The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Unified Locally Linear Classifiers with Diversity-Promoting Anchor Points

Chenghao Liu,[1,2] **Teng Zhang,**[1,3] **Peilin Zhao,**[4] **Jianling Sun,**[1,3] **Steven C.H. Hoi**[2]
[1]School of Computer Science and Technology, Zhejiang University, China
[2]School of Information Systems, Singapore Management University, Singapore
[3]Alibaba-Zhejiang University Joint Institute of Frontier Technologies, China
[4]School of Software Engineering, South China University of Technology, China
{twinsken, faramita, sunjl}@zju.edu.cn, chhoi@smu.edu.sg, peilinzhao@hotmail.com

Abstract

Locally Linear Support Vector Machine (LLSVM) has been actively used in classification tasks due to its capability of classifying nonlinear patterns. However, existing LLSVM suffers from two drawbacks: (1) a particular and appropriate regularization for LLSVM has not yet been addressed; (2) it usually adopts a three-stage learning scheme composed of learning anchor points by clustering, learning local coding coordinates by a predefined coding scheme, and finally learning for training classifiers. We argue that this decoupled approaches oversimplifies the original optimization problem, resulting in a large deviation due to the disparate purpose of each step. To address the first issue, we propose a novel diversified regularization which could capture infrequent patterns and reduce the model size without sacrificing the representation power. Based on this regularization, we develop a joint optimization algorithm among anchor points, local coding coordinates and classifiers to simultaneously minimize the overall classification risk, which is termed as Diversified and Unified Locally Linear Support Vector Machine(DU-LLSVM for short). To the best of our knowledge, DU-LLSVM is the first principled method that directly learns sparse local coding and can be easily generalized to other supervised learning models. Extensive experiments showed that DU-LLSVM consistently surpassed several state-of-the-art methods with a predefined local coding scheme (e.g. LLSVM) or a supervised anchor point learning(e.g. SAPL-LLSVM).

Introduction

Locally linear coding has been widely shown as a promising approach to approximate data on the nonlinear manifold (Van Gemert et al. 2008; Wang et al. 2010). The key idea behind this method is that a non-linear manifold behaves linearly in the local neighborhood, and that data on the manifold can be encoded locally in a local coordinate system established by a set of anchor points. Each data point can then be approximated through a linear combination of surrounding anchor points, and the weights are local coding coordinates which can be used for subsequent model training. Recently, several works have brought the advances of local coding techniques into classification task to enhance the capability of modeling nonlinear data (Yu, Zhang, and Gong 2009; Ladicky and Torr 2011; Gu and Han 2013). However, this work essentially divides the learning process into three independent stages: (1)computeing the anchor points; (2) encoding the training data with these anchor points; (3) training the classifier based on the results of encoding. More specifically, they first compute the anchor points with unsupervised learning methods such as clustering (Van Gemert et al. 2008) that does not take class label information into account, and then encodes the training data with a predefined local coding scheme. Finally, they feed the results of encoding to the downstream supervised classifier training process. We argue that these decoupled approaches oversimplify the original optimization problem, resulting in a large deviation due to the disparate purpose of each step. While local coding techniques minimizes the data reconstruction error and exploits unsupervised learning methods for anchor point learning, the primary objective of classifier learning is to minimize the classification error. The anchor points are obtained in an unsupervised fashion and the local coding coordinates are calculated with the predefined local coding scheme. Unsurprisingly, without making use of label information and optimizing a unified formulation, these anchor points and local coding coordinates are clearly not optimal for the classification task. It is desirable but challenging to achieve a unified optimization of anchor points, the local coding coordinates and the classifier model.

Furthermore, one key ingredient in the classification model is the regularization, which reduces overfitting by controlling the complexity of the model. While the effectiveness of ℓ_2-norm has been validated, there is still much room for improvement by designing a particular and appropriate regularization for locally linear classifiers, which has not been addressed yet. Specifically, each local classifier with respect to anchor points can capture nonlinear patterns according to discriminative information and geometric characters. Therefore, a regularization that encourages the local classifier to be diversified in terms of anchor points can reduce the correlation and redundancy between these classifier. Besides alleviating overfitting problem, this regularization can also (1) capture infrequent patterns: most local classifiers can capture frequent patterns that have dominant characteristic in the dataset, and promoting diversity among local classifiers can drive them to give infrequent and frequent patterns an equal treatment; they can also (2) reduce

the model size without sacrificing modelling power: low correlation and redundancy between local classifiers enhances the representational power of the overall model (Xie, Deng, and Xing 2015; Xie, Póczos, and Xing 2017).

In this paper, we propose a principled locally linear classifier called Diversified and Unified Locally Linear Support Vector Machine (DU-LLSVM). Instead of involving a three-stage learning scheme, we directly tackle the challenging joint optimization that should be treated adequately alongside locally linear classifiers. Our formulation is directly based on original locally linear classifiers where we optimize the anchor points, local coding coordinates and classifier simultaneously. By doing so, the proposed DU-LLSVM explicitly optimizes the anchor points and the local coding coordinates that fit the label information, and hence improves the accuracy. Besides, we propose a novel diversified regularization which encourages diversity between each local classifier, so that they can reduce the model size without compromising modelling power and better discover infrequent patterns. Experimental results on benchmark datasets show that DU-LLSVM outperforms state-of-the-art methods with predefined local coding scheme (LLSVM) or unsupervised anchor point learning (SAPL-LLSVM).

Our contribution are summarized as follows:

- We propose a local coding based diversified regularization in locally linear classifier that allows the local classifier to be close to being uncorrelated according to anchor points.
- We propose a joint optimization algorithm over the anchor points, local coding coordinates and classifiers to minimize classification risk simultaneously. To the best of our knowledge, DU-LLSVM is the first principled method that directly learns sparse local coding schemes and can be easily generalized to supervised learning model other than SVM.
- Through extensive experiments performed on benchmark datasets, we show that DU-LLSVM consistently surpasses several state-of-the-art methods with predefined local coding scheme (LLSVM) or supervised anchor point learning (SAPL-LLSVM).

