山西大学
Shanxi University

2019 届 博 士 学 位 论 文

监督学习算法预测性能比较的正则化交叉验证方法研究

作者姓名　王瑞波

指导教师　李济洪 教授

学科专业　软件工程

研究方向　机器学习与软件缺陷预测

培养单位　计算机与信息技术学院

学习年限　2014 年 9 月至 2019 年 6 月

二〇一九年六月

山西大学

2019 届博士学位论文

监督学习算法预测性能比较的正则化交叉验证方法研究

作者姓名 王瑞波

指导教师 李济洪 教授

学科专业 软件工程

研究方向 机器学习与软件缺陷预测

培养单位 计算机与信息技术学院

学习年限 2014 年 9 月至 2019 年 6 月

二〇一九年六月

Thesis for Doctor's Degree, Shanxi University, 2019

Research on Block-regularized Cross-Validation Methods for Comparing Supervised Algorithms

Student Name	Ruibo Wang
Supervisor	Prof. Jihong Li
Major	Software Engineering
Specialty	Machine Learning and Software Defect Prediction
Department	School of Computer and Information Technology
Research Duration	2014.9 – 2019.6

June 2019

目 录

Contents

图 目 录

表 目 录

中 文 摘 要

在数据驱动的智能信息系统中，机器学习模型是必用的。模型通常是由一个算法在大量数据上学习得到的。选择一个性能高的算法是系统不断升级的关键技术。事实上，算法比较是机器学习建模中基本问题之一。一个新发明的算法其性能是否优于旧算法，需要经过合理的统计检验才能得出可靠的结论。算法比较任务贯穿于建模过程中的算法选择、特征选择、模型选择及评估等各个阶段，是建模中关键环节。本文仅关注两个有监督学习算法的比较问题。

算法比较任务通常被描述为：给定一个数据集及两个机器学习算法，哪个算法可以产生性能更为优良的模型？算法比较任务可形式化为统计显著性检验问题，并采用经过精心设计的交叉验证以及合理的显著性检验方法来解决。

基于 5 折（10 折）交叉验证的 t 检验，因简单易用，被研究者广泛采用。然而，该方法采用的方差估计偏小，难以有效控制检验的第一类错误，易导致假阳性的结论。尽管其第一类错误在 5×2 交叉验证 t 检验及 F 检验中得到改进，但 5×2 交叉验证受随机数据切分的影响，也常常得到不可靠的结论。为此，面向算法比较任务，本文对给定的一个 IID 数据集，首先从数据的切分方式入手，构建了正则化交叉验证方法，给出了较为合理的方差估计，然后构造了合理的序贯检验方法，理论分析和实验验证其减小了检验的第一类错误，可以得到可靠的结论。进一步，将正则化交叉验证拓展到文本数据集，对预测标签的分布增加正则化条件，并给出了准确率、召回率和 F_1 值的后验分布，构建了算法比较的贝叶斯检验方法。

本文研究了正则化交叉验证的理论性质及构建方法。首先，从泛化误差的 repeated learning-testing(RLT) 估计入手，分析了 RLT 的随机切分对该估计的方差的影响，发现较差的切分方式会造成 RLT 中训练集间样本重叠过多，从而增大 RLT 估计的方差。因此，本文引入正则化条件约束重叠样本个数，优化 RLT 方法的切分方式，减小 RLT 估计的方差，构建正则化 RLT 方法。本文给出了正则化 RLT 方法的几种简易构造方法。作为 RLT 方法的一种特殊情形，$m\times 2$ 交叉验证在算法比较中使用更为广泛。为此，本文进一步考虑 $m\times 2$ 交叉验证的优化问题。本文分析重叠样本个数对泛化误差的 $m\times 2$ 交叉验证估计方差的影响，引入正则化条件将 $m\times 2$ 交叉验证的重叠样本个数约束至 $n/4$ 左右（n 为数据集大小），提出正则化 $m\times 2$ 交叉验证，证明了正则化 $m\times 2$ 交叉验证可有效地减少泛化误差估计

的方差，开发了正则化 $m \times 2$ 交叉验证的高效增量式构造算法。针对文本数据集，本文进一步引入卡方统计量来度量训练集和验证集上多种频次分布的差异，提出关于该差异度量的多种正则化条件，进一步优化正则化 $m \times 2$ 交叉验证，以构建适用于文本数据的正则化 $m \times 2$ 交叉验证方法。本文使用 IID 数据集和文本数据集上的大量实验，说明上述正则化交叉验证方法的优良性。

本文将算法比较任务形式化为假设检验问题，研究了基于正则化 $m \times 2$ 交叉验证的统计推断方法。针对泛化误差，因训练集间存在重叠样本，正则化 $m \times 2$ 交叉验证中多个 hold-out 估计间存在相关性，使基于正则化 $m \times 2$ 交叉验证统计推断不同于 IID 观测上的传统统计推断方法。本文从理论上确定了正则化 $m \times 2$ 交叉验证估计中相关系数的上下界，给出正则化 $m \times 2$ 交叉验证估计的合理方差估计，严格证明所采用的统计量服从 t 分布。通过合理设置相关系数，构造了一个相对保守的序贯 t 检验统计量，并给出序贯置信区间。区别于传统的 IID 序贯检验，当重复次数 m 趋于无穷时，该序贯置信区间的期望长度收敛于一个正值，可能导致序贯 t 检验在有限时刻内无法停止。为此，本文使用序贯置信区间期望长度的缩减率作为准则，选取序贯 t 检验的最大停止时刻。本文从理论分析和模拟实验两方面比较了现有的一些检验与本文提出的序贯 t 检验。实验结果表明该序贯 t 检验为保守统计推断，可有效控制第一类错误且具有更优的势函数，并可给出可靠的结论。实验结果也说明，在许多情形下，不宜采用固定的 m，而采用序贯的做法是必要的。针对文本数据，算法性能指标多为准确率、召回率和 F_1 值。它们的分布是偏峰的。因此，采用 t 检验不妥。针对准确率，召回率和 F_1 值，本文分析了正则化 $m \times 2$ 交叉验证估计中的相关性与准确率、召回率和 F_1 值的后验分布间的关系，给出了它们的精确后验分布，构造了合理的后验置信区间，进而提出了算法比较的贝叶斯检验方法，并在文本数据上的分词及命名实体识别实验证实了该贝叶斯检验的有效性。

本文以软件缺陷预测任务为例，针对缺陷数预测模型，将正则化 $m \times 2$ 交叉验证序贯 t 检验用于检验各聚合特征对模型性能是否有显著影响的问题中。针对缺陷倾向性预测模型，文本将基于正则化 $m \times 2$ 交叉验证的贝叶斯检验，用于比较 logistic 回归和随机森林两种分类算法在模型的准确率、召回率和 F_1 值上谁更优良。

本文提出的正则化交叉验证及其统计推断方法，提高了算法比较结论的可靠性，对有监督学习算法的建模具有重要意义。关于优化数据切分的正则化思想，可扩展到大规模数据的子抽样上，为分布式学习和建模提供新的思路和方法。

关键词: 正则化; 交叉验证; 数据切分; 文本数据; 算法比较; 统计推断; 软件缺陷预测

ABSTRACT

Machine learning model is an essential component in data-driven intelligent systems and is usually derived by employing a learning algorithm on a massive data set. Increasingly upgrading an intelligent system requires a reasonable algorithm comparison method to select the algorithm with the highest performance. In fact, algorithm comparison is a fundamental task to answer whether a novel algorithm is significantly better than a baseline algorithm, and it uses a reasonable statistical significance test to draw a reliable conclusion. An algorithm comparison method is usually used in major phases in modeling process, including algorithm selection, feature selection, model selection and evaluation. Therefore, algorithm comparison is very important to modeling process. This study concentrates on comparing two supervised algorithms.

Algorithm comparison is commonly defined as: Given two learning algorithms and a data set, which algorithm will produce more accurate models when trained and validated on data sets of the same size as the given data set? Algorithm comparison is usually formulated into a hypothesis testing problem, which is addressed with a reasonable significance test method on the basis of a well-designed cross-validation (CV).

The t-tests based on five-fold (ten-fold) CV is usually preferred due to its simplicity. However, because the tests adopt down-biased variance estimators, they have a drawback of uncontrollable type I errors, and thus they tend to make false positive conclusions. Although the drawback is improved in 5×2 CV t-test and F-test, these tests usually make unreliable conclusions due to the random partitioning of 5×2 CV. Therefore, for algorithm comparison on an IID data set, in this study, the data partitioning is optimized and block-regularized CV (BCV) methods are proposed. A reasonable variance estimator of a BCV estimator is adopted and a novel sequential test method is constructed. From theoretical and

experimental perspectives, it is illustrated that the sequential test has a reasonable type I error and can make reliable conclusions. Furthermore, BCV are generalized on text data sets by using the regularization conditions on discrete distribution of prediction labels. Moreover, based on BCV, the posterior distributions of commonly used evaluation metrics, including precision, recall and F_1 measure, are elaborated, and a Bayes test method on the basis of the posterior distributions is proposed for algorithm comparison.

In this study, BCV methods are developed and their theoretical properties and construction methods are investigated. Specifically, for repeated learning-testing (RLT) estimator of the generalization error, poor data partitioning may cause a large overlapping samples between any two training sets of RLT, and enlarge the variance in the RLT estimator. Therefore, through introducing a regularization condition on the number of the overlapping samples, the data partitioning of RLT is optimized to reduce the variance in the RLT estimator, and a block-regularized RLT (BRLT) is developed. Several efficient construction methods of BRLT are also proposed. Moreover, as a special version of RLT, $m \times 2$ CV has been widely used in algorithm comparison. Therefore, an optimization of data partitioning in $m \times 2$ CV is further considered. Specifically, the effect of the number of overlapping samples on the variance of $m \times 2$ CV estimator of the generalization error is investigated, and a regularization condition is used to constrain the number of overlapping samples to $n/4$ where n is data set size, and further the $m \times 2$ BCV, in which the variance of the estimator could be reduced, is proposed, and an efficient nested construction of $m \times 2$ BCV is developed. For a text data set, chi-squared statistics are used to measure differences of multiple frequency distributions between training and validation sets. Based on the chi-squared statistics, several regularization conditions are introduced to further optimize $m \times 2$ BCV. Considerable experiments on typical IID data sets and text data sets illustrate the novelty of the developed BCV methods.

In this study, algorithm comparison is formulated into a hypothesis testing problem, and a statistical inference based on $m \times 2$ BCV is investigated. Because the training sets of $m \times 2$ BCV possess overlapping samples, the hold-out estimators of the generalization error in $m \times 2$ BCV are evidently correlated, which makes the inference on $m \times 2$ BCV is distinguishable to the conventional inference methods on IID observations. The upper and lower bounds of the correlation coefficients in an $m \times 2$ BCV estimator is investigated on the basis of abundant theoretical analysis and considerable simulations. Furthermore, a reasonable variance estimator of $m \times 2$ BCV estimator is proposed, and a t-statistic is developed. With a reasonable setting of the correlation coefficients, a conservative t-test statistic is constructed, and sequential confidence intervals (CIs) are given. Different to the conventional sequential tests on IID observations, when repetition count m tends to infinity, the expectation of the length of the sequential CIs in this study converges to a positive constant, which lead to the inexistence of stopping times of the test in some situations. Thus, the reduction rate of the expectation of the length of the sequential CIs is considered as a criterion to select the maximum of stopping time of the sequential t-test. The theoretical and simulated comparisons on several existing tests and the proposed sequential t-test illustrate that the sequential t-test is capable to control the type I error and possesses higher power, and the test makes reliable conclusions. The comparison also shows the necessity of conducting a sequential test rather than fixed size m tests. Moreover, for a text data set, precision, recall and F_1 measure are commonly-used evaluation metrics, and their distributions are skewed. Thus, the sequential t-test is not suitable to these metrics. Therefore, the relationship of the posterior distributions of the three metrics with the correlation coefficients in $m \times 2$ BCV estimators are elaborated, and accurate posterior distributions and reasonable CIs of precision, recall and F_1 measure are provided, and a Bayes test on the posterior distributions is proposed for algorithm comparison. Several experiments on Chinese

word segmentation task and named entity recognition task are conducted to illustrate the validity of the Bayes test.

In this study, the task of software defect prediction is taken as an example to illustrate an application of $m \times 2$ BCV. In a defect count prediction model, the $m \times 2$ BCV sequential t-test is applied to test whether the aggregated features make a significant improvement in the model performance. For defect-prone prediction task, the Bayes test on the basis of $m \times 2$ BCV is applied to test the significance between the model performances of logistic regression and random forest in terms of precision, recall and F_1 measure.

In this study, BCV and the corresponding statistical inference methods are proposed to guarantee the reliability of algorithm comparison conclusions, which have important sense to modeling process of supervised learning methods. The regularization methods in optimal design of data partitioning can be used in sub-sampling on massive data sets to guide the modeling process of distributional learning algorithms.

Key words: Regularization; Cross validation; Data partitioning; Algorithm comparison; Statistical inference; Software defect prediction

第一章　绪论

1.1　研究背景及意义

机器学习模型已被广泛应用于现代智能信息系统中，如，语音识别系统、搜索引擎、信息检索及多模态自动问答系统等。机器学习模型通常使用各种类型的结构化数据或非结构化数据（如，文本数据）作为输入，来训练不同的学习算法并验证它们的性能。模型构建初期，研究者一般采用规模适中的数据，经过特征提取、特征选择、超参数选择、分类器选择及评估等阶段，最后将模型确定下来。之后，研究者再使用大规模的数据再次训练该模型中的参数，以得到较好的参数估计。这个过程中，在分类器选择、特征选择、超参数选择等多个阶段，研究者均需要采用合理的方法来比较多种算法在同一个数据集上产生的多个模型的性能（以下简称**算法比较**）。比如，在特征选择阶段，研究者需要判断新特征对模型的性能提升是否有显著作用；在超参数选择时，研究者需要比较新的超参数选择算法产生的模型与旧算法的模型的性能差异。由此可见，算法比较是机器学习中的一个基本问题，是建模过程中的重要技术。

本文仅考虑两个有监督学习算法的比较问题，并将该问题描述为：**给定两个学习算法及用于算法训练和验证的一个数据集，哪个算法可以产生性能更为优良的模型？**[1] 在算法比较中，交叉验证方法和统计显著性检验常结合使用。交叉验证方法是指对给定的一个数据集进行多次切分，形成多组训练集和验证集，先使用训练集训练机器学习算法，得到相应模型，之后在验证集上得到算法性能指标的估计（简称**指标估计**），最后综合分析多组指标估计来以推断出性能更为优良的算法。常用的交叉验证方法包括 repeated learning-testing （RLT）、K 折交叉验证[2]、5×2 交叉验证、组块 3×2 交叉验证等[3–5]。交叉验证中的多组估计，可用来构造 t 检验统计量，Wilcoxon 秩统计量等，进而结合显著性检验来进行算法比较[1]。在通常的 IID 数据集上，用于算法比较的显著性检验方法主要包括基于 K 折交叉验证的 t 检验方法、基于 5×2 交叉验证的 t 检验及 F 检验方法[6,7]、基于组块 3×2 交叉验证的 t 检验方法[8]、基于均衡 5×2 交叉验证的校正 t 检验[9] 等。

然而，在算法比较任务中，许多学者发现，现有的许多方法所采用的交叉验证方法是不当的，相应的显著性检验常得出不可靠的结论[10–15]。

从交叉验证的角度来看，5 折（10 折）交叉验证的训练集间存在很多重叠样本，对应的多组指标估计间存在较强的相关性。如果将这些指标估计看作独立同分

布，所构建的 t 检验会得出不可靠的比较结论[1,16]。Dietterich 等提出的 5×2 交叉验证[1]，将训练集和验证集的比例“设计”成 1:1，采用 2 折交叉验证来避免相应的两个训练集间存在重叠样本，减少了指标估计间的相关性，保留更大的验证集来改善指标估计的精度，并采用多次重复来增加方差估计的精度，进而产生了更为优良的算法比较方法。不过，5×2 交叉验证容易受到随机切分方式的影响。随机切分方式可能导致训练集间的重叠样本个数增大，也可能导致训练集和验证集间分布差异变大。特别是，后者在文本数据集上尤为明显。例如，汉语框架语义角色标注任务的相关研究也表明[17]：对于同一个语义框架，在汉语句子上，不同词元可配置的语义角色数量和句法模式是不同的，并且许多框架的语义角色的类别多达二三十种。交叉验证若按句子随机切分语料，可能会导致含较多语义角色的句子大多出现在训练集中，而含较少语义角色的句子大多被分到验证集中。这会导致语义角色标注模型的准确率（或召回率）时而偏高，时而偏低，方差较大。这样的准确率（或召回率）会导致算法比较的结论不可靠。再如，使用条件随机场算法构建分词模型时[18]，若将大多数长句子切分到训练集中，而验证集中主要为短句子，这种句子长度分布不均匀的切分，会使分词精度存在明显差异，结果或高或低，方差较大。因此，在算法比较中，研究者应以减少指标估计的方差为目标，对交叉验证的数据切分方式进行优化设计。

从显著性检验的角度来看，常用的假设检验方法并未考虑交叉验证中多组估计间的相关性，采用了偏小的方差估计，难以有效控制检验的第一类错误，易导致假阳性结论。Nadeau 等建议在假设检验中应充分考虑交叉验证估计中的相关性，并提倡构建保守的算法比较方法[16]。交叉验证估计中的相关性，打破了常用假设检验方法的前提假设——多组指标估计应独立同分布。相应的相关系数不仅影响检验统计量的形式，还影响检验统计量所服从分布的形式、参数及自由度。另外，假设检验还应考虑算法性能指标的特点。t 检验和 F 检验通常假设指标估计（近似）服从正态分布，适用于泛化误差、精度等算法性能指标[16]，但不适用于准确率、召回率和 F_1 值等指标，因为准确率、召回率和 F_1 值的分布为偏峰分布[19]。研究者应针对准确率、召回率和 F_1 值的特点，为其设计更为合理的显著性检验。因此，针对算法比较任务，研究者应在充分考虑交叉验证估计的相关系数及算法性能指标特点的前提下，设计合理的交叉验证估计的方差估计、显著性检验方法以及置信区间，形成更优的统计推断方法。

综上，**面向算法比较任务，交叉验证和统计推断方法的选用和优化应考虑数据集、算法性能指标等相关因素的特点，以保障算法比较结论的可靠性**。为此，针对

算法比较任务，本文将研究：**（一）如何设计优良的交叉验证方法，以减小指标估计的方差；（二）在充分考虑交叉验证的多组指标估计间的相关性及算法性能指标特点的前提下，如何构建合理的显著性检验方法，以得到更为可靠的结论**。

本文先从通常的 IID 数据集入手，再拓展到文本数据集。在通常的 IID 数据集上，本文主要考虑有监督的分类和回归问题，且假设数据集中的样例均独立同分布。在文本数据集上，本文主要考虑自然语言处理的分词、命名实体识别，语义角色标注等任务。在面向这些任务的交叉验证方法中，句子为数据切分的基本单元，且不同句子间假定为不相关的。对于算法性能指标，本文主要考虑泛化误差、精度，准确率，召回率以及 F_1 值等常用指标。其中，泛化误差中采用常用的损失函数，如，平方损失或 0-1 损失等。

1.2　数据切分的优化设计：正则化交叉验证方法

1.2.1　设计原则

对于交叉验证方法的设计，本文以减小训练集、验证集的差异，降低算法性能指标估计的方差为目标，从如下三个原则来考虑。

原则一：应使不同切分的两个训练集间的重叠样本个数尽量少且相同

相同数据集上的多次随机切分，会使任意两个训练集间存在重叠样本，而且重叠样本的个数是随机的。Markatou 等证明了重叠样本的个数服从超几何分布[20]。王钰等的研究工作表明，重叠样本个数越多，指标估计的方差也越大[8]。本文引理2.1 进一步揭示重叠样本个数与指标估计的方差间的关系可以用二次函数近似来描述。因此，**交叉验证方法的数据切分中，任意两个训练集间重叠样本个数应尽量少且尽量相同，以减小指标估计的方差**。

原则二：应使训练集和验证集分布尽可能的一致

理论上来说，机器学习要求训练集、验证集的数据分布相同。在通常的 IID 数据集上，随机切分会引入训练集和验证集间的分布差异，增大指标估计的方差。因此，研究者需要对训练集和验证集上的分布差异进行度量，然后使用该度量来指导训练集和验证集中样例的分配。相比结构化数据而言，文本数据是非结构化的，它似乎更难达到“分布相同”这个要求。在随机切分形成的训练集和验证集中，预测标记的分布、词分布等可能存在明显差异，导致指标估计波动较大，且更容易受到交叉验证的切分方法的影响。为此，基于通常的 IID 数据和文本数据各自的特点，交叉验证在切分数据时，应尽量减少训练集和验证集间的分布差异。

原则三：应选择适合算法比较的交叉验证方法，并有效地增加切分次数

在处理通常的 IID 数据时，针对算法比较问题，许多研究表明，常用的 5 折和 10 折交叉验证方法不如 5×2 交叉验证和组块 3×2 交叉验证方法[1,6–8]。5×2 和 3×2 交叉验证可自然拓展为 $m\times2$ 交叉验证。$m\times2$ 交叉验证具体指在给定的数据集上，重复做 m 次 2 折交叉验证，并综合使用 m 次结果来比较算法的性能。重复次数 m 的增加，可以带来更好的指标估计，有利于建立更好的显著性检验方法，如，序贯检验。另外，$m\times2$ 交叉验证的 m 次切分中，每一次数据切分使用对半的比例，相对容易使得训练集和验证集的分布趋于相同，有利于减小分布的不均匀带来的影响。这一点在在文本数据上尤为重要。因此，针对算法比较问题，本文着重研究基于 $m\times2$ 交叉验证的统计推断方法。

基于上述分析，本文引入**正则化**的概念。正则化交叉验证指：面向算法比较问题，基于上述三条原则，通过引入约束（正则化）条件，对交叉验证的切分方式进行优化，形成正则化交叉验证方法。

1.2.2 研究现状

交叉验证估计的方差的深入分析，是优化交叉验证方法的前提。下面，本文从交叉验证估计的方差理论分析和交叉验证方法的优化设计两方面来阐述研究现状。先前的研究工作主要是围绕通常的 IID 数据开展的。非结构化文本数据上的研究工作并不多。

一、交叉验证估计方差的理论分析

在通常的 IID 数据上，早期的一些研究工作主要集中于回归和分类问题上的泛化误差的交叉验证估计方差的理论分析。对于回归问题，McCarthy 等研究了 half-sampling 估计的方差理论表达式[21]。Burman 等给出了留一交叉验证、K 折交叉验证估计、RLT 估计等的方差的理论分解式[22]。特别是，他们给出 RLT 估计的方差的渐进展开形式，从理论上表明，随着切分次数的增加，RLT 估计的方差逐渐减小。对于分类问题，Kohavi 等给出了交叉验证估计的方差近似形式，并模拟分析了 5 折和 10 折交叉验证估计的方差性质[2]。Nadeau 等深入分析了 RLT 估计中的相关性[16]。他们给出了 RLT 估计方差的理论分解形式，并证明 hold-out 估计间的协方差主导了 RLT 估计方差。RLT 估计的方差不会随着验证集大小的增加而增加。Bengio 等证明了 K 折交叉验证估计的方差没有通用的无偏估计[23]。这使研究者意识到交叉验证的方差估计是统计推断中的难点所在。Markatou 等试图为 RLT 估计的方差提供一个渐进无偏估计。他们使用样本矩来分析 RLT 估计的方差，证明了 RLT 方法中重叠样本个数服从超几何分布，清楚地阐述了重叠样本与 RLT 估计的方差间的关系[20]。Rodrígue 等从“敏感度”的角度来研究 K 折交叉验证估计

和 RLT 估计的方差[24,25]。他们将交叉验证估计的方差分解为外部敏感度（由数据随机变化引起的方差）、内部敏感度（切分方法的随机变化引起的方差）和真实误差三部分。他们的模拟实验表明外部敏感度在交叉验证估计方差中占有很大比例。基于交叉验证估计的方差分析，Larsen 等和 Afendras 等进一步尝试确定交叉验证的最优切分比例和切分次数[26,27]。基于方差最小原则，Afendras 等认为数据切分的最优比例应为 1:1。Krstajic 等的研究工作表明 K 折交叉验证估计的方差易导致不可靠的算法比较结论，并推荐在实践中使用多次重复的交叉验证来选择模型[28]。

二、通常的 IID 数据上交叉验证方法的若干优化设计

在通常的 IID 数据上，随机的切分方式会产生训练集和验证集的分布差异。因此，现有的研究工作，主要从这个方面来优化交叉验证方法。

针对训练集和验证集分布差异的优化，当前研究工作主要从响应变量的分布和特征的分布两个侧面入手。响应变量在分类问题中也被称为类别标记。分层交叉验证强制约束类别标记在训练集和验证集上具有相同的分布（比例）。这可以改善泛化误差估计的方差。Breiman 等[29] 分析了分层交叉验证对算法性能估计的影响。在训练集和验证集上的特征的分布差异通常称为数据偏移（data shift ）现象[30]。通常的 IID 数据上，数据偏移现象已有很多研究工作。其中，Sugiyama 等以及 Moreno-Torres 等的研究工作[31,32] 具有代表性。

Sugiyama 等着眼于特征的联合概率分布，首先在训练集和验证集上估计该分布，然后，将所估出的两个分布的比值作为权重来校正泛化误差中的损失函数[31]。这种做法的优点是计算开销下。它可直接沿用通常的数据切分方法；但它的不足之处在于，它所用的特征的联合概率分布不容易被估出。与之不同，Moreno-Torres 等主要研究因切分不当所导致的数据偏移[32]。他们基于欧氏距离来度量数据集中任意两个样例的差异，通过启发式算法将所有样例“均匀”分配到训练集和验证集中，来保证训练集和验证集的分布差异最小。该方法的优点是引入距离度量来优化交叉验证的切分方法。他们验证了该交叉验证在算法比较的秩检验方法中具有优良性。缺点是，特征维度较高或特征取值离散时，欧氏距离往往失效。此外，Diamantidis 等使用聚类方法来减小交叉验证的训练集和验证集中特征分布的差异，来改善交叉验证估计的精度[33]。尽管这些研究主要优化了 K 折交叉验证，其方法可扩展到 $m \times 2$ 交叉验证中。López 等认为对于类别不均衡的分类数据，常用的交叉验证方法因采用随机切分会导致算法比较的结论不可靠。他们提出在切分数据时，应考虑训练集和验证集上数据分布的均衡性，使用“分布均衡”的交叉验证来选择算法[34]。

目前，对于文本数据，交叉验证优化方面的研究工作很少。本文认为，文本数据上的交叉验证方法优化方法，可借鉴 Moreno-Torres 等的思路：先构造训练集与验证集的分布差异的合理度量；然后，基于该度量，建立相应的准则来减少训练集和验证集的分布差异，以增强指标估计的稳定性。不过，不同于通常的 IID 数据，离散特征在文本数据上被广泛使用，如，常用的词特征是高维、稀疏且离散的。这种类型特征的度量方法需要在正则化交叉验证中深入研究。

1.2.3 研究内容

本文以减小指标估计的方差为目标，通过优化交叉验证方法的切分方式，来构建正则化交叉验证。对此，本文将研究内容分为三个阶段。

第一阶段：构建正则化 RLT 方法。具体指，以训练集间重叠样本个数作为度量，分析该度量对模型性能的 RLT 估计的影响，形成正则化条件，来优化 RLT 的切分方式。

RLT 方法指根据预先设定的训练集大小和切分次数，对给定的数据集实施多次随机切分[22,35]，形成的多组随机的 hold-out 验证。因此，RLT 方法也被重复多次重复的 hold-out 验证[36] 或随机交叉验证[37]。

本文首先研究 RLT 交叉验证，是因为：第一、一般的 RLT 方法在切分数据集时，未采用任何约束条件，其估计的理论形式简单，易于分析。第二、现有的很多研究工作已充分分析了泛化误差的 RLT 估计及其方差的主要性质[16,20,22]，这为本文构建正则化交叉验证奠定了良好的基础。第三、$m \times 2$ 交叉验证可以看作 RLT 交叉验证的一种特殊形式。RLT 交叉验证上的理论结果多数适用于 $m \times 2$ 交叉验证。

算法性能指标的 RLT 估计也可看作为多个 hold-out 估计的平均。因此，RLT 估计的方差可以分解成单个 hold-out 估计方差及任意两个 hold-out 估计间协方差的平均。实际上，RLT 估计的精度非常依赖于数据切分的质量。在 RLT 的多个训练集中，任意两个训练集间存在大小随机不定的重叠样本。不好的切分会导致较多的重叠样本会，进而增大 RLT 估计的方差。本文从理论上证明了**两个 hold-out 估计的协方差是关于训练集间重叠样本个数的下凸函数**。该函数为以重叠样本个数角度来优化交叉验证的切分提供了基础。

基于协方差函数的下凸性质，本文尝试优设计 RLT 方法的切分集合，来减少 RLT 估计的方差。在该优化设计中，数据集中的样例应被合理地分配到 RLT 的多个切分中。分配过程应尽量减少训练集间的重叠样本个数，避免增大 RLT 估计的方差。为此，本文引入重叠样本个数作为度量，建立相应的正则化条件，来构建了正则化 RLT 交叉验证。所用的正则化条件主要包含如下两条。

- RLT 中训练集所有重叠样本个数的总和达到最小。这个正则化条件等价于，数据集中的每条样例在所有训练集中出现频次均相同。
- RLT 中任意两个训练集间的重叠样本个数尽量相同。

本文证明了满足上述两个正则化条件的切分集合可减小 RLT 估计的方差。本文使用基于二水平正交表来高效构造正则化 RLT 的切分集合。具体内容见第二 章。

从上述两条正则化条件可知：（1）K 折交叉验证同时满足上述两个正则化条件。因此，K 折交叉验证是正则化 RLT 方法的一种特殊情况。（2）$m \times 2$ 交叉验证仅满足上述的第一个正则化条件。因此，$m \times 2$ 交叉验证可看作仅对 RLT 方法中样本的出现频次进行了正则化。

面向算法比较任务时，组块 3×2 交叉验证和 5×2 交叉验证均表现出了不错的效果[1,6–8]。这些交叉验证方法均是 $m \times 2$ 交叉验证的特殊情形。因此，本文下一步主要关注 $m \times 2$ 交叉验证，并将在正则化 RLT 方法中一些理论结果应用于优化 $m \times 2$ 交叉验证中。

第二阶段：构建正则化 $m \times 2$ 交叉验证方法。具体指，基于 RLT 的分析结果，面向算法比较，以 $m \times 2$ 交叉验证为研究对象，基于“正则化”的思想，优化 $m \times 2$ 交叉验证的切分方式。

$m \times 2$ 交叉验证指对给定的数据集做 m 次随机切分，并实施 m 次 2 折交叉验证。在 $m \times 2$ 交叉验证的所有训练集中，虽然数据集的每条样例出现的频次均相同，但是 $m \times 2$ 交叉验证的性能仍然易受随机切分的影响。也就是说，m 次随机切分对应的多个训练集间存在重叠样本，且重叠样本个数是随机的。重叠样本越多，$m \times 2$ 交叉验证估计的方差越大。因此，本文使用重叠样本个数来度量 $m \times 2$ 交叉验证的切分集合的优劣。然后，本文对重叠样本个数进行“正则化”，来优化 $m \times 2$ 交叉验证的切分，并构建正则化 $m \times 2$ 交叉验证，以减少 $m \times 2$ 交叉验证估计的方差。

基于 RLT 估计中协方差函数的下凸性质，本文进一步证明**两个 2 折交叉验证估计的协方差是关于重叠样本个数的下凸对称函数，且对称轴为 $\boldsymbol{n/4}$**，其中，n 为数据集大小。基于该性质，针对 $m \times 2$ 交叉验证的切分集合，本文建立如下正则化条件：

- $m \times 2$ 交叉验证中任意两个训练集间的重叠样本个数尽量靠近 $n/4$。

本文将满足该正则化条件的 $m \times 2$ 交叉验证称为正则化 $m \times 2$ 交叉验证。本文进一步证明，正则化 $m \times 2$ 交叉验证估计的方差小于随机切分情形下 $m \times 2$ 交叉验证估计的方差。本文基于二水平正交表，给出了正则化 $m \times 2$ 交叉验证的高效增量式构造方法。也就是说，当切分次数 m 逐步变大时，切分集合可以增量式地构造，

不需要重头构造。这为正则化 $m \times 2$ 交叉验证在算法比较任务中的高效应用奠定了重要基础。具体理论分析及正则化 $m \times 2$ 交叉验证的构造见第三 章。

在通常的 IID 数据上，正则化交叉验证可引入训练集和验证集的分布差异的已有的度量[30–33]，来进一步减少算法性能指标估计的方差。下一步，本文主要关注文本数据的正则化 $m \times 2$ 交叉验证的构建。

第三阶段：构建文本数据的正则化 $m \times 2$ 交叉验证方法。具体指，针对文本数据集的特点，构建训练集和验证集分布差异度量及正则化条件，将其融入正则化 $m \times 2$ 交叉验证，提出适用于文本数据集的正则化 $m \times 2$ 交叉验证。

针对文本数据集，本文构建训练集和验证集分布差异的有效度量，来进一步改善正则化 $m \times 2$ 交叉验证的质量。对于大部分自然语言处理任务，文本数据上训练集、验证集的分布差异可以从三个侧面来考虑：（1）预测标记的分布差异，如，语义角色标注任务中，语义角色的类别的分布差异；（2）模型所用的特征的分布差异，如，词频的分布差异；（3）其他相关因素，比如，句子长度分布的差异等。这些都可以归结为两个离散随机变量的分布差异度量问题。解决该问题可以采用多种方法，比如，KL 距离，卡方统计量等。这需要研究哪种方法适合度量什么差异。这里称上述分布的差异度量为常用度量。

本文基于常用度量，提出了构建正则化交叉验证方法的基本思路，并将文本数据上的正则化 $m \times 2$ 交叉验证的求解形式化为一个带有正则化参数的优化问题。在此基础上，本文主要讨论了基于卡方统计量的度量函数，并给出了正则化参数的选择方法。文本数据上的大多任务采用准确率、召回率、F_1 作为算法性能指标，因此，本文以最大化相应指标的信噪比为目标，来寻找优化的交叉验证方法。特别是，本文以语义角色标注实验为例，分别使用卡方统计量来度量“词元分布”和“语义角色类型分布”在训练集和验证集上的差异，并以此构造正则化条件，来进一步优化正则化 $m \times 2$ 交叉验证。实验结果表明，引入训练集和验证集分布差异的正则化条件后，正则化 $m \times 2$ 交叉验证估计的信噪比有了明显提升。这为后续深入研究文本数据上的正则化交叉验证方法提供了很好的支持。具体内容见第四 章。

1.2.4 解决的关键问题：正则化交叉验证的高效构造算法

对于正则化交叉验证，从理论上论证其存在性固然重要，更重要的是给出高效的构造算法。正则化交叉验证中的正则化条件通常要求重叠个数等度量满足在切分中满足“均衡”性质。因此，可以考虑采用实验设计中的正交表工具[38] 进行构造。对于正则化 RLT 方法，对于任意的训练集大小和切分次数，是否存在通用的高效的构造算法，目前仍是公开问题（见第2.2.5 节）。不过，在三种特定的情形下，本

文给出了正则化 RLT 的高效构造方法。特别是，当训练集的大小为 $n/2$ 时（n 为数据集大小），正则化 RLT 可被高效构造。对于正则化 $m \times 2$ 交叉验证，基于二水平正交表，本文给出了其高效增量式构造算法，为后续的序贯检验奠定了很好的基础。对于文本数据集上的正则化 $m \times 2$ 交叉验证，需要融合多种正则化条件。这就要求增加正则化条件个数时，切分集应递增式构造，决不能每增加一个正则化条件，就要重新构造切分集。本文在汉语框架语义角色标注任务中，对文本数据的正则化 $m \times 2$ 交叉验证的构造进行了初步尝试。此外，本文提出的正则化交叉验证的构造算法，均建立在数据块上，而不是单条样例上，因此，对于不同大小的数据集，这些构造算法均有良好的扩展性。

1.3　基于正则化 $m \times 2$ 交叉验证的统计推断方法

基于构造出的正则化 $m \times 2$ 交叉验证，本节阐述正则化 $m \times 2$ 交叉验证上的统计推断方法。

针对算法比较任务，交叉验证估计的方差估计在显著性检验方法中起着重要的作用。然而，Bengio 等指出交叉验证估计的方差没有通用的无偏估计[23]。在统计推断方法中，常用的方差估计通常低估了真实方差，会推导出激进的假设检验方法。激进的假设检验方法容易错误地声明两算法间存在显著差异，增大显著性检验的第一类错误，产生假阳性的算法比较结论，误导后续的研究工作。Nadeau 等不推荐在算法比较任务中使用激进的统计推断方法[16]。相反，本文通过对真实方差略微高估，构造相对保守的统计推断方法，有利于得到可靠的结论。

不同的算法性能指标的概率分布也不尽相同。Nadeau 等认为，泛化误差的分布近似服从正态分布[16]，符合 t 检验统计量的前提假设。因此，在这些算法性能指标上使用 t 检验是合理的。但是，准确率、召回率和 F_1 值等指标的分布为偏峰分布，且它们的取值范围为 [0,1]。t 检验并不适用于这些算法性能指标的比较，所得的比较结论也不可靠。这表明，算法性能指标的概率分布直接决定着统计推断方法。在构造统计推断方法时，应充分考虑算法性能指标的概率分布。

1.3.1　研究现状

一、通常的 IID 数据上基于交叉验证的算法比较方法

通常的 IID 数据上，为获得可靠的算法比较结论，很多研究工作在不断地改进显著性检验方法。基于 5 折（10 折）交叉验证的 t 检验，因简单易用，被研究者广泛采用。然而，该方法给出的方差的估计偏小，难以有效控制检验的第一类错误，易导致假阳性的结论。McNemar 检验也是一种重要的算法比较方法[39]。但是，

Keller 等指出当所比较的两个机器学习算法较为类似时，McNemar 检验容易给出不可靠的算法比较结论[40]。Dietterich 等比较了算法比较的多种显著性检验方法，发现基于 5×2 交叉验证的 t 检验具有比基于 5 折和 10 折交叉验证的 t 检验更优的势[1]。随后，Bouckaert 等指出 5×2 交叉验证 t 检验给出的算法比较结论的复现度较低，主要原因是该 t 检验统计量的分子中仅使用了从第一次 2 折交叉验证中的一个 hold-out 估计[41,42]。为了弥补该缺点，Alpaydin 等提出了性能更好的基于 5×2 交叉验证的 F 检验，以提高检验的势以及算法比较结论的复现度。不过，由于 F 检验自身的特点，该检验仅可用于比较两个算法的性能是否等价，并不适用于检验两个算法性能指标的差距是否超过用户给出的阈值[6]。后续，Yildiz 等提出的基于 5×2 交叉验证的合并 t 检验，使用了 5×2 交叉验证估计作为 t 检验统计量的分子[7]，以增强算法比较结论的复现度。但是，该 t 检验统计量假设 5×2 交叉验证的多组估计间相互独立，导致方差估计低估了真实方差。因此，基于 5×2 交叉验证的合并 t 检验难以有效控制第一类错误，易导致假阳性的结论。Nadeau 等分析了 RLT 的多组 hold-out 估计间的相关系数，给出了该相关系数的一个估计，并使用该相关系数的估计来增大方差估计，以改进 t 检验统计量，进而提高了算法比较结论的可靠性[16]。但是，他们并没意识到该相关系数不仅影响方差估计，还会影响到检验统计量的形式、检验统计量所服从的分布以及分布的参数及自由度等。Grandvalet 等在验证集上计算出 hold-out 估计间的相关性，并用它来校正 t 检验，以控制检验的第一类错误[43]。Isaksson 等人发现在小样本分类问题上，K 折交叉验证方法由于低估了算法性能的方差，导致了不可靠的统计推断结论[44]。Shen 等人从交叉验证切分的随机性及计算开销角度入手，提出了一种序贯交叉验证方法用于算法比较和选择[45]。Wang 等人给出了交叉验证估计的一种优良的方差估计[46]。该估计对算法比较性能的改进，有待于进一步论证。因此，为了改进显著性检验方法，本文需要深入研究交叉验证的多组估计间的相关性对统计推断的影响，并依此构造含方差估计、显著性检验等在内的优良的统计推断方法。

二、文本数据上基于交叉验证方法的算法比较方法

在文本数据的相关研究上，不用或滥用算法比较方法的现象非常明显。Dror 等统计了 2017 年 ACL 会议及 TACL 期刊中所有论文使用交叉验证和显著性检验的情况（ACL-2017 会议含论文 196 篇，TACL-2017 期刊含论文 37 篇），分析了自然语言处理领域显著性检验的应用现状[47]。他们发现，ACL-2017 中有 117 篇没有使用显著性检验，占比 60%，有 6 篇使用了错误的显著性检验；TACL-2017 中有 15 篇没有使用显著性检验，占比 41%；ACL-2017 中仅 23 篇使用了交叉验证，而

TACL-2017 中仅 5 篇使用了交叉验证。这些统计数据表明，使用“交叉验证 + 显著性检验”的方法来比较文本数据的算法能并没有得到广泛认可。主要的原因有两点：（1）文本数据上的算法性能指标种类很多，常用的显著性检验方法并不完全适用于这些指标。例如，Yeh 等指出准确率、召回率和 F_1 值不适合使用 t 检验进行比较[48]。（2）交叉验证的随机切分容易导致文本数据上的指标估计波动很大，使得常用的显著性检验给出的算法比较结论不可靠。

实际上，过去几十年中，仍有很多研究工作致力于验证常用的显著性检验方法是否适用于对比文本数据上的自然语言处理模型[47,49–58]。这些研究发现常用的“交叉验证 + 统计显著性检验”方法容易产生不可靠的算法比较结论。特别是，文本数据规模不大时，交叉验证希望通过反复切分数据，来提高指标估计的精度。但是，采用随机切分容易导致训练集、验证集的分布差异变大，造成指标估计的波动较大，结论的复现度不高。Halberstadt 明确表示不推荐将常用的交叉验证方法应用于语音识别任务[59]，其原因是交叉验证方法的验证集分布和数据的真实分布有很大差异，很可能挑选出不好的模型。同样，在基于 McNemar 检验的语音识别模型比较研究中，Gillick 等也指出指标估计容易受到数据集分布差异的影响[49]，得出不可靠的算法比较结论。Søgaard 等指出，文本数据集上算法比较的结论易受文本集大小、句子长度、验证集大小及算法性能指标等因素的影响[56]。若不考虑这些因素，直接采用常用的显著性检验方法，得到的结论并不可靠。为了在文本数据上得到可靠的算法比较结论，Søgaard 等进一步建议将显著性检验的显著性水平从 0.05 缩小至 0.0025[56]。当显著性检验的 p 值小于 0.0025 时，才应认为文本数据上两个算法所产生的模型的性能指标有显著差异。在词义消歧、词性标注等多个任务上，Daelemans 等使用 10 折交叉验证来选择算法并估计模型性能，得到的结果很不理想[51]。他们指出，在算法比较中，常用的交叉验证容易将算法性能之间的差异“淹没”到误差之中，进而推导出不可靠的比较结论。他们这样评述：“There is also no reason to believe that any comparative conclusions drawn on the basis of standard comparative machine learning methodology will hold when sufficient optimization of algorithm parameters and feature selection is available.”。类似的结论在 Berg-Kirkpatrick 等、Søgaard 等以及 Yeh 等的工作中也有详细的论述[48,54,56]。近年来，语义角色标注任务上的相关研究也发生了类似的现象[60]。为此，李济洪等提出组块 3×2 交叉验证方法进行算法比较[17]，也尝试将这种方法应用到分词、短语识别，语义角色识别等任务的模型比较中[18,61,62]。他们认为，在给定的规模适中文本数据集上，使用 2 折交叉验证更有利于使得训练集、验证集的分

布接近，能得到更稳定的指标估计。

三、针对准确率、召回率和 F_1 值上的算法比较方法

在通常的 IID 数据的分类任务以及文本数据的很多自然语言处理任务上，准确率、召回率和 F_1 值是普遍使用的算法性能指标。由于准确率、召回率和 F_1 的概率分布通常为偏峰分布，常用的 t 检验和 F 检验并不适用于这些算法性能指标上的比较。目前，针对算法比较任务，这些指标上统计推断方法的研究工作相对较少。其面临的主要障碍是无法获取这些指标的概率分布。对此，Goutte 等证明了 hold-out 验证下准确率和召回率的后验分布为 Beta 分布[19]。该后验分布使得从统计意义下直接比较模型的准确率和召回率成为可能[63] [64]。为了估计和比较算法的 F_1 值，Zhang 等使用了复杂的概率图模型和层次贝叶斯模型[65–67]。最近，王钰等证明了 F_1 值的后验分布的显式形式[68]。他们的工作表明，F_1 值的后验分布为关于 Beta-prime 分布的一个函数。另外，Caelen 等给出了混淆矩阵的一个贝叶斯解释，并提出了一种方法可推断出从混淆矩阵中导出的常用指标的后验分布[69]。这些研究工作为在准确率、召回率和 F_1 值上进一步构建基于正则化 $m \times 2$ 的统计推断方法奠定了重要的基础。

1.3.2 研究内容

对于算法比较任务，本文的统计推断方法主要研究对于给定的一个数据集及两个有监督学习算法，基于正则化 $m \times 2$ 交叉验证估计，如何构建合理的方差估计、显著性检验方法及置信区间，以得出可靠的算法比较结论。本文的研究内容主要分三个阶段展开。

第一阶段：针对正则化 $\boldsymbol{m \times 2}$ 交叉验证估计，构造一个合理的方差估计。

通用的正则化 $\boldsymbol{m \times 2}$ 交叉验证估计的方差的无偏估计并不存在[23,70]。其主要原因是，正则化 $m \times 2$ 交叉验证估计中的相关系数是不可估的[71]，也就是说，给定单个数据集，无法合理地估计出正则化 $m \times 2$ 交叉验证估计中的相关系数。因此，**如何估计正则化 $m \times 2$ 交叉验证估计的方差，成为面向算法比较统计推断的核心问题。**

针对算法比较的统计推断，Nadeau 等提出一个指导性原则[16]：**应该选择一个相对保守的方差估计。**所谓保守指相比真实方差，方差估计的期望值应相对偏大。他们指出，若使用激进的方差估计，低估了真实方差，容易产生假阳性的算法比较结论。

本文提出了正则化 $m \times 2$ 交叉验证的三种不同方差估计，并分析了它们的保守性质。基于理论分析和模拟验证，本文选出其中一种合理的方差估计，用于后续的

显著性检验方法中，具体内容见本文第五 章。

第二阶段：基于正则化 $m \times 2$ 交叉验证，构建优良的显著性检验方法。

针对泛化误差，本文提出了正则化 $m \times 2$ 交叉验证序贯 t 检验。该序贯 t 检验采用保守的方差估计来构造检验统计量，以有效控制第一类错误。所谓“序贯”，是指通过逐步增加重复次数 m，来增量式地获得相应的 hold-out 估计，并将其用于 t 检验中。采用“序贯”的思想，主要是考虑到在不同的算法比较任务中，所需要的重复次数 m 是不同的。例如，当数据集不同或算法不同时，正则化 $m \times 2$ 交叉验证估计中的相关性也会不同。若相关性较小，则 $m \times 2$ 交叉验证估计的精度高，可能需要少量的重复次数 m 便可以检验出算法性能的差异。反之，若相关性较大，则可能需要较多的重复次数 m。本文在模拟数据和真实数据上对比了不同的显著性检验方法。实验结果表明正则化 $m \times 2$ 交叉验证序贯 t 检验具有更小的一类错误和更优的势函数。此外，实验结果也表明了“序贯”的必要性。具体内容见本文第四章。

针对准确率、召回率和 F_1 值，本文提出了一种贝叶斯检验方法来检验两个算法在这些指标上的差异。在该检验中，正则化 $m \times 2$ 交叉验证估计中的相关系数被用来校正其准确率、召回率和 F_1 值的后验分布。在此基础上，将准确率、召回率和 F_1 值的比较问题形式化成假设检验问题。然后，采用蒙特卡罗方法来给出两个算法的准确率、召回率和 F_1 之差的分布函数及假设成立的概率。该贝叶斯检验将每个假设成立的概率作为输出。因此，该检验可以提供丰富的信息以供后续决策，且该检验一定程度上弥补了传统假设检验存在的一些问题[71]，比如，p 值的问题[72–74]。

第三阶段：基于正则化 $m \times 2$ 交叉验证，构造算法性能指标的置信区间。

针对泛化误差，本文基于序贯 t 检验统计量，给出了相应的序贯置信区间，并从理论上分析了该置信区间的相关性质。具体见本文第5.2.5.2 节。对于准确率、召回率和 F_1 值，基于正则化 $m \times 2$ 交叉验证，本文基于它们的后验分布，给出了它们的后验置信区间，通过实验证明这些置信区间比王钰等给出的置信区间[68] 更为精准。具体内容见第6.2.3 节。

1.3.3　解决的关键问题：算法性能指标的任意两个 2 折交叉验证估计间的相关性分析

正则化 $m \times 2$ 交叉验证的所有切分均实施在同一个数据集上，因此，这些切分对应的 hold-out 估计间明确存在着较强的相关性。这些相关性关系到算法比较的推断方法的有效性和算法比较结论的可靠性。因此，深入分析 hold-out 估计间的相关

系数是建立优良的统计推断方法的重要基础。对于单个算法的泛化误差，本文分析了其正则化 $m \times 2$ 交叉验证估计的相关系数的取值范围，发现对于大多数算法和数据集，相关系数均介于 0 到 1/2 之间。具体来讲，单个 2 折交叉验证中的任意两个 hold-out 估计的相关系数介于 0 到 1/2 之间，且随着数据集增大而趋于 0；两个不同的 2 折交叉验证的 hold-out 估计间的相关系数介于 1/4 到 1/2 之间。对于两个算法的泛化误差之差，本文揭示了该差的正则化 $m \times 2$ 交叉验证估计中的相关系数小于单个算法上的相关系数。在正则化 $m \times 2$ 交叉验证统计推断中，关于相关系数的这些理论分析为重复次数 m 的选取、方差估计的构造以及序贯 t 检验的建立奠定了基础。此外，对于准确率、召回率和 F_1 值，本文分析了正则化 $m \times 2$ 交叉验证估计中的相关系数与它们的后验分布的关系。这为构造更为精准的后验置信区间及贝叶斯检验提供了前提。

1.4 本文的主要贡献

本文的贡献主要有如下几点：

(1) **给出了正则化 RLT 交叉验证及其构造算法**。本文证明了泛化误差的 RLT 估计中，任意两个 hold-out 估计的协方差是关于它们的训练集间重叠样本个数的下凸函数。在重叠样本个数上引入正则化条件，通过最小化 RLT 估计的方差，来优化 RLT 方法的切分集合。进而，建立了正则化 RLT 方法，并证明泛化误差的正则化 RLT 估计的方差小于随机切分情形下 RLT 估计的方差。对关于切分次数和训练集大小的三种不同的设置，本文给出正则化 RLT 切分集的三种不同构造方法。特别是，对于训练集大小为数据集大小一半时的正则化 RLT，本文基于二水平正交表给出了其简单易用的切分构造方法，并证明了其最小方差性质。

(2) **提出了正则化 $m \times 2$ 交叉验证方法并给出其增量式构造算法**。本文深入分析了泛化误差的 $m \times 2$ 交叉验证估计的方差，揭示了任意两个 2 折交叉验证估计的协方差是关于它们的训练集间重叠样本个数的下凸对称函数，且对称轴为 $n/4$ （n 为数据集大小）。本文引入正则化条件，将 $m \times 2$ 交叉验证中所有的重叠样本个数约束至 $n/4$ 左右，来优化 $m \times 2$ 交叉验证的切分方式，并进一步提出正则化 $m \times 2$ 交叉验证。本文证明正则化条件可以减小泛化误差的正则化 $m \times 2$ 交叉验证估计的方差。进而，基于二水平正交表，本文给出了正则化 $m \times 2$ 交叉验证的一种高效增量式构造算法。在此基础上，针对文本数据，本文进一步引入正则化条件来约束训练集和验证集分布差异，进而构造适用于文本数据的正则化 $m \times 2$ 交叉验证方法。汉语框架语义标注任务上的实验说明了文本数据的正则化 $m \times 2$ 交叉验证方法的优

良性。

(3) **基于正则化 $m \times 2$ 交叉验证，构建了用于算法比较的一种序贯 t 检验**。本文深入研究了面向算法比较的正则化 $m \times 2$ 交叉验证上的统计推断方法。针对单个机器学习算法，本文深入分析了正则化 $m \times 2$ 交叉验证估计中相关性的理论性质及取值范围。针对泛化误差的正则化 $m \times 2$ 交叉验证估计，本文给出了其方差的一个合理的保守估计，并深入分析了该估计与 hold-out 估计的相关性之间的关系。对于算法比较任务，本文揭示了两个算法性能指标之差的正则化 $m \times 2$ 交叉验证估计中的相关性与单个算法的相关性之间的关系。基于正态假设，构造了一个相对保守的序贯 t 检验统计量，并给出了序贯置信区间，理论和实验结果验证了该序贯 t 检验为保守的统计推断，可以得到更为可靠的结论。

(4) **给出了基于正则化 $m \times 2$ 交叉验证的准确率、召回率、$\mathbf{F}_1$ 指标的贝叶斯检验方法**。本文研究了基于正则化 $m \times 2$ 交叉验证的准确率、召回率和 F_1 值的统计推断问题，阐明了正则化 $m \times 2$ 交叉验证估计中的相关性与准确率、召回率和 F_1 值的后验分布间的关系。基于正则化 $m \times 2$ 交叉验证，本文给出了准确率、召回率和 F_1 值的精准的后验分布及合理的后验置信区间。以此为基础，本文将基于准确率、召回率和 F_1 值的算法比较问题形式化成统计显著性检验问题，并给出了一种贝叶斯检验方法。该贝叶斯检验方法可以直接计算出所设立假设成立的概率，以及相应的决策。相比常用的假设检验方法，该贝叶斯假设不仅避免了常用的假设检验方法的一些缺点，还可提供更为丰富的决策信息。本文在文本数据的分词和命名实体识别任务上验证了该贝叶斯检验的有效性。

本文提出的正则化交叉验证及其统计推断方法，有利于提高算法比较结论的可靠性，对特征选择、模型选择以及评估等具有重要意义。本文的研究方法可拓广至其它非独立同分布的数据集上。特别是，正则化交叉验证的数据切分方法，可扩展到大规模数据的子抽样上，为分布式机器学习提供新的研究思路，也为大规模结构化数据集和文本数据集提供新的建模方法。

1.5　本文的内容安排

本文主体内容分为八章。其中，第二章至第四章主要研究了正则化交叉验证的理论性质和构造方法；第五章和第六章研究了基于正则化 $m \times 2$ 交叉验证的算法比较统计推断方法。第七章将正则化 $m \times 2$ 交叉验证应用在软件缺陷预测任务中。具体内容安排如下。

第一章为主要介绍了课题研究背景意义，以及本领域国内外研究进展，阐述了

现今面临的问题和挑战，并概括了本文的研究目标、研究内容及解决的关键问题。

第二章和第三章阐述了正则化 RLT 方法和正则化 $m \times 2$ 交叉验证的理论性质及构造方法。

第四章详细叙述了面向文本数据的正则化交叉验证方法。

第五章详细叙述了面向算法比较的正则化 $m \times 2$ 交叉验证的统计推断方法。

第六章基于正则化 $m \times 2$ 交叉验证给出了准确率、召回率和 F_1 值等指标上算法比较的统计推断方法。

第七章给出了正则化 $m \times 2$ 交叉验证在软件缺陷预测任务中的一个简单应用。

最后总结全文，并对今后的研究工作进行展望。

第二章　正则化 RLT 方法

RLT 方法通过对数据集多次切分以充分利用所有的观测。这使得 RLT 比常用的封闭测试[75] 或 hold-out 验证[76] 有着更好的性能。然而，泛化误差的 RLT 估计的精度（方差）非常依赖于数据切分的质量。在 RLT 中，数据集上被随机独立地切分为多个训练集和验证集。任意两个训练集间存在大小随机的重叠样本。过量的重叠样本将不必要的相关性引入到 RLT 估计中，增大该估计的方差。另外，在 RLT 的多个训练集中，每条样例出现的频次是随机的。这些随机频次会直接影响训练集间重叠样本个数之和，进而增大 RHS 估计的方差。此外，由于随机切分所导致的训练集和验证集的分布差异也会导致 RLT 估计不稳定。因此，RLT 的随机切分，并不适用于算法比较任务，需要研究新的切分方法。本章主要研究 RLT 方法的随机切分与 RLT 估计的方差之间的关系，并通过优化 RLT 的数据切分方法，以减少泛化误差的 RLT 估计的方差，进而构建正则化 RLT 方法。

2.1　记号及定义

假定数据集 $D_n = \{z_i : z_i = (\mathbf{x}_i, y_i), i = 1, \ldots, n\}$ 的 n 个独立同分布的样例来自于某一未知分布 $\mathcal{P}$。其中，$\mathbf{x}_i \in \mathbb{R}^p$ 为特征向量，$y_i \in \mathbb{R}$ 为响应变量。给定机器学习算法 $\mathcal{A}$，可定义损失函数 $L(\mathcal{A}(D_n), z)$。该损失函数度量了算法 $\mathcal{A}$ 在数据集 D_n 上训练的算法 $\mathcal{A}(D_n)$ 在某一个测试样例 z 上所产生的错误。其中，$z \sim \mathcal{P}$。常用的损失函数包括回归任务的平方损失、绝对损失，及分类任务的 0-1 损失。算法 $\mathcal{A}$ 在数据集 D_n 上的泛化误差为损失函数的期望值。即：

$$\mu(n) \triangleq E_{D_n, z}[L(\mathcal{A}(D_n), z)]。 \tag{2.1}$$

为清晰描述泛化误差的 RLT 估计，本章引入如下的记号：

- 记 $\mathcal{S} = (I^{(t)}, I^{(v)})$ 为指标集 $\mathcal{I} = \{1, 2, \ldots, n\}$ 上的一个**切分**。其中，$I^{(t)}, I^{(v)} \subset \mathcal{I}$，$I^{(t)} \cup I^{(v)} = \mathcal{I}$ 且 $I^{(t)} \cap I^{(v)} = \emptyset$。
- 记 $D^{(t)} = \{z_i : i \in I^{(t)}\}$ 和 $D^{(v)} = \{z_i : i \in I^{(v)}\}$ 分别为定义在切分 $\mathcal{S} = (I^{(t)}, I^{(v)})$ 上的**训练集**和**验证集**。其中，$D^{(t)}$ 和 $D^{(v)}$ 可泛称为**数据块**。$|D^{(t)}| = n_1$ 和 $|D^{(v)}| = n - n_1 = n_2$ 为**训练集大小**和**验证集大小**。不失一般性，本章假定 $n_2 \leq n_1 < n$。
- 记 $\hat{\mu}_{HO}(\mathcal{S}) \triangleq 1/n_2 \sum_{j \in D^{(v)}} L(\mathcal{A}(D^{(t)}), z_j)$ 为泛化误差 $\mu(n)$ 的 **hold-out 估计**。
- 记 $\mathbb{S} = \{\mathcal{S}_j : \mathcal{S}_j = (I_j^{(t)}, I_j^{(v)}), j = 1, 2, \ldots, J\}$ 为 RLT 的一个大小为 J 的**切分集**。

- 记 $\phi_{jj'} = |I_j^{(t)} \cap I_{j'}^{(t)}|$ 为切分 $\mathcal{S}_j$ 和 $\mathcal{S}_{j'}$ 间的**重叠样本个数**。其中，$j, j' = 1, 2, \ldots, J$。记 $\phi_{jj'} = x$，则 $n_1 - n_2 \leq x \leq n_1$。重叠样本个数矩阵 $\mathbf{\Phi} = (\phi_{jj'})$ 为切分集 $\mathbb{S}$ 的一个**度量**。
- 记 ψ_i 为样例 z_i 在切分集 $\mathbb{S}$ 上所有 J 个训练集中的**出现频次**。设 $\psi_i = c$，则 $0 \leq c \leq J$。在 D_n 上，有 $\sum_{i=1}^{n} \psi_i = n_1 J$。定义向量 $\mathbf{\Psi} = (\psi_1, \ldots, \psi_n)$。在切分集 $\mathbb{S}$ 上，$\mathbf{\Psi}$ 与 $\mathbf{\Phi}$ 有如下关系：

$$\sum_{j=1}^{J} \sum_{j'=j+1}^{J} \phi_{jj'} = \sum_{i=1}^{n} \binom{\psi_i}{2}, \tag{2.2}$$

其中，$\binom{0}{2} = \binom{1}{2} = 0$。

- 记 $\hat{\mu}_{RLT}(\mathbb{S}) \triangleq 1/J \sum_{j=1}^{J} \hat{\mu}_{HO}(\mathcal{S}_j)$ 为泛化误差 $\mu(n)$ 在切分集 $\mathbb{S}$ 上的 RLT 估计。

易知，$\hat{\mu}_{RLT}(\mathbb{S})$ 为 $\mu(n_1) = E_{D^{(t)},z}[L(\mathcal{A}(D^{(t)}), z)]$ 的一个无偏估计，且 $\mu(n_1)$ 与 $\mathbb{S}$ 无关[16]。但是，$Var(\hat{\mu}_{RLT}(\mathbb{S}))$ 易受切分集 $\mathbb{S}$ 的影响[24]。对于方差 $\mathrm{Var}[\hat{\mu}_{RLT}(\mathbb{S})]$，Nadeau 等、Markatou 等以及 Afendras 等给出了相关的理论分析[16,20,37]。基于这些研究工作，本节重点阐述泛化误差的 RLT 估计与其切分之间的关系。

RLT 估计的方差有如下分解[16,20,37]：

$$\mathrm{Var}[\hat{\mu}_{RLT}(\mathbb{S})] = \frac{1}{J^2} \sum_{j=1}^{J} \mathrm{Var}[\hat{\mu}_{HO}(\mathcal{S}_j)] + \frac{2}{J^2} \sum_{j=1}^{J} \sum_{j'=j+1}^{J} \mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_j), \hat{\mu}_{HO}(\mathcal{S}_{j'})]。 \tag{2.3}$$

其中，$\mathrm{Var}[\cdot]$ 和 $\mathrm{Cov}[\cdot, \cdot]$ 均取自数据集 D_n 和切分集 $\mathbb{S}$。式（2.3）右边两项分别有如下展开式。

对于第一项，有如下分解。

$$\begin{aligned} \mathrm{Var}[\hat{\mu}_{HO}(\mathcal{S}_j)] &= \mathrm{Var}_{\mathcal{S}_j}[E_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j)|\mathcal{S}_j]] + E_{\mathcal{S}_j}[\mathrm{Var}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j)|\mathcal{S}_j]] \\ &= \mathrm{Var}_{\mathcal{S}_j}[\mu(n_1)] + E_{\mathcal{S}_j}[\mathrm{Var}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j)|\mathcal{S}_j]] \\ &= E_{\mathcal{S}_j}[\mathrm{Var}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j)|\mathcal{S}_j]]。 \end{aligned} \tag{2.4}$$

其中，$\mathrm{Var}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j)|\mathcal{S}_j]$ 不依赖于切分 $\mathcal{S}_j$ 的具体值，因为损失函数 $L(\mathcal{A}(D^{(t)}), z_k)$ 的分布不依赖于切分 $\mathcal{S}$ 的具体值及指标 k[16,20]。因此，记 $\mathrm{Var}[\hat{\mu}_{HO}(\mathcal{S}_j)]$ 为常数 σ_1^2。

对于第二项 $\mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_j), \hat{\mu}_{HO}(\mathcal{S}_{j'})]$，有如下分解。

$$\begin{aligned} & \mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_j), \hat{\mu}_{HO}(\mathcal{S}_{j'})] \\ = \ & \mathrm{Cov}_{\mathcal{S}_j,\mathcal{S}_{j'}}[E_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j)|\mathcal{S}_j], E_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_{j'})|\mathcal{S}_{j'}]] + E_{\mathcal{S}_j,\mathcal{S}_{j'}}[\mathrm{Cov}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j), \hat{\mu}_{HO}(\mathcal{S}_{j'})|\mathcal{S}_j, \mathcal{S}_{j'}]] \\ = \ & \mathrm{Cov}_{\mathcal{S}_j,\mathcal{S}_{j'}}[\mu(n_1), \mu(n_1)] + E_{\mathcal{S}_j,\mathcal{S}_{j'}}[\mathrm{Cov}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j), \hat{\mu}_{HO}(\mathcal{S}_{j'})|\mathcal{S}_j, \mathcal{S}_{j'}]] \\ = \ & E_{\mathcal{S}_j,\mathcal{S}_{j'}}[\mathrm{Cov}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j), \hat{\mu}_{HO}(\mathcal{S}_{j'})|\mathcal{S}_j, \mathcal{S}_{j'}]]。 \end{aligned} \tag{2.5}$$

因为数据集 D_n 中样本均独立同分布，所以协方差 $\mathrm{Cov}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j),\hat{\mu}_{HO}(\mathcal{S}_{j'})|\mathcal{S}_j,\mathcal{S}_{j'}]$ 可看作重叠样本个数 $\phi_{jj'}$ 的一个函数。从度量 $\mathbf{\Phi}$ 的角度分析该协方差函数，则有

$$\mathrm{Cov}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j),\hat{\mu}_{HO}(\mathcal{S}_{j'})|\mathcal{S}_j,\mathcal{S}_{j'}]=\mathrm{Cov}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j),\hat{\mu}_{HO}(\mathcal{S}_{j'})|\phi_{jj'}=x]。\tag{2.6}$$

其中，$j\neq j'$ 且 $j,j'=1,\ldots,J$。

Markatou 等证明了重叠样本个数 $\phi_{jj'}$ 服从超几何分布[20]，其概率分布函数为

$$P(\phi_{jj'}=x)=\frac{\binom{n_1}{x}\binom{n-n_1}{n_1-x}}{\binom{n}{n_1}}。\tag{2.7}$$

其中，$E[\phi_{jj'}]=n_1^2/n$，$\mathrm{Var}[\phi_{jj'}]=n_1^2(n-n_1)^2/(n^2(n-1))$。

将式（2.4），（2.5）和（2.7）代入式（2.3）中，则有

$$\mathrm{Var}[\hat{\mu}_{RLT}(\mathbb{S})] = \frac{1}{J}\sigma_1^2+\frac{J-1}{J}\sum_{x=n_1-n_2}^{n_1}\frac{\binom{n_1}{x}\binom{n-n_1}{n_1-x}}{\binom{n}{n_1}}\mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_j),\hat{\mu}_{HO}(\mathcal{S}_{j'})|\phi_{jj'}=x]。\tag{2.8}$$

其中，$\mathrm{Var}[\cdot]$ 和 $\mathrm{Cov}[\cdot,\cdot]$ 均取自 D_n 和 $\mathbf{\Phi}$。因此，有 $\mathrm{Var}[\hat{\mu}_{RLT}(\mathbb{S})]=E_{\mathbf{\Phi}}\mathrm{Var}_{D_n}[\hat{\mu}_{RLT}(\mathbb{S})|\mathbf{\Phi}]$。

基于上述分析可知，若约束切分集 $\mathbb{S}$ 中 $\mathbf{\Phi}$ 的取值，则可能会减少 RLT 估计的方差。假设 $\mathbf{\Phi}'=(\phi'_{jj'})$ 为约束 $\mathbf{\Phi}$ 后的重叠个数，并设 $\mathbf{\Phi}'$ 对应的切分集为 $\mathbb{S}'=\{\mathcal{S}'_j\}$，则基于 $\mathbb{S}'$ 的 RLT 估计的方差有如下形式：

$$E_{\mathbf{\Phi}'}[\mathrm{Var}[\hat{\mu}_{RLT}(\mathbb{S}')|\mathbf{\Phi}']]=\frac{1}{J}\sigma_1^2+\frac{J-1}{J}E_{\phi'_{jj'}}\mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}'_j),\hat{\mu}_{HO}(\mathcal{S}'_{j'})|\phi'_{jj'}=x]。\tag{2.9}$$

