AI – Machine Learning

Artificial Intelligence
Research Group

Spam and email

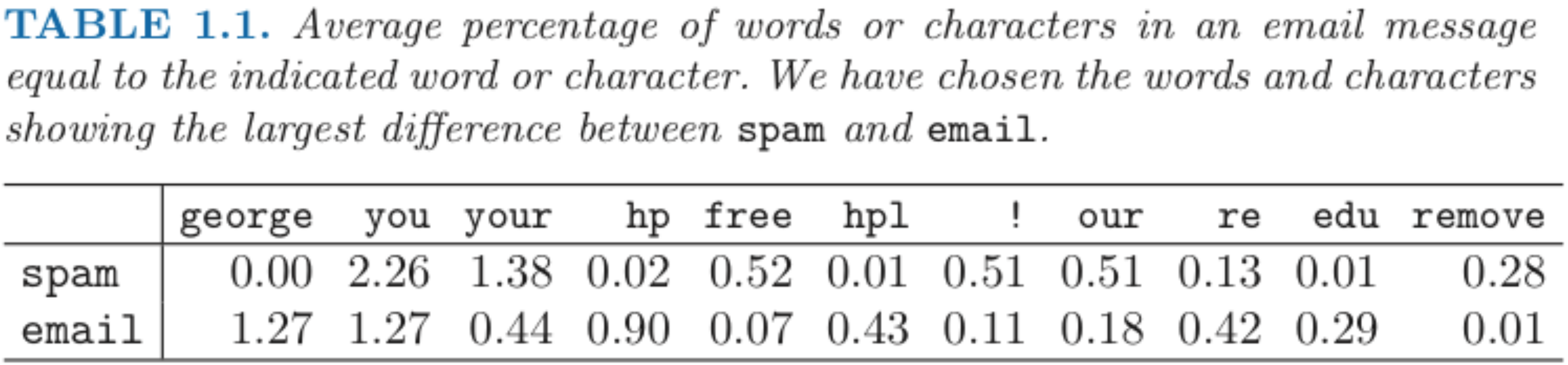


TABLE 1.1. *Average percentage of words or characters in an email message equal to the indicated word or character. We have chosen the words and characters showing the largest difference between* `spam` *and* `email`.

	george	you	your	hp	free	hpl	!	our	re	edu	remove
spam	0.00	2.26	1.38	0.02	0.52	0.01	0.51	0.51	0.13	0.01	0.28
email	1.27	1.27	0.44	0.90	0.07	0.43	0.11	0.18	0.42	0.29	0.01

Classification

Prostate cancer

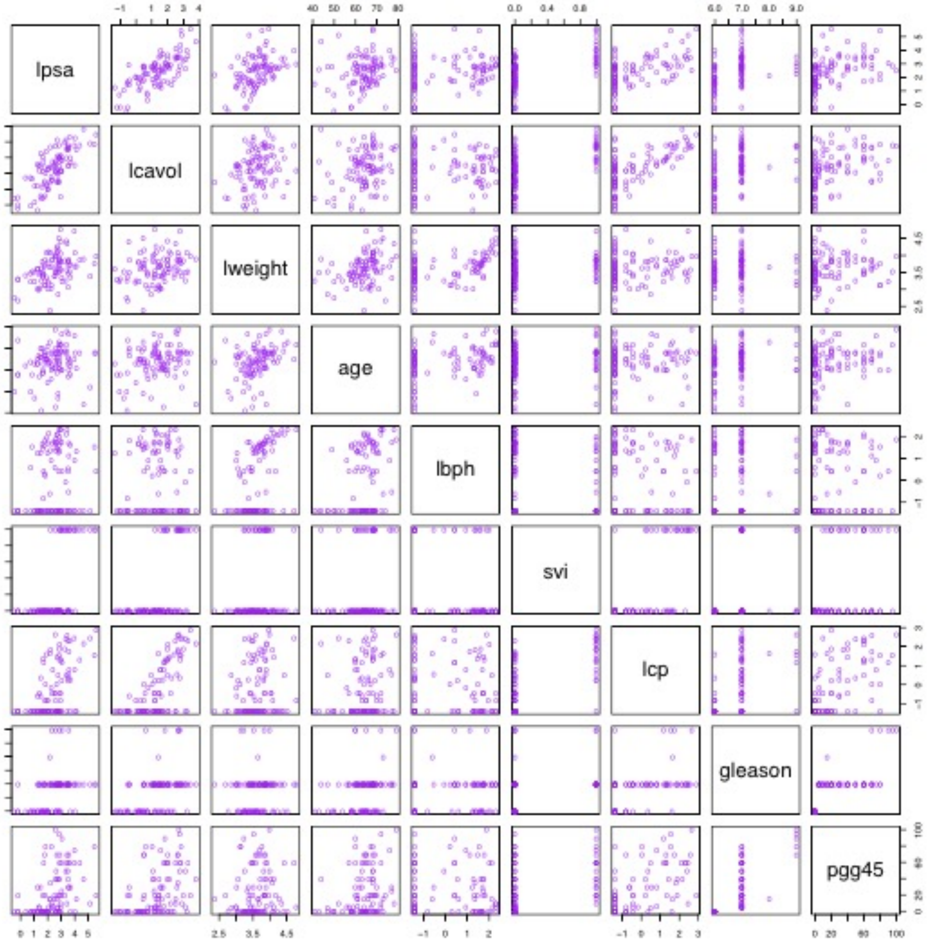


FIGURE 1.1. *Scatterplot matrix of the prostate cancer data. The first row shows the response against each of the predictors in turn. Two of the predictors,* `svi` *and* `gleason`*, are categorical.*

The data for this example come from a study by Stamey et al. (1989) that examined the correlation between the level of prostate specific antigen (PSA) and a number of clinical measures.

