Facial Age Estimation Using Clustered Multi-task Support Vector Regression Machine

Peter Xiang Gao
University of Waterloo
x33gao@uwaterloo.ca

Abstract

Automatic age estimation is the process of using a computer to predict the age of a person automatically based on a given facial image. While this problem has numerous real-world applications, the high variability of aging patterns and the sparsity of available data present challenges for model training. Here, instead of training one global aging function, we train an individual function for each person by a multi-task learning approach so that the variety of human aging processes can be modelled. To deal with the sparsity of training data, we propose a similarity measure for clustering the aging functions. During the testing stage, which involves a new person with no data used for model training, we propose a feature-based similarity measure for characterizing the test case. We conduct simulation experiments on the FG-NET and MORPH databases and compared our method with other state-of-the-art methods.

1 Introduction

Facial age estimation uses computers to estimate age by facial images. It can be widely applied in such areas as age-based access control, age-adaptive human computer interaction, and age-based advertising. For example, a computer may block violent content in a browser when it detects the user is an adolescent using its integrated camera; supermarket owners may estimate the average age of their customers by surveillance cameras and change their advertising strategy accordingly. Due to its potential in various kinds of applications, facial age estimation has been studied in fields as diverse as image processing, pattern recognition, and machine learning.

However, accurate age estimation presents two major challenges:

High variability of aging patterns. Factors such as angle, illumination, facial expressions, and makeup can reduce the robustness of the model. Further, variations between individual facial aging patterns make it difficult to use one global function for all people. Therefore, we try to build a model for each person, which falls into the category of *personalized age estimation*. Unlike *global age estimation*, which assumes that all people share the same facial aging pattern, *personalized age estimation* assumes that people may have their own facial aging patterns. It is generally agreed that personalized age estimation performs better than global age estimation [9].

Scarcity of training data. Building a model for each person's aging pattern requires facial images of that person at different ages. However, existing facial aging databases usually have less than 20 images per person, which is not enough for training individual models. To address this problem, we introduce the Clustered Multi-task Support Vector Regression Machine. A person's aging pattern is formulated as a task and an aging function is trained for this task. Similar tasks are clustered, which means people having similar aging patterns are in the same cluster. With clustered tasks, even if we do not have enough training samples for one person, we can still infer patterns from the same cluster to improve training.

Researchers have proposed several models to solve the facial age estimation problem. Xiao [12] showed that the k-Nearest Neighbor method has a low error rate during training. However, the method is overfitting, as the error is large when testing with another dataset. Lanitis [9] proposed the Weighted Person-specific Aging Function to build a stronger estimation model. Zhang [14] modelled age estimation as a multi-task Gaussian Process. However, Gaussian Process is slow as it needs to inverse a large kernel matrix. Fu [4] and Guo [7] proposed methods based on manifold learning, which essentially first reduces the dimensionality

of the input data and then applies subspace learning to each manifold. Guo [8] further improved the performance by using biologically inspired features. Suo [11] proposed to categorize human aging patterns as long-term and short-term. They managed to model the long-term aging patterns as a combination of short-term aging patterns by a smooth Markov Process. In their analysis, it is necessary to use the group-specified regression model which coincides with our multi-task learning approach.

In this paper, we propose the Clustered Multi-task Support Vector Regression Machine(CMTSVR) to solve the age estimation problem. Compared with other learning methods, CMTSVR can train a robust model with limited facial images per person. Through experiments, the estimation accuracy is competitive with other state-of-the-art methods.

2 Methodology

Each person's aging pattern can be represented as a series of face images at different ages, which is modelled as a *task*. We extract the features of the face images by the Active Appearance Model [2] using AM tools[1]. With feature extraction, shape, color, and texture information of a face image can be represented as a vector. We denote the features of i-th face image of the t-th task (or person) as a vector $\mathbf{x_{ti}}$ and the age of that image as y_{ti}. In this section, we introduce how to estimate y_{ti} based on the features of face image $\mathbf{x_{ti}}$.