若 $E_{\mathbf{\Phi}'}[\mathrm{Var}\mu_{RLT}[\mathbb{S}')|\mathbf{\Phi}']]>E_{\mathbf{\Phi}}[\mathrm{Var}[\mu_{RLT}(\mathbb{S})|\mathbf{\Phi}]]$，则将 $\mathbb{S}'$ 称为坏切分集。若 $E_{\mathbf{\Phi}'}[\mathrm{Var}\mu_{RLT}[\mathbb{S}')|\mathbf{\Phi}']]\leq E_{\mathbf{\Phi}}[\mathrm{Var}[\mu_{RLT}(\mathbb{S})|\mathbf{\Phi}]]$，则 $\mathbb{S}'$ 为优良的切分集。研究者通常青睐优良的切分集 $\mathbb{S}'$，因为优良的切分集可以保障统计推断过程及结论的可靠性。

式（2.8）和式（2.9）的区别在于两者的协方差项。因此，深入分析协方差函数 $\mathrm{Cov}_{D_n}[\hat{\mu}_{HO}(\mathcal{S}_j),\hat{\mu}_{HO}(\mathcal{S}_{j'})|\phi_{jj'}=x]$ 与重叠样本个数 $\phi_{jj'}$ 之间的关系是非常必要的。定理2.1给出了它们的关系。

引理 2.1　记 $e_j(\mathcal{S}_i)=L(\mathcal{A}(D_i^{(t)}),z_j)$ 为在测试样例 z_j 上的损失函数。记 $\mathcal{S}_1=(I_1^{(t)},I_1^{(v)})$ 和 $\mathcal{S}_2=(I_2^{(t)},I_2^{(v)})$ 为数据集 D_n 上的两个随机切分。记 $\phi=|I_1^{(t)}\cap I_2^{(t)}|$ 为 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 间的重叠样本个数。则有

(1) 对于 $i,j\in\{1,\ldots,n_2\}$，$\mathrm{Cov}[e_i(\mathcal{S}_1),e_j(\mathcal{S}_2)|\phi=x]$ 有如下形式：

$$\mathrm{Cov}[e_i(\mathcal{S}_1),e_j(\mathcal{S}_2)|\phi=x]=\begin{cases}\sigma^2 & i=j,\ i,j\in D\\ \omega & i\neq j,\ i,j\in D\\ \gamma & i\in C\text{ 且 }j\in B\\ \tau & \text{其它}\end{cases}$$

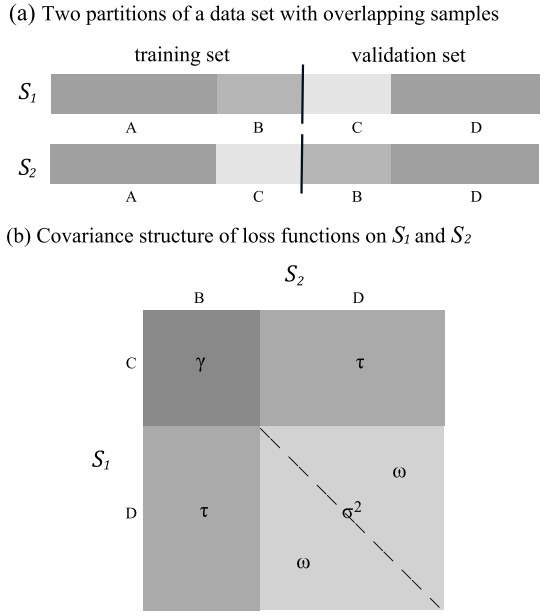

图 2.1 数据集的切分及 hold-out 估计间的协方差结构示例

其中，四个指标集 $A = I_1^{(t)} \cap I_2^{(t)}$，$B = I_2^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)})$，$C = I_1^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)})$ 及 $D = I_1^{(v)} \cap I_2^{(v)}$ 用来标识 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 间的重叠样本（见图2.1）。

(2) 定义 $f(x) \triangleq \mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_1), \hat{\mu}_{HO}(\mathcal{S}_2) | \phi = x]$，则 $f(x)$ 为关于 x 的二次多项式函数。其表达式如下：

$$f(x) = \frac{1}{n_2^2}[(\omega + \gamma - 2\tau)x^2 + \alpha_1 x + \alpha_0]。 \tag{2.10}$$

其中，$\alpha_1 = \sigma^2 - (2n_1 - 2n_2 + 1)\omega - 2n_1\gamma + 2(2n_1 - n_2)\tau$ 且 $\alpha_0 = (n_1 - n_2)[(n_1 - n_2 + 1)\omega + n_1^2\gamma - 2n_1\tau - \sigma^2]$。

因此，当 $\omega + \gamma > 2\tau$ 时，$f(x)$ 为关于 x 的一个下凸函数。

证明. 见第2.5.1节。 □

注记 2.1 设 $x_{min} = argmin_x f(x)$。则，下凸函数 $f(x)$ 在区间 $[x_{min}, n_1]$ 内单调递增。第2.4.1节中的模拟结果表明，当 n_1 趋近于 n 时，x_{min} 的取值趋近于 $n_1 - n_2$。

为阐述条件 $\omega + \gamma > 2\tau$ 的涵义，下面给出 σ^2，ω，γ 和 τ 的直观解释：

- ω 为定义在指标子集 $I_1^{(v)} \cap I_2^{(v)}$ 的两个损失函数的协方差函数。第一个损失函数的训练指标集为 $I_1^{(t)}$，测试样例为 z_i （$\forall i \in I_1^{(v)} \cap I_2^{(v)}$）。第二个损失函数使用定义在 $I_2^{(t)}$ 上的训练集，测试样例为 z_j （$\forall j \in I_1^{(v)} \cap I_2^{(v)}$ 且 $i \neq j$）。显然，测试样例 z_i 和 z_j 相互独立，且并不出现在彼此的训练集中。因此，ω 仅度量两个训练集所引起的相关性。另外，$i = j$ 对应参数 σ^2。

- 协方差 τ 中的两个损失函数分别使用 $I_1^{(t)}$ 和 $I_2^{(t)}$ 作为训练集。第一个损失函数的测试样例为 z_i （$\forall i \in I_1^{(v)} \cap I_2^{(v)}$），第二个损失函数的测试样例为 z_j（$\forall j \in I_1^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})$）或 z_j （$\forall j \in I_2^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})$）。显然，第二个损失函数的测试样例 z_j 出现在训练集 $I_1^{(t)}$ （或 $I_2^{(t)}$）中。因此，τ 不仅度量因两个训练集所引起的相关性，而且度量因测试样例 z_j 在训练集 $I_1^{(t)}$ （或 $I_2^{(t)}$）出现所引起的相关性。因此，τ 应大于 ω。
- 协方差 γ 所涉及的两个损失函数的测试样例为 z_i （$\forall i \in I_2^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})$）和 z_j （$\forall j \in I_1^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})$）。第一个损失函数的训练集为 $I_1^{(t)}$ ，测试样例为 z_i （$\forall i \in I_2^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})$）。第二个损失函数的训练集为 $I_2^{(t)}$，测试样例为 z_j（$\forall j \in I_1^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})$）。可见，测试样例 z_i 和 z_j 均出现在对方的训练集中。因此，γ 不仅度量两个训练集所引起的相关性，而且度量由两个测试样例在彼此训练集中出现所引起的相关性。因此，γ 大于 τ。即：$\gamma > \tau > \omega$。

条件 $\omega + \gamma - 2\tau$ 可看作 $\gamma - \tau$ 和 $\tau - \omega$ 两个差值的大小。即：当测试样例在训练集中逐步增多时，协方差的增量的变化情况。具体地，$\tau - \omega$ 对应的协方差值的增量由测试样例在训练集中出现的个数由零个变成一个时所引起。$\gamma - \tau$ 对应的协方差增量由测试样例的出现个数由一个变成二个时所引起。因此，条件 $\gamma + \omega > 2\tau$ 可重写为 $\gamma - \tau > \tau - \omega$。也就是说，随着测试样例的出现个数的增加，从 τ 到 γ 的增量要大于从 ω 到 τ 的增量。

2.2　正则化 RLT 的形式化描述

根据引理2.1可知，$\mathrm{Var}[\hat{\mu}_{RLT}(\mathbb{S})]$ （见式2.8）的最小化需要矩阵 $\mathbf{\Phi}$ 中所有元素 $\phi_{jj'}$ 同时达到它们的最小值。然而，因 $\phi_{jj'}$ 间有相关性，该条件很难满足。实际上，当 $\mathbf{\Phi}$ 中的某个元素达到其最小值 $n_1 - n_2$ 时，其它元素可能会远离该最小值。当同时最小化 $\mathbf{\Phi}$ 中的所有非对角线元素时，所有 $\phi_{jj'}$ 可达到的理想的最小值为 $\binom{Jn_1/n}{2} n / \binom{J}{2}$ （见定理2.1）。即使很多情形下，$\phi_{jj'}$ 不能达到这个理想值，它们也不应离这个值太远。基于此直觉，对于正则化 RLT 方法及其切分集，本节给出如下定义。

定义 2.1　若大小为 J 的切分集 $\mathbb{S}$ 满足如下正则化条件：

$$\left| \phi_{jj'} - \frac{\binom{Jn_1/n}{2} n}{\binom{J}{2}} \right| \leq k, \tag{2.11}$$

则称 $\mathbb{S}$ 为**正则化 RLT 切分集**，并记为 $\mathbb{S}^b = \{\mathcal{S}_j^b : j = 1, \ldots, J\}$。其中，$k$ 为正则化参数。基于 $\mathbb{S}^b$ 的 RLT 被称为**正则化 RLT**。$\mathbb{S}^b$ 对应的 $\mathbf{\Phi}$ 记为 $\mathbf{\Phi}^b$。

下面，给出与正则化 RLT 有关的一些记号。

- 记 $\hat{\mu}_{_{RLT}}(\mathbb{S}^b)$ 为泛化误差的**正则化 RLT 估计**。
- 当 $k=0$ 时[1]，将相应的切分集 $\mathbb{S}^b$ 记为 $\mathbb{S}^*=\{\mathcal{S}_j^*: j=1,\ldots,J\}$，称其为**均衡 RLT 切分集**。进而，记 $\mathbb{S}^*$ 对应的 $\boldsymbol{\Phi}$ 为 $\boldsymbol{\Phi}^*$。
- 记 $\hat{\mu}_{_{RLT}}(\mathbb{S}^*)$ 为泛化误差的**均衡 RLT 估计**。

正则化 RLT 及均衡 RLT 的优良性质在定理2.1中给出。

定理 2.1 当 $x_{min}\leq 2n\binom{Jn_1/n}{2}/(J(J-1))$ 时，在给定度量 $\boldsymbol{\Phi}$ 的条件下，$\exists k>0$，RLT 估计、正则化 RLT 估计和均衡 RLT 估计的方差有如下关系。

$$E_{\boldsymbol{\Phi}}[\mathrm{Var}[\mu_{_{RLT}}(\mathbb{S})|\boldsymbol{\Phi}]]\geq E_{\boldsymbol{\Phi}^b}[\mathrm{Var}[\mu_{_{RLT}}(\mathbb{S}^b)|\boldsymbol{\Phi}^b]]\geq E_{\boldsymbol{\Phi}^*}[\mathrm{Var}[\mu_{_{RLT}}(\mathbb{S}^*)|\boldsymbol{\Phi}^*]]。\tag{2.12}$$

证明. 因常数 $\sigma_1^2=\mathrm{Var}[\hat{\mu}_{_{HO}}(\mathcal{S})]$ 与方差最小化的过程无关，因此，可将最小化 $\mathrm{Var}[\hat{\mu}_{_{RLT}}(\mathbb{S})]$ 的问题转化为

$$\mathbb{S}^*=\arg min_{\mathbb{S}}\sum_{j=1}^{J}\sum_{j'=j+1}^{J}\mathrm{Cov}[\hat{\mu}_{_{HO}}(\mathcal{S}_j),\hat{\mu}_{_{HO}}(\mathcal{S}_{j'})|\phi_{jj'}]。\tag{2.13}$$

基于条件 $x_{min}\leq 2n\binom{Jn_1/n}{2}/(J(J-1))$，则式（2.13）有如下的最小化过程。

$$\begin{aligned}
&\sum_{j=1}^{J}\sum_{j'=j+1}^{J}\mathrm{Cov}\left[\hat{\mu}_{_{HO}}(\mathcal{S}_j),\hat{\mu}_{_{HO}}(\mathcal{S}_{j'})|\phi_{jj'}=x_{jj'}\right]\\
\geq\ &\binom{J}{2}\mathrm{Cov}\left[\hat{\mu}_{_{HO}}(\mathcal{S}_j),\hat{\mu}_{_{HO}}(\mathcal{S}_{j'})\middle|\phi_{jj'}=\frac{1}{\binom{J}{2}}\sum_{j=1}^{J}\sum_{j'=j+1}^{J}x_{jj'}\right]\\
=\ &\binom{J}{2}\mathrm{Cov}\left[\hat{\mu}_{_{HO}}(\mathcal{S}_j),\hat{\mu}_{_{HO}}(\mathcal{S}_{j'})\middle|\phi_{jj'}=\frac{1}{\binom{J}{2}}\sum_{i=1}^{n}\binom{\psi_i}{2}\right]\\
\geq\ &\binom{J}{2}\mathrm{Cov}\left[\hat{\mu}_{_{HO}}(\mathcal{S}_j),\hat{\mu}_{_{HO}}(\mathcal{S}_{j'})\middle|\phi_{jj'}=\frac{n}{\binom{J}{2}}\binom{Jn_1/n}{2}\right],
\end{aligned}\tag{2.14}$$

其中，$x_{jj'}$ 为 $\phi_{jj'}$ 在切分集 $\mathbb{S}$ 中的具体取值。

在式（2.14）中，因为 $f(x)$ 为关于 x 的下凸函数，则首先在 $f(x)$ 上使用了詹森不等式，然后基于式（2.2），将协方差转换为关于 ψ 的函数。接着，因为函数 $\binom{\psi}{2}$ 为关于 ψ 的下凸函数，再次使用詹森不等式。最后，基于 $f(x)$ 在 $[x_{min},n_1]$ 上的单调递增性质，则可得到式（2.14）所给的最小值。

显然，$\forall j\neq j'$，当 $\phi_{jj'}=\frac{n}{\binom{J}{2}}\binom{Jn_1/n}{2}$ 成立时，式（2.14）中所有等号成立。此时，正则化参数 $k=0$。因此，基于 $\mathbb{S}^*$ 的均衡 RLT 估计具有最小的方差。另外，

[1]当 n 可以整除 Jn_1 且 $\binom{J}{2}$ 可整除 $\binom{Jn_1/n}{2}n$ 时，有正则化参数 $k=0$。

因协方差函数 $f(x)$ 为关于正则化参数 $k \geq 0$ 的一个连续函数，故存在一个 k，使相应的 BRLT 估计满足：

$$E_{\boldsymbol{\Phi}}[\mathrm{Var}[\mu_{_{RLT}}(\mathbb{S})|\boldsymbol{\Phi}]] \geq E_{\boldsymbol{\Phi}^b}[\mathrm{Var}[\mu_{_{RLT}}(\mathbb{S}^b)|\boldsymbol{\Phi}^b]] \geq E_{\boldsymbol{\Phi}^*}[\mathrm{Var}[\mu_{_{RLT}}(\mathbb{S}^*)|\boldsymbol{\Phi}^*]]。$$

□

推论 2.1 对于任意两个均衡切分集 $\mathbb{S}_1^*$ 和 $\mathbb{S}_2^*$，有 $\mathrm{Var}[\hat{\mu}_{_{RLT}}(\mathbb{S}_1^*)|\Phi^*] = \mathrm{Var}[\hat{\mu}_{_{RLT}}(\mathbb{S}_2^*)|\Phi^*]$。

推论 2.2 K 折交叉验证为均衡 RLT 的一种特殊情况。此时，切分集大小为 K，训练集大小为 $n_1 = (K-1)n/K$。相应的正则化条件为 $|\phi_{jj'} - (K-2)n/K| = 0$。

推论 2.3 均衡 RLT 的切分集 $\mathbb{S}^*$ 有两条优良性质：（1）D_n 中每条样例在 J 个训练集中出现频次均相同；（2）任意两个训练集间重叠样本个数均相同。也就是说，$\mathbb{S}^*$ 满足：

(1) $\forall i, i', j, j' = 1, \ldots, J$，$|\phi_{ii'} - \phi_{jj'}| = 0$。

(2) $\forall i, j = 1, \ldots, n$，$|\psi_i - \psi_j| = 0$。

记正则化 RLT 上的出现频次向量 $\boldsymbol{\Psi}$ 为 $\boldsymbol{\Psi}^b$。根据式（2.11）中的正则化条件，可得 $\boldsymbol{\Phi}^b$ 和 $\boldsymbol{\Psi}^b$ 的具体取值（见附录2.5.2）。即便如此，直接构造正则化 RLT 的切分集仍然非常困难。具体地，RLT 的每一个切分集 $\mathbb{S}$ 可映射为一个 $0-1$ 矩阵 $\boldsymbol{P} = (p_{ij})_{n\times J}$。矩阵 $\boldsymbol{P}$ 中的第 i 行和第 j 列分别对应样例 z_i 和切分 $\mathcal{S}_j$。在矩阵 $\boldsymbol{P}$ 中，元素 $p_{ij} = 1$ 表示样例 z_i 在训练集 $D_j^{(t)}$ 中出现，反之亦然。矩阵 $\boldsymbol{P}$ 被称为切分集 $\mathbb{S}$ 的**映射阵**。另外，设 $\mathbf{1}_J$ 为长度为 J 的全 1 向量。映射阵 $\boldsymbol{P}^b$ 应满足 $\boldsymbol{P}^{b\top}\boldsymbol{P}^b = \boldsymbol{\Phi}^b$ 且 $\boldsymbol{P}^b\mathbf{1}_J = \boldsymbol{\Psi}^b$。然而，对于任意的 n_1 和 J，不存在 $\boldsymbol{P}^b$ 和 $\mathbb{S}^b$ 的通用快速构造方法。

不过，对于 n_1 和 J 的一些特定情形，可基于正交表来构造正则化 RLT 的切分集 $\mathbb{S}^b$。相应的构造算法在下述几小节中给出。对于任意的 n_1 和 J，正则化 RLT 的构造算法仍然是一个公开问题，具体在第2.2.5 节中讨论。

2.2.1 情形 1：$n_1 \geq (J-1)n/J$

此情形下，可将整个数据集 D_n 切分为至少 J 块，至多 $J+1$ 块，前 J 块中的每一块有 n_2 条样例。轮流将前 J 块中的每一块作为验证集，将 D_n 中的其余样本用作训练集，则可构造出相应的正则化 RLT 的切分集，且相应的正则化参数为 $k = 0$。当 $J = K$ 且 $n = n_2 J$ 时，相应的切分集对应于 K 折交叉验证。此时，$n_1 = (J-1)n/J$。

2.2.2 情形 2：$n_1 = n/2$ 且 J 为偶数

$n_1 = n/2$ 时的 RLT 称为 Repeated Half-Sampling（简记为 RHS）[77]，正则化 RLT 可类似地命名为正则化 RHS（简记为 BRHS）。正则化 RHS 的切分集 $\mathbb{S}^b$ 可基于二水平正交表 $OA(2J, 2J-1)$ 进行构造[2]。具体见算法2.1。该算法可使用两种不同的二水平正交表。第一种正交表中，试验数（$2J$）为 2 的幂数。相应的正交表被记为 $OA(2^p, 2^p-1)$ （其中，$p \geq 2$ 为因子数），对应于部分因子设计。另一种正交表中，试验数为 4 的倍数但非 2 的幂数。此时，正交表对应一个 Plackett-Burman 设计。这两种二水平正交表中，试验数 $2J$ 均为 4 的倍数。因此，可知 J 为偶数。

算法 2.1 正则化 RHS 的切分集的构造算法

输入： 数据集大小，n；切分集大小，J；

输出： 正则化 RHS 的切分集，$\mathbb{S}^b$。

1: 从常用正交表中查到 $OA(2J, 2J-1)$，其中，两个水平分别记为‘+’和‘-’；
2: $OA_1(2J-1, 2J-1) \leftarrow$ 删除 $OA(2J, 2J-1)$ 中全部水平均相同的一行；
3: $lv \leftarrow$ 取出 $OA_1(2J-1, 2J-1)$ 第一行中出现次数为 $J-1$ 的水平；
4: $OA_2(2J-2, J) \leftarrow$ 删除 $OA_1(2J-1, 2J-1)$ 中第一个水平为 lv 的列，并移除 $OA_1(2J-1, 2J-1)$ 中的第一行；
5: 将指标集 $\mathcal{I} = \{1, \ldots, n\}$ 等分为 $2J-2$ 个子块，记为 $B^{(1)}, \ldots, B^{(2J-2)}$；
6: 初始化 $\mathbb{S}^b \leftarrow \{\mathcal{S}_j : j = 1, \ldots, J\}$；
7: **for all** $j = 1; j \leq J; j++$ **do**
8: 初始化 $\mathcal{S}_j = (I_j^{(t)}, I_j^{(v)})$，其中：$I_j^{(t)} \leftarrow \emptyset$ 且 $I_j^{(v)} \leftarrow \emptyset$；
9: **for all** $i = 1; i \leq 2J-2; i++$ **do**
10: $e_{ij} \leftarrow OA_2(2J-2, J)$ 中第 ij 个元素；
11: **if** e_{ij} 为‘+’ **then**
12: $I_j^{(t)} \leftarrow I_j^{(t)} \cup B^{(i)}$
13: **else**
14: $I_j^{(v)} \leftarrow I_j^{(v)} \cup B^{(i)}$
15: **end if**
16: **end for**
17: **end for**
18: **return** $\mathbb{S}^b$

算法2.1在构造切分集 $\mathbb{S}^b$ 时，使用了数据集 D_n 的 $2J-2$ 个数据子块（见第5步），而不是直接使用 n 条样例。因此，该算法适用于大小不同的样本集。下例中给出了 $J = 6$ 时算法2.1的执行过程。随后，给出该算法的正确性证明。

例 2.1 当 $J = 6$ 时，应采用正交表 $OA(12, 11)$，见表2.1。

$OA(12, 11)$ 中第一行的水平相同，因此移除其第一行，得到 $OA_1(11, 11)$，见

[2] 常用的二水平正交表已在 Wu 等的著作 [38] 中给出。本节仅认为所用二水平正交表可直接得到，并不讨论正交表的构造方法。

表 2.1　正交表 $OA(12,11)$

	1	2	3	4	5	6	7	8	9	10	11
1	+	+	+	+	+	+	+	+	+	+	+
2	+	+	+	+	-	-	-	+	-	-	-
3	+	+	-	-	-	+	-	-	+	-	+
4	+	-	+	-	+	+	+	-	-	-	-
5	+	-	-	+	-	-	+	-	+	+	-
6	+	-	-	-	+	-	-	+	-	+	+
7	-	+	+	-	-	-	+	-	-	+	+
8	-	+	-	+	+	+	-	-	-	+	-
9	-	+	-	-	+	-	+	+	+	-	-
10	-	-	+	+	+	-	-	-	+	-	+
11	-	-	+	-	-	+	-	+	+	+	-
12	-	-	-	+	-	+	+	+	-	-	+

表2.2。子表 $OA_1(11,11)$ 的第一行中，水平“+”出现 5 次且水平“-”出现 6 次。因此，移除掉第一行中水平为“+”的列，然后移除第一行，得到子表 $OA_2(10,6)$，见表2.3。

表 2.2　子表 $OA_1(11,11)$.

	1	2	3	4	5	6	7	8	9	10	11
1	+	+	+	+	-	-	-	+	-	-	-
2	+	+	-	-	-	+	-	-	+	-	+
3	+	-	+	-	+	+	+	-	-	-	-
4	+	-	-	+	-	-	+	-	+	+	-
5	+	-	-	-	+	-	-	+	-	+	+
6	-	+	+	-	-	-	+	-	-	+	+
7	-	+	-	+	+	+	-	-	-	+	-
8	-	+	-	-	+	-	+	+	+	-	-
9	-	-	+	+	+	-	-	-	+	-	+
10	-	-	+	-	-	+	-	+	+	+	-
11	-	-	-	+	-	+	+	+	-	-	+

当 $J=6$ 时，$2J-2=10$，因此将 $I=\{1,2,\ldots,n\}$ 分为 10 等分，并记为 $B^{(1)},\ldots,B^{(10)}$，且分别对应子表 $OA_2(10,6)$ 中的 10 行。根据 $OA_2(10,6)$ 中的列，多次合并 10 个子块，则可得到正则化 RHS 的切分集 $\mathbb{S}^b$，见表2.4。

从表2.4的切分集中可得，每一个数据子块在 6 个训练集中分别出现三次，任意

表 2.3 子表 $OA_2(10,6)$

	1	2	3	4	5	6
$1\leftrightarrow B^{(1)}$	-	+	-	+	-	+
$2\leftrightarrow B^{(2)}$	+	+	+	-	-	-
$3\leftrightarrow B^{(3)}$	-	-	+	+	+	-
$4\leftrightarrow B^{(4)}$	+	-	-	-	+	+
$5\leftrightarrow B^{(5)}$	-	-	+	-	+	+
$6\leftrightarrow B^{(6)}$	+	+	-	-	+	-
$7\leftrightarrow B^{(7)}$	+	-	+	+	-	-
$8\leftrightarrow B^{(8)}$	+	-	-	+	-	+
$9\leftrightarrow B^{(9)}$	-	+	-	+	+	-
$10\leftrightarrow B^{(10)}$	-	+	+	-	-	+

表 2.4 $J=6$ 时的正则化 RHS 的切分集合 $\mathbb{S}^b$.

	$I_j^{(t)}$	$I_j^{(v)}$
$\mathcal{S}_1^b$	$B^{(2)},B^{(4)},B^{(6)},B^{(7)},B^{(8)}$	$B^{(1)},B^{(3)},B^{(5)},B^{(9)},B^{(10)}$
$\mathcal{S}_2^b$	$B^{(1)},B^{(2)},B^{(6)},B^{(9)},B^{(10)}$	$B^{(3)},B^{(4)},B^{(5)},B^{(7)},B^{(8)}$
$\mathcal{S}_3^b$	$B^{(2)},B^{(3)},B^{(5)},B^{(7)},B^{(10)}$	$B^{(1)},B^{(4)},B^{(6)},B^{(8)},B^{(9)}$
$\mathcal{S}_4^b$	$B^{(1)},B^{(3)},B^{(7)},B^{(8)},B^{(9)}$	$B^{(2)},B^{(4)},B^{(5)},B^{(6)},B^{(10)}$
$\mathcal{S}_5^b$	$B^{(3)},B^{(4)},B^{(5)},B^{(6)},B^{(9)}$	$B^{(1)},B^{(2)},B^{(7)},B^{(8)},B^{(10)}$
$\mathcal{S}_6^b$	$B^{(1)},B^{(4)},B^{(5)},B^{(8)},B^{(10)}$	$B^{(2)},B^{(3)},B^{(6)},B^{(7)},B^{(9)}$

两个训练集重叠两个数据子块。因此，表2.4 的切分集为 $J=6$ 时正则化 RHS 的切分集。算法2.1 正确性的理论证明在定理2.2 中给出。

定理 2.2 假定数据集 D_n 被等分[3] 为 $2J-2$ 个数据子块，给定 $OA(2J,2J-1)$，则基于算法2.1构造的切分集 $\mathbb{S}$ 为正则化切分集 $\mathbb{S}^b$，且相应的正则化参数 $k\leq J/2$。

证明. 见第2.5.3节。 □

除 RHS 和正则化 RHS 外，2 折交叉验证[78]、$m\times 2$ 交叉验证[1,6,7] 及正则化 $m\times 2$ 交叉验证（见第三 章）也是 $n_1=n/2$ 时的交叉验证方法。尽管基于这些交叉验证的泛化误差估计均为 $\mu(n/2)$ 的无偏估计，但正则化 RHS 估计比其余交叉验证估计有更小的方差。具体见定理2.3。

定理 2.3 记 $\hat{\mu}_{RHS}(\mathbb{S})$、$\hat{\mu}_{m\times 2}(\mathbb{S}_{m\times 2})$、$\hat{\mu}_{m\times 2}(\mathbb{S}^b_{m\times 2})$ 和 $\hat{\mu}_{RHS}(\mathbb{S}^b)$ 分别为泛化误差的 RHS 估计、$m\times 2$ 交叉验证估计、正则化 $m\times 2$ 交叉验证估计和正则化 RHS 估

[3] “等分”指数据子块大小之间最多不超过 1。

计。其中，RHS 估计和正则化 RHS 估计所用切分集[4] 的大小均为 J。当 $J = 2m$ 时，可得

$$\begin{aligned} E_{\boldsymbol{\Phi}}[\mathrm{Var}[\hat{\mu}_{RHS}(\mathbb{S})|\boldsymbol{\Phi}]] &\geq E_{\boldsymbol{\Phi}_{m\times 2}}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}_{m\times 2})|\boldsymbol{\Phi}_{m\times 2}]] \\ &\geq E_{\boldsymbol{\Phi}^b_{m\times 2}}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b_{m\times 2})|\boldsymbol{\Phi}^b_{m\times 2}]] \\ &\geq E_{\boldsymbol{\Phi}^b}[\mathrm{Var}[\hat{\mu}_{RHS}(\mathbb{S}^b)|\boldsymbol{\Phi}^b]]。 \end{aligned} \tag{2.15}$$

证明. 见第2.5.4节。 □

注记 2.2　定理2.3表明，当 $n_1 = n/2$ 且给定 J 时，泛化误差的正则化 RHS 估计具有比 RHS 估计、$m \times 2$ 交叉验证估计和正则化 $m \times 2$ 交叉验证估计更小的方差。

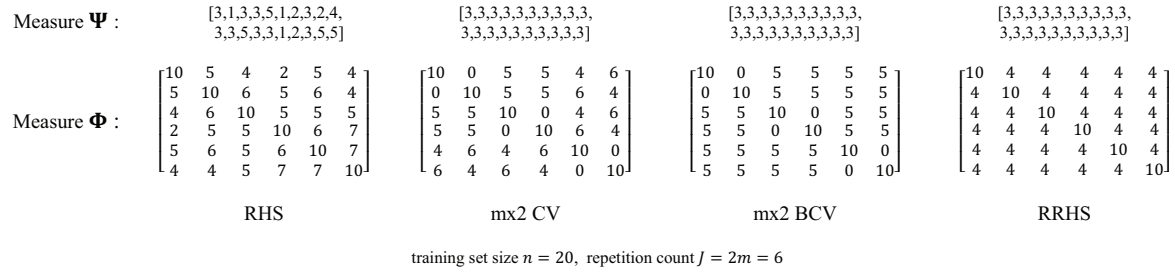

图 2.2　4 种交叉验证方法的度量 $\boldsymbol{\Psi}$ 和 $\boldsymbol{\Phi}$ 的比较

本质上，RHS，$m \times 2$ 交叉验证，正则化 $m \times 2$ 交叉验证和正则化 RHS 四种交叉验证方法的 $\boldsymbol{\Psi}$ 和 $\boldsymbol{\Phi}$ 具有不同的分布。图2.2 给出了当 $n = 20$ 且 $J = 2m = 6$ 时，$\boldsymbol{\Psi}$ 和 $\boldsymbol{\Phi}$ 的分布。在图2.2 中，RHS 的 $\boldsymbol{\Psi}$ 和 $\boldsymbol{\Phi}$ 的值及 $m \times 2$ 交叉验证的 $\boldsymbol{\Phi}$ 为随机给出的一组可能值。其余的值没有随机性。$m \times 2$ 交叉验证、正则化 $m \times 2$ 交叉验证和正则化 RHS 三种方法的 $\boldsymbol{\Psi}$ 值均相同。因此，这三种交叉验证均满足推论2.3 中的条件 $|\psi_i - \psi_j| = 0$。但是，RHS 的 $\boldsymbol{\Psi}$ 并不满足这个条件。这可能是 RHS 估计具有较大方差的主要原因。对比四种交叉验证方法的 $\boldsymbol{\Phi}$ 值，可见其随着 RHS，$m \times 2$ 交叉验证，正则化 $m \times 2$ 交叉验证和正则化 RHS 的顺序变得逐渐严格。对此，给出如下的具体分析。

- 对于 RHS，矩阵 $\boldsymbol{\Phi}$ 的每一个非对角线元素均为随机变量，且服从如式（2.7）所示的超几何分布。当正则化参数 $k = n/2$ 时，$\boldsymbol{\Phi}$ 才基本满足正则化条件。
- 对于 $m \times 2$ 交叉验证，$(\mathcal{S}_1, \mathcal{S}_2)$，$(\mathcal{S}_3, \mathcal{S}_4)$ 和 $(\mathcal{S}_5, \mathcal{S}_6)$ 三对切分对应三组 2 折交叉验证切分。因此，矩阵 $\boldsymbol{\Phi}$ 中元素 ϕ_{12}、ϕ_{21}、ϕ_{34}、ϕ_{43}、ϕ_{56} 和 ϕ_{65} 均为零。$\boldsymbol{\Phi}$

[4]为与 RHS 与正则化 RHS 的切分集区分，$m \times 2$ 交叉验证和正则化 $m \times 2$ 交叉验证的切分集分别记为 $\mathbb{S}_{m\times 2}$ 和 $\mathbb{S}^b_{m\times 2}$。其中，切分集 $\mathbb{S}^b_{m\times 2}$ 满足正则化条件 $\phi_{ij} - n/4 \leq k$。相应的矩阵 $\boldsymbol{\Phi}$ 分别记为 $\boldsymbol{\Phi}_{m\times 2}$ 和 $\boldsymbol{\Phi}^b_{m\times 2}$。

中其余元素为两个 2 折交叉验证的训练集间重叠样本个数，且其服从式（2.7）所示的超几何分布，相应的正则化参数为 $k = n/2$。

- 对于正则化 $m \times 2$ 交叉验证，$(\mathcal{S}_1, \mathcal{S}_2)$，$(\mathcal{S}_3, \mathcal{S}_4)$ 和 $(\mathcal{S}_5, \mathcal{S}_6)$ 三对切分仍对应三组 2 折交叉验证切分。其余的元素满足正则化条件 $|\phi_{ij} - n/4| \leq k$ 且在图2.2中 $k = 0$。然而，该矩阵并不能很好满足式（2.11）中所示条件，因为 $\mathbf{\Phi}$ 中存在零元素。
- 正则化 RHS 将正则化 $m \times 2$ 交叉验证中 2 折交叉验证的约束破坏掉。因此，矩阵 $\mathbf{\Phi}$ 中的非对角线元素满足正则化条件（式（2.11）），且 $k = 0$。因此，正则化 RHS 估计具有最小的方差。

不过，算法2.1仍有如下几个缺点。

(1) 当切分次数 J 为奇数时，正则化 RLT 切分集不能用算法2.1 来构造；

(2) 当正交表 $OA(2J, 2J-1)$ 不存在时，正则化切分集不能用算法2.1 来构造；

(3) 基于算法2.1，正则化 RHS 的切分集不能增量式构造。即，当 J 增加时，切分集需要重新构造；

综上，当 $n_1 = n/2$ 时，本文建议当切分次数 J 固定且 $OA(2J, 2J-1)$ 存在时，应优先选择正则化 RHS 方法；否则，可选择正则化 $m \times 2$ 交叉验证方法。

2.2.3 情形 3：$n_1 = (J+1)n/(2J)$ 且 J 为奇数

在本情形下，切分集的构造在算法2.2 中给出。该算法利用了正则化 RHS 构造算法在第4步移除掉的所有的正交表列。下面，以 $J = 5$ 且 $n_1 = 3n/5$ 为例说明算法执行产生的中间结果。算法所用的 $OA(12, 11)$ 和 $OA_1(11, 11)$ 如表2.1 和表2.2 所示。表2.5给出 $OA_3(11, 11)$。表2.6 给出了相应的切分集。

表 2.5 子表 $OA_3(10, 5)$

	1	2	3	4	5
$1 \leftrightarrow B^{(1)}$	+	+	-	-	-
$2 \leftrightarrow B^{(2)}$	+	-	+	-	-
$3 \leftrightarrow B^{(3)}$	+	-	-	+	-
$4 \leftrightarrow B^{(4)}$	+	-	-	-	+
$5 \leftrightarrow B^{(5)}$	-	+	+	-	-
$6 \leftrightarrow B^{(6)}$	-	+	-	+	-
$7 \leftrightarrow B^{(7)}$	-	+	-	-	+
$8 \leftrightarrow B^{(8)}$	-	-	+	+	-
$9 \leftrightarrow B^{(9)}$	-	-	+	-	+
$10 \leftrightarrow B^{(10)}$	-	-	-	+	+

算法 2.2 $n_1=(J+1)n/(2J)$ 且 J 为奇数时正则化 RLT 切分集的构造算法

输入: 数据集大小，n；切分集大小，J；

输出: 正则化 RLT 的切分集，$\mathbb{S}^b$。

从常用正交表中查到 $OA(2J+2,2J+1)$，其中，两个水平分别记为‘+’和‘-’；
$OA_1(2J+1,2J+1)\leftarrow$ 删除 $OA(2J+2,2J+1)$ 中全部水平均相同的一行；
$lv\leftarrow$ 取出 $OA_1(2J+1,2J+1)$ 第一行中出现次数为 $J+1$ 的水平；
$OA_2(2J,J)\leftarrow$ 删除 $OA_1(2J+1,2J+1)$ 中第一个水平为 lv 的列，并移除 $OA_1(2J+1,2J+1)$ 中的第一行；
将指标集 $\mathcal{I}=\{1,\ldots,n\}$ 等分为 $2J$ 个子块，记为 $B^{(1)},\ldots,B^{(2J)}$；
初始化 $\mathbb{S}^b\leftarrow\{\mathcal{S}_j:j=1,\ldots,J\}$；
for all $j=1;j\leq J;j++$ **do**
  初始化 $\mathcal{S}_j=(I_j^{(t)},I_j^{(v)})$，其中：$I_j^{(t)}\leftarrow\emptyset$ 且 $I_j^{(v)}\leftarrow\emptyset$；
  for all $i=1;i\leq 2J;i++$ **do**
    $e_{ij}\leftarrow OA_2(2J,J)$ 中第 ij 个元素；
    if e_{ij} 为‘+’ **then**
      $I_j^{(t)}\leftarrow I_j^{(t)}\cup B^{(i)}$
    else
      $I_j^{(v)}\leftarrow I_j^{(v)}\cup B^{(i)}$
    end if
  end for
end for
return $\mathbb{S}^b$

表 2.6　$n_1=3n/5$ 且 $J=5$ 时正则化 RLT 的切分集 $\mathbb{S}^b$

	$I_j^{(t)}$	$I_j^{(v)}$
$\mathcal{S}_1^b$	$B^{(5)},B^{(6)},B^{(7)},B^{(8)},B^{(9)},B^{(10)}$	$B^{(1)},B^{(2)},B^{(3)},B^{(4)}$
$\mathcal{S}_2^b$	$B^{(2)},B^{(3)},B^{(4)},B^{(8)},B^{(9)},B^{(10)}$	$B^{(1)},B^{(5)},B^{(6)},B^{(7)}$
$\mathcal{S}_3^b$	$B^{(1)},B^{(3)},B^{(4)},B^{(6)},B^{(7)},B^{(10)}$	$B^{(2)},B^{(5)},B^{(8)},B^{(9)}$
$\mathcal{S}_4^b$	$B^{(1)},B^{(2)},B^{(4)},B^{(5)},B^{(7)},B^{(9)}$	$B^{(3)},B^{(6)},B^{(8)},B^{(10)}$
$\mathcal{S}_5^b$	$B^{(1)},B^{(2)},B^{(3)},B^{(5)},B^{(6)},B^{(8)}$	$B^{(4)},B^{(7)},B^{(9)},B^{(10)}$

从表2.6中可见，每一个数据子块 $B^{(i)}$ 在 5 个训练集中出现 3 次，且任意 2 个训练集间重叠 3 个数据子块。因此，表2.6 所给切分集为正则化 RLT 的正则化切分集。基于算法2.2，所构造的正则化切分满足 $\phi_{ij}=(J+1)|B^{(i)}|/2$ 且 $k\leq(J+1)/2$。

2.2.4　正则化 RLT 切分集构造算法的时间复杂度分析

在分析 RLT 和正则化 RLT 切分集构造算法的时间复杂度之前，先给出构造算法所包含的基本操作及其时间复杂度。

- **打乱**：将大小为 n 的数据集中所有样例打乱为随机顺序。基于 Fisher−Yates 算法[79]，“打乱”操作的最优时间复杂度可以达到 $O(n)$。
- **切分**：将数据集被切分为 k 个数据子块。这些子块的大小可以不同。“切分”操作的时间复杂度为 $O(k)$。
- **合并**：k 个数据子块合并为一个数据块。这个操作为“切分”操作的逆操作，其时间复杂度为 $O(k)$。
- **查找一个正交表并切分该表**：根据切分次数 J，查找一个合适的正交表，并基于算法2.1和算法2.2 中第 2 步到第 4 步的启发式规则将正交表切分为子表。由于常用的正交表已在 Wu 等的工作[38] 中给出，可事先收集常用的二水平正交表，并提前将它们切分为子表，然后将这些子表存储在数据库中备用。只需执行一次本操作，并共享该子表库。因此，可将本操作的时间复杂度看作 $O(1)$。
- **从一个正交表中取出一列**：通过合理地存储正交表，可将此操作的时间复杂度降低到 $O(1)$。
- **遍历正交表中一列中所有的元素**：假定正交表的一列中有 k 个元素，遍历这些元素需要的时间复杂度为 $O(k)$。

假定数据集 D_n 上，RLT 的训练集大小为 n_1、切分次数为 J。RLT 的切分集的构造分三步：（1）在 n 条样例上应用随机打乱算法；（2）将 n 条样例切分为大小为 n_1 和 $n-n_1$ 两个数据子块；（3）将第（1）步和第（2）步重复 J 次。因此，该构造算法的时间复杂度为 $O(J(n+2))$。

下面，分析正则化 RLT 构造算法的时间复杂度。上几节中分别给出三种不同情形下的构造算法，这些构造算法的时间复杂度如下。

- 对于 $n_1 \geq (J-1)n/J$ （见第2.2.1 节），BRLT 切分集的构造分三步：（1）打乱数据集中的 n 条样例；（2）将数据集切成 J 个子块；（3）轮流合并 $J-1$ 个子块，并执行此步骤 J 次。因此，该构造的时间复杂度为 $O(n+J+J(J-1)) = O(n+J^2)$。
- 当 $n_1 = n/2$ 且 J 为偶数时（见第2.2.2 节），构造方法分六步：（1）根据切分次数 J 查找一个正交表，并对切分该表；（2）打乱数据集中的 n 条样例；（3）将 n 条样例等分成 $2J-2$ 个数据子块；（4）从子表中取出一列；（5）将 $J-1$ 个数据子块合并成训练集，并将另外 $J-1$ 个数据子块合并成验证集；（6）将第（4）步和第（5）步重复 J 次。该构造的时间复杂度为 $O(1+n+2J-2+J(1+(J-1)+(J-1))) = O(n+J(2J+1)-1) \approx O(n+2J^2)$。
- 当 $n_1 = n/2$ 且 J 为奇数时（见第2.2.3 节），构造方法与第二种情形类似。此

时，构造算法的时间复杂度为：$O(1+n+2J+J(1+J+J)) = O(n+J(2J+3)+1) \approx O(n+2J^2)$。

总之，RLT 构造算法的时间复杂度为 $O(J(n+2))$。该时间复杂度与数据集大小和切分次数线性相关。正则化 RLT 构造算法的时间复杂度为 $O(n+2J^2)$。这个时间复杂度与数据集大小线性相关，为切分次数的二次函数。不过，当切分次数较小时，该时间复杂度仍是可容忍的。

2.2.5　关于正则化 RLT 通用构造算法的讨论

基于正则化 RLT 的定义，在约束重叠样本个数的前提下，正则化 RLT 的构造算法需要将样本的指标集合理地分配到切分集中。因此，该构造算法可自然转化为关于样本指标的优化设计问题。在该优化设计中，每条样例在所有训练集上的出现频次应相同，且每两个训练集间的重叠样本个数也应相同。因此，该最优设计应具有“均衡性”。受该均衡性的启发，本章采用正交表来构造正则化 RLT 的切分集合。正交表为实验设计领域（Design of Experiment，DoE）的一个成熟且有力的工具。因为数据集应被切分为训练集和验证集两部分，所以，正交表应具有两个水平，且该正交表可以被高效地构造[38]。此外，尽管正交表的历史已有数十年之久，但近年来，正交表仍被很多研究工作所关注[17,80–83]。例如，正交表可被用以改善拉丁超立方抽样方法（Latin hypercube sampling，LHS）[84]。该抽样方法改进了随机蒙特卡洛抽样法，且可进一步改良为基于正交表的 LHS 方法[82]。在文本数据的相关任务中，正交表可用于选择优良特征[17,85]。正交表也可用于改进因子设计中的因果框架[80,86]。这些工作可为正则化 RLT 后续的研究提供更有力的工具。

除了第2.2.1节到第2.2.3 节给出的设置外，另外一些 n_1 和 J 的设置下正则化 RLT 切分集仍可采用正交表来构造。例如，三水平正交表 $OA(18,7)$ 可用于构造设置为 $n_1 = 2n/3$ 且 $J = 6$ 时的正则化 RLT 的切分集，且相应的正则化 RLT 估计的方差小于 2×3 交叉验证估计的方差[5]。然而，基于正交表的构造算法的缺点在于给定 n 和 J 后，需要提前准备对应的正交表。

构造正则化 RLT 切分集的另一种方法是将切分集的构造转化为构造映射矩阵的数学问题。在构造该矩阵时，映射矩阵的行和、列和及乘积均已知。然而，很多研究表明，对于该映射矩阵，不存在一个通用且有效的构造算法[87,88]。是否存在高效的近似构造算法，是后续的一个研究问题。

式（2.11）所示的正则化条件给出了判别优良切分集的一个准则。所以，若不考虑算法的计算开销，构造 $\mathbb{S}^b$ 的一个重要的方法是：首先，随机生成多个 RLT 候

[5] 2×3 交叉验证指具有将 3 折交叉验证方法随机重复 2 次。

选切分集；然后，使用该准则从这些候选切分集中挑选出 $\mathbb{S}^b$。

2.3 实验数据及设置

本章主要关注正则化 RLT 方法在分类和回归任务中的优良性质。实验分别使用了多个模拟数据集合真实数据集。在这些数据集上，多个典型的机器学习算法被使用。模拟数据集的设置方式与第3.5.1部分的模拟数据设置相同。

除模拟数据外，实验中也使用了一些真实数据集。这些数据集来自于 UCI 数据集。表2.7给出了这些数据集的基本信息，具体包括样例个数，特征数及类别数。在这些数据集中，具有缺失值的样例被删除。在 Letter 数据集中，26 类被归结为两类，即：A 到 M 为一类，N 到 Z 为另一类。每一个 UCI 数据集被用作总体，从中有放回地抽取出 n 个样本，以构成数据集 D_n。表2.7 的最后一类给出 n 的取值。在这些数据集上，使用了一些常用的机器学习算法，例如支持向量机（svm）和 k 近邻（knn）。

表 2.7 实验用 UCI 数据集的基本信息

数据集	样例个数	特征数	类别数	n
glass	214	9	7	150
iris	150	4	3	100
krvskp	3196	36	2	200
letter	20000	16	26	1000
optdigits	5620	64	10	300
pageblock	5473	10	5	300
spambase	4601	57	2	250
wave	5000	40	3	300
wdbc	569	30	2	300
yeast	1484	10	8	800

本章仍采用“数据集标记 + 算法标记 + 数据集大小 + 特征个数”来表示每种实验配置。本章的实验包括两部分：协方差模拟实验和方差比较实验。在协方差模拟实验中，对于每一个重叠样本个数，从数据总体中生成 1000 个数据集及 1000 个切分。在方差对比实验中，从数据总体中随机生成 10000 个数据集，并对于每种交叉验证，随机生成 1000 个切分集。则，共产生 1000 万个泛化误差的估计，并使用它们的样本方差来近似方差的理论值。

为比较两种不同的交叉验证方法，引入两种估计的方差约减率（reduction rate,

RR）来度量两种不同的交叉验证估计的方差约减率。方差约减率的表达式如下：

$$RR_{cv_1/cv_2} = 1 - \frac{E_{\mathbf{\Phi}_{cv_2}}[\mathrm{Var}[\hat{\mu}_{cv_2}(\mathbb{S}_{cv_2})|\mathbf{\Phi}_{cv_2}]]}{E_{\mathbf{\Phi}_{cv_1}}[\mathrm{Var}[\hat{\mu}_{cv_1}(\mathbb{S}_{cv_1})|\mathbf{\Phi}_{cv_1}]]}。\tag{2.16}$$

其中，cv_1 和 cv_2 为两种不同的交叉验证方法。

2.4　实验结果及分析

本节基于模拟实验来回答如下几个研究问题：

研究问题一： 协方差函数 $f(x)$ 是否为关于重叠样本个数 $\phi = x$ 的单调下凸函数？

研究问题二： 当 $n_1 = (K-1)n/K$ 时，泛化误差的 RLT 估计的方差与 K 折交叉验证估计的方差之间的差有多大？

研究问题三： 泛化误差的 RHS 估计、$m \times 2$ 交叉验证估计、正则化 $m \times 2$ 交叉验证估计和正则化 RHS 估计间方差有多大差距？

2.4.1　研究问题一的模拟实验

表2.8 给出了 RLT 和正则化 RLT 上一些常用统计量的取值。这些统计量包括：正则化条件（式（2.19））中的 $\phi^*_\Sigma/\binom{J}{2}$，RLT 中 ϕ 的期望 $E[\phi]$ 和方差 $\mathrm{Var}[\phi]$，及 $E[|\phi - \phi^*_\Sigma/\binom{J}{2}|]$ 的值。其中，统计量 $E[|\phi - \phi^*_\Sigma/\binom{J}{2}|]$ 的值为 10000 次模拟值，其余统计量为理论值。

表 2.8　正则化 RLT 和 RLT 上一些统计量的比较

n	n_1	$\phi^*_\Sigma/\binom{J}{2}$					$E[\phi]$	$\mathrm{Var}[\phi]$	$E[\lvert\phi - \phi^*_\Sigma/\binom{J}{2}\rvert]$				
		$J=2$	$J=5$	$J=10$	$J=50$	$J=100$			$J=2$	$J=5$	$J=10$	$J=50$	$J=100$
100	50	0	20	22.22	24.49	24.75	25	6.31	24.94	4.98	3.06	2.04	2.00
100	70	40	45	46.67	48.57	48.79	49	4.45	9.00	4.03	2.61	1.72	1.68
100	90	80	80	80	80.82	80.91	81	0.82	1.00	1.00	1.00	0.72	0.69
500	250	0	100	111.11	122.45	123.74	125	31.31	124.91	24.91	13.83	4.88	4.54
500	300	100	150	166.67	177.55	178.79	180	28.86	80.06	30.06	13.42	4.75	4.40
500	400	300	300	311.11	318.38	319.19	320	12.83	20.06	20.06	8.96	3.15	2.92
500	450	400	400	400	404.08	404.55	405	4.06	5.00	5.00	5.00	1.71	1.63
1000	500	0	200	222.22	244.90	247.47	250	62.56	249.94	49.94	27.72	7.49	6.59
1000	600	200	300	333.33	355.10	357.58	360	57.66	160.13	60.13	26.80	7.32	6.38
1000	800	600	600	622.22	636.74	638.38	640	25.63	40.05	40.05	17.82	4.83	4.22
1000	900	800	800	800	808.16	809.09	810	8.11	10.05	10.05	10.05	2.70	2.36

显然，$\phi^*_\Sigma/\binom{J}{2}$ 总是比 $E[\phi]$ 值小。随着 J 的增加，$\phi^*_\Sigma/\binom{J}{2}$ 的取值增加且收敛于

$E[\phi]$。另外，随着 J 的增加，$E[|\phi - \phi^*_\Sigma/\binom{J}{2}|]$ 快速减小，对应的正则化条件越来越严格。

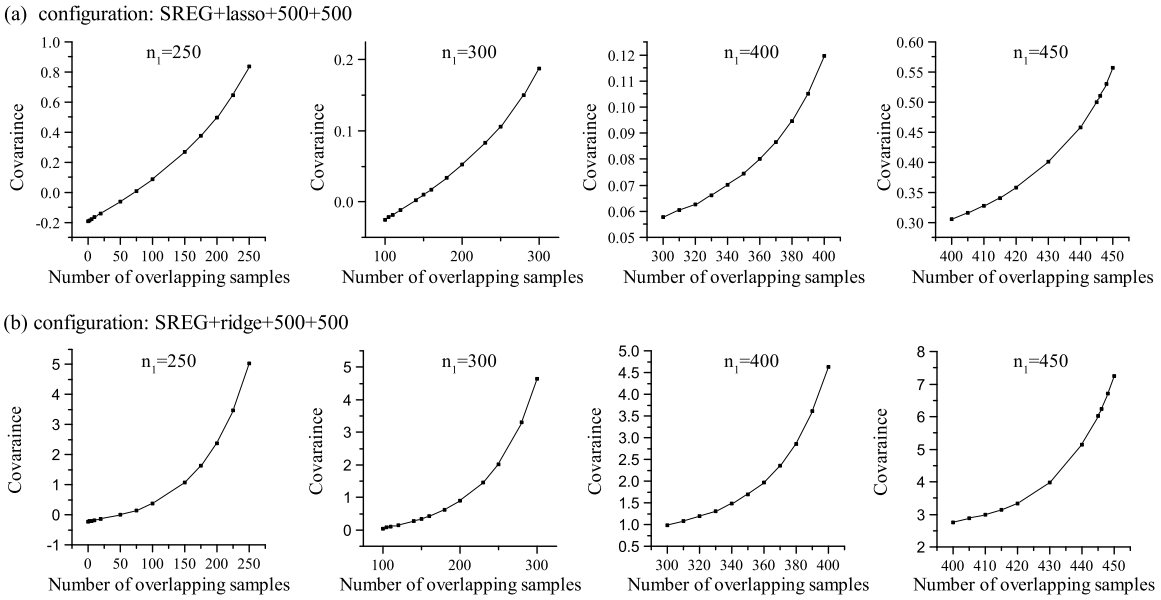

图 2.3 SREG 数据集上协方差函数 $f(x)$ 的模拟

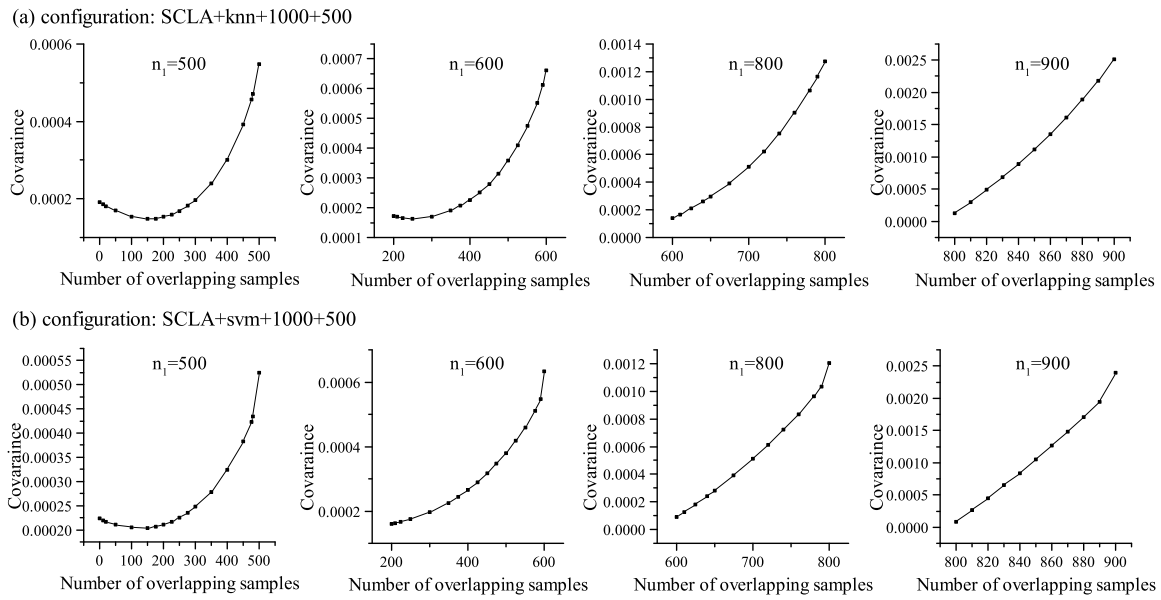

图 2.4 SCLA 数据集上协方差函数 $f(x)$ 的模拟

图2.3和图2.4 给出了 SREG 数据集和 SCLA 数据集上协方差函数 $f(x)$ 的模拟图像。这些图像表明，协方差函数 $f(x)$ 均为关于样本重叠个数的下凸函数。SREG 数据集上的协方差函数均单调递增。当 n_1 较小时，SCLA 数据集上的协方差函数开始时轻微递减、而后递增。这种非单调现象发生时，切分次数 J 相对较小，条

件 $x_{min} \leq 2n\binom{Jn_1/n}{2}/(J(J-1))$ 失效，相应的正则化 RLT 估计具有较大的方差。例如，在 $n_1 = 500$ 时，图2.4(a) 和图2.4(b) 的第一幅图像中，$f(x)$ 的最低点大约在 $x = 200$ 左右。若假设 $x_{min} = 200$，当 $J \geq 6$ 时，正则化 RLT 的方差最优性质并不能得到保证。

当协方差函数不是严格单调函数时，两个训练集间的少许重叠可以使得正则化 RLT 估计变得更稳定。当 n_1 增加且接近 n 时，$f(x)$ 严格单调递增。

2.4.2　研究问题二的模拟实验

本节中比较 K 折交叉验证估计与 RLT 估计的方差。K 折交叉验证为均衡 RLT 的一个特例。RLT 中，训练集大小 n_1 和切分集 J 与相应的 K 折交叉验证均相同。实验中，设置 $K = 2$、5 和 10。其中，5 折交叉验证和 10 折交叉验证被广泛使用。表2.9给出了模拟数据集和真实数据集上的实验结果。该实验结果包含了分类任务和回归任务的常用算法。

在表2.9中，当 $J = 2$ 且 $n_1 = n/2$，K 折交叉验证估计的大多数方差值均小于相应的 RLT 估计的方差值。不过，在实验配置“SREG+lm+500+100”和“SCLA+knn+1000+20”上，RLT 估计的方差值相对较小。这主要是因为当 $n_1 = n/2$ 时，协方差函数 $f(x)$ 为非单调函数。

表2.9表明，5 折交叉验证估计和 10 折交叉验证估计的方差比相应的 RLT 估计的方差更小。在分类数据集上，对应的方差约减率大于 10%。特别是，在 wdbc, optdigits 和 pageblock 数据集上，方差约减率大于 40%。

2.4.3　研究问题三的模拟实验

本节比较 RHS 估计、$m \times 2$ 交叉验证估计、正则化 $m \times 2$ 交叉验证估计和正则化 RHS 估计在多个模拟数据集和真实数据集上的方差。表2.10 给出了它们的方差值。表2.11给出了相应的方差约减率。

表2.10验证了定理2.3 的理论结果。在 4 种交叉验证估计中，正则化 RHS 估计的方差达到最小。该表的结果表明正则化 RHS 方法可明显地改善 RHS 估计的方差，相应的方差约减率的范围为 1% 到 24%。$m \times 2$ 交叉验证估计和正则化 $m \times 2$ 交叉验证估计的方差约减率均小于 10%。随着 J 或者 m 的增加，因为正则化条件中 $\phi^*_\Sigma/\binom{J}{2}$ 的值增加且收敛于 $n/4$，所以相应的方差约减率随之递减。

由于表2.10仅仅给出 4 种交叉验证估计的理论方差的点模拟值，进一步计算这些点模拟值的标准误是非常有必要的。本节中，标准误的模拟需要将重复 20 次实验。给定实验配置，每次模拟实验得到 1000 万个泛化误差估计值。由于计算开销

表 2.9 RLT 估计和 K 折交叉验证估计的方差比较

实验配置（数量级）	$\mathrm{Var}[\hat{\mu}_{RLT}(\mathbb{S})]$			$\mathrm{Var}[\hat{\mu}_{KFCV}(\mathbb{S}^b)]$			$RR_{RLT/KFCV}(\%)$		
	$J=2$	$J=5$	$J=10$	$J=2$	$J=5$	$J=10$	$J=2$	$J=5$	$J=10$
	$n_1=\frac{n}{2}$	$n_1=\frac{4n}{5}$	$n_1=\frac{9n}{10}$	$n_1=\frac{n}{2}$	$n_1=\frac{4n}{5}$	$n_1=\frac{9n}{10}$	$n_1=\frac{n}{2}$	$n_1=\frac{4n}{5}$	$n_1=\frac{9n}{10}$
SREG+lasso+500+500(10^{-1})	5.067	0.7542	3.475	3.273	0.7069	3.350	35.41	6.27	3.60
SREG+ridge+500+500	3.190	1.955	3.392	2.703	1.805	4.478	15.26	7.71	24.25
SREG+lm+500+100(10^{-2})	2.659	1.514	1.525	2.822	1.126	0.9287	-6.14	25.64	39.09
SCLA+svm+1000+20(10^{-4})	3.757	4.364	4.590	3.747	3.142	3.049	0.27	27.99	33.58
SCLA+knn+1000+20(10^{-4})	3.596	4.664	5.155	3.714	3.713	3.714	-3.30	20.39	27.96
glass+svm+150+9(10^{-3})	3.414	3.829	4.207	3.365	3.074	3.188	1.43	19.74	24.23
glass+knn+150+9(10^{-3})	1.992	1.706	1.720	1.936	1.215	1.052	2.81	28.80	38.84
iris+svm+100+4(10^{-3})	2.677	2.409	2.536	2.110	1.913	2.059	21.18	20.60	18.82
iris+knn+100+4(10^{-4})	7.776	5.681	5.121	7.378	3.826	3.200	5.12	32.65	37.50
krvskp+knn+200+36(10^{-3})	1.320	1.366	1.458	1.417	1.030	0.955	-7.29	24.58	34.50
krvskp+svm+200+36(10^{-3})	5.071	2.399	2.200	4.936	1.719	1.524	2.65	28.37	30.75
letter+svm+1000+16(10^{-4})	4.611	5.320	5.551	4.408	4.259	4.264	4.39	19.94	23.18
letter+knn+1000+16(10^{-4})	2.033	1.862	1.860	2.039	1.329	1.199	-0.30	28.59	35.53
optdigits+knn+300+64(10^{-4})	5.022	4.109	4.132	4.462	2.692	2.425	11.15	34.47	41.31
optdigits+svm+300+64(10^{-4})	3.975	6.363	8.920	3.795	6.041	8.023	4.52	5.06	10.07
pageblock+knn+300+10(10^{-4})	3.082	3.271	3.425	2.468	2.108	2.030	19.93	35.55	40.72
pageblock+svm+300+10(10^{-4})	3.999	4.381	4.623	2.687	2.644	2.723	32.81	39.65	41.10
spambase+knn+250+57(10^{-4})	9.971	9.854	10.438	8.793	7.178	6.885	11.81	27.15	34.04
spambase+svm+250+57(10^{-2})	2.444	1.914	1.862	1.953	1.712	1.778	20.09	10.54	4.46
wave+knn+300+40(10^{-3})	1.105	1.295	1.425	1.131	1.018	0.996	-2.33	21.41	30.15
wave+svm+300+40(10^{-4})	8.171	9.011	9.689	8.052	6.683	6.448	1.45	25.83	33.45
wdbc+svm+300+30(10^{-4})	2.595	3.334	3.753	2.131	2.574	2.843	17.90	22.80	24.24
wdbc+knn+300+30(10^{-4})	1.933	1.623	1.610	1.718	1.046	0.9036	11.16	35.53	43.88
yeast+svm+800+8(10^{-4})	16.76	8.099	8.309	14.63	6.340	6.314	12.68	21.72	24.00
yeast+knn+800+8(10^{-4})	4.698	4.798	5.075	4.961	3.658	3.305	-5.61	23.75	34.88

很大，本节仅给出了实验配置为“krvskp+knn+200+36”，“krvskp+svm+200+36”和“optdigits+knn+300+64”所对应的标准误结果。表2.12给出了标准误的结果，其格式为“均值 ± 标准误”。本节使用 3 倍标准误的置信区间来比较 4 种交叉验证估计，即：$[\text{mean} - 3\cdot\text{standard error}, \text{mean} + 3\cdot\text{standard error}]$。在上述 3 种实验配置上，正则化 RHS 方法的置信区间与其它 3 种交叉验证方法的置信区间均不重叠。这表明正则化 RHS 估计显著小于其它 3 种交叉验证估计。RHS 估计、$m\times 2$ 交叉验证估计和正则化 $m\times 2$ 交叉验证估计的方差之间没有显著差异。不过，这 3 种估计的方差的均值以 RHS、$m\times 2$ 交叉验证和正则化 $m\times 2$ 交叉验证的顺序递减。

表 2.10　RHS 估计、$m \times 2$ 交叉验证估计、正则化 $m \times 2$ 交叉验证估计和正则化 RHS 估计的方差比较

实验配置（数量级）	$\mathrm{Var}[\hat{\mu}_{RHS}(\mathbb{S})]$			$\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}_{m\times 2})]$			$\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b_{m\times 2})]$			$\mathrm{Var}[\hat{\mu}_{RHS}(\mathbb{S}^b)]$		
	$J=6$	$J=10$	$J=14$	$m=3$	$m=5$	$m=7$	$m=3$	$m=5$	$m=7$	$J=6$	$J=10$	$J=14$
SREG+lasso+500+500(10^{-1})	2.778	2.346	2.165	2.237	2.029	1.940	2.236	2.028	1.939	2.121	1.954	1.886
SREG+ridge+500+500	1.686	1.392	1.268	1.539	1.305	1.205	1.537	1.304	1.204	1.434	1.237	1.154
SREG+lm+500+100(10^{-2})	1.899	1.714	1.635	1.893	1.708	1.629	1.844	1.679	1.609	1.837	1.675	1.606
SCLA+svm+1000+20(10^{-4})	2.760	2.556	2.470	2.745	2.555	2.467	2.747	2.552	2.467	2.638	2.474	2.408
SCLA+knn+1000+20(10^{-4})	2.372	2.084	1.974	2.335	2.106	1.992	2.371	2.104	1.989	2.201	1.991	1.906
glass+svm+150+9(10^{-3})	2.222	1.993	1.895	2.209	1.979	1.882	2.215	1.985	1.886	2.097	1.906	1.830
glass+knn+150+9(10^{-4})	11.08	9.403	8.692	11.06	9.388	8.679	11.00	9.328	8.611	10.06	8.676	8.107
iris+svm+100+4(10^{-3})	1.592	1.381	1.286	1.416	1.277	1.218	1.414	1.275	1.215	1.358	1.238	1.187
iris+knn+100+4(10^{-4})	4.201	3.502	3.192	4.099	3.443	3.162	4.081	3.421	3.138	3.745	3.185	2.972
krvskp+knn+200+36(10^{-4})	8.078	6.833	6.303	8.063	6.808	6.287	7.740	6.654	6.178	7.358	6.297	5.885
krvskp+svm+200+36(10^{-3})	2.601	2.105	1.896	2.566	2.071	1.866	2.566	2.070	1.857	2.254	1.868	1.703
letter+svm+1000+16(10^{-4})	3.638	3.443	3.358	3.577	3.411	3.339	3.574	3.408	3.336	3.505	3.363	3.303
letter+knn+1000+16(10^{-5})	11.94	10.26	9.530	11.95	10.25	9.524	11.93	10.24	9.519	10.97	9.575	9.049
optdigits+knn+300+64(10^{-4})	2.891	2.466	2.279	2.708	2.350	2.200	2.702	2.341	2.197	2.506	2.214	2.093
optdigits+svm+300+64(10^{-4})	2.423	2.099	1.962	2.407	2.076	1.941	2.332	2.044	1.919	2.308	2.021	1.902
pageblock+knn+300+10(10^{-4})	2.050	1.841	1.751	1.845	1.721	1.666	1.843	1.717	1.661	1.770	1.667	1.625
pageblock+svm+300+10(10^{-4})	2.627	2.353	2.240	2.200	2.106	2.063	2.190	2.106	2.062	2.169	2.073	2.044
spambase+knn+250+57(10^{-4})	6.168	5.407	5.075	5.766	5.158	4.896	5.760	5.125	4.873	5.493	4.952	4.742
spambase+svm+250+57(10^{-2})	1.485	1.289	1.206	1.316	1.189	1.136	1.315	1.188	1.131	1.249	1.142	1.100
wave+knn+300+40(10^{-4})	7.418	6.678	6.331	7.406	6.660	6.321	7.375	6.624	6.308	6.988	6.324	6.101
wave+svm+300+40(10^{-4})	5.314	4.754	4.518	5.297	4.744	4.506	5.295	4.735	4.504	5.034	4.537	4.374
wdbc+svm+300+30(10^{-4})	1.727	1.548	1.474	1.574	1.462	1.414	1.573	1.461	1.413	1.533	1.435	1.393
wdbc+knn+300+30(10^{-5})	10.86	9.102	8.373	10.13	8.726	8.122	10.14	8.725	8.120	9.352	8.188	7.710
yeast+svm+800+8(10^{-4})	9.440	7.970	7.339	8.768	7.596	7.093	8.749	7.575	7.072	8.319	7.300	6.862
yeast+knn+800+8(10^{-4})	3.120	2.752	2.594	3.025	2.688	2.543	3.120	2.751	2.593	2.898	2.600	2.476