Regression

Handwritten digit recognition

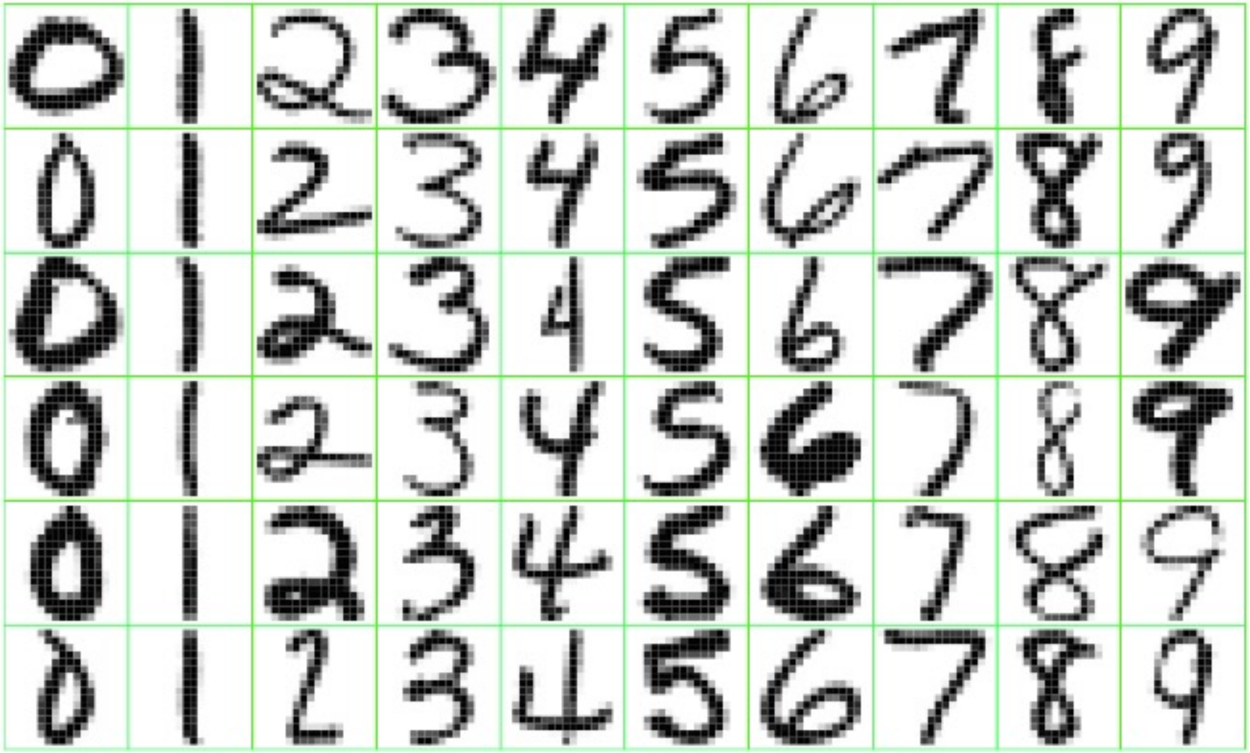
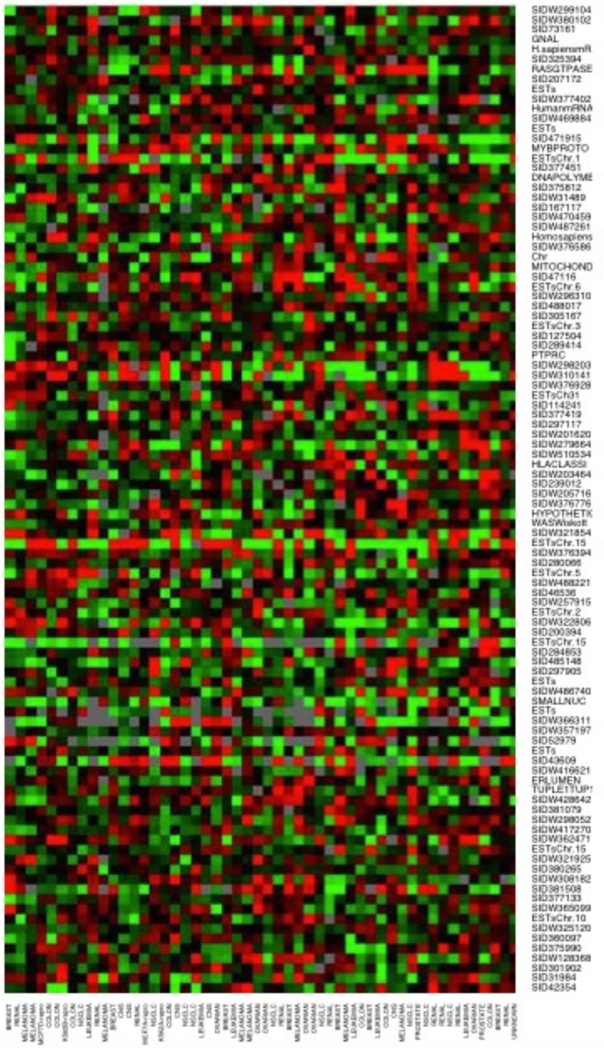


FIGURE 1.2. *Examples of handwritten digits from U.S. postal envelopes.*

Classification problem for which the error rate needs to be kept very low to avoid misdirection of mail.

DNA expression microarrays



Unsupervised Learning

- Which samples are most similar to each other, in terms of their expression profiles across genes?
- Which genes are most similar to each other, in terms of their expression profiles across samples?
- Do certain genes show very high (or low) expression for certain cancer samples?

Customized medical care

Function approximation

$X = \mathbb{R}^p, Y \in \mathbb{R}$

$f(x) = \mathrm{E}(Y|X = x)$

Linear Model

$f(x) = x^T \beta$

Dummy variables (for classification)

A K-level qualitative variable is represented by a vector of K binary variables or bits, only one of which is "on" at a time.

How to evaluate models?

