下下界得到：

$$\log\ p(\boldsymbol{Y}|\boldsymbol{D})\geqslant\int\log\ p(\boldsymbol{Y}|\boldsymbol{F}_D)q(\boldsymbol{F}_D)\mathrm{d}\boldsymbol{F}_D-KL(q(\boldsymbol{F}_U)\parallel p(\boldsymbol{F}_U|\boldsymbol{D}))\triangleq\Psi(\boldsymbol{\mu},\boldsymbol{\Sigma}) \tag{7}$$

其中 $KL(\cdot\parallel\cdot)$ 表示 KL 散度，$q(\boldsymbol{F}_D)=\int p(\boldsymbol{F}_D|\boldsymbol{F}_U,\boldsymbol{D})q(\boldsymbol{F}_U)\mathrm{d}\boldsymbol{F}_U=N(\boldsymbol{F}_D|\boldsymbol{A}\boldsymbol{\mu},\boldsymbol{B}+\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}^{\mathrm{T}})$，$\boldsymbol{A}=\boldsymbol{K}_{DU}\boldsymbol{K}_U^{-1}$，$\boldsymbol{B}=\boldsymbol{K}_D-\boldsymbol{K}_{DU}\boldsymbol{K}_U^{-1}\boldsymbol{K}_{DU}^{\mathrm{T}}$. 关于该下界的详细论述见文献[13]. 将 $q(\boldsymbol{F}_D)$、$q(\boldsymbol{F}_U)$、$p(\boldsymbol{F}_U|\boldsymbol{D})$、$p(\boldsymbol{Y}|\boldsymbol{F}_D)$ 的表达式代入 $\Psi(\boldsymbol{\mu},\boldsymbol{\Sigma})$，可以得到 $\Psi(\boldsymbol{\mu},\boldsymbol{\Sigma})$ 的表达式为

$$\Psi(\boldsymbol{\mu},\boldsymbol{\Sigma})=\Big(\sum_{i=1}^{n}r_i\int\log\ p(y_i|f_i)q(f_i)\mathrm{d}f_i\Big)-\frac{1}{2}\Big(-n+\log\ |\boldsymbol{K}_U|-\log\ |\boldsymbol{\Sigma}|+\boldsymbol{\mu}^{\mathrm{T}}\boldsymbol{K}_U^{-1}\boldsymbol{\mu}+\mathrm{tr}(\boldsymbol{K}_U^{-1}\boldsymbol{\Sigma})\Big) \tag{8}$$

其中 $\mathrm{tr}(\cdot)$ 表示矩阵的迹，$q(f_i)=N(f_i|\boldsymbol{A}_i\boldsymbol{\mu},B_{ii}+\boldsymbol{A}_i\boldsymbol{\Sigma}\boldsymbol{A}_i^{\mathrm{T}})$，$\boldsymbol{A}_i$ 表示矩阵 $\boldsymbol{A}$ 的第 i 行元素，B_{ii} 表示矩阵 $\boldsymbol{B}$ 的第 i 个对角元素. 对 $\Psi(\boldsymbol{\mu},\boldsymbol{\Sigma})$ 分别关于 $\boldsymbol{\mu}$、$\boldsymbol{\Sigma}$ 求导，可得

$$\frac{\partial\Psi(\boldsymbol{\mu},\boldsymbol{\Sigma})}{\partial\boldsymbol{\mu}}=\sum_{i=1}^{n}\Big(\Big(r_i\int\frac{\partial\log\ p(y_i|f_i)}{\partial f_i}\times q(f_i)\mathrm{d}f_i\Big)\boldsymbol{A}_i^{\mathrm{T}}\Big)-\boldsymbol{K}_U^{-1}\boldsymbol{\mu}=\boldsymbol{A}^{\mathrm{T}}\mathbf{diag}(\boldsymbol{r})\boldsymbol{d}-\boldsymbol{K}_U^{-1}\boldsymbol{\mu} \tag{9}$$

$$\frac{\partial\Psi(\boldsymbol{\mu},\boldsymbol{\Sigma})}{\partial\boldsymbol{\Sigma}}=\frac{1}{2}\sum_{i=1}^{n}\Big(\Big(r_i\int\frac{\partial^2\log\ p(y_i|f_i)}{\partial f_i^2}\times q(f_i)\mathrm{d}f_i\Big)\boldsymbol{A}_i^{\mathrm{T}}\boldsymbol{A}_i\Big)+\frac{1}{2}\boldsymbol{\Sigma}^{-1}-\frac{1}{2}\boldsymbol{K}_U^{-1}=\frac{1}{2}\boldsymbol{A}^{\mathrm{T}}\mathbf{diag}(\boldsymbol{w})\mathbf{diag}(\boldsymbol{r})\boldsymbol{A}+\frac{1}{2}\boldsymbol{\Sigma}^{-1}-\frac{1}{2}\boldsymbol{K}_U^{-1} \tag{10}$$