2.5　附录

2.5.1　引理2.1 的证明

因为 $B = I_2^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)})$，$C = I_1^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)})$ 且 $D = I_1^{(v)} \cap I_2^{(v)}$，所以 $|B| = |C| = n_1 - x$ 和 $|D| = n_2 - n_1 + x$，其中 x 为重叠样本个数。故，协方差函数 $f(x)$ 可表达为：

$$
\begin{aligned}
f(x) &\triangleq \mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_1), \hat{\mu}_{HO}(\mathcal{S}_2) | \phi = x] \\
&= \frac{1}{n_2^2} \mathrm{Cov}\left[\sum_{i \in I_1^{(v)}} e_i(\mathcal{S}_1), \sum_{j \in I_2^{(v)}} e_j(\mathcal{S}_2) \middle| \phi = x \right]
\end{aligned}
$$

表 2.11 RHS 估计、$m \times 2$ 交叉验证估计、正则化 $m \times 2$ 交叉验证估计和正则化 RHS 估计的方差约减率

实验配置	$RR_{RHS/BRHS}(\%)$			$RR_{m\times 2CV/BRHS}(\%)$			$RR_{m\times 2BCV/BRHS}(\%)$		
	$J=6$	$J=10$	$J=14$	$J=6$	$J=10$	$J=14$	$J=6$	$J=10$	$J=14$
	–	–	–	$(m=3)$	$(m=5)$	$(m=7)$	$(m=3)$	$(m=5)$	$(m=7)$
SREG+lasso+500+500	23.65	16.72	12.90	5.16	3.72	2.81	5.14	3.68	2.75
SREG+ridge+500+500	14.96	11.11	8.99	6.82	5.19	4.23	6.70	5.11	4.16
SREG+lm+500+100	3.28	2.27	1.75	2.97	1.94	1.41	0.41	0.20	0.17
SCLA+svm+1000+20	4.45	3.14	2.42	3.92	3.20	2.52	3.98	3.04	2.42
SCLA+knn+1000+20	5.72	4.49	3.46	7.21	5.47	4.33	7.18	5.38	4.20
glass+svm+150+9	5.07	3.66	2.79	5.60	4.35	3.44	5.33	3.97	2.99
glass+knn+150+9	9.18	7.58	6.59	9.04	7.74	6.73	8.52	6.99	5.85
iris+svm+100+4	14.71	10.34	7.71	4.12	3.06	2.55	4.01	2.89	2.34
iris+knn+100+4	10.86	9.08	6.88	8.64	7.50	5.99	8.25	6.92	5.29
krvskp+knn+200+36	8.91	7.84	6.64	8.74	7.50	6.40	4.93	5.36	4.75
krvskp+svm+200+36	13.35	11.25	10.21	12.17	9.80	8.75	12.16	9.73	8.30
letter+svm+1000+16	3.67	2.31	1.63	2.01	1.39	1.08	1.94	1.30	0.99
letter+knn+1000+16	8.16	6.67	5.04	8.19	6.59	4.99	8.11	6.53	4.94
optdigits+knn+300+64	13.33	10.19	8.14	7.46	5.76	4.87	7.28	5.42	4.71
optdigits+svm+300+64	4.75	3.74	3.02	4.14	2.68	2.00	1.03	1.13	0.86
pageblock+knn+300+10	13.62	9.47	7.19	4.07	3.13	2.41	3.95	2.91	2.15
pageblock+svm+300+10	17.46	11.91	8.74	1.42	1.57	0.92	0.97	1.59	0.87
spambase+knn+250+57	10.94	8.40	6.56	4.74	3.99	3.15	4.64	3.36	2.68
spambase+svm+250+57	15.86	11.39	8.81	5.06	3.95	3.17	5.00	3.82	2.76
wave+knn+300+40	5.80	5.31	3.63	5.65	5.05	3.47	5.25	4.54	3.28
wave+svm+300+40	5.28	4.57	3.19	4.98	4.38	2.93	4.93	4.20	2.90
wdbc+svm+300+30	11.23	7.29	5.49	2.56	1.85	1.52	2.51	1.80	1.47
wdbc+knn+300+30	13.86	10.04	7.92	7.72	6.17	5.08	7.75	6.15	5.05
yeast+svm+800+8	11.88	8.41	6.50	5.12	3.89	3.25	4.91	3.63	2.96
yeast+knn+800+8	4.19	3.30	2.64	7.11	5.52	4.55	7.11	5.50	4.53

$$
\begin{aligned}
= \quad & \frac{1}{n_2^2}\left\{ Cov\left[\sum_{i\in C} e_i(\mathcal{S}_1), \sum_{j\in B} e_j(\mathcal{S}_2)\middle|\phi = x\right]\right. \\
& +\mathrm{Cov}\left[\sum_{i\in C} e_i(\mathcal{S}_1), \sum_{k\in D} e_k(\mathcal{S}_2)\middle|\phi = x\right] \\
& \left.+\mathrm{Cov}\left[\sum_{k\in D} e_k(\mathcal{S}_1), \sum_{j\in B} e_j(\mathcal{S}_2)\middle|\phi = x\right]\right.
\end{aligned}
$$

表 2.12　三种实验配置上 RHS 估计、$m\times 2$ 交叉验证估计、正则化 $m\times 2$ 交叉验证估计和正则化 RHS 估计的方差的样本均值和标准误（$\times 10^{-4}$）

	$\mathrm{Var}[\hat{\mu}_{RHS}(\mathbb{S})]$	$\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}_{m\times 2})]$	$\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b_{m\times 2})]$	$\mathrm{Var}[\hat{\mu}_{RHS}(\mathbb{S}^b)]$
	krvskp+knn+200+36			
J=6(m=3)	8.0786±0.0288	8.0713±0.0267	7.7865±0.0215	7.3539±0.0172
J=10(m=5)	6.8355±0.0286	6.8380±0.0200	6.6172±0.0191	6.2720±0.0159
J=14(m=7)	6.3015±0.0194	6.2834±0.0219	6.1782±0.0115	5.8605±0.0205
	krvskp+svm+200+36			
J=6(m=3)	26.0090±0.0564	25.7413±0.0929	25.7038±0.0972	22.4936±0.0554
J=10(m=5)	21.0056±0.0429	20.7225±0.0532	20.6494±0.0439	18.6939±0.0492
J=14(m=7)	19.0159±0.0626	18.6325±0.0489	18.5628±0.0479	17.0150±0.0471
	optdigits+knn+300+64			
J=6(m=3)	2.8974±0.0041	2.7020±0.0028	2.7033±0.0035	2.5046±0.0042
J=10(m=5)	2.4669±0.0032	2.3464±0.0043	2.3396±0.0043	2.2154±0.0024
J=14(m=7)	2.2838±0.0032	2.2004±0.0025	2.1998±0.0038	2.0923±0.0037

$$+\mathrm{Cov}\left[\sum_{k\in D} e_k(\mathcal{S}_1), \sum_{k'\in D} e_{k'}(\mathcal{S}_2)\middle|\phi = x\right]\Bigg\}。 \tag{2.17}$$

基于式（2.10），可得 $f(x)=\frac{1}{n_2^2}[(\omega+\gamma-2\tau)x^2+\alpha_1 x+\alpha_0]$，其中 $\alpha_1=\sigma^2-(2n_1-2n_2+1)\omega-2n_1\gamma+2(2n_1-n_2)\tau$ 且 $\alpha_0=(n_1-n_2)[(n_1-n_2+1)\omega+n_1^2\gamma-2n_1\tau-\sigma^2]$。因此，当 $\omega+\gamma>2\tau$ 时，$f(x)$ 为重叠样本个数 x 的下凸函数。

2.5.2　正则化 RLT 中重叠样本个数矩阵与样例出现频次向量的取值

一般地，若向量 $\boldsymbol{v}=(v_1,\ldots,v_n)$ 满足 $|v_i-v_j|\le 1$ 且 $\sum_{i=1}^n v_i=s$，其中，$i,j=1,\ldots,n$，则向量 $\boldsymbol{v}$ 中有 $s-\lfloor s/n\rfloor n$ 个元素的值为 $\lfloor s/n\rfloor+1$，其余 $(\lfloor s/n\rfloor+1)n-s$ 个元素的值为 $\lfloor s/n\rfloor$。若 n 可整除 s，则 $\boldsymbol{v}$ 中的所有 n 个元素的取值为 s/n，且满足 $|v_i-v_j|\equiv 0$。

在正则化 RLT 中的度量 $\boldsymbol{\Psi}^b$ 中，$|\psi_i-\psi_j|\le 1$ 且 $\sum_{i=1}^n\psi_i=n_1J$，其中：$i,j=1,\ldots,n$。因此，在 $\boldsymbol{\Psi}^b$ 中，有 $Jn_1-\lfloor Jn_1/n\rfloor n$ 个元素的取值为 $\lfloor Jn_1/n\rfloor+1$，其余 $(\lfloor Jn_1/n\rfloor+1)n-Jn_1$ 个元素的取值为 $\lfloor Jn_1/n\rfloor$。显然，若 n 可整除 Jn_1，则 $\boldsymbol{\Psi}^b$ 中所有元素的取值为 Jn_1/n。

对于度量 $\boldsymbol{\Phi}^b$，见式（2.2），则所有重叠样本个数之和为：

$$\phi^*_\Sigma = \sum_{j=1}^{J}\sum_{j'=j+1}^{J}\phi_{jj'}=\sum_{i=1}^{n}\binom{\psi_i}{2}$$

$$= \binom{\lfloor Jn_1/n \rfloor + 1}{2}(Jn_1 - \lfloor Jn_1/n \rfloor n) + \binom{\lfloor Jn_1/n \rfloor}{2}((\lfloor Jn_1/n \rfloor + 1)n - Jn_1)。 \quad (2.18)$$

因 $|\phi_{ii'} - \phi_{jj'}| \leq 1$，其中 $i, i', j, j' = 1, \ldots, J$，则 $\mathbf{\Phi}^b$ 中有 $\phi^*_\Sigma - \lfloor \phi^*_\Sigma / \binom{J}{2} \rfloor \binom{J}{2}$ 个元素的取值为 $\lfloor \phi^*_\Sigma / \binom{J}{2} \rfloor + 1$，其余的 $(\lfloor \phi^*_\Sigma / \binom{J}{2} \rfloor + 1)\binom{J}{2} - \phi^*_\Sigma$ 个元素的取值为 $\lfloor \phi^*_\Sigma / \binom{J}{2} \rfloor$。

因此，广义的正则化条件为

$$|\phi_{jj'} - \frac{\phi^*_\Sigma}{\binom{J}{2}}| \leq k。 \quad (2.19)$$

其中，$j, j' = 1, \ldots, J$。

2.5.3 定理2.2 的证明

定理2.2可基于二水平正交表的正交性质来证明。具体地，在二水平正交表 $OA(2J, 2J-1)$ 中，任意两列的水平对均出现 $J/2$ 次。但是，在每一行中，水平“$+$”的频次与水平“$-$”的频次不相同。

算法2.1中，将正交表 $OA(2J, 2J-1)$ 裁剪成子表 $OA_2(2J-2, J)$ 后，水平对 $(+,+)$ 和 $(-,-)$ 的出现频次均减少，如表2.3 所示。具体地，在子表 $OA_2(2J-2, J)$ 中，水平对 $(+,+)$ 和 $(-,-)$ 均出现 $J/2-1$ 次，而水平对 $(+,-)$ 和 $(-,+)$ 的出现频次为 $J/2$。不过，在 $OA_2(2J-2, J)$ 的每一行中，水平“$+$”和“$-$”的出现频次均为 $J/2$。

因此，子表 $OA_2(2J-2, J)$ 满足两条约束：（1）任意两列中水平对 $(+,+)$ 的出现频次为 $J/2-1$；（2）每一行中水平 $+$ 的出现频次为 $J/2$。这两个约束严格满足 BRLT 的正则化条件。其中，当 $2J-2$ 整除 n 时，正则化参数 $k=0$。若 $2J-2$ 不能整除 n，因数据集 D_n 的 $2J-2$ 个子块的大小近似相等，所以正则化参数 k 也相对较小。另外，当 $2J-2$ 整除 n 时，正则化 RLT 则退化为均衡 RLT。

2.5.4 定理2.3的证明

定理2.3中，关于 $m \times 2$ 交叉验证估计的方差和正则化 $m \times 2$ 交叉验证估计的方差的不等式，即

$$E_{\mathbf{\Phi}_{m\times 2}}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}_{m\times 2})|\mathbf{\Phi}_{m\times 2}]] \geq E_{\mathbf{\Phi}^b_{m\times 2}}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b_{m\times 2})|\mathbf{\Phi}^b_{m\times 2}]], \quad (2.20)$$

已在定理3.1 中证明。

当 $J = 2m$ 时，关于 RLT 估计方差与 $m \times 2$ 交叉验证估计方差的不等式，即：

$$E_{\mathbf{\Phi}}[\mathrm{Var}[\hat{\mu}_{RHS}(\mathbb{S})|\mathbf{\Phi}]] \geq E_{\mathbf{\Phi}_{m\times 2}}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}_{m\times 2})|\mathbf{\Phi}_{m\times 2}]], \quad (2.21)$$

可通过将 J 个切分分为 m 组来证明。在 m 组中，每 1 组有两个切分，且每一组对应的 RLT 估计的方差大于 $m \times 2$ 交叉验证中每 1 个 2 折交叉验证估计的方差，因为 2 折交叉验证为均衡 RLT 的特例（见推论2.2）。另外，因为在 RLT 及 $m \times 2$ 交叉验证中，m 组切分均随机且独立，所以任意两组对应的两个 RLT 估计间的协方差与 $m \times 2$ 交叉验证中任意两个 2 折交叉验证估计间的协方差相同。因此，式（2.21）成立。

当正则化 $m \times 2$ 交叉验证和正则化 RLT 中正则化参数 k 相同时，因为正则化 $m \times 2$ 交叉验证和正则化 RLT 中 $\phi_{jj'}$ 的和均相同，在 $\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b_{m\times 2})|\mathbf{\Phi}^b_{m\times 2}]$ 所含的协方差函数上应用詹森不等式，则可证明：

$$E_{\mathbf{\Phi}^b_{m\times 2}}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b_{m\times 2})|\mathbf{\Phi}^b_{m\times 2}]] \geq E_{\mathbf{\Phi}^b}[\mathrm{Var}[\hat{\mu}_{RHS}(\mathbb{S}^b)|\mathbf{\Phi}^b]]。 \tag{2.22}$$

2.6　本章小结

本章针对 RLT 提出了一种优化的数据切分方法，称为正则化 RLT。正则化 RLT 通过正则化训练集间重叠样本个数来减少泛化误差的 RLT 估计的方差。本文给出了正则化 RLT 若干构造方法，并通过模拟数据集和真实数据集上的实验验证了正则化 RLT 方法的优良性质。

针对算法比较任务，5×2 交叉验证和 3×2 交叉验证已经取得了不错的效果。5×2 交叉验证和 3×2 交叉验证可自然拓广为 $m \times 2$ 交叉验证，其中，m 为 2 折交叉验证的重复次数。因为 $m \times 2$ 交叉验证为 RLT 方法的一种特殊情形，因此，RLT 中数据切分的优化方法可应用于 $m \times 2$ 交叉验证的优化问题中。

第三章　正则化 $m \times 2$ 交叉验证方法

$m \times 2$ 交叉验证指 m 次随机重复的 2 折交叉验证。$m \times 2$ 交叉验证的一些特殊版本，如，5×2 交叉验证[1,6,7] 和 3×2 交叉验证[8]，已在算法比较任务中取得了不错的效果。此外，2 折交叉验证引起简单易用也被广泛应用在多种机器学习任务中。例如，Nason 等使用 2 折交叉验证选取小波函数收缩所用的超参数[78]。Fan 等人和 Chen 等将 2 折交叉验证应用在超高维线性回归模型和线性可加模型的方差估计中[89,90]。Stanišić 等将 2 折交叉验证应用于频繁集挖掘任务中[91]。Wainer 等指出当样本量大于 1000 时，2 折交叉验证在支持向量机的超参数选择中取得不错的效果[92]。这些研究工作也均可通过 $m \times 2$ 交叉验证得以改进。

然而，$m \times 2$ 交叉验证的性能依赖于其采用的数据切分方式。坏的数据切分会导致 2 折交叉验证的训练集间重叠样本过多，进而增大泛化误差的 $m \times 2$ 交叉验证估计的方差。为此，本章通过在重叠样本个数上建立正则化条件，来优化 $m \times 2$ 交叉验证的数据切分方式，进而提出正则化 $m \times 2$ 交叉验证方法。

3.1　记号及定义

假定数据集 D_n 由 n 个样本组成，即：$D_n = \{z_i : z_i = (x_i, y_i), i = 1, \ldots, n\}$。其中，样例 z_i 均独立同分布地抽取自未知分布 $\mathcal{P}$，x_i 为特征向量，y_i 为响应变量。$\mathcal{A}(D_n)$ 为算法 $\mathcal{A}$ 在数据集 D_n 上训练所得模型。损失函数记为 $L(.,.)$。通常，0-1 损失常用于分类任务，平方损失常用于回归任务。算法 $\mathcal{A}$ 的泛化误差可定义为

$$\mu(n) \triangleq E_{D_n, z}[L(\mathcal{A}(D_n), z)]。 \tag{3.1}$$

本章主要研究泛化误差的 $m \times 2$ 交叉验证估计。在 $m \times 2$ 交叉验证中，数据集 D_n 被多次（m 次）随机切分为两等分, 每次切分对应单次 2 折交叉验证。为此，引入如下定义及记号。

定义 3.1　$\mathcal{S} \triangleq (I^{(t)}, I^{(v)})$ 被称为指标集 $\mathcal{I} = \{1, 2, \ldots, n\}$ 的一次**切分**。其中，$I^{(t)}$ 和 $I^{(v)}$ 均从 $\mathcal{I}$ 中随机无放回抽取。$I^{(t)}$ 和 $I^{(v)}$ 满足 $I^{(t)} \cup I^{(v)} = \mathcal{I}$, $I^{(t)} \cap I^{(v)} = \varnothing$ 且 $|I^{(t)}| = |I^{(v)}| = n/2$。那么，$\mathbb{S} = \{< \mathcal{S}_i, \mathcal{S}_i^{\top} >: \mathcal{S}_i = (I_i^{(t)}, I_i^{(v)}), \mathcal{S}_i^{\top} = (I_i^{(v)}, I_i^{(t)}), i = 1, 2, \ldots, m\}$ 为 $m \times 2$ 交叉验证的**切分集**。

定义 3.2　$D^{(t)} = \{z_i : i \in I^{(t)}\}$ 和 $D^{(v)} = \{z_i : i \in I^{(v)}\}$ 分别被称为**训练集**和**验证集**。则，$D_n = D^{(t)} \cup D^{(v)}$。在 2 折交叉验证中，$D^{(t)}$ 和 $D^{(v)}$ 被轮流地用作训练集和验证集。

注记 3.1 在 2 折交叉验证中，训练集 $D^{(t)}$ 和验证集 $D^{(v)}$ 被泛称为**数据块**。

定义 3.3 对于切分集 $\mathbb{S}$ 中的任意两个切分 $\mathcal{S}_i=(I_i^{(t)},I_i^{(v)})$ 和 $\mathcal{S}_j=(I_j^{(t)},I_j^{(v)})$ 定义 $\phi_{ij}=|I_i^{(t)}\cap I_j^{(t)}|$ 为 $\mathcal{S}_i$ 和 $\mathcal{S}_j$ 间的**重叠样本个数**。其中，$\phi_{ij}=x$，$0\le x\le n/2$ 且 $i,j=1,2,\ldots,m$。矩阵 $\mathbf{\Phi}=(\phi_{ij})$ 为切分集 $\mathbb{S}$ 的一个度量。

注记 3.2 切分集 $\mathbb{S}$ 的第 i 个和第 j 个切分分别为 $<\mathcal{S}_i,\mathcal{S}_i^{\top}>$ 及 $<\mathcal{S}_j,\mathcal{S}_j^{\top}>$。这两个切分间共有四个重叠样本个数，分别为：$|\mathcal{S}_i\cap\mathcal{S}_j|=\phi_{ij}$，$|\mathcal{S}_i^{\top}\cap\mathcal{S}_j|=n/2-\phi_{ij}$，$|\mathcal{S}_i\cap\mathcal{S}_j^{\top}|=n/2-\phi_{ij}$ 及 $|\mathcal{S}_i^{\top}\cap\mathcal{S}_j^{\top}|=\phi_{ij}$。显然，这四个重叠个数仅与 ϕ_{ij} 有关。因此，仅分析 ϕ_{ij} 的基本性质及分布即可。

实际上，ϕ_{ij} 为整型随机变量，其取值范围为 $[0,n/2]$。Markatou 等证明 ϕ_{ij} 服从超几何分布，且其期望为 $n/4$[20]。当切分集所含切分多于两个时，可得到多个重叠样本个数。为减少 $m\times 2$ 交叉验证估计的方差，这些重叠样本个数与期望 $n/4$ 间的多个差值应被同时正则化。基于此直觉，本章提出了一种新的数据切分方法来控制这些差值，并将该切分方法称为正则化 $m\times 2$ 交叉验证切分, 旨在控制每个重叠样本个数与 $n/4$ 间的差值，使其小于随机情形下该差值的期望值。随机情形下，该差值的期望值[93] 如下：

$$E\left[\left|\phi_{ij}-\frac{n}{4}\right|\right]=\frac{n^2}{4(n-1)}\frac{\binom{2n'-1}{n'}\binom{2n'-1}{n'-1}}{\binom{n-2}{2n'-1}}+\frac{n(n-2)}{8(n-1)}\frac{\binom{2n'-1}{n'-1}^2}{\binom{n-2}{2n'-2}}-\frac{n}{4}\frac{\binom{2n'}{n'}^2}{\binom{n}{2n'}}。\tag{3.2}$$

其中，n 为数据集大小且 $n'=n/4$。

基于上述分析，将正则化交叉验证的切分定义如下。

定义 3.4 给定 $m\times 2$ 交叉验证的切分集 $\mathbb{S}$ 及相应的重叠样本个数矩阵 $\mathbf{\Phi}$，若 $\mathbf{\Phi}$ 中的所有非对角元素 ϕ_{ij} （$i\neq j$）满足 $|\phi_{ij}-n/4|\le c$（$c\ge 0$ 为正则化参数），则切分集 $\mathbb{S}$ 被称为**正则化切分集**，并被记为 $\mathbb{S}^b$，对应的 $\mathbf{\Phi}$ 矩阵记为 $\mathbf{\Phi}^b$。在切分集 $\mathbb{S}^b$ 上定义的 $m\times 2$ 交叉验证被称为**正则化** $m\times 2$ **交叉验证**（简记为 $m\times 2$ BCV）。当 $n=4l,l\in N^+$ 且 $c=0$，即，$\phi_{ij}\equiv n/4$，相应的正则化 $m\times 2$ 交叉验证被称为**均衡** $m\times 2$ **交叉验证**，并将其切分集合及重叠样本个数矩阵记为 $\mathbb{S}^*$ 和 $\mathbf{\Phi}^*$。

注记 3.3 正则化参数 c 不应超过 $|\phi_{ij}-n/4|$ 的期望值（见式（3.2））。

下面，引入一些泛化误差估计的定义[5]。

定义 3.5 给定切分 $\mathcal{S}=(I^{(t)},I^{(v)})$，泛化误差 $\mu(n)$ 的 **hold-out 估计**定义为：

$$\hat{\mu}_{HO}(\mathcal{S})\triangleq\frac{1}{|I^{(v)}|}\sum_{j\in I^{(v)}}L(\mathcal{A}(D^{(t)});z_j)=\frac{2}{n}\sum_{j\in I^{(v)}}L(\mathcal{A}(D^{(t)});z_j)。\tag{3.3}$$

$\mu(n)$ 的 **2 折交叉验证估计**定义为：

$$\hat{\mu}(\mathcal{S}) \triangleq \frac{1}{2}\hat{\mu}_{HO}(\mathcal{S}) + \frac{1}{2}\hat{\mu}_{HO}(\mathcal{S}^{\top})。 \tag{3.4}$$

其中 $\mathcal{S}^{\top} = (I^{(v)}, I^{(t)})$。

$\mu(n)$ 的$m \times 2$ **交叉验证估计**定义为：

$$\hat{\mu}_{m\times 2}(\mathbb{S}) \triangleq \frac{1}{m}\sum_{i=1}^{m}\hat{\mu}(\mathcal{S}_i)。 \tag{3.5}$$

其中，$\hat{\mu}(\mathcal{S}_i)$ 为切分 $\mathcal{S}_i$ 对应的 2 折交叉验证估计。相应地，基于切分集 $\mathbb{S}^b$，$\mu(n)$ 的估计被称为**正则化** $m \times 2$ **交叉验证估计**，且记为 $\hat{\mu}_{m\times 2}(\mathbb{S}^b)$。

3.2　$m \times 2$ 交叉验证估计的方差的理论分析

泛化误差的 $m \times 2$ 交叉验证估计的方差有如下的分解。

$$\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})] = \frac{1}{m^2}\sum_{i=1}^{m}\mathrm{Var}[\hat{\mu}(\mathcal{S}_i)] + \frac{1}{m^2}\sum_{i\neq j, i,j=1,2,\ldots,m}\mathrm{Cov}[\hat{\mu}(\mathcal{S}_i), \hat{\mu}(\mathcal{S}_j)]。 \tag{3.6}$$

其中，方差 $\mathrm{Var}[\cdot]$ 取自数据集 D_n 和切分集 $\mathbb{S}$。本节假设数据集大小 n 为固定值，且仅使用 $\boldsymbol{\Phi}$ 来度量 $\mathbb{S}$。因此，$\forall S_i \in \mathbb{S}$，$\mathrm{Var}[\hat{\mu}(\mathcal{S}_i)]$ 均相等且与 i 无关。由于数据集 D_n 中所有样例 z_i 均独立同分布，协方差 $\mathrm{Cov}[\hat{\mu}(\mathcal{S}_i), \hat{\mu}(\mathcal{S}_j)]$ 仅依赖于重叠样本个数 $\phi_{ij} = |I_i^{(t)} \cap I_j^{(t)}|$。因此，

$$\begin{aligned}\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})|\boldsymbol{\Phi}] &= \mathrm{Var}\left[\frac{1}{m}\sum_{i=1}^{m}\hat{\mu}(\mathcal{S}_i)|\Phi\right] \\ &= \frac{1}{m}\mathrm{Var}\left[\hat{\mu}(\mathcal{S}_1)\right] + \frac{1}{m^2}\sum_{i\neq j, i,j=1}^{m}\mathrm{Cov}\left[\hat{\mu}(\mathcal{S}_i), \hat{\mu}(\mathcal{S}_j)|\phi_{ij}\right]。\end{aligned} \tag{3.7}$$

其中，$\mathrm{Var}[\cdot|\cdot]$ 和 $\mathrm{Cov}[\cdot,\cdot|\cdot]$ 均取自数据集 D_n。

受统计实验设计方法[38] 的启发，本节引入切分集 $\mathbb{S}$ 的特定设计，来减少随机变量 ϕ_{ij} 对 $\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})]$ 的影响。下面证明当 $\phi_{ij} = n/4$（$\forall i \neq j$）时，$\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})|\Phi]$ 达到其最小值，即：切分集 $\mathbb{S}^*$ 满足：

$$\mathbb{S}^* = argmin_{\mathbb{S}}\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})|\Phi]。 \tag{3.8}$$

对于上述优化问题，关键之处在于分析 $\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})|\Phi]$ 的表达式（见式（3.7））中协方差函数 $\mathrm{Cov}[\hat{\mu}(\mathcal{S}_i), \hat{\mu}(\mathcal{S}_j)|\phi_{ij}]$ 的理论性质。引理3.1 表明协方差函数 $\mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_i), \hat{\mu}_{HO}(\mathcal{S}_j)|\phi_{ij}]$ 为关于重叠样本个数的下凸函数。引理3.2 给出当 $\phi_{ij} = n/4$，$\mathrm{Cov}[\hat{\mu}(\mathcal{S}_i), \hat{\mu}(\mathcal{S}_j)|\phi_{ij}]$ 达到其最小值。

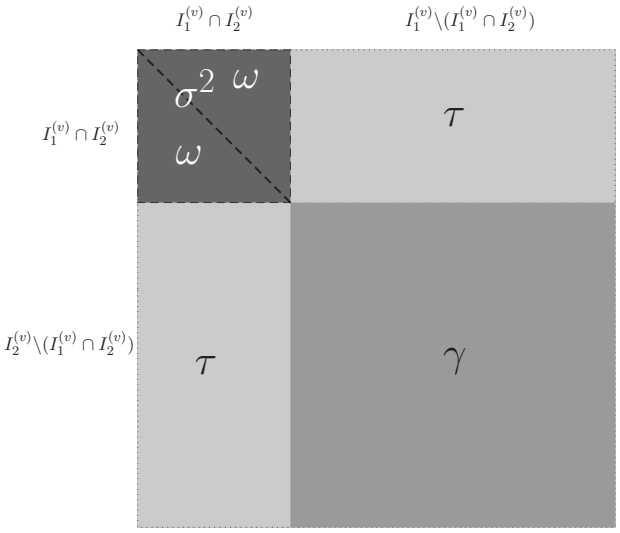

图 3.1 协方差函数结构及其参数示意

引理 3.1 设 $e_j(\mathcal{S}) \triangleq L(\mathcal{A}(D^{(t)}); z_j)$ 为样例 z_j 上的损失函数，其中，$j = 1, 2, \ldots, n/2$。令 $\mathcal{S}_1 = (I_1^{(t)}, I_1^{(v)})$ 和 $\mathcal{S}_2 = (I_2^{(t)}, I_2^{(v)})$ 为指标集 $\mathcal{I} = \{1, \ldots, n\}$ 上的两个随机切分，且 $\mathcal{S}_1$ 与 $\mathcal{S}_2$ 间的重叠样本个数为 ϕ，则有

(1) 对于 $i, j \in \{1, 2, \ldots, n/2\}$，当 $\phi = n/4$ 时，$\mathrm{Cov}[e_i(\mathcal{S}_1), e_j(\mathcal{S}_2)|\phi]$ 有如下形式：

$$\mathrm{Cov}[e_i(\mathcal{S}_1), e_j(\mathcal{S}_2)|\phi = n/4] = \begin{cases} \sigma^2 & i = j,\ i, j \in (I_1^{(v)} \cap I_2^{(v)}) \\ \omega & i \neq j,\ i, j \in (I_1^{(v)} \cap I_2^{(v)}) \\ \gamma & i \in I_1^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)}) \text{且} j \in I_2^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)}) \\ \tau & \text{其它} \end{cases} \tag{3.9}$$

其中，σ^2，ω，γ，及 τ 为常数，如图3.1 所示。

(2) 定义 $f(x) \triangleq \mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_1), \hat{\mu}_{HO}(\mathcal{S}_2)|\phi = x]$，则 $f(x)$ 为关于 x 的二次函数，即：

$$f(x) \triangleq \frac{4}{n^2}\left[(\omega + \gamma - 2\tau) \cdot x^2 + (\sigma^2 - \omega - n\gamma + n\tau) \cdot x + \frac{n^2}{4}\gamma\right]。 \tag{3.10}$$

因此，当 $\omega + \gamma \geq 2\tau$ 时，$f(x)$ 为关于 x 的下凸函数。

证明. 根据切分的定义（见定义3.1），可从切分 $\mathcal{S}_1 = (I_1^{(t)}, I_1^{(v)})$ 和 $\mathcal{S}_2 = (I_2^{(t)}, I_2^{(v)})$ 中得到四个指标子集：$I_1^{(t)} \cap I_2^{(t)}$、$I_1^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)}$、$I_1^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)})$ 和 $I_1^{(v)} \cap I_2^{(v)}$。

对于切分 $\mathcal{S}_1$，有

$$\mathcal{I} = I_1^{(t)} \bigcup I_1^{(v)} = \left[I_1^{(t)} \cap I_2^{(t)}\right] \cup \left[I_1^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})\right] \bigcup \left[I_1^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)})\right] \cup \left[I_1^{(v)} \cap I_2^{(v)}\right]。 \tag{3.11}$$

对于切分 $\mathcal{S}_2$，有

$$\mathcal{I} = I_2^{(t)} \bigcup I_2^{(v)} = \left[I_1^{(t)} \cap I_2^{(t)}\right] \cup \left[I_2^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})\right] \bigcup \left[I_2^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)})\right] \cup \left[I_2^{(v)} \cap I_2^{(v)}\right] 。 \tag{3.12}$$

可得：

$$I_1^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)}) = I_2^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)}), \quad I_2^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)}) = I_1^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)}) 。 \tag{3.13}$$

$$|I_1^{(t)} \cap I_2^{(t)}| = |I_1^{(v)} \cap I_2^{(v)}| = x, \quad |I_1^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})| = |I_2^{(t)} \backslash (I_1^{(t)} \cap I_2^{(t)})| = \frac{n}{2} - x 。 \tag{3.14}$$

则有：

$$\begin{aligned}
f(x) &\triangleq \mathrm{Cov}(\hat{\mu}_{HO}[\mathcal{S}_1), \hat{\mu}_{HO}(\mathcal{S}_2) | \phi = x] \\
&= \frac{4}{n^2} \mathrm{Cov}\left[\sum_{i \in I_1^{(v)}} e_i(\mathcal{S}_1), \sum_{j \in I_2^{(v)}} e_j(\mathcal{S}_2) \middle| \phi = x \right] \\
&= \frac{4}{n^2} \Bigg\{ \mathrm{Cov}\left[\sum_{i \in (I_1^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)}))} e_i(\mathcal{S}_1), \sum_{k \in (I_2^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)}))} e_k(\mathcal{S}_2) \middle| \phi = x \right] \\
&\quad + \mathrm{Cov}\left[\sum_{i \in (I_1^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)}))} e_i(\mathcal{S}_1), \sum_{j \in (I_1^{(v)} \cap I_2^{(v)})} e_j(\mathcal{S}_2) \middle| \phi = x \right] \\
&\quad + \mathrm{Cov}\left[\sum_{j \in (I_1^{(v)} \cap I_2^{(v)})} e_j(\mathcal{S}_1), \sum_{k \in (I_2^{(v)} \backslash (I_1^{(v)} \cap I_2^{(v)}))} e_k(\mathcal{S}_2) \middle| \phi = x \right] \\
&\quad + \mathrm{Cov}\left[\sum_{j \in (I_1^{(v)} \cap I_2^{(v)})} e_j(\mathcal{S}_1), \sum_{j \in (I_1^{(v)} \cap I_2^{(v)})} e_j(\mathcal{S}_2) \middle| \phi = x \right] \Bigg\} 。
\end{aligned} \tag{3.15}$$

因此，

$$\begin{aligned}
f(x) &= \frac{4}{n^2} \left[(\frac{n}{2} - x^2)\gamma + 2x(\frac{n}{2} - x)\tau + x\sigma^2 + (x^2 - x)\omega \right] \\
&= \frac{4}{n^2} \left[(\omega + \gamma - 2\tau) \cdot x^2 + (\sigma^2 - \omega - n\gamma + n\tau) \cdot x + \frac{n^2}{4}\gamma \right],
\end{aligned}$$

即：当 $\omega + \gamma > 2\tau$ 时，$f(x)$ 为关于 x 的下凸函数。 □

注记 3.4　实际上，参数 σ^2，ω，γ 和 τ 的值均与 x 有关。不过，引理3.1 主要关注这些参数在 $x = n/4$ 处的值，因为重叠样本个数的期望值为 $n/4$。

注记 3.5　在一般的损失函数、算法和数据分布的条件下，不容易证明条件 $\omega + \gamma > 2\tau$。不过，给定平方损失，对于均值回归、一元线性回归和多元线性回归，

可以证明条件 $\omega+\gamma>2\tau$ 成立。具体证明在第3.7 节中给出。另外，第3.5.2 节中的模拟实验结果也表明该条件是成立的。

引理 3.2 记 $\hat{\mu}(\mathcal{S}_1)$ 和 $\hat{\mu}(\mathcal{S}_2)$ 为在切分 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 定义的 $\mu(n)$ 的两个 2 折交叉验证。设 $g(x)\triangleq \mathrm{Cov}[\hat{\mu}(\mathcal{S}_1),\hat{\mu}(\mathcal{S}_2)|\phi=x]$。则 $\forall x\in[0,\frac{n}{2}]$，$g(x)=\frac{1}{2}(f(x)+f(\frac{n}{2}-x))$。那么，函数 $g(x)$ 具有如下两条性质：

(1) 对称性：$g(x)=g(\frac{n}{2}-x)$。

(2) 有界性：$g(\frac{n}{4})\leqslant g(x)\leqslant g(0)=\mathrm{Var}[\hat{\mu}(\mathcal{S}_i)], i=1,2,\ldots,m$。

证明. 若 $|I_1^{(t)}\cap I_2^{(t)}|=x$，则：

$$|I_1^{(v)}\cap I_2^{(v)}|=x,\quad |I_1^{(t)}\cap I_2^{(v)}|=|I_1^{(v)}\cap I_2^{(t)}|=\frac{n}{2}-x。\tag{3.16}$$

基于 2 折交叉验证估计的定义（见定义3.5），可得：

$$\begin{aligned} g(x) &\triangleq \mathrm{Cov}[\hat{\mu}(\mathcal{S}_1),\hat{\mu}(\mathcal{S}_2)|\phi=x]\\ &= \mathrm{Cov}\left[\frac{1}{2}(\hat{\mu}_{HO}(\mathcal{S}_1)+\hat{\mu}_{HO}(\mathcal{S}_1^T)),\frac{1}{2}(\hat{\mu}_{HO}(\mathcal{S}_2)+\hat{\mu}_{HO}(\mathcal{S}_2^T))\Big|\phi=x\right]\\ &= \frac{1}{4}\Big\{\mathrm{Cov}\left[\hat{\mu}_{HO}(\mathcal{S}_1),\hat{\mu}_{HO}(\mathcal{S}_2)\Big|\phi=x\right]+\mathrm{Cov}\left[\hat{\mu}_{HO}(\mathcal{S}_1),\hat{\mu}_{HO}(\mathcal{S}_2^T)\Big|\phi=\frac{n}{2}-x\right]\\ &+ \mathrm{Cov}\left[\hat{\mu}_{HO}(\mathcal{S}_1^T),\hat{\mu}_{HO}(\mathcal{S}_2)\Big|\phi=\frac{n}{2}-x\right]+\mathrm{Cov}\left[\hat{\mu}_{HO}(\mathcal{S}_1^T),\hat{\mu}_{HO}(\mathcal{S}_2^T)\Big|\phi=x\right]\Big\}。\end{aligned}\tag{3.17}$$

因 $f(x)\triangleq \mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_1),\hat{\mu}_{HO}(\mathcal{S}_2)|\phi=x]$，可得：

$$\mathrm{Cov}\left[\hat{\mu}_{HO}(\mathcal{S}_1^T),\hat{\mu}_{HO}(\mathcal{S}_2^T)|\phi=x\right]=f(x)。\tag{3.18}$$

$$\begin{aligned}\mathrm{Cov}\left[\hat{\mu}_{HO}(\mathcal{S}_1^T),\hat{\mu}_{HO}(\mathcal{S}_2)\Big|\phi=\frac{n}{2}-x\right] &= \mathrm{Cov}\left[\hat{\mu}_{HO}(\mathcal{S}_1),\hat{\mu}_{HO}(\mathcal{S}_2^T)\Big|\phi=\frac{n}{2}-x\right]\\ &= f(\frac{n}{2}-x)。\end{aligned}\tag{3.19}$$

则有

$$g(x)=\frac{1}{2}\left(f(x)+f\left(\frac{n}{2}-x\right)\right)。\tag{3.20}$$

显然，$g(x)=g(\frac{n}{2}-x)$，$g(x)$ 为对称函数，其对称轴为 $x=\frac{n}{4}$，且 $g(\frac{n}{4})=f(\frac{n}{4})$。

根据协方差函数的性质，可得

$$g(x)=\mathrm{Cov}[\hat{\mu}(\mathcal{S}_1),\hat{\mu}(\mathcal{S}_2)|\phi=x]\leqslant\sqrt{\mathrm{Var}[\hat{\mu}(\mathcal{S}_1)|\phi=x]\cdot\mathrm{Var}[\hat{\mu}(\mathcal{S}_2)|\phi=x]}。\tag{3.21}$$

因 $g(0)=\mathrm{Var}[\hat{\mu}(\mathcal{S}_i)|\phi=0]=\mathrm{Var}[\hat{\mu}(\mathcal{S}_i)], i=1,2,\ldots,m$ 且 $g(x)$ 为对称函数，可知

$$g(x)\leqslant g(0)=g\left(\frac{n}{2}\right)。\tag{3.22}$$

由于 $f(x)$ 为下凸函数（见引理3.1），基于詹森不等式，可得：

$$\frac{1}{2}f(x)+\frac{1}{2}f\left(\frac{n}{2}-x\right)\geqslant f\left(\frac{n}{4}\right)=g\left(\frac{n}{4}\right)。\tag{3.23}$$

即：

$$g(x)\geqslant g\left(\frac{n}{4}\right)。\tag{3.24}$$

其中，$x\in[0,\frac{n}{2}]$。 □

第3.5.3节给出了函数 $g(x)$ 的模拟图像，及基于 $x=\frac{n}{4}$ 处的参数 σ^2，ω，γ 和 τ 的模拟值所得的 $g(x)$ 的近似图像。

定理 3.1 给定数据集 D_n 上的切分集 $\mathbb{S}$ 及其度量 $\boldsymbol{\Phi}$，则泛化误差的 $m\times 2$ 交叉验证估计的方差满足如下不等式。

$$E_{\Phi}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})|\Phi]]\geqslant E_{\Phi^b}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b)|\Phi^b]]\geqslant E_{\Phi^*}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^*)|\Phi^*]]。\tag{3.25}$$

其中：

$$E_{\Phi^*}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^*)|\Phi^*]]=\frac{1}{2m}\sigma^2(1+\rho_1)+\frac{m-1}{m}\sigma^2\rho_2。\tag{3.26}$$

- $\sigma^2=f(\frac{n}{2})=\mathrm{Var}[\hat{\mu}_{HO}(\mathcal{S}_i)]$ 为 hold-out 估计的方差。
- $\rho_1=\frac{f(0)}{f(\frac{n}{2})}=\frac{\mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_i),\hat{\mu}_{HO}(\mathcal{S}_i^T)]}{\mathrm{Var}[\hat{\mu}_{HO}(\mathcal{S}_i)]}$ 为单个 2 折交叉验证估计所含的两个 hold-out 估计的相关系数。
- $\rho_2=\frac{f(\frac{n}{4})}{f(\frac{n}{2})}=\frac{\mathrm{Cov}[\hat{\mu}_{HO}(\mathcal{S}_i),\hat{\mu}_{HO}(\mathcal{S}_j)]}{\mathrm{Var}[\hat{\mu}_{HO}(\mathcal{S}_i)]}$ 为正则化 $m\times 2$ 交叉验证中任意两个 2 折交叉验证估计间的相关系数。

证明. 首先，为证明式（3.25）中的第一个不等式，引入随机变量 $\varphi=n/4-|\phi-n/4|$。其中，$\phi\in\Phi$。因 $0\leq\phi\leq n/2$，可得 $0\leq\varphi\leq n/4$。在 $g(\varphi)$ 上应用詹森不等式，可得：

$$E_{\phi}[g(\phi)]=E_{\varphi}[g(\varphi)]\geq g(E[\varphi])=g\left(\frac{n}{4}-E\left[\left|\phi-\frac{n}{4}\right|\right]\right)\geq g\left(\frac{n}{4}-c\right)。\tag{3.27}$$

因 $g(x)$ 为对称函数，则有

$$E_{\phi}[g(\phi)]\geq g\left(\frac{n}{4}\pm E\left[\left|\phi-\frac{n}{4}\right|\right]\right)\geq g\left(\frac{n}{4}\pm c\right)。\tag{3.28}$$

因此，式（3.25）中第一个不等式成立。因为 $g(\phi)$ 在 $\phi=n/4$ 处达到最小值，易推出式（3.25）中第二个不等式成立。

均衡 $m\times 2$ 交叉验证估计的方差有如下分解。

$$E_{\Phi^*}\left[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^*)|\Phi^*]\right]\;=\;E_{\Phi^*}\left[\mathrm{Var}\left[\frac{1}{m}\sum_{i=1}^{m}\hat{\mu}(\mathcal{S}_i)|\Phi^*\right]\right]$$

$$
\begin{aligned}
&= \frac{1}{m}g(0)+\frac{m-1}{m}g\left(\frac{n}{4}\right)\\
&= \frac{1}{m}\left(\frac{1}{2}\left(f(0)+f\left(\frac{n}{2}\right)\right)\right)+\frac{m-1}{m}f\left(\frac{n}{4}\right)\\
&= \frac{1}{2m}f\left(\frac{n}{2}\right)\left(1+\frac{f(0)}{f(\frac{n}{2})}\right)+\frac{m-1}{m}f\left(\frac{n}{2}\right)\frac{f(\frac{n}{4})}{f(\frac{n}{2})}\\
&= \frac{1}{2m}\sigma^2(1+\rho_1)+\frac{m-1}{m}\sigma^2\rho_2。
\end{aligned}
\tag{3.29}
$$

□

推论 3.1 对于均衡 $m\times 2$ 交叉验证的任意两个切分集 $\mathbb{S}_1^*$ 和 $\mathbb{S}_2^*$，有

$$
\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}_1^*)|\Phi]=\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}_2^*)|\Phi]。 \tag{3.30}
$$

推论 3.2 均衡 $m\times 2$ 交叉验证估计的方差为 m 的递减函数。当 m 增加时，$\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^*)|\Phi]$ 的第二部分 $\frac{m-1}{m}\sigma^2\rho_2$ 所占比重逐步增大。

3.3 正则化 $m\times 2$ 交叉验证切分集的增量式构造算法

上节给出正则化 $m\times 2$ 交叉验证的优良性质。不过，要将正则化 $m\times 2$ 交叉验证用于实际场景，需要有效地构造出正则化 $m\times 2$ 交叉验证的切分集 $\mathbb{S}^b$。McCarthy 等给出了切分集 $\mathbb{S}^b$ 的一种经典构造方法[21]。该构造方法将单个切分对应于正交表的一行。具体来说，设某二水平正交表有 k 列，则将数据集 D_n 等分为 k 块。第 i 块对应于正交表的第 i 列。然后，根据正交表中单行的水平分布，将数据块分别合并到对应切分中的训练集和验证集中。该算法弱点在于其并不能增量式构造，正则化 $(m+1)\times 2$ 交叉验证的切分集不能包含正则化 $m\times 2$ 交叉验证的切分集。因此，当 m 增大时，正则化 $m\times 2$ 交叉验证的切分集需要重新构造，相应的 $2m$ 个模型也需要重新训练和验证。

本节给出一种正则化 $m\times 2$ 交叉验证切分集 $\mathbb{S}^b$ 的增量式构造算法。当 m 增大时，该算法可以增量式地增加切分。该增量式构造算法及其理论性质在定理3.2 中给出。

定理 3.2 假定数据集 D_n 可切分为 $4k$ 等份数据子块[1]。记第 j 个数据子块为 $I_j^{(4k)}$, 其中，$k\in\{1,2,\ldots,n/4\}$。则切分集 $\mathbb{S}=\{\mathcal{S}_i=(I_i^{(t)},I_i^{(v)}),i=1,2,\ldots,4k-1\}$ 可基于二水平正交表 $OA(4k,2^{4k-1})$ 构造。构造步骤如下：

(1) 正交表 $OA(4k,2^{4k-1})$ 的第 j 行对应于第 j 个数据子块 $I_j^{(4k)}$，其第 i 列对应于 $\mathcal{S}_i=(I_i^{(t)},I_i^{(v)})$。其中，$I_i^{(t)}$ 由第 i 列中水平“+”对应的数据子块合并而成，$I_i^{(v)}$ 由第 i 列中水平“-”对应的数据子块合并而成。

[1]若 n 不能被 $4k$ 整除，则 $4k$ 个数据子块的大小相差应不超过 1。

(2) 将第（1）步应用于 $OA(4k, 2^{4k-1})$ 的所有列，可得 $4k-1$ 个切分 $\mathcal{S}_i$，其中，$i = 1, 2, \ldots, 4k-1$。

基于上述步骤构造的切分集 $\mathbb{S} = \{\mathcal{S}_i = (I_i^t, I_i^v), i = 1, 2, \ldots, 4k-1\}$ 为正则化切分集 $\mathbb{S}^b$，且该切分集满足正则化条件 $|\phi_{ij} - n/4| \leqslant k$，其中，$i \neq j$。

证明. 正交表 $OA(4k, 2^{4k-1})$ 可看作由 $4k$ 行和 $4k-1$ 列构成的矩阵。矩阵元素为“+”和“-”。在实验设计中，这两个元素被称为水平。在 $OA(4k, 2^{4k-1})$ 的任意两列中，水平共有 $(+,+)(+,-)(-,+)(-,-)$ 四种组合，且四种组合的出现次数相同，均为 $n/4$。也就是说，两列中相同的样本个数为 $n/4$。因此，$(4k-1) \times 2$ 个 2 折交叉验证切分 $\mathbb{S}$ 可由上述步骤构造。

由于 $4k$ 个数据子块的大小之差不超过 1，且任意两个切分间的重叠块数为 k，则任意两个切分的测试集大小之差不超过 k，即，$\forall i \neq j$，有 $|\phi_{ij} - n/4| \leqslant k$。　□

例 3.1　本例子给出正则化 7×2 交叉验证的切分集 $\mathbb{S}^b$ 的构造方法。该构造方法将指标集 $\mathcal{I}$ 等分为 $4k = 8$ 块（分别记为 $I_i^{(8)}, i = 1, 2, 3, \ldots, 8$），并使用正交表 $OA(8, 2^7)$ （见表3.1）进行构造。正则化 7×2 交叉验证的切分集 $\mathbb{S}^b$ 见表3.2。当 $n = 400$ 时，$m \times 2$ 交叉验证中 $|\phi_{ij} - n/4|$ 的期望值为 3.98。相比之下，本节所给出的构造方法可将 k 约束为 2。

表 3.1　正交表 $OA(8, 2^7)$

	a	b	ab	c	ac	bc	abc
$1 \leftrightarrow I_1^{(8)}$	+	+	+	+	+	+	+
$2 \leftrightarrow I_2^{(8)}$	-	+	-	+	-	+	-
$3 \leftrightarrow I_3^{(8)}$	+	-	-	+	+	-	-
$4 \leftrightarrow I_4^{(8)}$	-	-	+	+	-	-	+
$5 \leftrightarrow I_5^{(8)}$	+	+	+	-	-	-	-
$6 \leftrightarrow I_6^{(8)}$	-	+	-	-	+	-	+
$7 \leftrightarrow I_7^{(8)}$	+	-	-	-	-	+	+
$8 \leftrightarrow I_8^{(8)}$	-	-	+	-	+	+	-

注记 3.6　根据定理3.2 的构造算法，在数据集 D_n 上，切分集 $\mathbb{S}^b$ 的重复次数 m 的最大值应为 $n-1$，相应的 $OA(4k, 2^{4k-1})$ 为饱和正交表。

注记 3.7　王钰等给出的组块 3×2 交叉验证[8] 是正则化 $m \times 2$ 交叉验证在 $m = 3$ 时的特殊情形。组块 3×2 交叉验证的构造方法也是本节所给出的构造算法基在 $OA(4, 2^3)$ 时的一种特殊情形。

表 3.2 正则化 7×2 交叉验证的切分集 $\mathbb{S}^b$ 示例

切分	$I_i^{(t)}$	$I_i^{(v)}$
$\mathcal{S}_1$	$I_1^{(8)}, I_3^{(8)}, I_5^{(8)}, I_7^{(8)}$	$I_2^{(8)}, I_4^{(8)}, I_6^{(8)}, I_6^{(8)}$
$\mathcal{S}_2$	$I_1^{(8)}, I_2^{(8)}, I_5^{(8)}, I_6^{(8)}$	$I_3^{(8)}, I_4^{(8)}, I_7^{(8)}, I_8^{(8)}$
$\mathcal{S}_3$	$I_1^{(8)}, I_4^{(8)}, I_5^{(8)}, I_8^{(8)}$	$I_2^{(8)}, I_3^{(8)}, I_6^{(8)}, I_7^{(8)}$
$\mathcal{S}_4$	$I_1^{(8)}, I_2^{(8)}, I_3^{(8)}, I_4^{(8)}$	$I_5^{(8)}, I_6^{(8)}, I_7^{(8)}, I_8^{(8)}$
$\mathcal{S}_5$	$I_1^{(8)}, I_3^{(8)}, I_6^{(8)}, I_8^{(8)}$	$I_2^{(8)}, I_4^{(8)}, I_5^{(8)}, I_7^{(8)}$
$\mathcal{S}_6$	$I_1^{(8)}, I_2^{(8)}, I_7^{(8)}, I_8^{(8)}$	$I_3^{(8)}, I_4^{(8)}, I_5^{(8)}, I_6^{(8)}$
$\mathcal{S}_7$	$I_1^{(8)}, I_4^{(8)}, I_6^{(8)}, I_7^{(8)}$	$I_2^{(8)}, I_3^{(8)}, I_5^{(8)}, I_8^{(8)}$

正则化 7×2 交叉验证的切分集可基于正则化 3×2 交叉验证的切分集来构造。构造正则化 3×2 交叉验证的切分集时，数据集被等分为四个子集。四个子集中的每一个可进一步被分为两等分，共形成八个数据子集。这八个数据子集可被用来构造正则化 7×2 交叉验证的切分集。本质上，正则化 7×2 交叉验证的切分集应包含正则化 3×2 交叉验证的切分集。

一般地，当 $4k=2^p$ 时，正则化 $(2^p-1)\times 2$ 交叉验证的切分集 $\mathbb{S}^b$ 可基于正则化 $(2^{p-1}-1)\times 2$ 交叉验证的切分集 $\mathbb{S}^b$ 进行构造。这个构造的方法被称为增量式构造算法，具体步骤如下：

(1) 基于正交表 $OA(2^{p-1}, 2^{2^{p-1}-1})$（其中，$p\geq 3$）来构造正交表 $OA(2^p, 2^{2^p-1})$[38]。具体地，$OA(2^{p-1}, 2^{2^{p-1}-1})$ 对应于一个 Hardmard 矩阵 H。基于 H 所构造的矩阵 $\begin{bmatrix} H & H \\ H & -H \end{bmatrix}$ 仍为一个 Hardmard 矩阵，且该矩阵对应正交表 $OA(2^p, 2^{2^p-1})$。

(2) 对正则化 $2^{p-1}\times 2$ 交叉验证的切分集 $\mathbb{S}^b$ 所用的 2^{p-1} 个数据子块进行切分。具体规则为对于 $j\in\{1,2,\ldots,2^{p-1}\}$，第 j 个数据子块被等分为两个数据子块，分别记为第 j 个数据子块和第 $j+p$ 个数据子块。

(3) 基于正交表 $OA(2^p, 2^{2^p-1})$ 和第（2）步生成的数据子块，应用定理3.2 中的第（1）步，生成正则化 $(2^p-1)\times 2$ 交叉验证的第 $(j+p)$ 个切分。

下例给出从正则化 3×2 交叉验证到正则化 7×2 交叉验证的增量式构造过程。

例 3.2 正则化 3×2 交叉验证的切分集 $\mathbb{S}^b$ 可基于 $OA(4, 2^3)$ 构造（见表3.3）。相应的四个数据子块记为 $(I_1^{(4)}, I_2^{(4)}, I_3^{(4)}, I_4^{(4)})$。表3.1 左上角的 4×3 子块和左下角的 4×3 子块与表3.3 所给的 $OA(4, 2^3)$ 相同。将四个子块 $(I_1^{(4)}, I_2^{(4)}, I_3^{(4)}, I_4^{(4)})$ 依如下规则切分成八个子块 $(I_1^{(8)}, I_2^{(4)}, I_3^{(4)}, \ldots, I_8^{(8)})$。

$$I_1^{(4)} \leftrightarrow I_1^{(8)}, I_5^{(8)} \qquad I_2^{(4)} \leftrightarrow I_2^{(8)}, I_6^{(8)}$$

$$I_3^{(4)} \leftrightarrow I_3^{(8)}, I_7^{(8)} \qquad I_4^{(4)} \leftrightarrow I_4^{(8)}, I_8^{(8)}。$$

正则化 7×2 交叉验证的切分集 $\mathbb{S}^b$ 中切分 $\mathcal{S}_4, \mathcal{S}_5, \mathcal{S}_6, \mathcal{S}_7$ 可使用 $OA(8, 2^7)$ 中的最后四列构造。表3.4 中对比了正则化 3×2 交叉验证和正则化 7×2 交叉验证的切分集。可见，两个切分集中的前三个切分完全相同。

表 3.3　正交表 $OA(4, 2^3)$

	a	b	ab
$1 \leftrightarrow I_1^{(4)}$	+	+	+
$2 \leftrightarrow I_2^{(4)}$	-	+	-
$3 \leftrightarrow I_3^{(4)}$	+	-	-
$4 \leftrightarrow I_4^{(4)}$	-	-	+

表 3.4　正则化 3×2 交叉验证和正则化 7×2 交叉验证切分集的比照

Partition	$I_i^{(t)}$		$I_i^{(v)}$	
	3×2	7×2	3×2	7×2
$\mathcal{S}_1$	$I_1^{(4)}, I_3^{(4)}$	$I_1^{(8)}, I_3^{(8)}, I_5^{(8)}, I_7^{(8)}$	$I_2^{(4)}, I_4^{(4)}$	$I_2^{(8)}, I_4^{(8)}, I_6^{(8)}, I_8^{(8)}$
$\mathcal{S}_2$	$I_1^{(4)}, I_2^{(4)}$	$I_1^{(8)}, I_2^{(8)}, I_5^{(8)}, I_6^{(8)}$	$I_3^{(4)}, I_4^{(4)}$	$I_3^{(8)}, I_4^{(8)}, I_7^{(8)}, I_8^{(8)}$
$\mathcal{S}_3$	$I_1^{(4)}, I_4^{(4)}$	$I_1^{(8)}, I_4^{(8)}, I_5^{(8)}, I_8^{(8)}$	$I_2^{(4)}, I_3^{(4)}$	$I_2^{(8)}, I_3^{(8)}, I_6^{(8)}, I_7^{(8)}$
$\mathcal{S}_4$		$I_1^{(8)}, I_2^{(8)}, I_3^{(8)}, I_4^{(8)}$		$I_5^{(8)}, I_6^{(8)}, I_7^{(8)}, I_8^{(8)}$
$\mathcal{S}_5$		$I_1^{(8)}, I_3^{(8)}, I_6^{(8)}, I_8^{(8)}$		$I_2^{(8)}, I_4^{(8)}, I_5^{(8)}, I_7^{(8)}$
$\mathcal{S}_6$		$I_1^{(8)}, I_2^{(8)}, I_7^{(8)}, I_8^{(8)}$		$I_3^{(8)}, I_4^{(8)}, I_5^{(8)}, I_6^{(8)}$
$\mathcal{S}_7$		$I_1^{(8)}, I_4^{(8)}, I_6^{(8)}, I_7^{(8)}$		$I_2^{(8)}, I_3^{(8)}, I_5^{(8)}, I_8^{(8)}$

推论3.2表明泛化误差的正则化 $m \times 2$ 交叉验证估计的方差是关于 m 的递减函数。因此，在实际应用中，基于上述增量式构造算法，逐渐增加 m 的值以得到更为稳定的估计，是非常有用的。

3.4　重复次数 m 的选取

在实际应用中，给定一种选取 m 的方法是非常必要的。定理3.1 中的式（3.26）表明 $E[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}|\mathbf{\Phi}^*)]] = \frac{1}{2m}\sigma^2(1+\rho_1) + \frac{m-1}{m}\sigma^2\rho_2$。随着 m 的增加，除方差逐步递减外，方差约减的量也逐步递减。定义如下的方差约减率。

$$\begin{aligned} & \frac{E[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}|\Phi^*)]] - E[\mathrm{Var}[\hat{\mu}_{(m+1)\times 2}(\mathbb{S}|\Phi^*)]]}{E[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}|\Phi^*)]]} \\ = & \frac{1+\rho_1-2\rho_2}{(m+1)(1+\rho_1)+2(m^2-1)\rho_2}。 \end{aligned} \tag{3.31}$$

若约减率较小，如小于 α（5% 或 1%），则增加更多切分，并不会带来方差的显著变化。基于这个想法，为重复次数 m 选取合适的值。不过，式（3.31）含有未知参数 ρ_1 和 ρ_2。它们的值与数据集大小、所用算法等都有关，与重复次数 m 无关。王钰等给出的关于 ρ_1 和 ρ_2 的模拟实验[8] 表明 ρ_1 和 ρ_2 的取值分布在 $0 < \rho_1, \rho_2 < 1/2$ 的范围内。

设式（3.31）的方差约减率在区间 $0 < \rho_1, \rho_2 < 1/2$ 内的平均值为方差的平均约减率，并记为 ARRV。本节推荐使用 ARRV $< \alpha$ 来为 m 选取适当的值，即：

$$\text{ARRV} \triangleq 4 \int_0^{1/2} \int_0^{1/2} \frac{1 + \rho_1 - 2\rho_2}{(m+1)(1+\rho_1) + 2(m^2-1)\rho_2} d\rho_1 d\rho_2 < \alpha\%。 \tag{3.32}$$

表3.5 给出了 m 值与方差平均约减率的关系。从该表可知，王钰等给出的组块 3×2 交叉验证的 ARRV 小于 10%。若研究者希望方差平均约减率小于 $\alpha = 5\%$，则 m 的值需要大于 5。这可能是 5×2 交叉验证被很多研究者用于算法比较任务的主要原因之一[1,6,7]。另外，若方差约减率小于 $\alpha = 1\%$，则 m 不能小于 16。

表 3.5 方差平均约减率 ARRV

m	ARRV	$\alpha\%$	交叉验证方法
2	0.1552		
3	0.0984	$< 10\%$	正则化 3×2 交叉验证
4	0.0688		
5	0.0516		
6	0.0404	$< 5\%$	正则化 6×2 交叉验证
7	0.0324		
11	0.0168		
15	0.0105		
16	0.0095	$< 1\%$	正则化 16×2 交叉验证

3.5 模拟实验

本节基于模拟实验来回答如下四个研究问题。

研究问题一 ：函数 $f(x)$ 中的条件 $\omega + \gamma - 2\tau$ 是否大于 0?

研究问题二 ：基于 $x = n/4$ 处的参数值 σ^2, ω, γ and τ，$f(x)$ 及 $g(x)$ 被近似的程度有多好?

研究问题三 ：泛化误差的 $m \times 2$ 交叉验证估计与正则化 $m \times 2$ 交叉验证估计的方差之间的差有多大?