We first have a brief overview of the Multi-task Support Vector Regression Machine (§2.1). To build models with limited training data per person, we cluster people with similar aging patterns in the same cluster and build a regression function for that cluster (§2.2). The traditional Multi-task Support Vector Machine assumes that testing data belongs to an existing task in the training data. However, for face age estimation, testing data is usually not derived from any task in training data. We use a feature-based similarity measure to soft classify the testing data (§2.3). The soft classification indicates the goodness of each cluster's regression function to the new testing data. The final estimation is based on the weighted average of each cluster's regression function.

2.1 Multi-task Support Vector Regression Machine

We extend the Multi-task Support Vector Machine [3, 13] to the Multi-task Support Vector Regression Machine (MTSVR).

[1] http://personalpages.manchester.ac.uk/staff/timothy.f.cootes/software/am_tools_doc/index.html

Suppose we have T learning tasks and the t'th task has m_t data points, so that they can be represented by $\{\{(\mathbf{x}_{ti}, y_{ti})\}_{i=1}^{m_t}\}_{t=1}^{T}$. The regression function for task t is:

$$f_t(\mathbf{x}) = \mathbf{w_t}\phi(\mathbf{x}) + b_t = (\mathbf{w}_0 + \mathbf{v}_t)\phi(\mathbf{x}) + (b_0 + c_t) \quad (1)$$

where $\mathbf{w_t}$ is the weight and b_t is the shift. Note that $\mathbf{w}_0$ is the mean of the $\mathbf{w}_t$, and $\mathbf{v}_t$ is the difference between them. Unlike MTSVM, we introduce a symbol c_t for MTSVR, which measures how far away b_t deviates from their mean b_0. Function $\phi(\cdot)$ is the feature mapping function which allows us to have non-linear regression. To find the regression function for each task, we solve the following optimization problem:

$$\underset{\mathbf{w_0}, \mathbf{v_t}, b_0, c_t}{\text{minimize:}} \quad (2)$$

$$\frac{\lambda}{2T}\sum_{t=1}^{T}\mathbf{v}_t^2 + \frac{1}{2}\mathbf{w}_0^2 + \frac{\lambda}{2T}\sum_{t=1}^{T}c_t^2 + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}(\xi_{ti} + \xi_{ti}^*)$$

subject to:

$$\begin{cases} y_{ti} - (\mathbf{w}_0 + \mathbf{v}_t)\phi(\mathbf{x}_{ti}) - b_0 - c_t \le \varepsilon + \xi_{ti} \\ (\mathbf{w}_0 + \mathbf{v}_t)\phi(\mathbf{x}_{ti}) + b_0 + c_t - y_{ti} \le \varepsilon + \xi_{ti}^* \\ \xi_{ti}, \xi_{ti}^* \ge 0 \end{cases}$$

where λ is the parameter that controls the similarity between tasks. ξ_{ti} and ξ_{ti}^* are the slack variables for prediction error, and C is their penalty. The variable ε is the tolerance as any prediction error less than ε will not be penalized. The problem can be solved by the Lagrange Multiplier in the following way:

$$\begin{aligned} L = {} & \frac{\lambda}{2T}\sum_{t=1}^{T}\mathbf{v}_t^2 + \frac{1}{2}\mathbf{w}_0^2 + \frac{\lambda}{2T}\sum_{t=1}^{T}c_t^2 \quad (3) \\ & + C\sum_{t=1}^{T}\sum_{i=1}^{m_t}(\xi_{ti} + \xi_{ti}^*) - \sum_{t=1}^{T}\sum_{i=1}^{m_t}(\eta_{ti}\xi_{ti} + \eta_{ti}^*\xi_{ti}^*) \\ & - \sum_{t=1}^{T}\sum_{i=1}^{m_t}a_{ti}(\varepsilon + \xi_{ti} - y_{ti} + (\mathbf{w}_0 + \mathbf{v}_t)\phi(\mathbf{x}_{ti}) + b_0 + c_t) \\ & - \sum_{t=1}^{T}\sum_{i=1}^{m_t}a_{ti}^*(\varepsilon + \xi_{ti}^* - (\mathbf{w}_0 + \mathbf{v}_t)\phi(\mathbf{x}_{ti}) - b_0 - c_t + y_{ti}) \end{aligned}$$

