Project 4: Face Social Traits and Political Election Analysis by SVM

Yufei Hu (404944367)

December 3, 2017

1 Introduction

This project follows the paper published by Jungseock et al. in ICCV 2015. The rationale is that election outcomes can be predicted solely based on geometric and appearance facial features. Further, these features can be mapped to high-level concepts of perception such as attractiveness or trustworthiness.

I exploit such low-level facial features and high-level perceptual to analyze election outcomes and party affiliations (GOP vs DEM) of politicians. This study is motivated by prior behavior studies in psychology, which suggest that people judge others by facial appearance. Some evidence was also found in election and jury sentencing.

The purpose of this project is to study the social attributes of human faces using Support Vector Machines (SVM). The goal is to:
(1) train classifiers that can infer the perceived face social traits from low-level features, and
(2) apply the model to analyze the outcomes of real-world political elections.

2 Face Social Traits Regression

The goal of this task is to train binary SVMs (or SVRs) to predict the perceived traits (social attributes) from facial photographs. We can use the pre-computed facial keypoint locations and extract HoG (histogram of oriented gradient) features using the enclosed MATLAB function. I use the *libsvm* package to train the SVR models.

2.1 Regression by Landmarks

I first trained 14 SVR models only using the provided facial landmarks as features. I wrote a script which reads the annotation file and the landmark file. I then trained 14 models - one for each attribute dimension using the training examples. After training is done, I apply the learned classifiers on the test examples and measure performance (classification accuracy) of the classifiers. Since the labels are imbalanced (different number of positive vs negative examples), I will report the average precisions. Perform k-fold cross-validation to choose the *libsvm* parameters.

I acquired 491 images and their corresponding 14 trait annotations (491 × 14 matrix) as training dataset. The 14 trait annotations correspond to {*Old*, *Masculine*, *Baby-faced*, *Competent*, *Attractive*, *Energetic*, *Well-groomed*, *Intelligent*, *Honest*, *Generous*, *Trustworthy*, *Confident*, *Rich*, *Dominant*}. The pre-computed facial landmark coordinates (491 x 160 matrix). Each row represents 80 keypoint locations [x_1, x_2, ..., x_{80}, y_1, y_2, ... , y_{80}].

I removed the landmarks that are the same for all images, resulting a 491 × 152 matrix. To avoid features in greater numeric ranges dominating those in smaller numeric ranges, I linearly scale each feature to the range [0, 1]. I used function libsvmtrain() to train 14 epsilon-SVR models. The epsilon-SVR model objective function is described as:

$$\min_{\omega,b,\xi,\xi^*} \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{l}\xi_i + C\sum_{i=1}^{l}\xi_i^*$$
$$\text{subject to } \begin{cases} \omega^T\phi(x_i) + b - z_i \le \varepsilon + \xi_i \\ z_i - \omega^T\phi(x_i) - b \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0, \text{ i=1,...,l} \end{cases} \tag{1}$$

where x_i is feature vector, z_i is the target output, C>0, ε>0.