3.5.1　实验设置

本节的模拟实验包括回归和分类两种任务。实验数据及学习算法设置如下

- 模拟回归数据（简记为 SREG）：特征向量 x_i 中含 p 个独立的特征，均来自标准正态分布。响应值为 $y_i = \sqrt{3/p}\sum_{k=1}^{p} x_{ik} + \varepsilon_i$，其中，$\varepsilon_i \sim N(0,1)$。该数据集的设置来自于 Nadeau 等的工作[16]。令 $p < n$。模拟实验使用普通线性回归（记为 lm），岭回归（记为 rid）和 lasso 回归（记为 lso）来估计泛化误差，并使用平方损失。数据集大小和特征维度设置为 $n = 1000$ 和 $p = 100$。
- 模拟分类数据（简记为 SCLA）：数据集 $D_n = (x_i, y_i)_{i=1}^{n}$ 中，响应变量的先验分布为 $P(Y = 1) = P(Y = 0) = \frac{1}{2}$。当响应变量 Y 为 0 时，特征的分布为 $X|Y = 0 \sim N(0, I)$。当 Y 为 1 时，前 10% 的特征服从均值为 0.5 且方差为 1 的正态分布，其余特征从标准正态分布中抽取。模拟实验使用支持向量机（记为 svm）和 k 近邻（记为 knn）作为分类器。k 近邻算法使用 $k = 5$ 和三角核函数。实验中使用 0-1 损失函数。该数据设置来自 Tibshirani 等的工作[94]。数据集大小 n 和特征个数 p 分别设为 1000 和 20。

下面使用“数据集标记 + 算法标记 + 数据集大小 + 特征个数”来标记每种实验配置。例如，“SREG+lm+1000+20”表示实验中使用数据集大小为 1000、特征个数为 20 的 SREG 作为数据集、及普通线性回归作为机器学习算法。

3.5.2　问题一的模拟实验

该实验的目的在于验证函数 $f(x)$ 的二次项系数 $\omega + \gamma - 2\tau$ 是否大于 0。实验结果在表3.6中给出。结果表明在所有实验配置上，条件 $\omega + \gamma - 2\tau > 0$ 均成立。

表 3.6　模拟数据集上函数 $f(x)$ 的系数

实验配置	ρ_2	$\omega + \gamma - 2\tau$	$\sigma^2 - \omega - n\gamma + n\tau$	γ
SREG+lm+1000+100	0.4433	0.0023	3.7212	0.0029
SREG+rid+1000+100	0.442	0.0022	3.7782	0.0028
SREG+lso+1000+100	0.2822	0.0052	51.1965	-0.0005
SCLA+svm+1000+20	0.4308	0.0001	-0.127	0.0002
SCLA+knn+1000+20	0.3078	0.0001	-0.1592	0.0002

表3.6也给出相关系数 ρ_2 的值。可见，$0 < \rho_2 < 0.5$。

3.5.3　问题二的模拟实验

图3.2和图3.3分别给出了函数 $f(x)$ 和 $g(x)$ 在模拟实验上的函数图像以及相应的近似图像。近似图像是基于 $x = n/4$ 处参数 σ^2，ω，τ 及 γ 的值绘制所得。可见，

在 $x = n/4$ 附近，近似图像可以较好地逼近原函数图像。此外，图3.2 和图3.3 表明 $f(x)$ 为下凸函数且 $g(x)$ 为对称下凸函数。这与引理3.1 和引理3.2 吻合。

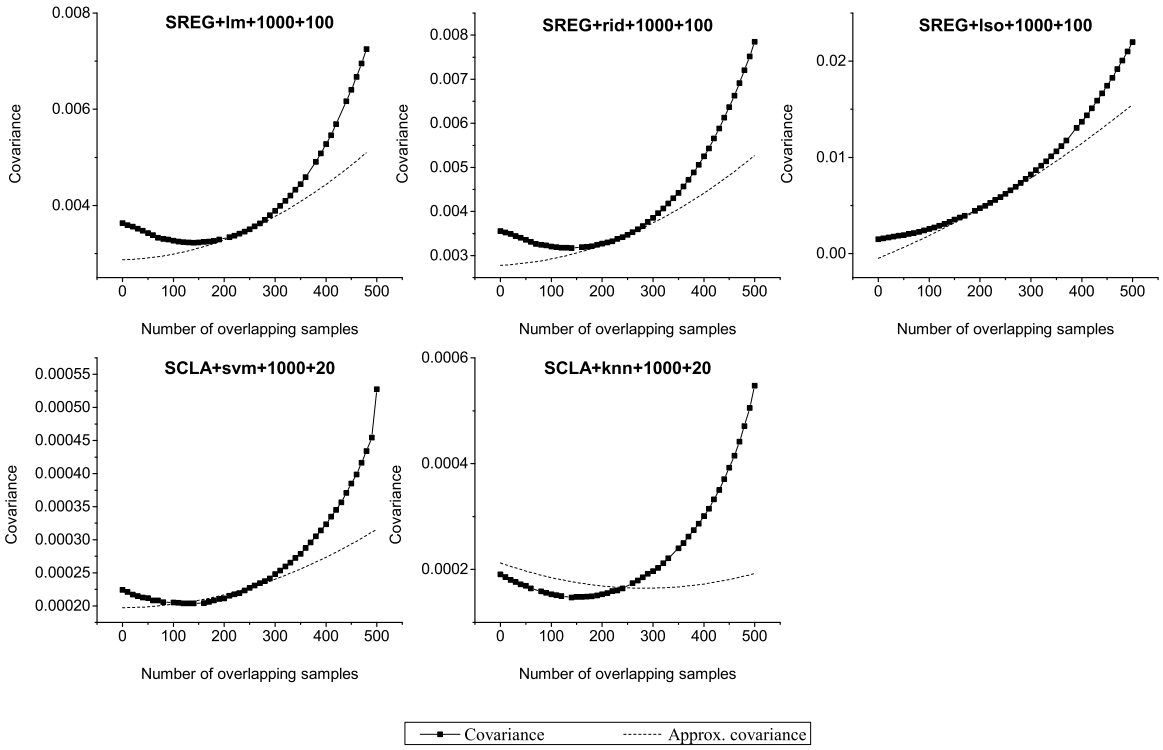

图 3.2 模拟数据集上函数 $f(x)$ 的图像

3.5.4 问题三的模拟实验

本实验的目的在于验证泛化误差的正则化 $m \times 2$ 交叉验证估计的方差不大于随机切分情形下 $m \times 2$ 交叉验证估计的方差。基于给定的实验配置，从数据总体中随机抽取 10000 个数据集。针对正则化 $m \times 2$ 交叉验证和 $m \times 2$ 交叉验证，分别抽取 1000 个切分集。对于每种交叉验证，共可获得 1000 万个泛化误差的估计。采用这些估计的样本方差估计来近似真实方差。在模拟实验中，设置 $m = 3, 5, 7, 9$。实验结果在表3.7中给出[2]。

为体现正则化 $m \times 2$ 交叉验证在方差上的优势，定义如下的方差约减率（记为 RR）。

$$\mathrm{RR} = \frac{E_{\Phi}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}|\Phi)]] - E_{\Phi^b}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b)|\Phi^b]]}{E_{\Phi}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})|\Phi]]} \cdot 100\%。 \tag{3.33}$$

在所有实验配置上，两种交叉验证估计的方差约减率在表3.8 中给出。

从表3.7和表3.8 中，可得如下结论：

(1) 对于所有的实验配置，$E_{\Phi}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S})|\Phi]] > E_{\Phi^b}[\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^b)|\Phi^b]]$ 均成

[2]表3.7中名为“数量级”的列表明每行实验结果应乘的数量级。

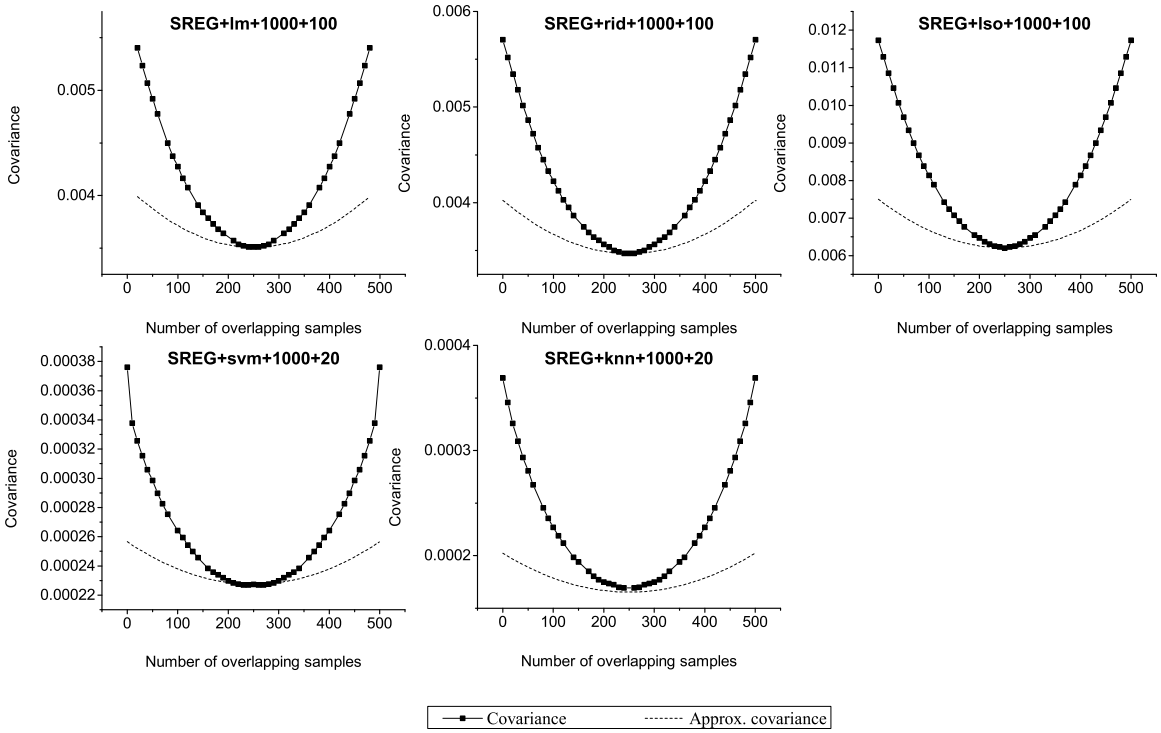

图 3.3　模拟数据集上函数 $g(x)$ 的图像

表 3.7　模拟数据上正则化 $m \times 2$ 交叉验证和 $m \times 2$ 交叉验证估计的方差比较

实验配置	数量级	m=3	m=5	m=7	m=9
正则化 $m \times 2$ 交叉验证					
SREG+lm+1000+100	10^{-4}	7.5674	4.5481	3.2458	2.5202
SREG+rid+1000+100	10^{-4}	7.4624	4.4860	3.2012	2.4850
SREG+lso+1000+100	10^{-3}	1.8099	1.0841	0.7731	0.5996
SCLA+svm+1000+20	10^{-5}	4.9337	2.9554	2.1120	1.6391
SCLA+knn+1000+20	10^{-5}	6.6614	3.9830	2.8371	2.2051
$m \times 2$ 交叉验证					
SREG+lm+1000+100	10^{-4}	7.5948	4.5602	3.2599	2.5355
SREG+rid+1000+100	10^{-4}	7.4901	4.4958	3.2149	2.4998
SREG+lso+1000+100	10^{-3}	1.8092	1.0852	0.7754	0.6036
SCLA+svm+1000+20	10^{-5}	4.9454	2.9658	2.1196	1.6493
SCLA+knn+1000+20	10^{-5}	6.6717	4.0049	2.8633	2.2228

立。也就是说，$m \times 2$ 交叉验证估计的方差均大于正则化 $m \times 2$ 交叉验证估计的方差。

(2) 当 m 从 3 增至 9 时，方差约减率也随之增加。这表明，当 m 增加时，正则化 $m \times 2$ 交叉验证可得到越来越稳定的方差估计。

表 3.8 模拟数据集上正则化 $m \times 2$ 交叉验证与 $m \times 2$ 交叉验证间的方差约减率

实验配置	方差约减率（%）			
	m=3	m=5	m=7	m=9
SREG+lm+1000+100	0.36	0.27	0.43	0.60
SREG+rid+1000+100	0.37	0.22	0.43	0.59
SREG+lso+1000+100	-0.04	0.10	0.30	0.66
SCLA+svm+1000+20	0.23	0.35	0.36	0.62
SCLA+knn+1000+20	0.15	0.55	0.91	0.80

3.6 真实数据集上的实验结果

本节在多个真实数据集上比较正则化 $m \times 2$ 交叉验证估计和 $m \times 2$ 交叉验证估计的方差。所有真实数据集均来自 UCI 数据集。这些数据集的基本信息如下：

(1) 字符识别数据集（LETTER）：用于识别一张图片上的字符。数据集含 16 个特征和 26 个类别。在使用该数据集时，采用的设置与 Bengio 等所用设置相同[23]。具体设置为：将类别 A-M 合并成第一类，将余下的类别合并成第二类。进而，该数据集被转化为一个二分类数据集。在该数据集上，分别采用支持向量机（svm）和 k 近邻（knn）作为机器学习算法。

(2) 酒质数据集（WQ）：用于预测酒的质量。本节实验仅使用白酒质量的数据集，具体有 4898 条样例和 11 个特征。响应变量为酒质等级。本节将其看作一个回归任务，并使用普通线性回归（lm）、岭回归（rid）和 lasso 回归（lso）作为机器学习算法。

(3) 空气质量数据集（AQ）：用于预测意大利的空气质量。数据集的响应变量为绝对湿度。样例中的日期和时间戳被移除。特征非甲烷碳氢化合物浓度（non-methane hydrocarbon concentration，NMHC）中包含缺失值，因此，该特征也被移除。此外，实验中移除了所有包含缺失值的样例。最终的数据集含 6941 条样例和 11 个特征。实验分别使用线性回归（lm）、岭回归（rid）和 lasso 回归（lso）作为机器学习算法。

(4) Connect-four 数据集（CON）：关于 connect-four 游戏的数据集。数据集含 67557 条样例、2 个类别以及 42 个特征。本节将其看作分类问题，并使用支持向量机（svm）和 k 近邻（knn）作为机器学习算法。

(5) 人口普查数据集（ADULT）：该数据集抽取自一个人口普查数据库中，含 32561 条样例和 14 个特征。移除掉含有含有缺失值的样例后，该数据集含 30162 条

观测。该数据集旨在预测一位公民的年收入是否超过 $50000。该任务可看作二类分类问题，并使用支持向量机（svm）和 k 近邻（knn）作为机器学习算法。

在真实数据上，方差计算的其它设置与在模拟数据上的设置相同。每组实验产生 1000 万次重复，含 1 万次数据集和 1000 次切分集。表3.9包含了正则化 $m \times 2$ 交叉验证估计和 $m \times 2$ 交叉验证估计的方差，相应的方差约减率在表3.10 中给出。

表 3.9 UCI 数据集上的方差比较

实验配置	数量级	m=3	m=5	m=7	m=9
正则化 $m \times 2$ 交叉验证					
LETTER+svm+500+256	10^{-5}	8.1657	4.8959	3.5017	2.7123
LETTER+knn+500+256	10^{-4}	1.0508	0.6271	0.4469	0.3465
WQ+lm+100+11	10^{-2}	1.1509	0.68899	0.48625	0.37031
WQ+rid+100+11	10^{-2}	1.4007	0.82894	0.55728	0.42205
WQ+lso+100+11	10^{-3}	3.6329	2.1599	1.5188	1.1654
AQ+lm+200+11	10^{-7}	7.4631	4.4451	3.1432	2.4231
AQ+rid+200+11	10^{-8}	3.7909	2.2534	1.5905	1.2153
AQ+lso+200+11	10^{-7}	8.2932	4.9378	3.4873	2.6908
CON+svm+500+42	10^{-4}	3.0035	1.6996	1.2739	0.99936
CON+knn+500+42	10^{-4}	1.4920	0.89203	0.63534	0.49354
ADULT+svm+500+14	10^{-5}	5.8201	3.5065	2.5294	1.9538
ADULT+knn+500+14	10^{-5}	7.7733	4.6526	3.3153	2.5719
$m \times 2$ 交叉验证					
LETTER+svm+500+256	10^{-5}	8.2011	4.9502	3.5278	2.7458
LETTER+knn+500+256	10^{-4}	1.0540	0.6339	0.4521	0.3518
WQ+lm+100+11	10^{-2}	1.1747	0.70407	0.49876	0.38629
WQ+rid+100+11	10^{-2}	1.4065	0.84562	0.60355	0.47136
WQ+lso+100+11	10^{-3}	3.6832	2.2152	1.5798	1.2307
AQ+lm+200+11	10^{-7}	7.5392	4.5222	3.2349	2.5162
AQ+rid+200+11	10^{-8}	3.8234	2.2979	1.6411	1.2772
AQ+lso+200+11	10^{-7}	8.3728	5.0326	3.5938	2.7991
CON+svm+500+42	10^{-4}	3.0443	1.8274	1.2922	1.0224
CON+knn+500+42	10^{-4}	1.5024	0.90046	0.64282	0.50065
ADULT+svm+500+14	10^{-5}	5.8495	3.6059	2.5599	1.9999
ADULT+knn+500+14	10^{-5}	7.8270	4.7058	3.3588	2.6101

表3.9 和表3.10 表明，所有实验配置上，正则化 $m \times 2$ 交叉验证估计的方差小于 $m \times 2$ 交叉验证估计的方差。当 m 增大时，正则化 $m \times 2$ 交叉验证估计的方差和 $m \times 2$ 交叉验证估计的方差均随之减少。在大多数配置上，方差约减率随之增加。

表 3.10 UCI 数据集上的方差约减率

实验配置	方差约减率（%）			
	m=3	m=5	m=7	m=9
LETTER+svm+500+256	0.43	1.10	0.74	1.22
LETTER+knn+500+256	0.31	1.07	1.14	1.50
WQ+lm+100+11	2.03	2.14	2.51	4.14
WQ+rid+100+11	0.41	1.97	7.67	10.46
WQ+lso+100+11	1.36	2.49	3.86	5.30
AQ+lm+200+11	1.01	1.70	2.83	3.70
AQ+rid+200+11	0.85	1.94	3.08	4.85
AQ+lso+200+11	0.95	1.88	2.96	3.87
CON+svm+500+42	1.34	6.99	1.41	2.25
CON+knn+500+42	0.69	0.93	1.16	1.42
ADULT+svm+500+14	0.50	2.76	1.19	2.30
ADULT+knn+500+14	0.69	1.13	1.29	1.46

3.7 附录：条件 $\omega+\gamma-2\tau$ 的理论证明

本节给出平方损失下条件 $\omega+\gamma>2\tau$ 的证明。该证明包含了均值回归、一元线性回归和多元线性回归三种情形。设 $\mathcal{S}_1=(I_1^{(t)},I_1^{(v)})$ 和 $\mathcal{S}_2=(I_2^{(t)},I_2^{(v)})$ 为指标集 $\mathcal{I}=\{1,\ldots,n\}$ 上的两个切分。相应的训练集和验证集分别为 $(D_1^{(t)},D_1^{(v)})$ 和 $(D_2^{(t)},D_2^{(v)})$。将损失函数 $L(\mathcal{A}(D^{(t)}),z_j)$ 简化为 $L(\hat{y}_{I^{(t)},j},y_j)$，其中，$y_j$ 为样例 z_j 的响应变量。基于机器学习算法 $\mathcal{A}$ 及训练集 $D^{(t)}=\{z_i|i\in I^{(t)}\}$，可得样例 z_j 的预测值，记为 $\hat{y}_{I^{(t)},j}$。数据集 D_n 可分解为如下四部分。

- $A=\{a|a\in I_1^{(t)}\cap I_2^{(t)}\}$ 为训练集 $D_1^{(t)}$ 和 $D_2^{(t)}$ 中相同的样例所对应的指标集。
- $B=\{b|b\in I_1^{(t)}\backslash A\}$ 为仅在训练集 $D_1^{(t)}$ 中出现但未在 $D_2^{(t)}$ 中出现的样例所对应的指标集。
- $C=\{c|c\in I_2^{(t)}\backslash A\}$ 为仅在训练集 $D_2^{(t)}$ 中出现但未在 $D_1^{(t)}$ 中出现的样例所对应的指标集。
- $D=\{d|d\in I_1^{(v)}\cap I_2^{(v)}\}$ 为包含在 D_n 中，但未在在训练集 $D_1^{(t)}$ 和 $D_2^{(t)}$ 中均出现的样例所对应的指标集。

基于平方损失，可知参数 ω，γ 和 τ 的表达式如下：

- $\omega=\mathrm{Cov}[(\hat{y}_{A\cup B,d}-y_d)^2,(\hat{y}_{A\cup C,d'}-y_{d'})^2]$, $\forall d,d'\in D$ 且 $d\neq d'$；
- $\gamma=\mathrm{Cov}[(\hat{y}_{A\cup B,c}-y_c)^2,(\hat{y}_{A\cup C,b}-y_b)^2]$, $\forall b\in B,c\in C$；
- $\tau=\mathrm{Cov}[(\hat{y}_{A\cup B,c}-y_c)^2,(\hat{y}_{A\cup C,d}-y_d)^2]$, $\forall c\in C,d\in D$，或 $\tau=\mathrm{Cov}[(\hat{y}_{A\cup B,d'}-$

$y_{d'})^2, (\hat{y}_{A\cup C,b} - y_b)^2], \forall b \in B, d' \in D$。

以下三小节分别给出三种回归情形下条件 $\omega + \gamma - 2\tau$ 的理论证明。

3.7.1　均值回归下 $\omega + \gamma > 2\tau$ 的证明

对于均值回归，算法仅使用数据集所含样例的 n 个响应值 $y_1, y_2, \ldots, y_n$。这些响应值均独立同分布地从一未知分布中抽取。假定该分布的均值和方差分别为 μ 和 ψ^2。均值回归使用训练集中所有样例的响应值的样本均值来预测单个测试样例的响应值，即：$\hat{y}_{A\cup B,d} = \bar{y}_{A\cup B}$，其中，$\bar{y}_{A\cup B} = 2\sum_{i\in A\cup B} y_i/n$。

协方差参数 ω，γ 和 τ 具有如下的分解形式：

$$\begin{aligned}\omega &= \mathrm{Cov}[\bar{y}^2_{A\cup B}, \bar{y}^2_{A\cup C}] - 2\mathrm{Cov}[\bar{y}^2_{A\cup B}, 2\bar{y}_{A\cup C}y_{d'}] + 2\mathrm{Cov}[\bar{y}^2_{A\cup B}, y^2_{d'}] \\ &\quad +\mathrm{Cov}[2\bar{y}_{A\cup B}y_d, 2\bar{y}_{A\cup C}y_{d'}] - 2\mathrm{Cov}[2\bar{y}_{A\cup B}y_d, y^2_{d'}] + \mathrm{Cov}[y^2_d, y^2_{d'}], \qquad (3.34)\end{aligned}$$

$$\begin{aligned}\gamma &= \mathrm{Cov}[\bar{y}^2_{A\cup B}, \bar{y}^2_{A\cup C}] - 2\mathrm{Cov}[\bar{y}^2_{A\cup B}, \bar{y}_{A\cup C}y_b] - 2\mathrm{Cov}[\bar{y}^2_{A\cup C}, \bar{y}_{A\cup B}y_c] \\ &\quad +\mathrm{Cov}[\bar{y}^2_{A\cup B}, y^2_b] + Cov[\bar{y}^2_{A\cup C}, y^2_c] + 4Cov[\bar{y}_{A\cup B}y_c, \bar{y}_{A\cup C}y_b] \\ &\quad -2\mathrm{Cov}[\bar{y}_{A\cup B}y_c, y^2_b] - 2\mathrm{Cov}[\bar{y}_{A\cup C}y_b, y^2_c] + \mathrm{Cov}[y^2_b, y^2_c], \qquad (3.35)\end{aligned}$$

$$\begin{aligned}\tau &= \mathrm{Cov}[\bar{y}^2_{A\cup B}, \bar{y}^2_{A\cup C}] - 2\mathrm{Cov}[\bar{y}^2_{A\cup B}, \bar{y}_{A\cup C}y_b] - \mathrm{Cov}[\bar{y}^2_{A\cup C}, 2\bar{y}_{A\cup B}y_d] \\ &\quad +\mathrm{Cov}[\bar{y}^2_{A\cup B}, y^2_b] + Cov[\bar{y}^2_{A\cup C}, y^2_d] + 4\mathrm{Cov}[\bar{y}_{A\cup B}y_d, \bar{y}_{A\cup C}y_b] \\ &\quad -2\mathrm{Cov}[\bar{y}_{A\cup B}y_d, y^2_b] - 2\mathrm{Cov}[\bar{y}_{A\cup C}y_b, y^2_d] + \mathrm{Cov}[y^2_b, y^2_d]。 \qquad (3.36)\end{aligned}$$

进而，可得：

$$\begin{aligned}\omega + \gamma - 2\tau &= 4\mathrm{Cov}[\bar{y}_{A\cup B}y_d, \bar{y}_{A\cup C}y_{d'}] + 4\mathrm{Cov}(\bar{y}_{A\cup B}y_c, \bar{y}_{A\cup C}y_b) - 8\mathrm{Cov}[\bar{y}_{A\cup B}y_d, \bar{y}_{A\cup C}y_b] \\ &= \frac{16}{n^2}\left\{\mathrm{Cov}\left[y_c\sum_{b\in B}y_b, y_b\sum_{c\in C}y_c\right] - 2\mathrm{Cov}\left[y_d\sum_{b\in B}y_b, y_b\sum_{c\in C}y_c\right]\right\} \\ &= \frac{16}{n^2}\left\{\sum_{b'\in B}\sum_{c'\in C}\mathrm{Cov}\left[y_cy_{b'}, y_by_{c'}\right] - 2\sum_{b'\in B}\sum_{c'\in C}\mathrm{Cov}\left[y_dy_{b'}, y_by_{c'}\right]\right\}。 \qquad (3.37)\end{aligned}$$

当 $b' \neq b$ 且 $c' \neq c$ 时，有 $\mathrm{Cov}[y_cy_{b'}, y_by_{c'}] = 0$ 和 $\mathrm{Cov}[y_dy_{b'}, y_by_{c'}] = 0$ 成立。因此，可得：

$$\begin{aligned}\omega + \gamma - 2\tau &= \frac{16}{n^2}\left\{\sum_{c'\in C}\mathrm{Cov}[y_cy_b, y_by_{c'}] + \sum_{b'\in B}\mathrm{Cov}[y_cy_{b'}, y_by_c] - \mathrm{Var}[y_by_c] - 2\sum_{c'\in C}\mathrm{Cov}[y_dy_b, y_by_{c'}]\right\} \\ &= \frac{16}{n^2}\left\{2\sum_{c'\in C}\mathrm{Cov}[y_cy_b, y_by_{c'}] - 2\sum_{c'\in C}\mathrm{Cov}[y_dy_b, y_by_{c'}] - \mathrm{Var}[y_by_c]\right\}\end{aligned}$$

$$= \frac{16}{n^2}\{\mathrm{Var}[y_b y_c] - 2\mathrm{Cov}[y_d y_b, y_b y_c]\}。 \tag{3.38}$$

其中，$\sum_{c'\in C}\mathrm{Cov}[y_c y_b, y_b y_{c'}] = \sum_{b'\in B}\mathrm{Cov}[y_c y_{b'}, y_b y_c]$。

因 $E[y_i] = \mu$ 且 $\mathrm{Var}[y_i] = \psi^2$，有 $\mathrm{Var}[y_b y_c] = \psi^4 + 2\mu^2\psi^2$ 和 $\mathrm{Cov}[y_d y_b, y_b y_c] = \mu^2\psi^2$。进而，可得：

$$\omega + \gamma - 2\tau = \frac{16}{n^2}\psi^4 > 0。 \tag{3.39}$$

因此，在均值回归时，条件 $\omega + \gamma > 2\tau$ 成立。

3.7.2 一元线性回归下 $\omega + \gamma > 2\tau$ 的证明

本节考虑一元线性回归下条件 $\omega + \gamma > 2\tau$ 的证明。不失一般性，假定一元线性回归中不带截距项，即：

$$y = \beta x + \epsilon。 \tag{3.40}$$

其中，$\epsilon \sim N(0, \psi^2)$ 且 x 的均值和方差分别为 ν 和 ϕ^2。

根据式（3.46），可得：

- $\omega = 2\psi^4 E\left[\frac{(\sum_{a\in A} x_a^2)^2 x_d^2 x_{d'}^2}{(\sum_{a\in A} x_a^2 + \sum_{b\in B} x_b^2)^2(\sum_{a\in A} x_a^2 + \sum_{c\in C} x_c^2)^2}\right]$, $\forall d, d' \in D$ 且 $d \neq d'$；
- $\gamma = 2\psi^4 E\left[\frac{(\sum_{a\in A} x_a^2)^2 x_b^2 x_c^2}{(\sum_{a\in A} x_a^2 + \sum_{b\in B} x_b^2)^2(\sum_{a\in A} x_a^2 + \sum_{c\in C} x_c^2)^2}\right]$, $\forall b \in B$ 且 $\forall c \in C$；
- $\tau = 2\psi^4 E\left[\frac{(\sum_{a\in A} x_a^2)^2 x_b^2 x_d^2}{(\sum_{a\in A} x_a^2 + \sum_{b\in B} x_b^2)^2(\sum_{a\in A} x_a^2 + \sum_{c\in C} x_c^2)^2}\right]$, $\forall d \in D$ 且 $\forall b \in B$，或 $\tau = 2\psi^4 E\left[\frac{(\sum_{a\in A} x_a^2)^2 x_c^2 x_{d'}^2}{(\sum_{a\in A} x_a^2 + \sum_{b\in B} x_b^2)^2(\sum_{a\in A} x_a^2 + \sum_{c\in C} x_c^2)^2}\right]$, $\forall d' \in D$ 且 $\forall c \in C$。

进而，有

$$\begin{aligned}
\omega + \gamma - 2\tau &= 2\psi^4 E\left[\frac{(\sum_{a\in A} x_a^2)^2(x_b^2 - x_{d'}^2)(x_c^2 - x_d^2)}{(\sum_{a\in A} x_a^2 + \sum_{b\in B} x_b^2)^2(\sum_{a\in A} x_a^2 + \sum_{c\in C} x_c^2)^2}\right] \\
&= 2\psi^4 E\left[\frac{(\sum_{a\in A} x_a^2)^2(x_b^2 - x_{d'}^2)(x_c^2 - x_d^2)}{(\sum_{a\in A} x_a^2 + \sum_{b'\neq b} x_{b'}^2 + x_b^2)^2(\sum_{a\in A} x_a^2 + \sum_{c'\neq c} x_{c'}^2 + x_c^2)^2}\right] \\
&= 2\psi^4 E\left[\frac{M^2(x_b^2 - x_{d'}^2)(x_c^2 - x_d^2)}{(M + K + x_b^2)^2(M + H + x_c^2)^2}\right] \\
&= 2\psi^4 E\left[\frac{M^2(x_b^2 - \nu^2 - \phi^2)(x_c^2 - \nu^2 - \phi^2)}{(M + K + x_b^2)^2(M + H + x_c^2)^2}\right]。
\end{aligned} \tag{3.41}$$

其中，期望取自所有 $\{x_i | i \in A \cup B \cup C\}$，且有 $M = \sum_{a\in A} x_a^2$，$K = \sum_{b'\neq b} x_{b'}^2$ 及 $H = \sum_{c'\neq c} x_{c'}^2$。

因此，式（3.41）可进一步表达为：

$$\begin{aligned}
\omega + \gamma - 2\tau &= 2\psi^4 E_{x_a, \forall a\in A}\left[M^2 E_{x_b, \forall b\in B}\left[\frac{x_b^2 - \nu^2 - \phi^2}{(M + K + x_b^2)^2} | x_a\right]\right. \\
&\quad \left. \cdot E_{x_c, \forall c\in C}\left[\frac{x_c^2 - \nu^2 - \phi^2}{(M + H + x_c^2)^2} | x_a\right]\right]。
\end{aligned} \tag{3.42}$$

易得：

$$E_{x_b,\forall b\in B}\left[\frac{x_b^2-\nu^2-\phi^2}{(M+K+x_b^2)^2}|x_a\right]=E_{x_c,\forall c\in C}\left[\frac{x_c^2-\nu^2-\phi^2}{(M+H+x_c^2)^2}|x_a\right]。\tag{3.43}$$

由此可得：

$$\omega+\gamma-2\tau=2\psi^4E_{x_a,\forall a\in A}\left[M^2E^2_{x_b,\forall b\in B}(\frac{x_b^2-\nu^2-\phi^2}{(M+K+x_b^2)^2}|x_a)\right]>0。\tag{3.44}$$

因此，对于一元线性回归，条件 $\omega+\gamma-2\tau>0$ 仍成立。

3.7.3　多元线性回归下 $\omega+\gamma>2\tau$ 的证明

对于多元线性回归，本节引入 Markatou 等给出的关于平方损失下协方差函数的一个重要的表达式[20]。具体如下：

假定数据集 $D_n=(\boldsymbol{x}_i,y_i)_{i=1}^n$ 中含 n 个独立同分布的样本。其中，y_i 为响应变量，$\boldsymbol{x}_i$ 为大小为 p 的特征向量，即：$\boldsymbol{x}_i=(x_{i1},\ldots,x_{ip})^\top$。设 $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_p)^\top$ 为系数向量。多元线性回归有如下形式。

$$\mathbf{Y}=\mathbf{X}\boldsymbol{\beta}+\boldsymbol{\epsilon}。\tag{3.45}$$

其中，$\mathbf{Y}=(y_1,\ldots,y_n)^\top$ 为响应向量，$\mathbf{X}=\begin{pmatrix}\boldsymbol{x}_1^\top\\ \vdots\\ \boldsymbol{x}_n^\top\end{pmatrix}$ 为设计矩阵，$\boldsymbol{\epsilon}=(\epsilon_1,\ldots,\epsilon_n)$ 为噪声向量。$\forall i$，$E\epsilon_i=0, Var(\epsilon_i)=\psi^2$。在 $\boldsymbol{\epsilon}$ 中，$\forall i,j$ 有 ϵ_i 和 ϵ_j 均独立同分布。当 $\mathbf{X}$ 中 $\boldsymbol{x}_i$ 为随机向量时，多元线性回归（式（3.45））常将 $\mathbf{X}$ 作为条件来进行分析。

设 $\mathcal{S}=(I^{(t)},I^{(v)})$ 为一个切分，相应的设计矩阵 $\boldsymbol{X}_{I^{(t)}}$ 由样例 $\{\boldsymbol{x}_i|i\in I^{(t)}\}$ 组成。设 $\hat{y}_{I^{(t)},i}$ 为切分 $\mathcal{S}$ 对应的训练集上 y_i 的预测值。基于切分 $\mathcal{S}_1$ 和 $\mathcal{S}_2$，以及测试样例 $(\boldsymbol{x}_i,y_i)$ 和 $(\boldsymbol{x}_{i'},y_{i'})$，相应的协方差函数有如下表达式[20]。

$$\begin{aligned}&\mathrm{Cov}\left[(\hat{y}_{I_1^{(t)},i}-y_i)^2,(\hat{y}_{I_2^{(t)},i'}-y_{i'})^2\middle|\boldsymbol{X}\right]\\ =\ &2\psi^4tr\left\{(\boldsymbol{x}_i\boldsymbol{x}_i^T)H^{-1}_{I_1^{(t)}}H_{I_1^{(t)}\cap I_2^{(t)}}H^{-1}_{I_2^{(t)}}(\boldsymbol{x}_{i'}\boldsymbol{x}_{i'}^T)H^{-1}_{I_1^{(t)}}H_{I_1^{(t)}\cap I_2^{(t)}}H^{-1}_{I_2^{(t)}}\right\}\end{aligned}\tag{3.46}$$

其中，$H_{I^{(t)}}=\boldsymbol{X}_{I^{(t)}}^\top\boldsymbol{X}_{I^{(t)}}$ 且 $i\neq i'$。

对于式（3.45）给出的多元线性回归模型，本节进一步假定设计矩阵 $\mathbf{X}$ 中的每一条样例均从一未知总体中独立同分布地抽取。该总体的均值为 $\boldsymbol{\nu}=(\nu_1,\ldots,\nu_p)$，协方差矩阵为 $\boldsymbol{\Sigma}$，其大小为 $p\times p$。

由式（3.46）可得：

- $\omega=2\psi^4E[\boldsymbol{x}_d^\top H^{-1}_{A\cup B}H_AH^{-1}_{A\cup C}(\boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top)H^{-1}_{A\cup B}H_AH^{-1}_{A\cup C}\boldsymbol{x}_d]$；
- $\gamma=2\psi^4E[\boldsymbol{x}_b^\top H^{-1}_{A\cup B}H_AH^{-1}_{A\cup C}(\boldsymbol{x}_c\boldsymbol{x}_c^\top)H^{-1}_{A\cup B}H_AH^{-1}_{A\cup C}\boldsymbol{x}_b]$；

- $\tau = 2\psi^4 E[\boldsymbol{x}_b^\top H_{A\cup B}^{-1} H_A H_{A\cup C}^{-1}(\boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top) H_{A\cup B}^{-1} H_A H_{A\cup C}^{-1}\boldsymbol{x}_b]$，
  或 $\tau = 2\psi^4 E[\boldsymbol{x}_d^\top H_{A\cup B}^{-1} H_A H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top) H_{A\cup B}^{-1} H_A H_{A\cup C}^{-1}\boldsymbol{x}_d]$。

其中，$E[\cdot]$ 取自整个设计矩阵 $\boldsymbol{X}$。

由此可得：

$$\omega + \gamma - 2\tau = 2\psi^4 tr\left\{E[(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top) H_{A\cup B}^{-1} H_A H_{A\cup C}^{-1} \cdot (\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top) H_{A\cup B}^{-1} H_A H_{A\cup C}^{-1}]\right\}。$$

根据 $H_A = \boldsymbol{X}_A^\top \boldsymbol{X}_A$，可得：

$$\omega + \gamma - 2\tau = 2\psi^4 tr\left\{E[\boldsymbol{X}_A H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top) H_{A\cup B}^{-1}\boldsymbol{X}_A^\top \boldsymbol{X}_A H_{A\cup C}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top) H_{A\cup B}^{-1}\boldsymbol{X}_A^\top]\right\}。$$

设计矩阵 $\boldsymbol{X}_A$ 由 $\{\boldsymbol{x}_a | a \in A\}$ 和 $\boldsymbol{X}_A^\top \boldsymbol{X}_A = \sum_{a'\in A} \boldsymbol{x}_{a'}\boldsymbol{x}_{a'}^\top$ 组成。因矩阵的迹为其对角线元素之和，可得：

$$\begin{aligned} & tr\{E[\boldsymbol{X}_A H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top) H_{A\cup B}^{-1}\boldsymbol{X}_A^\top \boldsymbol{X}_A H_{A\cup C}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top) H_{A\cup B}^{-1}\boldsymbol{X}_A^\top]\} \\ = & \sum_{a\in A}\sum_{a'\in A} E[\boldsymbol{x}_a^\top H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top) H_{A\cup B}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_{a'}^\top H_{A\cup C}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top) H_{A\cup B}^{-1}\boldsymbol{x}_a]。 \end{aligned} \tag{3.47}$$

因 $\boldsymbol{x}_{a'}^\top H_{A\cup C}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top) H_{A\cup B}^{-1}\boldsymbol{x}_a$ 为一随机数，则有

$$\boldsymbol{x}_{a'}^\top H_{A\cup C}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top) H_{A\cup B}^{-1}\boldsymbol{x}_a = \boldsymbol{x}_a^\top H_{A\cup B}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top) H_{A\cup C}^{-1}\boldsymbol{x}_{a'}。 \tag{3.48}$$

因此，有：

$$\begin{aligned} \omega + \gamma - 2\tau &= \sum_{a\in A}\sum_{a'\in A} E[\boldsymbol{x}_a^\top H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top) H_{A\cup B}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_a^\top H_{A\cup B}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top) H_{A\cup C}^{-1}\boldsymbol{x}_{a'}] \\ &= \sum_{a\in A}\sum_{a'\in A} E[tr\{H_{A\cup C}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_a^\top H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top) H_{A\cup B}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_a^\top H_{A\cup B}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top)\}] \\ &= \sum_{a\in A}\sum_{a'\in A} tr\{E_{\boldsymbol{x}_a,\boldsymbol{x}_{a'}}[E[H_{A\cup C}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_a^\top H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top)|\boldsymbol{x}_a,\boldsymbol{x}_{a'}] \\ &\quad \cdot E[H_{A\cup B}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_a^\top H_{A\cup B}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top)|\boldsymbol{x}_a,\boldsymbol{x}_{a'}]]\}。 \end{aligned} \tag{3.49}$$

对于任意 $a, a' \in A$，在 $\boldsymbol{x}_a, \boldsymbol{x}_{a'}$ 的条件下，有

$$E[H_{A\cup C}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_a^\top H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top)|\boldsymbol{x}_a,\boldsymbol{x}_{a'}] = E[H_{A\cup B}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_a^\top H_{A\cup B}^{-1}(\boldsymbol{x}_b\boldsymbol{x}_b^\top - \boldsymbol{x}_d\boldsymbol{x}_d^\top)|\boldsymbol{x}_a,\boldsymbol{x}_{a'}]。$$

由此可得：

$$\omega + \gamma - 2\tau = \sum_{a\in A}\sum_{a'\in A} tr\{E_{\boldsymbol{x}_a,\boldsymbol{x}_{a'}}[E^2[H_{A\cup C}^{-1}\boldsymbol{x}_{a'}\boldsymbol{x}_a^\top H_{A\cup C}^{-1}(\boldsymbol{x}_c\boldsymbol{x}_c^\top - \boldsymbol{x}_{d'}\boldsymbol{x}_{d'}^\top)|\boldsymbol{x}_a,\boldsymbol{x}_{a'}]]\} > 0。$$

因此，对于多元线性回归，条件 $\omega + \gamma - 2\tau > 0$ 成立。

3.8　本章小结

本章给出了数据集的一种新的交叉验证方法，称为正则化 $m \times 2$ 交叉验证。在正则化 $m \times 2$ 交叉验证中，训练集间重叠样本个数被约束至 $n/4$ 左右。本章给出了泛化误差的正则化 $m \times 2$ 交叉验证估计的优良性质，即：正则化 $m \times 2$ 交叉验证估计的方差小于随机切分情形下 $m \times 2$ 交叉验证估计的方差。基于二水平正交表，本章给出了正则化 $m \times 2$ 交叉验证的高效增量式构造方法。

对于非独立同分布的数据集中，例如，文本数据集，正则化交叉验证方法基本适用。不过，正则化交叉验证方法应根据数据集的特点进行扩展。正则化交叉验证方法不仅应控制两个训练集间重叠样本个数，而且应确保训练集和验证集间的分布差异较小。对于自然语言处理任务中使用的文本数据集，数据切分通常建立在句子、段落或文本上。因此，为了设计更好的数据切分，需要针对文本数据集的特点，引入训练集和验证集的各种频次分布的差异度量，如，词频分布、标记分布、句子长度分布等。进而，可使用这些度量来构造正则化条件，并进一步优化数据切分，以优化交叉验证估计的方差。为此，本文进一步提出适用于文本数据集的正则化交叉验证方法。

第四章　文本数据的正则化 $m \times 2$ 交叉验证初探

面向文本数据建模时，交叉验证方法是特征选择及算法比较任务中的常用方法。许多研究表明，文本数据模型的性能的估计对对交叉验证的数据切分方式较为敏感，不合理的切分方式可能会导致不稳定的性能估计值，使得实验结果的可靠性差。本章试图论证，基于正则化 $m \times 2$ 交叉验证，通过引入对训练集、验证集分布差异的约束，构造适用于文本数据的正则化 $m \times 2$ 交叉验证方法，有利于改善文本数据模型的性能估计。本章首先针对文本数据引入训练集与验证集分布差异的卡方度量，基于该度量构建数据切分的正则化条件，以最大化模型性能指标的信噪比为目标，给出了满足正则化条件的 $m \times 2$ 交叉验证的数据切分优化算法。最后，以自然语言处理中汉语框架语义角色标注任务为例，验证了文本数据上正则化 $m \times 2$ 交叉验证方法的有效性。

4.1　构造正则化交叉验证方法的基本思路

对给定的文本数据集做 m 次切分，实施 m 次 2 折交叉验证，被称为 $m \times 2$ 交叉验证。从形式上看，$m \times 2$ 交叉验证是 Dietterich 等提出的 5×2 交叉验证的扩展[1]。不过，依照 5×2 交叉验证所采用的随机切分来构建 $m \times 2$ 交叉验证，并不适用于文本数据的建模，需要研究新的切分算法。

面向文本数据，研究如何构建 $m \times 2$ 交叉验证的优化切分算法，以减小切分带来的训练集、验证集的分布差异，降低人为切分带来的模型性能指标估计（简称为指标估计）的波动，力求使得指标估计的方差达到最小，使得基于 $m \times 2$ 交叉验证的算法比较结论更为可靠。

为此，本章引入合适的差异度量函数，来度量训练集和验证集的分布差异，形成控制差异的多个约束条件（简称为正则化条件），以指标的信噪比最大化为目标，求解 $m \times 2$ 交叉验证的优化切分算法。

在文本数据集上，许多情形下，可基于标记集合的离散概率分布差异来度量两个数据集的分布差异。比如，在语义角色标注任务中，可将语义角色类型分布看作多项分布，使用分布一致性检验的卡方统计量来度量训练集和验证集的分布差异。

4.1.1　记号和定义

记文本数据集为 $D_n = \{d_1, d_2, \ldots, d_n\}$，其中，$d_i$ 为数据的基本切分单位（个体样本），可以是句子或篇章，依赖于具体的自然语言处理任务。如，在文本分类任

务中，d_i 为篇章；在分词、语义角色标注任务中，d_i 通常为句子。文本集 D_n 的索引集记为 $I = 1, 2, \ldots, n$，其中，n 为 D_n 所包含基本切分单位的个数。围绕文本数据集 D_n，下面引入一些必要的定义：

- $m \times 2$ **交叉验证**：索引集合 I 上的一个对半切分记为 $\mathcal{S} = (I^{(t)}, I^{(v))}$，其中，$I^{(t)} \cup I^{(v)} = I$，$I^{(t)} \cap I^{(v)} = \emptyset$ 且 $|I^{(t)}| = |I^{(v)}| = n/2$；$m \times 2$ 交叉验证的的切分集合记为：$\boldsymbol{P} = <\mathcal{S}_i, \mathcal{S}_i^{\top}> : \mathcal{S}_i = (I_i^{(t)}, I_i^{(v)}), \mathcal{S}_i^{\top} = (I_i^{(v)}, I_i^{(t))}, i = 1, 2, .., m$，称 $\mathcal{S}_i^{\top}$ 为 $\mathcal{S}_i$ 的对折切分，$<\mathcal{S}_i, \mathcal{S}_i^{\top}>$ 为一个切分对；基于某个切分 $\mathcal{S}_i = (I_i^{(t)}, I_i^{(v)})$，定义两个数据块 $D_i^{(t)} = \{d_k | k \in I_i^{(t)}\}$ 和 $D_i^{(v)} = \{d_k | k \in I_i^{(v)}\}$，其中，$D_i^{(t)}$ 通常被称为训练集，$D_i^{(v)}$ 被称为验证集；同理，对折切分 $\mathcal{S}_i^{\top} = (I_i^{(v)}, I_i^{(t)})$ 上，数据块 $D_i^{(v)}$ 为训练集，数据块 $D_i^{(t)}$ 为验证集；
- **算法性能指标的估计**：给定两个机器学习算法 $\mathcal{A}$、$\mathcal{B}$ 及算法性能指标 ν_A、ν_B（比如，准确率 P、召回率 R、F_1 值等），记 $\nu = \nu_A - \nu_B$ 为两个算法性能之差，ν 在切分 $\mathcal{S}_i = (I_i^{(t)}, I_i^{(v)})$ 上的 hold-out 估计记为 $\hat{\nu}_1^{(i)}$，在切分 $\mathcal{S}_i^{\top} = (I_i^{(v)}, I_i^{(t)})$ 上的 hold-out 估计记为 $\hat{\nu}_2^{(i)}$；则，ν 在切分对 $<\mathcal{S}_i, \mathcal{S}_i^{\top}>$ 上的 2 折交叉验证估计为 $\hat{\nu}^{(i)} = (\hat{\nu}_1^{(i)} + \hat{\nu}_2^{(i)})/2$；在切分集合 $\boldsymbol{P}$ 上，ν 的 $m \times 2$ 交叉验证估计为 m 次 2 折交叉验证估计的平均，记为

$$\hat{\nu}_m = \sum_{i=1}^{m} \hat{\nu}^{(i)}/m。 \tag{4.1}$$

  若 $\mathcal{B}$ 为空模型，则 $\hat{\nu}_m$ 就是算法 $\mathcal{A}$ 的指标估计；
- 记 $\boldsymbol{\Phi} = \{\Phi^{(0)}, \Phi^{(1)}, \ldots, \Phi^{(K)}\}$ 为差异度量函数集，其中：$\Phi^{(0)}$ 为任意两个训练集间重叠样本个数的度量，m 次切分共有 $m(m-1)/2$ 个不同的值组成向量：$\Phi^{(0)} = (\phi_{ij}^{(0)} = |I_i^{(t)} \cap I_j^{(t)}| : i, j = 1, 2, \ldots, m且i < j)$。定理3.1 表明，任意两训练集的重叠个数为 $n/4$ 时，指标估计的方差达到最小，并给出了切分算法；函数 $\Phi^{(k)} = (\phi_i^{(k)} = g_k(D_i^{(t)}, D_i^{(v)}) : i = 1, 2, \ldots, m)$ 为长度为 m 的差异度量向量，$g_k(D_i^{(t)}, D_i^{(v)})$ 为训练集 $D_i^{(t)}$ 与验证集 $D_i^{(v)}$ 的分布差异度量函数，其中，$k = 1, 2, \ldots, K$。比如，在语义角色标注任务中，$g_1(D_i^{(t)}, D_i^{(v)})$ 可以表示语义角色类型的标记分布在训练集、验证集上的差异，$g_2(D_i^{(t)}, D_i^{(v)})$ 可以表示给定的某个框架下，不同词元的数量分布在两个数据集上的差异，等等。K 的取值，视具体任务而定。一般情况下，至少包含 $\Phi^{(0)}$ 和 $\Phi^{(1)}$ 两个度量函数。

直观上讲，好的切分集合需满足：（1）基于上述度量函数集，任意的训练集 $D_i^{(t)}$ 和验证集 $D_i^{(v)}$ 之间的分布差异应尽量小；（2）任意两个不同的训练集 $D_i^{(t)}$ 和 $D_j^{(t)}$（$i \neq j$）之间的重叠样本个数要尽量少且一致。所引入的度量函数可成为分析

指标估计与切分集合关系的重要“桥梁”。

根据算法性能指标的 $m \times 2$ 交叉验证估计的定义可知，给定若干正则条件后，该估计的方差 $\mathrm{Var}[\hat{\mu}|\boldsymbol{\Phi}]$ 有如下分解形式。

$$\mathrm{Var}[\hat{\nu}_m|\boldsymbol{\Phi}] = \frac{1}{m^2}\sum_{i=1}^{m}\mathrm{Var}[\hat{\nu}^{(i)}|\Phi^{(1)},\ldots,\Phi^{(K)}] + \frac{1}{m^2}\sum_{i\neq j}\mathrm{Cov}[\hat{\nu}^{(i)},\hat{\nu}^{(j)}|\Phi^{(0)},\Phi^{(1)},\ldots,\Phi^{(K)}]。 \tag{4.2}$$

其中，式子右边第一项为单个 2 折交叉验证估计的方差。该方差与 $\Phi^{(0)}$ 无关。

4.1.2　正则化 $m \times 2$ 交叉验证求解的优化表示

正则化 $m \times 2$ 交叉验证通过优化切分集合 $\boldsymbol{P}$ 来控制度量 $\boldsymbol{\Phi}$ 的大小，以减小指标估计的方差，增大指标估计的信噪比。具体形式化为如下优化问题。

$$\boldsymbol{P}^* = \arg\min_{\boldsymbol{P}} \quad \frac{E[\hat{\nu}_m|\boldsymbol{\Phi}]}{\mathrm{stdev}[\hat{\nu}_m|\boldsymbol{\Phi}]} \tag{4.3}$$

$$s.t. \quad \begin{vmatrix} \phi_{ij}^{(0)} - E[\phi_{ij}^{(0)}] \end{vmatrix} \leq c_0 \\ \begin{vmatrix} \phi_i^{(k)} \end{vmatrix} \leq c_k \tag{4.4}$$

其中，$i,j = 1,2,\ldots,m$ 且 $k = 1,2,\ldots,K$。

式（4.4）中的约束条件被称为正则化条件，其中，称参数 c_0 及 c_k 为正则化参数，用来约束训练集与验证集的分布差异；$E[\hat{\mu}_m|\boldsymbol{\Phi}]$ 为指标估计的期望，$\mathrm{stdev}[\hat{\mu}_m|\boldsymbol{\Phi}]$ 为指标估计的标准差；正则化条件 $\left|\phi_{ij}^{(0)} - E[\phi_{ij}^{(0)}]\right| \leq c_0$ 表示任意两个不同的训练集 $D_i^{(t)}$ 和 $D_j^{(t)}$ （$i \neq j$）间的重叠样本个数要控制在期望 $E[\phi_{ij}^{(0)}] = n/4$ 附近。仅在约束 $\boldsymbol{\Phi}^{(0)}$ 下，第三章已给出式（4.3）的解。因此，本文主要考虑加入对度量 $\Phi^{(1)},\ldots,\Phi^{(K)}$ 的约束，如何求解式（4.3），以获得最优切分集合 $\boldsymbol{P}^*$。

4.1.3　训练集、验证集分布差异的度量函数

在文本数据上的很多自然语言处理任务，预测标记（比如，语义角色类型）及特征（比如，词性、词）大多是离散的，本文使用两离散随机变量分布一致性检验的卡方统计量来度量训练集与验证集之间的差异。

设随机变量 L 表示上述某种标记，其取值为一个离散标记集合 $\{l_1,\ldots,l_J\}$。随机变量 L 在训练集和验证集上的分布差异，可用如下卡方统计量来度量。

$$\chi^2 = \sum_{j=1}^{J}\frac{n^{(t)}(r_j^{(t)} - r_j)^2 + n^{(v)}(r_j^{(v)} - r_j)^2}{r_j}。 \tag{4.5}$$

其中，$n^{(t)}$ 和 $n^{(v)}$ 为其在训练集及验证集上的频次，r_j，$r_j^{(t)}$ 和 $r_j^{(v)}$ 为第 j 种标记值 l_j 在数据集、训练集及验证集上的频率。本文建议以单位自由度的差异度量函数 χ^2/J 来构成 $\boldsymbol{\Phi}^{(k)}$，其中 $k = 1,2,\ldots,K$。

4.1.4 正则化参数如何选

正则化条件 $\mathbf{\Phi}^{(0)}$ 对应的正则化参数 c_0 的选取方法在第三章中已深入讨论。通常可以选取 $c_0 = 1$，其含义是任意两训练集之间的重叠样本个数基本相等。对于其它的正则化参数，同样建议选取 $c_k \leq 1$，其中 $k = 1, 2, \ldots, K$。也就是说，统计量 χ^2 的单位自由度差异度量 $\mathbf{\Phi}^{(k)}$ 不超过 1。这样设置的含义是：从统计分布一致性检验的角度，约束训练集、验证集分布的差异不能“显著大”。实际操作中，可以先构建满足 $\mathbf{\Phi}^{(0)}$ 的多个切分，再从其中依次选取 $\mathbf{\Phi}^{(k)} \leq 1$ 的切分，依此思路，逐渐增加正则化条件, 直到留下适当的切分个数。

4.2 文本数据集上正则化 $m \times 2$ 交叉验证的切分集合的构造算法

构造最优切分集合 $\mathbf{\Phi}^*$ 的基本思路为：首先构造出满足重叠样本个数 $\mathbf{\Phi}^{(0)}$ 正则化条件的切分集合，其构造算法在第3.3 节中给出，并证明了最多可得到 n 个切分，这里 n 为样本个数（如句子数）；然后，逐个检测其它正则化条件 $|\phi_i^{(k)}|\ c_k$（$i = 1, 2, \ldots, m$ 且 $k = 1, 2, \ldots, K$）是否满足，留下满足所有条件的切分构成优化的切分集合 $\boldsymbol{P}^*$；若保留下来的切分不够，可以考虑加入近似满足正则化条件的切分。

给定正则化条件 $\mathbf{\Phi}^{(0)}$ 后，第三 章给出了正则化 $m \times 2$ 交叉验证方法的分块增量式构造算法。简单来说，就是先将数据集逐块等分（样本数相等），即一块分二块、二块分四块，形成“嵌套”的数据子块；然后基于这样的数据子块，利用二水平正交表的均衡性，构造出满足关于 $\mathbf{\Phi}^{(0)}$ 正则化条件的切分。这样的构造方法是一种增量式构造，即，可以保证已构造出来的切分不变，逐步加入新的切分，最终形成 m 组训练集和验证集。具体地，当 $m = 3$ 时，构造正则化 3×2 的切分，可将数据集 D_n 划分成大小相同的 4 份，然后任取两份作为训练集，其它两份作为验证集, 形成的 6 次实验对应 3 组 2 折交叉验证。当 $m > 3$ 时，构造正则化 $m \times 2$ 交叉验证的切分集需借助二水平正交表。以正则化 7×2 交叉验证为例，基于二水平正交表 $OA(8, 7)$，将数据集等分为 8 块，记为 $I_k^{(8)}$，其中，$k = 1, 2, \ldots, 8$，然后根据表3.2 所示规则可拼合成对应的 7 组训练集和测试集。

4.3 基于正则化 $m \times 2$ 交叉验证的序贯 t 检验

采用合理的统计显著性检验是得到可靠结论的保障。在自然语言处理的大部分文献中，大多在验证集或测试集上直接比较实验结果的绝对大小，或采用基于 5 折（10 折）交叉验证的统计检验。但正如王钰等指出，通常基于 5 折（10 折）交叉验证的统计检验是激进的[8]，因为，这些检验通常假定 5 折（10 折）训练、测试相互

独立，采用 5 折（10 折）实验结果的样本标准差来刻画指标估计的变差。这往往造成其真实没有显著差异而推断有显著差异。为此，本文推荐一种用于模型性能比较的正则化 $m \times 2$ 交叉验证序贯 t 检验方法（具体见第五 章）。该检验方法较为保守，可以得到更为可信的结论。具体描述如下：

记 $\nu = \nu_{\mathcal{A}} - \nu_{\mathcal{B}}$ 为两个算法性能指标的真实差异，$\nu_{\mathcal{B}}$ 为基线模型的性能。

原假设：$H_0 : \nu \leq 0$；备择假设：$H_1 : \nu > 0$；

记模型性能指标之差的 $m \times 2$ 交叉验证估计为

$$\hat{\nu}_m = \sum_{i=1}^{m} \hat{\nu}^{(i)}/m。 \tag{4.6}$$

其中，$\hat{\nu}^{(i)} = (\hat{\nu}_1^{(i)} + \hat{\nu}_2^{(i)})/2$ 为 ν 在第 i 次切分 $\mathcal{S}_i = (I_i^{(t)}, I_i^{(v)})$ 上的 2 折交叉验证估计。所采用的检验统计量如下式所示：

$$T_{m\times 2} = \frac{\hat{\nu}_m}{\hat{\sigma}_m} \sim C_m t(2m-1)。 \tag{4.7}$$

其中，

$$\hat{\sigma}_m = \sqrt{\frac{1}{2m}\sum_{i=1}^{m}\sum_{k=1}^{2}(\hat{\nu}_k^{(i)} - \hat{\nu}_m)^2}, \tag{4.8}$$

$$C_m = \sqrt{\frac{2m+1}{2m-1}}。 \tag{4.9}$$

$T_{m\times 2}$ 为近似服从自由度为 $2m-1$ 的 t 分布统计量，其置信区间为：

$$(\hat{\nu}_m - C_m\hat{\sigma}_m t_\alpha(2m-1), \quad \hat{\nu}_m + C_m\hat{\sigma}_m t_\alpha(2m-1))。 \tag{4.10}$$

该置信区间长度的期望的极限值为：

$$\lim_{m\to\infty} 2E[C_m\hat{\sigma}_m t_\alpha(2m-1)] = 2\sqrt{1-\rho}\sigma u_\alpha, \tag{4.11}$$

其中，ρ 为两个 2 折交叉验证估计间的相关系数，σ 为指标估计的标准差的真实值，u_α 为标准正态分布的上侧分位数。置信区间理论的长度为 $2\sqrt{\rho}\sigma u_\alpha$，因此，只要 $\rho \leq 0.5$，置信区间就是保守的。

设定停止策略 $\hat{\nu}_m > C_m\sigma_m t_\alpha(2m-1)$，其中，$t_\alpha(2m-1)$ 为 t 分布的 α 上分位数。执行如下步骤：

(1) 设定 $m = m_{start}$；

(2) 判断在当前 m 值时，停止策略 $\hat{\mu}_m > C_m\sigma_m t_\alpha(2m-1)$ 是否成立；

(3) 若停止策略不成立：

a) 判断 m 值是否达到预设最大值 m_{stop}；

b) 若 $m < m_{stop}$，则 $m = m + 1$，并跳至第（2）步，继续执行。

c) 若 $m \geq m_{stop}$，则停止实验，并接受原假设 H_0，即：认为两算法性能差异不显著；

(4) 若停止策略成立，则停止实验，并接受备择假设 H_1，判决两算法性能有显著差异；

其中，m_{start} 和 m_{stop} 为切分次数 m 的开始值和最大结束值。通常，设置 $m_{start} = 3$ 且 $m_{stop} = 20$。

若两两比较的模型对有 k 对时，根据多重比较的理论，则需要将显著水平 α 调整为 α/k. 比如，比如有 5 种不同的配置模型，则两两比较的总个数就有 $\binom{5}{2} = 10$ 对，即 $k = 10$。显著水平应调整为 $\alpha/10$。

从本质上讲，$T_{m\times 2}$ 统计量是信噪比的估计。因此，本章的目标是以最大化信噪比为目标，寻找合适的正则化交叉验证方法，以排除其它因素的干扰，在文本数据上得到更为可靠的算法比较结论。

4.4 实验及结果分析

本节以汉语框架语义角色标注的实验为例，说明不同的交叉验证方案，所导致的词元分布差异、角色分布差异，及其对指标估计的均值、标准差和信噪比的影响。这里，信噪比定义为：

信噪比 = 性能指标的期望/样本标准差

实验语料来自于汉语框架语义知识库（CFN）1.0 中的认知域框架例句。本文选取了 6692 条例句，涵盖 25 个框架。实验中，分别考虑语义角色识别任务和语义角色标注任务。语义角色识别任务的目标是给定一条汉语句子及目标词，识别出该目标词所搭配的语义角色块的边界。该任务并不考虑语义角色的类型，因此，可以在整个语料上建模。对于语义角色标注任务，在识别出语义角色块的同时，还要标注该语义角色块的类型。由于 CFN 中每个框架的语义角色类型不同，语义角色标注任务只能对每个框架单独建模。

本章将语义角色边界识别和语义角色标注任务均看作以词为单位的序列标注问题，采用条件随机场（CRF）模型，并结合 IOB2 标注策略建立模型。所用特征包括词、词性、位置及目标词。实验设置与李济洪等的研究工作[17] 相同。本文采用准确率 P、召回率 R 以及 F_1 值来评价模型性能。在对比两个不同的模型时，本文使用 ΔP、ΔR、ΔF_1 表示两模型性能指标的差。

基于上述实验设置，围绕交叉验证集正则化方法，本节主要回答如下三个问题：

研究问题一 ：语料的不同切分比例如何影响算法性能指标的估计？

研究问题二 ：训练集和验证集的分布差异如何影响算法性能指标的信噪比？

研究问题三 ：正则化 $m \times 2$ 交叉验证与随机 $m \times 2$ 交叉验证，哪个更适合于文本数据？

4.4.1　研究问题一的模拟实验

为了验证语料的不同切分比例对各指标估计的影响，将实验语料按训练集与验证集的 5 : 5、6 : 4、7 : 3、8 : 2、9 : 1 等不同比例的随机切分，并重复随机切分 100 次，分别计算所比较的两个模型（记为算法 $\mathcal{A}$ 和 $\mathcal{B}$）的性能的样本均值、标准差以及信噪比，以及这两个算法性能差异的样本均值、标准差以及信噪比。

本实验中，首先考虑语义角色识别任务。算法 $\mathcal{A}$ 是基于李济洪等给出的最优模板（#1）训练模型[17]，算法 $\mathcal{B}$ 使用的特征模板中将最优模板的词特征（及对应的组合特征）全部去除。

表4.1给出了不同切分比例下，算法性能的各项指标及相应的标准差。其中，P、R、F_1 为准确率、召回率和 F_1 值在 100 次切分上的平均值；stdev P、stdev R、stdev F_1 分别表示准确率、召回率和 F_1 值在 100 次切分上的样本标准差。表4.1 表明，随着数据切分比例的变化，指标的均值和方差都逐渐增大，但方差增加相对较快。图4.1分别给出了算法 $\mathcal{A}$ 和 $\mathcal{B}$ 的信噪比。图4.1 表明，随着训练集比例变大，信噪比在明显下降；当 5 : 5 切分时，算法性能的信噪比最大。

表 4.1　语义角色识别任务中语料不同切分比例对指标估计的影响

切分比例	算法 $\mathcal{A}$ 的性能						算法 $\mathcal{B}$ 的性能					
	P	stdev P	R	stdev R	F_1	stdev F_1	P	stdev P	R	stdev R	F_1	stdev F_1
5:5	0.7405	0.0062	0.6650	0.0059	0.7007	0.0048	0.6524	0.0067	0.5709	0.0066	0.6089	0.0051
6:4	0.7459	0.0062	0.6747	0.0064	0.7085	0.0054	0.6565	0.0076	0.5753	0.0069	0.6132	0.0060
7:3	0.7500	0.0066	0.6823	0.0069	0.7145	0.0056	0.6605	0.0075	0.5782	0.0075	0.6166	0.0065
8:2	0.7535	0.0087	0.6883	0.0079	0.7194	0.0072	0.6613	0.0101	0.5791	0.0107	0.6174	0.0090
9:1	0.7563	0.0130	0.6936	0.0137	0.7235	0.0121	0.6645	0.0132	0.5803	0.0140	0.6195	0.0123

表4.2给出了算法 $\mathcal{A}$ 和 $\mathcal{B}$ 的性能指标之差的均值、标准差及信噪比。同样可见，按 5 : 5 切分时，准确率、召回率和 F_1 值信噪比达到最大。

本文进一步围绕语义角色标注任务来验证切分比例对性能指标估计的影响。考虑到“陈述”框架动词词元（简记为“陈述$_v$”）具有较多的例句（1103 条），实验主要分析了该框架的语义角色标注模型。算法 $\mathcal{A}$ 采用了李济洪等给出的最优模板（#12）[17]；算法 $\mathcal{B}$ 去除了该最优特征模板中与词有关的特征。表4.3给出了相应的

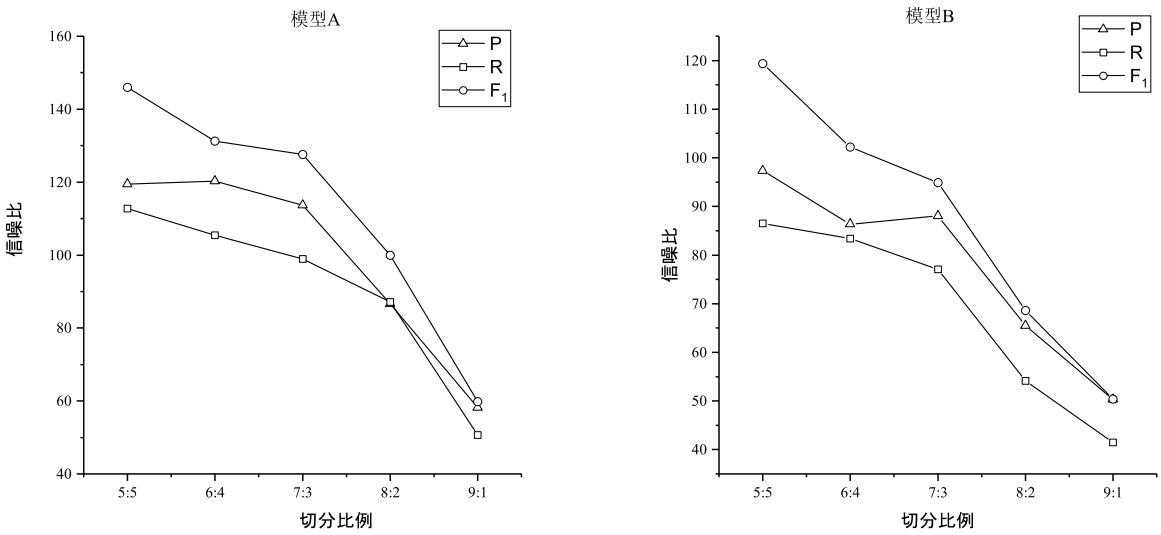

图 4.1 语义角色识别的性能指标的信噪比

表 4.2 语义角色识别任务中算法性能差异的均值和方差

切分比例	ΔP（%）	stdev ΔP	信噪比	ΔR （%）	stdev ΔR	信噪比	ΔF_1 （%）	stdev F_1	信噪比
5:5	8.81	0.0058	15.30	9.40	0.0057	16.59	9.18	0.0046	20.01
6:4	8.94	0.0067	13.40	9.95	0.0064	15.54	9.54	0.0052	18.48
7:3	8.94	0.0070	12.70	10.41	0.0075	13.96	9.79	0.0057	17.09
8:2	9.21	0.0085	10.89	10.92	0.0094	11.66	10.20	0.0079	12.96
9:1	9.18	0.0114	8.08	11.33	0.0126	8.97	10.41	0.0109	9.58

实验结果。

表4.3表明，随着数据切分比例的变化指标的均值和标准差都逐渐增大，其中，标准差增大较快。从信噪比角度来看（见图4.2），随着切分比例的变化，信噪比同样在明显下降。切分比例为 5:5 时，各指标的信噪比达到最大。表4.4 是两算法的性能指标之差的结果。同样表明，在按 5:5 切分时信噪比达到最大。

表 4.3 “陈述$_v$”框架语义角色标注任务上语料不同切分比例对指标的影响

切分比例	算法 $\mathcal{A}$ 的性能						算法 $\mathcal{B}$ 的性能					
	P（%）	stdev P	R （%）	stdev R	F_1 （%）	stdev F_1	P （%）	stdev P	R （%）	stdev R	F_1 （%）	stdev F_1
5:5	74.16	0.0114	61.84	0.0112	67.43	0.0099	62.33	0.0125	51.89	0.0129	56.63	0.0116
6:4	74.86	0.0121	63.50	0.0124	68.71	0.0106	63.15	0.0136	53.10	0.0126	57.68	0.0117
7:3	75.22	0.0161	64.76	0.0176	69.59	0.0154	63.87	0.0154	54.05	0.0175	58.54	0.0150
8:2	75.45	0.0169	65.64	0.0181	70.19	0.0157	64.09	0.0200	54.50	0.0168	58.90	0.0168
9:1	75.47	0.0260	66.32	0.0257	70.58	0.0236	63.91	0.0265	54.71	0.0263	58.94	0.0246

综上所述，随着切分比例的差异增大，标准差逐渐增加，信噪比逐渐变小，在

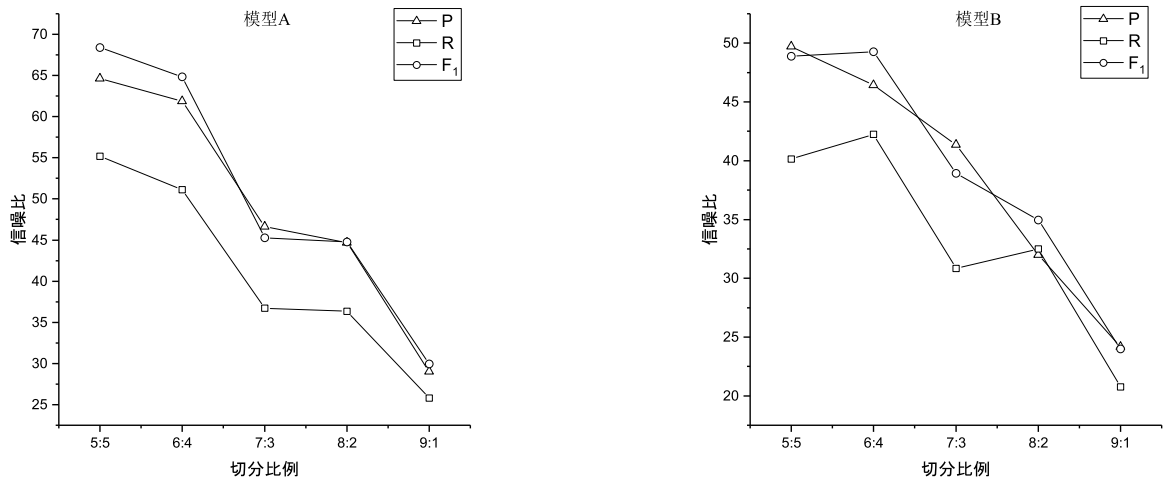

图 4.2 “陈述$_v$”语义角色标注任务的算法性能指标的信噪比

表 4.4 “陈述$_v$”框架下不同切分比例对两算法性能指标差的影响

切分比例	ΔP（%）	stdev ΔP	信噪比	ΔR （%）	stdev ΔR	信噪比	ΔF_1 （%）	stdev F_1	信噪比
5:5	11.83	0.0114	10.364	9.95	0.0089	11.229	10.81	0.0089	12.161
6:4	11.71	0.0123	9.472	10.40	0.0113	9.186	11.02	0.0103	10.656
7:3	11.35	0.0127	8.952	10.71	0.0127	8.437	11.05	0.0111	9.914
8:2	11.36	0.0168	6.776	11.14	0.0154	7.246	11.29	0.0144	7.821
9:1	11.57	0.0250	4.636	11.60	0.0253	4.578	11.65	0.0232	5.020