Related Work

Diversified Regularization

Existing locally linear classifiers follow the popular ℓ_2-norm regularization in SVM which promotes large margin (Mao et al. 2015; Ladicky and Torr 2011; Gu and Han 2013). A specific regularization that could uncover the particular structure has not yet been discovered. Meanwhile, diversity promoting regularization has been widely used in classification (Malkin and Bilmes 2008), ensemble learning (Yu, Li, and Zhou 2011), and latent space model (Zou and Adams 2012; Xie, Deng, and Xing 2015; Xie, Singh, and Xing 2017). Given the weight vectors $\{\mathbf{w}_i\}_{i=1}^K$, Yu et al. (Yu, Li, and Zhou 2011) define the regularizer as $\sum_{1\leq j\leq k\leq K}(1-c_{jk})$ where c_{jk} defines the cosine similarity between weight vector j and k. In (Xie, Deng, and Xing 2015), the score is defined as mean of $\{arccos(|c_{jk}|)\}$ minus the variance of $\{arccos(|c_{jk}|)\}$. A larger mean allows vectors to have larger angles overall and a small variance encourages vectors to differ evenly from each other. Zou et al. (Zou and Adams 2012) employ the determinantal point process (DPP) to encourage weight vectors to have a large volume by adjusting the vectors to be close to orthogonal. From the perspectives of uncorrelation and evenness, (Xie, Singh, and Xing 2017; Li et al. 2017) proposed decorrelation regularizer which captures global relations among weight vectors. However, it is unclear how to apply it for improving locally linear classifiers. Our work advocates such a novel diversified regularization but one focused on locally linear classifiers which is fundamentally different from the above objectives.

Locally Linear Coding

Local coding methods offer a powerful tool for approximating data on the nonlinear manifold. All these methods employ a set of anchor points to encode data as a linear combination of surrounding anchor points, so as to minimize approximation error. Specifically, let $\mathbf{V}=\{\mathbf{v}_i\}_{i=1}^m$ denote the set of m anchor points, any point $\mathbf{x}$ is then approximated as $\mathbf{x}\approx\sum_{i=1}^m\gamma_{\mathbf{x},\mathbf{v}_i}\mathbf{v}_i$, where $\gamma_{\mathbf{x},\mathbf{v}_i}$ is the local coding coordinates, depicting the degree of membership of $\mathbf{x}$ to the ith anchor point $\mathbf{v}_i$, constrained by $\sum_{i=1}^m\gamma_{\mathbf{x},\mathbf{v}_i}=1$. Different encoding schemes haven been proposed in literature, with (Liu, Wang, and Liu 2011) proposing localized soft-assignment coding, which was defined as:

$$\gamma_{\mathbf{x},\mathbf{v}_i}=\begin{cases}\frac{\exp(-\beta d(\mathbf{x},\mathbf{v}_i))}{\sum_{j\in N_k(\mathbf{x})}\exp(-\beta d(\mathbf{x},\mathbf{v}_j))} & j\in N_k(\mathbf{x})\\ 0 & \text{otherwise,}\end{cases} \quad (1)$$

where $N_k(\mathbf{x})$ represents the k-nearest neighborhood of $\mathbf{x}$ defined by the distance function. In this coding , by using an "early-cut-off" strategy, the unreliable distant anchor points could be removed even when a small β is used. Moreover, localized soft-assignment coding saves computation overhead comparing with previous soft-assignment coding, which involves all the anchor points in the coding phase. Other local coding methods include local coordinate coding (Yu, Zhang, and Gong 2009), inverse Euclidian distance based weighting (Van Gemert et al. 2008; Ladicky and Torr 2011), etc.

A number of locally linear classifiers have been proposed based on local coding techniques. (Ladicky and Torr 2011) calculates the local coordinates with fixed and predefined local coding scheme, and then treats the local coordinates as weights for assigning training data into different local regions. Separate model are trained for each local region and combined to form a locally linear classifier. (Gu and Han 2013) adopts K-means to partition the data into clusters and then trains a linear SVM for each cluster. Meanwhile, each cluster's model needs to align with a global model, which can be treated as a type of regularization.

One can thus easily see that the aforementioned locally linear classifiers are essentially decoupled approaches where anchor points learning, codes for training data and classifier training are obtained through independent steps. The classification performance of these locally linear classifiers depends heavily on the quality of the local coding, which further depends on the anchor points being used. Therefore, as

we will review later, it is well acknowledged that such decoupled relaxation suffers from a lager deviation from the optimum. To the best of our knowledge, DU-LLSVM is a research gap we aim to fill in this paper, and due to the generic formulation in SVM, we believe that DU-LLSVM can be easily generalized to supervised models other than SVM.

Methods

Locally Linear Support Vector Machine

Given a set of training samples $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, A standard linear SVM binary classifier takes the form, $f^{SVM}(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$. The optimal weight vector $\mathbf{w}$ and bias b are obtained by maximizing the soft margin, which penalises each instance by the hinge loss:

$$\min_{\mathbf{w},b} \frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{1}{N}\sum_{n=1}^{N}\max(0, 1 - y_n f^{SVM}(\mathbf{x}_n)), \quad (2)$$

where the first term is an ℓ_2 regularization encouraging large margin and the second term is a hinge loss minimizing the empirical loss. Linear SVM classifiers are sufficient for many tasks, but they fail to capture the intrinsic decision boundary in non-linear problems. In many cases, real data naturally groups into clusters and lies on nearly disjointed lower dimensional manifolds, so that linear SVM is inapplicable. One solution to address this limitation is the locally linear classifiers (Ladicky and Torr 2011) which leverages the manifold geometric structure to learn a non-linear function that can be effectively approximated by a linear function with an coding under appropriate localization conditions. In other words, we assume that in an sufficiently small region, a nonlinear decision boundary is approximately linear and each data point $\mathbf{x}$ can then be approximated with a linear combination of surrounding anchor points, which is usually called a local coding scheme. To encode local linearity with linear SVM classifier, the weight vector $\mathbf{w}$ along with the bias b should vary according to the location of the data point $\mathbf{x}$ in the feature space as:

$$f(\mathbf{x}) = \mathbf{w}(\mathbf{x})^T\mathbf{x} + b(\mathbf{x}) = \sum_{d=1}^{p} w_d(\mathbf{x})x_d + b(\mathbf{x}), \quad (3)$$

where data point $\mathbf{x}$ lies in a lower dimensional manifold of the feature space whose dimensionality is p.

