Eric Fischer
SID: 303759361

Face Social Traits and Political Election Analysis by SVM

In this project, we follow the paper 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.

We exploit such low-level facial features and high-level concepts of perception to analyze election outcomes and party affiliations (GOP vs DEM) of politicians. The goal is to:

- train classifiers that can infer the perceived face social traits from low-level features, and
- apply the model to analyze the outcomes of real-world political elections.

Part 1: Face Social Traits Classification (or Regression)

1.1: Classification by Landmarks

I used the Scikit-learn library to train 14 Support Vector Regression models, one for each attribute dimension (e.g. old, masculine, etc.) of the training examples. I performed k-fold cross-validation with 5 folds and an 80/20 train/test split and used GridSearchCV from Scikit-learn to optimize the SVR parameters. The optimal **C, γ,** and **ε** (C, gamma, epsilon) values were searched for in the following ranges:

- $C \in [2^{-5}, -2^{-3}, \ldots, 2^{15}]$
- $\gamma \in [2^{-15}, 2^{-13}, \ldots, 2^{3}]$
- $\varepsilon \in [2^{-7}, 2^{-5}, 2^{-3}, 2^{-1}]$

The optimal **C, γ,** and **ε** (C, gamma, epsilon) values found with the Radial Basis Function (RBF) kernel are summarized below.

Trait	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$\log_2 C$	1	-1	5	-5	-3	-1	-5	-3	5	5	5	3	-5	5
$\log_2 \gamma$	-7	-3	-9	-3	-3	-5	-3	-3	-9	-9	-9	-9	-3	-11
$\log_2 \varepsilon$	-3	-3	-3	-3	-3	-3	-7	-3	-3	-7	-3	-3	-7	-3

Table 1: Optimal **C, γ,** and **ε**

After training, I applied the 14 learned classifiers to the test examples left out of training, and I measured performance (i.e. classification average accuracy and precision) of the classifiers. As I used a regression analysis for the trait models, to perform classification, the real-valued scores of the predicted test labels and the ground-truth test labels were converted to a [-1, 1] binary classification based on a threshold specific to each trait: the mean of the ground-truth label for that trait across the 491 images supplied as data.

The figures and tables that follow show the mean square error, average accuracy, and average precision for both training and testing data for each of the 14 SVR trait models.