5:5 切分时标准差达到最小，信噪比达到最大。也就是说，从信噪比角度来看，按照 5:5 切分语料，更适用于文本数据上的算法比较任务。

4.4.2 研究问题二的模拟实验

本节按 5:5 切分的 2 折交叉验证下，分析训练集、验证集的分布差异对指标的信噪比的影响。本节主要基于语义角色标注任务，选取了例句数最多的 6 个框架（见表4.5）作为研究对象，进行对比试验。首先，对每个框架的例句随机进行 2 折交叉验证，重复 1000 次。然后，统计训练集、验证集上的词元分布频次及不同类型的语义角色的分布频次。根据（3）式所给的卡方统计量，分别计算词元及语义角色在训练集与验证集上的分布差异。

针对 1000 次的随机切分，引入两个正则化条件。针对词元的分布差异，引入第一个正则化条件：$|\phi_i^{(1)}| = \chi^2/J \leq c_1$；按语义角色类型分布差异，引入第二个正则化条件 $|\phi_i^{(2)}| = \chi^2/J \leq c_2$。其中，正则化参数 c_1，c_2 的选取方法为：在 1000 个 2 折交叉验证中，寻找满足 $\chi^2/J \leq c_1$ 及 $\chi^2/J \leq c_2$ （本实验中设置 $c_1 = c_2$）且切分个数不超过 $1000 \times 5\%$ 时的 c_1 和 c_2。即，选择满足两个正则化条件的前 50 个切

表 4.5 各框架语义角色标注任务的算法性能指标的信噪比

框架	句子个数	词元个数	角色个数	c_1，c_2	不好的切分			正则化切分		
					P	R	F_1	P	R	F_1
陈述$_v$	1603	84	10	1.09	58.10	51.91	65.35	64.21	54.54	70.19
感受	569	61	4	1.125	25.81	19.35	22.67	24.1	23.28	25.15
自主感知	499	27	12	1.11	33.31	25.61	35.17	34.76	26.49	35.36
拥有	170	15	3	1	17.34	14.5	16.44	18.71	17.5	19.41
知觉特性	345	36	5	1.17	28.05	22.76	32.68	31.04	24.02	33.27
记忆	298	17	9	1.11	33.74	16.85	24.52	29.43	20.27	28.77

分作为优良切分，称为“正则化切分”。同样，选择最不满足两个正则化条件的 50 个切分作为“不好的切分”。分别计算“正则化切分”和“不好的切分”下的模型各性能指标的信噪比，并对其对比分析。实验结果见表4.5。

表4.5给出了实验所用的 6 个汉语框架的基本信息，包括每个框架的句子数、词元个数以及语义角色种类数，以及正则化条件中的正则化参数值 c_1 和 c_2。在表4.5的后 6 列，分别给出了“正则化切分”以及“不好的切分”下，算法准确率、召回率和 F_1 值的信噪比。

从表4.5可知，除“感受”框架的准确率外，“不好的切分”对应的准确率、召回率和 F_1 值的信噪比均比“正则化切分”下对应的信噪比小。这说明词元分布差异和角色类型分布差异的约束对指标的信噪比有明显的影响。也就是说，若采用了不好的切分，指标的信噪比会下降，容易导致不可靠的算法比较结论。

4.4.3 研究问题三的模拟实验

本节主要比较正则化 $m \times 2$ 交叉验证与随机切分的 $m \times 2$ 交叉验证下各指标的信噪比。不失一般性，这里设置 m 为 3。本节主要基于语义角色标注任务，以“陈述$_v$”为例进行分析。具体地，将“陈述”框架动词词元的所有例句作为语料，分别构造正则化 3×2 交叉验证以及随机切分的 3×2 交叉验证。对于每一种交叉验证，执行 500 次随机切分，并基于 500 次的实验结果计算模型各性能指标的均值、标准差及信噪比。构造 3×2 交叉验证时，只需将整个按 5:5 切分，随机切分三次。正则化 3×2 交叉验证的构造方式如下。

对于正则化 3×2 交叉验证，将语料均分为 4 份，将“陈述$_v$”框架的 84 个词元按其数量均匀分配到 4 份数据中，任取 2 份合起来做训练集，剩余 2 份做验证集，这样就可以得到 3 组 2 折交叉验证方案，并分别引入下面的正则化条件：

(1) **正则化 3×2 交叉验证的基本正则化条件**：不同切分之间的样本（例句）重叠个数最多相差 1。由于构造时，将语料均分为 4 分，并两两组合，所得到的切分满足正则化条件 $|\phi_{ij}^{(0)} - E[\phi_{ij}^{(0)}]| \leq 1$。关于正则化 $m \times 2$ 交叉验证的一般构造，可以使用第3.3 节给出的算法；

(2) **正则化 3×2 交叉验证的第一个正则化条件**：计算词元分布差异的卡方值，并对其正则化，即 $|\phi_i^{(1)}| \leq c1$，$i = 1, 2, 3$。由于正则化 3×2 交叉验证的构造将词元按其数量均匀分配到训练集和验证集，故按训练集、验证集中词元数量的差异计算的卡方统计量除以 $J = 84$ （自由度）得到相应的 $\phi_i^{(1)}$；

(3) **正则化 3×2 交叉验证的第二个正则化条件**：按语义角色类型进行正则化，即 $|\phi_i^{(2)}| \leq c_2$，$i = 1, 2, 3$。由于“陈述”框架具有 10 种语义角色，在计算其卡方值时，需将自由度除以 10；

正则化参数 c_1 和 c_2 的值，需根据 500 次切分中满足正则化条件的个数来确定。

表4.6给出了“陈述$_v$”语义角色标注模型上，3×2 交叉验证和正则化 3×2 交叉验证的实验结果。从表4.6 可见，正则化 3×2 交叉验证对应的准确率、召回率和 F_1 值得标准差均有所降低。另外，正则化 3×2 交叉验证的信噪比均大于 3×2 交叉验证的信噪比，说明引入正则化条件后的正则化 3×2 交叉验证能够改善指标估计，有利于得到更为可靠的算法比较结论。

表 4.6　“陈述$_v$”语义角色标注模型上 3×2 交叉验证与正则化 3×2 交叉验证的比较（500 次切分）

交叉验证方法	P			R			F_1		
	均值	标准差	信噪比	均值	标准差	信噪比	均值	标准差	信噪比
3×2 交叉验证	0.7432	0.0115	64.59	0.6194	0.0122	50.84	0.6756	0.0102	66.47
正则化 3×2 交叉验证	0.7443	0.0111	67.24	0.6212	0.0115	54.03	0.6772	0.0095	71.02

为了进一步说明正则化条件的作用，从 3×2 交叉验证的 500 次切分中，挑选出 10 次“不好的切分”$(c1, c2 \geq 1.055)$。然后，从正则化 3×2 交叉验证的 500 次切分中，挑选出 10 次“较好的切分”（$c1, c2 \leq 0.35$）。针对选出的切分，分别计算算法性能的各项指标。结果在表4.7 中给出。

表4.7的结果表明，根据正则化条件所选出的正则化 3×2 交叉验证切分方案明显比 3×2 交叉验证中“不好的切分”对应的 F_1 值的信噪高出许多。

综上，基于 5:5 的切分比例，通过合理地引入正则化条件，使用正则化 $m \times 2$ 交叉验证后，算法性能指标的信噪比有明显改善。

表 4.7 “陈述$_v$”语义角色标注任务上 3×2 交叉验证与正则化 3×2 交叉验证的比较（500 次切分）

交叉验证方法	$c1,c2$	P			R			F_1		
		均值	标准差	信噪比	均值	标准差	信噪比	均值	标准差	信噪比
3×2 交叉验证	≥ 1.055	0.7452	0.0123	60.82	0.6211	0.0155	40.15	0.6775	0.0112	60.60
正则化 3×2 交叉验证	≤ 0.35	0.7432	0.0118	62.79	0.6205	0.0089	70.10	0.6763	0.0076	89.19

4.5 本章小结

基于交叉验证的算法比较是统计机器学习中的常用方法。本章针对文本数据的特点，研究了如何切分数据来构建合理的交叉验证方案，得到如下几点结论：

第一，文本数据的切分比例应当采用 5:5，即 2 折交叉验证，其它比例的切分都会增大指标估计的方差，降低指标的信噪比。

第二，卡方统计量可用来度量数据切分导致的训练集和验证集的预测标记分布差异，并可用于构造数据切分的正则化条件。

第三，在 $m\times2$ 交叉验证中，应当以最大化信噪比为目标，采用正则化的数据切分，构建正则化 $m\times2$ 交叉验证。本文以汉语框架语义角色标注为例，说明了如何设置正则化条件，及如何构造文本数据上的正则化 $m\times2$ 交叉验证。实验结果也验证了该方法的有效性。

传统的将数据简单切分为训练集、验证集、测试集的方法，在数据量较大时才较为有效。对文本数据，当数据量较大时，计算开销也非常大。此时，要得到方差的估计是困难的，进而导致后续的统计显著性检验及统计推断任务无法进行。本文认为，在模型初选时，应避免使用很大的数据集。一条可行思路是：首先选取适当大小的数据，并借助于正则化 $m\times2$ 交叉验证，对数据有效利用，来选出有效的特征，甄别出优良的模型。当确定出较好的模型后，再使用较大的数据，获得该模型的参数估计。本文的目标在于对给定的文本数据上，给出一种交叉验证的有效建模方法。文中所提到的正则化条件是文本数据切分时的最基本的约束条件，读者完全根据自然语言的具体任务，根据实验者的经验来设置正则化条件。不过，当多个正则化条件并存时，如何构建高效的切分仍然是后续需要研究的问题。此外，除卡方统计量外，采用其它差异度量，比如相似度等，是否更为有效，也是下一步需要研究的问题。

第五章　针对泛化误差的正则化 $m \times 2$ 交叉验证统计推断方法

针对算法比较任务，当算法性能指标为泛化误差时，本章给出了基于正则化 $m \times 2$ 交叉验证的统计推断方法。首先，将算法比较形式化为一个假设检验问题；然后，给出泛化误差的正则化 $m \times 2$ 交叉验证估计及其合理的方差估计。进而，基于该方差估计，严格构造了一个服从 t 分布的统计量。通过引入正则化 $m \times 2$ 交叉验证估计中相关系数的合理设置，得到一个保守的 t 检验统计量。本章构建了用于算法比较的序贯 t 检验，并给出了相应的序贯置信区间及其理论性质。

5.1　问题描述

在有监督学习任务中，假定机器学习算法 $\mathcal{A}$ 为数据集 D_n 上的基线算法。若研究者新发明了一个学习算法 $\mathcal{B}$，他需要证明算法 $\mathcal{B}$ 的性能更为优良，且算法 $\mathcal{B}$ 的泛化误差比算法 $\mathcal{A}$ 的泛化误差小 Δ。研究者的这些论断需要使用一个可靠的统计显著性检验进行验证。

具体来说，记 $\mu_{\mathcal{A}}$ 和 $\mu_{\mathcal{B}}$ 为算法 $\mathcal{A}$ 和算法 $\mathcal{B}$ 的泛化误差。记 $\mu = \mu_{\mathcal{B}} - \mu_{\mathcal{A}}$ 为算法 $\mathcal{A}$ 和算法 $\mathcal{B}$ 的泛化误差的真实差值。上述的算法比较问题可定义为如下复合假设检验问题。

$$H_0 : \ \mu \leq \Delta; \quad H_1 : \ \mu > \Delta。 \tag{5.1}$$

其中，Δ 为研究者自定义的阈值。该阈值反映了研究者对两算法的泛化误差之差的期望。在算法比较任务中，可使用基于交叉验证的 t 检验来解决上述的假设检验问题。在一个大小为 n 的数据集 D_n 上，交叉验证可递增地执行每个切分上的训练和验证过程，并序贯地产生 μ 的估计值。受这种做法的启发，本章使用统计序贯检验来解决假设检验问题（5.1），具体关注序贯 t 检验及其理论分析。序贯 t 检验使用递增的方式计算 μ 的估计值及相应的方差，然后决定是否应该拒绝原假设 H_0。一个优良的检验方法应具有更小的第一类错误和第二类错误。通常，**第一类错误**为错误拒绝原假设的概率，即，P(拒绝 $H_0|H_0$ 为真)；**第二类错误**为当备择假设为真时，未拒绝 H_0 的概率，即，P(未拒绝 $H_0|H_1$ 为真)。

5.2　泛化误差差值 μ 的正则化 $m \times 2$ 交叉验证序贯置信区间

本节先简单回顾正则化 $m \times 2$ 交叉验证方法，然后基于正则化 $m \times 2$ 交叉验证构造相应的 t 检验统计量，并给出 μ 的序贯置信区间及其理论性质。

5.2.1 回顾正则化 $m \times 2$ 交叉验证方法

正则化 $m \times 2$ 交叉验证是带有正则化条件的 $m \times 2$ 交叉验证，具体介绍见第三章。通过将训练集间的重叠样本个数约束至 $n/4$，该正则化条件可减少随机切分对 $m \times 2$ 交叉验证估计的方差的影响。正则化 $m \times 2$ 交叉验证估计的方差小于随机切分情形下 $m \times 2$ 交叉验证估计的方差（见定理3.1），并且正则化 $m \times 2$ 交叉验证具有高效增量式构造算法。王钰等的研究工作表明现有的显著性检验方法，如 5×2 交叉验证 t 检验[1] 及合并 5×2 交叉验证 F 检验 [6] 等，都可使用正则化 $m \times 2$ 交叉验证来改进 [8,9]。改进后的显著性检验可给出更可靠的统计推断结论。受这些研究工作的启发，本章在算法比较任务中应用正则化 $m \times 2$ 交叉验证来构造显著性检验。

5.2.2 泛化误差差值 μ 的正则化 $m \times 2$ 交叉验证估计

若非明确声明，本章余下部分的统计推断过程均使用正则化 $m \times 2$ 交叉验证方法。将算法 $\mathcal{A}$ 和 $\mathcal{B}$ 的泛化误差的差值 μ 的正则化 $m \times 2$ 交叉验证估计记为 $\hat{\mu}_{m\times 2}$，其定义如下

$$\hat{\mu}_{m\times 2} \triangleq \frac{1}{m}\sum_{j=1}^{m}\hat{\mu}^{(j)} = \frac{1}{2m}\sum_{j=1}^{m}\sum_{k=1}^{2}\hat{\mu}_k^{(j)} = \frac{1}{2m}\sum_{j=1}^{m}\sum_{k=1}^{2}(\hat{\mu}_{\mathcal{B},k}^{(j)} - \hat{\mu}_{\mathcal{A},k}^{(j)}) \triangleq \hat{\mu}_{\mathcal{B},m\times 2} - \hat{\mu}_{\mathcal{A},m\times 2}。 \tag{5.2}$$

其中：

- 记 $\hat{\mu}_{\mathcal{A},m\times 2}$ 和 $\hat{\mu}_{\mathcal{B},m\times 2}$ 分别为算法 $\mathcal{A}$ 和算法 $\mathcal{B}$ 的正则化 $m \times 2$ 交叉验证估计；
- 记 $\hat{\mu}^{(j)} = \sum_{k=1}^{2}\hat{\mu}_k^{(j)}/2$ 为 μ 的第 j 个 2 折交叉验证估计，其中，$j = 1, 2, \ldots, m$。算法 $\mathcal{A}$ 和算法 $\mathcal{B}$ 的第 j 个 2 折交叉验证估计分别记为 $\hat{\mu}_{\mathcal{A}}^{(j)}$ 和 $\hat{\mu}_{\mathcal{B}}^{(j)}$；
- 记 $\hat{\mu}_k^{(j)} = \hat{\mu}_{\mathcal{B},k}^{(j)} - \hat{\mu}_{\mathcal{A},k}^{(j)}$ 为关于 μ 的第 j 个 2 折交叉验证中第 k 个 hold-out 估计。记 $\hat{\mu}_{\mathcal{A},k}^{(j)}$ 和 $\hat{\mu}_{\mathcal{B},k}^{(j)}$ 分别为算法 $\mathcal{A}$ 和算法 $\mathcal{B}$ 的 hold-out 估计。其中，$j = 1, \ldots, m$ 且 $k = 1, 2$；

  易知，$\hat{\mu}_{m\times 2}$ 为 $\mu(n/2)$ 的一个无偏估计[16]。$\hat{\mu}_{m\times 2}$ 的方差具有如下表达式。

$$\mathrm{Var}[\hat{\mu}_{m\times 2}] = \frac{1}{2m}\sigma^2 + \frac{1}{2m}\sigma^2\rho_1 + \frac{m-1}{m}\sigma^2\rho_2。 \tag{5.3}$$

其中，参数 σ^2、ρ_1 和 ρ_2 的含义如下：

- $\sigma^2 \triangleq \mathrm{Var}[\hat{\mu}_k^{(j)}]$ 为 hold-out 估计 $\hat{\mu}_k^{(j)}$ 的方差，其中，$j = 1, \ldots, m$ 且 $k = 1, 2$。该方差与 j 和 k 无关；
- $\rho_1 \triangleq \mathrm{Cov}[\hat{\mu}_1^{(j)}, \hat{\mu}_2^{(j)}]/\sigma^2$ 为 2 折交叉验证估计 $\hat{\mu}^{(j)}$ 中两个 hold-out 估计间的相关系数，被称为组内相关系数，且与 j 无关。记 $\rho_{\mathcal{A},1}$ 和 $\rho_{\mathcal{B},1}$ 分别为算法 $\mathcal{A}$ 和算

法 $\mathcal{B}$ 的组内相关系数；

- $\rho_2 \triangleq \mathrm{Cov}[\hat{\mu}_1^{(j)}, \hat{\mu}_2^{(j')}]/\sigma^2$ 为不同切分上两个 2 折交叉验证估计的相关系数，其中，$j \neq j'$。它也被称为组间相关系数。记 $\rho_{\mathcal{A},2}$ 和 $\rho_{\mathcal{B},2}$ 分别为算法 $\mathcal{A}$ 和算法 $\mathcal{B}$ 的组间相关系数；

当 m 趋于无穷时，$\mathrm{Var}[\hat{\mu}_{m\times 2}]$ 递减且收敛于 $\rho_2\sigma^2$。王钰等的研究工作表明，ρ_2 的取值通常大于零[8]。本文第三 章的实验结果也支持这一点。因此，$\mathrm{Var}[\hat{\mu}_{m\times 2}]$ 与定义在独立同分布的样例上的方差不同。当样本量趋于无穷时，后者收敛于零。因此，若算法比较方法假设 $\rho_2 = 0$，则该算法比较方法容易产生假阳性的统计推断结论。

引理 5.1　记 $\hat{\boldsymbol{\mu}}_m = (\hat{\mu}_1^{(1)}, \hat{\mu}_2^{(1)}, \ldots, \hat{\mu}_1^{(m)}, \hat{\mu}_2^{(m)})^\top$ 为正则化 $m \times 2$ 交叉验证中所有 hold-out 估计构成的向量，则有

$$E[\hat{\boldsymbol{\mu}}_m] = \mu \mathbf{1}_{2m}, \qquad \mathrm{Cov}[\hat{\boldsymbol{\mu}}_m] = \sigma^2 \boldsymbol{\Sigma}_{2m}, \tag{5.4}$$

其中，$\mathbf{1}_{2m}$ 是 $2m$ 个元素均为 1 的向量，相关系数矩阵 $\boldsymbol{\Sigma}_{2m}$ 为

$$\boldsymbol{\Sigma}_{2m} = \begin{pmatrix} 1 & \rho_1 & \rho_2 & \rho_2 & \cdots & \rho_2 & \rho_2 \\ \rho_1 & 1 & \rho_2 & \rho_2 & \cdots & \rho_2 & \rho_2 \\ \rho_2 & \rho_2 & 1 & \rho_1 & \cdots & \rho_2 & \rho_2 \\ \rho_2 & \rho_2 & \rho_1 & 1 & \cdots & \rho_2 & \rho_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \rho_2 & \rho_2 & \rho_2 & \rho_2 & \cdots & 1 & \rho_1 \\ \rho_2 & \rho_2 & \rho_2 & \rho_2 & \cdots & \rho_1 & 1 \end{pmatrix}。 \tag{5.5}$$

进而，假定 $\hat{\boldsymbol{\mu}}_m$ 为多元正态分布，则有

$$\frac{\hat{\mu}_{m\times 2} - \mu}{\sqrt{\sigma^2(1 + \rho_1 + 2(m-1)\rho_2)/2m}} \sim \mathcal{N}(0,1)。 \tag{5.6}$$

证明. 见第5.9.1节。 □

由 $\lim\limits_{m\to\infty} \sqrt{\sigma^2(1 + \rho_1 + 2(m-1)\rho_2)/2m} = \sqrt{\rho_2}\sigma$，可得

$$\lim_{m\to\infty} p(\hat{\mu}_{m\times 2} - \sqrt{\rho_2}\sigma u_{\alpha/2} < \mu < \hat{\mu}_{m\times 2} + \sqrt{\rho_2}\sigma u_{\alpha/2}) = 1 - \alpha。 \tag{5.7}$$

其中，$u_{\alpha/2}$ 为标准正态分布的上侧 $\alpha/2$ 分位数。

注记 5.1　正则化 $m \times 2$ 交叉验证上，当 $\rho_2 > 0$ 时，μ 的置信区间长度的期望值趋于 $2\sqrt{\rho_2}\sigma u_{\alpha/2}$，而不是 0。

5.2.3 正则化 $m \times 2$ 交叉验证估计的方差估计

在引入 $\mathrm{Var}[\hat{\mu}_{m\times 2}]$ 的估计之前，先给出如下定理：

定理 5.1 不存在 $\mathrm{Var}[\hat{\mu}_{m\times 2}(\mathbb{S}^*)]$ 的通用无偏估计。

所谓“通用”无偏估计指该估计在任意的样本分布下都是无偏的。该定理的证明与 Bengio 等及杨杏丽等给出的证明[23,70] 是类似的，这里不再赘述。

本小节所给出的 $\mathrm{Var}[\hat{\mu}_{m\times 2}]$ 的估计既与单个 2 折交叉验证估计内的样本方差有关（简称为组内样本方差），也与 m 个 2 折交叉验证估计间的样本方差（简称为组间样本方差）有关。与[8] 类似，本节将 $\mathrm{Var}[\hat{\mu}_{m\times 2}]$ 的估计定义为组内样本方差与组间样本方差的线性组合。

$$\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}] = \lambda_1 \frac{1}{m^2}\sum_{i=1}^{m}\sum_{k=1}^{2}(\hat{\mu}_k^{(i)} - \hat{\mu}^{(i)})^2 + \lambda_2 \frac{1}{m-1}\sum_{i=1}^{m}(\hat{\mu}^{(i)} - \hat{\mu}_{m\times 2})^2。 \tag{5.8}$$

其中，λ_1 和 λ_2 为两个超参数，可用于权衡组内样本方差和组间样本方差的比重。通常，组内样本方差为 $\mathrm{Var}[\hat{\mu}_{m\times 2}]$ 的下偏估计。

式（5.8）的 $\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]$ 的期望值为

$$E[\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]] = \frac{1}{2m}\sigma^2(1+\rho_1) + \frac{1}{m}\sigma^2\left[\left(\lambda_1 + \frac{m}{2}\lambda_2 - \frac{1}{2}\right) - \left(\lambda_1 - \frac{m}{2}\lambda_2 + \frac{1}{2}\right)\rho_1 - m\lambda_2\rho_2\right]。$$

超参数 λ_1 和 λ_2 的值决定上述方差估计是上偏还是下偏。然而，为 λ_1 和 λ_2 找到一组合适且通用值是较为困难的。下面，给出超参数 λ_1 和 λ_2 的几组取值及相应的方差估计量。

(1) 第一个估计量 $\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$，对应 $\lambda_1 = \frac{m}{2}$ 且 $\lambda_2 = 0$。

$$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}] \triangleq \frac{1}{2m}\sum_{i=1}^{m}\sum_{k=1}^{2}(\hat{\mu}_k^{(i)} - \hat{\mu}^{(i)})^2。 \tag{5.9}$$

$$E[\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]] = \frac{1}{2m}\sigma^2(1+\rho_1) + \frac{m-1}{m}\sigma^2\left[\frac{1}{2} - \frac{m+1}{2(m-1)}\rho_1\right]。 \tag{5.10}$$

$$E[\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]] - \mathrm{Var}[\hat{\mu}_{m\times 2}] = \frac{m-1}{m}\sigma^2\left[\frac{1}{2} - \frac{m+1}{2(m-1)}\rho_1 - \rho_2\right]。 \tag{5.11}$$

当 ρ_1 和 ρ_2 满足 $\frac{1}{2} > \rho_2 + \frac{m+1}{2(m-1)}\rho_1$，该估计量为上偏估计，即：$E[\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]] > \mathrm{Var}[\hat{\mu}_{m\times 2}]$。

(2) 第二个估计量 $\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$，对应 $\lambda_1 = \frac{m}{2}$ 且 $\lambda_2 = \frac{m-1}{m}$。

$$\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}] \triangleq \frac{1}{2m}\sum_{i=1}^{m}\sum_{k=1}^{2}(\hat{\mu}_k^{(i)} - \hat{\mu}_{m\times 2})^2。 \tag{5.12}$$

$$E[\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]] = \frac{1}{2m}\sigma^2(1+\rho_1) + \frac{m-1}{m}\sigma^2\left[1 - \frac{1}{m-1}\rho_1 - \rho_2\right]。 \tag{5.13}$$

$$E[\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]] - \mathrm{Var}[\hat{\mu}_{m\times 2}] = \frac{m-1}{m}\sigma^2\left[1 - \frac{1}{m-1}\rho_1 - 2\rho_2\right]。 \tag{5.14}$$

当 ρ_1 和 ρ_2 满足 $\frac{1}{2} > \rho_2 + \frac{1}{2(m-1)}\rho_1$，该估计量为上偏估计，即：$E[\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]] > \mathrm{Var}[\hat{\mu}_{m\times 2}]$。

(3) 第三个估计量 $\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$，对应 $\lambda_1 = \frac{m}{2}$ 且 $\lambda_2 = \frac{m+1}{m}$。

$$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}] \triangleq \frac{1}{2m}\sum_{i=1}^{m}\sum_{k=1}^{2}(\hat{\mu}_k^{(i)} - \hat{\mu}^{(i)})^2 + \frac{m+1}{m(m-1)}\sum_{i=1}^{m}(\hat{\mu}^{(i)} - \hat{\mu}_{m\times 2})^2, \tag{5.15}$$

$$E[\widehat{Var}_3[\hat{\mu}_{m\times 2}]] = \frac{1}{2m}\sigma^2(1+\rho_1) + \frac{m-1}{m}\sigma^2\left[\frac{m}{m-1} - \frac{m+1}{m-1}\rho_2\right]。 \tag{5.16}$$

$$E[\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]] - \mathrm{Var}[\hat{\mu}_{m\times 2}] = \frac{m-1}{m}\sigma^2\left[\frac{m}{m-1} - \frac{2m}{m-1}\rho_2\right]。 \tag{5.17}$$

当 ρ_1 和 ρ_2 满足 $\frac{1}{2} > \rho_2$ 时，该估计量为上偏估计，即：$E[\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]] > \mathrm{Var}[\hat{\mu}_{m\times 2}]$。

注记 5.2　王钰等的研究工作表明[8]，若损失函数不依赖于数据分布及机器学习算法，则 $\rho_2 = 0.5$。然而，$\rho_2 = 0.5$ 为一个理想值。在实际中，损失函数的分布通常与所用算法及数据有关。因此，ρ_2 通常小于 1/2。王钰等所给的实验结果[8] 验证了 $\frac{1}{2} > \rho_2, \rho_1 > 0$。

下面，比较正则化 $m \times 2$ 交叉验证估计的三个方差估计。在算法比较任务中，为得到可靠的统计推断结论，较为保守的方差估计常被优先选择。图5.1 为当 $m = 3, 5, 7, 9$ 且三个方差估计为保守估计时，(ρ_1, ρ_2) 对应的可行域。图5.1 表明在 $0 < \rho_1, \rho_2 < 1/2$ 的大部分区域，$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$ 的保守性无法保证。尽管随着 m 增加，(ρ_1, ρ_2) 的可行域范围增大，但可行域最多可覆盖整个区域的 3/4 （见表5.1）。对于 $\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$，它的表达式简单明了，其上偏条件 $\frac{1}{2} > \rho_2 + \frac{1}{2(m-1)}\rho_1$ 覆盖了 $0 < \rho_1, \rho_2 < 1/2$ 绝大部分范围。当 m 增加时，可行域基本覆盖了整个区域。第三个方差估计，$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$，对应的 (ρ_1, ρ_2) 的可行域覆盖了整个区域。因此，$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$ 为三个估计中最易保持保守性质的估计。

表 5.1　方差的三个估计量的比较

	方差估计量	保守估计的条件	(ρ_1, ρ_2) 可行域所占比例
1	$\widehat{Var}_1(\hat{\mu}_{m\times 2})$	$\frac{1}{2} > \rho_2 + \frac{m+1}{2(m-1)}\rho_1$	$\frac{3m-5}{4(m-1)} \times 100\%$
2	$\widehat{Var}_2(\hat{\mu}_{m\times 2})$	$\frac{1}{2} > \rho_2 + \frac{1}{2(m-1)}\rho_1$	$\frac{4m-5}{4(m-1)} \times 100\%$
3	$\widehat{Var}_3(\hat{\mu}_{m\times 2})$	$\frac{1}{2} > \rho_2$	100%

基于上述分析，推荐 $\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$ 作为 $\mathrm{Var}[\hat{\mu}_{m\times 2}]$ 的估计。第5.7 节中的模拟实

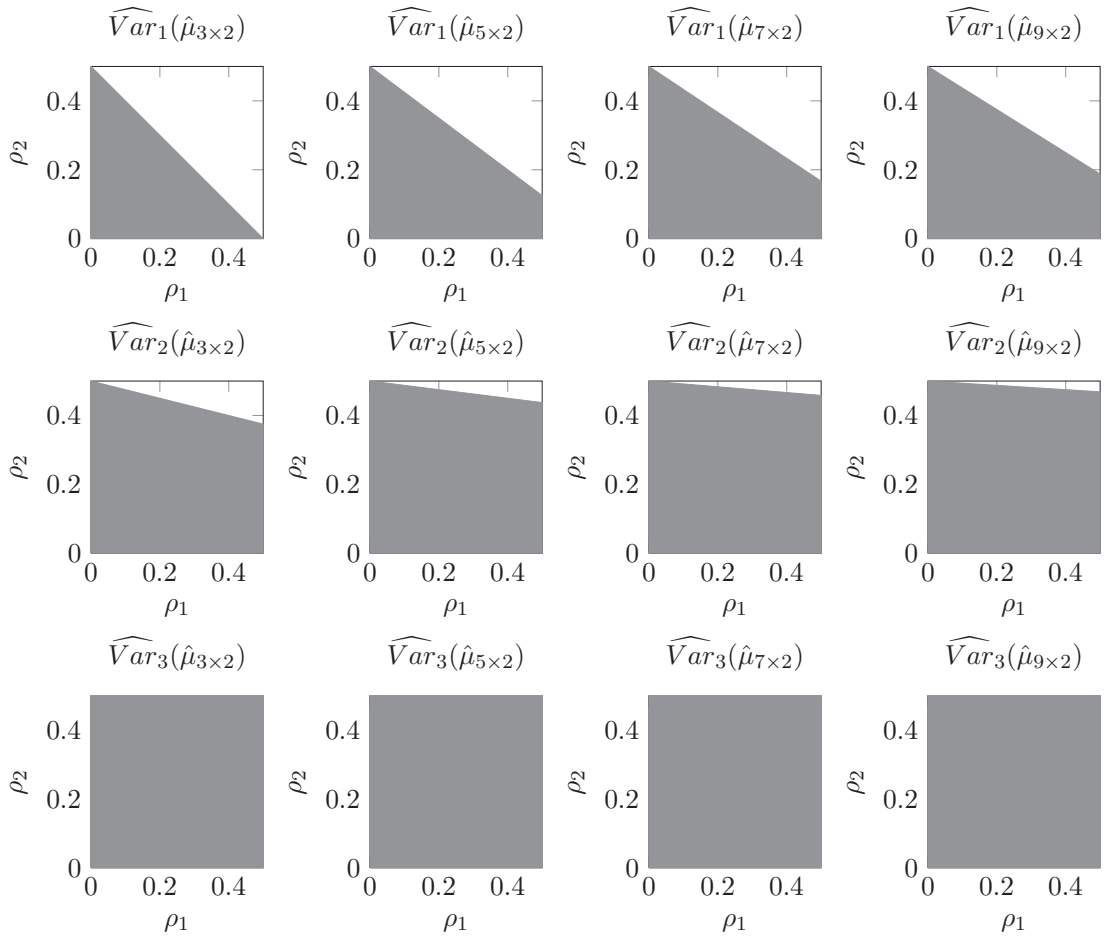

图 5.1 方差估计为保守估计时所对应 (ρ_1, ρ_2) 的可行域

验也同样表明 $\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times2}]$ 为 $\mathrm{Var}[\hat{\mu}_{m\times2}]$ 的一个较为合理的估计。因此，在统计推断中仅使用方差估计 $\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times2}]$，并将其简记为 $\widehat{\mathrm{Var}}[\hat{\mu}_{m\times2}]$。

引理 5.2 假定 $\hat{\boldsymbol{\mu}}_m \sim \mathcal{N}(\mu\mathbf{1}_{2m}, \sigma^2\boldsymbol{\Sigma}_{2m})$，则有

$$\frac{2m}{v_m\sigma^2}\widehat{\mathrm{Var}}[\hat{\mu}_{m\times2}] \sim \chi^2(f_m), \tag{5.18}$$

其中，v_m 和 f_m 分别为：

$$v_m = \frac{m(1-\rho_1)^2 + (m-1)(1+\rho_1-2\rho_2)^2}{2m-(1+\rho_1+2(m-1)\rho_2)}, \tag{5.19}$$

$$f_m = \frac{(2m(1-\rho_2)-(1+\rho_1-2\rho_2))^2}{m(1-\rho_1)^2+(m-1)(1+\rho_1-2\rho_2)^2}。 \tag{5.20}$$

进而，$\widehat{\mathrm{Var}}[\hat{\mu}_{m\times2}]$ 的期望也重写为

$$E[\widehat{\mathrm{Var}}[\hat{\mu}_{m\times2}]] = \frac{1}{2m}v_m f_m \sigma^2。 \tag{5.21}$$

证明. 见第5.9.2 节。 □

5.2.4　基于正则化 $m \times 2$ 交叉验证的 t 检验统计量

定理 5.2　假定 $\hat{\boldsymbol{\mu}}_m \sim \mathcal{N}(\mu \mathbf{1}_{2m}, \sigma^2 \boldsymbol{\Sigma}_{2m})$。则有

$$T_m \triangleq \frac{\hat{\mu}_{m\times 2} - \mu}{C_m \hat{\sigma}_m} \sim t(f_m)。 \tag{5.22}$$

其中 $\hat{\sigma}_m = \sqrt{\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]}$，且 $\hat{\sigma}_m$ 的期望、C_m 和 f_m 分别为：

$$E[\hat{\sigma}_m] = \sqrt{\frac{C_m}{m}} \frac{\Gamma(\frac{f_m+1}{2})}{\Gamma(\frac{f_m}{2})} \sigma, \tag{5.23}$$

$$C_m = \sqrt{\frac{1 + \rho_1 + 2(m-1)\rho_2}{2m - (1 + \rho_1 + 2(m-1)\rho_2)}}, \tag{5.24}$$

$$f_m = \frac{(2m(1-\rho_2) - (1 + \rho_1 - 2\rho_2))^2}{m(1-\rho_1)^2 + (m-1)(1 + \rho_1 - 2\rho_2)^2}。 \tag{5.25}$$

证明. 见第5.9.3 节。 □

由于 T_m 中含有未知参数 ρ_1 和 ρ_2，它不能被直接用作统计量来解决问题（5.1）。这些参数主要影响 T_m 中标准差估计的调节参数 C_m 和 t 分布的自由度 f_m。关于参数 ρ_1 和 ρ_2 对 C_m 和 f_m 的影响，具体分析如下：

- 当 $\rho_1 = \rho_2 = 0$ 时，$\hat{\mu}_{m\times 2}$ 中所有的 hold-out 估计是线性无关的。此时，$C_m = 1/\sqrt{2m}$ 且 $f_m = 2m - 1$。T_m 退化为定义在 $2m$ 个独立的观测上的传统 t 检验统计量。另外，当 $m \to \infty$ 时，σ_m^2 收敛于零，且相应的 μ 的置信区间长度的期望值也收敛于零；
- 当 $\rho_2 = 0$ 但 $\rho_1 > 0$ 时，$\hat{\mu}_{m\times 2}$ 中任意两个 2 折交叉验证估计线性无关，但每个 2 折交叉验证估计中的两个 hold-out 估计正相关。此时，$C_m = \sqrt{(1+\rho_1)/(2m - (1+\rho_1))}$ 且 $f_m = (2m - (1+\rho_1))^2/(m(1-\rho_1)^2 + (m-1)(1+\rho_1)^2)$。当 $m \to \infty$，C_m 收敛于零且 f_m 总是小于 $2m-1$，μ 的置信区间长度的期望值也收敛于零。特别是，当 $\rho_1 = 1$ 时，$C_m = \sqrt{1/(m-1)}$ 且 $f_m = m - 1$。T_m 退化为在 m 个独立同分布观测上定义的传统 t 统计量；
- 当 $\rho_1 > 0$ 且 $\rho_2 > 0$ 时，情况变得相对复杂。具体地，当 $m \to \infty$ 时，C_m 收敛于一个正的常数，即：$\sqrt{\rho_2/(1-\rho_2)} > 0$。显然，该常数与 ρ_1 无关，且该常数说明 ρ_2 在 C_m 中起重要作用。对于 f_m，当 ρ_1 趋于 ρ_2 时，f_m 趋于 $2m-1$。此情形下，μ 的置信区间长度的期望值收敛于正值。王钰等的研究工作[8] 及第三 章的研究表明，此情形为算法比较任务中的常见情形。因此，本章研究主要关注此情形；

为了将 T_m 转化成一个适用于问题（5.1）的检验统计量，需要为 C_m 和 f_m 找到一组合理的设置，且该设置中不应包含未知参数 ρ_1 和 ρ_2。

5.2.5 相关系数 ρ_1 和 ρ_2 的分析

以前的一些研究工作已经分析了相关系数 ρ_1 和 ρ_2 的一些理论性质[8,16]。第三章也对它们进行了初步分析。其中，作为开创性工作，Nadeau 等主要分析了 RLT 估计的相关系数[16]，并且给出了相关系数的合理估计，即：$\rho = n_2/n$，其中，n 为数据集大小，n_2 为验证集大小。对于正则化 $m \times 2$ 交叉验证估计，王钰等的研究[8] 表明，ρ_1 和 ρ_2 的取值依赖于具体的数据分布、机器学习算法、数据集大小及损失函数类型。在严格的假设下，王钰等[8] 证明 $\rho_2 = 1/2$。该假设及相应的结论在如下的引理中给出。

引理 5.3 记 $D_k^{(j)}$ 为估计 $\hat{\mu}_k^{(j)}$ 中所用的训练集。记 $L(\mathcal{A}(D_k^{(j)}), z)$ 为定义在测试样例 z 上的损失函数。假设 $L(\mathcal{A}(D_k^{(j)}), z)$ 的值仅依赖与测试样例 z 及训练集大小。也就是说，损失函数的取值不依赖于特定的机器学习算法及训练集 $D_k^{(j)}$ 中的样本。则，$\rho_1 = 0$ 和 $\rho_2 = 1/2$。

证明. 王钰等已证明 $\rho_2 = 1/2$[8]。$\rho_1 = 0$ 的证明类似。 □

引理5.3中的假设过于严苛，并不适用于真实情况。通常，有很多因素可以影响损失函数的分布，例如，数据总体分布、损失函数的类型、算法类型等。另外，相关系数 ρ_1 和 ρ_2 与算法 $\mathcal{A}$ 和 $\mathcal{B}$ 的相关系数，即，$\rho_{\mathcal{A},1}$，$\rho_{\mathcal{A},2}$，$\rho_{\mathcal{B},1}$ 和 $\rho_{\mathcal{B},2}$ 密切相关。下面，先分析单个算法的相关系数的性质，然后阐述这些系数与 ρ_1 和 ρ_2 的关系。

对于少许算法及数据分布，相关系数 $\rho_{\mathcal{A},1}$ 和 $\rho_{\mathcal{A},2}$ 的表达式可精确写出。例如，对于样本均值回归，当样本集 D_n 为正态分布时，$\rho_{\mathcal{A},1}$ 和 $\rho_{\mathcal{A},2}$ 的理论值在下例给出。

例 5.1 假定数据集为 $D_n = \{(x_i, y_i)\}_{i=1}^n$ 且响应值为 $y_i \sim \mathcal{N}(\mu, \psi^2)$。考虑样本均值回归，将其记作算法 $\mathcal{A}$。在样本均值回归中，决策函数为训练集中所有响应值 y_i 的均值。泛化误差中使用平方损失，则有

$$\rho_{\mathcal{A},1} = \frac{8}{n+6}, \qquad \rho_{\mathcal{A},2} = \frac{n+5}{2n+12}。 \tag{5.26}$$

显然，参数 $\rho_{\mathcal{A},1}$ 和 $\rho_{\mathcal{A},2}$ 的值与 μ 和 ψ^2 无关。当 $n > 11$ 时，可得 $0 < \rho_{\mathcal{A},1} < 1/2$ 且 $0 < \rho_{\mathcal{A},2} < 1/2$。当 $n \to \infty$ 时，$\rho_{\mathcal{A},1} \to 0$ 且 $\rho_{\mathcal{A},2} \to 1/2$。

证明. 具体证明见第5.9.4 节. □

对于单个算法，第三章中阐述了 $m \times 2$ 交叉验证估计的方差与训练集间重叠样本个数之间的关系，并给出了 $m \times 2$ 交叉验证估计中两个 hold-out 估计的协方差函

数的分解式（见引理3.1 的函数 $f(x)$）。从该分解式中，可推导出相关系数 $\rho_{\mathcal{A},1}$ 和 $\rho_{\mathcal{A},2}$ 的上下界，见如下定理。

定理 5.3　假设数据集 D_n 被随机等分为两个子集，分别为训练集 $D^{(t)}$ 和验证集 $D^{(v)}$。设 $L(\mathcal{A}(D^{(t)}), z)$ 为定义在测试样例 z 上的损失函数。将模型 $\mathcal{A}(D^{(t)})$ 在两个独立同分布的测试样例上的损失函数间的线性相关系数定义为

$$\rho_{\mathcal{A},0} \triangleq \frac{\mathrm{Cov}[L(\mathcal{A}(D^{(t)}), z_i), L(\mathcal{A}(D^{(t)}), z_j)]}{\mathrm{Var}[L(\mathcal{A}(D^{(t)}), z_i)]}。 \tag{5.27}$$

其中，$z_i, z_j \in D^{(v)}$ 且 $i \neq j$。则有

(1)

$$\frac{1}{2} - \frac{1}{2} \cdot \frac{1+(n-1)\rho_{\mathcal{A},0}}{2+(n-2)\rho_{\mathcal{A},0}} < \rho_{\mathcal{A},2} < \frac{1+\rho_{\mathcal{A},1}}{2}。 \tag{5.28}$$

(2) 当 $|f'(n/2)| > |f'(0)|$ 时，有

$$\rho_{\mathcal{A},1} < \frac{1+(n-1)\rho_{\mathcal{A},0}}{2+(n-2)\rho_{\mathcal{A},0}}。 \tag{5.29}$$

其中，$f(x)$ 为引理3.1 中定义的协方差函数。

证明. 见第5.9.5 节。 □

注记 5.3　将式（5.29）代入式（5.28），可得

$$\frac{1}{2} - \frac{1}{2} \cdot \frac{1+(n-1)\rho_{\mathcal{A},0}}{2+(n-2)\rho_{\mathcal{A},0}} < \rho_{\mathcal{A},2} < \frac{1}{2} + \frac{1}{2} \cdot \frac{1+(n-1)\rho_{\mathcal{A},0}}{2+(n-2)\rho_{\mathcal{A},0}}。 \tag{5.30}$$

注记 5.4　因 $f(x)$ 为下凸函数，则有 $f'(n/2) > f'(0)$。若 $f'(0) < 0$，则条件 $|f'(0)| < |f'(n/2)|$ 表明 $f'(n/2) > -f'(0)$，即，$f(x)$ 在 $x \to (n/2)^-$ 时的绝对变化速率大于其在 $x \to 0^+$ 时的绝对变化速率。实际上，第三 章的模拟实验验证了 $|f'(0)| < |f'(n/2)|$ 的有效性。

王钰等及本文第三章中提供了关于 $\rho_{\mathcal{A},1}$ 和 $\rho_{\mathcal{A},2}$ 的很多模拟实验[8]。这些实验覆盖了很多种数据集及机器学习算法。这些模拟实验均支持式（5.28）和式（5.29）中相关系数的上下界。另外，这些模拟实验表明数据集 D_n 增大时，$\rho_{\mathcal{A},1}$ 趋于 0，且在很多情形下，$\rho_{\mathcal{A},2}$ 均小于 1/2。然而，这些模拟实验均关注单个学习算法，而非两个算法的性能指标之差。当比较算法 $\mathcal{A}$ 和算法 $\mathcal{B}$ 的性能时，下述引理阐述了 ρ_1 和 ρ_2 与 $\rho_{\mathcal{A},1}$、$\rho_{\mathcal{A},2}$、$\rho_{\mathcal{B},1}$ 及 $\rho_{\mathcal{B},2}$ 之间的关系。

引理 5.4　设算法 $\mathcal{A}$ 和算法 $\mathcal{B}$ 的 hold-out 估计间的相关系数为

$$\rho_{\mathcal{A},\mathcal{B}} = \frac{\mathrm{Cov}[\mu^{(j)}_{\mathcal{A},k}, \mu^{(j')}_{\mathcal{B},k'}]}{\sqrt{\mathrm{Var}[\mu^{(j)}_{\mathcal{A},k}], \mathrm{Var}[\mu^{(j')}_{\mathcal{B},k'}]}}。 \tag{5.31}$$

则可得如下结论。

- 若 $\rho_{\mathcal{A},\mathcal{B}} = 0$，则有 $min(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1}) \leq \rho_1 \leq max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})$ 且 $min(\rho_{\mathcal{A},2}, \rho_{\mathcal{B},2}) \leq \rho_2 \leq max(\rho_{\mathcal{A},2}, \rho_{\mathcal{B},2})$。
- 假定 $\rho_{\mathcal{A},\mathcal{B}}$ 的取值不依赖 k，k'，j 和 j'，且 $\rho_{\mathcal{A},\mathcal{B}} > 0$. 则有 $\rho_1 \leq max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})$ 且 $\rho_2 \leq max(\rho_{\mathcal{A},2}, \rho_{\mathcal{B},2})$。

证明. 见第5.9.6节. □

引理5.4表明两个算法的差的相关系数小于单个算法相关系数的最大值。第5.8.2节给出了相应的模拟实验。

5.2.5.1 参数 C_m 和 f_m 的设置

在给出 C_m 和 f_m 的设置之前，先阐述显著性检验的保守性质。**保守的显著性检验**指当原假设为真时，该显著性检验拒绝原假设的概率小于名义上的显著性水平 α。反之，**激进的显著性检验**表明该显著性检验拒绝原假设的概率大于名义上的显著性水平 α。

算法比较任务应青睐较为保守的检验，因为保守的检验可以减少假阳性结论的发生，提高算法比较结论的可靠性[8,16]。为了得到一个相对保守的序贯 t 检验，需要为 T_m 中的参数 C_m 和 f_m 设置一组合理的取值。本章给出如下设置。

- **参数 C_m 的设置**：当 ρ_1 和 ρ_2 增加时，C_m 随之增加。因此，在 C_m 中，设置 $\rho_1 = \rho_2 = 0.5$，可得
$$\hat{C}_m \triangleq \sqrt{\frac{2m+1}{2m-1}}。 \tag{5.32}$$
- **参数 f_m 的设置**：根据 f_m 的定义，可得 $\infty > f_m > m$，其中，$0 < \rho_1, \rho_2 < 1$。然而，当 $f_m = m$ 时，相应检验过度保守。另外，参考王钰等给出的组块 3×2 交叉验证 t 检验的自由度[8]，设置 f_m 为
$$\hat{f}_m \triangleq 2m - 1。 \tag{5.33}$$
当 $\rho_1 = \rho_2$ 时，$f_m = 2m - 1$。

如上所述，对于 $\hat{C}_m$ 和 $\hat{f}_m$，T_m 为保守 t 检验统计量的充要条件为 $\hat{C}_m \cdot t_\alpha(\hat{f}_m) > C_m \cdot t_\alpha(f_m)$。将该条件成立时 ρ_1 和 ρ_2 所在区域记为
$$\Theta_{m,\alpha} \triangleq \left\{(\rho_1, \rho_2) : \hat{C}_m \cdot t_\alpha(\hat{f}_m) > C_m \cdot t_\alpha(f_m)\right\}。 \tag{5.34}$$
其中 $0 < \rho_1, \rho_2 < 1$。图5.2给出了在 m 和 α 的一些取值下，区域 $\Theta_{m,\alpha}$ 的范围。图5.2 的每幅子图中，垂直线和水平线分别对应 $\rho_1 = 1/2$ 和 $\rho_2 = 1/2$。着色区域（包括浅色区域和深色区域）表示 $\Theta_{m,\alpha}$。当 m 增大时，$\Theta_{m,\alpha}$ 范围变大，且其上边界变

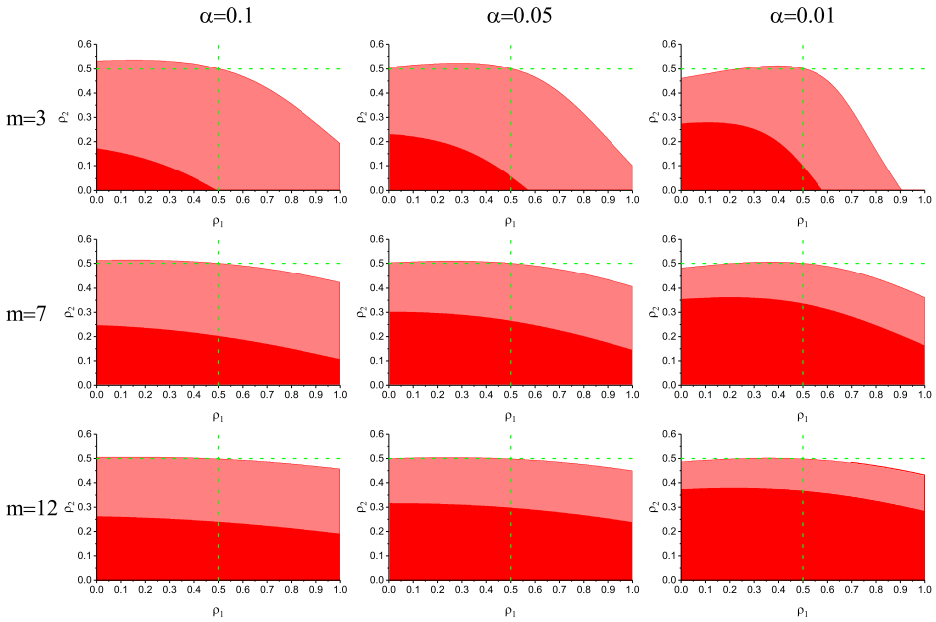

图 5.2　相关系数 ρ_1 和 ρ_2 的保守区域

得平坦。当 m 趋于无穷时，$\Theta_{m,\alpha}$ 趋于 $\Theta_{\infty,\alpha} = \{(\rho_1, \rho_2) : \rho_1 \in (0,1), \rho_2 \in (0, 1/2)\}$。这表明当 m 较大时，t 检验在 $0 < \rho_2 < 0.5$ 区域内为保守检验。另外，当 α 减少时，$\Theta_{m,\alpha}$ 的范围收缩。不过，在 m 和 α 的各组取值下，$\Theta_{m,\alpha}$ 基本覆盖了区域 $0 \leq \rho_1, \rho_2 \leq 1/2$。

在保守的显著性检验中，过度保守的检验亦可能发生，此时，原假设的拒绝概率变得很小。本章地引入经验概率 $\alpha/5$ 作为过度保守检验对应的拒绝概率，并将区域 $\Theta_{m,\alpha}$ 分割为如下两个子区域。

- **相对保守区域：**记该区域为 $\Theta^{(R)}_{m,\alpha} \triangleq \{(\rho_1, \rho_2) : C_m t_{\alpha/5}(f_m) > \hat{C}_m t_\alpha(\hat{f}_m) > C_m t_\alpha(f_m)\}$。在图5.2 中，该区域对应于浅色区域。当 m 增加时，该区域的下边界上升，且该区域变窄。当 α 减少时，该区域收缩，且对 ρ_2 的约束越发严格。当 m 趋于无穷时，该区域趋近于 $\Theta^{(R)}_{\infty,\alpha} = \{(\rho_1, \rho_2) : \sqrt{\rho_2} u_{\alpha/5} > \sqrt{1-\rho_2} u_\alpha > \sqrt{\rho_2} u_\alpha, 1 > \rho_1 > 0\}$。其中，$u_\alpha$ 为标准正态分布的上侧 α 分位数。例如，$\Theta^{(R)}_{\infty,0.1} = \{(\rho_1, \rho_2) : \rho_1 \in (0,1), \rho_2 \in (0.28, 0.50)\}$ 且 $\Theta^{(R)}_{\infty,0.05} = \{(\rho_1, \rho_2) : \rho_1 \in (0,1), \rho_2 \in (0.33, 0.50)\}$；
- **过度保守区域：**记该区域为 $\Theta^{(O)}_{m,\alpha} \triangleq \{(\rho_1, \rho_2) : \hat{C}_m t_\alpha(\hat{f}_m) > C_m t_{\alpha/5}(f_m)\}$。在图5.2 中，该区域对应于深色区域。若 ρ_1 和 ρ_2 的取值位于该区域内，则当置信水平为 $\alpha = 0.05$ 时，第一类错误实为 $\alpha < 0.01$。当 m 趋于无穷时，该区域趋近于 $\Theta^{(O)}_{\infty,\alpha} = \{(\rho_1, \rho_2) : \sqrt{1-\rho_2} u_\alpha > \sqrt{\rho_2} u_{\alpha/5}, 1 > \rho_1 > 0\}$。例如，

$\Theta^{(O)}_{\infty,0.1} = \{(\rho_1,\rho_2) : \rho_1 \in (0,1), \rho_2 \in (0,0.28)\}$，且 $\Theta^{(O)}_{\infty,0.05} = \{(\rho_1,\rho_2) : \rho_1 \in (0,1), \rho_2 \in (0,0.33)\}$；

表5.2和5.3 中给出了 f_m 和 C_m 的取值，以及在 ρ_1 和 ρ_2 的不同水平下，$\hat{f}_m$ 和 $\hat{C}_m$ 的取值。从这两个表中，可以观察到：第一，随着 m 增加，C_m 和 $\hat{C}_m$ 均下降，且当 $0 \le \rho_1, \rho_2 \le 1/2$ 时，$\hat{C}_m \ge C_m$；第二，f_m 随着 ρ_2 的增加而减少，但其随着 ρ_1 和 m 的增加而增加。特别是，当 $\rho_1 = \rho_2$ 时，f_m 退化为 $2m-1$；当 $\rho_2 = 0.5$ 和 $\rho_1 = 0$ 时，f_m 退化为 m。

表 5.2 在 ρ_1、ρ_2 和 m 的不同水平下，C_m、$\hat{C}_m$、f_m 和 $\hat{f}_m$ 的取值

m	ρ_1	C_m						$\hat{C}_m$	f_m						$\hat{f}_m$
		0.0	0.1	0.2	0.3	0.4	0.5		0.0	0.1	0.2	0.3	0.4	0.5	
3	0.0	0.45	0.55	0.65	0.76	0.87	1.00	1.18	5.00	4.94	4.74	4.35	3.75	3.00	5.00
	0.1	0.47	0.58	0.68	0.79	0.90	1.03		4.95	5.00	4.93	4.67	4.17	3.43	
	0.2	0.50	0.60	0.71	0.82	0.94	1.07		4.80	4.94	5.00	4.91	4.57	3.92	
	0.3	0.53	0.63	0.73	0.85	0.97	1.11		4.55	4.75	4.92	5.00	4.88	4.42	
	0.4	0.55	0.65	0.76	0.87	1.00	1.14		4.23	4.45	4.69	4.90	5.00	4.83	
	0.5	0.58	0.68	0.79	0.90	1.03	1.18		3.86	4.07	4.32	4.59	4.86	5.00	
4	0.0	0.38	0.50	0.62	0.73	0.86	1.00	1.13	7.00	6.92	6.62	6.04	5.14	4.00	7.00
	0.1	0.40	0.52	0.64	0.75	0.88	1.03		6.93	7.00	6.90	6.52	5.77	4.65	
	0.2	0.42	0.54	0.65	0.77	0.90	1.05		6.72	6.91	7.00	6.87	6.37	5.39	
	0.3	0.44	0.56	0.67	0.80	0.93	1.08		6.39	6.66	6.89	7.00	6.82	6.14	
	0.4	0.46	0.58	0.69	0.82	0.95	1.11		5.95	6.25	6.57	6.86	7.00	6.75	
	0.5	0.48	0.60	0.71	0.84	0.98	1.13		5.45	5.73	6.07	6.44	6.81	7.00	
5	0.0	0.33	0.47	0.59	0.72	0.85	1.00	1.11	9.00	8.89	8.50	7.72	6.52	5.00	9.00
	0.1	0.35	0.48	0.61	0.73	0.87	1.02		8.91	9.00	8.87	8.37	7.37	5.87	
	0.2	0.37	0.50	0.62	0.75	0.89	1.04		8.64	8.89	9.00	8.83	8.17	6.86	
	0.3	0.39	0.52	0.64	0.77	0.90	1.06		8.22	8.56	8.86	9.00	8.77	7.86	
	0.4	0.40	0.53	0.65	0.78	0.92	1.08		7.67	8.05	8.45	8.82	9.00	8.67	
	0.5	0.42	0.55	0.67	0.80	0.94	1.11		7.05	7.40	7.82	8.29	8.75	9.00	
6	0.0	0.30	0.45	0.58	0.71	0.85	1.00	1.09	11.00	10.87	10.38	9.41	7.90	6.00	11.00
	0.1	0.32	0.46	0.59	0.72	0.86	1.02		10.89	11.00	10.84	10.21	8.97	7.09	
	0.2	0.33	0.47	0.60	0.73	0.87	1.03		10.57	10.86	11.00	10.79	9.97	8.33	
	0.3	0.35	0.49	0.62	0.75	0.89	1.05		10.05	10.47	10.83	11.00	10.71	9.58	
	0.4	0.36	0.50	0.63	0.76	0.90	1.07		9.39	9.85	10.33	10.78	11.00	10.59	
	0.5	0.38	0.51	0.64	0.77	0.92	1.09		8.65	9.07	9.57	10.14	10.70	11.00	
7	0.0	0.28	0.43	0.57	0.70	0.84	1.00	1.07	13.00	12.85	12.27	11.10	9.29	7.00	13.00
	0.1	0.29	0.44	0.58	0.71	0.85	1.01		12.87	13.00	12.80	12.06	10.57	8.31	
	0.2	0.31	0.45	0.59	0.72	0.87	1.03		12.49	12.84	13.00	12.75	11.76	9.80	
	0.3	0.32	0.47	0.60	0.73	0.88	1.04		11.89	12.37	12.80	13.00	12.66	11.31	
	0.4	0.33	0.48	0.61	0.75	0.89	1.06		11.12	11.65	12.21	12.74	13.00	12.52	
	0.5	0.35	0.49	0.62	0.76	0.90	1.07		10.25	10.74	11.32	11.98	12.64	13.00	

表 5.3　在 ρ_1、ρ_2 和 m 的不同水平下，C_m、$\hat{C}_m$、f_m 和 $\hat{f}_m$ 的取值（续）

8	0.0	0.26	0.42	0.56	0.69	0.84	1.00	1.06	15.00	14.82	14.15	12.79	10.67	8.00	15.00
	0.1	0.27	0.43	0.57	0.70	0.85	1.01		14.85	15.00	14.77	13.91	12.16	9.53	
	0.2	0.28	0.44	0.58	0.71	0.86	1.03		14.41	14.82	15.00	14.71	13.56	11.27	
	0.3	0.30	0.45	0.59	0.72	0.87	1.04		13.72	14.28	14.77	15.00	14.60	13.03	
	0.4	0.31	0.46	0.60	0.73	0.88	1.05		12.84	13.44	14.09	14.70	15.00	14.44	
	0.5	0.32	0.47	0.61	0.74	0.89	1.06		11.85	12.41	13.07	13.83	14.59	15.00	
9	0.0	0.24	0.41	0.55	0.69	0.84	1.00	1.06	17.00	16.80	16.03	14.48	12.06	9.00	17.00
	0.1	0.26	0.42	0.56	0.70	0.85	1.01		16.83	17.00	16.74	15.76	13.76	10.75	
	0.2	0.27	0.43	0.57	0.71	0.85	1.02		16.33	16.79	17.00	16.67	15.36	12.74	
	0.3	0.28	0.44	0.58	0.72	0.86	1.03		15.55	16.18	16.74	17.00	16.55	14.75	
	0.4	0.29	0.45	0.59	0.72	0.87	1.05		14.56	15.24	15.98	16.66	17.00	16.36	
	0.5	0.30	0.46	0.59	0.73	0.88	1.06		13.44	14.08	14.83	15.68	16.53	17.00	
10	0.0	0.23	0.40	0.55	0.69	0.83	1.00	1.05	19.00	18.77	17.91	16.17	13.44	10.00	19.00
	0.1	0.24	0.41	0.55	0.69	0.84	1.01		18.81	19.00	18.71	17.61	15.36	11.97	
	0.2	0.25	0.42	0.56	0.70	0.85	1.02		18.26	18.77	19.00	18.63	17.16	14.21	
	0.3	0.26	0.43	0.57	0.71	0.86	1.03		17.39	18.09	18.70	19.00	18.50	16.48	
	0.4	0.27	0.44	0.58	0.72	0.87	1.04		16.29	17.04	17.86	18.62	19.00	18.29	
	0.5	0.28	0.44	0.59	0.73	0.88	1.05		15.04	15.75	16.58	17.53	18.48	19.00	
11	0.0	0.22	0.40	0.54	0.68	0.83	1.00	1.05	21.00	20.75	19.79	17.86	14.82	11.00	21.00
	0.1	0.23	0.40	0.55	0.69	0.84	1.01		20.79	21.00	20.68	19.46	16.96	13.19	
	0.2	0.24	0.41	0.56	0.70	0.85	1.02		20.18	20.74	21.00	20.59	18.96	15.68	
	0.3	0.25	0.42	0.56	0.70	0.86	1.03		19.22	19.99	20.67	21.00	20.44	18.20	
	0.4	0.26	0.43	0.57	0.71	0.86	1.04		18.01	18.84	19.74	20.58	21.00	20.21	
	0.5	0.27	0.43	0.58	0.72	0.87	1.05		16.64	17.42	18.33	19.38	20.42	21.00	

5.2.5.2　t 检验统计量及其序贯置信区间

将 $\hat{C}_m$ 和 $\hat{f}_m$ 代入 T_m （见式（5.22）），得如下 t 检验统计量。

$$T_m^{(S)} \triangleq \frac{\hat{\mu}_{m\times 2} - \mu}{\hat{C}_m \hat{\sigma}_m} \sim t_\alpha(\hat{f}_m)。 \tag{5.35}$$

其中 $\hat{\sigma}_m = \sqrt{\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]}$，$\hat{C}_m = \sqrt{(2m+1)/(2m-1)}$ 且 $\hat{f}_m = 2m-1$。上标 $^{(S)}$ 用来区分含未知参数的统计量 T_m 与检验统计量 $T_m^{(S)}$。

基于检验统计量 $T_m^{(S)}$，μ 的相应 $1-\alpha$ 置信区间为

$$\mathrm{CI}_m \triangleq (\hat{\mu}_{m\times 2} - \hat{C}_m \hat{\sigma}_m t_{\alpha/2}(\hat{f}_m), \quad \hat{\mu}_{m\times 2} + \hat{C}_m \hat{\sigma}_m t_{\alpha/2}(\hat{f}_m))。 \tag{5.36}$$

其中 $t_{\alpha/2}(\hat{f}_m)$ 为自由度为 $\hat{f}_m$ 的标准 t 分布的上侧 $\alpha/2$ 分位数。$\mu_{\mathcal{A}}$ 和 $\mu_{\mathcal{B}}$ 的置信区间可类似定义，并分别记为 $\mathrm{CI}_{\mathcal{A},m}$ 和 $\mathrm{CI}_{\mathcal{B},m}$。

下述定理给出了置信区间 CI_m 的一些理论性质。

定理 5.4　当 $m \geq 2$ 时，置信区间 CI_m 有如下三条理论性质。

(1) 随着 m 的增加，CI_m 长度的期望值减少，即

$$E[2\hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)] < E[2\hat{C}_{m+1}\hat{\sigma}_{m+1} t_{\alpha/2}(\hat{f}_{m+1})]; \tag{5.37}$$

(2) CI_m 长度的期望值下界为 $2\sqrt{1-\rho_2}\sigma u_{\alpha/2}$，即

$$\lim_{m\to\infty} E[2\hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)] = 2\sqrt{1-\rho_2}\sigma u_{\alpha/2}。 \tag{5.38}$$

其中，$u_{\alpha/2}$ 为标准正态分布的上侧 $\alpha/2$ 分位数；

(3) 当 $\rho_2 < 1/2$ 时，CI_m 为保守置信区间，即：

$$\lim_{m\to\infty} P(\mu \in \mathrm{CI}_m) > 1-\alpha。 \tag{5.39}$$

证明. 见第5.9.7 节。 □

基于定理5.4，称置信区间 CI_m 为 μ 关于 m 的**序贯置信区间**。

5.3 正则化 $m \times 2$ 交叉验证序贯 t 检验

通常，传统的序贯检验[95] 由三部分组成：决策边界、停止时间（停时）及决策规则。对于假设检验问题（5.1），基于序贯置信区间 CI_m，可构造相应的序贯检验。该序贯检验含如下三部分：

(1) **决策边界**：I_U

$$I_U \triangleq \Delta + \hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)。 \tag{5.40}$$

其中，$\hat{\sigma}_m = \sqrt{\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]}$，且 $t_{\alpha/2}$ 为标准 t 分布的上侧 $\alpha/2$ 分位数；

(2) **停时**：m_{stop}

在传统的序贯检验中，m_{stop} 的定义为 $m_{stop} \triangleq \inf\{m : m \geq m_{start} \bigwedge \hat{\mu}_{m\times 2} > I_U\}$。也就是说，停时对应于 $\hat{\mu}_{m\times 2}$ 首次超过决策边界的时刻。m_{start} 为 m 的开始时间。然而，由于 CI_m 长度的期望值收敛于常数 $2\sqrt{1-\rho_2}\sigma u_{\alpha/2}$ 而不是零（见定理5.4 中的第二条性质），上述停时的定义不不能保证基于统计量 $T_m^{(S)}$ 的序贯检验可在有限时刻内停止（见第5.4 节）。因此，引入一个合理的阈值 M 来强迫序贯检验在 $m \geq M$ 时停止。本节定义如下 m_{stop}。

$$m_{stop} = \min\left\{\inf\{m : m \geq m_{start} \bigwedge \hat{\mu}_{m\times 2} > I_U\}, M\right\}。 \tag{5.41}$$

