多视图学习 reviewed by Xu Chang

Author: shushen Date: 2020-9-13

摘要部分（**abstract**）

1. 多视图学习方法分类

> *协同训练: 交替训练，以最大程度地在两个不同的数据视图上达成共识*

> *多核学习: 利用自然地对应于不同视图的内核，并线性或非线性地组合内核以提高学习性能*

> *子空间学习: 通过假设输入视图是从该潜在子空间生成的，来获得多个视图共享的潜在子空间*

> *共同特点：要利用共识原则或补充原则来确保多视图学习的成功*

2. 多视图研究的方向和意义

> *如何利用多视图？如何构建多视图？如何评估多视图的价值？*

第一章 引言部分

1. 多视图提出的原因

> *单视图学习方法下，将不同方面或者视角下的数据特征合并在一个视图中来适应学习器，但是*
>
> *这种方式在样本数不够的前提下，容易造成过拟合（特征维度过高，样本数太少），于是提出*
>
> *多视图来学习单一视图特有的统计特性并利用视图之间的冗余来提高学习性能；*

2. 协同训练的发展和重要假设

> *经典的协同训练算法：在两个视图上交替训练以使两个视图尽可能地在未标记的数据上达到共识*
>
> *算法流程：*
>
> *输入：标记数据集L，未标记数据集U*
>
> - *用$L1$训练视图$X1$上的分类器$f1$，用$L2$训练视图$X2$上的分类器$f2$；*
> - *用$f1$和$f2$分别对未标记数据U进行分类；*

> - *把f1对U的分类结果中，前k个最置信的数据（正例 p个反例n个）及其分类结果加入L2；把f2对U的分类结果中，前k个最置信的数据及其分类结果加入L1；把这2（p+n）个数据从U中移除；*
> - *重复上述过程，直到U为空集。*
>
> *输出：分类器f1和f2。*

> *变体1：给未标记数据分配标签时使用可变概率的标签*
>
> *变体2： 主动学习与协同训练相结合，即人工参与*
>
> *变体3：视协同训练为组合标签在不同视图上的传播，再引入“分歧”，注：不是很理解*
>
> *变体4 or 贡献：提出了共正则化规范，扩展了正则化使用的范围*

> *三个重要假设：*
>
> 1. *sufficiency - each view is sufficient for classification on its own（每个视图的特征都足够完成自己的分类任务）*
> 2. *compatibility- the target function of both views predict the same labels for co-occurring features with a high probability（两个视图的目标函数可以根据共同出现的特征在很大的可能性下预测出同一个标签）*
> 3. *conditional independence- views are conditionally independent given the label（在给定标签时，各个视图是条件独立的），注：不是很理解*

3. 多核学习

> *多核学习中的多个内核自然地对应于不同的视图，并且线性或非线性地组合内核可以提高学习性能。*
>
> *多核学习的发展状况没有清晰的脉络；*

4. 子空间学习

> *先假设所有视图来自于一个共同的子空间，再根据相关性分析算法（CCA）算法从多个视图中得出各个视图的基向量，从而得到共享的潜在子空间；*
>
> *当前研究方向和进展：*
>
> 1. *不同视图之间的联系度量方式，例如欧式距离和马尔可夫网路，以及视图之间的互相预测*
> 2. *子空间学习用于聚类和回归*
> 3. *不止步于CCA，还可以在引入标签后进行判别分析*
> 4. *共享子空间和私有子空间对于学习性能的提高*

5. 结论

> *关于多视图学习的研究常常与机器学习领域的其他问题相关，比如主动学习、域自适应、Adaboost等；*

第二章 多视图学习的基本原理

> *多视图学习相比于单视图的关键是识别冗余视图，多视图丰富了数据信息，但是如果不能够适当地*
>
> *处理冗余信息，则可能会降低学习性能。于是多视图的两个关键原则是：共识和互补。*

2.1 共识原则

> *如下不等式已经被证明：*
>
> $$P(f^1 \neq f^2) \geq \max\{P_{err}(f^1), P_{err}(f^2)\}.$$
>
> *由不等式可知，两个独立假设不一致的概率为任意一个假设的错误率的上限, 可以通过最小化两个假设的不一致率来最小化单个假设的不一致率；*
>
> *许多算法都利用了这一原则，比如协同训练算法是通过最小化带标记样本的误差和最大化无标记样本与带标记样本的一致性来实现最终的分类学习；而“共（协同）正则化”概念也是体现了这一原则，公式如下：*
>
> $$\min \quad \sum_{i \in U}[f^1(x_i) - f^2(x_i)]^2 + \sum_{i \in L} V(y_i, f(x_i)),$$
>
> *其中的第一项描述了两个视图的不一致损失，第二项描述了预测结果在损失函数下的经验损失；*
>
> *此外还有许多研究工作体现了这一原则；*