其中 $\mathbf{diag}(\boldsymbol{r})$ 表示以向量 $\boldsymbol{r}$ 为对角元的对角矩阵，$\boldsymbol{r}=(r_1\quad r_2\quad\cdots\quad r_n)^{\mathrm{T}}$，$\boldsymbol{d}=(d_1\quad d_2\quad\cdots\quad d_n)^{\mathrm{T}}$，$d_i=\int\frac{\partial\log\ p(y_i|f_i)}{\partial f_i}q(f_i)\mathrm{d}f_i$，$\boldsymbol{w}=(w_1\quad w_2\quad\cdots\quad w_n)^{\mathrm{T}}$，$w_i=\int\frac{\partial^2\log\ p(y_i|f_i)}{\partial f_i^2}q(f_i)\mathrm{d}f_i$，$d_i$ 和 w_i 可以通过 Gauss-Hermite 求积公式计算得到. 在得到梯度公式(9)和(10)后，可以采用各种基于梯度的优化方法求解 $\boldsymbol{\mu}$、$\boldsymbol{\Sigma}$，本文采用有限储存拟牛顿方法(L-BFGS)[15]来对其求解.

1.4　计算预测概率

在得到后验概率 $p(\boldsymbol{F}_U|\boldsymbol{D},\boldsymbol{Y})$ 的近似表达式以后，可以得到 f_* 的后验概率分布：

$$p(f_*|\boldsymbol{D},\boldsymbol{Y},\boldsymbol{x}_*)=\int p(f_*|\boldsymbol{F}_U,\boldsymbol{D},\boldsymbol{x}_*)\times p(\boldsymbol{F}_U|\boldsymbol{D},\boldsymbol{Y})\mathrm{d}\boldsymbol{F}_U\approx\int p(f_*|\boldsymbol{F}_U,\boldsymbol{D},\boldsymbol{x}_*)q(\boldsymbol{F}_U)\mathrm{d}\boldsymbol{F}_U=N(f_*|\boldsymbol{K}_{*U}\boldsymbol{K}_U^{-1}\boldsymbol{\mu},\boldsymbol{K}_*+\boldsymbol{K}_{*U}\boldsymbol{K}_U^{-1}(\boldsymbol{\Sigma}-\boldsymbol{K}_U)(\boldsymbol{K}_{*U}\boldsymbol{K}_U^{-1})^{\mathrm{T}}) \tag{11}$$

然后可以计算样本 $\boldsymbol{x}_*$ 是正样本的概率：

$$p(y_*=+1|\boldsymbol{D},\boldsymbol{Y},\boldsymbol{x}_*)=\int\frac{1}{1+\mathrm{e}^{-f_*}}\times p(f_*|\boldsymbol{D},\boldsymbol{Y},\boldsymbol{x}_*)\mathrm{d}f_* \tag{12}$$

1.5　诱导子集 $\boldsymbol{U}$ 的选取

诱导子集 $\boldsymbol{U}$ 的选取对算法的性能会有一定的影响，但是目前还没有统一的策略，主要的策略是选取训练样本的聚类中心作为诱导子集[16]. 为了避免 $\boldsymbol{U}$ 中各个类别样本的不均衡导致算法分类面的偏移，本文将利用 K 均值聚类算法分别从正负类中选取相同数目的样本来构成 $\boldsymbol{U}$. 基本流程如下：假设拟选取 m 个样本构成诱导子集 $\boldsymbol{U}$，则先利用 K 均值聚类算法分别对正负类样本进行聚类，各生成 $m/2$ 个聚类中心，然后把这些聚类中心合并构成诱导子集 $\boldsymbol{U}$. 为了降低计算复杂度，本文采用 Chen[17] 实现的一种快速 K 均值聚类算法来完成 $\boldsymbol{U}$ 的选取，其中算法的迭代次数设定为 150. 该算法有时会存在生成的聚类中心数目小于设定的数目的情况，对于这种情况，将随机从样本较多的聚类中选取相应数目的样本来补齐.