which is subject to $a_{ti}, \eta_{ti}, a_{ti}^*, \eta_{ti}^* \ge 0$. $a_{ti}, \eta_{ti}, a_{ti}^*, \eta_{ti}^*$ are all dual variables introduced by the Lagrange Multiplier. By setting partial derivatives of primal variables to 0 in Eq. 3, we find the optimal condition and obtain the dual problem. We obtain all the a_{ti} and a_{ti}^* by solving the dual problem. The optimal condition implies that $\mathbf{w}_0$, $\mathbf{v}_t$ and b_t can be expressed by a_{ti} and a_{ti}^*:

$$\mathbf{w}_0 = \sum_{t=1}^{T}\sum_{i=1}^{m_t}(a_{ti} - a_{ti}^*)\phi(\mathbf{x}_{ti}) \quad (4)$$

$$\mathbf{v}_t = \frac{T}{\lambda}\sum_{i=1}^{m_t}(a_{ti} - a_{ti}^*)\phi(\mathbf{x}_{ti}) \quad (5)$$

$$c_t = \frac{T}{\lambda}\sum_{i=1}^{m_t}(a_{ti} - a_{ti}^*) \quad (6)$$

and we find b_0 by plugging the training data of each task to

$$y_{ti} = f_t(\mathbf{x}_{ti}) = (\mathbf{w}_0 + \mathbf{v}_t)\phi(\mathbf{x}_{ti}) + b_0 + c_t$$

2.2 Clustered MTSVR

People with similar aging patterns have similar aging functions learned by MTSVR and we can cluster them by the similarity between aging functions. To compare similarity between two tasks s and t at point $\mathbf{x}$, we measure the squared difference of their aging functions.

$$\begin{aligned} d_{s,t}(\mathbf{x}) &= (f_s(\mathbf{x}) - f_t(\mathbf{x}))^2 \\ &= ((\mathbf{v}_s - \mathbf{v}_t)\phi(\mathbf{x}) + (c_s - c_t))^2 \end{aligned}$$

To measure the average similarity between task s and t on all data points of these two tasks, we define a distance function:

$$D(s,t) = \frac{\sum_{\mathbf{x}_{ti}} d_{s,t}(\mathbf{x}_{ti}) + \sum_{\mathbf{x}_{sj}} d_{s,t}(\mathbf{x}_{sj})}{m_t + m_s} \quad (7)$$

where m_t and m_s are the number of data points in task t and task s. With the function $D(s,t)$, we calculate a distance matrix $\mathbf{D}$ between all pairs of tasks for clustering. We then use Multi-Dimensional Scaling (MDS) to embed data points to a lower dimensional space where the pairwise distance of the points is preserved. Finally, we cluster the tasks by the k-means method and model a regression function for each cluster:

$$f_k(\mathbf{x}) = (\mathbf{w}_0 + \mathbf{v}_k)\phi(\mathbf{x}) + (b_0 + c_k) \quad (8)$$

Here, instead of modelling each person as a task, each cluster is modelled as a task.

$$\underset{\mathbf{w_0},\mathbf{v_k},b_0,c_k}{\text{minimize:}} \quad (9)$$

$$\frac{\lambda}{2K}\sum_{k=1}^{K}\mathbf{v}_k^2 + \frac{1}{2}\mathbf{w}_0^2 + \frac{\lambda}{2K}\sum_{k=1}^{K}c_k^2 + C\sum_{k=1}^{K}\sum_{i=1}^{m_k}(\xi_{ki} + \xi_{ki}^*)$$

subject to:

$$\begin{cases} y_{ki} - (\mathbf{w}_0 + \mathbf{v}_k)\phi(\mathbf{x}_{ki}) - b_0 - c_k \le \varepsilon + \xi_{ki} \\ (\mathbf{w}_0 + \mathbf{v}_k)\phi(\mathbf{x}_{ki}) + b_0 + c_k - y_{ki} \le \varepsilon + \xi_{ki}^* \\ \xi_{ki}, \xi_{ki}^* \ge 0 \end{cases}$$

Again, we solve Eq. 8,9 by the Lagrange Multipliers.