Expected (squared) Prediction Error (EPE)

$$\begin{aligned} EPE(f) &= E(Y - f(X))^2 \\ &= \int [y - f(x)]^2 \Pr(dx, dy) \end{aligned}$$

The solution of minimize EPE is

$f(x) = \mathrm{E}(Y|X = x)$

Linear basis expansions

$$EPE(f) = E(Y - f(X))^2$$

$$f(x) = x^T\beta$$

Plugging the linear model for $f(x)$ into EPE and differentiating we can solve theoretically:

$$\beta = [E(XX^T)]^{-1}E(XY)$$

Linear Basis Expansions

$$f_\theta(x) = \sum_{k=1}^{K} h_k(x)\theta_k$$

Statistical research

Other models?

Sure. For instance, ***K-nearest-neighbor****.*

Other criteria?

Yes. For example, ***Maximum likelihood****.*

Other parameter estimation techniques?

Many. ***Iterative method, numerical optimization, etc.***

Nearest-neighbor methods

Nearest-neighbor methods use those observations in the training set closest in input space to x to form its output.

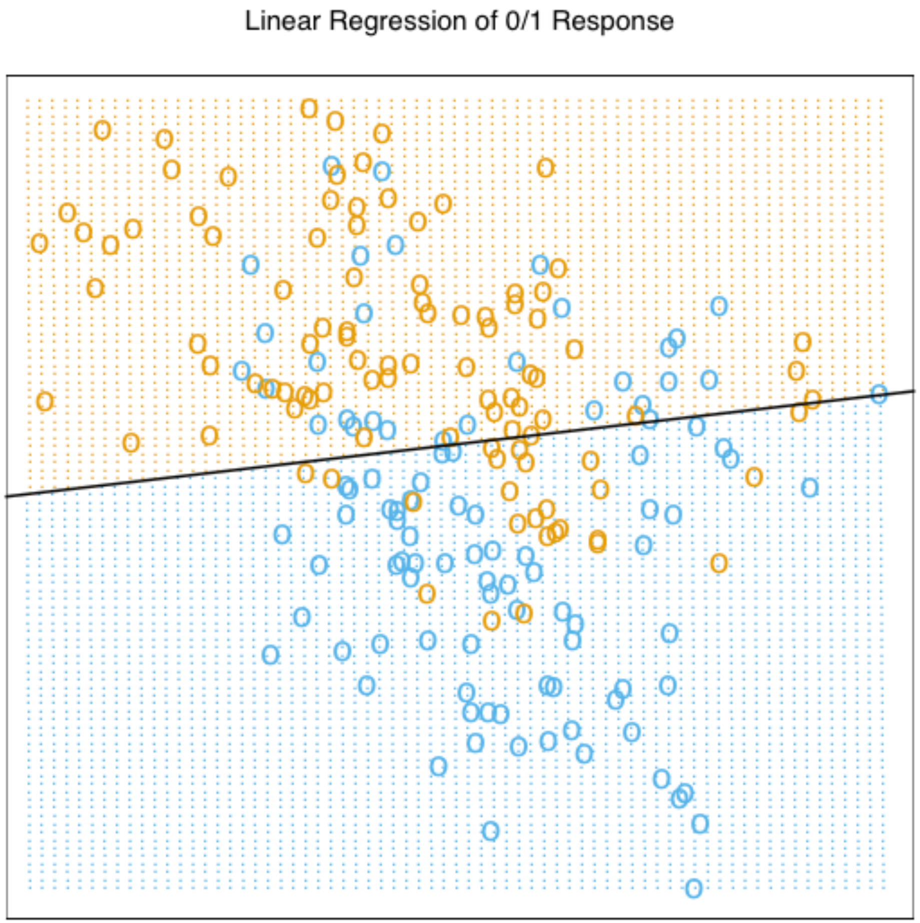
$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i$$

$$\hat{f}(x) = Ave(y_i \mid x_i \in N_k(x))$$

Two approximations are happening here:

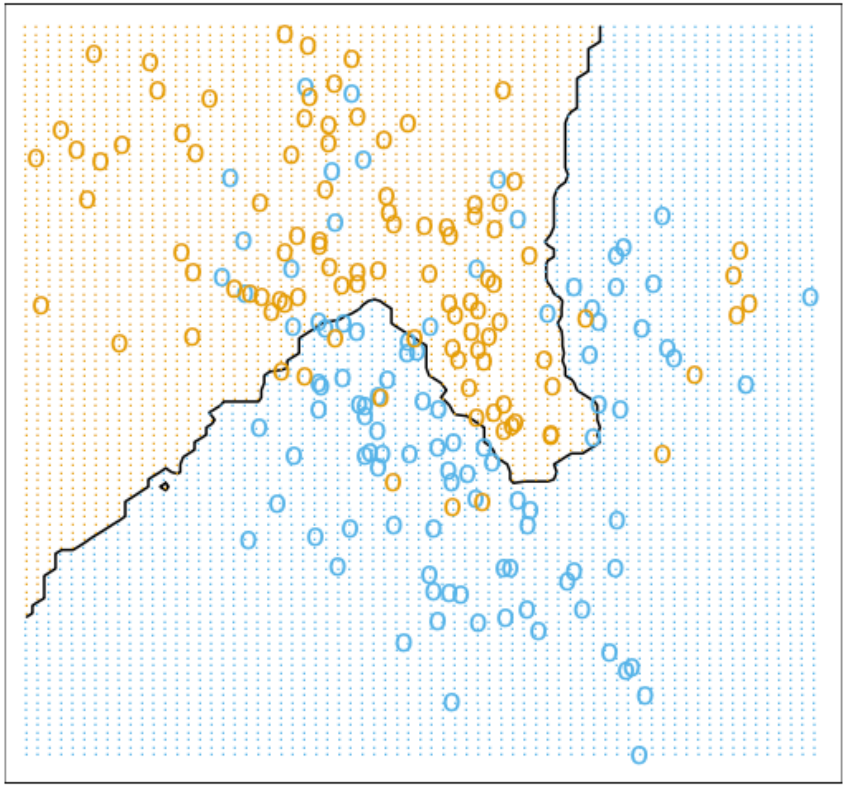
- Expectation is approximated by averaging over sample data;
- Conditioning at a point is relaxed to conditioning on some region "close" to the target point.