2　仿真实验

本章将在 10 个大规模类不平衡二分类数据集上验证本文所建算法的性能. 这 10 个数据集全

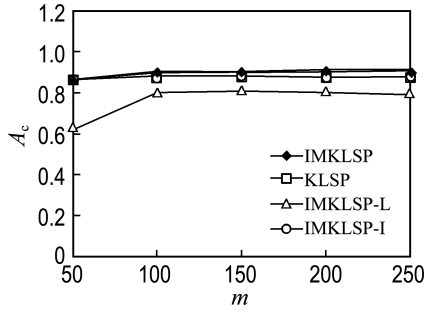
部来自于 UCI 数据库[18]，其中，mnist0 和 mnist9 数据集是在原始的 mnist 多类数据集基础上分别以手写数字 0 和 9 所在类别为正类、其他类别为负类形成的二分类数据集；covtype3、covtype4 和 covtype7 数据集是分别以原始的 covtype 多类数据集中的第 3、4 和 7 类别为正类、其他类别为负类形成的二分类数据集；poker3 和 poker4 数据集是分别以原始的 poker 多类数据集的第 3、4 类别为正类、其他类别为负类形成的二分类数据集；关于这些数据集的详细信息见表 1. 本文的主要创新在于对文献[13]提出的变分高斯过程算法(KLSP 算法)进行了改进，使其可以有效处理类不平衡问题，因此下面将主要对本文所建算法(简称 IMKLSP)和 KLSP 算法进行对比、分析. 两个算法都采用式(2)所示的协方差函数，并且利用有限储存拟牛顿方法[15]进行求解，其中，协方差函数的参数 α 和 β 的取值是分别在集合 $\{10^{-6}, 10^{-5}, \cdots, 10^{5}\}$ 和 $\{3^{-4}d_u, 3^{-3}d_u, \cdots, 3^{6}d_u\}$ (d_u 为诱导子集与训练集的样本之间的平均距离)中通过交叉验证法选取确定的，有限储存拟牛顿方法的存储长度 M 的取值为 10，算法迭代次数 L 的取值为 20. 以下所有实验结果都是通过随机选取 1/2 的样本作为训练数据，1/2 的样本作为测试数据，然后重复进行 5 次实验得到的平均结果.

表 1 验证算法性能所用数据集

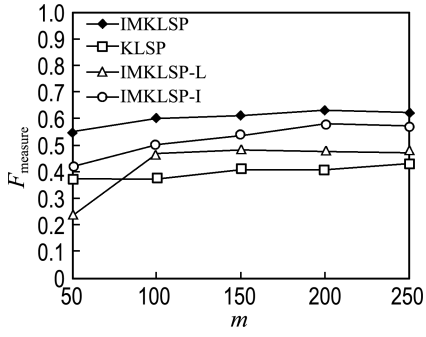
Tab. 1 The datasets for validating the performance of the algorithms

数据集	特征个数	正样本个数	负样本个数	正负样本比
mnist0	780	5 923	54 077	1∶9.1
mnist9	780	5 949	54 051	1∶9.1
w8a	300	1 933	62 767	1∶32.5
ijcnn1	22	13 565	128 126	1∶9.4
skin	3	50 859	194 195	1∶3.8
covtype3	54	35 754	545 258	1∶15.3
covtype4	54	2 747	578 265	1∶210.5
covtype7	54	20 510	560 502	1∶27.3
poker3	10	48 828	976 182	1∶20.0
poker4	10	21 634	1 003 376	1∶46.4