(3) **决策规则**：

显著性检验问题5.1 的决策可通过如下规则给出。

- 若 $\hat{\mu}_{m\times 2} > I_U$，则拒绝 H_0。这表明算法 $\mathcal{B}$ 的性能比算法 $\mathcal{A}$ 显著高 Δ；

- 否则，不能拒绝 H_0。也就是说，数据集 D_n 不能提供足够的信息来判别 $\mathcal{A}$ 和算法 $\mathcal{B}$ 间的性能指标之差是否大于 Δ；

算法5.1给出了正则化 $m \times 2$ 交叉验证序贯 t 检验的算法轮廓。在该算法中，正则化 $m \times 2$ 交叉验证由第三 章所给的增量式构造方法来构造。该构造方法可增量式地产生每个切分，并依次纳入检验中，使得序贯检验方法高效执行。

算法 5.1 正则化 $m \times 2$ 交叉验证序贯 t 检验

输入: 数据集，D_n；
待比较的算法，$\mathcal{A}$ 和 $\mathcal{B}$；
开始时刻，m_{start}；
假设检验问题（5.1）中的阈值，Δ；
显著性水平，α；

输出: 假设检验问题（5.1）的结论：“拒绝 H_0”或不拒绝“H_0”；

初始化 $\hat{\boldsymbol{\mu}} = ()$ 以存储估计 $\hat{\mu}_{m\times 2}$ 中所有的 hold-out 估计；
for all $m = m_{start}; m \leq M; m++$ **do**
　根据第三 章中的增量式构造算法增量地生成 $\mathbb{S}^b$ 中的正则化切分；
　从 $\mathbb{S}^b$ 中取出未使用的所有切分，将其记作 $\mathbb{S}^b_u$；
　基于 D_n，$\mathcal{A}$，$\mathcal{B}$ 和 $\mathbb{S}^b_u$，计算新的 hold-out 估计 $\hat{\mu}^{(j)}_k$；
　将新的 hold-out 估计 $\hat{\mu}^{(j)}_k$ 追加至向量 $\hat{\boldsymbol{\mu}}$ 中；
　基于 $\hat{\boldsymbol{\mu}}$ 中所有的 hold-out 估计，根据式（5.2），计算 $\hat{\mu}_{m\times 2}$；
　基于 $\hat{\boldsymbol{\mu}}$ 中所有的 hold-out 估计，根据式（5.12），计算 $\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]$；
　根据式（5.32）和式（5.33），计算 $\hat{C}_m$ 和 $\hat{f}_m$；
　根据式（5.40），基于显著性水平 α，计算决策边界 I_U；
　if $\hat{\mu}_{m\times 2} > I_U$ **then**
　　return “拒绝 H_0”；
　end if
end for
return “不拒绝 H_0”；

下述定理给出正则化 $m \times 2$ 交叉验证序贯 t 检验的第一类错误的理论性质。

定理 5.5　当 $(\rho_1, \rho_2) \in \Theta_{m,\alpha} \cap \Theta_{m,\beta}$ 时，正则化 $m \times 2$ 交叉验证序贯 t 检验的第一类错误小于 α。

证明. 根据式（5.34），当 $(\rho_1, \rho_2) \in \Theta_{m,\alpha}$ 时, 有 $\hat{C}_m t_\alpha(\hat{f}_m) > C_m t_\alpha(f_m)$ 成立。当 H_0 成立时，拒绝 H_0 的概率（第一类错误）为

$$\begin{aligned} P(\text{reject } H_0 | \mu < \Delta, m = E[m_{stop}]) &= P(\hat{\mu}_{m\times 2} \geq \Delta + \widehat{C}_m \cdot \hat{\sigma}_m t_\alpha(\hat{f}_m) | \mu < \Delta, m = E[m_{stop}]) \\ &< P(\hat{\mu}_{m\times 2} > \mu + \widehat{C}_m \cdot \hat{\sigma}_m t_\alpha(\hat{f}_m) | \mu < \Delta, m = E[m_{stop}]) \\ &\leq P(\hat{\mu}_{m\times 2} > \mu + C_m \cdot \hat{\sigma}_m t_\alpha(f_m) | \mu < \Delta, m = E[m_{stop}]) \end{aligned}$$

$$
\begin{aligned}
&= P(\frac{\hat{\mu}_{m\times 2}-\mu}{C_m\cdot\hat{\sigma}_m} > t_\alpha(f_m)|\mu<\Delta, m=E[m_{stop}]) \\
&= \alpha。
\end{aligned} \tag{5.42}
$$

□

5.4 停时 m_{stop} 的分析

在序贯假设检验中，一个重要的问题是：停时 m_{stop} 是否为有界值？也就是说，对于一个有限的正常数 c, 式 $P(\hat{\mu}_{m\times 2} > I_U|m_{stop}\leq c<\infty)=1$ 是否成立？另外，如果希望正则化 $m\times 2$ 交叉验证序贯 t 检验最多在 M 步内停止，那么，如何合理地估计 M？

实际上，在传统的序贯检验中，当样本独立同分布时，可以从理论上保证停时为有界的正整数[95]。也就是说，序贯检验方法可以在有限的时刻内在假设 H_0 和假设 H_1 之间做一个决策。然而，对于正则化 $m\times 2$ 交叉验证序贯 t 检验，并不能从理论上保证停时 m_{stop} 为有限值。其主要原因是 $\hat{\mu}_{m\times 2}$ 中所有的 hold-out 估计 $\hat{\mu}_k^{(j)}$ 均正相关，且当 m 趋于无穷时，序贯置信区间 CI_m 的期望长度不收敛与零（参考定理5.4）。因此，本章引入一个预先定义的最大停时 M 来保证正则化 $m\times 2$ 交叉验证序贯 t 检验在有限时刻内可以停止。也就是说，当正则化 $m\times 2$ 交叉验证序贯 t 检验执行时，若执行时刻达到 M 且未拒绝 H_0，则该检验自动停止且输出“未拒绝 H_0”。这意味着没有足够的证据可以拒绝 H_0。

起始时刻 m_{start} 和最大停时 M 的设置准则如下：

- 对于开始时刻 m_{start}，本文推荐 $m_{start}=3$，因为基于组块 3×2 交叉验证的 t 检验已经在王钰等的研究工作[8] 中被深入研究。该研究工作表明组块 3×2 交叉验证 t 检验具有比 5×2 交叉验证成对 t 检验更高的势和更好的复现度[1]；
- 最大停时 M 可以基于置信区间长度的期望约减率（RRCI_α）来估计。该期望约减率的形式如下：

$$
\mathrm{RRCI}_\alpha(m) \triangleq \frac{E[\hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)] - E[\hat{C}_{m+1}\hat{\sigma}_{m+1}t_{\alpha/2}(\hat{f}_{m+1})]}{E[\hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)]}。 \tag{5.43}
$$

若 RRCI_α 较小，例如，其小于 γ （5% 或 1%），额外的切分对于序贯检验的贡献很小，因此，这些切分便可以被忽略掉。基于该想法，可以解出 M 的具体值。然而，RRCI_α 与未知参数 ρ_1 和 ρ_2 有关。不过，可以使用 RRCI_α 在 $0<\rho_1,\rho_2<1/2$ 范围内的期望值。该期望值被称为置信区间长度的平均约减

表 5.4　在 m 和 α 的不同水平下 $\mathrm{ARRCI}_\alpha(m)$ 的取值

m	ARRCI_α									
	$\alpha=0.1$	γ	$\alpha=0.05$	γ	$\alpha=0.01$	γ	$\alpha=0.005$	γ	$\alpha=0.001$	γ
3	0.0989	<10%	0.1184		0.1682		0.1910		0.2455	
4	0.0595		0.0701	<10%	0.0973	<10%	0.1099		0.1406	
5	0.0401	<5%	0.0467	<5%	0.0636		0.0715	<10%	0.0907	<10%
6	0.0291		0.0336		0.0450	<5%	0.0503		0.0634	
7	0.0221		0.0254		0.0337		0.0375	<5%	0.0468	<5%
8	0.0175		0.0199		0.0262		0.0290		0.0360	
9	0.0142		0.0161		0.0210		0.0232		0.0286	
10	0.0118		0.0133		0.0172		0.0190		0.0233	
11	0.0099	<1%	0.0112		0.0144		0.0159		0.0194	
12	0.0085		0.0096	<1%	0.0122		0.0134		0.0164	
13	0.0074		0.0083		0.0105		0.0116		0.0140	
14	0.0065		0.0072		0.0092	<1%	0.0100		0.0122	
15	0.0057		0.0064		0.0081		0.0088	<1%	0.0107	
16	0.0051		0.0057		0.0071		0.0078		0.0094	<1%
17	0.0046		0.0051		0.0064		0.0070		0.0084	
18	0.0041		0.0046		0.0057		0.0063		0.0075	
19	0.0037		0.0042		0.0052		0.0057		0.0068	
20	0.0034		0.0038		0.0047		0.0051		0.0061	

率，记为 ARRCI_α。其表达式为

$$\mathrm{ARRCI}_\alpha(m) \triangleq 4\int_0^{\frac{1}{2}}\int_0^{\frac{1}{2}} 1-\frac{E[\hat{C}_{m+1}\hat{\sigma}_{m+1}t_{\alpha/2}(\hat{f}_{m+1})]}{E[\hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)]}d\rho_1 d\rho_2。 \tag{5.44}$$

给定显著性水平 α，建议 M 的选取准则为

$$M=\max_m\{m:\mathrm{ARRCI}_\alpha(m)\le\gamma\times 100\%\}。 \tag{5.45}$$

其中，γ 为预先设定的参数。一些常用的 M 值和 ARRCI_α 在表5.4 中给出。

表5.4表明当显著性水平 α 减少时，停时 M 随之增加。显著性水平 α 的取值较小时，检验中需要更多的切分。当 $\alpha=0.05$ 时，若希望 ARRCI_α 小于 $\gamma=5\%$，则 M 的值应为大于五，这可能是算法比较任务中 5×2 交叉验证被青睐 [1,6,7] 的原因之一。

5.5　算法比较任务中现有的 t 检验

对于算法比较任务，现有的 t 主要包括 5×2 交叉验证成对 t 检验[1]、合并 5×2 交叉验证 t 检验[7] 以及组块 3×2 交叉验证 t 检验[8] 等。这些检验方法可用

正则化 $m \times 2$ 交叉验证扩展，以便于解决问题（5.1）。

另外，合并 5×2 交叉验证 F 检验[6] 和校准的均衡正则化 5×2 交叉验证 F 检验[9] 为 5×2 交叉验证成对 t 检验的两个重要改进版本。但是，F 检验不对应明确的置信区间，因此，这两种方法不适用于问题（5.1）。

下面几小节深入分析了上述的几种 t 检验方法。

5.5.1 5×2 交叉验证成对 t 检验

5×2 交叉验证成对 t 检验[1] 使用五次重复的 2 折交叉验证来构建相应的检验统计量：

$$T_{5\times 2CV} = \frac{\hat{\mu}_1^{(1)} - \mu}{\sqrt{\sum_{i=1}^{5} S_i^2/5}} \sim t(5)。 \tag{5.46}$$

其中，$\hat{\mu}_1^{(1)}$ 为第一个 2 折交叉验证中的 hold-out 估计，且 $S_i^2 = (\hat{\mu}_1^{(i)} - \hat{\mu}^{(i)})^2 + (\hat{\mu}_2^{(i)} - \hat{\mu}^{(i)})^2$ 为第 i 个 2 折交叉验证估计的样本方差估计。在正态假设下，$\hat{\mu}_1^{(1)}$ 和 $\sum_{i=1}^{5} S_i^2/5$ 为 t 检验统计量的两部分。假设五个 2 折交叉验证估计相互独立，则该检验统计量的自由度为五。不过，该检验的一个明显的弱点是复现度较低[8,42]。

基于正则化 $m \times 2$ 交叉验证，上述的 t 统计量可以泛化为如下的形式：

$$T_m^{(P)} = \frac{\hat{\mu}_1^{(1)} - \mu}{\sqrt{\sum_{i=1}^{m} S_i^2/m}} \sim t(m)。 \tag{5.47}$$

统计量 $T_m^{(P)}$ 应该自适应地选择切分重复次数，而不是将其经验性地设置为五。基于统计量 $T_m^{(P)}$，μ 的 $1-\alpha$ 序贯置信区间为

$$\mathrm{CI}_m^{(P)} \triangleq \left(\hat{\mu}_1^{(1)} - t_{\alpha/2}(m) \sqrt{\sum_{i=1}^{m} S_i^2/m}, \quad \hat{\mu}_1^{(1)} + t_{\alpha/2}(m) \sqrt{\sum_{i=1}^{m} S_i^2/m} \right)。 \tag{5.48}$$

序贯置信区间 $\mathrm{CI}_m^{(P)}$ 的长度为 $2t_{\alpha/2}(m)\sqrt{\sum_{i=1}^{m} S_i^2/m}$。对于 $0 < \rho_1, \rho_2 < 1/2$，当 $m \to \infty$ 时，$\mathrm{CI}_m^{(P)}$ 的期望长度收敛于 $2\sqrt{1-\rho_1}\sigma u_{\sigma/2}$，而不是零。进而，若 $\rho_1 > 0$，则基于 $\mathrm{CI}_m^{(P)}$ 的显著性检验是激进的。

基于 $\mathrm{CI}^{(P)}$，相应的决策边界 $I_U^{(P)}$ 为

$$I_U^{(P)} = \Delta_1 + t_{1-\beta}(m) \sqrt{\sum_{i=1}^{m} S_i^2/m}。 \tag{5.49}$$

基于 $T_m^{(P)}$ 和 $I_U^{(P)}$，可定义一个与算法5.1 相似的序贯检验。该算法通过比较 $\hat{\mu}_1^{(1)}$ 与 $I_U^{(P)}$ 来决定原假设 H_0 是否被拒绝。该序贯检验方法被称为**正则化 $m \times 2$ 交叉验证成对序贯 t 检验**。

注记 5.5 正则化 $m \times 2$ 交叉验证成对序贯 t 检验具有如下一些不足：

(1) 当 $\rho_1 > 0$，基于 $\mathrm{CI}_m^{(P)}$ 的检验是激进的；

(2) 因为 $T_m^{(P)}$ 的分子和分母不独立，统计量 $T_m^{(P)}$ 不严格服从 t 分布。另外，即便 m 趋于无穷，$T_m^{(P)}$ 也不为一个合理的 t 统计量；

(3) $T_m^{(P)}$ 的分子仅使用 $\hat{\mu}_1^{(1)}$，并没有使用其它 hold-out 估计 $\hat{\mu}_k^{(j)}$。因此，该检验的结果容易受数据集的特定切分的影响。因此，该检验的复现度较低；

(4) 该检验要求单个 hold-out 估计 $\hat{\mu}_k^{(j)}$ 需服从正态分布。与正则化 $m \times 2$ 交叉验证序贯 t 检验相比，该假设较为严格。因为正则化 $m \times 2$ 交叉验证序贯 t 检验仅假设聚合估计 $\hat{\mu}_{m\times 2}$ 为正态分布。基于极限中心定理，当 m 较大时，聚合估计更容易服从正态分布；

5.5.2　合并 5×2 交叉验证成对 t 检验

5×2 交叉验证成对 t 检验的一个改进版本为合并 5×2 交叉验证成对 t 检验[7]。其检验统计量如下：

$$T_{5\times 2CCV} = \frac{\hat{\mu}_{5\times 2} - \mu}{\sqrt{\frac{1}{10}\sum_{i=1}^{5} S_i^2/5}} \sim t(5)。 \tag{5.50}$$

基于正则化 $m \times 2$ 交叉验证，$T_{5\times 2CCV}$ 可泛化为：

$$T_m^{(C)} = \frac{\hat{\mu}_{m\times 2} - \mu}{\sqrt{\frac{1}{2m}\sum_{i=1}^{m} S_i^2/m}} \sim t(m)。 \tag{5.51}$$

其中，$S_i^2 = (\hat{\mu}_1^{(i)} - \hat{\mu}^{(i)})^2 + (\hat{\mu}_2^{(i)} - \hat{\mu}^{(i)})^2$。

基于统计量 $T_m^{(C)}$，μ 的 $1-\alpha$ 置信区间为：

$$\mathrm{CI}_m^{(C)} \triangleq \left(\hat{\mu}_{m\times 2} - t_{\alpha/2}(m)\sqrt{\frac{1}{2m}\sum_{i=1}^{m} S_i^2/m}, \quad \hat{\mu}_{m\times 2} + t_{\alpha/2}(m)\sqrt{\frac{1}{2m}\sum_{i=1}^{m} S_i^2/m} \right)。 \tag{5.52}$$

注记 5.6　基于统计量 $T_m^{(C)}$ 的显著性检验的一个严重的缺陷是该检验不能控制其第一类错误。具体地，$\mathrm{CI}_m^{(C)}$ 的期望长度为

$$E\left[2t_{\alpha/2}(m)\sqrt{\frac{1}{2m}\sum_{i=1}^{m} S_i^2/m} \right] = 2u_{\alpha/2}\sqrt{\frac{1}{2m}\sigma^2(1-\rho_1)}。 \tag{5.53}$$

当 $m \to \infty$ 时，该期望长度趋于零。另外，根据引理5.1，当 $\hat{\mu}_{m\times 2} \in (\Delta - \sqrt{\rho_2}\sigma u_{\alpha/2}, \Delta + \sqrt{\rho_2}\sigma u_{\alpha/2})$ 时，在 α 的置信概率意义下，原假设 H_0 不能被拒绝。基于 $T_m^{(C)}$ 的检验的第一类错误为：

$$P(\text{reject } H_0 | \mu < \Delta) = P\left(\frac{\hat{\mu}_{m\times 2} - \mu}{\sqrt{\frac{1}{2m}\sum_{i=1}^{m} S_i^2/m}} > t_\alpha(m) \mid \mu < \Delta \right), \tag{5.54}$$

其中，$m = E[m_{stop}]$。因此，当 $\hat{\mu}_{m\times 2} > \Delta + \sqrt{\frac{1}{2m}\sum_{i=1}^{m} S_i^2/m} \cdot t_\alpha(m)$ 且 $m \to \infty$ 时，$P(\text{reject } H_0|\mu < \Delta) \to 1$。也就是说，该检验不能控制第一类错误。

因该检验具有上述严重的缺陷，因此，在模拟实验中不再考虑该检验。

5.5.3 组块 3×2 交叉验证 t 检验

组块 3×2 交叉验证 t 检验[8] 采用了如下的检验统计量：

$$T_{3\times 2BCV} \triangleq \frac{\hat{\mu}_{3\times 2} - \mu}{\sqrt{\widehat{\text{Var}}[\hat{\mu}_{3\times 2}]}} \sim t(5)。 \tag{5.55}$$

其中 $\widehat{\text{Var}}[\hat{\mu}_{3\times 2}] = \frac{1}{6}\sum_{i=1}^{3}\sum_{k=1}^{2}(\hat{\mu}_k^{(i)} - \hat{\mu}_{3\times 2})^2$。

基于正则化 $m\times 2$ 交叉验证，$T_{3\times 2BCV}$ 可泛化为

$$T_m^{(B)} \triangleq \frac{\hat{\mu}_{m\times 2}}{\hat{\sigma}_m} \sim t(2m-1)。 \tag{5.56}$$

其中，$\hat{\sigma}_m = \sqrt{\widehat{\text{Var}}[\hat{\mu}_{m\times 2}]} = \sqrt{\frac{1}{2m}\sum_{i=1}^{m}\sum_{k=1}^{2}(\hat{\mu}_k^{(i)} - \hat{\mu}_{m\times 2})^2}$。

对比式（5.56）和式（5.35），可知统计量 $T_m^{(B)}$ 为随机变量 T_m 的另一个实现。统计量 $T_m^{(B)}$ 和 $T_m^{(S)}$ 的唯一区别是 C_m 的取值不同。前者采用了 $\hat{C}_m = 1$，而后者参用了 $\hat{C}_m = \sqrt{(2m+1)/(2m-1)}$。

本章推荐统计量 $T_m^{(S)}$，而不是 $T_m^{(B)}$。其主要原因在于 $\hat{C}_m = \sqrt{(2m+1)/(2m-1)} > 1$ 更容易对应一个较大的停时 m_{stop}，且它使得检验在前期更为保守。当 m 增加时，$\hat{C}_m$ 收敛于 1, 且 $T_m^{(S)}$ 逼近 $T_m^{(B)}$。

因 $T_m^{(B)}$ 和 $T_m^{(S)}$ 间仅有细微差别，在模拟实验中不再考虑该检验。

5.5.4 所有 t 检验的综合比较

表5.5综合比较了本节的所有 t 检验方法。第二行和第三行的 t 检验方法不再进一步考虑，其主要原因是：第二个 t 检验不能控制第一类错误（见第5.5.2 节）；第三个 t 检验与第四个 t 检验非常相似（见第5.5.3 节）。因此，下面主要关注 5×2 交叉验证成对 t 检验及本章所提的 t 检验。在后续的实验中，它们分别被记为“98paired”和“bmx2”。

5.6 实验设置和评价标准

在 t 检验的比较实验中，采用如下实验设置：

数据集和算法 ：实验中使用了两个数据集。第一个数据集为玩具数据集。在该玩具数据集中，正则化 $m\times 2$ 交叉验证中的所有 hold-out 估计均从多元高斯分布中生成。另一个数据集来自于 Nadeau 等的研究工作[16]。在该数据集中，UCI

表 5.5 所有 t 检验方法的综合比较（“LOC”表示置信区间长度）

标记	统计量	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{C}_m$	DOF	$\lim_{m\to\infty} E[\text{LOC}]$
(1) 正则化 $m \times 2$ 交叉验证成对 t 检验: [1]						
98paired	$T_m^{(P)}$	$\hat{\mu}_1^{(1)}$	$\frac{1}{m}\sum_{i=1}^{m} S_i^2$	-	m	$2\sqrt{1-\rho_1}\sigma u_{\sigma/2}$
(2) 正则化 $m \times 2$ 交叉验证合并 t 检验: [7]						
13comb	$T_m^{(C)}$	$\hat{\mu}_{m\times 2}$	$\frac{1}{2m}\sum_{i=1}^{m} S_i^2/m$	-	m	0
(3) 正则化 $m \times 2$ 交叉验证组块 t 检验: [8]						
14blocked	$T_m^{(B)}$	$\hat{\mu}_{m\times 2}$	$\widehat{Var}(\hat{\mu}_{m\times 2})$	1	$2m-1$	$2\sqrt{1-\rho_2}\sigma u_{\alpha/2}$
(4) 正则化 $m \times 2$ 交叉验证序贯 t 检验: this study						
bmx2	$T_m^{(S)}$	$\hat{\mu}_{m\times 2}$	$\widehat{Var}(\hat{\mu}_{m\times 2})$	$\sqrt{\frac{2m+1}{2m-1}}$	$2m-1$	$2\sqrt{1-\rho_2}\sigma u_{\alpha/2}$

Letter 数据集被用作数据总体，并从中独立且有放回地抽取 1000 个数据集。在这些数据集上，使用分类树和快速最近邻（First nearest neigbor，简记为 FNN）作为机器学习算法。这些算法的相关设置与 Nadeau 等[16] 的设置相同。

性能指标 ： 泛化误差被用作度量机器学习算法性能指标。泛化误差为损失函数的期望值。在所生成的 1000 个数据集上，$\mu_{\mathcal{A}}$，$\mu_{\mathcal{B}}$，μ 及相应的 ρ_1 and ρ_2 的模拟值也被给出。

显著性水平 ： $\alpha = 0.05$。

开始时刻 ： $m_{start} = 3$。

最大停时 ： $M = 12$ 且 $\gamma = 0.01$。

期望停时的计算方式 ： 计算 $E[m_{stop}]$ 时，记录 1000 个数据集上的 1000 个停时。这 1000 个停时的均值被用作 $E[m_{stop}]$ 的模拟值。

Δ 的设置 ： 在势函数模拟实验中，通常会固定 Δ 并变化 μ，然后计算某个检验的第一类错误和第二类错误。然而，这种实验方式的时间开销很大。在下节中，将 μ 固定并变动 μ 的值。基于这种方式绘制出的势函数线被称为“伪势函数线”。该计算方式被 Nadeau 等[16] 采用，且为传统势函数的一种不错的替代方法。在所有的模拟实验中，Δ 的值被离散化。

评价准则 ： 第一类错误和伪势函数被用作评价准则。当 $\mu \leq \Delta$ 时，第一类错误随着 Δ 的缩小而增大，当 $\mu = \Delta$ 时，第一类错误达到最大。该第一类错误的最大值在实验结果中给出。较小的第一类错误对应于一个更为保守的假设检验。另外，为公平期间，实验结果对各 t 检验方法的显著性水平进行调整，使该检验的第一类错误达到 α，并绘制相应的伪势函数。若两种 t 检验方法均拥有可接受的第一

类错误，则具有更陡势函数的 t 检验方法的性能更优。在模拟实验中，第一类错误及伪势函数的值均基于 1000 个数据上模拟得到。

5.7 正则化 $m \times 2$ 交叉验证估计的方差的三个估计的比较实验

本节的比较实验采用了第3.5.1节所给的实验数据和实验设置。表5.6 和表5.7 在模拟数据集上比较了第5.2.3 所给出的三个方差估计量。

表 5.6 实验配置"SREG+rid+n+20" 上三个方差估计的比较

m	Variance	n=100	n=200	n=300	n=400	n=500
3	$\mathrm{Var}[\hat{\mu}_{m\times 2}]$	0.192125	0.030131	0.014093	0.008821	0.006376
3	$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$	0.151037	0.015067	0.007338	0.005044	0.003932
3	$\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$	0.250211	0.027984	0.012346	0.007712	0.005587
3	$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$	0.349384	0.040901	0.017353	0.010380	0.007242
5	$\mathrm{Var}[\hat{\mu}_{m\times 2}]$	0.172204	0.027544	0.013121	0.008288	0.006042
5	$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$	0.153129	0.015071	0.007338	0.005040	0.003932
5	$\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$	0.272423	0.030586	0.013356	0.008240	0.005922
5	$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$	0.332070	0.038343	0.016366	0.009840	0.006916
7	$\mathrm{Var}[\hat{\mu}_{m\times 2}]$	0.163635	0.026455	0.012683	0.008057	0.005902
7	$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$	0.152596	0.015064	0.007336	0.005040	0.003931
7	$\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$	0.280363	0.031687	0.013779	0.008472	0.006063
7	$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$	0.322952	0.037228	0.015927	0.009616	0.006773

表 5.7 实验配置"SCLA+svm+n+20" 上的三个方差估计的比较

m	Variance	n=100	n=200	n=300	n=400	n=500
3	$\mathrm{Var}[\hat{\mu}_{m\times 2}]$	0.002662	0.001442	0.000960	0.000721	0.000558
3	$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$	0.001270	0.000646	0.000436	0.000333	0.000270
3	$\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$	0.002908	0.001384	0.000898	0.000657	0.000516
3	$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$	0.004545	0.002122	0.001359	0.000981	0.000761
5	$\mathrm{Var}[\hat{\mu}_{m\times 2}]$	0.002337	0.001293	0.000867	0.000657	0.000509
5	$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$	0.001270	0.000645	0.000437	0.000332	0.000271
5	$\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$	0.003235	0.001532	0.000990	0.000721	0.000565
5	$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$	0.004217	0.001975	0.001266	0.000916	0.000713
7	$Var[\hat{\mu}_{m\times 2}]$	0.002195	0.001231	0.000828	0.000629	0.000488
7	$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$	0.001271	0.000645	0.000437	0.000332	0.000271
7	$\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$	0.003375	0.001595	0.001029	0.000749	0.000586
7	$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$	0.004077	0.001912	0.001227	0.000888	0.000692

从表5.6 和表5.7 中，可知：

(1) 在大多数实验配置上，$\widehat{\mathrm{Var}}_1[\hat{\mu}_{m\times 2}]$ 低估了方差。$\widehat{\mathrm{Var}}_3[\hat{\mu}_{m\times 2}]$ 是方差的上偏估计。

(2) $\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$ 在大多数实验配置上为方差的上偏估计。特别是，当 m 从 3 增至 7 时，该方差估计在所有实验配置上均上偏。因此，在实际应用中，$\widehat{\mathrm{Var}}_2[\hat{\mu}_{m\times 2}]$ 为一个较为合适的选择。

5.8 模拟数据上的实验

下面，首先给出玩具数据集上的实验结果，然后给出 UCI Letter 数据集上的实验结果。

5.8.1 玩具数据集上的实验

在玩具数据集中，正则化 $m \times 2$ 交叉验证中的所有 hold-out 估计值，即 $\hat{\boldsymbol{\mu}}_m$，来自于一多元正态分布中，其均值和协方差在式（5.4）中给出。将 μ 和 σ^2 分别设置为零和一。从该分布中生成 $\hat{\boldsymbol{\mu}}_m$ 的 1000 条样例。其中，ρ_1 和 ρ_2 取范围 $0 \leq \rho_1, \rho_2 \leq 1/2$ 内的若干不同水平。阈值 Δ 取从 -3 到 3 内的离散值。在 $\hat{\boldsymbol{\mu}}_m$ 的 1000 条样例，分别对比上节中的两种 t 检验方法。此外，将 ρ_1 和 ρ_2 的真值带入到 C_m 和 f_m 的表达式中，可得到一个基准 t 检验。该检验对应的实验结果被标记为“ideal”。

玩具数据集上的第一类错误的最大值在表5.8 中给出。从该表可知：一、基准 t 检验的第一类错误与名义上的显著性水平 0.05 相一致；二、标记为“98paired”的正则化 $m \times 2$ 交叉验证成对序贯 t 检验中，第一类错误随着 ρ_1 的增加而增加。当 $\rho_1 = 0.0$ 时，第一类错误在 $\alpha = 0.05$ 左右，仍可接受。然而，当 $\rho_1 > 0.0$ 时，第一类错误超过 α，相应的检验也容易产生激进的结论。三、正则化 $m \times 2$ 交叉验证序贯 t 检验（标记为“bmx2”）的第一类错误均小于 $\alpha = 0.05$。这表明该检验为保守的。随着 ρ_2 增加，正则化 $m \times 2$ 交叉验证序贯 t 检验的第一类错误也相应增加。当 $\rho_2 = 0.05$ 时，第一类错误达到 $\alpha = 0.05$。

图5.3 给出了三种 t 检验的伪势函数图像。在每一幅子图中，黑色的垂直线表示 μ 的真值。蓝色的水平线表示显著性水平 $\alpha = 0.05$。每一行多幅子图对应相同的 ρ_1，每一列多幅子图对应相同的 ρ_2。每一幅子图中，比较了三种不同的 t 检验。其中，因基准 t 检验具有名义上的第一类错误，无须对其校正。对于其余两种 t 检验，分别将其在 $\Delta = \mu$ 处的第一类错误校正为 $\alpha = 0.05$。该图表明，正则化 $m \times 2$ 交叉验证成对序贯 t 检验的势函数最为平缓。当 ρ_2 较小时，正则化 $m \times 2$ 交叉验证序贯 t 检验的势函数比基准 t 检验的势函数更陡，这表明正则化 $m \times 2$ 交叉验证序

表 5.8 玩具数据集上三种 t 检验的第一类错误的最大值

t 检验方法	ρ_1	ρ_2					
		0.0	0.1	0.2	0.3	0.4	0.5
98paired	0.0	0.047	0.051	0.037	0.039	0.052	0.041
	0.1	0.059	0.048	0.058	0.043	0.060	0.057
	0.2	0.055	0.056	0.066	0.057	0.068	0.048
	0.3	0.079	0.079	0.064	0.087	0.074	0.078
	0.4	0.091	0.087	0.092	0.089	0.080	0.095
	0.5	0.115	0.120	0.109	0.104	0.099	0.112
bmx2	0.0	0.000	0.000	0.000	0.012	0.023	0.048
	0.1	0.000	0.000	0.001	0.001	0.022	0.047
	0.2	0.000	0.000	0.001	0.006	0.026	0.045
	0.3	0.000	0.000	0.002	0.013	0.019	0.054
	0.4	0.000	0.000	0.003	0.004	0.020	0.051
	0.5	0.000	0.000	0.005	0.012	0.021	0.054
ideal	0.0	0.050	0.051	0.043	0.056	0.047	0.048
	0.1	0.051	0.043	0.047	0.034	0.054	0.048
	0.2	0.050	0.050	0.061	0.055	0.060	0.047
	0.3	0.050	0.054	0.043	0.046	0.044	0.057
	0.4	0.047	0.048	0.053	0.043	0.050	0.055
	0.5	0.045	0.040	0.048	0.042	0.037	0.054

贯 t 检验较为保守。当 ρ_2 增加时，正则化 $m \times 2$ 交叉验证序贯 t 检验的势函数收敛于基准 t 检验的势函数。

图5.4 给出了三种 t 检验在玩具数据集上的停时。当 $\Delta > \mu$ 时，三种 t 检验均需要最大停时以拒绝 H_0。当 $\Delta \leq \mu$ 时，正则化 $m \times 2$ 交叉验证序贯 t 检验与正则化 $m \times 2$ 交叉验证成对序贯 t 检验具有明显的不同。具体来说，当 Δ 远离 μ 时，正则化 $m \times 2$ 交叉验证序贯 t 检验的期望停时比正则化 $m \times 2$ 交叉验证成对序贯 t 检验的期望停时更小。相比，当 Δ 靠近 μ，正则化 $m \times 2$ 交叉验证序贯 t 检验需要更大的期望停时来更好地区分两个算法性能指标之差的显著性。期望停时上的这些区别表明正则化 $m \times 2$ 交叉验证序贯 t 检验的期望停时的分配更为合理。

基于上述分析可知，正则化 $m \times 2$ 交叉验证序贯 t 检验具有更小的一类错误、更陡的势函数，且具有更合理的期望停时。因此，正则化 $m \times 2$ 交叉验证序贯 t 检验比正则化 $m \times 2$ 交叉验证成对序贯 t 检验更优。

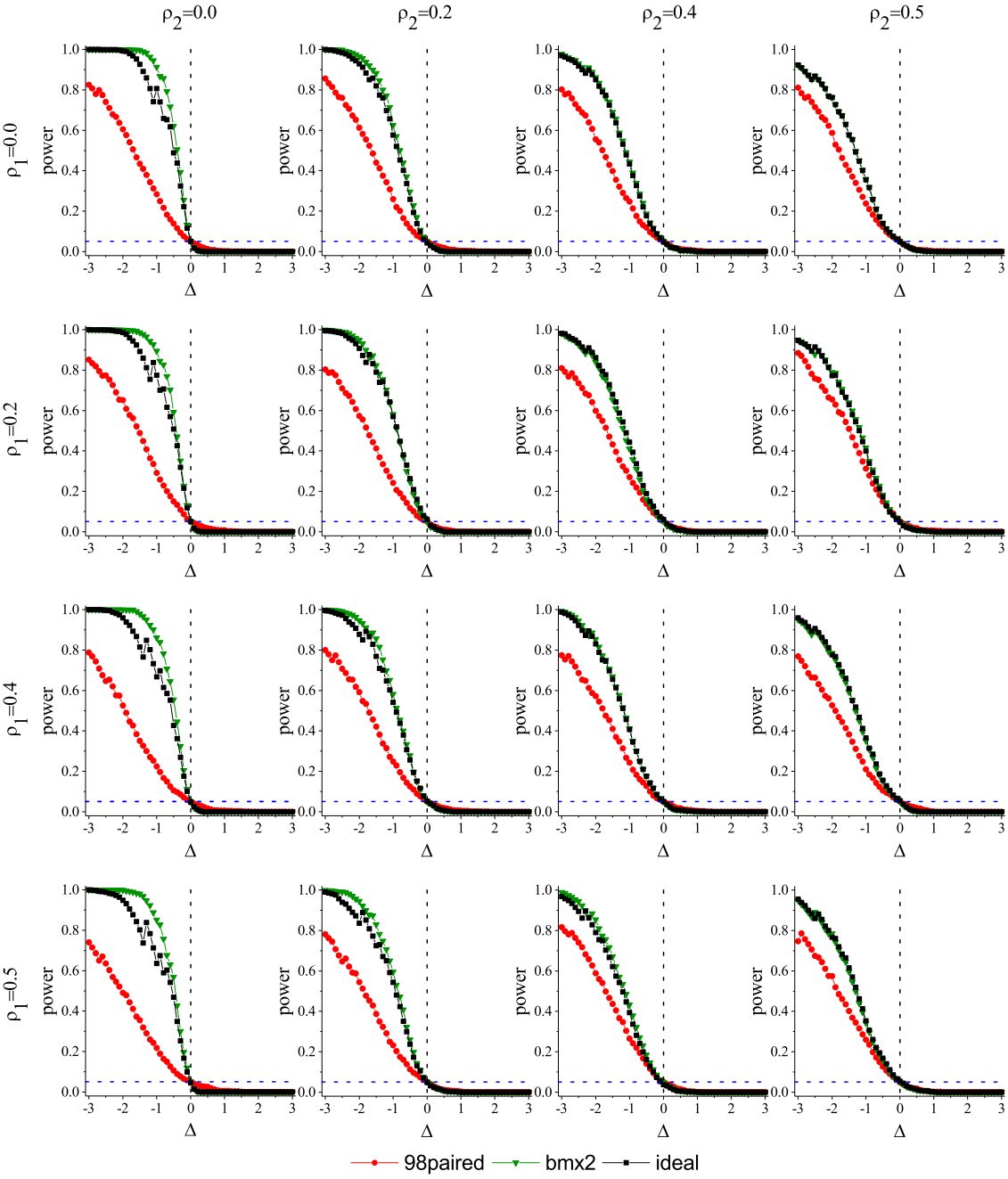

图 5.3　玩具数据集上三种 t 检验的伪势函数图

5.8.2　UCI Letter 数据集上的实验

UCI Letter 数据集上的分类任务旨在识别 26 个手写字母。该数据集含 20000 条样例、16 个特征和 26 个类别。实验中，Letter 数据集被用作数据总体。从该总体中，1000 个数据集被有放回地独立抽出。每一个数据集的大小为 300。在 1000 个数据集上，对比如下两个分类器：

- 分类树（算法 $\mathcal{A}$）：R 软件中的“tree”包被用作训练模型。

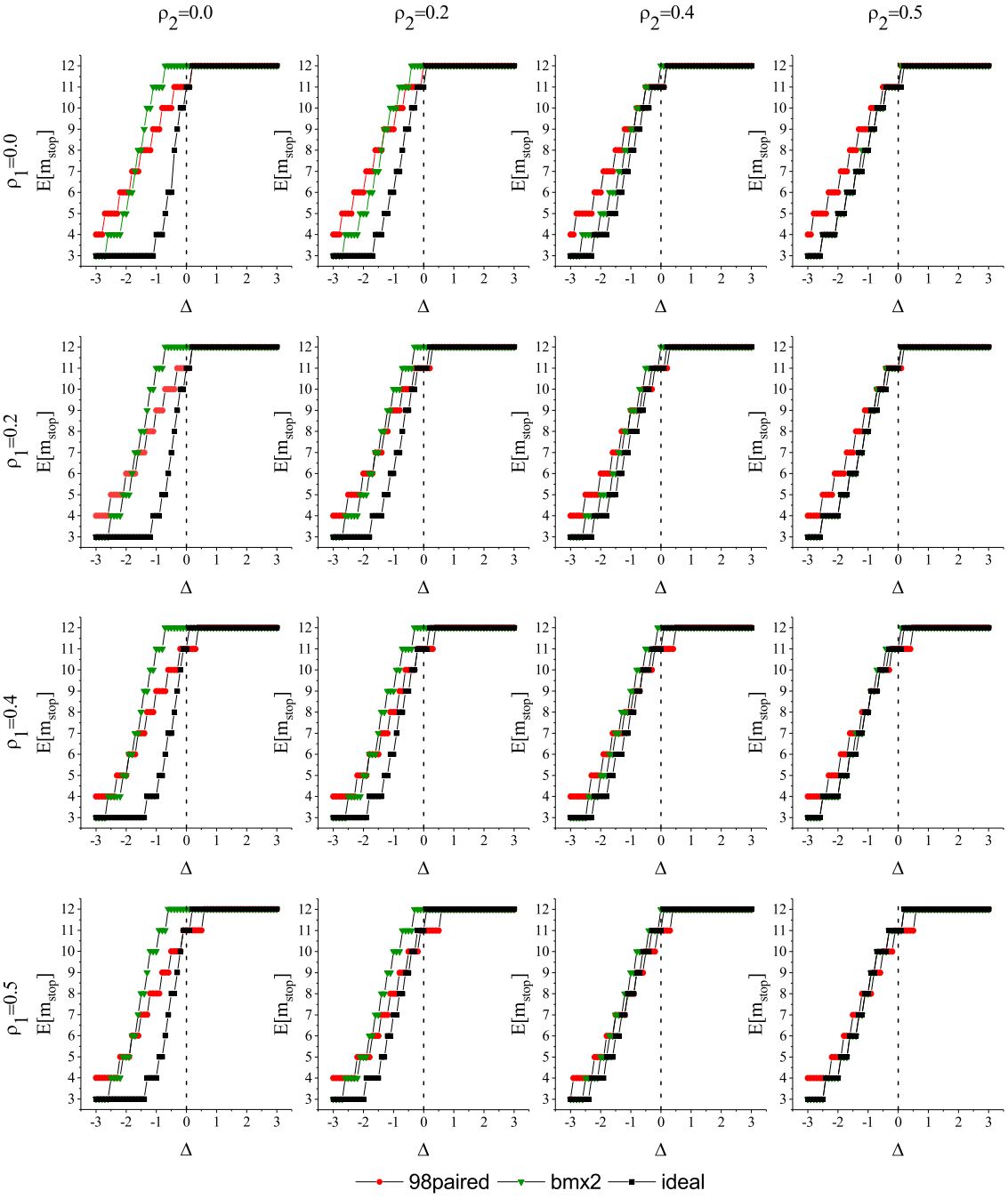

图 5.4 玩具数据集上三种 t 检验的期望停时

- 第一最近邻（算法 $\mathcal{B}$）：第一最近邻规则基于一个加权距离进行分类。具体地，两个向量的加权距离为：

$$d(X^{(1)}, X^{(2)}) = \sum_{k=1}^{3} \omega^{2-k} \sum_{i \in C_k} (X^{(1)} - X^{(2)})^2。 \tag{5.57}$$

其中，$C_1 = \{1, 3, 9, 16\}$，$C_2 = \{2, 4, 6, 7, 8, 10, 12, 14, 15\}$ 和 $C_3 = \{5, 11, 13\}$ 分别为三种特征指标集合，且它们的权重分别为 ω、1 和 ω^{-1}。

在实验中，0-1 损失被用来度量两个算法的泛化误差。参数 ω 采用了六种设置，相应的六组实验分别标记为“Letter1”到“Letter6”。实验的上述设置与 Nadeau 等[16] 的设置相同。表5.9 给出了六组实验的基本信息，并给出了 μ、ρ_1 和 ρ_2 等的真值。

表 5.9 UCI Letter 数据集上六组实验的基本信息

	实验 1	实验 2	实验 3	实验 4	实验 5	实验 6
标记	Letter1	Letter2	Letter3	Letter4	Letter5	Letter6
n	300	300	300	300	300	300
w	1	5	10	17.25	25	2048
	算法 $\mathcal{A}$：第一最近邻					
$\mu_{\mathcal{A}}$	0.5366	0.5895	0.6283	0.6627	0.6863	0.7742
$\rho_{\mathcal{A},1}$	0.4857	0.5159	0.5378	0.5483	0.5574	0.5738
$\rho_{\mathcal{A},2}$	0.3488	0.3664	0.3697	0.3754	0.3781	0.3816
	算法 $\mathcal{B}$：分类树					
$\mu_{\mathcal{A}}$	0.6928	0.6928	0.6928	0.6928	0.6928	0.6928
$\rho_{\mathcal{B},1}$	0.2923	0.2923	0.2923	0.2923	0.2923	0.2923
$\rho_{\mathcal{B},2}$	0.2324	0.2324	0.2324	0.2324	0.2324	0.2324
	算法 $\mathcal{A}$ 和 $\mathcal{B}$ 性能指标之差					
μ	0.1561	0.1033	0.0644	0.0301	0.0065	-0.0814
ρ_1	0.1542	0.1885	0.2048	0.2175	0.2274	0.2698
ρ_2	0.1583	0.1729	0.1778	0.1881	0.1921	0.2239

在 Letter 数据集上，比较了正则化 $m \times 2$ 交叉验证序贯 t 检验（标记为“bmx2”）和正则化 $m \times 2$ 交叉验证成对序贯 t 检验（标记为“98paired”），基准 t 检验被略去。表5.10中给出了两种 t 检验的第一类错误的最大值。显然，正则化 $m \times 2$ 交叉验证成对序贯 t 检验的第一类错误均超过显著性水平 $\alpha = 0.05$。特别是在第六组实验上，第一类错误的最大值达到了 0.075。相比而言，正则化 $m \times 2$ 交叉验证序贯 t 检验的第一类错误均小于 0.05。也就是说，正则化 $m \times 2$ 交叉验证成对序贯 t 检验容易产生激进的推断并导致假阳性结论。不过，正则化 $m \times 2$ 交叉验证序贯 t 检验相对保守。

图5.5 和图5.6 分别给出了伪势函数和期望停时上的比较。正则化 $m \times 2$ 交叉验证序贯 t 检验的势函数比正则化 $m \times 2$ 交叉验证成对序贯 t 检验第势函数更陡。这表明正则化 $m \times 2$ 交叉验证序贯 t 检验在鉴别算法性能指标之差时更有优势。另

表 5.10 Letter 数据集上两种 t 检验的第一类错误的最大值

	Letter1	Letter2	Letter3	Letter4	Letter5	Letter6
98paired	0.062	0.059	0.068	0.062	0.058	0.075
bmx2	0.002	0.001	0.001	0.002	0.003	0.006

外，图5.6 表明正则化 $m \times 2$ 交叉验证序贯 t 检验的期望停时更为合理。总之，在 Letter 数据集上，对于算法比较任务，正则化 $m \times 2$ 交叉验证序贯 t 检验比正则化 $m \times 2$ 交叉验证成对序贯 t 检验更为优良。

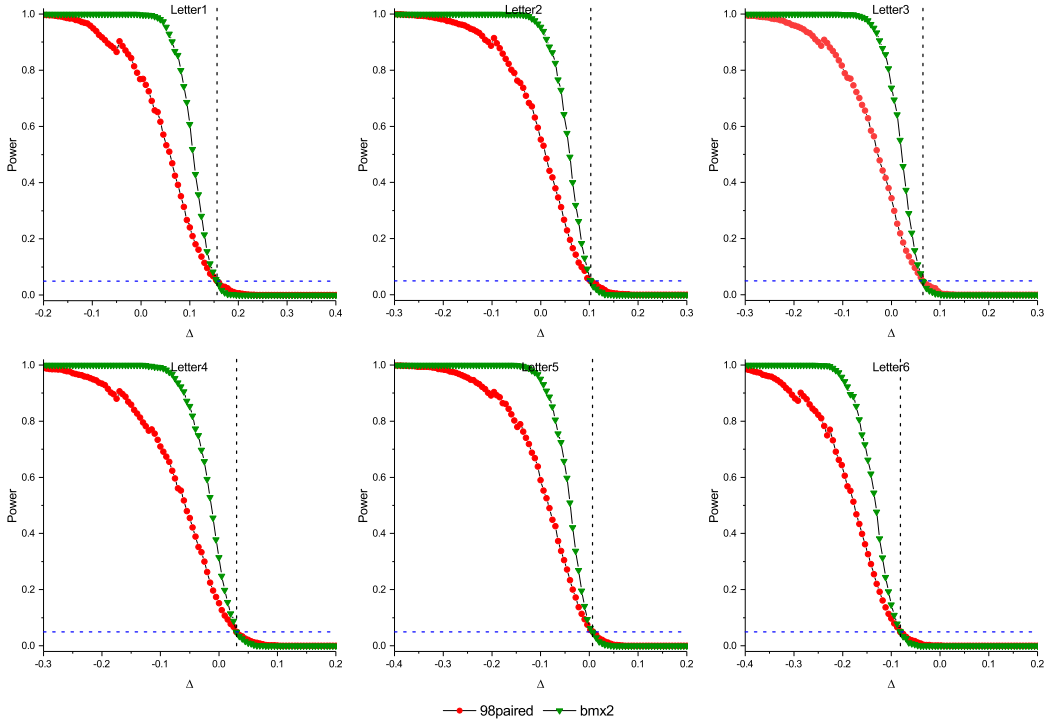

图 5.5 Letter 数据集上两个 t 检验的伪势函数

5.9 附录

5.9.1 引理5.1 的证明

假设常用的性能指标（例如，0-1 损失）的分布不依赖于训练集和测试样例的特定实现。另外，$E[\hat{\boldsymbol{\mu}}_m]$ and $Cov[\hat{\boldsymbol{\mu}}_m]$ 均是关于 D_n 和正则化 $m \times 2$ 交叉验证的正则化切分集，且数据集 D_n 中的所有样例均独立同分布。基于这些前提条件，易得式（5.4）中 $E[\hat{\boldsymbol{\mu}}_m]$ 和 $Cov[\hat{\boldsymbol{\mu}}_m]$ 的具体形式。

假定 $\hat{\boldsymbol{\mu}}$ 服从多元高斯分布，即 $\hat{\boldsymbol{\mu}} \sim \mathcal{N}(\mu \mathbf{1}_{2m}, \sigma^2 \boldsymbol{\Sigma}_{2m})$，则所有的 hold-out 估计

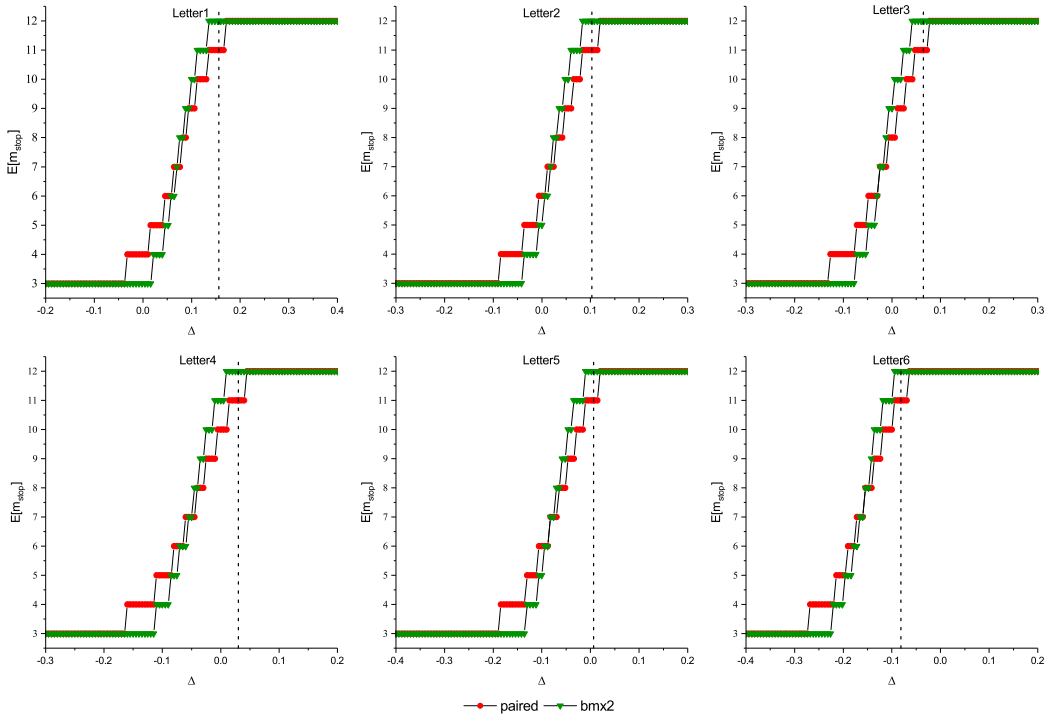

图 5.6　Letter 数据集上两个 t 检验的期望停时比较

$\hat{\mu}_k^{(j)}$ 均服从正态分布，其中，$k = 1, 2$ 且 $j = 1, \ldots, m$。则有:

$$\hat{\mu}_{m\times 2} - \mu = \frac{1}{2m}\sum_{j=1}^{m}\sum_{k=1}^{2}\hat{\mu}_k^{(j)} - \mu = \frac{1}{2m}\mathbf{1}^\top(\hat{\boldsymbol{\mu}}_m - \mu\mathbf{1})。 \tag{5.58}$$

由此，可得：

$$E[\hat{\mu}_{m\times 2} - \mu] = \frac{1}{2m}\mathbf{1}^\top E[\hat{\boldsymbol{\mu}}_m - \mu\mathbf{1}] = 0, \tag{5.59}$$

$$Var[\hat{\mu}_{m\times 2} - \mu] = \frac{\sigma^2}{4m^2}\mathbf{1}^\top\boldsymbol{\Sigma}_{2m}\mathbf{1} = \frac{\sigma^2}{2m}(1 + \rho_1 + 2(m-1)\rho_2)。 \tag{5.60}$$

因此有：

$$\frac{\hat{\mu}_{m\times 2} - \mu}{\sqrt{\sigma^2(1 + \rho_1 + 2(m-1)\rho_2)/2m}} \sim \mathcal{N}(0, 1)。 \tag{5.61}$$

5.9.2　引理5.2 的证明

将 $\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]$ 写作矩阵形式，则有：

$$\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}] = \frac{1}{2m}\sum_{j=1}^{m}\sum_{k=1}^{2}(\hat{\mu}_k^{(j)} - \hat{\mu}_{m\times 2})^2 = \frac{1}{2m}(\hat{\boldsymbol{\mu}}_m - \mu\mathbf{1})^\top\mathbf{B}(\hat{\boldsymbol{\mu}}_m - \mu\mathbf{1})。 \tag{5.62}$$

其中，$\mathbf{B} = \mathbf{I} - \mathbf{1}\mathbf{1}^\top/2m$。

用 σ^2 除以 $\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]$，则有：

$$\frac{1}{\sigma^2}\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}] = \frac{1}{2m}\left(\sigma^{-1}\boldsymbol{\Sigma}_{2m}^{-1/2}(\hat{\boldsymbol{\mu}}_m - \mu\mathbf{1})\right)^\top\boldsymbol{\Sigma}_{2m}^{1/2}\mathbf{B}\boldsymbol{\Sigma}_{2m}^{1/2}\left(\sigma^{-1}\boldsymbol{\Sigma}_{2m}^{-1/2}(\hat{\boldsymbol{\mu}}_m - \mu\mathbf{1})\right) \tag{5.63}$$

其中，$\sigma^{-1}\mathbf{\Sigma}_{2m}^{-1/2}(\hat{\boldsymbol{\mu}}_m - \mu\mathbf{1}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$。

记 λ_j 为矩阵 $\mathbf{\Sigma}_{2m}^{1/2}\mathbf{B}\mathbf{\Sigma}_{2m}^{1/2}$ 的第 j 个特征值。则有：

$$\frac{2m}{v_m\sigma^2}\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}] \sim \chi^2(f_m)。 \tag{5.64}$$

其中 $v_m = \sum_{j=1}^m \lambda_j^2 / \sum_{j=1}^m \lambda_j$ 且 $f_m = (\sum_{j=1}^m \lambda_j)^2 / \sum_{j=1}^m \lambda_j$。

因矩阵的所有特征值之和为矩阵的迹，则有：

(1) $\sum_{j=1}^m \lambda_j = tr(\mathbf{\Sigma}_{2m}^{1/2}\mathbf{B}\mathbf{\Sigma}_{2m}^{1/2}) = 2m - (1 + \rho_1 + 2(m-1)\rho_2)$。

(2) $\sum_{j=1}^m \lambda_j^2 = tr(\mathbf{\Sigma}_{2m}^{1/2}\mathbf{B}\mathbf{\Sigma}_{2m}^{1/2}\mathbf{\Sigma}_{2m}^{1/2}\mathbf{B}\mathbf{\Sigma}_{2m}^{1/2}) = m(1-\rho_1)^2 + (m-1)(1+\rho_1-2\rho_2)^2$。

由此，可得：

$$v_m = \frac{m(1-\rho_1)^2 + (m-1)(1+\rho_1-2\rho_2)^2}{2m - (1+\rho_1+2(m-1)\rho_2)}, \tag{5.65}$$

$$f_m = \frac{[2m(1-\rho_2) - (1+\rho_1-2\rho_2)]^2}{m(1-\rho_1)^2 + (m-1)(1+\rho_1-2\rho_2)^2}。 \tag{5.66}$$

另外，假设随机变量 $x \sim \chi^2(f_m)$，则 $E[x] = f_m$。因此

$$E\left[\frac{2m}{v_m\sigma^2}\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]\right] = f_m。 \tag{5.67}$$

上式可重写为：

$$E\left[\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]\right] = \frac{1}{2m}v_m f_m \sigma^2。 \tag{5.68}$$

5.9.3 定理5.2 的证明

给出证明之前，引入如下两条引理。

引理 5.5 对于随机向量 $\mathbf{x} \in \mathbb{R}^n$，假定 $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma}_n)$，其中，$\mathbf{\Sigma}_n \geq 0$。则对于任意的 $\mathbf{B} \in \mathbb{R}^{m\times n}$，有 $\mathbf{Bx} \sim \mathcal{N}(\mathbf{B}\boldsymbol{\mu}, \mathbf{B}\mathbf{\Sigma}_n\mathbf{B}^\top)$。

引理 5.6 对于随机向量 $\mathbf{x} \in \mathbb{R}^n$，假定 $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma}_n)$，其中，$\mathbf{\Sigma}_n > 0$。则，对于两个实矩阵 $\mathbf{A} \in \mathbb{R}^{n\times n}$ 和 $\mathbf{B} \in \mathbb{R}^{n\times n}$ 且满足 $\mathbf{A\Sigma B} = 0$，则 $\mathbf{x}^\top\mathbf{Ax}$ 独立于 $\mathbf{x}^\top\mathbf{Bx}$。

下面给出定理5.2的证明。引理5.1 和引理5.2 表明 T_m 的分子 $\hat{\mu}_{m\times 2} - \mu$ 和分母 $\mathrm{Var}[\hat{\mu}_{3\times 2}]$ 分别服从正态分布和卡方分布。下面证明 $\hat{\mu}_{m\times 2} - \mu$ 和 $\mathrm{Var}[\hat{\mu}_{3\times 2}]$ 间的独立性。

基于式（5.58）和式（5.62）中 $\hat{\mu}_{m\times 2} - \mu$ 和 $\mathrm{Var}[\hat{\mu}_{3\times 2}]$ 的矩阵形式，根据引理5.6，有：

$$\mathbf{A}\mathbf{\Sigma}_{2m}\mathbf{B} = \frac{1}{2m}\mathbf{1}^\top\Sigma_{2m}(\mathbf{I} - \frac{1}{2m}\mathbf{1}\mathbf{1}^\top) = 0, \tag{5.69}$$

其中，$A = \mathbf{1}^\top/2m$ 且 $\mathbf{B} = \mathbf{I} - \mathbf{1}\mathbf{1}^\top/2m$。则有 $\hat{\mu}_{m\times 2} - \mu$ 独立于 $\mathrm{Var}[\hat{\mu}_{3\times 2}]$。因此，$T_m$ 服从 t 分布，即：

$$\frac{\hat{\mu}_{m\times 2} - \mu}{\sqrt{\sigma^2(1+\rho_1+2(m-1)\rho_2)/2m} \cdot \sqrt{\frac{2m}{v_m f_m \sigma^2}\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]}} \sim t(f_m)。 \tag{5.70}$$

其中，v_m 和 f_m 分别在式（5.65）和式（5.66）中给出。

简化上式，可得：

$$T_m \triangleq \frac{\hat{\mu}_{m\times 2}}{C_m \hat{\sigma}_m} \sim t(f_m), \tag{5.71}$$

其中 $\hat{\sigma}_m = \sqrt{\widehat{Var}[\hat{\mu}_{m\times 2}]}$ 且 $C_m = \sqrt{\frac{1+\rho_1+2(m-1)\rho_2}{2m-(1+\rho_1+2(m-1)\rho_2)}}$

最后，证明 $\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]$ 的期望。记 $\hat{\sigma}_m^2 = \widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]$ 且

$$E[\hat{\sigma}_m^2] = \frac{1}{2m}(1+\rho_1)\sigma^2 + \frac{m-1}{m}\sigma^2\left(1 - \frac{1}{m-1}\rho_1 - \rho_2\right)。 \tag{5.72}$$

设 $y = \frac{2m}{C_m\sigma^2}\hat{\sigma}_m^2 \sim \chi^2(f_m) = \frac{1}{2^{f_m/2}\Gamma(f_m/2)} y^{\frac{f_m}{2}-1}e^{-\frac{y}{2}}$，可得 $\hat{\sigma}_m$ 的累积分布函数如下：

$$\begin{aligned} P(\hat{\sigma}_m \le z) &= P\left(\frac{2m}{C_m\sigma^2}\hat{\sigma}_m^2 \le \frac{2m}{C_m\sigma^2}z^2\right) = P\left(Y \le \frac{2m}{C_m\sigma^2}z^2\right) \\ &= \int_0^{\frac{2m}{C_m\sigma^2}z^2} \frac{1}{2^{f_m/2}\Gamma(f_m/2)} y^{\frac{f_m}{2}-1}e^{-\frac{y}{2}}dy。 \end{aligned} \tag{5.73}$$

因此，$\hat{\sigma}_m$ 的概率密度函数为：

$$\begin{aligned} f_{\hat{\sigma}_m}(z) &= p(\hat{\sigma}_m = z) = \frac{1}{2^{f_m/2}\Gamma(f_m/2)}\left(\frac{2m}{C_m\sigma^2}z^2\right)^{\frac{f_m}{2}-1} e^{-\frac{\frac{2m}{C_m\sigma^2}z^2}{2}} \frac{2m}{C_m\sigma^2}2z \\ &= \frac{1}{2^{\frac{f_m}{2}-1}\Gamma(f_m/2)}\left(\frac{2m}{C_m\sigma^2}\right)^{\frac{f_m}{2}} z^{f_m-1}e^{-\frac{mz^2}{C_m\sigma^2}} \\ &= \frac{2}{\Gamma(f_m/2)}\left(\frac{m}{C_m\sigma^2}\right)^{\frac{1}{2}}\left(\frac{mz^2}{C_m\sigma^2}\right)^{\frac{f_m-1}{2}} e^{-\frac{mz^2}{C_m\sigma^2}}。 \end{aligned} \tag{5.74}$$

$\hat{\sigma}$ 的期望为：

$$\begin{aligned} E[\hat{\sigma}_m] &= \int_0^{+\infty} \frac{2}{\Gamma(f_m/2)}\left(\frac{m}{C_m\sigma^2}\right)^{\frac{1}{2}} z\left(\frac{mz^2}{C_m\sigma^2}\right)^{\frac{f_m-1}{2}} e^{-\frac{mz^2}{C_m\sigma^2}}dz \\ &= \int_0^{+\infty} \frac{2}{\Gamma(f_m/2)}\left(\frac{mz^2}{C_m\sigma^2}\right)^{\frac{f_m}{2}} e^{-\frac{mz^2}{C_m\sigma^2}}dz \\ &= \int_0^{+\infty} \frac{2}{\Gamma(f_m/2)}\left(\frac{mz^2}{C_m\sigma^2}\right)^{\frac{f_m+2}{2}-1} e^{-\frac{mz^2}{C_m\sigma^2}}dz \\ &= \int_0^{+\infty} \frac{2}{\Gamma(f_m/2)} y^{\frac{f_m+2}{2}-1}e^{-y}\frac{C_m\sigma^2}{2m}\sqrt{\frac{m}{C_m\sigma^2}}y^{-\frac{1}{2}}dy \\ &= \int_0^{+\infty} \frac{\sqrt{C_m\sigma^2/m}}{\Gamma(f_m/2)} y^{\frac{f_m+1}{2}-1}e^{-y}dy \end{aligned}$$

$$= \sqrt{\frac{C_m\sigma^2}{m}}\frac{\Gamma(\frac{f_m+1}{2})}{\Gamma(\frac{f_m}{2})}。 \tag{5.75}$$

5.9.4 样本均值中 $\rho_{\mathcal{A},1}$ 和 $\rho_{\mathcal{A},2}$ 的理论分析

记数据集为 $D_n = \{(x_i, y_i)\}_{i=1}^n$，其中，$n$ 条样例均独立同分布。设 $\mathcal{S}_j = (S^{(j)}, T^{(j)})$ 为 D_n 上的一个切分。其中，$S^{(j)}$ 和 $T^{(j)}$ 分别对应训练集和验证集。假定 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 为 D_n 上的两个随机切分，且满足 $|S^{(1)} \cap S^{(2)}| = x$，其中，$x$ 为 $S^{(1)}$ 和 $S^{(2)}$ 间的重叠样本个数。基于 $\mathcal{S}_1$ 和 $\mathcal{S}_2$，D_n 可以被切分为如下四部分：

- $A = \{a|a \in S^{(1)} \cap S^{(2)}\}$ 为 $S^{(1)}$ 和 $S^{(2)}$ 间的共同样本；
- $B = \{b|b \in S^{(1)} \backslash A\}$ 为出现在 $S^{(1)}$ 但不在 $S^{(2)}$ 中的样本；
- $C = \{c|c \in S^{(2)} \backslash A\}$ 为出现在 $S^{(2)}$ 但不在 $S^{(1)}$ 中的样本；
- $D = \{d|d \in T^{(1)} \cap T^{(2)}\}$ 为 D_n 中均不出现在 $S^{(1)}$ 和 $S^{(2)}$ 中的样本；

基于上述四部分，可得 $|A| = |D| = x$，$|B| = |C| = n/2 - x$，$D^{(1)} = A \cup B$ 且 $D^{(2)} = A \cup C$。

样本均值回归中的决策函数为训练集中所有响应值 y_i 的样本均值。例如，基于训练集 $A \cup B$ 的样本均值回归在测试集 y_d 上的预测值为：$\hat{y}_{A\cup B,d} = \bar{y}_{A\cup B} = 2\sum_{y\in A\cup B} y/n$。

样本均值回归的性能使用泛化误差来度量，泛化误差中采用平方损失。记 $\hat{\mu}_{HO}(\mathcal{S}_1)$ 和 $\hat{\mu}_{HO}(\mathcal{S}_2)$ 分别为切分 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 对应的两个 hold-out 估计。第三 章给出了协方差函数 $\hat{\mu}_{HO}(\mathcal{S}_1)$ and $\hat{\mu}_{HO}(\mathcal{S}_2)$ 的一个分解式：

$$f(x) = Cov[\hat{\mu}_{HO}(\mathcal{S}_1), \hat{\mu}_{HO}(\mathcal{S}_2)|x] = \frac{4}{n^2}\left[(\omega + \gamma - 2\tau)x^2 + (\sigma^2 - \omega - n\gamma + n\tau)x + \frac{n^2}{4}\gamma\right]。 \tag{5.76}$$

其中参数 σ^2、ω、γ 和 τ 的定义如下：

$$Cov[e_i(\mathcal{S}_1), e_j(\mathcal{S}_2)|x] = \begin{cases} \sigma^2 & i = j,\ y_i, y_j \in D \\ \omega & i \neq j,\ y_i, y_j \in D \\ \gamma & y_i \in C \text{ 及 } y_j \in B \\ \tau & \text{其它} \end{cases} \tag{5.77}$$

且 $e_i(\mathcal{S}_1) = (\hat{y}_{A\cup B,i} - y_i)^2$，$e_j(\mathcal{S}_2) = (\hat{y}_{A\cup C,j} - y_j)^2$。

假定 $\forall i = 1, \ldots, n$，有 $y_i \sim \mathcal{N}(\mu, \psi^2)$。则参数 σ^2、γ、ω 和 τ 的表达式如下：

- $\sigma^2 = Cov[(\bar{y}_{A\cup B} - y_d)^2, (\bar{y}_{A\cup C} - y_d)^2] = (\frac{32x^2}{n^4} + \frac{16x}{n^2} + 2)\psi^4$；
- $\omega = Cov[(\bar{y}_{A\cup B} - y_d)^2, (\bar{y}_{A\cup C} - y_{d'})^2] = \frac{32}{n^4}x^2\psi^4$；
- $\gamma = Cov[(\bar{y}_{A\cup B} - y_c)^2, (\bar{y}_{A\cup C} - y_b)^2] = \frac{32(n-x)^2}{n^4}\psi^4$；
- $\tau = Cov[(\bar{y}_{A\cup B} - y_c)^2, (\bar{y}_{A\cup C} - y_d)^2] = \frac{8(n-2x)^2}{n^4}\psi^4$；