Linear regression

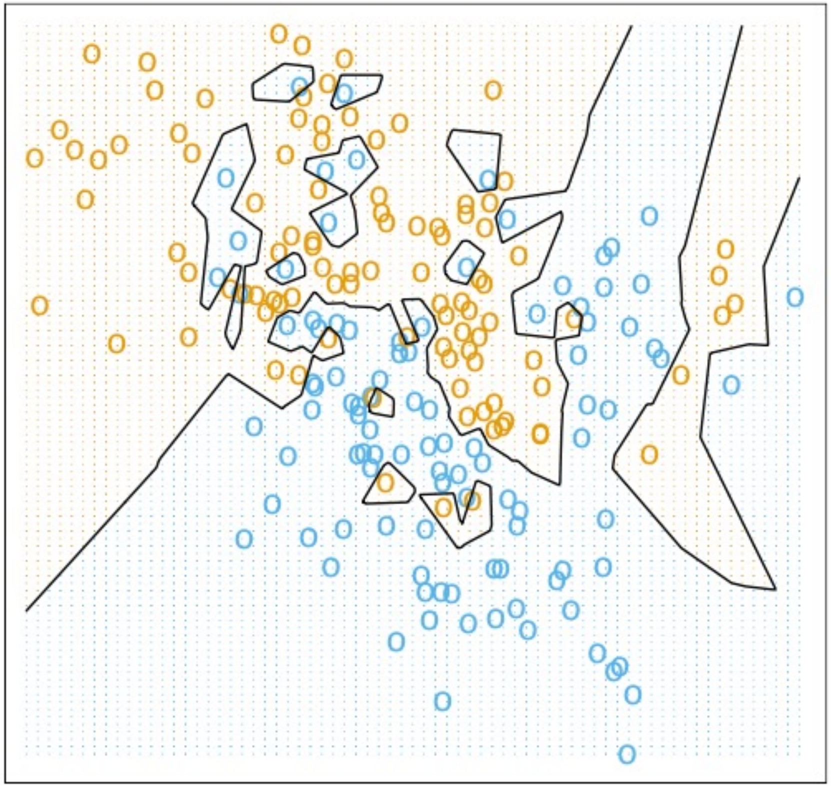


The effective number of parameters of linear regression is *p*.

Nearest neighbor classifier



15-Nearest Neighbor Classifier



1–Nearest Neighbor Classifier

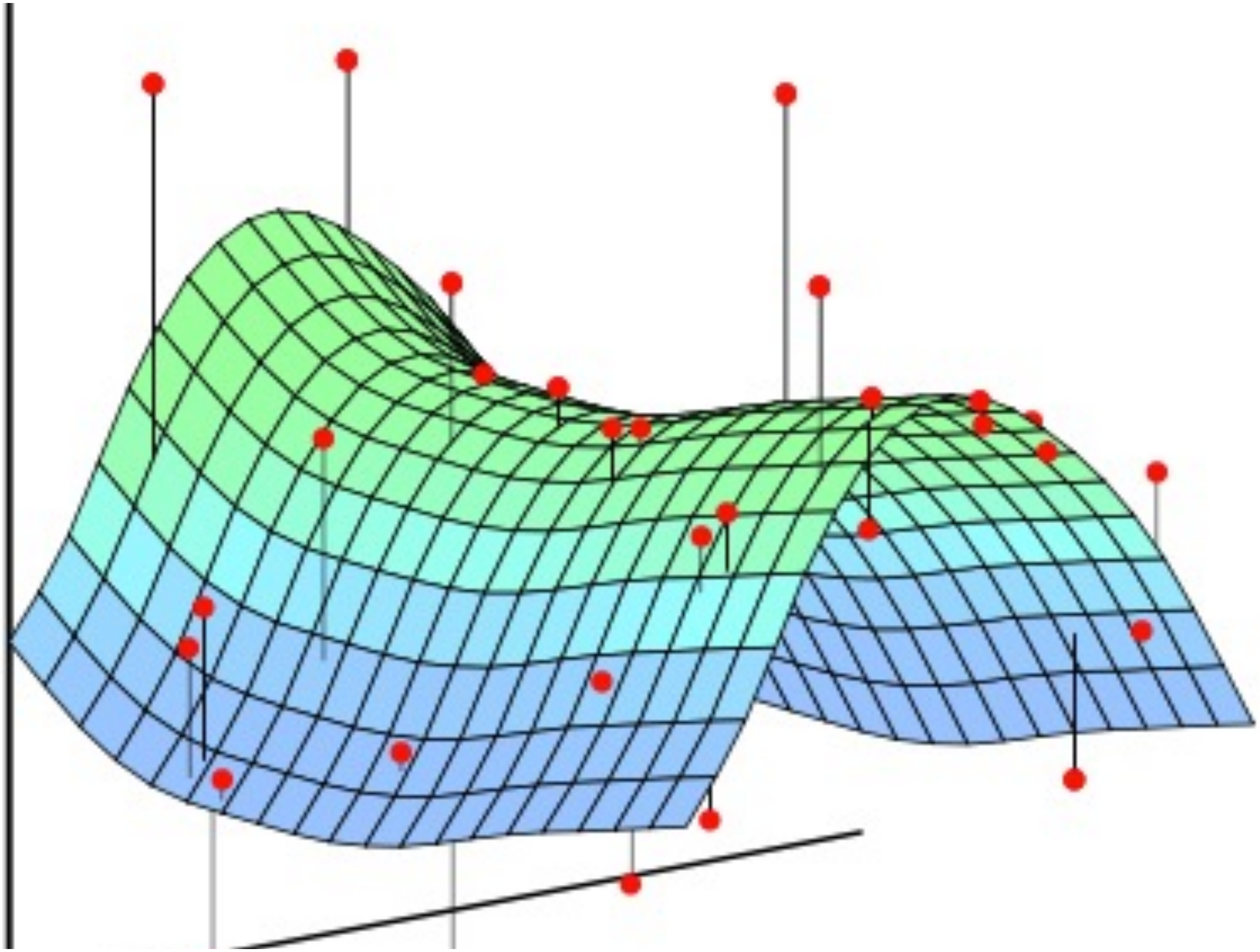
The effective number of parameters of k-nearest neighbors is N/k and is generally bigger than p, and decreases with increasing k.