An important property of local coding is that any Lipschitz function $\psi(x)$ defined on a lower dimensional space can be approximated by a linear combination of function values $\psi(\mathbf{v})$ of the set of anchor points as $\psi(\mathbf{x}) \approx \sum_{j=1}^{m} \gamma_{\mathbf{x},\mathbf{v}_j}\psi(\mathbf{v}_j)$, within the boundary given in (Yu, Zhang, and Gong 2009). According to (Yu, Zhang, and Gong 2009; Ladicky and Torr 2011), smoothness and constrained curvature implies that the function $\mathbf{w}(\mathbf{x})$ and $\mathbf{b}(\mathbf{x})$ are Lipschitz smooth in the feature space $\mathbf{x}$. Thus we can approximate the weight functions $w_i(\mathbf{x})$ and bias function $b(\mathbf{x})$ in Equation (3) employing the local coding as:

$$w_d(\mathbf{x}) = \sum_{j=1}^{m} \gamma_{\mathbf{x},\mathbf{v}_j} w_d(\mathbf{v}_j) \quad b(\mathbf{x}) = \sum_{j=1}^{m} \gamma_{\mathbf{x},\mathbf{v}_j} b(\mathbf{v}_j). \quad (4)$$

Substituting the above equations (4) into Equation (3), we get the LLSVM decison function:

$$\begin{aligned} f_{\mathbf{W},\mathbf{b},\mathbf{v},\gamma_{\mathbf{x},\mathbf{v}}}(\mathbf{x}) &= \sum_{d=1}^{p}\sum_{j=1}^{m} \gamma_{\mathbf{x},\mathbf{v}_j} w_d(\mathbf{v}_j)x_d + \sum_{j=1}^{m} \gamma_{\mathbf{x},\mathbf{v}_j} b(\mathbf{v}_j) \\ &= \sum \gamma_{\mathbf{x},\mathbf{v}}^{\top}(\mathbf{W}\mathbf{x} + \mathbf{b}) = \sum_{j=1}^{m} \gamma_{\mathbf{x},\mathbf{v}_j} f_{\mathbf{v}_j}^{SVM}(\mathbf{x}), \end{aligned} \quad (5)$$

where $\mathbf{W} = [\mathbf{w}(\mathbf{v}_1), ..., \mathbf{w}(\mathbf{v}_m)]^T$ denotes a $m \times p$ matrix composed by stacking the m classifier weight vectors in rows. $\mathbf{b} = [b(\mathbf{v}_1), ..., b(\mathbf{v}_m)]^T$ and $\gamma_{\mathbf{x},\mathbf{v}} = [\gamma_{\mathbf{x},\mathbf{v}_1}, ..., \gamma_{\mathbf{x},\mathbf{v}_m}]^\top$ are m-dimensional vectors of bias terms and local coordinates respectively. This transformation can be seen as a finite kernel transforming a $p+1$-dimensional problem into a $m(p+1)$-dimensional one. It can also be interpreted as defining a locally liner classifier as the weighted average of m separate linear classifiers with respect to each anchor point, where the weights are determined by the local coding coordinates.

To evaluate $f_{\mathbf{W},\mathbf{b},\mathbf{v},\gamma_{\mathbf{x},\mathbf{v}}}(\mathbf{x})$ for each data point $\mathbf{x}$, we need to calculate the corresponding local coding coordinates $\gamma_{\mathbf{x},\mathbf{v}}$, that further depend on the anchor points $\mathbf{v}$ being used and the local coding scheme. This means the prediction function $f_{\mathbf{W},\mathbf{b},\mathbf{v},\gamma_{\mathbf{x},\mathbf{v}}}(\mathbf{x})$ depends on the model parameters $\mathbf{W}$ and b, the anchor point variable $\mathbf{v}$ and local coding coordinates $\gamma_{\mathbf{x},\mathbf{v}}$. Similar to the optimization problem (2) of SVM, the optimization problem for LLSVM is formulated as:

$$\min_{\mathbf{W},\mathbf{b},\mathbf{v},\gamma_{\mathbf{x}_n,\mathbf{v}}} \frac{\lambda}{2}\|\mathbf{W}\|_F^2 + \frac{1}{N}\sum_{n=1}^{N} \ell(y_n, f_{\mathbf{W},\mathbf{b},\mathbf{v},\gamma_{\mathbf{x}_n,\mathbf{v}}}(\mathbf{x}_n)), \quad (6)$$

where $\|\mathbf{W}\|_F^2 = \sum_{j=1}^{m}\sum_{d=1}^{p} W_{jd}^2$. Directly optimize $\mathbf{W}, \mathbf{b}, \mathbf{v}, \gamma_{\mathbf{x}_n,\mathbf{v}}$ simultaneously is very hard due to the non-convexity of the objective (6). This leads to a natural three-step approach taken by existing methods (Ladicky and Torr 2011; Yu, Zhang, and Gong 2009; Gu and Han 2013), which first estimates the anchor points for each data point by adopting K-means clustering, and then evaluate the local coding coordinates with a predefined scheme by utilizing exponential decay scheme or inversely-proportional decay scheme, finally they feed the anchor points and the local coding coordinates to the downstream supervised model training. Obviously, this two-step learning procedure is inconsistent with the objective function and rather suboptimal as the prediction information is not used in discovering the anchor points and the local coding scheme. This motivates a joint optimization method for local linear classifier which we will address later.

Local Coding Based Diversified Regularization

In this work, we propose a diversified regularization by taking two factors into consideration. First, we encourage uncorrelation between each local classifier. Less correlation is equivalent to more diversity, which would allow for local classifiers to be mutually different and thus improve their overall representation power. Second, we hope that the local classifier could contribute differently to the modeling of

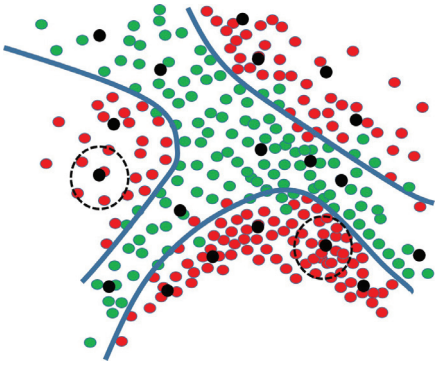


Figure 1: An illustration of diversified regularization in LLSVM. Red and green points correspond to positive and negative samples, black stars correspond to the anchor points and blue lines are the decision boundary. The dash line shows the range of a anchor point that the interior data points are relevant to.

data when considering the local coding system. For example, as shown in Figure 1, the left anchor point with the dotted line circle contains fewer data points as compared to the right anchor point. Therefore, placing more weight on the right local classifier may achieve a better performance as a whole. An intuitive way to compute the importance of each local score is to count the total weight that the total dataset has contributed to that anchor point. Specifically, we define the important score with anchor point $\mathbf{v}_i$ as $q_i = \frac{\sum_{n=1}^{N} \gamma_{\mathbf{x}_n,\mathbf{v}_i}}{\sum_{j=1}^{m}\sum_{n=1}^{N} \gamma_{\mathbf{x}_n,\mathbf{v}_j}} = \sum_{n=1}^{N} \gamma_{\mathbf{x}_n,\mathbf{v}_i}$ [1].

