Political Election Analysis by SVM

1 Learning SVRs

1.1 Classification by Landmarks

We use the provided facial landmarks as features to train 14 SVRs. The landmarks of these 491 images are a 491×160 matrix. Each row represent 80 keypoint locations. We remove all those columns are the same for all images and also rescale them to the range [0,1], and then use function libsvmtrain() to get epsilon-SVR models by setting option as '-s 3'. We first use option '-v 5' to get the mse under different parameters, and then select the best parameters. Using the best parameters, we write our own k-fold cross-validation function to get the average accuracies and precisions on training and testing data each of the 14 models. The results of accuracies and selected parameters are shown below.

1.1.1 Average accuracies and precisions on training and testing data for each of the 14 models

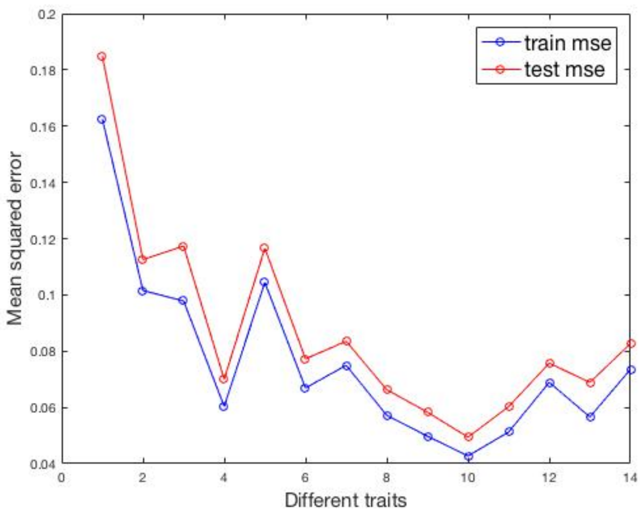

Figure 1: Average accuracies for each of the 14 models

Since we use linear kernel, there are 2 parameters we need to specify in epsilon-SVR model:(1) the parameter C, (2) the parameter epsilon for each trait. We summarize the resulting parameters for each of 14 trait in Figure 2.

1.1.2 The LIBSVM parameters of the 14 models

Trait Number	1	2	3	4	5	6	7	8	9	10	11	12	13	14
log2(C)	-3	-5	-3	-5	-5	-5	-5	-5	-5	-5	-5	-5	-3	-5
log2(epsilon)	-1	-3	-3	-3	-3	-3	-3	-3	-9	-7	-3	-3	-3	-3

Figure 2: Parameters of SVR models only using landmarks as features

1.2 Classification by Rich Features

In this question, we introduce HoG (Histogram of oriented gradient) features and repeat the steps in the above question to train and test espilon-SVR classifiers. We just concatenate two type of features: HoG features as well as rescaled landmarks into one matrix. Since most of the values of HoG features are in the range [0,1], we did not rescale the HoG features. (Note: I change the sbin parameter in the C++ script from 8 to 32 so that we can control the dimension of HoG features for each image and make it easy to compute) . The training and testing error are summarized in the Figure 3 below.

1.2.1 Average accuracies and precisions on training and testing data for each of the 14 models (Rich Features)

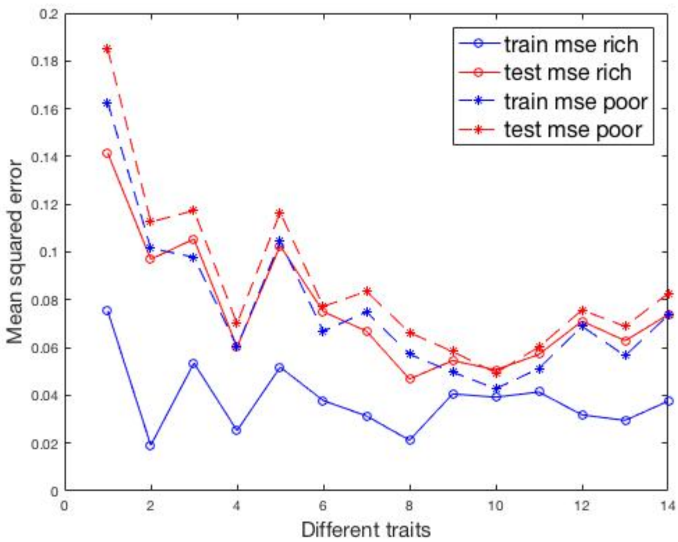

Figure 3: Average accuracies for each of the 14 models

Same as the above question, we also need to specify C and epsilon for each of the 14 models, and the result are summarized as below.

1.2.2 The LIBSVM parameters of the 14 models (Rich Features)

Trait Number	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***	***10***	***11***	***12***	***13***	***14***
log2(C)	−7	−5	−5	−7	−7	−7	−5	−5	−7	−7	−7	−7	−7	−7
log2(epsilon)	−3	−3	−3	−5	−9	−3	−3	−3	−5	−3	−5	−9	−3	−9

Figure 4: Parameters of SVR models using rescaled landmarks and HoG features

1.2.3 The names of the features you have used

Rescaled landmarks and HoG features of the images.