Bias-variance decomposition

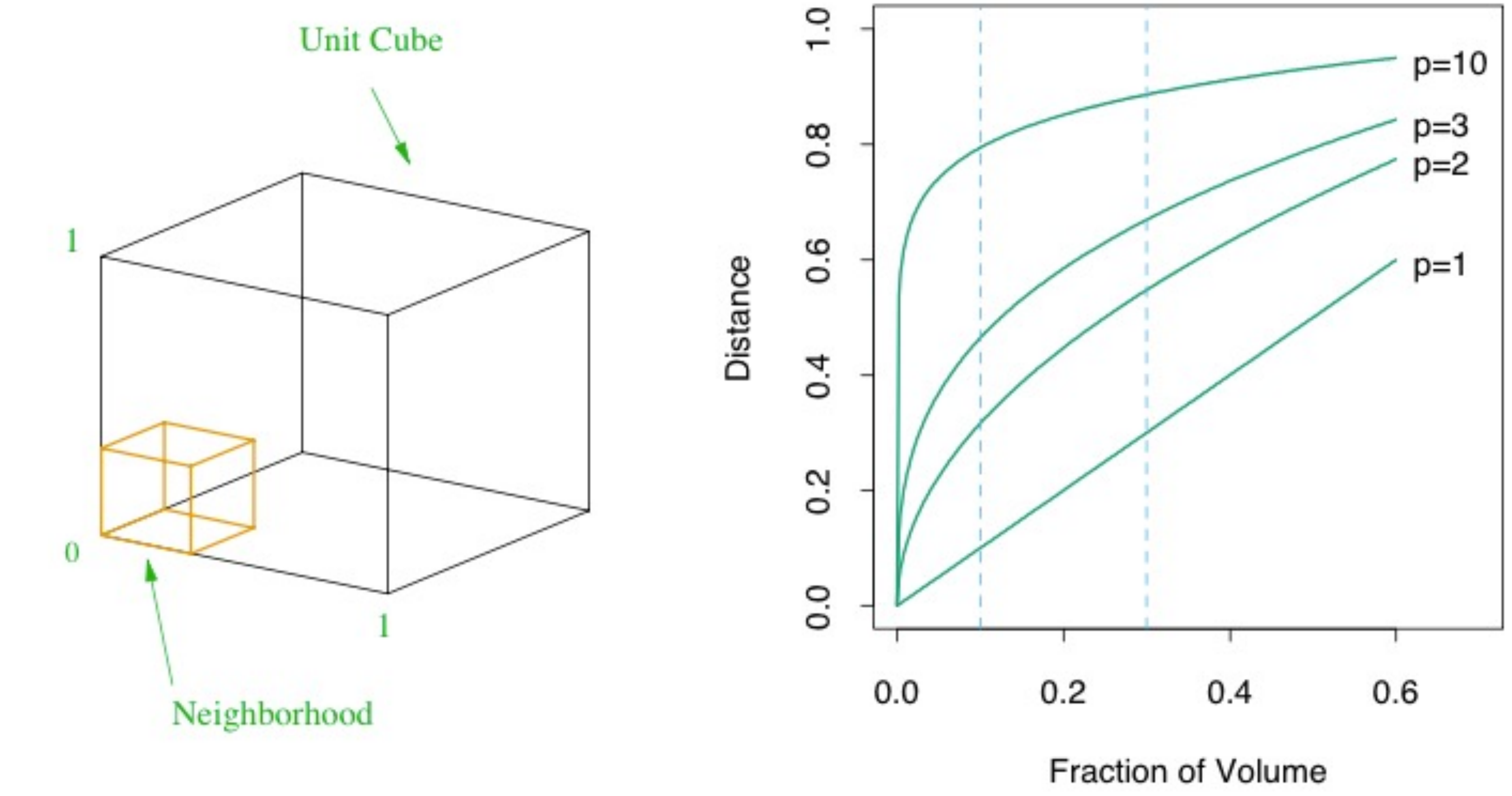
Mean Squared Error (MSE)

$$\begin{aligned} MSE(x_0) &= E_{\mathcal{T}}[f(x_0) - \hat{y}_0)]^2 \\ &= E_{\mathcal{T}}[\hat{y}_0 - E_{\mathcal{T}}(\hat{y}_0)]^2 + [E_{\mathcal{T}}(\hat{y}_0) - f(x_0)]^2 \\ &= Var_{\mathcal{T}}(\hat{y}_0) + Bias^2(\hat{y}_0) \end{aligned}$$