2.3 Assigning Unknown Testing Data

The aggregate prediction function is defined as the weighted sum of each cluster's regression function.

$$F(\mathbf{x}) = \sum_{k=1}^{K}\varpi_k(\mathbf{x})f_k(\mathbf{x}). \quad (10)$$

where $\varpi_k(\mathbf{x})$ is the likelihood that testing point $\mathbf{x}$ belongs to cluster k. We use the k-Nearest-Neighbour (kNN) in the feature space to find out the value of $\varpi_k(\mathbf{x})$. In the feature space, the distance between two data points $\mathbf{x}_{ti}$ and $\mathbf{x}_{sj}$ is represented by

$$\begin{aligned} &\sqrt{\phi(\mathbf{x}_{ti})^2 - 2\phi(\mathbf{x}_{ti})\phi(\mathbf{x}_{sj}) + \phi(\mathbf{x}_{sj})^2} \\ =&\sqrt{2 - 2\Phi(\mathbf{x}_{ti}, \mathbf{x}_{sj})} \end{aligned}$$

where the kernel function is the Radial Basis Function $\Phi(\mathbf{x}_{ti}, \mathbf{x}_{sj}) = \phi(\mathbf{x}_{ti})\phi(\mathbf{x}_{sj}) = e^{-\gamma(\mathbf{x}_{ti} - \mathbf{x}_{sj})^2}$. Parameter γ denotes the width of the kernel. A larger kernel value between two points implies a closer distance. Based on this knowledge, we use the following ways to assign weight to each task when a testing data point is given. Let $D_k(\mathbf{x}) = \frac{\sum_{\mathbf{x}_{ki}} \Phi(\mathbf{x}_{ki}, \mathbf{x})}{m_k}$ denote the average kernel value between points in cluster k and testing data $\mathbf{x}$, the weight of cluster k is $\varpi_k(\mathbf{x}) = \frac{D_k(\mathbf{x})}{\sum_{\hat{k}} D_{\hat{k}}(\mathbf{x})}$. We can predict testing data by Eq. 10.

3 Experiment

3.1 Dataset

There are two public aging databases available, the FG-NET [1] database and the MORPH [10] database. The FG-NET database contains 1,002 facial images of 82 people aged between 0 and 69. For each person, the dataset contains 6 to 18 images at different ages. The MORPH Album 2 contains over 20,000 images of about 4,000 people. Since the FG-NET has more face images per person, we use it for training and the MORPH is used for testing.

3.2 Results

We measure performance in two metrics. The first one is Mean Absolute Error (MAE), which is the average error between prediction and the true value. For testing data with n images, let y_i be the true age and $\hat{y_i}$ be the estimated age, the MAE is

$$\text{MAE} = \frac{\sum |y_i - \hat{y_i}|}{n}$$

The cumulative score s_i denotes the percentage of testing points whose error is less than or equal to i. Given n testing points, s_i is defined as

$$s_i = \frac{\sum_{j=1}^{n} \delta_{|y_j - \hat{y_j}| \le i}}{n}$$

Reference	Method	MAE
[6]	AAS	14.83
[6]	WAS	8.06
[6]	AGES	6.77
[5]	KAGES	6.18
[14]	SVR	5.91
[7]	LARR	5.07
[12]	mKNN	4.93
[14]	MTWGP	4.83
[8]	BIF	4.77
-	CMTSVR	**4.37**

Table 1. MAE on FG-NET

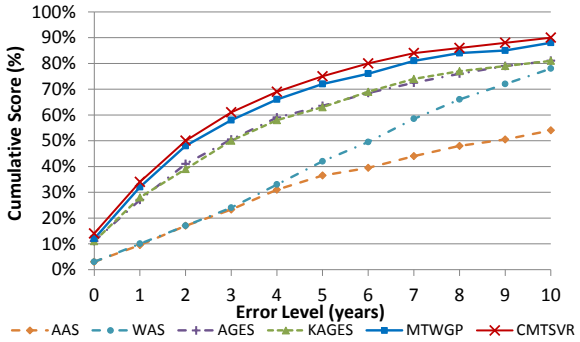

Figure 1. Cumulative scores on FG-NET

We first train the FG-NET database and test it by the Leave One Person Out scheme. All images of each person are excluded once from training data for testing purpose. The MAE of CMTSVR is 4.37 with a standard derivation of 5.46. Table 1 shows the Mean Absolute Errors of different age estimation methods on FG-NET database. Figure 1 shows the cumulative scores of the methods on the FG-NET database. We see that for both MAE and the cumulative scores, CMTSVR performs better than other methods.