We define the diversity among local classifier from a statistical perspective. For two local classifier $\mathbf{w}_{j_1}$ and $\mathbf{w}_{j_2}$ (here we omit the bias term b for brevity), we promote $\mathbf{w}_{j_1}$ and $\mathbf{w}_{j_2}$ to be close to being orthogonal, making their inner product $\langle \mathbf{w}_{j_1}, \mathbf{w}_{j_2} \rangle$ close to zero and their norms $\|\mathbf{w}_{j_1}\|^2$ and $\|\mathbf{w}_{j_2}\|^2$ in proportion to their important scores q_{j_1} and q_{j_2}. Therefore, the novel diversified regularization can be achieved in the following manner: computing the Gram matrix $\mathbf{G} = \mathbf{W}^\top\mathbf{W}$, normalizing the matrix $\tilde{\mathbf{G}} = \frac{\mathbf{G}}{tr(\mathbf{G})}$ in the sense that $tr(\tilde{\mathbf{G}}) = 1$, where $tr(\cdot)$ denotes the trace of a matrix, and then promoting $\tilde{\mathbf{G}}$ to be close to $\mathbf{Q} = diag(\mathbf{q})$. Off the diagonal of $\tilde{\mathbf{G}}$ and $\mathbf{Q}$ are $\langle \mathbf{w}_{j_1}, \mathbf{w}_{j_2} \rangle$ and zero respectively. On the diagonal of $\tilde{\mathbf{G}}$ and $\mathbf{Q}$ are $\|\mathbf{w}_{j_1}\|^2$ and q_{j_1} respectively. Making $\tilde{\mathbf{G}}$ close to $\mathbf{Q}$ effectively encourages $\langle \mathbf{w}_{j_1}, \mathbf{w}_{j_2} \rangle$ to be zero and $\|\mathbf{w}_{j_1}\|^2$ in proportion to its important scores q_{j_1}, which is consistent with our original intention.

To measure the closeness between $\tilde{\mathbf{G}}$ and $\mathbf{Q}$, we use the Bregman matrix Divergence (Tsuda, Rätsch, and Warmuth 2005). If F is a real-valued strictly convex differentiable function on the parameter domain $\mathbb{R}^{d\times d}$ and $f(w) = \nabla_{\mathbf{W}} F(\mathbf{W})$, then the Bregman matrix divergence between two matrices $\mathbf{W}_1$ and $\mathbf{W}_2$ is defined as $\Delta_F(\mathbf{W}_1, \mathbf{W}_2) = F(\mathbf{W}_1) - F(\mathbf{W}_2) - tr((\mathbf{W}_1 - \mathbf{W}_2)f(\mathbf{W})^\top)$. If we choose $F(\mathbf{W}) = tr(\mathbf{W}\log\mathbf{W} - \mathbf{W})$, which is called quantum entropy, the Bregman matrix divergence becomes the quantum relative entropy (Nielsen and Chuang 2011), which is formulated as $\Delta_F(\mathbf{W}_1, \mathbf{W}_2) = tr(\mathbf{W}_1\log\mathbf{W}_1 - \mathbf{W}_1\log\mathbf{W}_2 - \mathbf{W}_1 + \mathbf{W}_2)$. Given the definition of quantum relative entropy, we can employ it to measure the closeness between $\tilde{\mathbf{G}}$ and $\mathbf{Q}$ to promote the diversity in locally linear classifiers. Since $tr(\tilde{\mathbf{G}}) = tr(\mathbf{Q}) = 1$, we have $\Delta_F(\mathbf{Q}, \tilde{\mathbf{G}}) = tr(\mathbf{Q}\log\mathbf{Q} - \mathbf{Q}\log\frac{\mathbf{W}^\top\mathbf{W}}{tr(\mathbf{W}^\top\mathbf{W})})$. Dropping the constant, we define the Local Coding based Diversified regularization as

$$R(\mathbf{W}) = tr(-\mathbf{Q}\log\frac{\mathbf{W}^\top\mathbf{W}}{tr(\mathbf{W}^\top\mathbf{W})}), \tag{7}$$

where $R(\mathbf{W})$ is a smooth and convex function that is easy to optimize.

Unified Optimization Framework

Algorithm 1 Local Coding Coordinates (LLC) Optimization Algorithm

Input: data point $\mathbf{x}$ and anchor points $\mathbf{V} = \{\mathbf{v}_1, \cdots, \mathbf{v}_m\}$
Initialization: $\lambda_0 = \mathbf{u}_1 + 1$, $k = 0$ and compute the vector of ascending ordered distance $\mathbf{u} \in \mathbb{R}^m$
while $\lambda_k > \mathbf{u}_{k+1}$ and $k \leq n-1$ **do**
  Update $k \leftarrow k+1$
  Compute λ based on (11)
end while
Output: The number of nearest anchor points k, compute the local coding coordinates $\gamma_{\mathbf{x},\mathbf{v}}$ based on (10)

Adding the diversified regularization, our objective function is

$$\min_{\mathbf{W},\mathbf{b},\mathbf{v},\gamma} \frac{\lambda_1}{2}\|\mathbf{W}\|_F^2 + \frac{\lambda_2}{2}R(\mathbf{W}) + \frac{1}{N}\sum_{n=1}^{N}\ell(y_n, f_{\mathbf{W},\mathbf{b},\mathbf{v}}(\mathbf{x}_n)), \tag{8}$$

The objective function (8) is a non-convex optimization problem, and thus we iteratively optimize $\gamma_{\mathbf{x},\mathbf{v}}$, $\mathbf{V}$, $\mathbf{W}$, $\mathbf{b}$ until convergence to obtain a local minimum.