Least squares fitting of a linear model

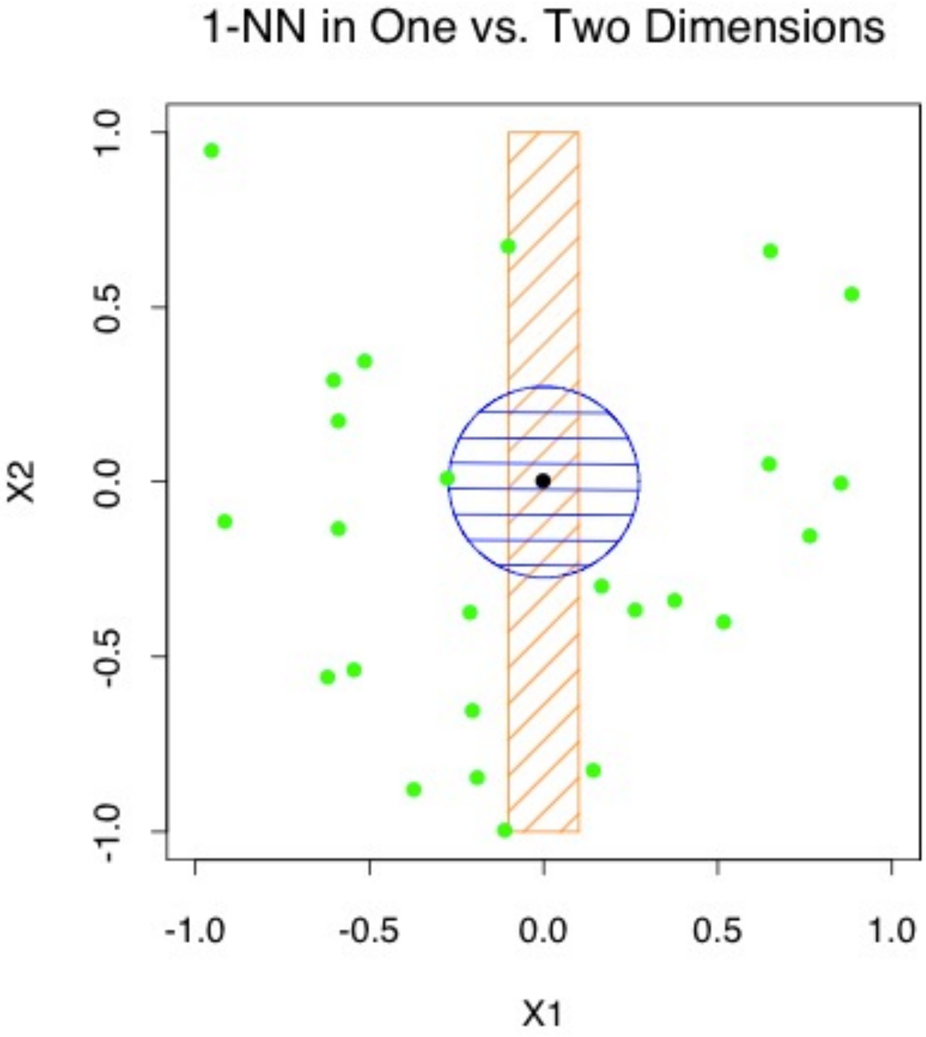


Curse of dimensionality



The expected edge length is $e_p(r) = r^{1/p}$, $e_{10}(0.01) = 0.63$ and $e_{10}(0.1) = 0.80$

One vs. two nearest neighbor



Problem of high dimensions

- To capture 1% or 10% of the data to form a local average, we must cover 63% or 80% of the range of each input variable. Such neighborhoods are no longer "***local***." Reducing *r* dramatically does not help much either, since the fewer observations we average, the higher is the variance of our fit.
- Most data points are closer to the boundary of the sample space than to any other data point. The reason that this presents a problem is that prediction is much more difficult near the edges of the training sample. One must extrapolate from neighboring sample points rather than interpolate between them.
- The sampling density is proportional to $N^{1/p}$, where p is the dimension of the input space and N is the sample size. Thus, if $N_1 = 100$ represents a dense sample for a single input problem, then $\mathrm{N}_{10} = 100^{10}$ is sample size required for the same sampling density with 10 inputs.

Linear model vs. k-nearest-neighbor

Linear model	K-nearest-neighbor
Stable	Unstable
Inaccurate	Accurate
High bias low variance	High variance low basis
Linear decision boundary assumption	No any stringent assumptions about the underlying data

Enhanced models

A large subset of the most popular techniques in use today are variants of these two simple procedures.

- Kernel methods use weights that decrease smoothly to zero with distance from the target point, rather than the effective 0/1 weights used by k-nearest neighbors.
- In high-dimensional spaces the distance kernels are modified to emphasize some variable more than others.
- Local regression fits linear models by locally weighted least squares, rather than fitting constants locally.
- Linear models fit to a basis expansion of the original inputs allow arbitrarily complex models.
- Projection pursuit and neural network models consist of sums of non-linearly transformed linear models.

Statistics and data

- In God we trust, all others bring data. – William Edwards Deming (1900 - 1993)
- How to extract useful information from data to make us better understand the world.
- We should pay close attention to how statisticians deal with the data.

Where the errors come from?

- Data itself (e.g. noisy)
- Models (whether the models used fit to the data)
- Parameter estimations (different estimation methods).

Error decomposition

Consider the prediction of the new response at input X.

$$Y = f(X) + \varepsilon$$

Then the expected prediction error of an estimate $f(X) = X^T\beta$ is

$$\begin{aligned} E(Y - \hat{f}(X))^2 &= \sigma^2 + E(X^T\hat{\beta} - f(X))^2 \\ &= \sigma^2 + Var(\hat{f}(X)) + [E(\hat{f}(X)) - f(X)]^2 \\ &= \sigma^2 + Var(\hat{f}(X)) + Bias^2(\hat{f}(X)) \end{aligned}$$

Cannot be controlled

Shrinkage methods - ridge

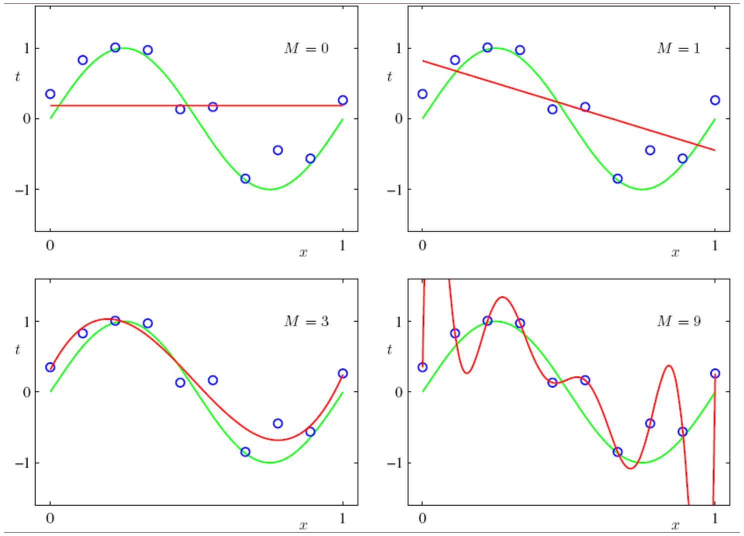
$$\hat{\beta}^{ridge} = \underset{\beta}{\arg\min}\left\{\sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{p}\beta_j^2\right\}$$

$$\hat{\beta}^{ridge} = \underset{\beta}{\arg\min}\sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 \quad \text{subject to} \quad \sum_{i=1}^{p}\beta_j^2 \le t$$

When there are many correlated variables in a linear regression model, their coefficients can become poorly determined and exhibit high variance. A wildly large positive coefficient on one variable can be canceled by a similarly large negative coefficient on its correlated cousin. By imposing a ***size constraint*** on the coefficients, this problem is alleviated.

