

# Letter Classification

Matt Oehler and Jackson Curtis

Stat 536

April 13, 2018

PLACE OF BIRTH				NAME		RELATION		BIRTH DATA		PERSONAL DESCRIPTION		OCCUPATION		PLACE OF BIRTH		OTHER NAMES OR NAMES LOCATED BY THE FBI		CITIZENSHIP, ETC.		OCCUPATION AND RESIDENCY		EDUCATION		VETERAN			
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26		
Year	Month	Day	Year	Month	Day	Year	Month	Day	Year	Month	Day	Year	Month	Day	Year	Month	Day	Year	Month	Day	Year	Month	Day	Year	Month	Day	
1926	12	13	1931	Young, Edward H.	Head	P	30	M	45	M	64	W	64	W	64	W	64	W	64	W	64	W	64	W	64	W	
				Edwards, Eleanor	Wife H			F	44	39	M	54	W	54	W	54	W	54	W	54	W	54	W	54	W	54	W
				Wardner, William	Daughter			F	44	14	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
1910	12	13	1936	Bishop, James Earl, Samuel	Head H	P	270	F	44	14	M	54	W	54	W	54	W	54	W	54	W	54	W	54	W	54	W
				Black, Robert	Wife H			F	44	14	M	54	W	54	W	54	W	54	W	54	W	54	W	54	W	54	W
				Black, Robert	Daughter			F	44	14	M	54	W	54	W	54	W	54	W	54	W	54	W	54	W	54	W
				Black, Robert	Wife H			F	44	14	M	54	W	54	W	54	W	54	W	54	W	54	W	54	W	54	W
1911	18	13	1937	Tricker, Leslie	Head H	P	24	M	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
				Tricker, Mary	Wife H			F	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
				Tricker, William	Daughter			F	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
1918	13	13	1938	Boyle, Thomas	Head H	P	24	M	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
				Boyle, Thomas	Wife H			F	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
				Boyle, Thomas	Daughter			F	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
1916	13	13	1938	Thompson, John	Head H	P	270	M	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
				Thompson, John	Wife H			F	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
				Thompson, John	Daughter			F	44	12	S	44	W	44	W	44	W	44	W	44	W	44	W	44	W	44	W
1918	13	13	1938	Wright, Robert	Head H	P	24	M	44	12	S	44	W	44	W	44	W	44	W	44</							

# Goals

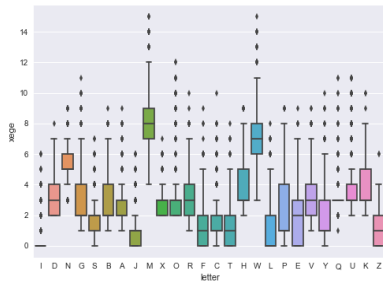
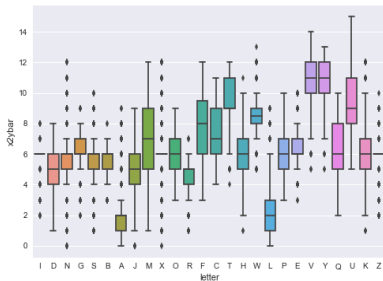
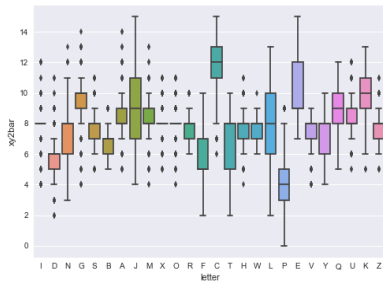
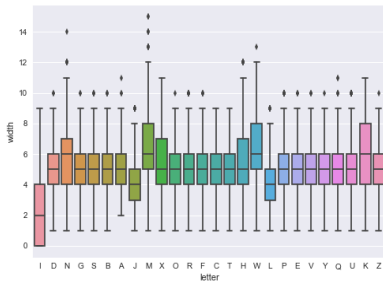
## Goals of the Analysis:

- ▶ Use the data of letter characteristics to create a model that can accurately classify handwritten letters
- ▶ Automate document transcription by using the model to predict letters given their characteristics

## Data Description

	Variable Name	Description
1	letter	capital letter (26 values from A to Z)
2	xbox	horizontal position of box (integer)
3	ybox	vertical position of box (integer)
4	width	width of box (integer)
5	high	height of box (integer)
6	pix	total # of pixels (integer)
7	xbar	mean x of pixels in box (integer)
8	ybar	mean y of pixels in box (integer)
9	x2bar	mean x variance (integer)
10	y2bar	mean y variance (integer)
11	xybar	mean x y correlation (integer)
12	x2ybr	mean of $x * x * y$ (integer)
13	xy2br	mean of $x * y * y$ (integer)
14	xege	mean edge count left to right (integer)
15	xegvy	correlation of x-ege with y (integer)
16	yege	mean edge count bottom to top (integer)
17	yegvx	correlation of y-ege with x (integer)

# Data Description



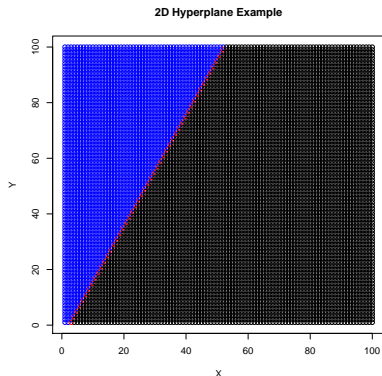
# Model Selection

We explored using Support Vector Machines and Random Forests, but found that Support Vector Machines seemed to perform better. Some advantages of SVMs are:

- ▶ They perform well in high-dimensional space (we can use all the variables)
- ▶ They are computationally efficient
- ▶ They are flexible according to the parameters that you choose to define them (e.g. kernel functions)

# Intro To Support Vector Machines

Support vector machines are based on the idea of dividing hyperplanes.