We first present our optimization method for the local coding coordinates $\gamma_{\mathbf{x},\mathbf{v}}$. Considering $\mathbf{w}(\mathbf{x})$, we seek to find the best local approximation in a sense of minimizing the distance between this approximation and the ground truth. Assuming that for any data point $\mathbf{x}$, the ground truth holds that $\mathbf{w}_{\mathbf{x}} = \mathbf{w}(\mathbf{x}) + \epsilon_{\mathbf{x}}$, where $\mathbf{w}(\cdot)$ is a Lipschitz continuous function that for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^p$ it holds that $|\mathbf{w}(\mathbf{x}_1) - \mathbf{w}(\mathbf{x}_2)| \leq L \cdot d(\mathbf{x}_1, \mathbf{x}_2)$ for some distance function $d(\cdot,\cdot)$ and $\epsilon_{\mathbf{x}}$ denotes a noise term that $\mathbb{E}[\epsilon_{\mathbf{x}}|\mathbf{x}] = 0$ and $|\epsilon_{\mathbf{x}}| \leq b$ for some given $b > 0$. To minimize the absolute distance between our approximation and the ground truth $\mathbf{w}(\mathbf{x})$, we need to solve the following optimization problem:

$$\min_{\gamma_{\mathbf{x},\mathbf{v}}} |\sum_{i=1}^{m}\gamma_{\mathbf{x},\mathbf{v}_i}\mathbf{w}_{\mathbf{v}_i} - \mathbf{w}(\mathbf{x})| s.t. \sum_{i=1}^{m}\gamma_{\mathbf{x},\mathbf{v}_i} = 1; \gamma_{\mathbf{x},\mathbf{v}_i} \geq 0, \forall i.$$

[1] Since we adopt stochastic gradient descent method, we accumulatively calculate the important score that historic data has contributed to.

Decomposing the above objective into a sum of bias and variance terms, we can transforms it into

$$|\sum_{i=1}^{m}\gamma_{\mathbf{x},\mathbf{v}_i}\mathbf{w}_{\mathbf{v}_i}-\mathbf{w}(\mathbf{x})|\leq|\sum_{i=1}^{m}\gamma_{\mathbf{x},\mathbf{v}_i}\epsilon_{\mathbf{v}_i}|+L\sum_{i=1}^{m}\gamma_{\mathbf{x},\mathbf{v}_i}d(\mathbf{v}_i,\mathbf{x}).$$

By Hoeffding's inequality it follows that $|\sum_{i=1}^{m}\gamma_{\mathbf{x},\mathbf{v}_i}\epsilon_{\mathbf{v}_i}|\leq C\|\gamma_{\mathbf{x},\mathbf{v}}\|_2$ for $C=b\sqrt{2\log(\frac{2}{\delta})}$, w.p. at least $1-\delta$. With a guarantee for solving the original objective with a high probability,, we can formulate the new problem as the following optimization:

$$\min_{\gamma_{\mathbf{x},\mathbf{v}}} C\|\gamma_{\mathbf{x},\mathbf{v}}\|_2+\gamma_{\mathbf{x},\mathbf{v}}^{\top}\mathbf{u}\quad s.t.\sum_{i=1}^{m}\gamma_{\mathbf{x},\mathbf{v}_i}=1;\gamma_{\mathbf{x},\mathbf{v}_i}\geq 0,\forall i,\ (9)$$

where $\mathbf{u}=\{Ld(\mathbf{v}_1,\mathbf{x}),\cdots,Ld(\mathbf{v}_m,\mathbf{x})\}$. Its Lagrangian is $\|\gamma_{\mathbf{x},\mathbf{v}}\|_2+\gamma_{\mathbf{x},\mathbf{v}}^{\top}\mathbf{u}+\lambda(1-\sum_{i=1}^{m}\gamma_{\mathbf{x},\mathbf{v}_i})-\sum_{i=1}^{m}\theta_i\gamma_{\mathbf{x},\mathbf{v}_i}$, where $\lambda\in\mathbb{R}$ and $\theta_1,\cdots,\theta_m$ are the Lagrange multipliers. Since the optimization problem is convex, we use KKT conditions to find its global minimum. Setting the partial derivative of $\mathcal{L}(\gamma_{\mathbf{x},\mathbf{v}},\theta,\lambda)$ with respect to $\gamma_{\mathbf{x},\mathbf{v}}$ to zero gives: $\frac{\gamma_{\mathbf{x},\mathbf{v}_i}}{\|\gamma_{\mathbf{x},\mathbf{v}}\|_2}=\lambda-\mathbf{u}_i+\theta_i$. Let $\gamma^{\star}_{\mathbf{x},\mathbf{v}}$ be the optimal local coding coordinates. According to the KKT conditions, if $\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}>0$ then $\theta_i=0$. Otherwise, for any i such that $\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}=0$ it implies $\lambda\leq\mathbf{u}_i$. Substituting it into the equality constraint $\sum_i\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}=1$, we have

$$\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}=\frac{\lambda-\mathbf{u}_i}{\sum_{\gamma^{\star}_{\mathbf{x},\mathbf{v}_j}>0}(\lambda-\mathbf{u}_j)}=\frac{\lambda-Ld(\mathbf{v}_i,\mathbf{x})}{\sum_{\gamma^{\star}_{\mathbf{x},\mathbf{v}_j}>0}(\lambda-Ld(\mathbf{v}_j,\mathbf{x}))}.\quad(10)$$

It demonstrates that the optimal weight $\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}$ of anchor point $\mathbf{v}_i$ is proportional to $-d(\mathbf{v}_i,\mathbf{x})$, whose weight decay is quite slow compared to the popular exponential decay scheme or inversely-proportional decay scheme that is used in (Mao et al. 2015; Gu and Han 2013; Ladicky and Torr 2011). It also shows that parameter λ has a cutoff effect such that only the nearest anchor points that $\lambda-Ld(\mathbf{v}_i,\mathbf{x})>0$ are considered for encoding data point $\mathbf{x}$, while the weights for the remaining anchor points are all set to zero. This is consistent with the previous predefined local coding scheme. The solution to find the optimal distance threshold λ is simple. Denoting by k the number of nonzero weights which correspond to the k smallest value of $\mathbf{u}$, we easily obtain $1=\sum_{\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}>0}\frac{\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}}{\|\gamma^{\star}_{\mathbf{x},\mathbf{v}}\|_2}=\sum_{\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}>0}(\lambda-\mathbf{u}_i)^2$, which is equivalent to $k\lambda^2-2\lambda\sum_{i=1}^{k}\mathbf{u}_i+(\sum_{i=1}^{k}\mathbf{u}_i^2-1)=0$. Solving this quadratic equation with respect to λ and ignoring the solution that violate $\gamma^{\star}_{\mathbf{x},\mathbf{v}_i}\geq 0$, we get

$$\lambda=\frac{1}{k}(\sum_{i=1}^{k}\mathbf{u}_i+\sqrt{k+(\sum_{i=1}^{k}\mathbf{u}_i)^2-k\sum_{i=1}^{k}\mathbf{u}_i^2}).\quad(11)$$

Note that the objective (9) is a convex optimization problem, which can be efficiently solved using off-the-shelf toolbox. Here we follow the method in (Anava and Levy 2016; Liu et al. 2017b). The key idea is to greedily add neighbors according to their distance from $\mathbf{x}$ until a stopping condition is achieved. Our algorithm is presented in Algorithm 1.