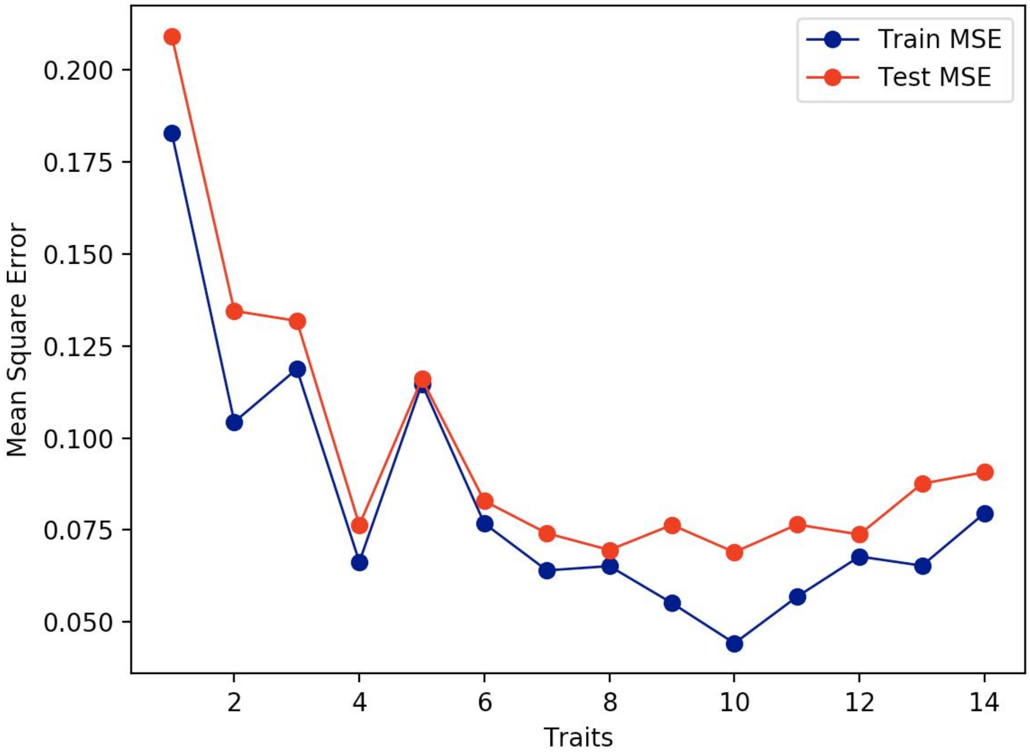

Figure 2: Mean square error (MSE) on training and testing data for 14 trait models

Trait	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Training MSE	.1828	.1043	.1187	.0663	.1147	.0768	.0840	.0652	.0551	.0442	.0568	.0778	.0653	.0796
Testing MSE	.2092	.1345	.1318	.0764	.1160	.0828	.0741	.0696	.0764	.0690	.0765	.0738	.0876	.0907

Table 3: Mean square error (MSE) values

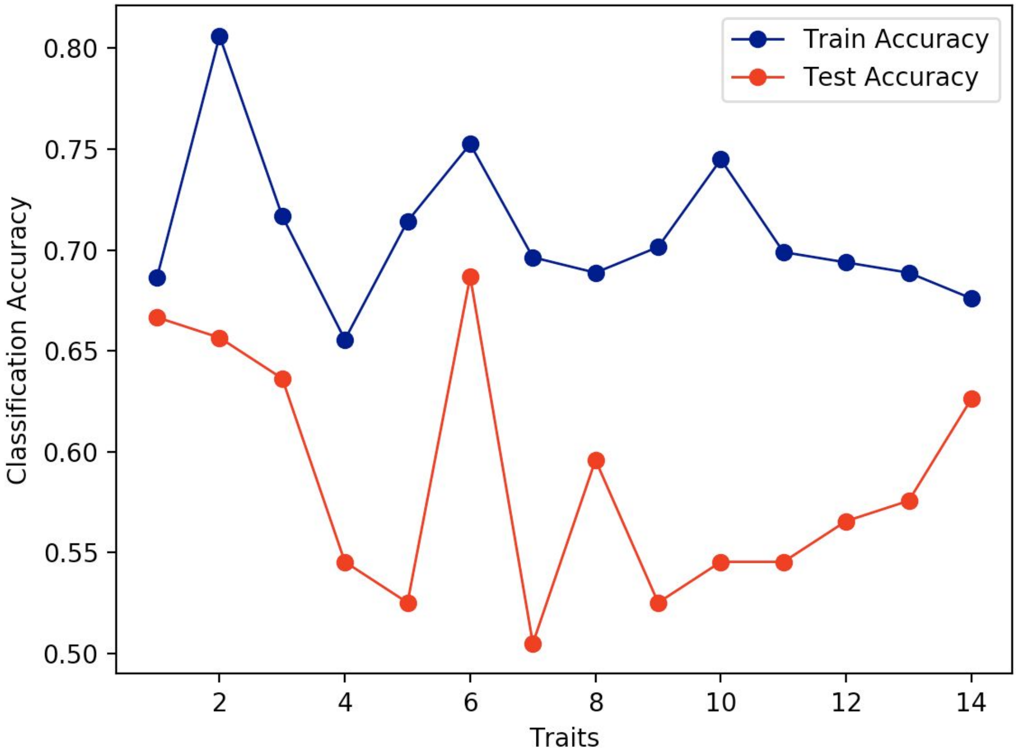

Figure 4: Average accuracies on training and testing data for 14 trait models

Trait	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Training Accuracy	.6862	.8061	.7168	.6556	.7143	.7526	.6964	.6888	.7015	.7449	.6990	.6939	.6888	.6760
Testing Accuracy	.6667	.6566	.6364	.5454	.5253	.6869	.5051	.5960	.5253	.5454	.5454	.5657	.5758	.6263