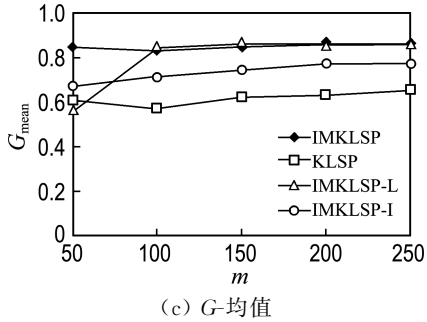
本文主要是通过改进 KLSP 算法的似然函数定义和诱导子集选择这两个环节来实现最终的 IMKLSP 算法的. 为了验证改进这两个环节的必要性，首先在 ijcnn1 数据集上对 KLSP、IMKLSP、IMKLSP-L 和 IMKLSP-I 4 个算法进行了比较，其中 IMKLSP-L 表示通过只改变似然函数的定义环节而不改变诱导子集选择环节所建立的算法，IMKLSP-I 表示通过只改变诱导子集选择环节而不改变似然函数的定义环节所建立的算法. 图 1 给出了诱导子集的样本数目 m 取不同值时，上述 4 个算法在精度(A_c)、F-测量值($F_{measure}$)和 G-均值(G_{mean})3 个不同评价指标上的实验结果. 从图 1 可以看出，虽然 KLSP 算法在精度这个评价指标上取得了与 IMKLSP 算法相当的结果，但是在另外两个指标 F-测量值和 G-均值上的结果



(a) 精度



(b) F-测量值



(c) G-均值

图 1 不同诱导子集样本数目下各个算法在 ijcnn1 数据集上的实验结果

Fig. 1 The experimental results of each algorithm on ijcnn1 dataset when varying the sample number of inducing set

却远远小于 IMKLSP 算法，这正好说明了 KLSP 算法不能很好地对类不平衡问题中的少数类样本做出预测. 进一步将 IMKLSP-L 和 IMKLSP-I 算法与 IMKLSP 算法进行对比可以看出，这两个算法的性能都要比 IMKLSP 算法差，这表明改进似然函数定义和诱导子集选择这两个环节都是必要的. 特别的，似然函数定义环节对算法的影响要比诱导子集选择环节的影响大. 此外，从图 1 可以看出，只要诱导子集的样本个数达到 100，各个算法就已经能取得较好的预测结果.

表 2 列出了诱导子集的样本个数 m 取 200 时，KLSP 和 IMKLSP 算法在 10 个 UCI 数据集上的预测结果，可以看出，除在 skin 和 mnist0 数据集上两个算法取得了相当的实验结果外，在其他的 8 个数据集上，无论是在 F-测量值还是 G-均值指标上，IMKLSP 算法的性能都远远优于 KLSP 算法. 而在 skin 和 mnist0 数据集上之所以两个算法的性能相当是因为这两个样本集中的正负样本比例不太悬殊，而且 KLSP 算法在这两个数据集上的预测结果本身就比较高，改进的空间比较小. 总之，以上实验结果表明在处理大规模类不平衡问题方面，IMKLSP 算法要明显优于 KLSP 算法.

表 2　IMKLSP 与 KLSP 算法在 10 个大规模类不平衡 UCI 数据集上的性能对比

Tab. 2　The performance comparison of IMKLSP and KLSP algorithms on ten large-scale class-imbalanced UCI datasets

数据集	F_{measure}		G_{mean}	
	KLSP	IMKLSP	KLSP	IMKLSP
mnist0	0.973	0.964	0.981	0.991
mnist9	0.827	0.846	0.906	0.963
w8a	0.071	0.278	0.478	0.826
ijcnn1	0.404	0.630	0.633	0.865
skin	0.971	0.971	0.992	0.992
covtype3	0.553	0.557	0.807	0.907
covtype4	0.016	0.382	0.538	0.866
covtype7	0.261	0.317	0.515	0.902
poker3	0.087	0.151	0.402	0.637
poker4	0.042	0.094	0.415	0.719

3　结　语

本文基于最近提出的变分高斯过程模型建立了一种可以有效处理大规模类不平衡问题的核分类算法，在 10 个 UCI 数据集上的实验结果表明，该算法可以取得较高的预测精度. 本文算法也还有需要改进的地方，其中一个问题就是建立适合该算法的快速聚类算法，在实验过程中发现同一个实验重复进行两次有时会取得不一样的结果，甚至能相差两个百分点，通过分析发现，这主要是由诱导子集选取环节中采用的 K 均值聚类算法造成的，该聚类算法虽然可以进行快速聚类，但是算法的初始聚类中心每次都是随机给定的，这会导致得到的聚类中心有时会有一定的误差；此外，考虑到目前基于交叉验证法选取协方差函数参数的策略不仅效率低而且不能得到最优参数，如何利用模型的边缘似然函数自动选择协方差函数参数也是下一步需要考虑的问题.