The coefficients for ploynomials

	$M=0$	$M=1$	$M=6$	$M=9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43



$$\widetilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w}) - t_n\}^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

$$\|\mathbf{w}\|^2 \equiv \mathbf{w}^{\mathrm{T}}\mathbf{w} = w_0^2 + w_1^2 + \ldots + w_M^2$$

Shrinkage methods - ridge

$$RSS(\lambda) = (\mathrm{y} - \mathrm{X}\beta)^T (\mathrm{y} - \mathrm{X}\beta) + \lambda\beta^T \beta$$

$$\hat{\beta}^{ridge} = (\mathrm{X}^T \mathrm{X} - \lambda \mathrm{I})^{-1} \mathrm{X}^T \mathrm{y}$$

The solution adds a positive constant to the diagonal of $\mathrm{X}^T\mathrm{X}$ before inversion. This makes the problem nonsingular, even if $\mathrm{X}^T\mathrm{X}$ is not of full, rank, and was the main motivation for ridge regression when it was first introduced in statistics (Hoeral and Kennard, 1970).

Shrinkage methods - ridge

The ***singular value decomposition*** (SVD) of the centered input matrix X gives us some additional insight into the nature of ridge regression.

$\mathrm{X} = \mathrm{U}\mathrm{D}\mathrm{V}^{T}$ Here U and V are orthogonal matrices.

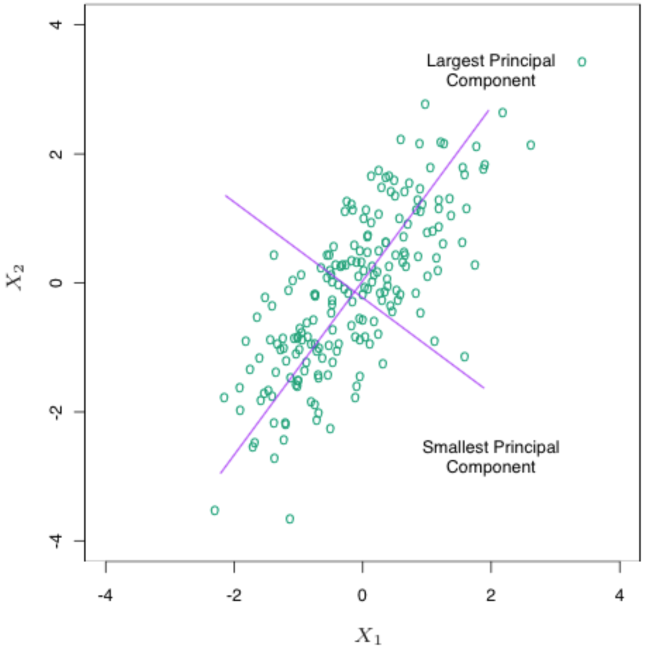
$$\begin{aligned}
\mathrm{X}\hat{\beta}^{ridge} &= \mathrm{X}(\mathrm{X}^{T}\mathrm{X} - \lambda \mathrm{I})^{-1}\mathrm{X}^{T}\mathrm{y} \\
&= \mathrm{U}\,\mathrm{D}(\mathrm{D}^{2} + \lambda \mathrm{I})^{-1}\mathrm{D}\,\mathrm{U}^{T}\mathrm{y} \\
&= \sum_{j=1}^{p} \mathrm{u}_{j} \frac{d_j^2}{d_j^2 + \lambda} \mathrm{u}_j^{T}\mathrm{y}
\end{aligned}$$

$$\hat{\beta}^{ridge} = \hat{\beta} / (1 + \lambda)$$

Shrinkage methods - ridge

$$\hat{\beta}^{ridge} = \hat{\beta} / (1+\lambda)$$

This means that a greater amount of shrinkage is applied to the coordinates of basis vectors with smaller d_j^2



The implicit assumption is that the response will tend to ***vary most*** in the directions of ***high variance*** of the inputs. This is often a reasonable assumption.

$$\mathrm{df}(\lambda) = \mathrm{tr}[\mathrm{X}(\mathrm{X}^T\mathrm{X}+\lambda \mathrm{I})^{-1}\mathrm{X}^T]$$
$$= \sum_{j=1}^{p} \frac{d_j^2}{d_j^2+\lambda}$$

