Low-rank geometric mean metric learning

Mukul Bhutani 1 Pratik Jawanpuria 2 Hiroyuki Kasai 3 Bamdev Mishra 2

Introduction. We propose a scalable solution for the Mahalanobis metric learning problem (Kulis, 2012). The Mahalanobis distance is defined as $d_A(x, x') = (x - x')^\top A(x - x')$, where $x, x' \in \mathbb{R}^d$ are input vectors and $A$ is a $d \times d$ symmetric positive definite (SPD) matrix. The objective is to learn a suitable SPD matrix $A$ from the given data. Since $A$ is a $d \times d$ SPD matrix, most state-of-the-art metric learning algorithms scale poorly with the number of features $d$ (Harandi et al., 2017). To mitigate this, a pre-processing step of dimensionality reduction (e.g., by PCA) is generally applied before using popular algorithms like LMNN and ITML (Weinberger & Saul, 2009; Davis et al., 2007).

Recently, (Zadeh et al., 2016) proposed the geometric mean metric learning (GMML) formulation, which enjoys a closed-form solution. However, it requires matrix $A$ to be positive definite, which makes it unsuitable in a high dimensional setting. To alleviate this concern, we propose a low-rank decomposition of $A$ in the GMML setting. Low-rank constraint also has a natural interpretation in the metric learning setting, since the group of similar points in the given dataset reside in a low-dimensional subspace. We jointly learn the low-dimensional subspace along with the metric. We show that the optimization is on the Grassmann manifold and propose a computationally efficient algorithm.

On real-world datasets, we achieve competitive results comparable with the GMML algorithm, even though we work on a smaller dimensional space.

Problem formulation. We follow a weekly supervised approach in which we are provided two sets $S$ and $D$ containing pairs of input points belonging to same and different classes respectively. Taking inspiration from GMML, we formulate the objective function as:

$$\min_{A \succeq 0} \quad \text{Tr}(AS) + \text{Tr}(A^\top D)$$

subject to $\text{rank}(A) = r$, 

(1)

where $S := \sum_{(x_i, x_j) \in S} (x_i - x_j)(x_i - x_j)^\top$, $D := \sum_{(x_i, x_j) \in D} (x_i - x_j)(x_i - x_j)^\top$, and $A^\dagger$ is the pseudoinverse of $A$.

Exploiting a particular fixed-rank factorization (Meyer et al., 2011), we factorize rank-$r$ matrix $A = UBU^\top$, where $U$ is an orthonormal matrix of size $d \times r$ and $B \succ 0$ is of size $r \times r$. Consequently, we rewrite (1) as:

$$\min_{U^\top U = I} \min_{B \succ 0} \quad \text{Tr}(UBU^\top S) + \text{Tr}(UB^{-1}U^\top D).$$

(2)

If we define $\tilde{S} = U^\top SU$ and $\tilde{D} = U^\top DU$, then the inner minimization problem has a closed-form solution as the geometric mean of $S^{-1}$ and $D$ (Zadeh et al., 2016). Using this fact, the outer optimization problem is readily checked to be only on the column space of $U$. The set of column spaces is the abstract Grassmann manifold, which is defined as the set of $r$-dimensional subspaces in $\mathbb{R}^d$. Equivalently, (2) is an optimization problem on the Grassmann manifold.

Extending the idea to a setting which weights the sets $S$ and $D$ unequally, we obtain the formulation

$$\min_{U^\top U = I} \min_{B \succ 0} \quad (1 - t)\delta_R(B, (U^\top SU)^{-1}) + t\delta_R^2(B, U^\top DU),$$

(3)

where $\delta_R$ denotes the Riemannian distance on the SPD manifold and $t \in [0, 1]$ is a hyperparameter. Similarly to (2), the problem (3) is also on the Grassmann manifold as the inner problem has a closed-form solution as the weighted geometric mean between $S^{-1}$ and $D$.

Results. Our proposed algorithm LR-GMML is implemented using the off-the-shelf conjugate gradients solver of Manopt (Boumal et al., 2014). The codes are available at https://github.com/muk343/LR-GMML. We compare LR-GMML with GMML on publicly available UCI datasets by measuring the classification error for a k-NN classifier following the procedure in (Zadeh et al., 2016). Parameter $t$ is optimized for both the algorithms and average errors over five random runs are reported in Figure 1.

Figure 1. Classification error rates of k-NN classifier comparing LR-GMML with GMML. We obtain comparable performance in lower ranks.
References