2.2 互补原则

> *互补原则指出，在多视图设置中，数据的每个视图可能包含一些其他视图不具备的知识;因此，可以使用多个视图来全面、准确地描述数据。在涉及多视图数据的机器学习问题中，可以利用多视图下的互补信息，利用互补原理来提高学习性能。*
>
> *比如使用在某一视图上学习的分类器对未标记的数据进行标记，然后将这些新标记的数据加入另一个视图中进行下一次迭代的分类器训练，第二个视图共享了第一个视图的互补信息，这种共享也可以很容易实现相互。*
>
> *比如对同一个学习器使用不同的配置可以看作是两个不同的视图，当两个学习器的差异率大于总的错误率时（不一致损失大于经验损失），这种协同学习可以改进学习器的性能（其原理与2.1 节中的公式有关）。第一个学习器所标记出的样本如果对于第二个学习器有用，那么说明视图1 包含了视图2 所缺少的信息，这是一种互补。两个学习器将不断地交换互补信息来提高性能。*

通过诸多场景继续阐述了多视图学习以及两个原则在多视图学习中的体现。

第三章 视图生成

> *多视图学习的重点是获取冗余视图，这也是与单视图学习的主要区别。多视图生成的目的不仅是获取不同属性的视图，还涉及到如何保证视图充分表示数据，满足学习假设的问题。*
>
> *接下来讨论了如何生成有效视图*

3.1 视图构造

> *视图构造的必要性：在无法轻松获取多视图数据的时候，无法直接进行多视图学习，所以第一步需要从单一视图数据来构造多视图数据。*

> *不同的视图构造对应于不同特征集的划分，即传统的机器学习中特征选择的任务，但区别是传统的机器学习中的特征选择是选择出一个单一的特征集合，而此处的特征集划分目的是划分出多个不相交的子集，其中最传统的方法是将单一视图对应的特征集合随机划分为不同的子集。但是这一项工作很复杂，**其划分效果依赖于分类器的选择和数据域的特点**。*

3.1.1 多视图构造的部分方法

> *Bootstrapping算法: 对总体样本进行多次重复采样（有放回）来抽取足够代表母体的新样本，然后对新样本计算母体样本的近似分布；*
>
> *随机子空间算法（RSM）：融合了Bootstrapping算法的优点，但是其随机过程是在特征空间中进行，每一次随机过程都在所有的特征空间中选择少量的维数（如果是2），对于n维特征空间将会有2^n个不同的视图；采用SVM将多个采样子空间组合在一起以构造强大的学习器是一个可选的方法（此处相似于多分类SVM的基本思想，构造多个分类器）。*

> *example1: 高光谱图像数据的视图生成 文章地址*
>
> *example2: 使用不同数量的单词组成术语以构造文本文档的不同视图（表示）文章地址*
>
> *example3: 新的特征分解算法（PMC， Pseudo Multi-view Co-training），将特征空间自动划分为两个互斥子集，优化器可以描述为：*
>
> $$\min_{\mathbf{w}_1,\mathbf{w}_2} \quad \log(e^{\mathcal{L}(\mathbf{w}_1;L)} + e^{\mathcal{L}(\mathbf{w}_2;L)}),$$
>
> *然后通过如下约束来划分不同的视图：*
>
> $$\forall i, 1 \le i \le d, \quad \mathbf{w}_1^i \mathbf{w}_2^i = 0.$$
>
> *对于维度i，两个分类器中至少有一个将其置为0；每一次迭代中优化器都需要去寻找一个最优的特征空间分割；*

> *核函数：对于非线性可分的问题，常常采用低维特征空间映射到线性可分高维特征空间，然后在高维特征空间中实现分类和回归等任务，但是高维空间中的维度灾难导致了向量之间的计算很复杂，甚至不可计算，而一般我们完成任务时只需要高维空间中两个向量之间的内积（标量），于是假设一个函数，其输入样本中任意两个低维特征空间向量后的输出与两个向量映射到高维空间中做内积的结果相等，那么这个函数即为核函数。*
>
> *当有了这个核函数之后，我们不再需要显示的低维空间到高维空间的映射，核函数隐式地完成了这一任务，所以核函数设计的好坏，直接关系到空间映射的合理性以及可能性， 直接影响完成任务的效果。*

3.1.2 多视图构造方法总结

> *方法一：直接通过随即方法从源数据构造多个视图*
>
> *方法二：分割原始单视图得到多视图划分，例如使用不同核函数*
>
> *方法三：对特征空间进行子集的划分得到特征空间的多视图*

3.2 多视图的评估

如何分析和评价生成的多视图，也是视图生成的任务之一：分析多视图之间的关系，分析由于违反多视图假设或者视图噪声带来的问题。

3.2.1 违反多视图假设或者视图噪声带来的问题

> 1. *view sufficiency在实践中通常不成立，比如视频帧检测中低维的直方图特征无法区分一架飞机和一只鹰；*
>
>   *现有的解决办法：先使用带标记数据初始化两个分类器，再使用初始化分类器标记未标记的数据，再通过这一部分刚标记的数据训练两个额外的分类器，之后将四个分类器进行加权组合（在验证集上测试）来检验从刚刚标记的数据上分类器可以获得多少好处，如果刚刚得到的两个新的分类器噪声过大而无法使用，则回到第一步，相关文献*