将参数 σ^2、ω、γ 和 τ 的上述表达式代入式（5.76）中，可得：

$$f(x) = \frac{8\psi^4}{n^4}\left[36x^2 + n(n-20)x + 4n^2\right]。\tag{5.78}$$

进而，可得：

$$f(0) = \frac{32}{n^2}\psi^4,\quad f(\frac{n}{4}) = \frac{2(n+5)}{n^2}\psi^4,\quad f(\frac{n}{2}) = \frac{4(n+6)}{n^2}\psi^4。\tag{5.79}$$

相应的 ρ_1 和 ρ_2 为：

$$\rho_{\mathcal{A},1} = \frac{f(0)}{f(\frac{n}{2})} = \frac{\frac{32}{n^2}\psi^4}{\frac{4(n+6)}{n^2}\psi^4} = \frac{8}{n+6},\tag{5.80}$$

$$\rho_{\mathcal{A},2} = \frac{f(\frac{n}{4})}{f(\frac{n}{2})} = \frac{\frac{2(n+5)}{n^2}\psi^4}{\frac{4(n+6)}{n^2}\psi^4} = \frac{n+5}{2n+12}。\tag{5.81}$$

5.9.5　定理5.3 的证明

该定理的证明需应用引理3.1 和引理3.2。这两个引理中分别给出了协方差函数 $f(x)$ 和 $g(x)$ 的理论性质。图5.7 给出了 $f(x)$ 和 $g(x)$ 的图像。其中，图5.7(a) 给出了函数 $g(x)$。该函数为一个对称下凸函数。图5.7(b) 和 (c) 给出了两种不同类型的函数 $f(x)$。在图5.7(b) 中，$f(x)$ 为单调下凸函数。图5.7(c) 中，$f(x)$ 为下凸函数，该函数开始时递减然后递增。在不同的数据总体、算法类型及性能度量下，这两种类型的 $f(x)$ 都可能出现。

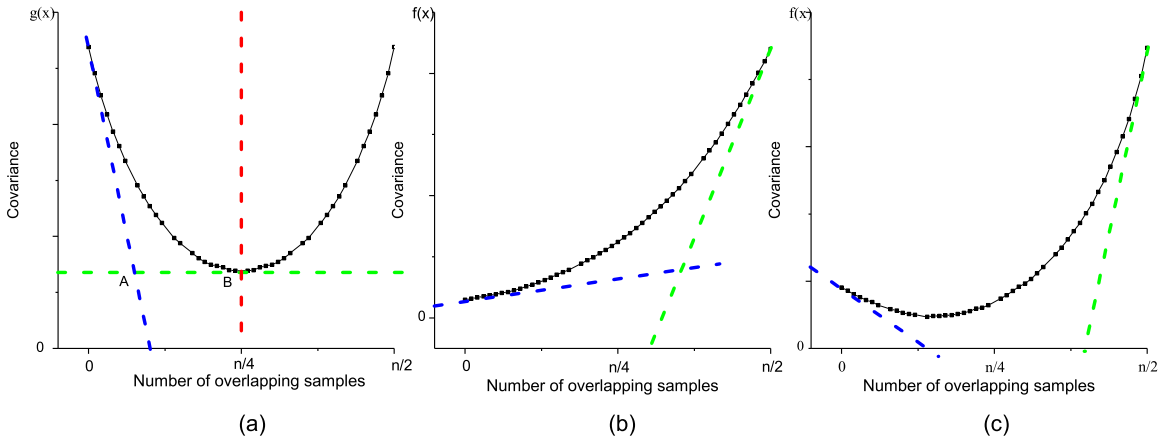

图 5.7　函数 $f(x)$ 和 $g(x)$ 的图像

根据引理3.2中给出的 $g(x)$ 的下凸对称性质，可得：

$$g(\frac{n}{4}) = f(\frac{n}{4}) < g(\frac{n}{2}) = \frac{1}{2}\left(f(0) + f(\frac{n}{2})\right)。\tag{5.82}$$

因为 $f(n/2) > 0$，将上述不等式左右两边同时除以 $f(n/2)$，可得：

$$\frac{f(\frac{n}{4})}{f(\frac{n}{2})} < \frac{1}{2}\left(\frac{f(0)}{f(\frac{n}{2})} + 1\right)。\tag{5.83}$$

因为 $\rho_{\mathcal{A},1} = f(0)/f(n/2)$ 且 $\rho_{\mathcal{A},2} = f(n/4)/f(n/2)$，则有：

$$\rho_{\mathcal{A},2} < \frac{1+\rho_{\mathcal{A},1}}{2}。 \tag{5.84}$$

因此，式（5.28）中 ρ_2 的上界得证。

图5.7(a) 中，红色的折线表示 $g(x)$ 的对称轴，蓝色的折线表示 $g(x)$ 在 $x=0$ 处的切线，绿色的折线表示水平线 $y = g(n/4)$。记 $\mathrm{A} = (x_\mathrm{A}, y_\mathrm{A})$ 为蓝色折线和绿色折线的交叉点。记 $\mathrm{B} = (x_\mathrm{B}, y_\mathrm{B})$ 为红色折线和绿色折线的交叉点。根据 $g(x)$ 的对称下凸性质，易得 $x_\mathrm{A} < x_\mathrm{B}$ 且 $y_\mathrm{A} = y_\mathrm{B}$。根据 $x_\mathrm{A} < x_\mathrm{B}$，易证式（5.28）中 ρ_2 的上界。具体来说，在式（2.10）中，当 $x=0$ 时，参数 σ^2、ω 和 τ 均消失，记此时的 γ 为 γ_0。当 $x = n/2$ 时，参数 γ 和 τ 消失，将相应的参数 σ^2 和 ω 记为 σ_0^2 和 ω_0。由此，可得：

$$f(0) = \gamma_0，\ \text{且} \tag{5.85}$$

$$f(\frac{n}{2}) = \omega_0 + \frac{2}{n}(\sigma_0^2 - \omega_0)。 \tag{5.86}$$

根据 $f(x)$ 的定义，$f(x)$ 的梯度为

$$f'(x) = \frac{4}{n^2}\left(2(\omega+\gamma-2\tau)x + (\sigma^2 - \omega - n\gamma + n\tau)\right)。 \tag{5.87}$$

因 $g(x) = 1/2(f(x) + f(n/2-x)$，有 $g'(x) = 1/2(f'(x) - f'(n/2-x))$。那么，

$$g'(0) = -\frac{2}{n}\gamma_0 - \frac{2}{n^2}(\sigma_0^2 + (n-1)\omega_0)。 \tag{5.88}$$

因为 $g(x)$ 在 $x=0$ 处的切线为 $y = g'(0)x + g(0)$ 且 $y_\mathrm{A} = g(n/4)$，可得：

$$x_\mathrm{A} = \left(g(\frac{n}{4}) - g(0)\right)/g'(0)。 \tag{5.89}$$

因为 $x_\mathrm{A} < x_\mathrm{B}$ 且 $x_\mathrm{B} = n/4$，则有

$$\left(g(\frac{n}{4}) - g(0)\right)/g'(0) < \frac{n}{4}。 \tag{5.90}$$

因为 $g'(0) < 0$，$g(\frac{n}{4}) = f(n/4)$ 且 $g(0) = 1/2(f(0) + f(n/2))$，可得：

$$f(0) + f(\frac{n}{2}) - 2f(\frac{n}{4}) < \gamma_0 + \frac{1}{n}(\sigma_0^2 + (n-1)\omega_0)。 \tag{5.91}$$

因为 $f(0) = \gamma_0$，$\rho_{\mathcal{A},2} = f(\frac{n}{4})/f(\frac{n}{2})$ 且 $f(n/2) > 0$，则有：

$$1 - 2\rho_{\mathcal{A},2} < \frac{1}{n}\frac{\sigma_0^2 + (n-1)\omega_0}{f(n/2)}。 \tag{5.92}$$

将式（5.86）代入上述的不等式中，并代入 $\rho_{\mathcal{A},0}$。根据 $\rho_{\mathcal{A},0}$ 的定义，可得 $\rho_{\mathcal{A},0} = \omega_0/\sigma_0^2$。进而有：

$$\rho_{\mathcal{A},2} > \frac{1}{2} - \frac{1}{2} \cdot \frac{1+(n-1)\rho_{\mathcal{A},0}}{2+(n-2)\rho_{\mathcal{A},0}}。 \tag{5.93}$$

由此，$\rho_{\mathcal{A},2}$ 在式（5.28）中的下界得证。

下面证明 $\rho_{\mathcal{A},1}$ 在式（5.29）中的上界。在图5.7(b) 和 (c) 中，$f(x)$ 在 $x = 0$ 和 $x = n/2$ 处的切线分别用蓝线和绿线表示，且这两条线的斜度分别为 $f'(0)$ 和 $f'(n/2)$。在图5.7(b) 中，$f(x)$ 为单调递增且下凸函数，因此有 $f'(n/2) > f'(0)$。在图5.7(c) 中，尽管 $f(x)$ 在开始时有所递减（$f'(0) < 0$），但可合理假设 $f'(n/2) > -f'(0)$。这个假设在第三 章中的很多实验中得到验证。因此，可合理地假设 $|f'(n/2)| > |f'(0)|$。

根据式（5.87），可得 $f'(0) = -4\gamma_0/n$。当 $\gamma_0 > 0$ 时，$|f'(0)| = 4\gamma_0/n$。当 $\gamma_0 < 0$ 时，$|f'(0)| = -4\gamma_0/n > 4\gamma_0/n$。进而，$|f'(n/2)| = f'(n/2)$。因此，可得 $f'(n/2) > 4\gamma_0/n$。因为 $f'(n/2) = 4/n^2(\sigma_0^2 + (n-1)\omega_0)$，可得

$$\frac{4}{n^2}(\sigma_0^2 + (n-1)\omega_0) > \frac{4}{n}\gamma_0。 \tag{5.94}$$

因为 $f(0) = \gamma_0$，将上述不等式的左右两边同时除以 $f(0)$，可得：

$$\rho_{\mathcal{A},1} < \frac{1+(n-1)\rho_{\mathcal{A},0}}{2+(n-2)\rho_{\mathcal{A},0}}。 \tag{5.95}$$

因此，式（5.29）中 $\rho_{\mathcal{A},1}$ 的上界得以证明。

5.9.6　引理5.4的证明

下面主要提供定理中关于 ρ_1 的证明。ρ_2 的证明与此相似。

根据 ρ_1 的定义，可得对于 $k \neq k'$，有：

$$\begin{aligned}
\rho_1 &\triangleq \frac{\mathrm{Cov}[\hat{\mu}_k^{(j)}, \hat{\mu}_{k'}^{(j)}]}{\mathrm{Var}[\hat{\mu}_k^{(j)}]} = \frac{\mathrm{Cov}[\hat{\mu}_{\mathcal{B},k}^{(j)} - \hat{\mu}_{\mathcal{A},k}^{(j)}, \hat{\mu}_{\mathcal{B},k'}^{(j)} - \hat{\mu}_{\mathcal{A},k'}^{(j)}]}{\mathrm{Var}[\hat{\mu}_{\mathcal{B},k}^{(j)} - \hat{\mu}_{\mathcal{A},k}^{(j)}]} \\
&= \frac{\mathrm{Cov}[\hat{\mu}_{\mathcal{B},k}^{(j)}, \hat{\mu}_{\mathcal{B},k'}^{(j)}] + \mathrm{Cov}[\hat{\mu}_{\mathcal{A},k}^{(j)}, \hat{\mu}_{\mathcal{A},k'}^{(j)}] - 2\mathrm{Cov}[\hat{\mu}_{\mathcal{B},k}^{(j)}, \hat{\mu}_{\mathcal{A},k'}^{(j)}]}{\mathrm{Var}[\hat{\mu}_{\mathcal{B},k}^{(j)}] + \mathrm{Var}[\hat{\mu}_{\mathcal{A},k}^{(j)}] - 2\mathrm{Cov}[\hat{\mu}_{\mathcal{B},k}^{(j)}, \hat{\mu}_{\mathcal{A},k}^{(j)}]} \\
&= \frac{\rho_{\mathcal{B},1}\mathrm{Var}[\hat{\mu}_{\mathcal{B},k}^{(j)}] + \rho_{\mathcal{A},1}\mathrm{Var}[\hat{\mu}_{\mathcal{A},k'}^{(j)}] - 2\rho_{\mathcal{A},\mathcal{B}}\sqrt{\mathrm{Var}[\hat{\mu}_{\mathcal{B},k}^{(j)}]\mathrm{Var}[\hat{\mu}_{\mathcal{A},k'}^{(j)}]}}{\mathrm{Var}[\hat{\mu}_{\mathcal{B},k}^{(j)}] + \mathrm{Var}[\hat{\mu}_{\mathcal{A},k}^{(j)}] - 2\rho_{\mathcal{A},\mathcal{B}}\sqrt{\mathrm{Var}[\hat{\mu}_{\mathcal{B},k}^{(j)}]\mathrm{Var}[\hat{\mu}_{\mathcal{A},k}^{(j)}]}}。
\end{aligned} \tag{5.96}$$

因为 $\rho_{\mathcal{A},\mathcal{B}} > 0$，$max(\rho_{\mathcal{A},1}$ 且 $\rho_{\mathcal{B},1}) \leq 1$，可得：

$$-2\rho_{\mathcal{A},\mathcal{B}} \leq -2max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})\rho_{\mathcal{A},\mathcal{B}}。 \tag{5.97}$$

因为 $\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k}] = \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k'}]$，$\rho_{\mathcal{A},1} \leq max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})$ 且 $\rho_{\mathcal{B},1} \leq max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})$，则有：

$$\begin{aligned}\rho_1 &\leq \frac{max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})\left(\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k'}] - 2\rho_{\mathcal{A},\mathcal{B}}\sqrt{\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}]\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k'}]}\right)}{\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k}] - 2\rho_{\mathcal{A},\mathcal{B}}\sqrt{\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}]\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k}]}} \\ &= max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})。\end{aligned} \tag{5.98}$$

当 $\rho_{\mathcal{A},\mathcal{B}} = 0$ 时，因为 $min(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1}) \leq \rho_{\mathcal{A},1}$ 且 $min(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1}) \leq \rho_{\mathcal{B},1}$，可得：

$$\begin{aligned}\rho_1 &= \frac{\rho_{\mathcal{B},1}\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \rho_{\mathcal{A},1}\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k'}]}{\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k}]} \\ &\geq \frac{min(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})\left(\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k'}]\right)}{\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k}]} = min(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})。\end{aligned} \tag{5.99}$$

进而，因为 $\rho_{\mathcal{A},1} \leq max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})$ 且 $\rho_{\mathcal{B},1} \leq min(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})$，则有：

$$\begin{aligned}\rho_1 &= \frac{\rho_{\mathcal{B},1}\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \rho_{\mathcal{A},1}\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k'}]}{\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k}]} \\ &\leq \frac{max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})\left(\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k'}]\right)}{\mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{B},k}] + \mathrm{Var}[\hat{\mu}^{(j)}_{\mathcal{A},k}]} = max(\rho_{\mathcal{A},1}, \rho_{\mathcal{B},1})。\end{aligned} \tag{5.100}$$

5.9.7 定理5.4 的证明

下面依次给出定理5.4 中三条性质的证明。

5.9.7.1 定理5.4 的第一条性质的证明

因为 $\hat{f}_{m+1} > \hat{f}_m$，当 $\alpha < 0.5$ 时，可得：

$$t_{\alpha/2}(\hat{f}_m) > t_{\alpha/2}(\hat{f}_{m+1}) > 0。 \tag{5.101}$$

对于函数 $E[\hat{C}_m\hat{\sigma}_m]$，根据式（5.23），可得：

$$\begin{aligned}E[\hat{C}_m\hat{\sigma}_m] &= \hat{C}_m E[\hat{\sigma}_m] = \sqrt{\frac{2m+1}{2m-1}}\sqrt{\frac{v_m}{m}}\frac{\Gamma(\frac{f_m+1}{2})}{\Gamma(\frac{f_m}{2})}\sigma \\ &= \sigma\sqrt{v_m}\cdot\sqrt{\frac{2m+1}{(2m-1)m}}\frac{\Gamma(\frac{f_m+1}{2})}{\Gamma(\frac{f_m}{2})}。\end{aligned} \tag{5.102}$$

其中，$\sigma > 0$ 为常数。除了 σ 外，式（5.102）右部的余下部分可分为两部分，分别为 $\sqrt{v_m}$ 和 $\sqrt{\frac{2m+1}{(2m-1)m}}\frac{\Gamma(\frac{f_m+1}{2})}{\Gamma(\frac{f_m}{2})}$。

对于第一部分，根据它在式（5.19）中的定义，则对于 $0 < \rho_1, \rho_2 < 1/2$，有 $v_m > 0$。对 $\sqrt{v_m}$ 求导，可得

$$\frac{\partial\sqrt{v_m}}{\partial m} = \frac{\partial\sqrt{v_m}}{\partial v_m}\cdot\frac{\partial v_m}{\partial m} = \frac{1}{2\sqrt{v_m}}\frac{\partial v_m}{\partial m}$$

$$
\begin{aligned}
&= \frac{\partial\sqrt{v_m}}{\partial v_m} \cdot \Big(Z^{-2}\Big((1-\rho_1)^2 + (1+\rho_1-2\rho_2)^2\Big)\Big(2m - (1+\rho_1+2(m-1)\rho_2)\Big) \\
&\quad -(2-2\rho_2)\Big(m(1-\rho_1)^2 + (m-1)(1+\rho_1-2\rho_2)^2\Big)\Big) \\
&= -\frac{\partial\sqrt{v_m}}{\partial v_m} \cdot \Big(\Big(1+\rho_1+(2m-1)\rho_2\Big)(1-\rho_1)^2 - (3+\rho_1-4\rho_2)(1+\rho_1-2\rho_2)^2\Big)。
\end{aligned} \tag{5.103}
$$

其中 $Z = 2m - (1+\rho_1+2(m-1)\rho_2)$。对于 $0 < \rho_1, \rho_2 < 1/2$，则有 $\partial\sqrt{v_m}/\partial m < 0$。也就是说，$\sqrt{v_m}$ 为关于 m 的递减函数。由此可得：

$$
\sqrt{v_m} > \sqrt{v_{m+1}} > 0。 \tag{5.104}
$$

对于第二部分，记 $q(m) = \sqrt{\frac{2m+1}{(2m-1)m}}\frac{\Gamma(\frac{f_m+1}{2})}{\Gamma(\frac{f_m}{2})}$，表5.11 和表5.12 表明当 $m \geq 2$ 时，$q(m)$ 是关于 m 的递减函数。因此，可得：

$$
\sqrt{\frac{2m+1}{(2m-1)m}}\frac{\Gamma(\frac{f_m+1}{2})}{\Gamma(\frac{f_m}{2})} > \sqrt{\frac{2m+3}{(2m+1)(m+1)}}\frac{\Gamma(\frac{f_{m+1}+1}{2})}{\Gamma(\frac{f_{m+1}}{2})} > 0。 \tag{5.105}
$$

将 σ、式（5.104）、式（5.105）和式（5.101）乘在一起，可得：

$$
E[\hat{C}_m\hat{\sigma}_m \cdot t_\alpha(\hat{f}_m)] > E[\hat{C}_{m+1}\hat{\sigma}_{m+1} \cdot t_\alpha(\hat{f}_{m+1})] > 0。 \tag{5.106}
$$

也就是说，CI_m 的期望长度为关于 m 的递减函数。

5.9.7.2　定理5.4 的第二条性质的证明

CI_m 的期望长度，即：$E[2\hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)]$，可被重写为：

$$
E[2\hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)] = 2\hat{C}_m\sqrt{\frac{v_m\sigma^2}{m}}\frac{\Gamma(\frac{f_m+1}{2})}{\Gamma(\frac{f_m}{2})}t_{\alpha/2}(\hat{f}_m) = 2\hat{C}_m\sqrt{\frac{v_m f_m}{m}}\sigma\frac{\Gamma(\frac{f_m+1}{2})}{\sqrt{f_m}\Gamma(\frac{f_m}{2})}t_{\alpha/2}(\hat{f}_m)。
$$

根据式（5.21），可得 $\sqrt{v_m f_m/m} = \sqrt{2\sigma^{-2}E[\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]]}$。将其带入上式，可得：

$$
E[2\hat{C}_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)] = 2\hat{C}_m\sqrt{2E[\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]]}\frac{\Gamma(\frac{f_m+1}{2})}{\sqrt{f_m}\Gamma(\frac{f_m}{2})}t_{\alpha/2}(\hat{f}_m)。 \tag{5.107}
$$

因为

$$
\lim_{m\to\infty}\hat{C}_m = 1, \tag{5.108}
$$

$$
\lim_{m\to\infty} E[\widehat{\mathrm{Var}}[\hat{\mu}_{m\times 2}]] = \sigma^2(1-\rho_2), \tag{5.109}
$$

$$
\lim_{\alpha\to\infty}\frac{\Gamma(\frac{\alpha+1}{2})}{\sqrt{\alpha}\Gamma(\frac{\alpha}{2})} = \frac{1}{\sqrt{2}}, \tag{5.110}
$$

$$
\lim_{m\to\infty} f_m = \infty, \tag{5.111}
$$

表 5.11 函数 $q(m)$ 的模拟值

ρ_1	m	f_m						$q(m)$					
		ρ_2 :0.0	0.1	0.2	0.3	0.4	0.5	0.0	0.1	0.2	0.3	0.4	0.5
0.0	2	3.000	2.970	2.864	2.667	2.373	2.000	1.030	1.024	1.003	0.962	0.897	0.809
	3	5.000	4.944	4.742	4.349	3.753	3.000	1.028	1.021	0.998	0.952	0.876	0.771
	4	7.000	6.919	6.622	6.036	5.136	4.000	1.024	1.017	0.994	0.945	0.866	0.754
	5	9.000	8.894	8.503	7.723	6.519	5.000	1.020	1.014	0.990	0.941	0.859	0.744
	6	11.000	10.870	10.385	9.412	7.903	6.000	1.017	1.011	0.987	0.938	0.855	0.737
	7	13.000	12.845	12.266	11.101	9.287	7.000	1.015	1.009	0.985	0.935	0.852	0.733
	8	15.000	14.821	14.148	12.789	10.671	8.000	1.014	1.007	0.984	0.933	0.849	0.730
	9	17.000	16.796	16.030	14.479	12.056	9.000	1.012	1.006	0.982	0.932	0.847	0.727
	10	19.000	18.772	17.912	16.168	13.440	10.000	1.011	1.005	0.981	0.931	0.846	0.725
	11	21.000	20.747	19.795	17.857	14.825	11.000	1.010	1.004	0.980	0.930	0.845	0.723
	12	23.000	22.723	21.677	19.547	16.209	12.000	1.010	1.003	0.979	0.929	0.844	0.722
0.1	2	2.972	3.000	2.962	2.829	2.579	2.215	1.024	1.030	1.022	0.995	0.943	0.861
	3	4.951	5.000	4.930	4.672	4.172	3.433	1.022	1.028	1.020	0.990	0.930	0.833
	4	6.930	7.000	6.898	6.519	5.769	4.651	1.018	1.024	1.016	0.985	0.922	0.820
	5	8.910	9.000	8.867	8.366	7.367	5.870	1.015	1.020	1.012	0.982	0.917	0.812
	6	10.890	11.000	10.836	10.214	8.966	7.090	1.012	1.017	1.010	0.979	0.914	0.807
	7	12.870	13.000	12.805	12.063	10.565	8.309	1.010	1.015	1.007	0.977	0.911	0.803
	8	14.850	15.000	14.774	13.911	12.165	9.528	1.009	1.014	1.006	0.975	0.909	0.800
	9	16.830	17.000	16.743	15.760	13.764	10.748	1.007	1.012	1.005	0.974	0.908	0.798
	10	18.810	19.000	18.712	17.609	15.364	11.967	1.006	1.011	1.003	0.973	0.907	0.796
	11	20.791	21.000	20.681	19.457	16.963	13.186	1.005	1.010	1.003	0.972	0.906	0.795
	12	22.771	23.000	22.651	21.306	18.563	14.406	1.004	1.010	1.002	0.971	0.905	0.794
0.2	2	2.882	2.965	3.000	2.951	2.778	2.455	1.006	1.023	1.030	1.020	0.985	0.916
	3	4.800	4.939	5.000	4.909	4.571	3.920	1.005	1.021	1.028	1.017	0.978	0.898
	4	6.721	6.914	7.000	6.868	6.368	5.388	1.001	1.017	1.024	1.013	0.973	0.889
	5	8.643	8.889	9.000	8.828	8.167	6.857	0.999	1.013	1.020	1.010	0.969	0.883
	6	10.565	10.864	11.000	10.787	9.966	8.327	0.996	1.011	1.017	1.007	0.966	0.879
	7	12.488	12.840	13.000	12.747	11.765	9.797	0.994	1.009	1.015	1.005	0.964	0.876
	8	14.411	14.815	15.000	14.707	13.564	11.267	0.993	1.007	1.014	1.003	0.962	0.874
	9	16.333	16.791	17.000	16.667	15.364	12.737	0.992	1.006	1.012	1.002	0.961	0.872
	10	18.256	18.766	19.000	18.627	17.163	14.207	0.991	1.005	1.011	1.001	0.960	0.871
	11	20.179	20.742	21.000	20.586	18.963	15.677	0.990	1.004	1.010	1.000	0.959	0.869
	12	22.102	22.717	23.000	22.546	20.763	17.148	0.989	1.003	1.010	0.999	0.958	0.869

$$\lim_{m\to\infty} t_{\alpha/2}(\hat{f}_m) = u_{\alpha/2}, \tag{5.112}$$

可得：

$$\lim_{m\to\infty} 2E[\hat{C}'_m\hat{\sigma}_m t_{\alpha/2}(\hat{f}_m)] = 2\sigma\sqrt{1-\rho_2}u_{\alpha/2}。 \tag{5.113}$$

也就是说，CI_m 的期望长度的下界为 $2\sigma\sqrt{1-\rho_2}u_{\alpha/2}$。

表 5.12　函数 $q(m)$ 的模拟值（续）

ρ_1	m	f_m						$q(m)$					
		ρ_2 :0.0	0.1	0.2	0.3	0.4	0.5	0.0	0.1	0.2	0.3	0.4	0.5
0.3	2	2.730	2.854	2.955	3.000	2.935	2.701	0.975	1.001	1.021	1.030	1.017	0.969
	3	4.555	4.753	4.922	5.000	4.878	4.418	0.976	1.000	1.019	1.028	1.014	0.960
	4	6.385	6.657	6.891	7.000	6.823	6.139	0.974	0.996	1.015	1.024	1.010	0.954
	5	8.218	8.561	8.859	9.000	8.768	7.861	0.972	0.994	1.012	1.020	1.006	0.950
	6	10.052	10.466	10.828	11.000	10.714	9.584	0.971	0.991	1.009	1.017	1.004	0.947
	7	11.886	12.371	12.797	13.000	12.659	11.307	0.969	0.990	1.007	1.015	1.001	0.944
	8	13.720	14.277	14.766	15.000	14.605	13.031	0.968	0.988	1.006	1.014	1.000	0.942
	9	15.554	16.182	16.736	17.000	16.551	14.754	0.967	0.987	1.004	1.012	0.999	0.941
	10	17.389	18.088	18.705	19.000	18.497	16.478	0.966	0.986	1.003	1.011	0.997	0.940
	11	19.223	19.994	20.674	21.000	20.442	18.202	0.966	0.985	1.002	1.010	0.997	0.939
	12	21.058	21.899	22.643	23.000	22.388	19.926	0.965	0.985	1.002	1.010	0.996	0.938
0.4	2	2.522	2.667	2.814	2.941	3.000	2.909	0.931	0.962	0.992	1.018	1.030	1.012
	3	4.232	4.455	4.688	4.898	5.000	4.829	0.937	0.964	0.992	1.016	1.028	1.008
	4	5.951	6.250	6.568	6.857	7.000	6.750	0.938	0.963	0.989	1.012	1.024	1.004
	5	7.672	8.048	8.448	8.817	9.000	8.672	0.937	0.962	0.987	1.009	1.020	1.000
	6	9.395	9.846	10.330	10.776	11.000	10.595	0.937	0.960	0.985	1.007	1.017	0.998
	7	11.118	11.645	12.211	12.736	13.000	12.517	0.936	0.959	0.983	1.005	1.015	0.996
	8	12.841	13.444	14.093	14.696	15.000	14.440	0.935	0.958	0.982	1.003	1.014	0.994
	9	14.564	15.244	15.975	16.656	17.000	16.363	0.935	0.957	0.980	1.002	1.012	0.993
	10	16.288	17.043	17.857	18.615	19.000	18.286	0.934	0.956	0.980	1.001	1.011	0.992
	11	18.012	18.843	19.739	20.575	21.000	20.209	0.934	0.956	0.979	1.000	1.010	0.991
	12	19.736	20.643	21.621	22.535	23.000	22.132	0.934	0.955	0.978	0.999	1.010	0.990
0.5	2	2.273	2.416	2.579	2.756	2.919	3.000	0.874	0.907	0.943	0.980	1.014	1.030
	3	3.857	4.070	4.319	4.595	4.861	5.000	0.890	0.917	0.948	0.981	1.012	1.028
	4	5.452	5.735	6.067	6.440	6.806	7.000	0.894	0.919	0.948	0.979	1.008	1.024
	5	7.049	7.402	7.818	8.287	8.751	9.000	0.896	0.920	0.947	0.977	1.005	1.020
	6	8.647	9.070	9.570	10.135	10.696	11.000	0.897	0.919	0.946	0.975	1.003	1.017
	7	10.246	10.739	11.322	11.983	12.642	13.000	0.897	0.919	0.945	0.973	1.001	1.015
	8	11.845	12.409	13.074	13.832	14.587	15.000	0.897	0.919	0.944	0.972	0.999	1.014
	9	13.444	14.078	14.827	15.680	16.533	17.000	0.897	0.919	0.943	0.971	0.998	1.012
	10	15.044	15.748	16.580	17.529	18.479	19.000	0.897	0.918	0.943	0.970	0.997	1.011
	11	16.644	17.417	18.333	19.378	20.425	21.000	0.897	0.918	0.942	0.970	0.996	1.010
	12	18.243	19.087	20.086	21.227	22.371	23.000	0.897	0.918	0.942	0.969	0.995	1.010

5.9.7.3　定理5.4 的第三条性质的证明

若 CI_m 为保守的，则 CI_m 的期望长度的下界应大于式（5.7）中其真实值。则可得：

$$2\sigma\sqrt{1-\rho_2}u_{\alpha/2} > 2\sigma\sqrt{\rho_2}u_{\alpha/2}。\tag{5.114}$$

因此，可得 $\rho_2 < 1/2$。

5.10 本章小结

本章将算法比较任务形式化为一个含有用户自定义参数的复合显著性检验问题。该复合显著性检验不仅可以判断两个算法是否等价，还可以判断两个算法性能指标之差是否超过用户自身的期望。因此，该复合假设检验更富有信息量。

基于正则化 $m \times 2$ 交叉验证，本章给出了一套面向算法比较的统计推断方法。本章给出了泛化误差 $m \times 2$ 交叉验证估计的方差的一个保守估计，并构造了一个严格的 t 统计量及相应的置信区间。本文进一步给出一种基于正则化 $m \times 2$ 交叉验证的序贯 t 检验。该序贯 t 检验使用保守的统计推断过程，可以有效地减少假阳性的算法比较结论。进而，本章通过控制置信区间长度的缩减率来为序贯 t 检验选择合适的最大停时。本章从理论和实验两方面来比较该序贯 t 检验与原有的显著性检验。实验结果证实了该序贯 t 检验的优良性，及“序贯”方式的必要性。

在序贯 t 检验假设算法中，本章假设算法性能指标，如，泛化误差，服从正态分布。然而，在文本数据上，常用的算法性能指标并不服从正态分布，例如，准确率、召回率和 F_1 值。对于这些算法性能指标，如何给出基于正则化 $m \times 2$ 交叉验证的统计推断方法，值得进一步研究。

第六章　针对准确率、召回率和 F_1 值的正则化 $m \times 2$ 交叉验证统计推断方法

准确率、召回率和 F_1 是有监督学习算法的常用性能指标。这些性能指标的分布通常是偏峰分布[19]，现有的基于交叉验证的 t 检验并不适用于准确率、召回率和 F_1 的比较。特别是，基于 t 分布得到的准确率、召回率和 F_1 值的置信区间有可能超过 [0,1]，这显然是不合理的。为此，本章考虑使用正则化 $m \times 2$ 交叉验证来校正准确率、召回率和 F_1 值的后验分布，并获得它们的更为精准的区间估计。进而，将算法比较任务转化为显著性检验问题，并提出一种贝叶斯检验来解决该问题。该贝叶斯检验可以基于后验分布直接计算假设成立的概率，并给出比原有 t 检验更丰富的决策信息。本章最后使用文本数据上的分词及命名实体识别实验来说明该贝叶斯检验的有效性。

6.1　问题引入

基于准确率、召回率和 F_1 值的算法比较是机器学习领域的核心任务。从统计观点来看，直接在同一测试集上比较两个算法的准确率、召回率和 F_1 的点估计是不科学的，且易导致比较结论复现度低[58]。一般地，算法比较任务可以被形式化为统计显著性检验问题。针对算法比较任务，目前已有很多流行的显著性检验方法，例如，基于 K 折交叉验证的 t 检验[51],5×2 交叉验证 t 检验及 F 检验，和正则化 3×2 交叉验证 t 检验[8] 等。然而，这些显著性检验并不适用于准确率、召回率和 F_1 值。因为准确率、召回率和 F_1 值为偏峰分布[68] 的，且它们的取值范围为 [0,1]。

本章所提出的的贝叶斯检验比常用的假设检验方法方法[47] 的决策信息更丰富。该贝叶斯检验有三个重要组成部分：（1）正则化 $m \times 2$ 交叉验证[8,85]。它通过有设计地多次重复 2 折交叉验证来优化切分数据集；（2）准确率、召回率和 F_1 值的校正的后验分布以及精准的置信区间，而不是采用原有的正态分布近似；（3）准确率、召回率和 F_1 值的一个贝叶斯检验。它可计算出一个模型优于另一个模型的概率。

从理论上讲，切分数据集时，应尽量保证训练集和验证集的经验分布相同。因此，在比较模型时，2 折交叉验证将语料等分为二份，可能是一个比较合理的选择。正则化 $m \times 2$ 交叉验证重复了多次 2 折交叉验证切分，且其切分可通过正则化条件进行约束。例如，对于文本数据集上的命名实体识别任务，可对训练集和验证集

间不同类型的命名实体块的分布比例进行约束。这些正则化条件可以减少训练集和验证集的分布差异，使得算法比较更为可信。特别是，模型性能评价指标的正则化 $m \times 2$ 交叉验证估计具有方差最小性质。这确保了基于正则化 $m \times 2$ 交叉验证的检验具有更优的势及更好的复现度[8]。

实际上，用 t 分布近似准确率、召回率和 F_1 值的分布是不合适的[?]。王钰等基于组块 3×2 交叉验证给出了 F_1 值的后验分布及置信区间[68]，但该分布并未考虑组块 3×2 交叉验证估计内的相关性。这使得该分布并不准确且不太适用于算法比较。

本章基于正则化 $m \times 2$ 交叉验证给出了准确率、召回率和 F_1 值的准确的后验分布及之心区间，并使用一个贝叶斯检验来计算比较中假设成立的概率。这比常用的假设检验方法方法给出的决策信息更为丰富。最后，本章使用文本数据上的分词和命名实体识别实验来验证该贝叶斯的有效性。

6.2 基于正则化 $m \times 2$ 交叉验证的准确率、召回率及 F_1 值的后验分布

假定 D_n 是一个数据集，其中 n 是 D_n 的大小。例如，在通常的 IID 数据集上，n 为样例的数目；在命名实体的文本语料库中，n 为句子条数。

计算准确率、召回率和 F 值时，数据集 D_n 通常被切分为两部分：一个训练集 S 和一个验证集 T，且满足 $D_n = S \cup T$。假设训练集和验证集的大小为 $|S| = |T| = n/2$。记一个切分为 (S, T)，且记验证集 T 上的混淆矩阵为 $\mathcal{M} = (\mathrm{TP}, \mathrm{FP}, \mathrm{FN}, \mathrm{TN})$，其中，TP，FP，FN，TN 分别表示 true positive，false positive, false negative 和 true negative。基于 $\mathcal{M}$, 准确率、召回率和 F_1 值的计算方式如下：

$$\text{准确率：}\ \mathrm{P} = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FP})}, \tag{6.1}$$

$$\text{召回率：}\ \mathrm{R} = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FN})}, \tag{6.2}$$

$$\mathrm{F}_1\text{值：}\ \mathrm{F}_1 = \frac{2\mathrm{PR}}{\mathrm{P} + \mathrm{R}}。 \tag{6.3}$$

Goutte 等给出了准确率和召回率的概率涵义[19]。具体来讲，混淆矩阵 $\mathcal{M}$ 服从参数为 $\pi = (\pi_{TP}, \pi_{FN}, \pi_{FP}, \pi_{TN})$ 的多项分布。其中，$\pi_{TP} + \pi_{FN} + \pi_{FP} + \pi_{TN} = 1$。则，准确率和召回率分别为下述概率的估计：

$$p = P(l = +|z = +), \tag{6.4}$$

$$r = P(z = +|l = +)。 \tag{6.5}$$

其中，l 和 z 分别表示样本的真实类别和预测类别，且 $+$ 为正类别的标记。相应地，F_1 用来估计 $f_1=2pr/(p+r)$。

记 n_+ 为数据集 D_n 中的正例数。设 (S,T) 为正则化 $m\times 2$ 交叉验证中的一个切分，则 T 中的正例数满足：

$$\mathrm{TP}+\mathrm{FN}=n_+/2。\tag{6.6}$$

6.2.1　Hold-out 验证上准确率、召回率和 F_1 的后验分布

Goutte 等的研究工作表明 $\mathrm{TP}|\mathrm{TP}+\mathrm{FN}$ 服从参数为 $n_+/2$ 和 r 的二项分布[19]。则，

$$\mathrm{Var}[\mathrm{R}]=\mathrm{Var}\left[\frac{2}{n_+}\mathrm{TP}\right]=\frac{2r(1-r)}{n_+},\tag{6.7}$$

其中 $\mathrm{Var}[\cdot]$ 取自 D_n。式（6.7）的证明见第6.5.1 节。

假设 r 服从 Beta 分布，即 $r\sim Be(\lambda,\lambda)$，则 r 的后验分布为 $r|\mathcal{M}\sim Be(\mathrm{TP}+\lambda,\mathrm{FN}+\lambda)$。当 $\lambda=1$ 时，该后验分布的众数为

$$mode[r|\mathcal{M}]=\mathrm{R}。\tag{6.8}$$

类似地，假定 $p\sim Be(\lambda,\lambda)$，则有 $p|\mathcal{M}\sim Be(\mathrm{TP}+\lambda,\mathrm{FP}+\lambda)$，且它的众数为

$$mode[p|\mathcal{M}]=\mathrm{P}。\tag{6.9}$$

基于准确率和召回率的后验分布，王钰等证明了 F_1 值的后验分布[68] 为

$$P(f_1=t|\mathcal{M})=\frac{2^a(1-t)^{a-1}(2-t)^{-a-b}t^{b-1}}{B(a,b)},\tag{6.10}$$

其中，$B(\cdot,\cdot)$ 为 Beta 函数，其参数为 $a=\mathrm{FP}+\mathrm{FN}+2\lambda$ 及 $b=\mathrm{TP}+\lambda$。

6.2.2　基于正则化 $m\times 2$ 交叉验证的准确率、召回率和 F_1 值的后验分布

正则化 $m\times 2$ 交叉验证的定义及切分方式在第3.3 节中给出。这里不再赘述。设 $\boldsymbol{\mathcal{M}}=\{\mathcal{M}^{(j)}\}_{j=1}^m\triangleq\{(\mathcal{M}_1^{(j)},\mathcal{M}_2^{(j)})\}_{j=1}^m$ 为正则化 $m\times 2$ 交叉验证上的混淆矩阵集，其大小为 $2m$。其中，混淆矩阵 $\mathcal{M}_1^{(j)}$ 对应的训练集为 S_j，验证集为 T_j。混淆矩阵 $\mathcal{M}_2^{(j)}$ 对应的训练集为 T_j，验证集为 S_j。记 $\mathcal{M}_k^{(j)}\triangleq(\mathrm{TP}_k^{(j)},\mathrm{FP}_k^{(j)},\mathrm{FN}_k^{(j)},\mathrm{TN}_k^{(j)})$。

本节所关注的是后验分布 $P(p|\boldsymbol{\mathcal{M}})$，$P(r|\boldsymbol{\mathcal{M}})$ 和 $P(f_1|\boldsymbol{\mathcal{M}})$ 的推导。

给定 $\boldsymbol{\mathcal{M}}$，准确率、召回率和 F_1 的微平均值为

$$\mathrm{P}_{m\times 2}=\frac{\frac{1}{m}\sum_{j=1}^m\frac{1}{2}\sum_{k=1}^2\mathrm{TP}_k^{(j)}}{\frac{1}{m}\sum_{j=1}^m\frac{1}{2}\sum_{k=1}^2(\mathrm{TP}_k^{(j)}+\mathrm{FP}_k^{(j)})},\tag{6.11}$$

$$\mathrm{R}_{m\times 2}=\frac{\frac{1}{m}\sum_{j=1}^m\frac{1}{2}\sum_{k=1}^2\mathrm{TP}_k^{(j)}}{\frac{1}{m}\sum_{j=1}^m\frac{1}{2}\sum_{k=1}^2(\mathrm{TP}_k^{(j)}+\mathrm{FN}_k^{(j)})},\tag{6.12}$$

$$\mathrm{F}_{1,m\times 2} = \frac{2\mathrm{P}_{m\times 2}\mathrm{R}_{m\times 2}}{\mathrm{P}_{m\times 2}+\mathrm{R}_{m\times 2}}。 \tag{6.13}$$

本节首先关注召回率的后验分布 $P(r|\boldsymbol{\mathcal{M}})$ 的推导。由于 $\mathrm{TP}_k^{(j)}+\mathrm{FN}_k^{(j)}=n_+/2$ 为一个与 j 和 k 无关的常数，$\mathrm{R}_{m\times 2}$ 可重写为

$$\mathrm{R}_{m\times 2}=\frac{1}{m}\sum_{j=1}^{m}\mathrm{R}^{(j)}=\frac{1}{2m}\sum_{j=1}^{m}\sum_{k=1}^{2}\mathrm{R}_k^{(j)}。 \tag{6.14}$$

其中

$$\mathrm{R}_k^{(j)}=\frac{\mathrm{TP}_k^{(j)}}{\mathrm{TP}_k^{(j)}+\mathrm{FN}_k^{(j)}}, \tag{6.15}$$

$$\mathrm{R}^{(j)}=\frac{\frac{1}{2}\sum_{k=1}^{2}\mathrm{TP}_k^{(j)}}{\frac{1}{2}\sum_{k=1}^{2}(\mathrm{TP}_k^{(j)}+\mathrm{FN}_k^{(j)})}。 \tag{6.16}$$

因此，$\mathrm{R}_{m\times 2}$ 的方差为

$$\mathrm{Var}\left[\mathrm{R}_{m\times 2}\right]=\frac{1+\rho_1+(2m-2)\rho_2}{mn_+}r(1-r)。 \tag{6.17}$$

式（6.17）的证明见第6.5.2 节。ρ_1 和 ρ_2 为 $\mathrm{R}_{m\times 2}$ 所包含的 hold-out 估计间的相关系数。具体地，ρ_1 和 ρ_2 的定义为：

- 定义 $\sigma=\mathrm{Var}\left[\mathrm{R}_k^{(j)}\right]$。根据式（6.7），可得 $\sigma=2r(1-r)/n_+$。
- $\rho_1=\mathrm{Cov}\left[\mathrm{R}_1^{(j)},\mathrm{R}_2^{(j)}\right]/\sigma$ 为召回率的同一个 2 折交叉验证估计 $\mathrm{R}^{(j)}$ 中两个 hold-out 估计的相关系数。
- $\rho_2=\mathrm{Cov}\left[\mathrm{R}_k^{(j)},\mathrm{R}_{k'}^{(j')}\right]/\sigma$ 为不同的两个 2 折交叉验证中召回率的两个 hold-out 估计的相关系数。其中，$j\neq j'$ 且 $k,k'=1,2.$

然而，$\boldsymbol{\mathcal{M}}$ 中的 2m 个混淆矩阵彼此相关，使得似然函数 $p(\boldsymbol{\mathcal{M}}|r)$ 不能写作 $\prod_{j=1}^{m}\prod_{k=1}^{2}p(\mathcal{M}_k^{(j)}|r)$，导致准确率、召回率和 F_1 值的后验分布不能解析表达。这是本节面临的主要难点。

为了解决该难点，本节引入了一个有效混淆矩阵 $\mathcal{M}_e=(\mathrm{TP}_e,\mathrm{FP}_e,\mathrm{FN}_e,\mathrm{TN}_e)$。该矩阵用于度量混淆矩阵集 $\boldsymbol{\mathcal{M}}$ 等价于多少个独立的样本。基于 $\mathcal{M}_e$，可得 $r|\mathcal{M}_e\sim Be(\mathrm{TP}_e+\lambda,\mathrm{FN}_e+\lambda)$，且 $\mathrm{R}_{m\times 2}$ 的方差可以重写为

$$\mathrm{Var}[\mathrm{R}_{m\times 2}]=\frac{r(1-r)}{\mathrm{TP}_e+\mathrm{FN}_e}。 \tag{6.18}$$

比较式（6.17）与式（6.18），可得

$$\begin{aligned}\mathrm{TP}_e+\mathrm{FN}_e &= \frac{mn_+}{1+\rho_1+(2m-2)\rho_2}\\ &= \frac{\sum_{j=1}^{m}\sum_{k=1}^{2}\left(\mathrm{TP}_k^{(j)}+\mathrm{FN}_k^{(j)}\right)}{1+\rho_1+(2m-2)\rho_2}。\end{aligned} \tag{6.19}$$

根据式（6.8），可得

$$mode[r|\boldsymbol{\mathcal{M}}] = \frac{\mathrm{TP}_e}{\mathrm{TP}_e + \mathrm{FN}_e} = \mathrm{R}_{m\times 2}。 \tag{6.20}$$

基于式（6.12），式（6.19）和式（6.20），可得 TP_e 和 FN_e 的表达式为

$$\mathrm{TP}_e = \frac{1}{1+\rho_1+(2m-2)\rho_2}\sum_{j=1}^{m}\sum_{k=1}^{2}\mathrm{TP}_k^{(j)}, \tag{6.21}$$

$$\mathrm{FN}_e = \frac{1}{1+\rho_1+(2m-2)\rho_2}\sum_{j=1}^{m}\sum_{k=1}^{2}\mathrm{FN}_k^{(j)}。 \tag{6.22}$$

根据式（6.9），可得

$$mode[p|\boldsymbol{\mathcal{M}}] = \frac{\mathrm{TP}_e}{\mathrm{TP}_e + \mathrm{FP}_e} = \mathrm{P}_{m\times 2}。 \tag{6.23}$$

基于式（6.11），（6.21）和式（6.23），可得 FP_e 的表达式为

$$\mathrm{FP}_e = \frac{1}{1+\rho_1+(2m-2)\rho_2}\sum_{j=1}^{m}\sum_{k=1}^{2}\mathrm{FP}_k^{(j)}。 \tag{6.24}$$

显然，TP_e、FP_e 和 TN_e 均含有未知参数 ρ_1 和 ρ_2。它们的关系如下：

- 当 $\rho_1 = \rho_2 = 0$ 时，$\mathrm{TP}_e = \sum_{j=1}^{3}\sum_{k=1}^{2}\mathrm{TP}_k^{(j)}$。$\mathrm{FN}_e$ 和 FP_e 也有类似形式。这表明 $r|\boldsymbol{\mathcal{M}}$ 的后验分布等价于在 $2m$ 个独立的语料上获得的后验分布。
- 当 $\rho_1 = \rho_2 = 1$ 时，TP_e、FP_e 和 FN_e 分别为 $\boldsymbol{\mathcal{M}}$ 中所有的 TP、FP 和 FN 的平均值。这种情形下，基于正则化 $m \times 2$ 交叉验证的后验分布与基于单个 hold-out 验证的后验分布非常类似，且正则化 $m \times 2$ 交叉验证中的多个数据切分对后验分布没有实质性贡献。

实际上，召回率可以变相地看作定义在 0-1 损失函数上且仅对属于正类的样例取期望的泛化误差。在泛化误差的正则化 $m \times 2$ 交叉验证估计中，ρ_1 和 ρ_2 的性质已在第三 章和第五章进行了深入的研究。这些研究将 ρ_1 和 ρ_2 的取值范围大体约束在 $0 \le \rho_1 \le 1/2$ 和 $1/4 \le \rho_2 \le 1/2$ 之间。本文认为，这个范围也适用于召回率估计 $\mathrm{R}_{3\times 2}$ 中的 ρ_1 和 ρ_2。因此，为了消除 TP_e，FN_e 和 FP_e 中的未知参数，本节将它们在范围 $0 \le \rho_1 \le 1/2$ 和 $1/4 \le \rho_2 \le 1/2$ 中取平均可得：

$$\begin{aligned}
\mathrm{TP}_e &\approx 8\int_{0.25}^{0.5}\int_{0}^{0.5}\frac{\sum_{j=1}^{3}\sum_{k=1}^{2}\mathrm{TP}_k^{(j)}}{1+\rho_1+4\rho_2}d\rho_1 d\rho_2 \\
&= \frac{4}{m-1}\left\{[(1+\rho_1+(2m-2)\rho_2)\ln(1+\rho_1+(2m-2)\rho_2)]-\rho_2\right\}\Big|_{\rho_1=0}^{0.5}\Big|_{\rho_2=0.25}^{0.5} \\
&\quad \sum_{j=1}^{m}\sum_{k=1}^{2}\mathrm{TP}_k^{(j)}
\end{aligned}$$

$$= \frac{4}{m-1}\left(\ln\frac{(m+0.5)^{m+0.5}(0.5m+0.5)^{0.5m+0.5})}{m^m(1+0.5m)^{1+0.5m}}\right)\sum_{j=1}^{m}\sum_{k=1}^{2}\mathrm{TP}_k^{(j)}。 \tag{6.25}$$

相应地，可得

$$\mathrm{FN}_e \approx \frac{4}{m-1}\left(\ln\frac{(m+0.5)^{m+0.5}(0.5m+0.5)^{0.5m+0.5})}{m^m(1+0.5m)^{1+0.5m}}\right)\sum_{j=1}^{3}\sum_{k=1}^{2}\mathrm{FN}_k^{(j)}, \tag{6.26}$$

$$\mathrm{FP}_e \approx \frac{4}{m-1}\left(\ln\frac{(m+0.5)^{m+0.5}(0.5m+0.5)^{0.5m+0.5})}{m^m(1+0.5m)^{1+0.5m}}\right)\sum_{j=1}^{3}\sum_{k=1}^{2}\mathrm{FP}_k^{(j)}。 \tag{6.27}$$

当 $m=3$ 时，TP_e、FN_e 和 FP_e 分别为

$$\mathrm{TP}_e \approx 0.3688\sum_{j=1}^{3}\sum_{k=1}^{2}\mathrm{TP}_k^{(j)}, \tag{6.28}$$

$$\mathrm{FN}_e \approx 0.3688\sum_{j=1}^{3}\sum_{k=1}^{2}\mathrm{FN}_k^{(j)}, \tag{6.29}$$

$$\mathrm{FP}_e \approx 0.3688\sum_{j=1}^{3}\sum_{k=1}^{2}\mathrm{FP}_k^{(j)}。 \tag{6.30}$$

总之，基于正则化 $m\times 2$ 交叉验证，准确率、召回率和 F_1 值的后验分布为

$$P(p=t|\boldsymbol{\mathcal{M}}) = \frac{t^{\mathrm{TP}_e+\lambda}(1-t)^{\mathrm{FP}_e+\lambda}}{B(\mathrm{TP}_e+\lambda,\mathrm{FP}_e+\lambda)}, \tag{6.31}$$

$$P(r=t|\boldsymbol{\mathcal{M}}) = \frac{t^{\mathrm{TP}_e+\lambda}(1-t)^{\mathrm{FN}_e+\lambda}}{B(\mathrm{TP}_e+\lambda,\mathrm{FN}_e+\lambda)}, \tag{6.32}$$

$$P(f_1=t|\boldsymbol{\mathcal{M}}) = \frac{2^{\bar{a}}(1-t)^{\bar{a}-1}(2-t)^{-\bar{a}-\bar{b}}t^{\bar{b}-1}}{B(\bar{a},\bar{b})}。 \tag{6.33}$$

其中，$p\sim Be(\lambda,\lambda)$，$r\sim Be(\lambda,\lambda)$，$B(\cdot,\cdot)$ 为 Beta 函数，且其参数为 $\bar{a}=\mathrm{FP}_e+\mathrm{FN}_e+2\lambda$ 与 $\bar{b}=\mathrm{TP}_e+\lambda$。本章中，设置 $\lambda=1$。

6.2.3 基于正则化 $m\times 2$ 交叉验证的准确率、召回率和 F_1 值的置信区间

基于上节给出的后验分布，准确率、召回率和 F_1 的置信区间很容易求出。置信概率为 $1-\alpha$ 时，准确率的置信区间为

$$\mathrm{CI}_p = [Be_{\frac{\alpha}{2}}(\mathrm{TP}_e+\lambda,\mathrm{FP}_e+\lambda), Be_{1-\frac{\alpha}{2}}(\mathrm{TP}_e+\lambda,\mathrm{FP}_e+\lambda)]。 \tag{6.34}$$

召回率的置信区间为

$$\mathrm{CI}_r = [Be_{\frac{\alpha}{2}}(\mathrm{TP}_e+\lambda,\mathrm{FN}_e+\lambda), Be_{1-\frac{\alpha}{2}}(\mathrm{TP}_e+\lambda,\mathrm{FN}_e+\lambda)]。 \tag{6.35}$$

F_1 值的置信区间为

$$\mathrm{CI}_{f_1}=\left[\frac{2}{2+Be'_{1-\frac{\alpha}{2}}},\frac{2}{2+Be'_{\frac{\alpha}{2}}}\right]。\tag{6.36}$$

其中，Be'_{α} 为 Beta-prime 分布的 α 分位数，且该 Beta-prime 分布的参数为 $\mathrm{FP}_e+\mathrm{FN}_e+2\lambda$ 和 $\mathrm{TP}_e+\lambda$。

上述的置信区间比原先提出的置信区间[68] [64] 更为准确，因为后验分布的参数均使用正则化 $m\times 2$ 交叉验证中的相关系数进行校正。以 F_1 值为例，考虑 $m=3$。王钰等给出了不同于本文的一个置信区间[68]。该置信区间使用了 $\boldsymbol{\mathcal{M}}$ 中所有 TP, FP 和 FN 的平均值，其形式等同于式（6.36）在 $\rho_1=\rho_2=1$ 时的特殊形式。因此，该置信区间更为保守。也就是说，该置信区间的置信度大于名义上的置信概率（$1-\alpha$）。但是，本节所给的置信区间缓解了这种保守性，因此，它更为精确。具体如下例所示。

例 6.1　考虑 Wang 等给出的一个模拟二类分类实验[68]。在该实验中，数据集的一个样本为 $Z=(X,Y)$，其中，$P(Y=1)=P(Y=0)=\frac{1}{2}$，$X|Y=0\sim N(\mu_0,\Sigma_0)$，$X|Y=1\sim N(\mu_1,\Sigma_1)$。设置 $\mu_0=(0,0)$，$\mu_1=(0.5,0.5)$，$\Sigma_0=\Sigma_1=I_2$。数据集大小为 $n=600$，且 $\alpha=0.05$。基于 Logistic 回归算法，王钰等所给的置信区间的置信度和区间长度[68] 分别为 99.6% 和 0.117。相比而言，本节所给的置信区间的置信度和区间长度为 94.5% 和 0.0854。显然，本节的置信区间的置信度更靠近 $1-\alpha$，且其区间长度更短。这表明本节所给的置信区间更为精确。

6.3　基于正则化 $m\times 2$ 交叉验证的贝叶斯检验

对于一个机器学习任务，假设 $\mathcal{A}$ 为在数据集 D_n 上的一个优良算法。当新开发一个机器学习算法 $\mathcal{B}$ 时，需要将它与算法 $\mathcal{A}$ 进行比较，以判断算法 $\mathcal{B}$ 的性能是否显著优于 $\mathcal{A}$。该比较问题可以转化为如下假设检验问题。

$$H_0:\nu_{\mathcal{B}}-\nu_{\mathcal{A}}\leq 0\ \ v.s.\ \ H_1:\nu_{\mathcal{B}}-\nu_{\mathcal{A}}>0。\tag{6.37}$$

其中，$\nu_{\mathcal{A}}$ 和 $\nu_{\mathcal{B}}$ 分别为算法 $\mathcal{A}$ 和 $\mathcal{B}$ 的性能指标值。本节中，算法性能指标仅采用准确率、召回率和 F_1 值。

本节采用贝叶斯检验[96] 来解决问题（6.37）。贝叶斯检验可以避免常用的假设检验方法[1] [6] [7] 中的很多缺点[71]，例如，p 值的问题。更重要的是，贝叶斯检验可以直接计算出假设成立的概率。这可有效地帮助用户做出更为合理的决策。因此，贝叶斯检验近几年广受好评，且被推荐为用于分析实验结果的标准工具[71]。

本节的贝叶斯检验使用准确率、召回率和 F_1 值的正则化 $m \times 2$ 交叉验证后验分布来分别计算问题（6.37）中两个假设成立的概率，即：$P(H_0)$ 和 $P(H_1)$。接着，该检验使用启发式规则来推导出相应的决策。例如，Casella 等给出的一种启发式规则[96] 为：若 $P(H_0) \geq P(H_1)$，则接受 H_0；否则，接受 H_1。

在给出该贝叶斯检验之前，先引入一些必要的记号：记算法 $\mathcal{A}$ 的 $\boldsymbol{\mathcal{M}}$ 为 $\boldsymbol{\mathcal{M}}_{\mathcal{A}}$；记算法 $\mathcal{A}$ 的 TP_e，FN_e 和 FP_e 为 $\mathrm{TP}_{e,\mathcal{A}}$，$\mathrm{FN}_{e,\mathcal{A}}$ 和 $\mathrm{FP}_{e,\mathcal{A}}$。记算法 $\mathcal{A}$ 的准确率、召回率和 F_1 值为 $p_{\mathcal{A}}$，$r_{\mathcal{A}}$ 和 $f_{1,\mathcal{A}}$。对于算法 $\mathcal{B}$，记号可类似定义。设 ν 为准确率、召回率和 F_1 值中的一种指标，其值由用户给定。例如，当用户设置 ν 为召回率时，则比较 $r_{\mathcal{A}}$ 和 $r_{\mathcal{B}}$。

构造贝叶斯检验的关键之处在于如何计算 $\nu_{\mathcal{A}} - \nu_{\mathcal{B}}$ 的分布。然而，该分布并不对应常用的分布函数。因此，本节采用蒙特卡洛模拟对其近似。以召回率为例，给定 $\boldsymbol{\mathcal{M}}_{\mathcal{A}}$ 和 $\boldsymbol{\mathcal{M}}_{\mathcal{B}}$，假设 $r_{\mathcal{A}}$ 和 $r_{\mathcal{B}}$ 相互独立，则概率 $p(r_{\mathcal{A}} - r_{\mathcal{B}} \leq 0 \big| \boldsymbol{\mathcal{M}}_{\mathcal{A}}, \boldsymbol{\mathcal{M}}_{\mathcal{B}})$ 等价于

$$\int_0^1 \int_0^1 \mathbb{I}(r_{\mathcal{A}} - r_{\mathcal{B}} \leq 0) P(r_{\mathcal{A}} | \boldsymbol{\mathcal{M}}_{\mathcal{A}}) P(r_{\mathcal{B}} | \boldsymbol{\mathcal{M}}_{\mathcal{B}}) dr_{\mathcal{A}} dr_{\mathcal{B}}。 \tag{6.38}$$

其中，$\mathbb{I}(\cdot)$ 为示性函数，即，当括号内的条件成立时，其值为 1，否则为 0。显然，式（6.38）没有显式形式。本节所用的蒙特卡洛方法具体包含两步：（1）从后验分布 $P(r_{\mathcal{A}} | \boldsymbol{\mathcal{M}}_{\mathcal{A}})$ 和 $P(r_{\mathcal{B}} | \boldsymbol{\mathcal{M}}_{\mathcal{B}})$ 中抽取出足够数量的样例，并将其分别记为 $\{s_{i,\mathcal{A}}\}_{i=1}^{L}$ 和 $\{s_{i,\mathcal{B}}\}_{i=1}^{L}$。（2）采用如下式所示的经验值来估计式（6.38）：

$$\frac{1}{L} \sum_{i=1}^{L} \mathbb{I}(s_{i,\mathcal{A}} - s_{i,\mathcal{B}} \leq 0)。 \tag{6.39}$$

其中，L 设置为 1 000 000。

基于上述分析，算法6.1 给出了该贝叶斯检验的伪代码。基于 ν 的取值，该算法可以分别对准确率、召回率和 F_1 值进行检验。不同的性能指标对应于不同的假设检验问题。即使检验中使用了相同的数据集，在不同的性能指标上，贝叶斯检验所给出的最终决策也不同。基于不同的算法性能指标，该贝叶斯检验可以帮助用户从不同的视角细粒度地呈现算法 $\mathcal{A}$ 和 $\mathcal{B}$ 之间的差别。

贝叶斯检验和常用的假设检验方法分别来自于贝叶斯学派和频率学派。这两个学派对于统计的认识在哲学层面有较大的分歧。不过，当算法性能指标的分布已知时，贝叶斯检验可能会给出更有信息的推断过程和决策。目前为止，用于比较贝叶斯检验和常用的假设检验方法的准则尚缺乏。因此，本章不准备对两种方法进行比较，仅使用文本数据上的分词和命名实体识别任务的实验来说明贝叶斯检验的有效性。

算法 6.1 用于比较两个算法的准确率、召回率和 F_1 值的贝叶斯检验

输入： 数据集，D_n；
待比较的算法，$\mathcal{A}$ 和 $\mathcal{B}$；
算法性能指标，ν；
重复次数，m；

输出： 问题（6.37）中两个假设成立的概率及集合 {“接受 H_0”，“接受 H_1”} 中的一个决策；

在数据集 D_n 上，使用第3.3 节的构造算法来构造正则化 $m \times 2$ 交叉验证的切分集 $\mathbb{P}$；
使用数据集 D_n 和切分集 $\mathbb{P}$，训练算法 $\mathcal{A}$ 和 $\mathcal{B}$，并测试其性能，将测试所得的混淆矩阵分别记为 $\boldsymbol{\mathcal{M}}_{\mathcal{A}}$ 和 $\boldsymbol{\mathcal{M}}_{\mathcal{B}}$；
在 $\boldsymbol{\mathcal{M}}_{\mathcal{A}}$ 和 $\boldsymbol{\mathcal{M}}_{\mathcal{B}}$ 上，使用式（6.25），（6.26）和（6.27）分别得到 $(\mathrm{TP}_{e,\mathcal{A}}, \mathrm{FN}_{e,\mathcal{A}}, \mathrm{FP}_{e,\mathcal{A}})$ 和 $(\mathrm{TP}_{e,\mathcal{B}}, \mathrm{FN}_{e,\mathcal{B}}, \mathrm{FP}_{e,\mathcal{B}})$；
if ν 的值为“准确率” **then**
    $P(\nu_{\mathcal{A}}|\boldsymbol{\mathcal{M}}_{\mathcal{A}}) \leftarrow$ 将 $\mathrm{TP}_{e,\mathcal{A}}$ 和 $\mathrm{FP}_{e,\mathcal{A}}$ 代入式（6.31）中；
    $P(\nu_{\mathcal{B}}|\boldsymbol{\mathcal{M}}_{\mathcal{B}}) \leftarrow$ 将 $\mathrm{TP}_{e,\mathcal{B}}$ 和 $\mathrm{FP}_{e,\mathcal{B}}$ 代入式（6.31）中；
else
    if ν 的值为“召回率” **then**
        $P(\nu_{\mathcal{A}}|\boldsymbol{\mathcal{M}}_{\mathcal{A}}) \leftarrow$ 将 $\mathrm{TP}_{e,\mathcal{A}}$ and $\mathrm{FN}_{e,\mathcal{A}}$ 代入式（6.32）中；
        $P(\nu_{\mathcal{B}}|\mathcal{M}_{\mathcal{B}}) \leftarrow$ 将 $\mathrm{TP}_{e,\mathcal{B}}$ 和 $\mathrm{FN}_{e,\mathcal{B}}$ 代入式（6.32）中
    end if
else
    if ν 的值为“F_1” **then**
        $P(\nu_{\mathcal{A}}|\boldsymbol{\mathcal{M}}_{\mathcal{A}}) \leftarrow$ 将 $\mathrm{TP}_{e,\mathcal{A}}$，$\mathrm{FP}_{e,\mathcal{A}}$ 和 $\mathrm{FN}_{e,\mathcal{A}}$ 代入式（6.33）中；
        $P(\nu_{\mathcal{B}}|\boldsymbol{\mathcal{M}}_{\mathcal{B}}) \leftarrow$ 将 $\mathrm{TP}_{e,\mathcal{B}}$，$\mathrm{FP}_{e,\mathcal{B}}$ 和 $\mathrm{FN}_{e,\mathcal{B}}$ 代入式（6.33）中；
    end if
end if
使用蒙特卡洛近似估计 $P(\nu_{\mathcal{A}} - \nu_{\mathcal{B}} \leq 0 \big| \boldsymbol{\mathcal{M}}_{\mathcal{A}}, \boldsymbol{\mathcal{M}}_{\mathcal{B}})$（参考式（6.39））；
$P(H_0) \leftarrow P(\nu_{\mathcal{A}} - \nu_{\mathcal{B}} \leq 0 \big| \boldsymbol{\mathcal{M}}_{\mathcal{A}}, \boldsymbol{\mathcal{M}}_{\mathcal{B}})$；
$P(H_1) \leftarrow 1 - P(\nu_{\mathcal{A}} - \nu_{\mathcal{B}} \leq 0 \big| \boldsymbol{\mathcal{M}}_{\mathcal{A}}, \boldsymbol{\mathcal{M}}_{\mathcal{B}})$；
if $P(H_0) \geq P(H_1)$ **then**
    return （$P(H_0)$，$P(H_1)$，“接受 H_0”）；
else
    return （$P(H_0)$，$P(H_1)$，“接受 H_1”）；
end if

6.4 实验及分析

本节主要考虑文本数据上中文分词任务（CWS），命名实体识别任务（NER）等。这些任务均可以被形式化为序列标注问题，并采用适当的标记集合和序列标注算法来解决。常用标记集合包括 IOB2，IOBES 等[97] [98]。常用的序列标注算法包括条件随机场（CRFs)[99] 及长短期记忆神经网络算法（LSTM）等[100] [101]。

本节通过在中文分词模型及命名实体识别模型上实施贝叶斯检验来回答如下研究问题：细粒度的标记集合是否可以改善中文分词模型及命名实体识别的性能？

中文分词模型及命名实体识别模型的性能指标通常包括准确率、召回率和 F_1。在它们的计算方式中，TP 表示预测正确的块数。FN 表示语料中未被正确预测的块数。FP 表示被预测出但不包含在语料中的块数。

实验中考虑三种不同的块务：

(1) 中文分词任务（CWS）：为一条生句子识别出合理的词序列。词可看作由字组成的块，且每个字均包含在一个块中。Bakeoff-2005 CWS PKU 训练语料被用作实验语料 D_n。

(2) 命名实体识别任务（NER）：仅识别命名实体的边界，不标注其类型。CoNLL 2003 英文 NER 训练语料被用作实验语料 D_n，其中包含“PER”、“LOC”、“ORG”和“MISC”四种命名实体。词语被用作标注单位，且语料中存在很多块外词。

(3) 组织名识别任务（NER-ORG）：仅识别组织名类型的实体。语料与 NER 任务相同。但组织名类型的实体块数明显小于命名实体任务中命名实体的块数，且块外词主导了整个语料。

对于上述三种任务，本节主要使用条件随机场序列标注算法。其它算法在未来工作中进行研究。所有实验均设置 $m = 3$。

6.4.1 中文分词任务：对比“BMES”和“BB_2B_3MES”