Algorithm 2 Diversified and Unified Locally Linear SVM (DU-LLSVM)

Input: Training Data $(\mathbf{x}_n,y_n)_{n=1}^{N}$, the number of anchor points m, parameters $\lambda,t_0,skip,\mu$.
Output: Classifier variables $\mathbf{W},\mathbf{b}$ and anchor points $\mathbf{v}$, $t=0$.

```
Initialize anchor points v by K-means.
while no convergence do
  Sample a data point x randomly.
  Compute the local coordinate γ_{x,v} according to Algorithm 1 and the incurred loss ℓ_t.
  if ℓ_t > 0 then
    for each nearest anchor point v_i to data point x do
      update v_i via Equation (12)(13).
      update the Classifier parameters W and b via Equation (15).
    end for
  end if
  if t mod skip == 0 then
    update weight matrices W via Equation (16).
  end if
  Update: t ← t + 1
end while
```

For the anchor points and local classifiers estimation, we apply the SGD method to the objective in (6) which is simple and efficient. Since data point $\mathbf{x}$ is approximated as a linear combination of its k-nearest anchor points, only k-nearest anchor points need to be optimized at each iteration. To update the anchor point $\mathbf{v}$, we take partial derivative of $\gamma_{\mathbf{x},\mathbf{v}}$ with respect to $\mathbf{v}$, from which we obtain a $p\times m$ matrix, among which only k columns are nonzero. The ith column of $\frac{\partial\gamma_{\mathbf{x},\mathbf{v}}}{\partial\mathbf{z}_i}$ is computed as:

$$\frac{\mathbf{s}\mu(\lambda-\mu d(x,\mathbf{v}_i)-\sum_{\gamma_{\mathbf{x},\mathbf{v}_i}>0}(\lambda-\mu d(x,\mathbf{v}_i)))}{\sum_{\gamma_{\mathbf{x},\mathbf{v}_j}>0}(\lambda-\mu d(x,\mathbf{v}_j))^2},\quad(12)$$

where $\mathbf{s}=\frac{\partial d(\mathbf{x},\mathbf{v}_i)}{\partial\mathbf{v}_i}$ and $\mu=L/C$, namely the Lipschitz to noise ratio. The other nonzero columns are computed as:

$$-\frac{\mathbf{s}\mu}{\sum_{\gamma_{\mathbf{x},\mathbf{v}_j}>0}(\lambda-Ld(\mathbf{x},\mathbf{v}_j))^2},\quad(13)$$

where $\mathbf{v}_j$ belongs to the k-nearest neighbours of $\mathbf{x}$ and it is not equal to $\mathbf{v}_i$. Then we can update anchor points $\mathbf{v}_i$ via the following formula:

$$\mathbf{v}_i^{(t+1)}\leftarrow\mathbf{v}_i^{(t)}+\frac{1}{\rho_{anchor}(t+t_0)}y\frac{\partial\gamma_{\mathbf{x},\mathbf{v}}}{\partial\mathbf{v}_i}(\mathbf{W}^{(t)}\mathbf{x}+\mathbf{b}^{(t)}),\quad(14)$$

where t denotes the current iteration number and t_0 is a positive constant that avoids too large steps in the first few iterations. We follow the optimal learning rate $\frac{1}{\rho(t+t_0)}$ given by (Shalev-Shwartz et al. 2011). The classifier variables W and

Dataset	#Training	#Test	#feature
phishing	7370	3685	68
Magic04	12680	6340	10
IJCNN	49990	91701	22
w8a	49749	14951	300
connect-4	40740	20368	126
Covtype	387342	193670	54

Table 1: Basic statistics of datasets.

b can be updated by:

$$\mathbf{W}^{(t+1)} \leftarrow \mathbf{W}^{(t)} + \frac{1}{\rho(t+t_0)} y(\gamma_{\mathbf{x},\mathbf{v}} \mathbf{x}^T),$$
$$\mathbf{b}^{(t+1)} \leftarrow \mathbf{b}^{(t)} + \frac{1}{\rho(t+t_0)} y\gamma_{\mathbf{x},\mathbf{v}}. \quad (15)$$

To speed up the training process, we adopt a similar strategy to (Bordes, Bottou, and Gallinari 2009) and perform a regularization update every $skip$ iterations by

$$\mathbf{W}^{t+1} \leftarrow \mathbf{W}^{t+1} - \frac{skip}{t+t_0}(\mathbf{W}^{t+1} + \nabla R(\mathbf{W}^{t+1})). \quad (16)$$

where $\nabla R(\mathbf{W}) = \frac{2\mathbf{W}}{tr(\mathbf{W}^\top \mathbf{W})} - \mathbf{W}(W^\top \mathbf{W})^{-1}\mathbf{Q}$. Algorithm 2 summarizes the proposed DU-LLSVM algorithm.

Experiments

In this section, we empirically investigate our proposed DU-LLSVM and U-LLSVM, compare them with three related methods and exhibit the experimental results on six benchmark datasets.

Experimental Setup

We conduct experiments on six real-world datasets which were normalized to have zero mean and unit variance in each dimension. The statistics of the datasets after preprocessing are summarized in Table 1. To make a fair comparison, all the algorithms are repeated over 5 experimental runs of different random permutation.