> 2. *视图分歧（视图噪声）：由于噪声的存在，每个样本在同一个视图下，并不是一直属于同一个class，特别是当样本在前景和背景中混淆时；*
>
>   *通过定义一个条件视图熵来衡量样本在视图中的不一致性，其变量之一是视图，变量之二是条件作用的发生，变量之三是样本；比如当条件作用发生在背景这种容易产生噪声的环境中时，熵增大；*
>
>   *在协同训练中通过给熵设计阈值，来舍弃在给定视图中噪声太大（熵太大）的样本，以提高学习性能；*

> 3. *多核噪声：在多内核学习中，不同的内核可能使用来自不同表示的输入，可能来自一系列的模式或来源。这些表示可能具有与不同内核对应的不同的相似性度量，可以看作是数据的不同视图。在这种情况下，组合内核是组合多个信息源的一种可能方式;但是在现实生活中，信号源可能会被不同的噪声所破坏，所以当一些核是有噪声或不相关时，在学习过程中需要优化核权值。*
>
>   *对多核进行加权已经被证明可以较为有效地抑制简单噪声，但是无法处理异方差噪声、数据缺失等问题。*

> *其他解决办法啊：每个视图上学习局部加权相关论文*

3.2.2 两种置信度的提出

> *视图间置信度和视图内置信度的提出：其对应于视图充分性和视图间依赖性的问题*
>
> *考虑到样本X与M个视图相关，观察数据分别表示为X1，…，XM;根据互信息定义，X的视图间置信度（视图依赖性）定义为*
>
> $$C_{inter}(X)=\sum_{i=1}^{M}\sum_{j=i}^{M}\frac{1}{I(X^i,X^j)},$$
>
> *其中I 描述了两个视图（ i 和j ）间交互信息的多少；交互信息越少，视图间的置信度越高；*
>
> *X的视图内置信度（视图充分性）定义为：*
>
> $$C_{intra}(X)=\sum_{i=1}^{M}\frac{1}{F(X_i^L,X_i^U,S_i)},$$
>
> *F 衡量样本X在L（带标签数据）和U（未标记数据）中的观察的一致性，S为相似矩阵，遍历所有的视图，计算总的不一致性，其值越高，视图内置信度越低；*

3.2.3 典型相关分析的应用

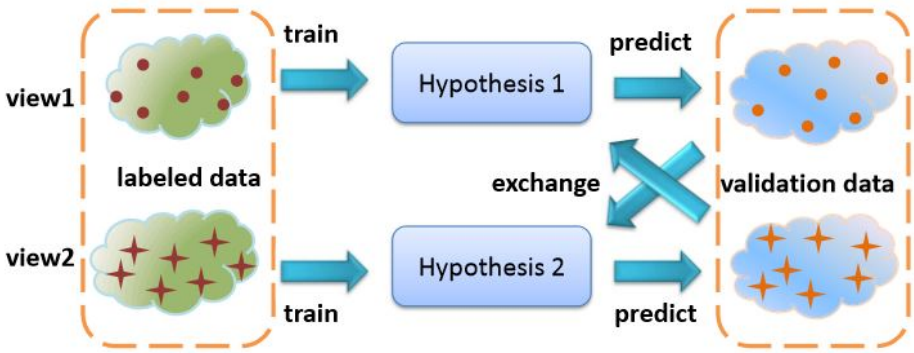
> *典型相关分析（CCA）：视图之间的相关性是基于子空间的多视图学习方法的一个重要因素，典型相关分析(CCA)来描述两个视图之间的线性关系，目的是计算两个不同视图的低维数共享嵌入变量，使两个视图之间的相关性在低维空间中最大化，但是CCA只能描述低维空间下的线性系统。*
>
> *kernel CCA：*
>
> *在两个视图之间寻找两个非线性相关的投影，来使相关系数最大，如下图，X 和Y分别是两个视图，Wx和Wy是识别出的投影向量，然后通过如下方式寻找最大化的协方差系数：*
>
> $$\rho=\frac{w_x^T X Y^T w_y}{\sqrt{(w_x^T X X^T w_x)(w_y^T Y Y^Y w_y)}}$$
>
> *最后我们将会得到一组从大到小的协方差系数，然后根据协方差系数和通过不同的测度方式计算视图的相关性大小*

第四章 多视图组合方法

> *组合多个视图的一种传统方法是将所有多个视图连接成一个视图，以适应单一视图学习设置。但是，这种连接会导致对一个小的训练样本进行过拟合，并且在物理上没有意义，因为每个视图都有一个特定的统计属性。因此，采用多视图结合的先进方法来实现相对于单视图学习算法在学习性能上的提高。*