Table 5: Average accuracy values

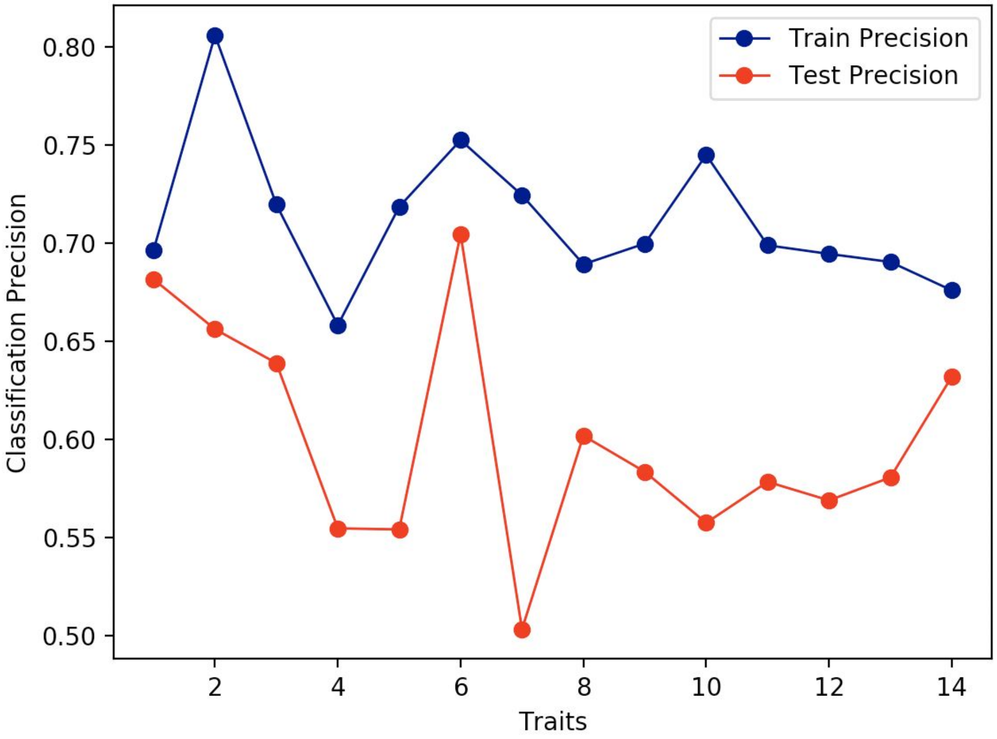

Figure 6: Average precisions on training and testing data for 14 trait models

Trait	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Training Precision	.6966	.8062	.7197	.6582	.7186	.7526	.7246	.6893	.6998	.7451	.6990	.6946	.6905	.6760
Testing Precision	.6816	.6562	.6388	.5545	.5540	.7047	.5030	.6018	.5834	.5573	.5783	.5688	.5805	.6320

Table 7: Average precision values

1.2: Classification by Rich Features

By concatenating for each of the 491 images the original landmark features and HoG (histogram of oriented gradient) features extracted from an image, and then retraining the 14 SVR trait models with these new features, we can observe improved training and testing accuracy and precision in almost all the models individually.

Again, I performed k-fold cross-validation with 5 folds and an 80/20 train/test split and used GridSearchCV from Scikit-learn to optimize the SVR parameters. The optimal **C, γ,** and **ε** (C, gamma, epsilon) values were searched for in the following range:

- $C \in [2^{-5}, -2^{-4}, \ldots, 2^{15}]$
- $\gamma \in [2^{-15}, 2^{-14}, \ldots, 2^{3}]$
- $\varepsilon \in [2^{-7}, 2^{-6}, \ldots, 2^{-1}]$

I adopted this slightly more refined grid search, checking each (as opposed to every other) exponential power of 2 within the listed ranges, as it lended better accuracy and precision on both training and testing data, but more importantly testing data. The optimal **C, γ,** and **ε** (C, gamma, epsilon) values found, again using the Radial Basis Function (RBF) kernel, are summarized below. Note that the greatly increased feature size per image (from 160 features, i.e. landmark coordinates, to 1960 features, i.e. landmark coordinates + 1800 hog features in this case) made the SVR models more prone to overfitting the training data. Thus, lower values of C were experimented with to increase regularization (i.e. increase bias, decrease variance) and secure optimal testing accuracy.