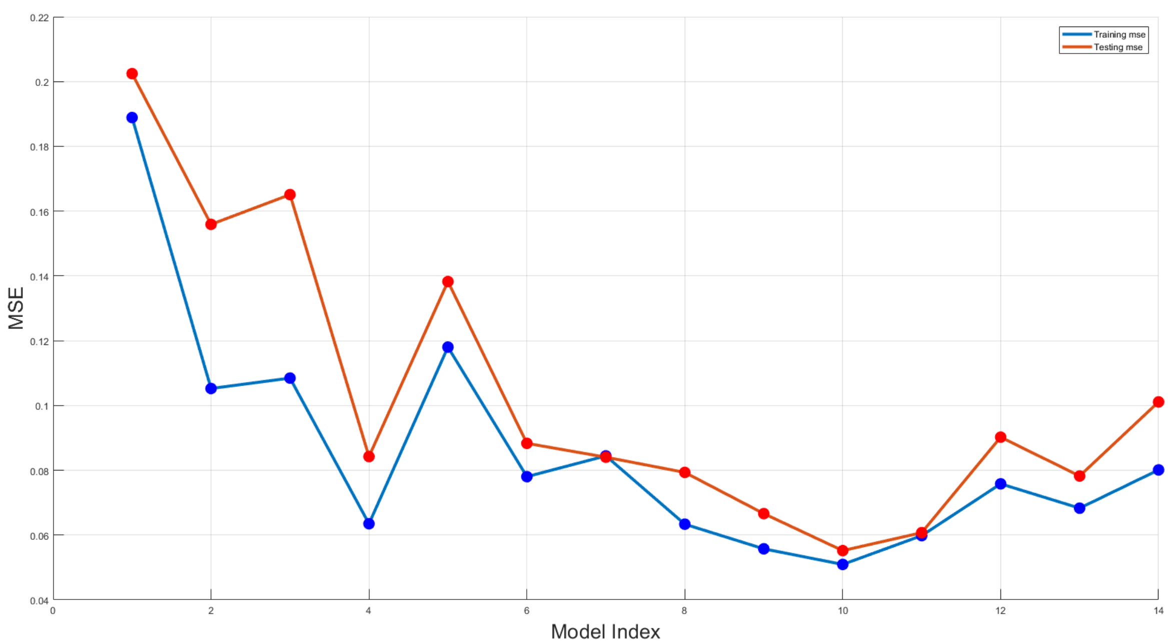

Figure 1: Training and testing MSE of the 14 SVR models

The outcomes of these 14 SVR models are illustrated in Figure 1. It is seen that the training MSEs (mean square errors) are generally smaller than testing MSEs. This phenomenon fits my intuition quite well.

Table 1: Parameters of SVR models using landmarks as input features

Trait index	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$\log_2(C)$	9	11	5	1	1	5	-5	13	9	13	11	1	13	13
$\log_2(\lambda)$	-7	-9	-1	-1	-1	-3	-17	-15	-9	-11	-11	-1	-15	-13
$\log_2(\varepsilon)$	-5	-5	-7	-5	-5	-5	-5	-7	-5	-5	-9	-5	-9	-5

Three hyperparameters were derived for the 14 epsilon-SVR models based on log grid search and 5-fold cross validation. For the parameter C, I searched from 2^{-5} to 2^{13}; for the parameter λ, I searched from 2^{-17} to 2^{-1}; and for the parameter ε, I searched from 2^{-9} to 2^{-5}. The optimal parameters are listed in the table above.

2.2 Regression by Rich Features

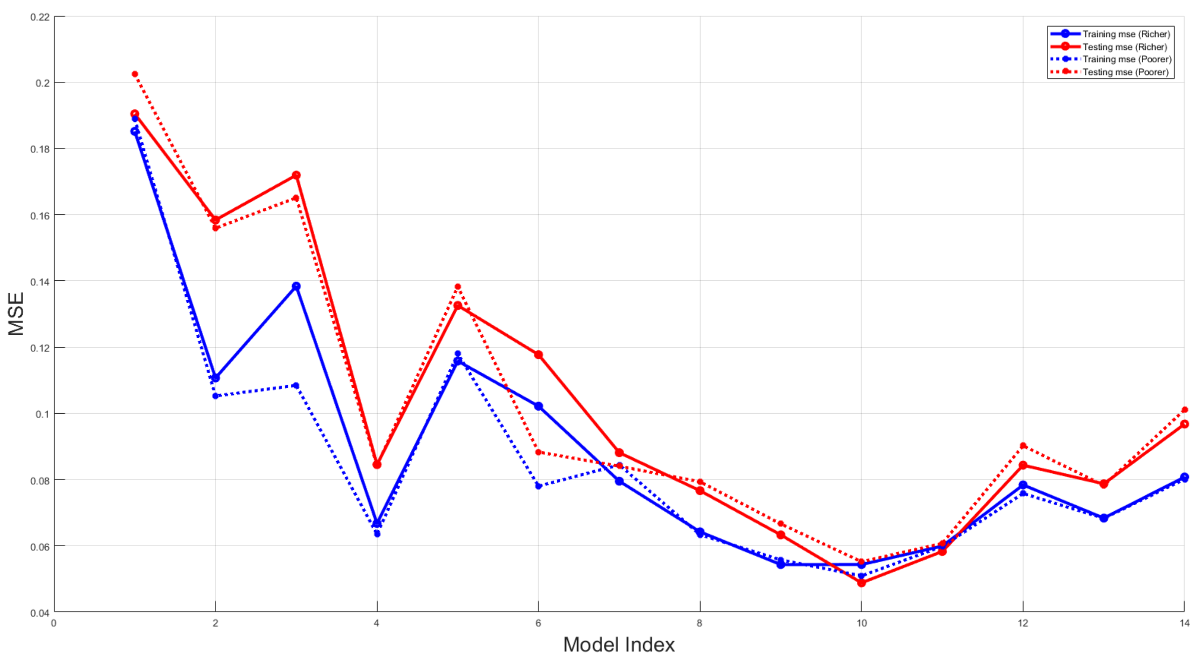

Figure 2: Training and testing MSE of the 14 SVR models using richer input features