4.1 经典的协同训练概要

> *不再赘述,补充一个图*



4.2 协同训练如何组合多视图

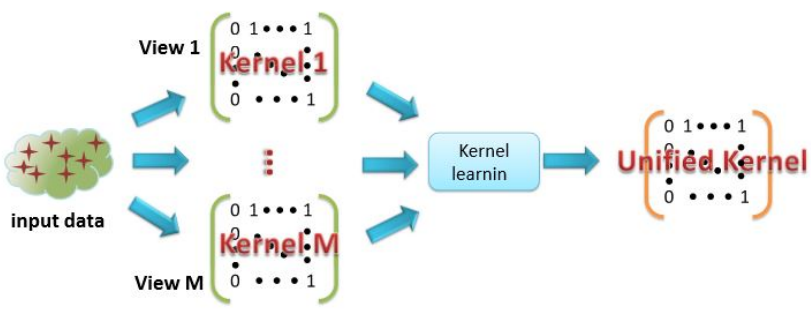
> *多视图学习主要应用半监督学习，将未标记数据作为验证集和将验证过程作为视图之间交换信息的重要步骤，但是对于有监督学习和无监督学习，多视图的组合方式会有所差别。*

> 1. *在无监督学习中没有验证集可以用，于是对于多个视图对应分类器的训练以及多个视图的联合训练都在同一个数据集上进行；*
>
>   *多视图学习应用到无监督学习的典型-多视图数据的光谱聚类算法多视图光谱聚类*

> 2. *有监督学习中的隐式验证集引入：引入函数来表示每个样本在分类器中的预测结果与每个视图的条件独立性，这种方式实际上隐式地连接了所有的视图，起到了验证集的作用（个人认为这种验证方式较弱）。*

4.3 多核学习概要

> *不同的内核可能对应不同的相似度量方式或来自不同表示(可能来自许多源或方式)的输入，因此组合内核是集成多个信息源并获取更好学习性能的一种方式，补充一个图:*



4.4 多核学习如何组合多视图

1. *线性组合方法*

   直接线性组合和加权线性组合：

   - Direct summation kernel

$$K(x_i, x_j) = \sum_{k=1}^{M} K_k(x_i, x_j),$$

   - Weighted summation kernel

$$K(x_i, x_j) = \sum_{k=1}^{M} d_k K_k(x_i, x_j).$$

   加权线性组合根据对权重施加限制的方式有许多版本

2. *非线性组合方法*

   基核的乘积或其他产生方式，比如指数和幂：

$$K(x_i, x_j) = exp(-\sum_{k=1}^{M} d_k x_i^T \mathbf{A}_k x_j),$$

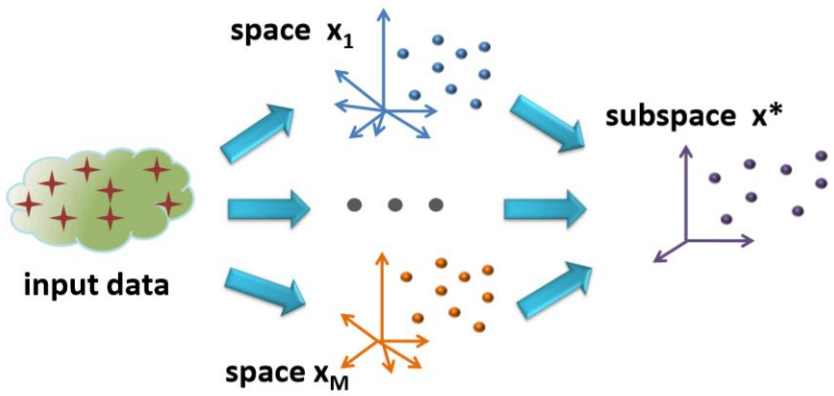
$$K(x_i, x_j) = (d_0 + \sum_{k=1}^{M} d_k x_i^T \mathbf{A}_k x_j)^n.$$

   基于核回归和核的多项式组合的非线性核组合方法：效果不好

4.5 子空间学习与组合多视图

基于子空间学习的方法的目的是通过假设输入视图是从这个潜在子空间生成的来获得一个被多个视图共享的潜在子空间; 典型相关分析 (CCA) 可以看作是 PCA 的多视图版本。

补充一张 基于CCA的多视图学习图



通过最大化子空间中两个视图之间的相关性，CCA 在每个视图上输出一个最优投影; 然而，由于CCA构造的子空间是线性的，因此不可能直接将其应用于许多具有非线性的真实世界数据集，由此引入 KCCA(Kernel CCA), 可以简单地理解为先通过核函数将低维特征映射到线性可分的高维空间，再施行 CCA。

> *列举了几篇基于子空间的多视图学习的典型文章，其组合多视图的方法不尽相同*

4.6 组合多视图方法的总结(分三类算法)

> 1. *联合训练式算法通常根据不同的观点训练不同的学习者，然后迫使他们在不同的观点之间保持一致。因此，这种方法可以看作是多种视图的后组合，因为在训练基础学习者时，视图是未组合的*

> 2. *多核学习方法中是首先根据单视图来单独计算核，然后通过基于核的组合来组合多视图的，这类方式是在训练之前或者训练中组合多视图*

> 3. *基于子空间学习的方法是通过假设输入视图是从一个潜在视图生成来获得一个合适的子空间。这种方法可以看作是多个视图的**先验组合**，因为多个视图是以**共享子空间假设**为前提的*