中文分词任务可被形式化为字层面的序列标注任务。本节考虑两种不同的标记标记集合：“BMES”和“BB_2B_3MES”。它们分别对应算法 $\mathcal{A}$ 和 $\mathcal{B}$。标记集合“BB_2B_3MES”在标记集合“BMES”的基础上增加了两个标记‘B_2’和‘B_3’来细粒度地刻画字在词中的位置。Zhao 等的研究结论[102] 表明算法 $\mathcal{B}$ 优于算法 $\mathcal{A}$，但他们并未考察它们的差别是否显著。本节将对它们的差别是否显著进行考察。

在分词任务中，字的一元、二元和三元组合特征被用作特征，且各类特征的窗口均设置为 [-2,2]。所对比的两个分词模型的正则化 3×2 交叉验证后验分布在图6.1 中给出。各性能指标的置信区间在表6.1 中给出。图6.1 中，每条概率密度线的开始位置和结束位置分别为其 0.001 分位数和 0.999 分位数。实线对应基于式（6.31）、（6.32）和（6.33）的概率密度函数。

从图6.1 中可以得出两点结论。第一，本章所给出的后验分布的概率密度函数比王钰等给出的概率密度函数[68] 更高且更尖。这说明本章的后验分布经过正则化 $m \times 2$ 交叉验证中相关性的校正，变得更为精准。第二，对于算法 $\mathcal{A}$ 和 $\mathcal{B}$，召回率和 F_1 值的两个后验分布的重叠面积小于准确率的两个后验分布的重叠面积。这说

表 6.1　准确率、召回率和 F_1 值的置信区间（$\alpha = 0.05$）

任务类型	算法性能指标 ν	标记集 1 (%)	标记集 2 (%)
中文分词任务		BMES	BB_2B_3MES
	准确率	[95.55, 95.62]	[95.60, 95.67]
	召回率	[95.04,95.11]	[95.16,95.23]
	F_1 值	[95.30,95.36]	[95.39,95.44]
命名实体识别任务		IOB2	IOBES
	准确率	[90.59, 91.30]	[90.70,91.41]
	召回率	[87.69,88.48]	[87.78, 88.57]
	F_1 值	[89.21,89.77]	[89.32,89.87]
组织名识别任务		IOB2	IOBES
	准确率	[91.37,92.86]	[91.85,93.31]
	召回率	[64.89,67.11]	[64.45,66.68]
	F_1 值	[76.06,77.74]	[75.93,77.61]

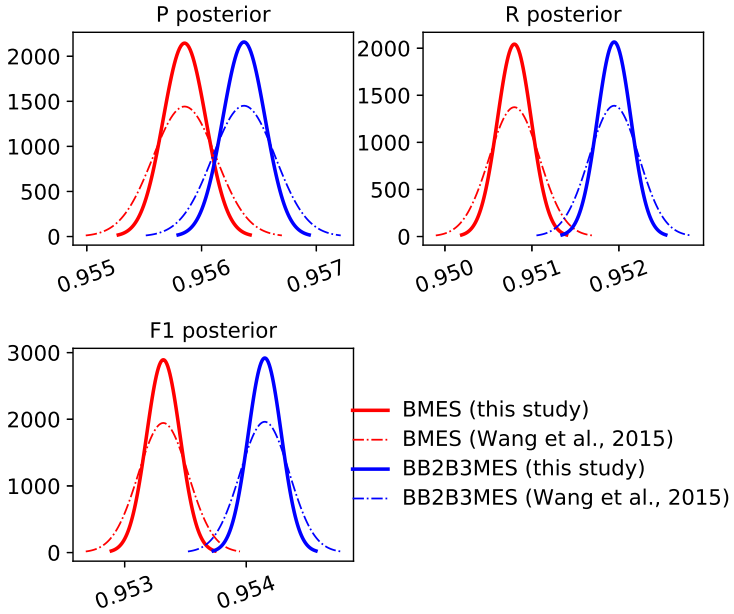

图 6.1　中文分词任务中准确率、召回率和 F_1 值的正则化 3×2 交叉验证后验分布

明，细粒度的标记集合主要改进了召回率和 F_1 值。

表6.2给出了中文分词任务上贝叶斯检验的决策信息及两个假设成立的概率。对于准确率，假设 H_1 成立的概率为 0.97。对于召回率和 F_1 值，假设 H_1 成立的概率非常接近 1。表6.2同样表明细粒度的标记集合显著改进了中文分词模型的性能，且对于召回率和 F_1 值的改进要大于准确率。

表 6.2 中文分词任务中贝叶斯检验的决策信息

算法性能指标 ν	$P(H_0)$	$P(H_1)$	决策
准确率	0.024	0.976	接受 H_1
召回率	0.001	0.999	接受 H_1
F_1 值	0.001	0.999	接受 H_1

6.4.2 命名实体识别任务：对比“IOB2”和“IOBES”

在命名实体识别任务中，词语和雌性被用作特征。特征模板具体包含词语和词性的一元组、二元组和三元组。每种特征的窗口均设为 [-2,2]。标记集合“IOB2”和“IOBES”分别对应于模型 $\mathcal{A}$ 和 $\mathcal{B}$。标记集合“IOBES”在“IOB2”的基础上增加了‘E’和‘S’两个标记。

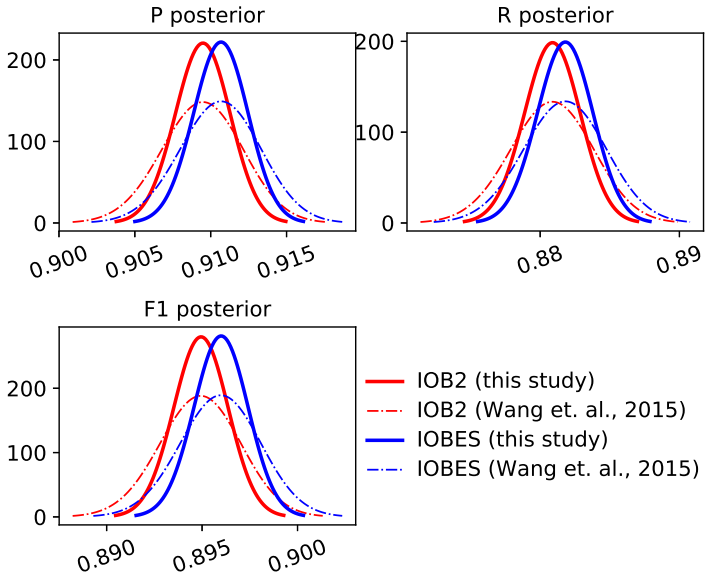

图 6.2 命名实体识别任务中准确率、召回率和 F_1 值的正则化 3×2 交叉验证后验分布

表 6.3 命名实体识别任务中贝叶斯检验的决策信息

算法性能指标 ν	$P(H_0)$	$P(H_1)$	决策
准确率	0.320	0.680	接受 H_1
召回率	0.373	0.627	接受 H_1
F_1 值	0.300	0.700	接受 H_1

图6.2中给出了准确率、召回率和 F_1 值的后验分布。在三种指标上，算法 $\mathcal{A}$ 和 $\mathcal{B}$ 的后验分布有很大的重叠。这表明算法 $\mathcal{B}$ 对命名体识别任务的改进并不明显。相

应的置信区间在表6.1 中给出。两个模型的置信区间也有很大的重叠，这同样表明算法 $\mathcal{A}$ 和 $\mathcal{B}$ 之间的差别并不显著。表6.3 给出了本任务上贝叶斯检验的决策信息。尽管三个决策均为“接受 H_1”，但其改进并不明显，因为所有性能指标上，$P(H_1)$ 的概率均低于 0.8。另外，细粒度标记集“IOBES”对准确率的改进大于召回率，因为准确率对应的 $P(H_1) = 0.78$ 大于召回率对应的 $P(H_1) = 0.68$。

6.4.3　组织名识别任务：对比“IOB2”和“IOBES”

组织名识别任务所用的特征设置与命名实体任务的设置相同。但是，本任务中，标记的分布比命名实体任务中标记的分布更偏，其中，标记“O”所占的比例更大。因此，本任务中，贝叶斯给出的决策信息与命名实体任务中的决策信息明显不同。图6.3 给出了准确率、召回率和 F_1 值的后验分布，并表明算法 $\mathcal{B}$ 的改进并不明显。令人惊讶的是，对于召回率和 F_1 值，算法 $\mathcal{B}$ 的后验分布图像偏移到算法 $\mathcal{A}$ 的左边。这表明细粒度标记集“IOBES”带来了召回率和 F_1 值的下降。这可能是由于细粒度标记集使得标注标记的分布更偏所导致的。

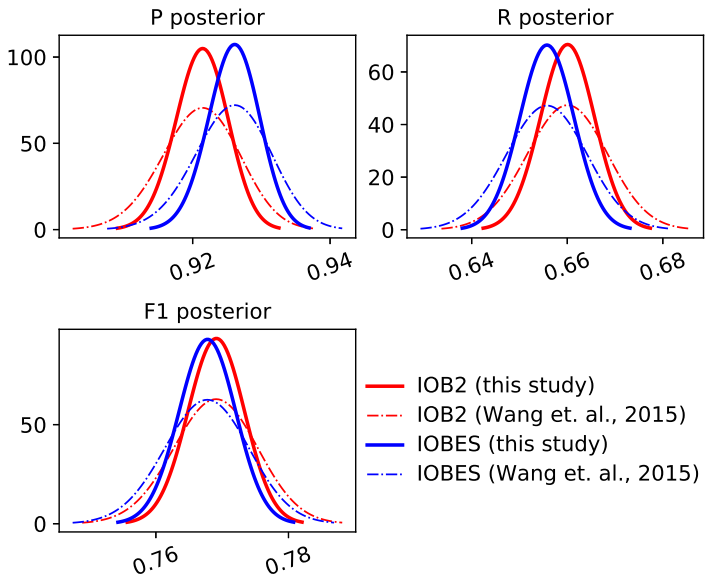

图 6.3　组织名识别任务中准确率、召回率和 F_1 值的正则化 3×2 交叉验证后验分布

表 6.4　组织名识别任务中贝叶斯检验的决策信息

算法性能指标 ν	$P(H_0)$	$P(H_1)$	决策
准确率	0.191	0.809	接受 H_1
召回率	0.706	0.294	接受 H_0
F_1 值	0.587	0.413	接受 H_0

表6.4中给出了贝叶斯检验的决策信息。细粒度标记集合对于准确率的改进的概率超过了 0.8，即：$P(H_1) = 0.85$。但是，细粒度标记集却带来了召回率和 F_1 值的下降。如表所示，对于召回率，有 $P(H_0) = 0.76$；对于 F_1 值，有 $P(H_0) = 0.62$。

6.4.4 小结

上述分词及命名实体识别任务的三组实验结果表明了本章的贝叶斯检验的有效性。该贝叶斯检验可为每种任务的准确率、召回率和 F_1 值提供准确的置信区间及假设成立的概率 $P(H_0)$ 和 $P(H_1)$。这些结果可以为后续的析因提供有价值的信息，且可以帮助用户做出更为合理的决策。

6.5 附录

6.5.1 式（6.7）的推导

根据式（6.2）和式（6.6），可得

$$\mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = \frac{2\mathrm{TP}}{n_+}。 \tag{6.40}$$

由于 $\mathrm{TP}|\mathrm{TP} + \mathrm{FN}$ 服从参数为 $n_+/2$ 和 r 的二项分布，其中，$\mathrm{TP} + \mathrm{FN} = n_+/2$ 为常数，可得

$$\mathrm{Var}[\mathrm{TP}] = \frac{n_+}{2} r(1-r)。 \tag{6.41}$$

根据式（6.40），可得

$$\mathrm{Var}[\mathrm{R}] = \mathrm{Var}[\frac{2\mathrm{TP}}{n_+}] = \frac{4}{n_+^2}\mathrm{Var}[\mathrm{TP}] = \frac{2r(1-r)}{n_+}。 \tag{6.42}$$

6.5.2 式（6.17）的推导

根据式（6.14）可知, $\mathrm{Var}\left[\mathrm{R}_{3\times 2}\right]$ 有如下分解

$$\mathrm{Var}\left[\mathrm{R}_{m\times 2}\right] = \mathrm{Var}\left[\frac{1}{m}\sum_{j=1}^{m}\mathrm{R}^{(j)}\right] = \frac{1}{m^2}\left\{\sum_{j=1}^{m}\mathrm{Var}\left[\mathrm{R}^{(j)}\right] + \sum_{\substack{j=1\\ j\neq j'}}^{m}\sum_{j'=1}^{m}\mathrm{Cov}\left[\mathrm{R}^{(j)}, \mathrm{R}^{(j')}\right]\right\}。 \tag{6.43}$$

假定 $\mathrm{Var}\left[\mathrm{R}^{(j)}\right]$ 不依赖于特定的 $\mathbf{P}_j$，则对于任意的 j，$\mathrm{Var}\left[\mathrm{R}^{(j)}\right]$ 是相同的。进而，对于任意的 $j \neq j'$，因为任意两个切分的训练集间重叠样本个数恒等于 $n/4$，可合理地假设 $\mathrm{Cov}\left[\mathrm{R}^{(j)}, \mathrm{R}^{(j')}\right]$ 均相同且不依赖于 j 和 j'。由此可得：

$$\mathrm{Var}\left[\mathrm{R}_{m\times 2}\right] = \frac{1}{m}\left\{\mathrm{Var}\left[\mathrm{R}^{(j)}\right] + 2\mathrm{Cov}\left[\mathrm{R}^{(j)}, \mathrm{R}^{(j')}\right]\right\}。 \tag{6.44}$$

因为 $\mathrm{R}^{(j)} = (\mathrm{R}_1^{(j)} + \mathrm{R}_2^{(j)})/2$，假定 $\mathrm{Var}\left[\mathrm{R}_1^{(j)}\right] = \mathrm{Var}\left[\mathrm{R}_2^{(j)}\right]$，则有

$$\mathrm{Var}\left[\mathrm{R}^{(j)}\right] = \mathrm{Var}\left[\frac{1}{2}(\mathrm{R}_1^{(j)} + \mathrm{R}_2^{(j)})\right] = \frac{1}{2}\left\{\mathrm{Var}\left[\mathrm{R}_k^{(j)}\right] + \mathrm{Cov}\left[\mathrm{R}_k^{(j)}, \mathrm{R}_{k'}^{(j)}\right]\right\}。 \tag{6.45}$$

其中 $k \neq k'$。进而，根据式（6.7）以及本章中 ρ_1 的定义，可得

$$\mathrm{Var}\left[\mathrm{R}_k^{(j)}\right] = 2r(1-r)/n_+, \tag{6.46}$$

$$\mathrm{Cov}\left[\mathrm{R}_k^{(j)}, \mathrm{R}_{k'}^{(j)}\right] = 2\rho_1 r(1-r)/n_+。 \tag{6.47}$$

将式（6.46）和式（6.47）代入式（6.45）中，可得

$$\mathrm{Var}\left[\tilde{\mathrm{R}}^{(j)}\right] = \frac{1+\rho_1}{n_+} r(1-r)。 \tag{6.48}$$

类似地，假定 $\mathrm{Cov}\left[\mathrm{R}_k^{(j)}, \mathrm{R}_{k'}^{(j')}\right]$ 不依赖于 k 和 k'，则有

$$\mathrm{Cov}\left[\mathrm{R}^{(j)}, \mathrm{R}^{(j')}\right] = \mathrm{Cov}\left[\mathrm{R}_k^{(j)}, \mathrm{R}_{k'}^{(j')}\right]。 \tag{6.49}$$

其中 $k, k' = 1, 2$。根据本章中 ρ_2 的定义，可得

$$\mathrm{Cov}\left[\mathrm{R}^{(j)}, \mathrm{R}^{(j')}\right] = \frac{2\rho_2}{n_+} r(1-r)。 \tag{6.50}$$

将式（6.48）和式（6.50）代入式（6.44）中，可得

$$\mathrm{Var}\left[\mathrm{R}_{m\times 2}\right] = \frac{1+\rho_1+(2m-2)\rho_2}{mn_+} r(1-r)。 \tag{6.51}$$

6.6 本章小结

本章基于正则化 $m \times 2$ 交叉验证推导出了精确的准确率、召回率和 F_1 值的后验分布。该后验分布是用于比较两个算法的重要基础。基于这些后验分布，本章给出了一种贝叶斯检验方法。该检验可直接计算出算法比较中的假设成立的概率。这可以帮助用户做出更为合理的决策。最后，本章使用分词、命名实体识别任务的相关实验来验证该贝叶斯检验的有效性。

未来的工作应基于正则化 $m \times 2$ 交叉验证，深入分析混淆矩阵上的其它性能指标的后验分布及置信区间。这也是机器学习领域的重要研究问题。

第七章　正则化 $m \times 2$ 交叉验证在软件缺陷预测任务上的应用

本章给出正则化 $m \times 2$ 交叉验证及其统计推断方法的一个具体应用场景：软件缺陷预测任务。

软件缺陷预测任务旨在使用机器学习算法，从软件的历史数据中学习出缺陷预测模型，然后对新的软件模块潜在的缺陷进行预测，以帮助软件开发人员在后续开发中合理分配资源，节省时间，提高软件开发效率[103]。缺陷预测模型的预测结果有多种表现形式。预测结果既可以仅简单呈现新软件模块是否有缺陷，也可以给出新软件模块中缺陷数量，甚至可以告知用户新软件模块的源代码中缺陷出现的具体位置。其中，仅预测新软件模块是否具有缺陷的任务，被称为**缺陷倾向性预测任务**。预测新软件模块中缺陷数量的任务，被称为**缺陷数预测任务**。缺陷倾向性预测任务和缺陷数预测任务是当前软件缺陷预测的主流任务。本章仅关注这两类任务。

7.1　软件缺陷预测任务的特点

在软件缺陷预测任务中，软件历史数据常以软件模块为基本单位进行收集。每个软件模块的特性可以通过多种度量进行量化。常用的度量多达五六十种，主要包括一些静态的软件复杂度度量[104]，如代码行数、Halstead 复杂度、圈复杂度等，以及一些软件运行时的动态度量[105]，如传递消息数、接受消息数、类实例数等。软件历史数据常以 IID 数据集的形式进行存储和使用。IID 数据集中的每条样例对应于一个软件模块，每个特征对应于软件模块的一种度量。基于这种形式的软件历史数据，软件缺陷预测任务常被看作一个机器学习的建模问题，也必须采用合理的算法比较方法来对比和分析各种不同的学习算法，以保障选择到优良的缺陷预测模型。这为本文的正则化 $m \times 2$ 交叉验证提供了一个重要的应用场景。

缺陷倾向性预测任务旨在判断一个新软件模块是否有缺陷，对应于机器学习中的二分类问题，可用 logistic 回归、支持向量机等分类器进行建模。缺陷倾向性预测任务的历史数据集使用正类和负类来标识软件模块是否存在缺陷，其中，正类表示软件模块存在缺陷。常见的缺陷倾向性数据集规模通常不大，含有数百到数千条样例不等，且类别不均衡问题非常明显[106]。正类的样例占比通常小于 20%。从类别不均衡的角度分析，常用的 5 折和 10 折交叉验证容易使得正类样例在验证集中出现的频次过少，可能会导致缺陷倾向性预测模型的指标估计大幅度波动，不利于得到可靠的算法比较结论。相比而言，在缺陷倾向性预测数据上，正则化 $m \times 2$ 交

叉验证不仅可以将训练集间的重叠个数“正则化”到 $n/4$ 左右（n 为数据集大小），还可以引入“分层”约束将正例在训练集和验证集中占比“正则化”成相同的比例。这可以保障正则化 $m \times 2$ 交叉验证在缺陷倾向性预测任务上得到可靠的算法比较结论。

缺陷数预测任务使用软件模块的缺陷个数作为响应变量。该任务常被形式化为回归问题，并采用线性回归，泊松回归及负二项回归等算法进行建模[107,108]。在缺陷数预测任务的数据集上，缺陷数常常表现出重尾分布[109]。此类数据集上，若切分方法不当，会导致训练集和验证集上缺陷数的分布明显不同，所形成的的算法比较结论也不可靠。不过，正则化 $m \times 2$ 交叉验证，可引入关于缺陷数的正则化条件，来有效地减少训练集和验证集上缺陷数的分布差异。针对缺陷数的正则化条件，只需将数据集中所有样例按缺陷数从大到小排序，然后轮询地排布在 $n/4$ 等份数据子块上（见定理3.2），再使用正交表对 $n/4$ 个数据子块进行多次拼合，便可得到满足缺陷数的正则化条件的正则化 $m \times 2$ 交叉验证。

7.2 软件缺陷预测任务中算法比较方法的研究现状

大多数软件缺陷预测的研究工作主要致力于搜寻性能优良的学习算法[110]、学习算法的最优参数[111] 以及可明显改善预测模型性能的优良度量[112–114]。这些研究工作均需要使用优良的算法对比方法来确保所得结论的可靠性。不过，目前来看，大多数研究工作采用了多种不同的算法比较方法，得出的结论也相互矛盾[115]，复现性差。

下面分别从交叉验证和显著性检验两方面来分析算法比较方法在软件缺陷预测任务中的使用情况。

在交叉验证的使用情况来看，软件缺陷预测方面的研究工作对交叉验证方法的选择多种多样[116]，较为主观。例如，Lessmann 等采用十折交叉验证来构建模型[110]。Jing 等研究者采用 RHS 方法来验证软件缺陷预测模型的性能[117]。Gong 等采用 20 次随机的 RLT 方法，切分比例为 7:3[118]。这些研究工作并未给所用交叉验证方法的选择标准和依据。为了解决交叉验证方法的选择问题，Tantithamthavorn 等分析了 256 篇相关文献，选取了 12 种常用的交叉验证方法，并在 101 个公开的缺陷预测数据集上比较了这些交叉验证估计的方差和偏差。他们发现 out-of-sample bootstrap 方法可以很好地权衡方差和偏差。因此，他们推荐将 out-of-sample bootstrap 用于后续的软件缺陷预测建模中。不过，他们选取交叉验证的准则是模型性能估计的优劣，而不是从算法比较方法入手。此外，上述研究工

作所用的交叉验证方法均采用随机的数据切分，并未结合软件缺陷预测数据集的特点对交叉验证的数据切分进行优化设计。本文认为，正则化 $m \times 2$ 交叉验证，在算法比较任务中取得了不错的效果，可为软件缺陷预测中算法选择和度量选择等任务的研究奠定很好的基础。

从显著性检验的角度来看，软件缺陷预测的大多数工作倾向于采用非参检验。例如，Jing 等采用了 Friedman 检验[117]。Tantithamthavorn 等人推荐使用 Scott-Knott ESD 检验[116]。Lessmann 等人采用了 Wilcoxon 秩检验[110]。不过，相比非参检验，参数检验具有较优的势。研究者应结合算法性能指标的分布特性，尽量选择合适的参数检验。在缺陷倾向性预测中，当算法性能指标为准确率、召回率和 F_1 值时，可选择第六 章提出的贝叶斯检验。在缺陷数预测任务中，平均绝对误差、平均相对误差、fault percentile average（FPA）[119] 常作为算法性能指标。它们的分布可近似看作正态分布。因此，正则化 $m \times 2$ 交叉验证序贯 t 检验（见第五 章）可应用于这三种算法性能指标。

7.3　正则化 $m \times 2$ 交叉验证序贯 t 检验在缺陷数预测任务上的应用

在软件缺陷预测任务的建模过程中，正则化 $m \times 2$ 交叉验证可以用在算法选择、超参数选择、度量选择等多个阶段，来提高这些阶段的结论的可靠性。本章所用的正则化 $m \times 2$ 交叉验证对重叠样本个数和缺陷个数的分布均进行了“正则化”。

下面以 KC1 回归数据集上的缺陷数预测为例，展示正则化 $m \times 2$ 交叉验证在度量选择上的具体应用。以下的实验结果取自同一个研究小组关于静态软件缺陷预测模型的相关工作[120]。

对于面向对象语言开发的软件中，缺陷数预测模型通常需要判断一个类文件中所含潜在缺陷的数目。类层面的度量大多由类所含多个方法的度量聚合而成。聚合方式具体包括求和（sum），取平均（avg），最大（max）和最小（min）四种方式。举例来说，在方法层面，度量“NUM_OPERATORS”用来量化一个方法所含的操作数的个数。在类层面，该度量可以被抽取成四种聚合度量，分别为“sumNUM_OPERATORS”、“avgNUM_OPERATORS”、“maxNUM_OPERATORS”和“minNUM_OPERATORS”。这四种度量具体指“类文件所含的操作数总数”，“类中方法所含操作数的平均值”、“类中方法所含操作数的最大值”和“类中方法所含操作数的和值”。KC1 回归数据集共包含 145 个类。每个类包含 94 个度量，其中 10 个为常用的 C&K 度量，其余 84 个由方法层面的 21 个度量聚合而成。

在 KC1 回归数据集上，首先使用最大信息系数准则初选出 52 个优良度量，然后对每个度量做幂次变换以增强该度量与预测数间的线性相关性，再用主成分分析对度量矩阵压缩变换，并采用负二项回归建立缺陷数预测模型[120]。在此过程中，主成分个数的确定、主成分的具体解释及主成分对模型贡献的评定是建模的关键之处。

在上述建模过程中，主成分个数选取验证集上最大的性能指标 FPA 估计对应的主成分个数。主成分个数被确定后，因子载荷变换被用于解释主成分的作用。本节将实验分为“对度量做幂次变换”和“不对度量做幂次变换”两种情况进行讨论。

- 当不做度量的幂次变换时，主成分个数被确定为 9 个。这些主成分上的因子载荷旋转[120]，揭示了前 3 个因子有很好的解释性：第 1 个因子对应 sum 聚合特征；第 2 个因子对应 avg 聚合特征；第 3 个因子对应 max 聚合特征；其余 6 个特征没有明显的对应关系。
- 对度量做幂次变换后，主成分个数被确定为 2 个。通过因子载荷旋转[120]，2 个主成分对应的因子有良好的解释：第 1 个因子对应 max 和 avg 类型的聚合特征；第 2 个因子对应 sum 聚合特征。

为了确定第一主成分对缺陷数预测模型的贡献，本文设立如下假设检验问题。

$$H_0 : \mu_{\mathcal{A}} - \mu_{\mathcal{B}} \leq 0; \quad v.s. \quad H_1 : \mu_{\mathcal{A}} - \mu_{\mathcal{B}} > 0。 \tag{7.1}$$

其中，$\mu_{\mathcal{A}}$ 和 $\mu_{\mathcal{B}}$ 分别为原缺陷数预测模型性能及删除了第一主成分后的模型性能。本节采用的性能指标为 FPA。本节使用正则化 $m \times 2$ 交叉验证 t 检验来解决该假设检验问题。检验结果在表7.1 中给出。

表7.1表明，删除第一主成分后，序贯检验仅适用 3×2 交叉验证便可得到检验结论，p 值均小于 0.05，模型的 FPA 值显著减小。也就是说，第一主成分显著影响缺陷数预测模型的性能。

7.4 基于正则化 $m \times 2$ 交叉验证的贝叶斯检验在缺陷倾向性预测任务上的应用

缺陷倾向性预测任务常被看作二分类问题，可使用常用的分类算法进行建模，并用准确率、召回率和 F_1 值等算法性能指标来评价。本节使用正则化 $m \times 2$ 交叉验证贝叶斯检验（见第六章）来比较 logistic 回归与随机森林两种分类算法在缺陷倾向性预测任务中的性能。所用的正则化 $m \times 2$ 交叉验证不仅约束了训练集间的重叠样本个数，而且对正类样例和负类样例在训练集和验证集上的比例也进行了“正则化”。本节设置 $m = 3$。

表 7.1　缺陷数预测模型中第一主成分的贡献

		FPA	
		不做幂次变换	做幂次变换
原模型	指标估计	0.796	0.789
	标准差	0.032	0.033
删除第一主成分后的模型	指标估计	0.561	0.668
	标准差	0.077	0.063
序贯 t 检验	m	3	3
	自由度	5	5
	统计量的值	3.038	3.041
	p 值	0.014	0.009
	结论	拒绝 H_0	拒绝 H_0

实验使用 KC1 分类数据集，来预测类文件所含的每个方法是否具有缺陷。该数据集含 21 个度量、2107 条样例。其中，正类（含缺陷）样例 325 条、负类（不含缺陷）样例 1782 条，缺陷率为 15.4%。每条样例对应类文件中的一个方法。

在建模过程中，本节首先使用下采样方法确保训练集内正例和负例分布比例相同，然后使用主成分分析对度量矩阵压缩变换，并采用分类算法建立软件倾向性预测模型。其中，主成分个数使用 F_1 值来确定，具体选取 F_1 值的正则化 3×2 交叉验证估计的最大值对应的主成分个数。表7.2 给出了 logistic 回归和随机森林两种算法的主成分个数及其对应的准确率、召回率和 F_1 值。

表 7.2　缺陷倾向性预测任务中 Logistic 回归和随机森林的准确率、召回率和 F_1 值的比较

	logistic 回归	随机森林
主成分个数	4	15
准确率（%）	32.52	31.59
召回率（%）	69.44	79.18
F_1 值（%）	44.29	45.16
准确率的置信区间（%）	[29.30, 35.91]	[28.64, 34.70]
召回率的置信区间（%）	[64.48, 73.97]	[74.68, 83.05]
F_1 值的置信区间（%）	[40.67, 47.89]	[41.74, 48.56]

在缺陷倾向预测任务上，logistic 回归和随机森林的准确率、召回率和 F_1 值的后验分布在图7.1 中给出。从该图中可以观察到：（1）本文所给后验分布的概率密度函数比王钰等所给后验分布[68] 的概率密度函数更高更瘦。这是因为本文使

用了正则化 $m \times 2$ 交叉验证估计中的相关系数对后验分布进行校正，使其更为准确；（2）对于准确率和 F_1 值，logistic 回归和随机森林的后验分布有很大的重叠。这说明这两种分类算法在准确率和 F_1 值上没有显著的差别；（3）对于召回率，随机森林的后验分布在 logistic 的后验分布的右侧，且重叠很少。这说明随机森林显著改善了缺陷倾向性预测模型的召回率。此外，针对 logistic 回归和随机森林，表7.2给出了准确率、召回率和 F_1 值的 95% 后验置信区间。从这些后验置信区间也可看出，两种分类算法在准确率和 F_1 值上的差别不明显，但在召回率上有明显差别。

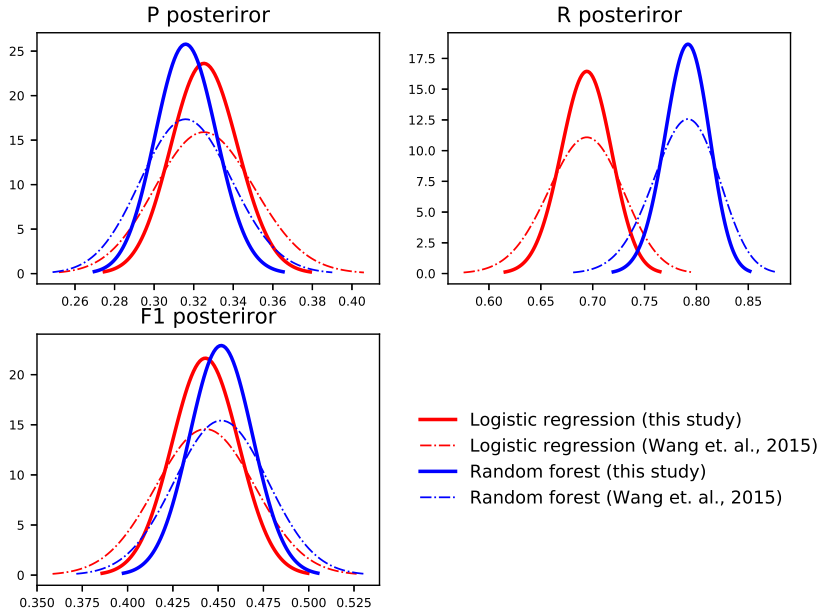

图 7.1 缺陷倾向性预测任务中准确率、召回率和 F_1 值的正则化 3×2 交叉验证后验分布

表 7.3 缺陷倾向性预测任务中贝叶斯检验的决策信息

算法性能指标 ν	$P(H_0)$	$P(H_1)$	决策
准确率	0.657	0.343	接受 H_0
召回率	0.001	0.999	接受 H_1
F_1 值	0.365	0.635	接受 H_1

为了进一步比较 logistic 回归和随机森林在缺陷倾向性预测模型上的性能，本节设立了如下假设检验问题。

$$H_0 : \nu_{\mathcal{B}} - \nu_{\mathcal{A}} \leq 0 \quad v.s. \quad H_1 : \nu_{\mathcal{B}} - \nu_{\mathcal{A}} > 0。 \tag{7.2}$$

其中，$\nu_{\mathcal{A}}$ 和 $\nu_{\mathcal{B}}$ 分别为 logistic 回归和随机森林的性能指标值。这里，算法性能指

标仅采用准确率、召回率和 F_1 值。

本节使用基于正则化 3×2 交叉验证的贝叶斯检验（见算法6.1）来解决问题（7.2）。该检验的决策在表7.3中给出。表7.3表明，随机森林算法的准确率以 $P(H_1) = 0.343$ 的概率略差于 logistic 回归的准确率。不过，随机森林的召回率显著好于 logistic 回归的召回率，相应的概率达到 $P(H_1) = 0.999$。综合来看，随机森林的 F_1 值略优于 logistic 回归的 F_1 值，但优势不明显，相应的概率为 $P(H_1) = 0.635$。

7.5　本章小结

本章将正则化 $m \times 2$ 交叉验证应用在软件缺陷预测任务中，来说明正则化 $m \times 2$ 交叉验证在建模中的作用。针对缺陷数预测模型，本章将正则化 $m \times 2$ 交叉验证序贯 t 检验用于检验各聚合特征对模型性能是否有显著影响的问题中。针对缺陷倾向性预测模型，本章将基于正则化 $m \times 2$ 交叉验证的贝叶斯检验用于比较 logistic 回归和随机森林两种分类算法的性能。

在软件缺陷预测任务中，算法比较方法直接影响着结论的可靠性，是非常重要的研究课题。在未来工作中，本文将针对软件缺陷预测任务的特点，进一步发展正则化 $m \times 2$ 交叉验证及其统计推断方法。

结论及展望

优良的算法比较方法是构建现代智能软件的核心技术，也为研究者高效开发新型的机器学习算法提供了重要保障。本文面向算法比较任务，针对通常的 IID 数据和文本数据的特点，综合分析泛化误差、准确率、召回率和 F_1 值等算法性能指标的特性，从交叉验证数据切分的优化及算法比较的统计推断两个方面进行研究，初步形成了一个基于正则化交叉验证的算法比较统计推断的方法体系。本文以软件缺陷预测任务为例，给出了正则化交叉验证在缺陷数预测模型上的一个具体应用。本文的主要研究结论和创新总结如下：

一、揭示了训练集间重叠样本个数与算法性能指标的交叉验证估计的方差间的关系。 对泛化误差的 RLT 估计的方差及 $m \times 2$ 交叉验证估计的方差进行深入分析。研究了交叉验证估计所含的任意两个 hold-out 估计间的协方差与训练集间重叠样本个数的关系，发现两个 hold-out 估计的协方差是训练集间重叠样本个数的下凸函数。进而，证明泛化误差的两个 2 折交叉验证估计的协方差为其训练集间重叠样本个数的下凸对称函数，且对称轴为 $n/4$ （n 为数据集大小）。这些性质为正则化交叉方法的理论创新提供了前提。

二、建立了正则化交叉验证方法。以最小化泛化误差的交叉验证估计的方差为目标，建立相应的优化问题，并引入相应的正则化条件，来优化交叉验证的数据切分方式，以建立正则化交叉验证方法。对于 RLT 方法，引入了两个正则化条件：1）所有重叠样本个数的总和最小；和 2）训练集间所有重叠样本个数尽量相同。进而，提出了正则化 RLT 方法，从理论上保证了泛化误差的正则化 RLT 估计的方差小于随机切分情形下 RLT 估计的方差。对于 $m \times 2$ 交叉验证，引入一个正则化条件将训练集间重叠样本个数约束至 $n/4$ （n 为数据集大小），并建立了正则化 $m \times 2$ 交叉验证方法。本文证明泛化误差的正则化 $m \times 2$ 交叉验证估计的方差小于随机切分情形下 $m \times 2$ 交叉验证估计的方差。对于文本数据，本文引入卡方度量来度量训练集和验证集间多种频次分布的不同。基于该卡方度量构造正则化条件来约束该分布之差，并将该正则化条件引入正则化 $m \times 2$ 交叉验证中，构造了适用于文本数据的正则化 $m \times 2$ 交叉验证方法。本文以汉语框架语义角色标注任务为例，说明了文本数据的正则化 $m \times 2$ 交叉验证方法的有效性。

三、提出了正则化交叉验证方法的高效构造算法。对于正则化 RLT 方法，本文针对训练集大小和切分次数的三种特定情形，给出了快速的构造算法。对于正则

化 $m \times 2$ 交叉验证，本文给出了一种高效增量式构造算法。正则化条件通常要求重叠样本个数在多个训练集上具有均衡相等的性质，因此可以利用实验设计中的常用工具——正交表——对正则化交叉验证的数据切分进行高效构造。特别是，正则化 $m \times 2$ 交叉验证的增量式构造算法为后续的序贯检验提供了重要基础。

四、面向算法比较任务，提出了基于正则化 $m \times 2$ 交叉验证的统计推断方法。基于正则化 $m \times 2$ 交叉验证，本文给出了面向算法比较的统计推断方法，主要包括方差估计的构造、置信区间的构造以及显著性检验。针对泛化误差，本文给出了 $m \times 2$ 交叉验证估计方差的一个合理的保守估计，并提出了一个服从 t 分布的检验统计量，进而给出了对应的置信区间，并证明了该置信区间的期望长度下界、保守性等理论性质。本文深入分析了正则化 $m \times 2$ 交叉验证估计中相关性的理论性质，并给出了这些相关性的合理范围。在此基础上，本文提出了基于 $m \times 2$ 交叉验证的序贯 t 检验。该序贯 t 检验可以根据不同的算法、不同的数据自适应地调节重复次数 m。该序贯 t 检验具有较小的一类错误及较优的势函数，可有效地减少假阳性算法比较结论的发生。在通常的 IID 数据的相关实验上，本文证明了序贯 t 检验的优良性以及“序贯”方式的必要性。针对准确率、召回率和 F_1 值，本文基于正则化 $m \times 2$ 交叉验证给出了它们的精准后验分布，并推导出合理的置信区间。本文进一步给出了一种贝叶斯检验来推断两个算法在准确率、召回率和 F_1 值上的差异。在文本数据的分词及命名实体识别任务的实验结果证实了该贝叶斯检验的有效性。

总体来说，针对算法比较任务，基于正则化交叉验证的统计推断理论框架已初步形成。然而，还有一些问题需要进行深入分析，今后的工作计划将从如下几方面开展。

第一、研究通用高效的正则化交叉验证的构造问题。本文第2.2.5 节的讨论表明正则化 RLT 的一般构造仍是一个公开问题。另外，针对文本数据上的不同任务，如何针对训练集和验证集频次分布差异，构造有效的正则化条件，并将其融入正则化 $m \times 2$ 交叉验证的增量式构造算法中，仍需要进一步研究。

第二、进一步研究两个 hold-out 估计的协方差函数的理论性质。本文第2.4.1 节和第3.5.3节关于该协方差的模拟结果表明：该协方差函数为下凸函数，但在某些情形下，该协方差函数并不具有单调性。在这些情形下，随着训练集间重叠样例个数的增加，该协方差函数先下降后上升。这表明训练集间的少许重叠可能会带来更为稳定的交叉验证估计。本文后续会对此情形进行深入研究。

第三、开拓正则化重抽样方法和正则化子抽样方法。着重研究 Bootstrap 方法的正则化技术，从重叠样本个数、样本出现的频次及单条样例被重复抽中的频次

三个维度，研究它们对泛化误差的 Bootstrap 估计的方差和偏差的影响，并建立 Bootstrap 抽样方法的优化方法，进而给出正则化重抽样方法的理论框架。对于大规模数据的分布式计算，以 Bag of Little Bootstrap[121] 为切入点，研究数据切分的正则化技术对分布式子抽样方法的影响，并进一步建立基于正则化子抽样的统计推断方法。

总之，面向有监督学习算法的比较任务，应设计合理的统计推断方法，以得到可靠的算法比较结论。在面向算法比较的统计推断中，应尽量排除人为有意或无意引入的随机噪声，确保推断过程更为有效。相比于现有的算法比较方法，正则化交叉验证的统计推断方法已在算法比较任务中展现出一定的优势。不过，以正则化交叉验证为切入点，揭示针对算法比较的统计推断中一些机理，并拓展出更为优良的算法比较方法，仍有很长的一段路要走。

参 考 文 献

[1] Thomas G Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms [J]. Neural computation. 1998, 10(7):1895–1923.

[2] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection [C]. International Joint Conference on Artificial Intelligence. 1995.

[3] Sylvain Arlot, Alain Celisse. A survey of cross-validation procedures for model selection [J]. Statistics Surveys. 2010, 4:40–79.

[4] 杨柳, 王钰. 泛化误差的各种交叉验证估计方法综述 [J]. 计算机应用研究. 2015, (5):1287–1290.

[5] Jerome Friedman, Trevor Hastie, Robert Tibshirani. The elements of statistical learning [M], vol. 1. Springer series in statistics New York, NY, USA:, 2001.

[6] Ethem Alpaydin. Combined 5×2 CV F-test for comparing supervised classification learning algorithms [J]. Neural computation. 1999, 11(8):1885–1892.

[7] Olcay Taner Yildiz. Omnivariate rule induction using a novel pairwise statistical test [J]. Knowledge and Data Engineering, IEEE Transactions on. 2013, 25(9):2105–2118.

[8] Yu Wang, Ruibo Wang, Huichen Jia, Jihong Li. Blocked 3×2 cross-validated t-Test for comparing supervised classification learning algorithms [J]. Neural Computation. 2014, 26(1):208–235.

[9] Yu Wang, Jihong Li, Yanfang Li. Choosing between two classification learning algorithms based on calibrated balanced 5×2 cross-validated F-test [J]. Neural Processing Letters. Aug 2017, 46:1–13.

[10] Steven L Salzberg. On comparing classifiers: Pitfalls to avoid and a recommended approach [J]. Data mining and knowledge discovery. 1997, 1(3):317–328.

[11] Troy Raeder, T Ryan Hoens, Nitesh V Chawla. Consequences of variability in classifier performance estimates [C]. 2010 IEEE International Conference on Data Mining. IEEE, 2010, 421–430.

[12] John PA Ioannidis. Why most published research findings are false [J]. PLoS medicine. 2005, 2(8):e124.

[13] Roger D Peng. Reproducible research in computational science [J]. Science. 2011, 334(6060):1226–7.

[14] J. T. Leek, R. D. Peng. Opinion: Reproducible research can still be wrong: Adopting a prevention approach [J]. Proc Natl Acad Sci U S A. 2015, 112(6):1645–1646.

[15] Jeffrey T Leek, Leah R Jager. Is most published research really false [J]. bioRxiv. 2017, 4(1):109–122.

[16] Claude Nadeau, Yoshua Bengio. Inference for the generalization error [J]. Machine Learning. 2003, 52(3):239–281.

[17] 李济洪, 王瑞波, 王蔚林, 李国臣. 汉语框架语义角色的自动标注 [J]. 软件学报. 2010, 21(4):597–611.

[18] 宋毅君, 王瑞波, 史立校. 中文分词任务中标注集合的选择方法 [J]. 山西大学学报（自然科学版）. 2016, 39(2):204–209.

[19] Cyril Goutte, Eric Gaussier. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation [C]. European Conference on Information Retrieval. Springer, 2005, 345–359.

[20] Marianthi Markatou, Hong Tian, Shameek Biswas, George Hripcsak. Analysis of variance of cross-validation estimators of the generalization error [J]. Journal of Machine Learning Research. 2005, 6:1127–1168.

[21] Philip J. McCarthy. The use of balanced half-sample replication in cross-validation Studies [J]. Journal of the American Statistical Association. 1976, 71(355):596–604.

[22] Prabir Burman. A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods [J]. Biometrika. 1989, 76(3):503–514.

[23] Yoshua Bengio, Yves Grandvalet. No unbiased estimator of the variance of K-fold cross-validation [J]. J Mach Learn Res. 2004, 5:1089–1105. 1.

[24] Juan D Rodríguez, Aritz Perez, Jose A Lozano. Sensitivity analysis of k-fold cross validation in prediction error estimation [J]. IEEE transactions on pattern analysis and machine intelligence. 2010, 32(3):569–575.

[25] Juan D Rodríguez, Aritz Pérez, Jose A Lozano. A general framework for the statistical analysis of the sources of variance for classification error estimators [J]. Pattern Recognition. 2013, 46(3):855–864.

[26] Georgios Afendras, Marianthi Markatou. Optimality of training/test size and resampling effectiveness of cross-validation estimators of the generalization error [J]. arXiv preprint arXiv:151102980. 2015.

[27] Jan Larsen, Cyril Goutte. On optimal data split for generalization estimation and model selection [C]. Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (Cat. No. 98TH8468). IEEE, 1999, 225–234.

[28] Damjan Krstajic, Ljubomir J Buturovic, David E Leahy, Simon Thomas. Cross-validation pitfalls when selecting and assessing regression and classification models [J]. Journal of cheminformatics. 2014, 6(1):10.

[29] Leo Breiman, Philip Spector. Submodel selection and evaluation in regression. The X-random case [J]. International statistical review/revue internationale de Statistique. 1992:291–319.

[30] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, Neil D Lawrence. Dataset shift in machine learning [M]. The MIT Press, 2009.

[31] Masashi Sugiyama, Matthias Krauledat, Klaus-Robert MÃžller. Covariate shift adaptation by importance weighted cross validation [J]. Journal of Machine Learning Research. 2007, 8(May):985–1005.

[32] Jose G Moreno-Torres, José A Sáez, Francisco Herrera. Study on the impact of partition-induced dataset shift on k-fold cross-validation [J]. Neural Networks and Learning Systems, IEEE Transactions on. 2012, 23(8):1304–1312.

[33] NA Diamantidis, D Karlis, Emmanouel A Giakoumakis. Unsupervised stratification of cross-validation for accuracy estimation [J]. Artificial Intelligence. 2000, 116(1):1–16.

[34] Victoria López, Alberto Fernández, Francisco Herrera. On the importance of the validation technique for classification with imbalanced datasets: Addressing covariate shift when data is skewed [J]. Information Sciences. 2014, 257:1–13.

[35] Leo Breiman. Classification and regression trees [M]. Routledge, 2017.

[36] Ji-Hyun Kim. Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap [J]. Computational statistics & data analysis. 2009, 53(11):3735–3745.

[37] Georgios Afendras, Marianthi Markatou. Optimality of training/test size and resampling effectiveness in cross-validation [J]. Journal of Statistical Planning and Inference. 2019, 199:286–301.

[38] CF Jeff Wu, Michael S Hamada. Experiments: planning, analysis, and optimization [M], vol. 552. John Wiley & Sons, 2011.

[39] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages [J]. Psychometrika. 1947, 12(2):153–157.

[40] Mikaela Keller, Samy Bengio, Siew Y Wong. Benchmarking non-parametric statistical tests [C]. Advances in neural information processing systems. 2006, 651–658.

[41] Remco R Bouckaert. Estimating replicability of classifier learning experiments [C]. Proceedings of the twenty-first international conference on Machine learning. ACM, 2004, 15.

[42] Remco R Bouckaert, Eibe Frank. Evaluating the replicability of significance tests for comparing learning algorithms [M], Springer, 2004. 3–12.

[43] Yves Grandvalet, Yoshua Bengio. Hypothesis testing for cross-validation [J]. Montreal Universite de Montreal, Operationnelle DdIeR. 2006, 1285.

[44] Anders Isaksson, Mikael Wallman, Hanna Göransson, Mats G Gustafsson. Cross-validation and bootstrapping are unreliable in small sample classification [J]. Pattern Recognition Letters. 2008, 29(14):1960–1965.

[45] Hui Shen, William J Welch, Jacqueline M Hughes-Oliver. Efficient, adaptive cross-validation for tuning and comparing models, with application to drug discovery [J]. The Annals of Applied Statistics. 2011, 5(4):2668–2687.

[46] Qing Wang, Bruce Lindsay. Variance estimation of a general u-statistic with application to cross-validation [J]. Statistica Sinica. 2014:1117–1141.

[47] Rotem Dror, Gili Baumer, Segev Shlomov, Roi Reichart. The hitchhiker' s guide to testing statistical significance in natural language processing [C]. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, 2018, 1383–1392.

[48] Alexander S. Yeh. More accurate tests for the statistical significance of result differences [C]. COLING 2000, 18th International Conference on Computational Linguistics, Proceedings of the Conference, 2 Volumes, July 31 - August 4, 2000, Universität des Saarlandes, Saarbrücken, Germany. 2000, 947–953.

[49] L. Gillick, S. J. Cox. Some statistical issues in the comparison of speech recognition algorithms [C]. International Conference on Acoustics. 1989.

[50] Alexander Yeh. More accurate tests for the statistical significance of result differences [J]. 2000:947–953.

[51] Walter Daelemans, Véronique Hoste. Evaluation of machine learning methods for natural language processing tasks [C]. Proceedings of the Third International Conference on Language Resources and Evaluation, LREC 2002, May 29-31, 2002, Las Palmas, Canary Islands, Spain. 2002.

[52] Philipp Koehn. Statistical significance tests for machine translation evaluation [C]. Proceedings of the 2004 conference on empirical methods in natural language processing. 2004.

[53] Stefan Riezler, John T Maxwell. On some pitfalls in automatic evaluation and significance testing for MT [C]. Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. 2005, 57–64.

[54] Taylor Berg-Kirkpatrick, David Burkett, Dan Klein. An empirical investigation of statistical significance in nlp [C]. Proceedings of the 2012 Joint Conference

on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2012, 995–1005.

[55] Anders Søgaard. Estimating effect size across datasets [C]. Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2013, 607–611.

[56] Anders Søgaard, Anders Johannsen, Barbara Plank, Dirk Hovy, Héctor Martínez Alonso. What's in a p-value in NLP? [C]. Proceedings of the eighteenth conference on computational natural language learning. 2014, 1–10.

[57] Aurélie Névéol, Kevin Cohen, Cyril Grouin, Aude Robert. Replicability of research in biomedical natural language processing: a pilot evaluation for a coding task [C]. Proceedings of the Seventh International Workshop on Health Text Mining and Information Analysis. 2016, 78–84.

[58] Rotem Dror, Gili Baumer, Marina Bogomolov, Roi Reichart. Replicability analysis for natural language processing: Testing significance with multiple datasets [J]. Transactions of the Association for Computational Linguistics. 2017, 5:471–486.

[59] Andrew K. Halberstadt. Heterogeneous acoustic measurements and multiple classifiers for speech recognition [D]. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1999.

[60] 王瑞波. 基于条件随机场模型的汉语框架语义角色自动标注研究 [D]. 硕士论文, 山西大学, 2009.

[61] 李国臣, 党帅兵, 王瑞波, 李济洪. 基于字的分布表征的汉语基本块识别 [J]. 中文信息学报. 2014, 28(6):18–25.

[62] 王瑞波, 李济洪, 李国臣, 杨耀文. 基于 Dropout 正则化的汉语框架语义角色识别 [J]. 中文信息学报. 2017, 31(1):147–154.

[63] Peng Zhang, Wanhua Su. Statistical inference on recall, precision and average precision under random selection [C]. Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on. IEEE, 2012, 1348–1352.

[64] Yu Wang, Jihong Li. Credible intervals for precision and recall based on a k-fold cross-validated beta distribution [J]. Neural Computation. 2016, 28(8):1694–1722.

[65] Dell Zhang, Jun Wang, Xiaoxue Zhao. Estimating the uncertainty of average F_1 scores [C]. Proceedings of the 2015 International Conference on The Theory of Information Retrieval. ACM, 2015, 317–320.

[66] Dell Zhang, Jun Wang, Xiaoxue Zhao, Xiaoling Wang. A bayesian hierarchical model for comparing average F_1 scores [C]. Data Mining (ICDM), 2015 IEEE International Conference on. IEEE, 2015, 589–598.

[67] Dell Zhang, Jun Wang, Emine Yilmaz, Xiaoling Wang, Yuxin Zhou. Bayesian performance comparison of text classifiers [C]. Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. ACM, 2016, 15–24.

[68] Yu Wang, Jihong Li, Yanfang Li, Ruibo Wang, Xingli Yang. Confidence interval for f_1 measure of algorithm performance based on blocked 3×2 cross-validation [J]. IEEE Transactions on Knowledge & Data Engineering. 2015, 27(3):651–659.

[69] Olivier Caelen. A Bayesian interpretation of the confusion matrix [J]. Annals of Mathematics and Artificial Intelligence. 2017, 81(3-4):429–450.

[70] 杨杏丽, 王钰, 王瑞波, 李济洪. 基于组块 3×2 交叉验证的预测误差估计的方差 [J]. 应用概率统计. 2014, 30(4):372–380.

[71] Alessio Benavoli, Giorgio Corani, Janez Demsar, Marco Zaffalon. Time for a change: a tutorial for comparing multiple classifiers through Bayesian analysis [J]. Journal of Machine Learning Research. 2016, 18.

[72] Geoffrey David Cumming. Replication and p Intervals: p values predict the future only vaguely, but confidence intervals do much better [J]. Perspectives on Psychological Science. 2008, 3(4):286–300.

[73] Nuzzo Regina. Scientific method: statistical errors [J]. Nature. 2014, 506(7487):150–152.

[74] Jean Gaudart, Laetitia Huiart, Paul J Milligan, Rodolphe Thiebaut, Roch Giorgi. Reproducibility issues in science, is P value really the only answer? [J]. Proceedings of the National Academy of Sciences. 2014, 111(19):E1934–E1934.

[75] Luc Devroye, T Wagner. Distribution-free performance bounds with the resubstitution error estimate (Corresp.) [J]. IEEE Transactions on Information Theory. 1979, 25(2):208–210.

[76] Avrim Blum, Adam Kalai, John Langford. Beating the hold-out: Bounds for k-fold and progressive cross-validation [C]. COLT. 1999, vol. 99, 203–208.

[77] B Hafidi, A Mkhadri. Repeated half sampling criterion for model selection [J]. Sankhyā: The Indian Journal of Statistics. 2004:566–581.

[78] Guy P Nason. Wavelet shrinkage using cross-validation [J]. Journal of the Royal Statistical Society Series B (Methodological). 1996:463–479.

[79] Donald E Knuth. The art of computer programming, volume 2 (3rd ed.): seminumerical algorithms [J]. American Mathematical Monthly. 1998, 77(8):900.

[80] Jiannan Lu. Covariate adjustment in randomization-based causal inference for 2K factorial designs [J]. Statistics & Probability Letters. 2016, 119:11–20.

[81] Fasheng Sun, Boxin Tang. A method of constructing space-filling orthogonal designs [J]. Journal of the American Statistical Association. 2017, 112(518):683–689.

[82] Boxin Tang. Orthogonal array-based latin hypercubes [J]. Journal of the American Statistical Association. 1993, 88(424):1392–1397.

[83] Xiangshun Kong, Mingyao Ai, Kwok Leung Tsui. Design for sequential follow-up experiments in computer emulations [J]. Technometrics. 2018, 60(1):61–69.

[84] Michael Stein. Large sample properties of simulations using latin hypercube sampling [J]. Technometrics. 1987, 29(2):143–151.

[85] Jihong Li, Ruibo Wang, Weilin Wang, Bo Gu, Guochen Li. Automatic labeling of semantic role on Chinese FrameNet using conditional random fields [C]. Ieee/wic/acm International Joint Conference on Web Intelligence and Intelligent Agent Technology. 2009, 259–262.

[86] Jiannan Lu. On randomization-based and regression-based inferences for 2^k factorial designs [J]. Statistics & Probability Letters. 2016, 112:72–78.

[87] Alexander Barvinok. On the number of matrices and a random matrix with prescribed row and column sums and 0–1 entries [J]. Advances in Mathematics. 2010, 1(224):316–339.

[88] Richard A Brualdi. Matrices of zeros and ones with fixed row and column sum vectors [J]. Linear algebra and its applications. 1980, 33:159–231.

[89] Jianqing Fan, Shaojun Guo, Ning Hao. Variance estimation using refitted cross-validation in ultrahigh dimensional regression [J]. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2012, 74(1):37–65.

[90] Zhao Chen, Jianqing Fan, Runze Li. Error variance estimation in ultrahigh-dimensional additive models [J]. Journal of the American Statistical Association. 2018, 113(521):315–327.

[91] Predrag Stanišić, Savo Tomović. Frequent itemset mining using two-fold cross-validation model [C]. 2012 Mediterranean Conference on Embedded Computing (MECO). IEEE, 2012, 229–232.

[92] Jacques Wainer, Gavin Cawley. Empirical evaluation of resampling procedures for optimising SVM hyperparameters [J]. The Journal of Machine Learning Research. 2017, 18(1):475–509.

[93] Wang Yu, Jihong Li, Yanfang Li. Measure for data partitioning in $m \times 2$ cross-validation [J]. Pattern Recognition Letters. 2015, 65:211–217.

[94] Ryan J. Tibshirani, Robert Tibshirani. A bias correction for the minimum error rate in cross-validation [J]. Annals of Applied Statistics. 2009, 3(2):822–829.

[95] Abraham Wald. Sequential tests of statistical hypotheses [J]. Annals of Mathematical Statistics. 1945, 16(2):117–186.

[96] George Casella, Roger L Berger. Statistical inference [M], vol. 2. Duxbury Pacific Grove, CA, 2002.

[97] Taku Kudo, Yuji Matsumoto. Chunking with support vector machines [C]. Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies. Association for Computational Linguistics, 2001, 1–8.

[98] Hong Shen, Anoop Sarkar. Voting between multiple data representations for text chunking [C]. Conference of the Canadian Society for Computational Studies of Intelligence. Springer, 2005, 389–400.

[99] John D. Lafferty, Andrew McCallum, Fernando C. N. Pereira. Conditional random fields: probabilistic models for segmenting and labeling sequence data [C]. Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001. 2001, 282–289.

[100] Sepp Hochreiter, Jürgen Schmidhuber. Long short-term memory [J]. Neural computation. 1997, 9(8):1735–1780.

[101] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, Chris Dyer. Neural architectures for named entity recognition [C]. Proceedings of NAACL-HLT. 2016, 260–270.

[102] Hai Zhao, Chang-Ning Huang, Mu Li. An improved Chinese word segmentation system with conditional random field [C]. Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing. 2006, 162–165.

[103] Murray Wood, Marc Roper, Andrew Brooks, James Miller. Comparing and combining software defect detection techniques: a replicated empirical study. Software Engineering—ESEC/FSE'97, Springer, 1997. 262–277.

[104] Sheng Yu, Shijie Zhou. A survey on metric of software complexity [C]. 2010 2nd IEEE International Conference on Information Management and Engineering. IEEE, 2010, 352–356.

[105] Jitender Kumar Chhabra, Varun Gupta. A survey of dynamic software metrics [J]. Journal of computer science and technology. 2010, 25(5):1016–1029.

[106] Chakkrit Tantithamthavorn, Shane McIntosh, Ahmed E Hassan, Kenichi Matsumoto. An empirical comparison of model validation techniques for defect prediction models [J]. IEEE Transactions on Software Engineering. 2017, 43(1):1–18.

[107] Kehan Gao, Taghi M. Khoshgoftaar. A comprehensive empirical study of count models for software fault prediction [J]. IEEE Transactions on Reliability. 2007, 56(2):223–236.

[108] Thomas J Ostrand, Elaine J Weyuker, Robert M Bell. Predicting the location and number of faults in large software systems [J]. IEEE Transactions on Software Engineering. 2005, 31(4):340–355.

[109] R. E. Mullen, S. S. Gokhale. Software defect rediscoveries: a discrete lognormal model [C]. 16th IEEE International Symposium on Software Reliability Engineering (ISSRE'05). 10 pp.–212.

[110] Stefan Lessmann, Bart Baesens, Christophe Mues, Swantje Pietsch. Benchmarking classification models for software defect prediction: A proposed framework and novel findings [J]. IEEE Transactions on Software Engineering. 2008, 34(4):485–496.

[111] Chakkrit Tantithamthavorn, Shane McIntosh, Ahmed E Hassan, Kenichi Matsumoto. Automated parameter optimization of classification techniques for defect prediction models [C]. 2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE). IEEE, 2016, 321–332.

[112] Kehan Gao, Taghi M Khoshgoftaar, Huanjing Wang, Naeem Seliya. Choosing software metrics for defect prediction: an investigation on feature selection techniques [J]. Software: Practice and Experience. 2011, 41(5):579–606.

[113] A Gunes Koru, Hongfang Liu. An investigation of the effect of module size on defect prediction using static measures [J]. Model driven engineering languages and systems. 2005, 30(4):1–5.

[114] Raimund Moser, Witold Pedrycz, Giancarlo Succi. A comparative analysis of the efficiency of change metrics and static code attributes for defect prediction [C]. Proceedings of the 30th international conference on software engineering. ACM, 2008, 181–190.

[115] Martin Shepperd, David Bowes, Tracy Hall. Researcher bias: The use of machine learning in software defect prediction [J]. IEEE Transactions on Software Engineering. 2014, 40(6):603–616.

[116] Chakkrit Tantithamthavorn, Shane Mcintosh, Ahmed Hassan, Kenichi Matsumoto. An empirical comparison of model validation techniques for defect prediction models [J]. IEEE Transactions on Software Engineering. 2017, 43(1):1–18.

[117] Xiao-Yuan Jing, Fei Wu, Xiwei Dong, Baowen Xu. An improved SDA based defect prediction framework for both within-project and cross-project class-imbalance problems [J]. IEEE Transactions on Software Engineering. 2017, 43(4):321–339.

[118] Lina Gong, Shujuan Jiang, Lili Bo, Li Jiang, Junyan Qian. A novel class-imbalance learning approach for both within-project and cross-project defect prediction [J]. IEEE Transactions on Reliability. 2019.

[119] Xiaoxing Yang, Ke Tang, Xin Yao. A learning-to-rank approach to software defect prediction [J]. IEEE Transactions on Reliability. 2015, 64(1):234–246.

[120] 崔军, 刘亚娜, 郭新峰, 王瑞波, 李济洪. 基于最大信息系数的软件缺陷预测模型 [J]. 应用概率统计. 2019, 35(5):（正在刊出）.

[121] Ariel Kleiner, Ameet Talwalkar, Purnamrita Sarkar, Michael I Jordan. A scalable bootstrap for massive data [J]. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2014, 76(4):795–816.

攻读博士学位期间取得的研究成果

发表论文:

(1) **Ruibo Wang**, Jihong Li(*). Bayes test of precision, recall, and F_1 measure for comparison of two natural language processing models [C]. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (Long Paper), Florence Italy, 2019 （Accepted, CCF A）

(2) **Ruibo Wang**, Jihong Li(*), Xingli Yang, Jing Yang. Block-regularized repeated learning-testing for estimating generalization error [J]. Information Sciences. 2019, 477:246-264. （CCF B, SCI）

(3) **Ruibo Wang**, Yu Wang, Jihong Li(*), Xingli Yang, Jing Yang. “Block-regularized $m \times 2$ cross-validated estimator of the generalization error [J]. Neural Computation. 2017, 29(2): 519-554. （CCF B, SCI）

(4) **王瑞波**, 王钰, 李济洪 (*). 面向文本数据的正则化交叉验证方法 [J]. 中文信息学报. 2019, 33(5): 55-66.

(5) **王瑞波**, 李济洪 (*), 李国臣, 杨耀文. 基于 Dropout 正则化的汉语框架语义角色识别 [J]. 中文信息学报. 2017, 31(1): 147-154.

基金项目:

(1) 国家自然科学基金青年科学基金项目, 61806115, 面向文本数据模型比较的正则化交叉验证方法, 2019 年 1 月 -2021 年 12 月, 23 万元, 在研, 主持

(2) 国家社会科学基金面上项目, 16BTJ034, 文本数据挖掘分类算法性能对照的序贯检验方法研究, 2016 年 7 月 -2019 年 7 月, 20 万元, 在研, 参加

(3) 国家自然科学基金青年科学基金项目, 61503228, 方差正则化的分类模型选择方法研究, 2016 年 1 月 -2018 年 12 月, 18 万元, 已结题, 参加

专利:

(1) 国家发明专利, ZL201610471449.5, 一种机器学习系统学习性能的评估方法, 2018 年 7 月 24 日（授权公告日）, 第一发明人

软件著作:

(1) 计算机软件著作权登记证书, 2018SR615306, 中文文本语料及知识库管理系统 (V1.0), 2018 年 3 月 7 日, 山西大学, 第二完成人

致　谢

值此论文完成之际，谨向给予我无私帮助的老师和同学们致以诚挚的谢意！

首先感谢我的导师李济洪教授，我的每一步成长跟他的教诲是分不开的。我在博士五年的学习和研究中倾注了李老师极大的心血。李老师不仅是我的授业恩师，更是我的人生导师。他为我提供了优越的实验环境，营造了轻松和谐的学习氛围，给我创造了很多学术交流的机会，开阔了我的眼界。李老师渊博的专业知识、敏锐的学术洞察力、严谨的治学态度、忘我的工作精神、对事业孜孜不倦的追求和创新精神，给我留下了难忘的记忆，并激励我终生奋发进取。

感谢李国臣老师和崔军老师。我的成长，离不开他们的百般呵护。希望我的努力，没有辜负他们对我的一片期望。

五年来，计算机与信息学院的各位老师和同学在学习、生活上给予了我很多指导和帮助，使我顺利地完成了博士研究生阶段的学习。非常感谢梁吉业老师，李德玉老师，李茹老师、王素格老师、王文剑老师、钱宇华老师和曹付元老师等对我的关怀、指导和帮助。感谢廖健班长、王杰副班长、李明涛博士、成红红博士、许行博士、张超博士、李飞江博士、李旸博士、王学军博士等的帮助。

感谢现代教育技术学院所有领导、老师和同事对我的关怀和帮助。特别感谢宋毅君副院长和王炜老师给予了我大量的帮助。他们为我解决了很多工作和生活上的困难，让我可以拥有大块时间用来思考。非常感谢我的好伙伴——山西大学高性能计算平台——夜以继日的工作。没有他的帮助，我将无法完成博士期间的研究工作。

感谢实验室一起同甘共苦的所有老师和同学。感谢王钰师兄、曹学飞博士、郭新峰老师、谷波老师、牛倩老师、杨静师妹、赵存秀师妹、路佳佳师妹、刘亚娜学妹、房立超师妹、侯云瑶师妹、丁浩杰师弟的陪伴，和他们的每周十二小时的讨论，激发了我对统计机器学习、自然语言处理和程序理解等领域相关研究问题的大量思考。特别感谢杨杏丽师妹，她帮助我解决了大量的数学推导。

感谢我的家人。他们是我此生的挚爱。感谢父母亲在我三十多年的生活中给予的无尽的关怀和支持。感谢岳父岳母对我多年的关爱和帮助。感谢我的妻子，任璐璐女士，在生活上给予我无微不至的照顾。感谢你，我的女儿，王辰鑫。你给予我大量的快乐和无穷的幸福，也让我更懂得责任和担当。希望你能健康快乐成长！

感谢参加我论文答辩的各位专家和教授，祝您们工作顺利、身体健康！

未来将何去何从？初心不改、砥砺前行！

个人简况及联系方式

姓　　名：王瑞波

性　　别：男

籍　　贯：山西 潞城

主要经历：

- 2014.09—2019.07　山西大学　计算机与信息技术学院　　软件工程　　博士
- 2006.09—2009.07　山西大学　计算机与信息技术学院　　计算机软件与理论　硕士
- 2002.09—2006.07　山西师范大学　数学与计算机科学学院　信息与计算科学　　学士

联系方式：wangruibo@sxu.edu.cn

承　诺　书

本人郑重声明：所呈交的学位论文，是在导师指导下独立完成的，学位论文的知识产权属于山西大学。如果今后以其他单位名义发表与在读期间学位论文相关的内容，将承担法律责任。除文中已经注明引用的文献资料外，本学位论文不包括任何其他个人或集体已经发表或撰写过的成果。

作者签名：

2019 年 6 月 1 日

学位论文使用授权声明

本人完全了解山西大学有关保留、使用学位论文的规定，即：学校有权保留并向国家有关机关或机构送交论文的复印件和电子文档，允许论文被查阅和借阅，可以采用影印、缩印或扫描等手段保存、汇编学位论文。同意山西大学可以用不同方式在不同媒体上发表、传播论文的全部或部分内容。

保密的学位论文在解密后遵守此协议。

作者签名：

导师签名：

2019 年 6 月 1 日