1.2.4 Comparsion to classification by landmarks (1.1)

From Figure 3 above, we can see the mse of rich features is lower than only using landmarks as features, which means adding HoG features make our model more accurate. No matter only using landmarks or using rich features, the training mse (blue) are lower the testing mse (red), meaning that model have a better performance on training than on testing dataset, which is also easy to understand. After adding HoG features, we can see although train mse poor have a relatively large decrease, the test mse do not have such large difference, and I think this may caused by overfitting.

2 Election Outcome Prediction

2.1 Direct Prediction by Rich Features

In this question, we also concatenate rescaled landmarks and HoG features as our features, and using the features to train a RankSVM to predict the election outcome. The RankSVM function is downloaded from URL:" olivier.chapelle.cc/primal/". The only parameter we need to specify is parameter C, we use grid search method to find the best C. We train the models for governor and senator separately and summarize the results as below.
(Notes:
1). For Senators, my training accuracy is 0.8566 under parameter $log_2 C = -9$, I checked that under other parameters the training accuracy is also near 1, but the testing accuracy is relative low, which may casue by overfitting. Thus I choose the parameter with highest testing accuracy
2). I construct my own functions my_feature_prep.m and myranksvm.m to process features and do k-fold cross validation model training.
)

2.1.1 Average accuracies on training and testing data as well as the chosen model parameters

	training acc	*testing acc*	*Log2(C)*
Governors	1	0.6607	-4
Senators	0.8566	0.6228	-9

Figure 5: Results of direct prediction by rich features

2.2 Prediction by Face Social Traits

In this question, we want to consider a two-layer-model to predict the election outcome. We first apply the rich features of Governors/Senators into the 14 models that we trained in section 1.2 to predict the scores of 14 social traits for each candidate (We implement for Governors and Senators seperately), and then use the social traits as features to predict the election outcome. We summarize our result as below.

2.2.1 Average accuracies on training and testing data as well as the chosen model parameters

	training acc	*testing acc*	*Log2(C)*
Governors	0.8036	0.5714	5
Senators	0.8561	0.6719	2

Figure 6: Results of direct prediction by Face Social Traits

2.2.2 Comparsion to direct prediction by rich features

Compared with the results in section 2.1, we found out the testing accuracy for senators is a little bit improved while the testing accuracy for governors becomes worse. However, the good thing is that this model is more interpretable.

2.3 Analysis of Results

In order to find out which social traits lead to the election victory, we caculate the correlation table between each social traits ($F_{win} - F_{lose}$) and absolute value of voting difference and get the correlation table as well as radar graphs as below.

Correlation	Governors	Senators
Old	0.01358	0.13667
Masculine	0.15393	0.02409
Baby-faced	0.00730	-0.04770
Competent	0.13890	0.17604
Attractive	0.29003	-0.03805
Energetic	0.17433	-0.16016
Well-groomed	0.34285	0.07864
Intelligent	-0.04791	0.15078
Honest	-0.16364	-0.02568
Generous	-0.05186	-0.08986
Trustworthy	-0.05345	-0.00589
Confident	0.24104	-0.12967
Rich	0.35904	0.13800
Dominant	0.16904	-0.05764

Figure 7: Correlation table

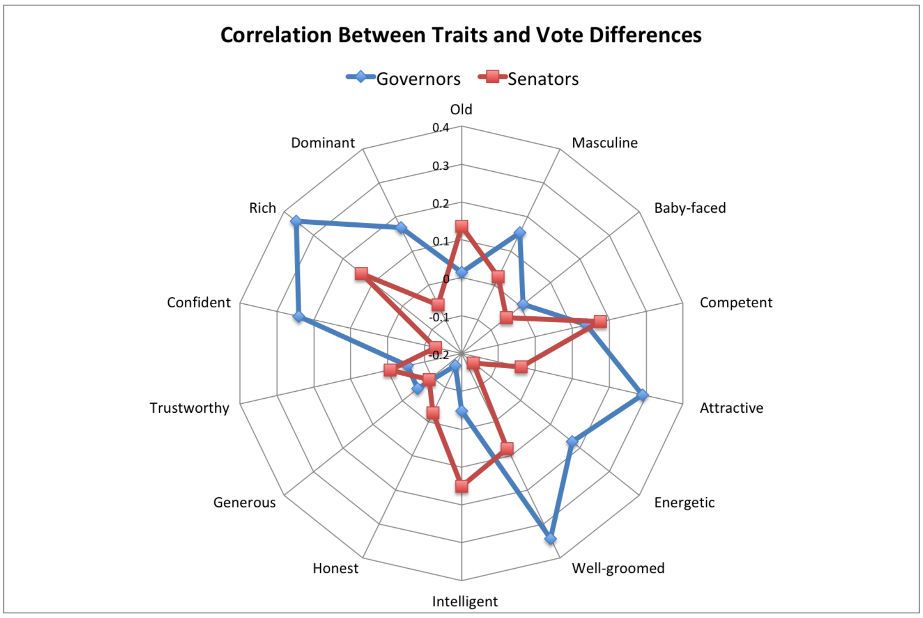

Figure 8: Radar graph of correlations

From the above results, we could summarize that rich, well-groomed and attractive have most positive effects for governors, while for senators, competent,intelligent and rich gave most positive effects. People do not like Baby-faced and Generous candidates.