Machine Learning

Image recognition

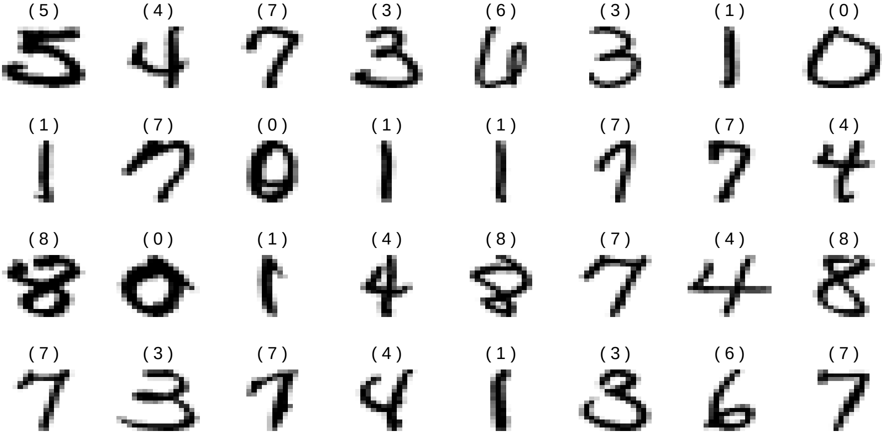
Sebastian Engelke

Master in Business Analytics

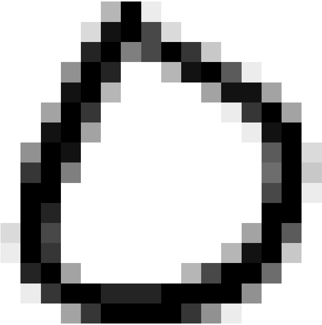
UNIVERSITÉ DE GENÈVE

Example: handwritten digit classification

Dataset from [ESL] webpage. Normalized handwritten digits (16×16 grayscale images), automatically scanned from envelopes by the U.S. Postal Service. Observations consists of the digit $y_i \in \{0, \ldots, 9\}$ and the $p = 256$ grayscale values $x_i = (x_{i1}, \ldots, x_{i256})$. The training and test sets contain 7290 and 2006 examples, respectively.

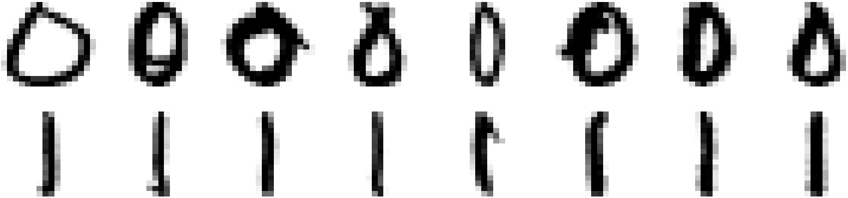


Digital black-white images

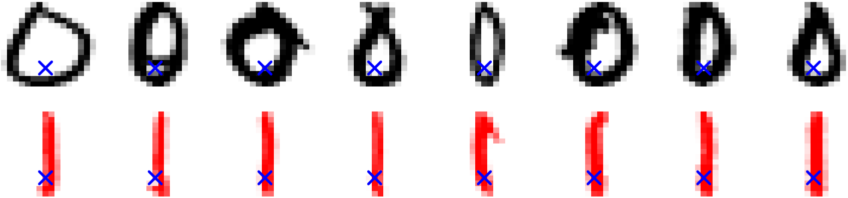


0	0	0	0	0	70	240	33	0	0	0	0	0	0	0	0
0	0	0	0	36	230	255	181	43	0	0	0	0	0	0	0
0	0	0	3	218	255	131	180	248	199	59	0	0	0	0	0
0	0	0	112	255	212	12	4	82	230	253	169	26	0	0	0
0	0	8	225	255	81	0	0	0	11	95	234	227	99	1	0
0	0	95	255	196	2	0	0	0	0	0	21	198	255	91	0
0	8	235	255	94	0	0	0	0	0	0	0	22	235	244	7
0	109	255	225	3	0	0	0	0	0	0	0	0	157	255	49
0	200	255	122	0	0	0	0	0	0	0	0	0	150	255	55
10	247	252	17	0	0	0	0	0	0	0	0	0	173	255	28
17	253	211	1	0	0	0	0	0	0	0	0	8	239	239	5
51	255	180	0	0	0	0	0	0	0	0	0	92	255	162	0
28	254	204	0	0	0	0	0	0	0	4	72	230	253	60	0
0	208	244	88	15	0	0	0	13	85	176	255	253	152	0	0
0	26	194	255	252	206	206	206	240	255	255	241	116	0	0	0
0	0	6	91	190	255	255	255	255	192	105	24	0	0	0	0