- ▶ Red points (the hyperplane) are where
$$0 = 50 + 10y - 20x$$
- ▶ Blue is where
$$0 < 50 + 10y - 20x$$
- ▶ Black is where
$$0 > 50 + 10y - 20x$$

In our case, instead of 2 dimensions, we will define a plane in 16-dimensional hyperspace.

## Choosing a Hyperplane

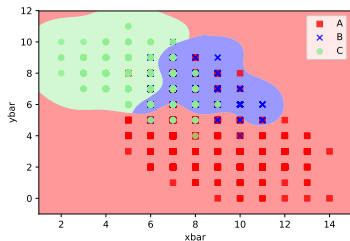
Calculate the hyperplane that separates the two classes perfectly and maximizes the minimum distance between the points and the line (this distance is called the margin).

What if we can't get perfect separation?

- ▶ Introduce slack variables
- ▶ Observations with slack greater than 0 are allowed to lie close to or on the wrong side of the plane, but we penalize each slack variable based on its distance from the margin of our SVM.
- ▶ We use a tuning parameter to set the biggest penalty we're willing to accept.



# Extending to non-linear planes



- ▶ SVMs won't work if we limit ourselves to linear planes
- ▶ Basis functions allow us to remap our explanatory variables so that the result is linearly separable
- ▶ Common basis functions include polynomial, radial, and neural nets
- ▶ Small values of the tuning parameter will increase the non-linearity of the hyperplanes.

## Extending to multiple classes

SVM planes split one class from another, but what about when we have 26 classes?

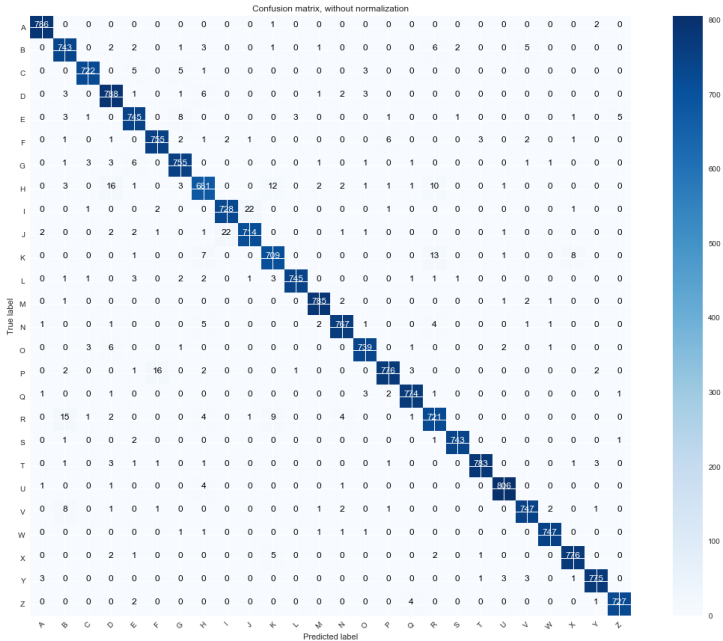
- ▶ Build a SVM using A and B data only, then A vs. C, etc
- ▶ This builds  $\binom{k}{2}$  SVMs, 325 in our case.
- ▶ Tally the 325 votes, majority becomes final prediction

# Parameter tuning

We performed a grid search and looked at overall accuracy for various parameter possibilities to choose:

- ▶ Kernel (we chose radial)
- ▶ Tuning parameter (11 worked best)
- ▶ Gamma—Precision of radial Gaussian distribution (setting it to auto,  $1/\#$  features, worked best).

## Model Performance



# Overall Accuracy

Method	Out of Sample Accuracy
Random Forest	96.70%
Support Vector Machine	97.77%

**Table 1:** SVMs slightly outperform Random Forests with 1000 trees and max 3 features

## Letter Ranking

Accuracy	Letter		
0.9987	A	0.9801	O
0.9947	S	0.9776	D
0.9926	U	0.9766	E
0.9920	W	0.9764	V
0.9911	Y	0.9745	N
0.9905	Z	0.9726	B
0.9886	X	0.9652	F
0.9885	Q	0.9639	J
0.9874	M	0.9621	K
0.9862	T	0.9617	R
0.9842	L	0.9616	I
0.9823	C	0.9614	P
0.9819	G	0.9319	H

Table 2: Ranking of which letters are easiest to predict

## Variable Importance

SVM variable importance is tough, but a rough idea can be given by looking at how far accuracy drops when the model is built and tested without that variable.

Top Five Most Important	Top Five Least Important
1. yege 2. x2bar 3. y2bar 4. xybar 5. x2ybar	1. xbox 2. width 3. xbar 4. high 5. pix

# Conclusions

## **Goals:**

- ▶ We accomplished our goals by creating a model that can accurately predict a letter given its characteristics

## **Shortcomings:**

- ▶ Since we used an SVM, the results aren't interpretable, but that's ok since we are more interested in prediction
- ▶ Since we are working in a high-dimensional space, there isn't a good way to visualize the SVM and its decision boundaries



## Future Work

- ▶ Results could be improved by ensembling several different approaches.
- ▶ Added features would probably help SVM (does well in high dimensions)
- ▶ At the extreme, neural nets could be used on pixel by pixel information