Trait	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$\log_2 C$	0	2	1	-1	1	0	1	0	0	6	6	0	2	2
$\log_2 \gamma$	-7	-6	-7	-5	-10	-7	-6	-6	-8	-15	-15	-9	-11	-11
$\log_2 \varepsilon$	-3	-4	-4	-4	-4	-4	-4	-5	-3	-7	-4	-5	-3	-4

Table 8: Optimal **C, γ,** and **ε**

The figures and tables that follow show the average accuracy and precision for both training and testing data for each of the 14 SVR rich trait models (trained with landmarks and HoG features), in comparison with the 14 SVR trait models trained with only landmark features.

All models have improved *training* accuracy and precision, and all but two models have improved *testing* accuracy and precision, when using HoG features in addition to landmarks. This demonstrates that HoG features improve the binary classification performance.

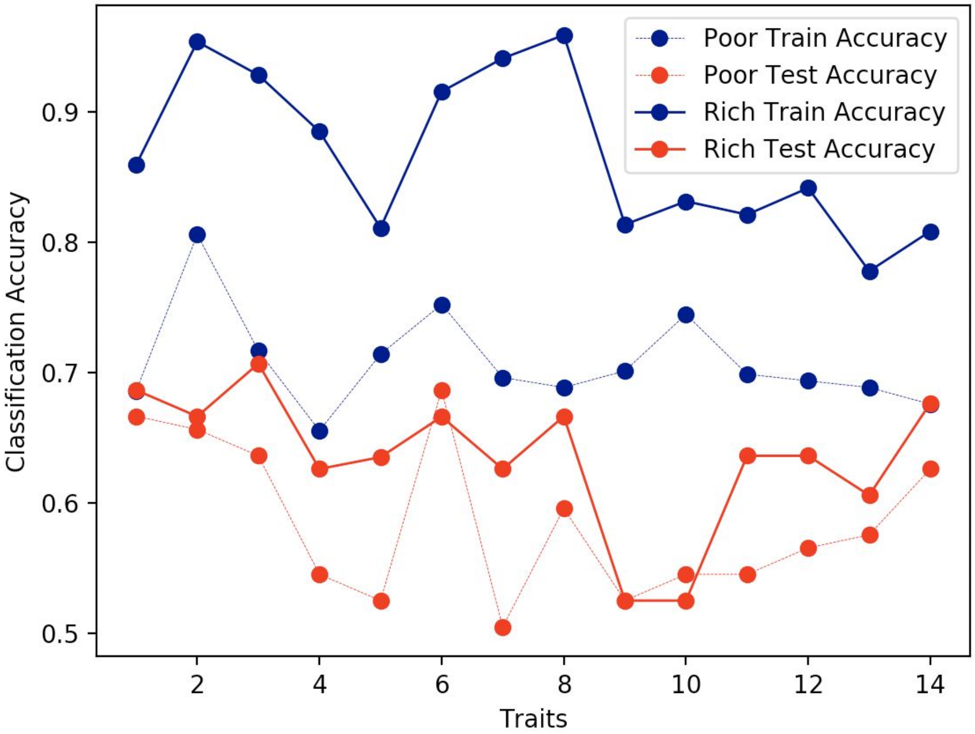

Figure 9: Average accuracies of 14 rich SVR models (trained with landmarks and hog features) compared to non-rich, or "poor", SVR models (trained with only landmarks)