Digital black-white images

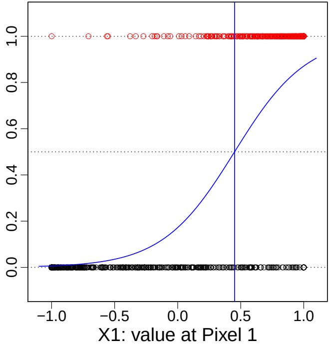


Two-class classification: one predictor

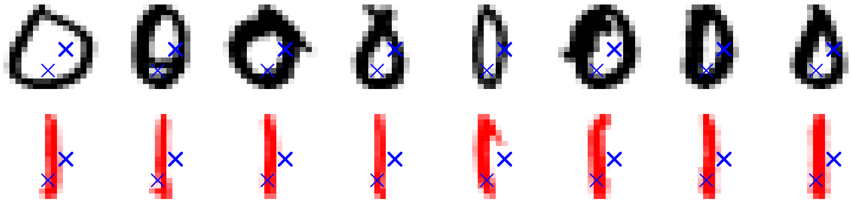


- Value at Pixel 1 (blue cross) is the predictor
  $X_1 =$ "value at Pixel 1" $\in [-1, 1]$
- The class is $Y \in \{0, 1\} = \{\texttt{zero}, \texttt{one}\}$
- We apply logistic regression to obtain a `classifier` based only on Pixel 1:

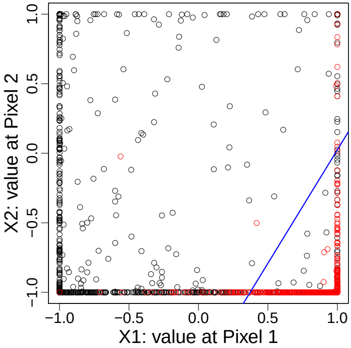
$$\widehat{G}(x) = \begin{cases} \texttt{zero} & \text{if (Pixel 1)} < 0.45 \\ \texttt{one} & \text{otherwise.} \end{cases}$$



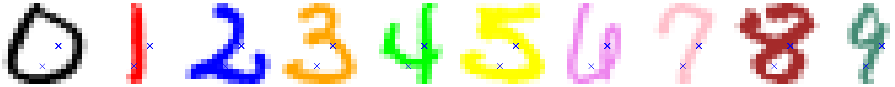
Two-class classification: two predictors



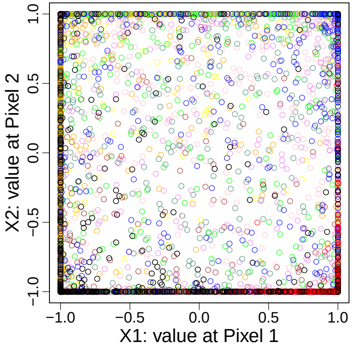
- $p = 2$: Predictors are
  X_1 = "value at Pixel 1" $\in [-1, 1]$
  X_2 = "value at Pixel 2" $\in [-1, 1]$
- The class is $Y \in \{0, 1\} = \{\texttt{zero}, \texttt{one}\}$
- We apply logistic regression to obtain a `classifier` based only on X_1 and X_2.



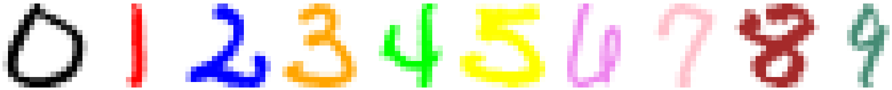
10-class classification: two predictors



- $p = 2$: Predictors are
  X_1 = "value at Pixel 1" $\in [-1, 1]$
  X_2 = "value at Pixel 2" $\in [-1, 1]$
- The class is
  $Y \in \{0, 1, \ldots, 9\} = \{\texttt{zero}, \texttt{one}, \ldots, \texttt{nine}\}$
- We can apply `multinomial regression`, based on X_1 and X_2, but with only two pixels this a very hard task!
- Luckily we have $p = 256$ predictors (pixel values) to perform this task...