The effective degrees of freedom

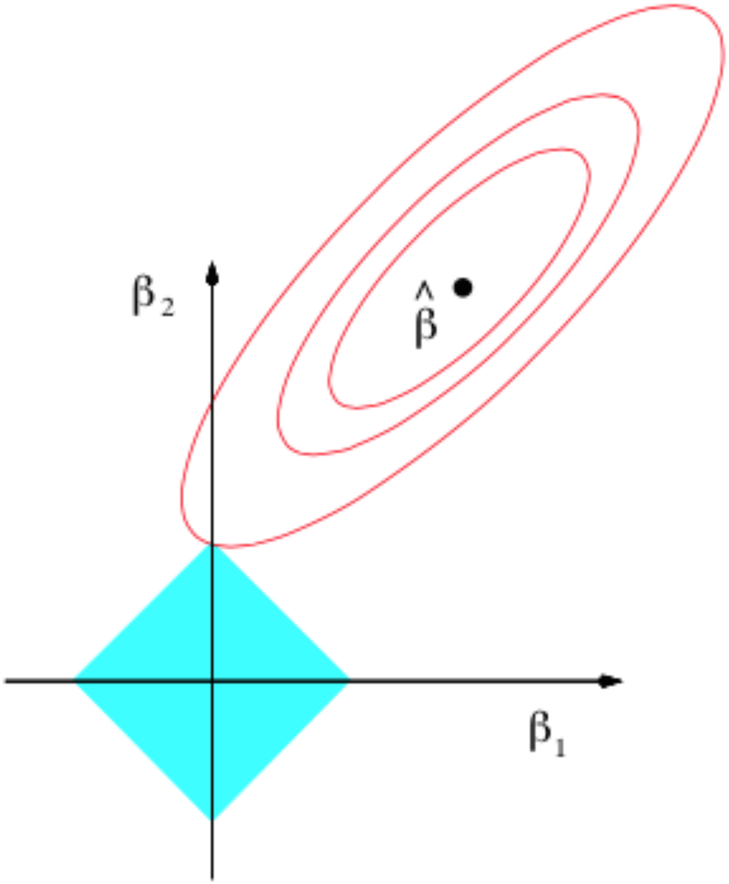
Shrinkage methods - lasso

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \left\{ \frac{1}{2} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

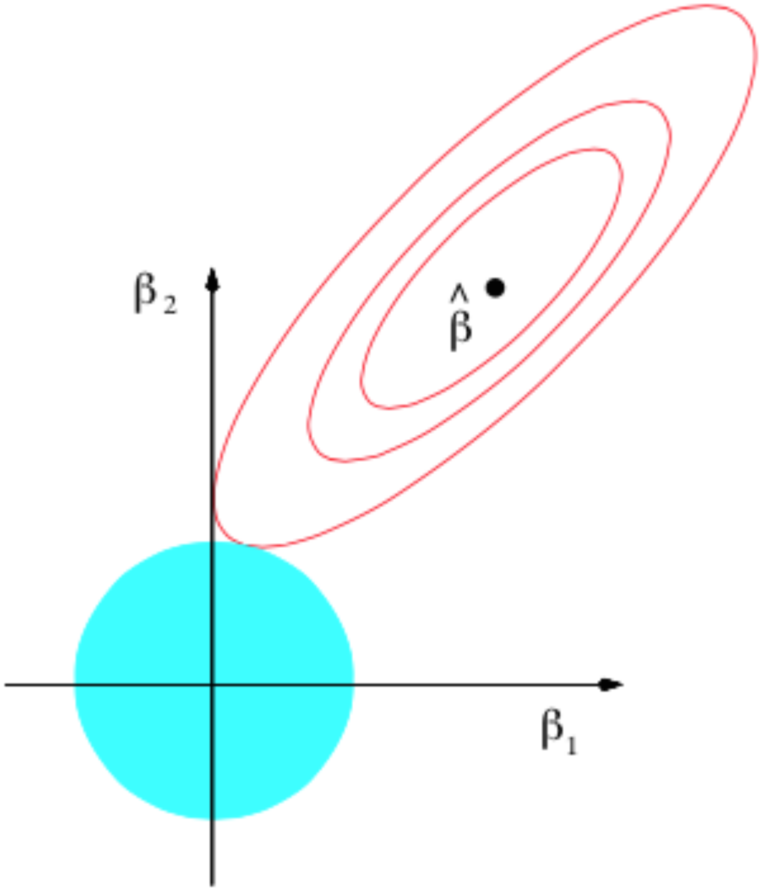
$$\hat{\beta}^{lasso} = \arg\min_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 \quad \text{subject to} \quad \sum_{i=1}^{p} |\beta_j| \le t$$

The L_2 ridge penalty is replaced by the L_1 penalty. This latter constraint makes the solutions nonlinear in the y_i, and there is no closed form expression as in ridge regression. Computing the lasso solution is a quadratic programming problem.

Ridge and Lasso regression



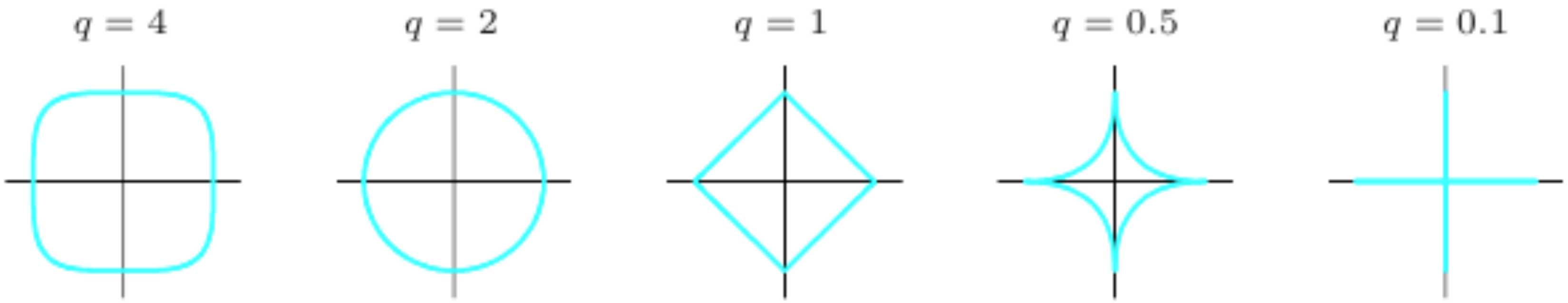
Lasso regression



Ridge regression

Bayes estimations

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \left|\beta_j\right|^q \right\}$$



The ***elastic-net penalty*** $\lambda \sum_{j=1}^{p} \left(\alpha\beta_j^2 + (1-\alpha)\left|\beta_j\right| \right)$

The elastic-net selects variables like the lasso, and shrinks together the coefficients of correlated predictors like ridge

Any questions?

AI Research Group
Fudan University