Trait	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Train Accuracy	.6862	.8061	.7168	.6556	.7143	.7526	.6964	.6888	.7015	.7449	.6990	.6939	.6888	.6760
Test Accuracy	.6667	.6566	.6364	.5454	.5253	.6869	.5051	.5960	.5253	.5454	.5454	.5657	.5758	.6263
Rich Train Accuracy	.8597	.9541	.9286	.8852	.8112	.9158	.9413	.9592	.8138	.8316	.8214	.8418	.7781	.8087
Rich Test Accuracy	.6868	.6666	.7071	.6263	.6354	.6664	.6263	.6667	.5253	.5253	.6364	.6364	.6061	.6768

Table 10: Average accuracy values

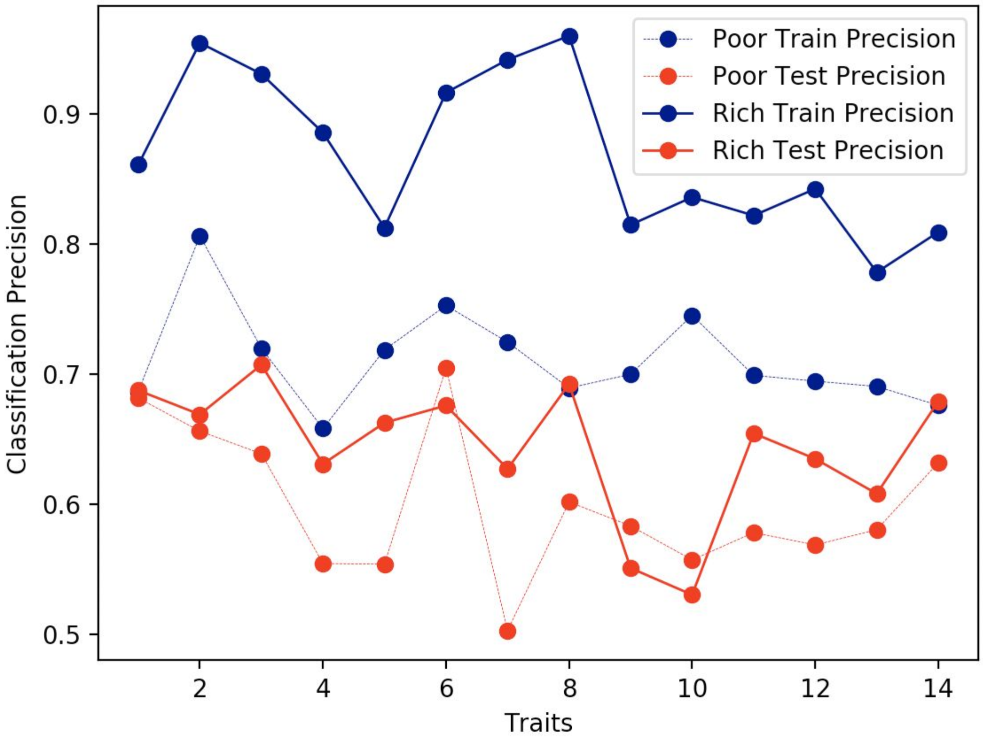

Figure 11: Average precisions of 14 rich SVR models (trained with landmarks and hog features) compared to non-rich, or "poor", SVR models (trained with only landmarks)

Trait	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Train Precision	.6966	.8062	.7197	.6582	.7186	.7526	.7246	.6893	.6998	.7451	.6990	.6946	.6905	.6760
Test Precision	.6816	.6562	.6388	.5545	.5540	.7047	.5030	.6018	.5834	.5573	.5783	.5688	.5805	.6320
Rich Train Precision	.8606	.9543	.9304	.8853	.8123	.9161	.9413	.9596	.8146	.8358	.8217	.8422	.7781	.8090
Rich Test Precision	.6875	.6690	.7071	.6310	.6626	.6759	.6272	.6923	.5511	.5305	.6545	.6349	.6081	.6790