第五章 协同训练风格算法

5.1 协同训练的假设

> *协同训练划分两个不同的视图：*
>
> 1. *充足性：每个视图本身就足以进行分类，可以独立完成分类工作（实际情况没那么好，所以称作假设）*
> 2. *兼容性：两个视图的目标函数都极可能对共同出现的特征给出一样的预测标签*
> 3. *条件独立性：在进行预测时，各个视图之间是条件独立的（实际情况中不能满足，使用替代方案）*

> *条件独立假设------->弱依赖假设------->扩展假设-------->多样性（差异性）假设*
>
> *上述是对条件独立性假设逐步放宽，并且每一种假设都得到了理论和实践证明，目前使用最多的是最后一种，假设：*
>
> *只要协同训练的两个学习器之间的差异性大于其单独的错误率，那么随着协同训练的进行，最终得到的单视图相比于单视图学习器就会有改善，这种改善是通过不同视图下的分类器为对方标注越来越多的样本（交换信息）来实现的。*

5.2 经典协同训练介绍

> *较为简单，且已经在第一章和第四章做过介绍，不再赘述*

5.3 协同期望最大化算法（Co-EM）

> *协同训练在条件独立假设成立的前提下，每一个分类器为对方增加的样本等同于随机样本，其将会为对方视图增加一定信息量，分类器的学习是前进的；如果条件假设不成立或者更弱了，那么增加的信息量会更少，协同训练可能会失败。*
>
> *Co-EM的思想是，分类器不再为未标记的数据给出标签，而是给出下一次迭代需要使用的概率标签，最终的分类器也输出类的概率。*

5.4 协同正则化（Co-regularization）

> *最近一些文章和研究工作将协同训练以更加规范的形式描述为一个协同正则化的过程，其最终的优化器描述如下：*

$$(f_*^1, f_*^2) = \min_{f^1 \in H^1, f^2 \in H^2} \gamma_1 \|f^1\|_{H^1}^2 + \gamma_2 \|f^2\|_{H^2}^2 + \mu \sum_{i \in U} [f^1(x_i) - f^2(x_i)]^2 + \sum_{i \in L} V(y_i, f(x_i)).$$

> *其中H1和H2为两个视图对应的希尔伯特空间，f1和f2为待求解的预测函数，U为未标记样本，L未标记样本，式子第一项和第二项通过再生希尔伯特空间的二范数来度量复杂度（这里不太理解，希尔伯特空间相关的不太懂）；第三项描述了分类器对未标记样本的一致性误差，第三项描述了分类器对于标记样本在损失函数为V时的经验误差；其他字母为可以调整的权重值；*

5.5 协同回归（Co-regression）

> *协同训练风格的半监督回归算法1：使用了两个k近邻(k-nearest neighbor, kNN)回归元，在学习过程中，每一个回归元对另一个未标记的数据进行标记，通过对未标记样本进行标记对已标记样本的影响来估计这种标记的置信度。*
>
> *另一种协同回归算法，其目标函数如下：*

$$Q(f) = \sum_{v=1}^{M} \Big[\sum_{x \in X_v} V(y(x), f_v(x)) + \nu \|f_v(\cdot)\|^2 \Big] + \lambda \sum_{u,v=1}^{M} \sum_{z \in Z} V(f_u(z), f_v(z)),$$

> *其中共有M个视图，在希尔伯特空间中完成复杂性的度量（第一项的第二个子项）；第一项的第一个子项为对标记样本的预测值与真实值之间的损失；第二项描述了两两视图对于未标记数据的看法一致性损失；*

5.6 协同聚类（Co-clustering）

> *协同训练最初是为半监督学习设计的，但是其也可以用于无监督学习问题（聚类）*
>
> *主流的聚类算法都可以设计出协同训练版本：基本思想是在一个视图下运行经典的聚类算法然后将其结果信息交还给另一个视图，再于另一个视图下运行经典的聚类算法，以此迭代。*
>
> *简要介绍了Kumar和Daum的光谱聚类算法（前面已经出现过多次，这篇文章或许应该看一下）*

5.7 基于图的协同训练（ Graph-based Co-training）

> *主要讲了这篇文章的算法结构-> 基于图的协同训练*

5.8 多学习器算法（Multi-learner Algorithms）

> *第一种：利用未标记数据来提高标准监督学习算法的性能*
>
> *这种算法不要求“条件独立性”，只需要假设可以将样本空间划分为一组等价类；*

> *这里等价类没有说明等价关系，在没看文章的前提下理解为一个等价类中的子集可以对未标记样本产生同样的预测结果；*
>
> *就本叙述来看，这种算法本质上是将经典协同训练里的正负样本的加入，改造为适应于多分类的“等价类划分”和“等价类”标记，即对于未标记样本的标记不是正负标记，而是类别标记，隐式地舍弃了冗余视图？*

> *第二种：三训练协同训练算法*
>
> *这种算法从原始的标记数据中生成三个分类器，然后使用未标记数据对分类器进行改进。当两个分类器对于一个未标记数据达成一致的时候，才对该数据进行标记。这种三训练算法不需要视图的充足性和冗余视图*