In this section, I have included not just the original landmarks but also the HoG (histogram of oriented gradient) as input features and followed the exact same procedures as described in the previous section. The training and testing MSEs are shown in the figure above, together with the training and testing MSEs which only take landmarks as inputs. It can be seen that the training and testing MSEs with richer input features decreased by comparing to the original ones. The reason is that as the inputs include more information, the SVR models can find patterns between the data and the labels more easily.

Table 2: Parameters of SVR models using landmarks and HoG as input features

Trait index	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$\log_2(C)$	1	1	7	3	-3	1	-3	-5	3	-3	9	-1	3	9
$\log_2(\lambda)$	-1	-1	-17	-13	-3	-1	-3	-3	-11	-5	-17	-5	-13	-17
$\log_2(\varepsilon)$	-9	-5	-5	-7	-5	-9	-5	-5	-5	-5	-7	-5	-5	-9

Similarly, three hyperparameters were derived for the 14 epsilon-SVR models based on log grid search and 5-fold cross validation respectively. For the parameter C, I searched from 2^{-5} to 2^{13}; for the parameter λ, I searched from 2^{-17} to 2^{-1}; and for the parameter ε, I searched from 2^{-9} to 2^{-5}. The optimal parameters are listed in the table above.

3 Election Outcome Prediction

3.1 Direct Prediction by Rich Features

In this section, I used exactly the same features that I derived in section 2.2 and train an SVM classifier to classify the election outcomes. *ranksvm()* is used to train the model. The objective function of this model can be described below:

$$\begin{aligned} &\min \frac{1}{2}\|\omega\|_2^2 + C\sum \xi_{i,j} \\ &\text{subject to } \begin{cases} \omega^T f(I_i) + 1 - \xi_{i,j} \\ \xi_{i,j} \geq 0, \ \forall (i, j) \in D \end{cases} \end{aligned} \tag{2}$$

where D is the given data and their ranking orders, other parameters share the same meaning as those in equation (1).

I then trained 2 models which predict election outcomes of governors and senators respectively.

Table 3: Training and testing accuracy and optimal hyperparameter of the 2 SVM models

	Training accuracy	Testing accuracy	$\log_2(C)$
Governors	1.0000	0.7647	-3
Senators	1.0000	0.7778	9

Only one hyperparameter was derived for the 2 SVM models based on log grid search and 5-fold cross validation respectively. I searched from 2^{-15} to 2^{15} for the parameter C. The optimal parameters together with the training and testing accuracies of these 2 models are listed in the table above.

3.2 Prediction by Face Social Traits

I finally constructed a two-layer-model in which I first projected each facial image into a 14-dimensional attribute space and then perform binary classification of the election outcome in the obtained feature space. Specifically, I applied the regressors that I trained in section 1.2 to each politician's image and collected all the outputs of 14 regressors as a new feature vector that represents the image.

Since each race comprises two candidates, I defined a pair of politicians as one data point by subtracting a trait feature vector from another vector, and train a binary classifier without bias term. I then trained SVM classifiers using these new feature vectors as input to predict the election outcomes.

Table 4: Training and testing accuracy and optimal hyperparameter of the 2 SVM models which takes face trait feature vectors as input.

	Training accuracy	Testing accuracy	$\log_2(C)$
Governors	0.8462	0.5882	10
Senators	0.6000	0.7222	-1

Compared with the results listed in Table 3, all training and testing accuracies decrease. This worse performance may be due to the errors introduced by the 14 face traits extractors in the first stage.