The MORPH database does not have enough images per person to train a model. We only use it for testing. The MAE of our model is 5.62 with a standard deviation of 4.99. The MAEs of different methods on the MORPH database are shown in Table 2. We also compared the cumulative score of the methods in Figure 2. For most cases, the cumulative scores of CMTSVR outperform other methods.

4 Conclusion

In this paper, we propose Clustered Multi-task Support Vector Regression to solve the facial age estimation problem. We solve the high variability of aging patterns by using Multi-task learning and solve the scarcity of training data by clustering people with similar aging patterns. By testing on two public database, namely FG-NET and MORPH, we show that our approach is competitive to other state-of-the-arts methods.

Reference	Method	MAE
[6]	AAS	20.93
[14]	mKNN	10.31
[6]	WAS	9.32
[6]	AGES	8.83
[14]	LARR	7.94
[14]	SVR	7.20
[14]	MTWGP	6.28
-	CMTSVR	**5.62**

Table 2. Comparison of MAE on MORPH

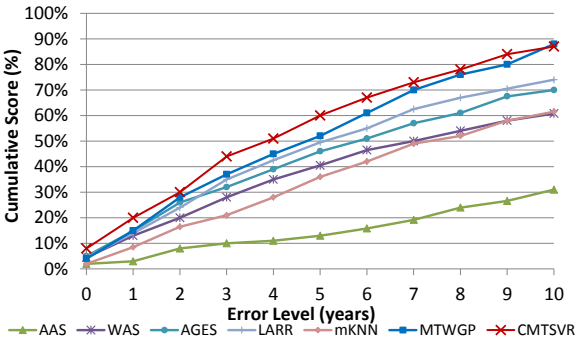

Figure 2. Cumulative scores on MORPH

References

[1] Fg-net database http://www.fgnet.rsunit.com/index.php.

[2] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[3] T. Evgeniou and M. Pontil. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04.

[4] Y. Fu and T. Huang. Human age estimation with regression on discriminative aging manifold. *IEEE Transactions on Multimedia,*.

[5] X. Geng, K. Smith-Miles, and Z.-H. Zhou. Facial age estimation by nonlinear aging pattern subspace. In *Proceedings of the 16th ACM international conference on Multimedia*, MM '08.

[6] X. Geng, Z.-H. Zhou, and K. Smith-Miles. Automatic age estimation based on facial aging patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*

[7] G. Guo, Y. Fu, C. Dyer, and T. Huang. Image-based human age estimation by manifold learning and locally adjusted robust regression. *IEEE Transactions on Image Processing*.

[8] G. Guo, G. Mu, Y. Fu, and T. Huang. Human age estimation using bio-inspired features. In *IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009.*

[9] A. Lanitis, C. Taylor, and T. Cootes. Toward automatic simulation of aging effects on face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence,*.

[10] K. Ricanek and T. Tesafaye. Morph: a longitudinal image database of normal adult age-progression. In *7th International Conference on Automatic Face and Gesture Recognition, 2006. FGR 2006.*

[11] J. Suo, X. Chen, S. Shan, and W. Gao. Learning long term face aging patterns from partially dense aging databases. In *IEEE 12th International Conference on Computer Vision, 2009*.

[12] B. Xiao, X. Yang, Y. Xu, and H. Zha. Learning distance metric for regression by semidefinite programming with application to human age estimation. In *Proceedings of the 17th ACM international conference on Multimedia*, MM '09.

[13] S. Xu, X.-d. Qian, L.-j. Zhu, X. An, and L.-d. Zhang. Multi-task least-squares support vector regression machines and their applications in nir spectral analysis. In *Guang Pu 2011*.

[14] Y. Zhang and D.-Y. Yeung. Multi-task warped gaussian process for personalized age estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.