> *第三种: 多训练 SVM (multi-training SVM)*
>
> *首先，利用随机子空间法可以从原始输入特征中获得一系列特征子集，即数据的多个视图。然后可以在这些生成的视图上学习多个分类器，并可以在半监督的相关性反馈设置中互相训练。最后，利用多数投票规则生成最优分类器*
>
> *不看其文章的话，暂时没看出来其SVM体现在哪里。*

第六章 多核学习

> *多核学习中的内核（不同特征空间映射方式）自然对应于不同内核，即对应于不同视图，对于基本内核的组合可以提高学习性能，因此多核学习可以看作是多视图学习的一部分。*

6.1 Boosting Methods

> *Ensemble method: “集成”多种机器学习方法*
>
> *Boosting Method：先使用一种简单方法得到一个结果，再使用其他方法来重点处理错误数据以“提升”效果*

> *多重加性回归核算法：the Multiple Additive Regression Kernels (MARK)*
>
> *考虑一个由大量的核函数和参数组成的大型核矩阵库，然后将核函数进行线性组合，每一个核函数可以是任意类型，比如核函数K 可以是RBF核函数的不同参数版本，其决策函数如下：*

$$f(x) = \sum_{i=1}^{N} \sum_{k=1}^{M} d_i^k K_k(x_i^m, x^m) + b,$$

> *其中i 为样本下标，k为核下标， d为加权因子；*
>
> *与集成学习类似，这种方法将不同的核视为不同的弱学习器，由此组成一个强学习器，*
>
> *而在多视图学习中这种动态生成的弱学习器又可以视为不同的视图；*

6.2 半正定规划（SDP）

> *一般形式的半正定规划如下：*
>
> $$\begin{aligned} \min_{x} \quad & c^T x \\ \text{s.t.} \quad & F(x) = F_0 + x_1 F_1 + \cdots + x_n F_n \geq 0 \\ & Ax = b, \end{aligned}$$
>
> *其中c为给定的数据向量，Fm为给定的数据矩阵，两个约束为线性矩阵不等式和线性等式；*
>
> *当数据的所有标签都是已知的，从数据中学习一个优化核矩阵的任务就是找到与标签集合y最大对齐的核矩阵K，那么这个问题可以表述为：*
>
> $$\begin{aligned} \max_{A,K} \quad & \langle K, yy^T \rangle \\ \text{s.t.} \quad & trace(A) \leq 1 \\ & \begin{pmatrix} A & K^T \\ K & I \end{pmatrix} \geq 0 \\ & K \geq 0. \end{aligned}$$
>
> *其中y是给定的样本集合，矩阵A的tr(A)小于1，其他约束与经典的SDP类似；目标函数为核矩阵完成分类器后与标签的误差最小；*
>
> *当部分标签未知时的情况较复杂，一点没看懂；*

6.4 半正定规划（SDP）、半无线线性规划（SILP）、简单多核学习(SimpleMKL)

> *这一类方法都属于将最优核的寻找问题转化为一个线性规划或者二线规划问题，其约束函数与学习器性能相关*

6.5 Group-LASSO方法

> *LASSO:最小绝对值收敛和选择算法，是一种回归算法，本质上是线性回归的L1正则化，即为损失函数后面添加一个L1正则化项；*
>
> *Group-LASSO：特征分组之后的LASSO（Group），同时给目标函数添加每一组的L2范数来惩罚，保证组间稀疏性；*

> *Composite Kernel Learning (CKL): 是受Group-LASSO启发，考虑了核之间的群体结构（Group）后对MKL进行了改进，即为Simple MKL的群体结构版本；*
>
> *之后有研究将L2范数推广到p范数，也有研究将对组的惩罚项采用对数惩罚；*

6.6 多核学习的边界

> *已经被证明：当选择的核是来自多个基本核的凸组合时，分类器的误差是有边界的，并且边界收敛；*
>
> *有研究工作表明：边界复杂度越大以及核数越多，那么误差边界也会增大；*
>
> *有研究工作表明：误差的下界将会随着内核族的伪维数的增加而增大；*
>
> *还有诸多情况下的误差下界证明已经被研究过；*

第七章 子空间学习

> *基于子空间学习的方法的目的是通过假设输入视图是从该子空间生成的，从而获得一个被多个视图共享的潜在子空间。除了典型相关分析(CCA)之外，还有其他更有效的构建子空间的方法。*

7.1 典型相关性分析（CCA）

> *典型相关分析(CCA)是一种建模两组(或更多)变量之间关系的技术，它已成功地应用于处理多视图数据的各种学习问题。*

7.1.1 CCA回顾

> *关于CCA算法计算相关性最大的投影向量的原理介绍，不再赘述*

7.1.2 Kernel CCA

> *典型相关分析(CCA)是一种线性特征提取算法，但对于许多呈现非线性的真实世界数据集，线性投影不可能捕捉到数据的属性。核方法通过将数据映射到高维空间，然后在该空间中应用线性方法，KCCA提供了一种处理非线性的方法。*