In our experiments, we compared the following methods:

- **SVM**: Standard linear SVM without any local coding embedding, which is a baseline method .
- **LLSVM**: Locally Linear SVM (Ladicky and Torr 2011). We first initialized the anchor points by K-means clustering and encode the training data with predefined local soft-assignment coding, and finally fed the coding results to the training model.
- **SAPL-LLSVM**: LLSVM with Supervised Anchor Points Learning (Mao et al. 2015). This method learns the anchor points but using a fixed localized soft-assignment coding scheme, which indicates that the number of anchor points is fixed for the whole dataset. This algorithm can be viewed as a strong baseline to validate the efficacy of the unified optimization framework we proposed.
- **U-LLSVM**: The proposed Unified LLSVM without the diversified regularization.

phishing	Test loss	Acc(%)	Test time
SVM	0.1964± 0.0006	92.24± 0.10	×1
LLSVM	0.1861± 0.0042	92.66± 0.23	×8.91
SAPL-LLSVM	0.1509± 0.0026	93.92± 0.18	×9.27
U-LLSVM	0.1026± 0.0026	95.66± 0.17	×8.66
DU-LLSVM	**0.0965± 0.0020**	**95.84± 0.34**	×8.91
Magic04	Test loss	Acc(%)	Test time
SVM	0.5004 ± 0.0014	78.33± 0.13	×1
LLSVM	0.4289 ± 0.0016	79.57± 0.30	×21.56
SAPL-LLSVM	0.3923± 0.0031	81.60± 0.18	×22.04
U-LLSVM	0.3339± 0.0017	84.88± 0.36	×22.92
DU-LLSVM	**0.3238± 0.0029**	**85.14± 0.44**	×22.92
IJCNN	Test loss	Acc(%)	Test time
SVM	0.1863± 0.0006	90.50± 0.11	×1
LLSVM	0.1130± 0.0039	94.51± 0.30	×15.52
SAPL-LLSVM	0.1054± 0.0015	95.17± 0.21	×15.78
U-LLSVM	0.0683± 0.0024	97.66± 0.11	×16.30
DU-LLSVM	**0.0584± 0.0018**	**98.26± 0.12**	×15.42
w8a	Test loss	Acc(%)	Test time
SVM	0.0567± 0.0003	97.07± 0.05	×1
LLSVM	0.0333 ± 0.0019	98.45± 0.25	×11.72
SAPL-LLSVM	0.0274± 0.0005	98.74 ± 0.18	×12.78
U-LLSVM	0.0262± 0.0004	98.74± 0.07	×10.01
DU-LLSVM	**0.0228± 0.0028**	**98.96± 0.31**	×10.28
connect-4	Test loss	Acc(%)	Test time
SVM	0.4257± 0.0008	81.50± 0.12	×1
LLSVM	0.3196± 0.0021	87.30± 0.25	×7.52
SAPL-LLSVM	0.2901± 0.0013	88.19± 0.15	×8.78
U-LLSVM	0.2663± 0.0030	89.15± 0.23	×9.30
DU-LLSVM	**0.2225± 0.0020**	**90.94± 0.21**	×9.92
Covtype	Test loss	Acc(%)	Test time
SVM	0.6535± 0.0007	68.73± 0.06	×1
LLSVM	0.4809± 0.0024	79.17± 0.04	×15.18
SAPL-LLSVM	0.4359± 0.0013	80.61± 0.10	×16.72
U-LLSVM	0.4190± 0.0031	81.75 ± 0.11	×14.30
DU-LLSVM	**0.4095± 0.0040**	**81.99± 0.21**	×14.52

Table 2: Comparison of different algorithms in terms of test loss, classification accuracy and test time (normalized to test time of SVM).

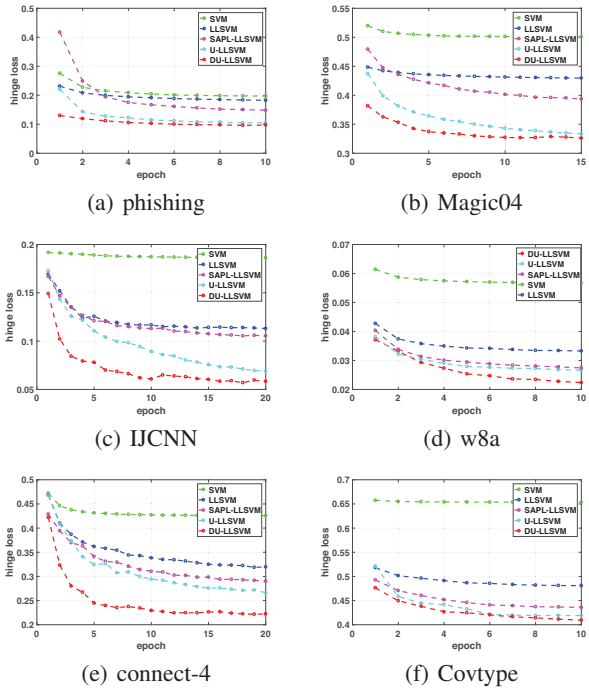
- **DU-LLSVM**: The proposed Diversified and Unified LLSVM.

For parameter settings, we performed grid search and cross validation to select the best parameters for each algorithm on the training set. We tuned the number of anchor points m from range $[10, 20, 50, 100]$, the nearest neighbouring parameter in LLSVM and SAPL-LLSVM from range $[2, 3, 5, 8, 10]$, the learning rate parameter ρ from range $[0.01, 0.001, 0.0001, 0.00001]$, learning rate parameter for anchor point from range $[0.01, 0.001, 0.0001]$, Lipschitz to noise ratio parameter μ from range $[0.01, 0.1, 0.5, 1, 10, 100]$, and $skip$ parameter from range $[10, 100, 1000, 10000]$.

Experimental Results and Analysis

As shown in Table 2, DU-LLSVM and U-LLSVM significantly outperform other baselines which validates the efficacy of unified optimization over the anchor point, local cod-

ing coordinates and SVM model. Moreover, DU-LLSVM is slightly better than U-LLSVM which confirms that promoting the diversity in local classifier could uncover infrequent patterns, and thus improve the performance. The test time of DU-LLSVM and U-LLSVM is comparable with SAPL-LLSVM and LLSVM even we adopt a unified optimization framework and impose a diversified regularization. Figure 2 demonstrates the convergence rate of each algorithm depending on epoch. It is clear to see that DU-LLSVM converges to a lower hinge loss value compared with other baselines.



(a) phishing (b) Magic04

(c) IJCNN (d) w8a

(e) connect-4 (f) Covtype

Figure 2: Epoch-wise demonstration of different algorithms with hinge loss on test data.

To illustrate the power of the local coding coordinates optimization method, we give the average number of nearest neighbours conditioned on epoch in Figure 4. It is obvious that the average number of nearest neighbours decreases and is prone to convergence. This is because the local coding scheme is being optimized, and Algorithm 1 will converge to the optimal number of nearest neighbours for each data point.