3.3 Analysis of Results

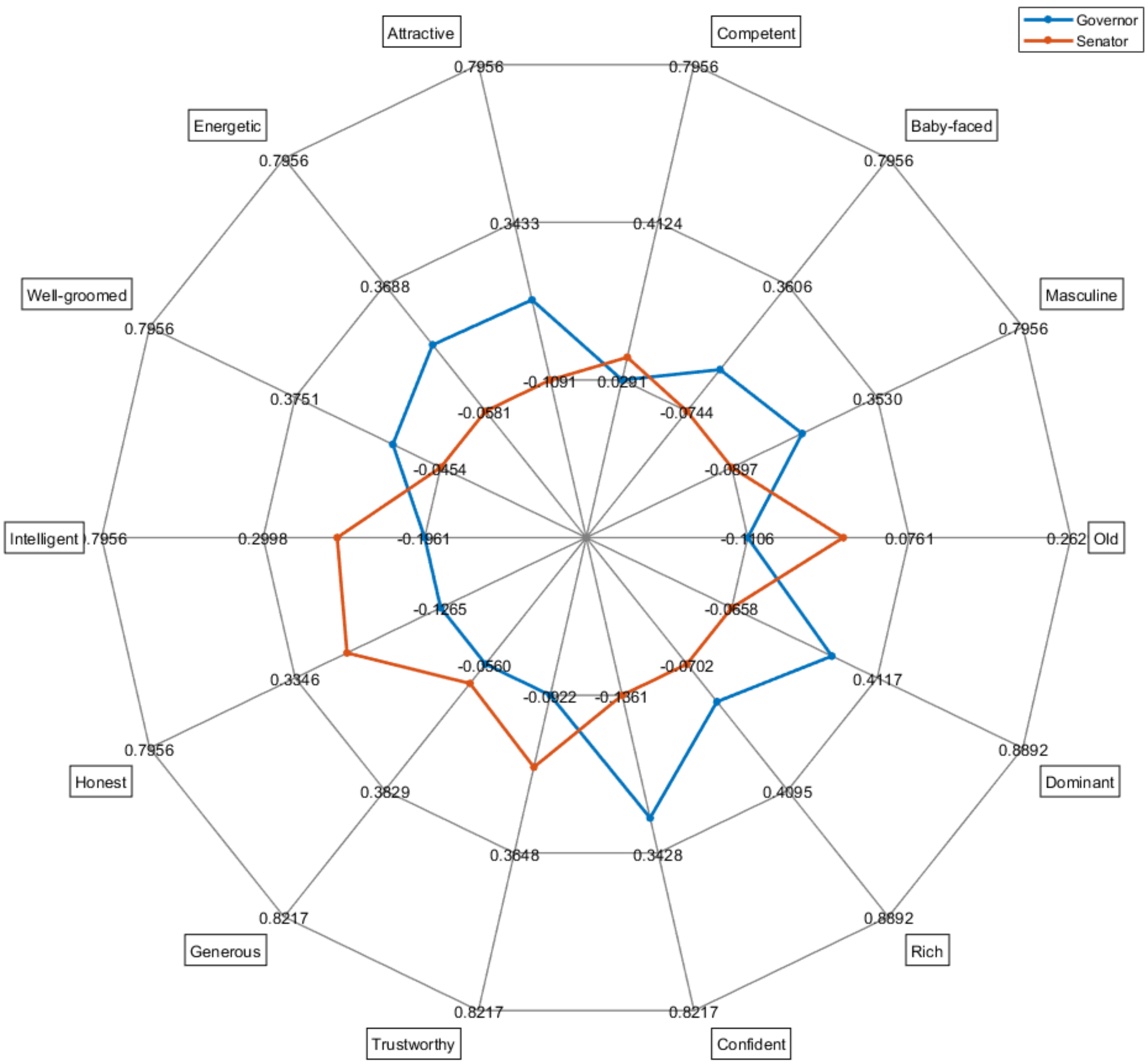

Figure 3: Radar plot of correlations between face traits and vote share differences

In order to understand which particular face trait contributes the most positive or negative effects to a success in election, I computed the correlation between each face trait difference and voting share difference. The radar plot above summarizes all these correlations clearly. It can be seen if a governor candidate wants to succeed in an election, he has to appear to be attractive, energetic, well-groomed, dominant, and confident. While appearing to be intelligent, honest, and old will possibly bring bad effects on election results. Similarly, to a senator candidate, he would better look more intelligent, honest, and old rather than dominant, rich, or attractive. Some explanations given by the radar plot actually quite make sense.