> *KCCA的推导如下：*
>
> *令 投影向量 $wx = X\alpha$ and $wy = Y\beta$，代入 如下经典的相关系数计算式子中*

$$\rho = \frac{w_x^T X Y^T w_y}{\sqrt{(w_x^T X X^T w_x)(w_y^T Y Y^Y w_y)}}$$

> *得到如下推导：*

$$\rho = \max_{\alpha,\beta} \frac{\alpha^T X^T X Y^T Y \beta}{\sqrt{\alpha^T X^T X X^T X \alpha \times \beta^T Y^T Y Y^T Y \beta}}$$

> *其中利用核矩阵概念，转化为如下优化问题：*

$$\begin{aligned} \max_{\alpha,\beta} &= \frac{\alpha^T K_x K_y \beta}{\sqrt{\alpha^T K_x^2 \alpha \times \beta^T K_y^2 \beta}} \\ \text{s.t.} &\quad \alpha^T K_x^2 \alpha = 1, \quad \beta^T K_y^2 \beta = 1. \end{aligned}$$

约束条件转化为如下特征值分解问题：

$$\begin{pmatrix} 0 & K_x K_y \\ K_y K_x & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \lambda \begin{pmatrix} K_x^2 & 0 \\ 0 & K_y^2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

根据分解所得的特征向量α β，求相关系数大小，最后得到使相关系数最大的两个特征向量，其标识着使得两个样本空间相关系数最大的投影方向；

7.1.3 CCA理论分析

这部分好难看懂，放弃

7.1.4 CCA的相关算法和研究

监督学习：将标签看作一个视图，样本数据根据标签提供的信息，定向投影到低维空间中，但是这种方法本质上没有使用样本数据的多个视图，不算多视图学习；

广义多视图分析(GMA)：同时利用有监督和无监督学习来进行特征提取，可以被视为CCA-监督学习的扩展，并有潜力取代CCA（未看具体文章的前提下暂时理解为相对于CCA-监督学习，其在无监督学习部分加入了多视图）

CCA聚类：在CCA所得的共享潜在子空间中进行传统聚类；

CCA 用于减少对标记样本数的依赖：在只有一个标记样本的前提下，使用CCA的相似度量方式来不断地增加正负样本数量，最终完成仅依赖一个样本的半监督学习

PCA与CCA结合来实现更加有效的降维；

7.2 多视图Fisher判别分析

Fisher判别分析：

从各个总体中抽取样本p来构造判别函数，先将样本投影到一维空间，如果所得的特征向量足以区分各个总体时即使用该特征向量构造判别函数，如果不可以，就将样本投影到二维空间，再联合两个特征向量构造判别函数，以此类推

CCA 中忽略了标签信息，于是采用Fisher判别分析来为多视图构造包含标签信息的投影方向；

7.2.1 两个视图的Fisher 判别分析

> *通过引入标签信息来改进CCA在向量空间中的投影方向，最后转化为如下优化问题：*
>
> $$\rho = \frac{w_a^T X_a^T y y^T X_b^T w_b}{\sqrt{(w_a^T X_a^T B X_a w_a + \mu \|w_a\|^2) \cdot (w_b^T X_b^T B X_b w_b + \mu \|w_b\|^2)}},$$
>
> *其中Xa和Xb分别代表不同的视图，wa和wb代表两个视图的权重（投影）向量，文中没有做过多介绍，我将y理解为标签信息的引入，二范数项目的不清；*
>
> *由于wa和wb的尺度不会影响优化问题的求解，于是优化问题转化为如下等式约束规划问题：*
>
> $$\begin{aligned} w_a^T X_a^T B X_a w_a + \mu \|w_a\|^2 &= 1, \\ w_b^T X_b^T B X_b w_b + \mu \|w_b\|^2 &= 1. \end{aligned}$$
>
> *最终通过拉格朗日乘子将等式约束优化转化为如下优化问题：*
>
> $$L = w_a^T X_a^T y y^T X_b^T w_b - \frac{\lambda_a}{2}(w_a^T X_a^T B X_a w_a + \mu \|w_a\|^2 - 1) - \frac{\lambda_b}{2}(w_b^T X_b^T B X_b w_b + \mu \|w_b\|^2 - 1)$$
>
> *通过求微分可以求得 投影向量wa和wb；*
>
> *由于初始优化器的构造方法不明确，建议详细看论文相关论文*

7.2.2 两个视图的Fisher 判别分析(核版本)

> *与KCAA的构造思路类似*

7.3 多视图嵌入

> *对多维特征进行降维是常用的一种分类器学习方法，但是当特征具有多重意义或者特征之间存在底层连接时，一般的特征降维方法比如PCA只会将单个特征独立看待进行降维，这将会丢失掉底层连接信息。于是需要一种更加先进的方法同时对多个特征进行嵌入，生成一个更具有意义的、所有特征共享的低维嵌入。这个观点来自于反复出现的文章“多视图光谱嵌入”*