Figure 3 shows the test error of U-LLSVM and DU-LLSVM with the number of anchor points m ranging from 10 to 100. The performance of both U-LLSVM and DU-LLSVM increases with an increasing number of anchor points and stabilizes as m exceeds a certain threshold. Moreover, DU-LLSVM achieves better performance with fewer anchor points which confirms that the diversified regularization could reduce the model size without compromising modelling power and better discover infrequent patterns.

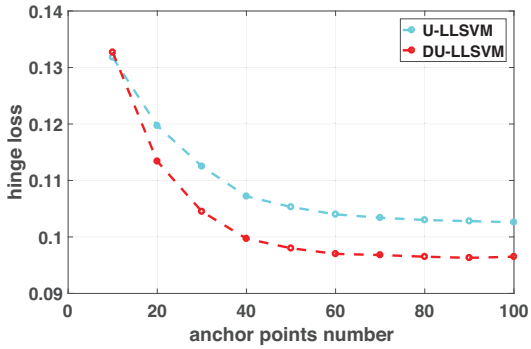
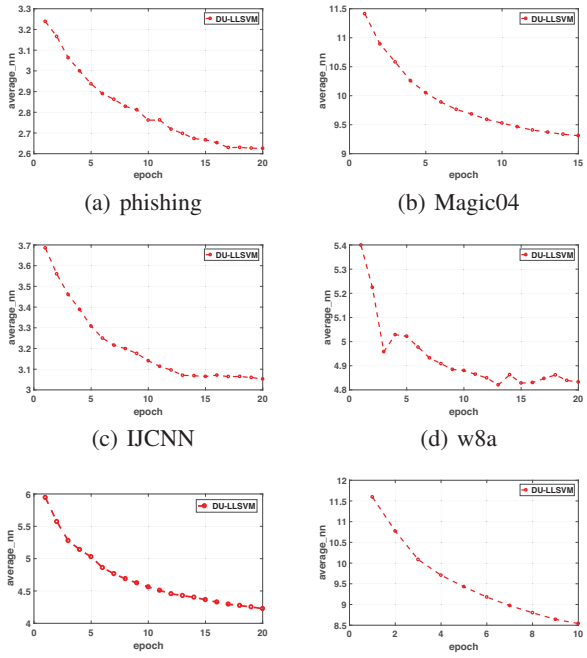


Figure 3: anchor points number vs hingeloss.



(a) phishing (b) Magic04

(c) IJCNN (d) w8a

(e) connect-4 (f) Covtype

Figure 4: Epoch-wise demonstration of the average number of nearest neighbour in DU-LLSVM.

Conclusion

In this work, we propose a local coding based diversified regularization which could capture infrequent patterns and reduce model size without sacrificing the representation power. We develop a joint optimization algorithm over the anchor points, local coding coordinates and classifiers to minimize classification risk simultaneously. Extensive experiments validated that DU-LLSVM consistently surpassed several state-of-the-art methods that use either predefined local coding scheme (LLSVM) or supervised anchor point learning(SAPL-LLSVM). Directions for future work include employing it for recommendation task (Liu et al. 2017a).

Acknowledgments

This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centres in Singapore Funding Initiative.

References

Anava, O., and Levy, K. 2016. k*-nearest neighbors: From global to local. In *Advances in Neural Information Processing Systems*, 4916–4924.

Bordes, A.; Bottou, L.; and Gallinari, P. 2009. Sgd-qn: Careful quasi-newton stochastic gradient descent. *Journal of Machine Learning Research* 10(Jul):1737–1754.

Gu, Q., and Han, J. 2013. Clustered support vector machines. In *Artificial Intelligence and Statistics*, 307–315.

Ladicky, L., and Torr, P. 2011. Locally linear support vector machines. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 985–992.

Li, J.; Zhou, H.; Xie, P.; and Zhang, Y. 2017. Improving the generalization performance of multi-class SVM via angular regularization. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 2131–2137.

Liu, C.; Jin, T.; Hoi, S. C. H.; Zhao, P.; and Sun, J. 2017a. Collaborative topic regression for online recommender systems: an online and bayesian approach. *Machine Learning* 106(5):651–670.

Liu, C.; Zhang, T.; Zhao, P.; Zhou, J.; and Sun, J. 2017b. Locally linear factorization machines. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 2294–2300.

Liu, L.; Wang, L.; and Liu, X. 2011. In defense of soft-assignment coding. In *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, 2486–2493.

Malkin, J., and Bilmes, J. 2008. Ratio semi-definite classifiers. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 4113–4116. IEEE.

Mao, X.; Fu, Z.; Wu, O.; and Hu, W. 2015. Optimizing locally linear classifiers with supervised anchor point learning. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, 3699–3706.

Nielsen, M. A., and Chuang, I. L. 2011. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. New York, NY, USA: Cambridge University Press, 10th edition.

Shalev-Shwartz, S.; Singer, Y.; Srebro, N.; and Cotter, A. 2011. Pegasos: Primal estimated sub-gradient solver for svm. *Mathematical programming* 127(1):3–30.

Tsuda, K.; Rätsch, G.; and Warmuth, M. K. 2005. Matrix exponentiated gradient updates for on-line learning and bregman projection. *Journal of Machine Learning Research* 6(Jun):995–1018.

Van Gemert, J. C.; Geusebroek, J.-M.; Veenman, C. J.; and Smeulders, A. W. 2008. Kernel codebooks for scene categorization. In *European conference on computer vision*, 696–709. Springer.

Wang, J.; Yang, J.; Yu, K.; Lv, F.; Huang, T.; and Gong, Y. 2010. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 3360–3367. IEEE.

Xie, P.; Deng, Y.; and Xing, E. 2015. Diversifying restricted boltzmann machine for document modeling. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1315–1324. ACM.

Xie, P.; Póczos, B.; and Xing, E. P. 2017. Near-orthogonality regularization in kernel methods. In *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence, UAI 2017, Sydney, Australia, August 11-15, 2017*.

Xie, P.; Singh, A.; and Xing, E. P. 2017. Uncorrelation and evenness: a new diversity-promoting regularizer. In *International Conference on Machine Learning*, 3811–3820.

Yu, Y.; Li, Y.-F.; and Zhou, Z.-H. 2011. Diversity regularized machine. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, 1603–1608. AAAI Press.

Yu, K.; Zhang, T.; and Gong, Y. 2009. Nonlinear learning using local coordinate coding. In *Advances in neural information processing systems*, 2223–2231.

Zou, J. Y., and Adams, R. P. 2012. Priors for diversity in generative latent variable models. In *Advances in Neural Information Processing Systems*, 2996–3004.