参考文献：

[1] Ranganathan A, Yang Ming-hsuan, Ho J. Online sparse Gaussian process regression and its applications [J]. **IEEE Transactions on Image Processing**, 2011, **20**(2):391-404.

[2] Williams C, Barber D. Bayesian classification with Gaussian processes [J]. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, 1998, **20**(12):1342-1351.

[3] Kim Hyun-chul, Lee Jaewook. Clustering based on Gaussian processes [J]. **Neural Computation**, 2007, **19**(11):3088-3107.

[4] Chu W, Sindhwani V, Ghahramani Z, *et al*. Relational learning with Gaussian processes [J]. **Advances in Neural Information Processing Systems**, 2007:289-296.

[5] Engel Y, Mannor S, Meir R. Reinforcement learning with Gaussian processes [C] // **Proceedings of the 22nd International Conference on Machine Learning**. New York: ACM Press, 2005:201-208.

[6] HE Jian-jun, GU Hong, WANG Zhe-long. Bayesian multi-instance multi-label learning using Gaussian process prior [J]. **Machine Learning**, 2012, **88**(1-2):273-295.

[7] Kim Hyun-chul, Ghahramani Z. Bayesian Gaussian process classification with the EM-EP algorithm [J]. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, 2006, **28**(12):

1948-1959.

[8] Opper M, Archambeau C. The variational Gaussian approximation revisited [J]. **Neural Computation**, 2008, **21**(3):786-792.

[9] Gibbs M N, MacKay D J C. Variational Gaussian process classifiers [J]. **IEEE Transactions on Neural Networks**, 2000, **11**(6):1458-1464.

[10] Opper M, Winther O. Mean field methods for classification with Gaussian processes [J]. **Advances in Neural Information Processing Systems**, 1999:309-315.

[11] Lawrence N, Seeger M, Herbrich R. Fast sparse Gaussian process methods: The informative vector machine [J]. **Advances in Neural Information Processing Systems**, 2003:609-616.

[12] Naish-Guzman A, Holden S. The generalized FITC approximation [C] // **Advances in Neural Information Processing Systems 20-Proceedings of the 2007 Conference**. New York: Curran Associates Inc., 2009.

[13] Hensman J, Matthews A G, Ghahramani Z. Scalable variational Gaussian process classification [J]. **Journal of Machine Learning Research**, 2015, **38**:351-360.

[14] Ganganwar V. An overview of classification algorithms for imbalanced datasets [J]. **International Journal of Emerging Technology and Advanced Engineering**, 2012, **2**(4):42-47.

[15] Nocedal J, Wright S J. **Numerical Optimization** [M]. 2nd ed. New York: Springer, 2006:195-204.

[16] Hensman J, Fusi N, Lawrence N D. Gaussian processes for big data [C] // **Uncertainty in Artificial Intelligence-Proceedings of the 29th Conference, UAI 2013**. Arlington: AUAI Press, 2013:282-290.

[17] Chen M. Kmeans clustering [CP/OL]. [2013-01-05]. http://www.mathworks.com/matlabcentral/fileexchange/24616-kmeans-clustering.

[18] Bache K, Lichman M. UCI machine learning repository [DB/OL]. [2013-01-05]. http://archive.ics.uci.edu/ml.

Variational Gaussian process classification algorithm for large-scale class-imbalanced data

MA Biao[1], ZHOU Yu[2], HE Jian-jun[*1,2]

(1. College of Information and Communication Engineering, Dalian Nationalities University, Dalian 116600, China;
2. Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China)

Abstract: Variational Gaussian process classifier is an effective fast kernel algorithm proposed recently for large-scale data classification. However, for the class-imbalanced problem, it usually achieves lower accuracy on the samples of minority class. By assigning different index weight coefficients to the likelihood functions and constructing an inducing set containing equal numbers of positive and negative samples to avoid hyperplane biased toward the side of minority class, an improved variational Gaussian process classification algorithm is proposed, which can deal with the large-scale class-imbalanced problem effectively. The experimental results of ten large-scale UCI datasets show that the proposed algorithm can achieve much higher accuracy than the original one for class-imbalanced problem.

Key words: class-imbalanced problem; Gaussian process; variational inference; large-scale data classification