> *multi-view spectral embedding (MSE),多视图光谱嵌入算法可同时编码多个视图的特征，实现物理意义上的嵌入。*
>
> *其本质是将不同的视图视为特征嵌入的物理单位，最后将来自不同的视图的低维特征进行全局对齐，得到一个所有视图共享的低维特征嵌入。*
>
> *之后还有许多对多视图嵌入算法的研究，包括但不限于**“进一步对来自不同视图的低维特征进行选择”**、**“针对对齐方式的改进”**。*

7.4 多视图度量学习

> *受跨媒体检索任务的启发，Quadrianto和Lampert(2011)研究了度量学习问题，经典的多视图度量学习的原则是：**如果样本是相关的，则将它们拉在一起，如果它们不是相关的，则将它们分开***
>
> *多视角度量学习的表述如下：*
>
> *对于两个不同的数据点的集合X 、Y，在其中一个集合中的某一数据点T，另一个集合中一定存在一个点集S，其中的数据点与T相似，假设X对应的特征空间为d1维的R1 ，Y对应的特征空间为d2的R2，我们需要找到两个投影函数，将其投影到共同的特征空间：D维的R；该特征空间包含了数据点的相似关系，且D小于d1和d2；*
>
> *当投影函数为线性函数的时候，令两个投影函数的参数集合为 w1和w2，则我们会得到如下目标函数：*
>
> $$L(w_1, w_2, X, Y, S) = \sum_{i,j=1}^{m} L^{i,j}(w_1, w_2, x_i, y_j, S_{x_i}) + \eta\Omega(w_1) + \gamma\Omega(w_2),$$
>
> *关于目标函数的构造细节未做过多说明，只解释了损失函数L可以分为相似项和不同项两个部分来体现：相似样本在低维空间中距离拉近，不同样距离拉远的思想。*
>
> *然后描述了多视图度量学习在图像分类问题中的贡献：*

7.5 潜在空间模型

> *上述方法专注于使用多视图数据进行有意义的降维（包括CCA风格的方法也是），但是还有一些专注于分析不同视图之间关系的方法。这些方法用于建立潜空间模型，通过潜变量，多个视图可以相互连接，信息可以从一个视图传播到另一个视图。*
>
> *这部分数学公式的推导太多且综述里对公式的解释不够清晰，所以没仔细看。*

第八章 多视图学习的应用

> *自然语言处理（文本、语音、邮件、网页等）：主要是web文档分类、语音中的情绪分类等问题。*

> *计算机视觉：人体动作识别、多视图目标识别、结合主动学习进行图像标注、基于内容的图像检索、*
>
> *面部表情识别、外观搜索、人脸识别、高光谱遥感图像分类、卡通人物检索等问题。*

> *降低数据量依赖、降低标记数据依赖、消除计算量问题。*

第九章 性能评价

9.1 多视图学习的数据集

> *已有多个数据集被广泛应用于多视图学习实验中，简要介绍如下*
>
> *WebKB: 是多视图学习中使用的最著名的数据集，首次在该数据集上评估联合训练算法。这个数据集由8282个学术网页组成，这些网页来自四所大学的计算机科学系网站:康奈尔大学、华盛顿大学、威斯康星大学和德克萨斯大学。这些网页可以分为六个类别:学生、职员、教员、系、课程和专题。有两个视图分别包含页面上的文本和超链接的锚文本。*
>
> *Citeseer: 是一个科学出版物的集合，包含了属于六个类的3312个文档。每个文档有三个自然视图: 文本视图(包括论文的标题和摘要) 、两个链接视图是入站引用和出站引用。*
>
> *来自UCI的一些流行的数据集也适合于评估多视图学习。例如，internet广告数据集包含来自各种web页面的图像，这些页面的特征可以是广告，也可以是非广告。这些实例根据6个视图进行描述，它们是图像的几何图形、基本url、图像url、目标url、锚文本和alt文本。*
>
> *在图像标注、图像分类和图像检索的实验中，还经常用到其他一些多媒体数据集，包括TRECVID2003视频数据集、Caltech256等。我们提取不同的视觉特征来代表数据的多个视图，如颜色直方图，边缘方向直方图，小波纹理。*

9.2 模型评价

> *罗列了一些论文和研究中的模型的效果*

第十章 总结

> *本文主要内容的总结：*
>
> 1. *与其采用一个视图作为特征或者将不同视图的特征简单地拼接起来（本质上还是单视图），从多视图数据中考虑不同视图的多样性更加有效；*
> 2. *根据多视图学习的趋势和发展将多视图学习分为三类：协同训练、多核学习、子空间学习；*
> 3. *多视图成功的关键是利用了视图之间的共识原则和互补原则；*
> 4. *如何构造视图和如何评估视图也值得研究，并且多视图被证明确实对学习性能有贡献；*

> *研究方向启发：*
>
> 1. *不同视图的特性对最终结果的影响很大，所以对于如何构建、分析和评估多视图是和重要的课题；*
> 2. *三种不同风格的多视图学习各有优点，但是很独立，因此开发一个包含诸多优点的多视图学习框架是值得关注的方向；*