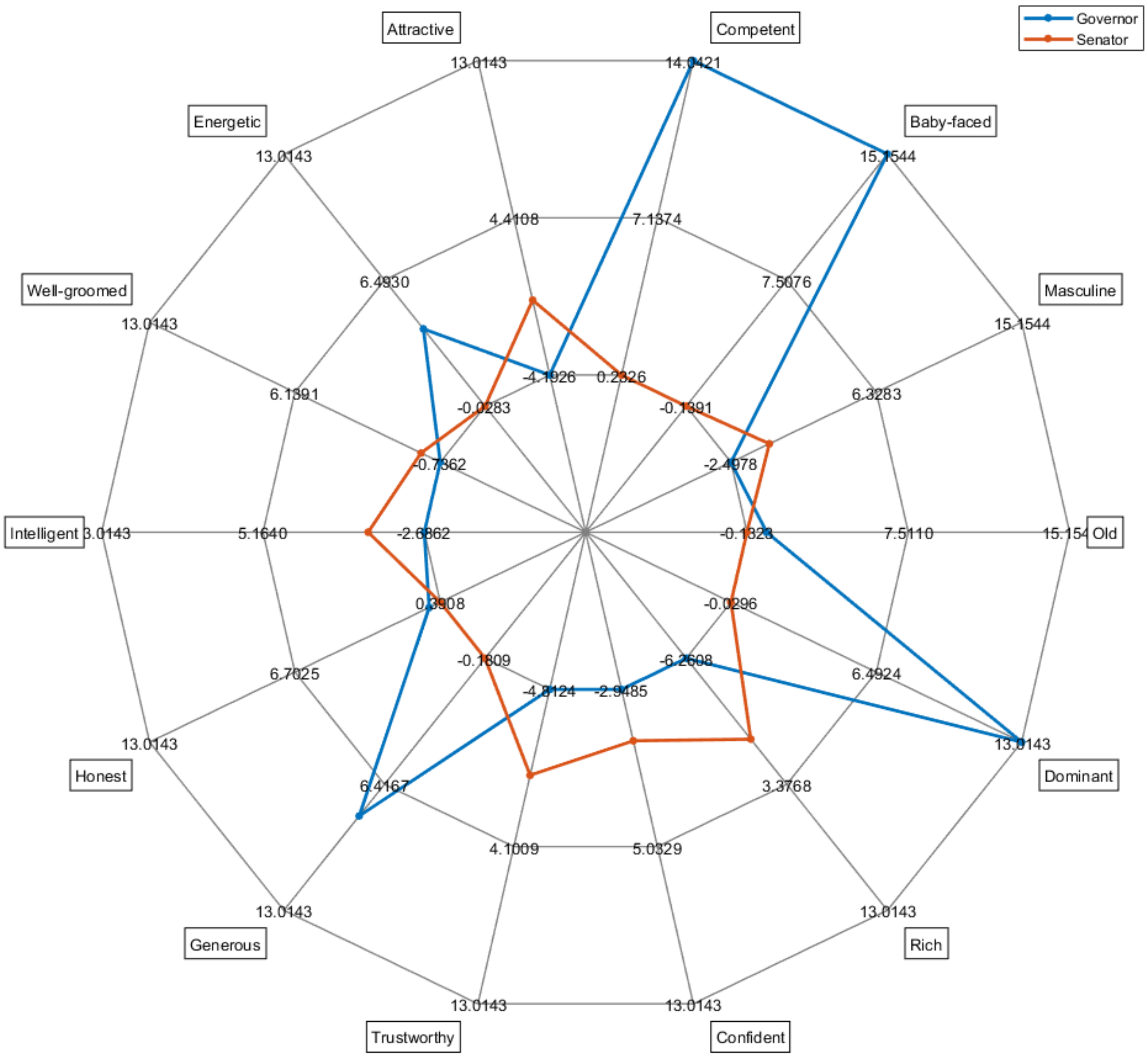

Figure 4: Radar plot of projection parameters ω in RankSVM

Finally, I have plotted the radar plot of the projection parameters derived in the two RankSVM models and hope to dig out some extra useful information. As you can see, some results fit with Figure 3 quite well while others don't appear so.