Table 12: Average precision values

Part 2: Election Outcome Prediction

2.1: Direct Prediction by Rich Features

Again using rich features, or concatenated landmarks and hog features, I trained two LinearSVC models from Scikit-learn to predict the election outcomes for governor and senator races. I performed k-fold cross-validation with 5 folds and an 80/20 train/test split and only needed to optimize one LinearSVC parameter, C. I also disabled the learning parameter b, or the intercept, so that it would not be used in cost calculations. The cost C was searched for in the range C ∈ [2^{-15}, -2^{-14}, …, 2^{15}] and the optimal C was 2^{-2} for governors and senators. The average accuracies on training and testing data are shown below.

	$\log_2 C$	Training Accuracy	Testing Accuracy
Governors	-2	1	.6522
Senators	-2	1	.7084

Table 13: Optimal C and training and testing average accuracies for predicting election outcomes

2.2: Prediction by Face Social Traits

In a two-layer model to predict election outcomes, we can use the real-valued outputs of the 14 rich SVR models (trained with landmarks and HoG features) and project these 14-dimensional vectors, each representing an image, into a feature space in which we predict the election outcomes between two candidates by performing an SVCLinear binary classification of these vectors. Note that I defined two politicians as one data point by subtracting a trait feature vector A from another B and then training the SVCLinear binary classifier with the value $\boldsymbol{F_{AB} = F_A - F_B}$.

Then we can compare this two-layer election prediction by face social traits to the previous direct prediction by rich features described above.

Here for the first layer, I used my previously constructed 14 rich SVR models for each attribute dimension (e.g. old, rich, etc.). Hence the chosen model parameters for these models are displayed above in section *1.2: Classification by Rich Features*. For the second SVCLinear layer, I again performed k-fold cross-validation with 5 folds and a 80/20 train/test split, and the C parameter was searched for in the range C ∈ [2^{-15}, -2^{-14}, …, 2^{15}], with the optimal C being 2^{-4} for governors and senators. I also disabled the learning parameter b, or the intercept, so that it would not be used in cost calculations. The average accuracies on training and testing data are shown below.

	$\log_2 C$	Training Accuracy	Testing Accuracy
Governors	-4	.7191	.6087
Senators	-4	.6739	.5417

Table 14: Optimal C, γ, and ε and training and testing average accuracies

In comparison with direct prediction by rich features, it was clear that the two-layer prediction by social traits had slightly worse mean testing accuracies for governors and senators. The mean testing accuracy for direct prediction by rich features was .6522 and .7084 for governors and senators, but the two-layer prediction by face social traits averaged lower .6087 and .5417 testing accuracies for governors and senators. This supports the notion that, for predicting election results with this quantity and quality of data, landmarks and HoG features are likely better feature descriptors for election candidates than the outputs of the 14 rich SVR trait models.

2.3: Analysis of Results

We are in a good position now to observe correlations between facial attributes and election outcomes. We can answer the question: which facial attributes truly lead to electoral success?

To answer this question, we can derive correlation coefficients between each of the facial attributes and the desired outcome of winning the election. Note that larger correlations tending toward 1 indicate a facial attribute and electoral success are strongly correlated, while smaller correlations tending toward -1 indicate they are less correlated.

As we can see below, the traits *Rich, Well-groomed, Attractive,* and *Confident* are more correlated with success for governors while the traits *Intelligent, Competent, Well-groomed,* and *Rich* are more correlated with success for senators.

Trait Num.	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Trait	Old	Masculine	Baby-faced	Competent	Attractive	Energetic	Well-groomed	Intelligent	Honest	Generous	Trustworthy	Confident	Rich	Dominant
Governors	-.0568	.1699	.0080	.0419	.2613	.1563	.2804	.0306	-.1976	-.2239	-.0747	.2378	.2974	.1193
Senators	.0422	-.0215	-.1333	.1138	.0234	-.0427	.1033	.1985	-.0350	-.2054	-.0920	-.0645	.0904	.0218

Table 15: Correlations between facial attributes and electoral success