Digit classification



- **Training data**: $(x_1, y_1), \ldots, (x_n, y_n)$, $n = 7290$, are used for model fitting/selection, where
  - predictors $x_i = \texttt{i-th image} \in \mathbb{R}^p$ are the grayscale values of the $p = 256$ pixels;
  - outcomes $y_i = \texttt{i-th digit} \in \{\texttt{zero}, \texttt{one}, \ldots, \texttt{nine}\}$.
- **Test data**: $(\tilde{x}_1, \tilde{y}_1), \ldots, (\tilde{x}_m, \tilde{y}_m)$, $m = 2006$, are not used for model fitting/selection, but only to evaluate the performance afterwards.

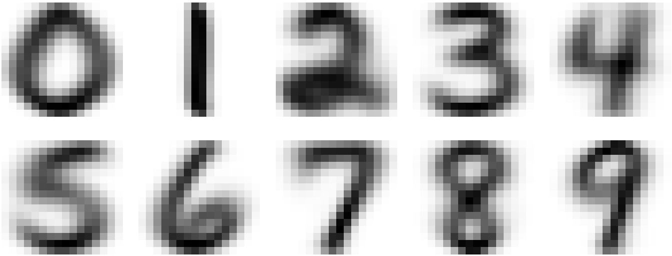
- linreg: direct linear regression
- LDA1: LDA based on the values x_i
- LDA2: LDA based on x_i and x_i^2
- QDA: different cov. matrix for each class
- log: logistic regression (multinomial)

	Training error	Test error
linreg	7.6%	13.1%
LDA1	6.2%	11.5%
LDA2	3.9%	10.2%
QDA	1.8%	13.5%
log	0.01%	11.1%

- QDA is worst of the four methods since its variance is too high (overfitting).

Digit classification: visualization of LDA1

- Figure shows estimates $\widehat{\mu}_1, \ldots, \widehat{\mu}_q$ for LDA1.
- This gives an impression of the centers of the classes, i.e., how American people write in ``average'' the 10 digits.

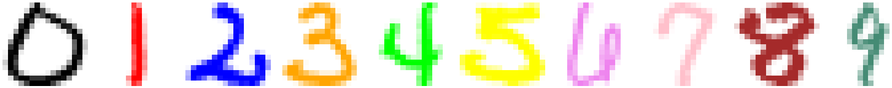


Example: digit classification

The table shows some misclassified test data from LDA1. The 10 columns on the right give the `predicted class probabilities` (in %) for $k \in \{0, \ldots, 9\}$.

Obs		0	1	2	3	4	5	6	7	8	9
	(6)	59	0	0	0	0	0	40	0	0	0
	(2)	94	0	0	6	0	0	0	0	0	0
	(3)	0	0	0	11	0	0	0	0	89	0
	(3)	0	0	0	1	0	99	0	0	0	0
	(8)	1	0	0	81	1	11	0	0	5	0
	(4)	0	0	0	0	15	0	0	0	0	85
	(9)	0	0	0	0	0	0	0	100	0	0
	(2)	100	0	0	0	0	0	0	0	0	0

Digit classification with regularization



- We can update our results on the `digit classification`.
- When estimating the logistic mulitnomial regression, we introduce a `ridge penality` and a `lasso penality` in the likelihood.
- The value of $\widehat{\lambda}$ for the tuning parameters were found by 5-fold CV.

- linreg: direct linear regression;
- LDA1: LDA based on the values x_i;
- LDA2: LDA based on x_i and x_i^2;
- QDA: different cov. matrix for each class;
- log: logistic regression (multinomial).
- log-ridge: logistic regression (multinomial) with ridge penality.
- log-lasso: logistic regression (multinomial) with lasso penality.

	Training error	Test error
linreg	7.6%	13.1%
LDA1	6.2%	11.5%
LDA2	3.9%	10.2%
QDA	1.8%	13.5%
log	0.01%	11.1%
log-ridge	4.1%	9.0%
log-lasso	2.7%	8.8%