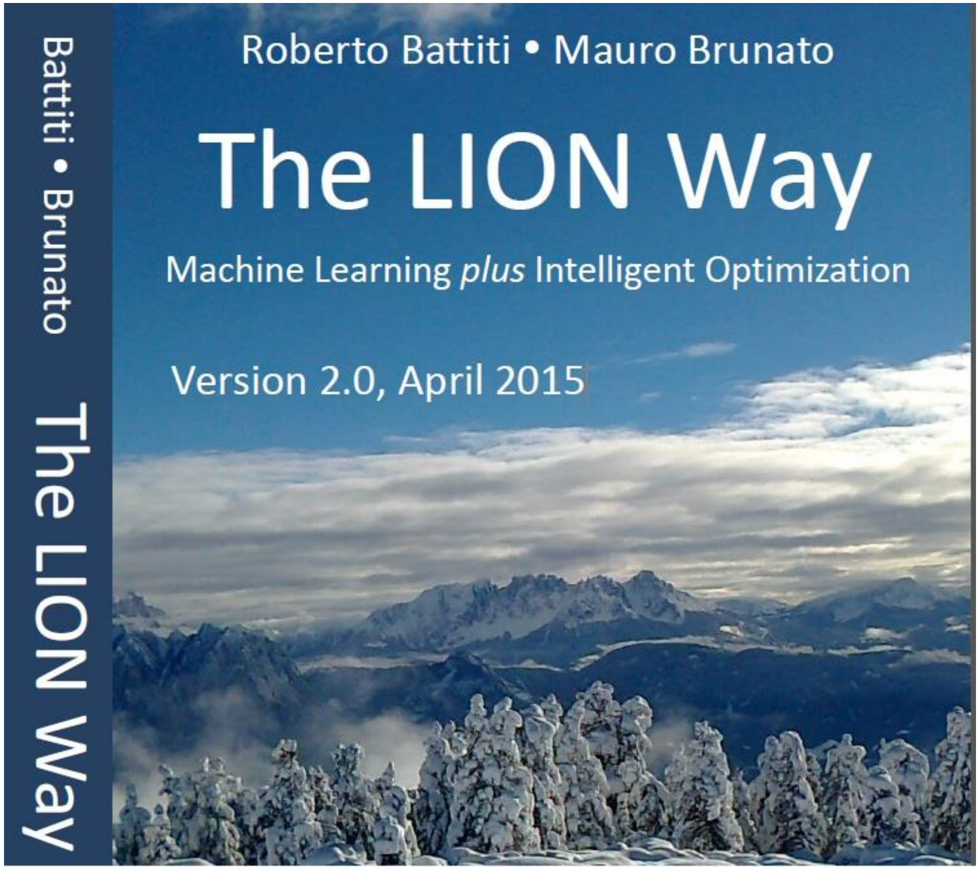
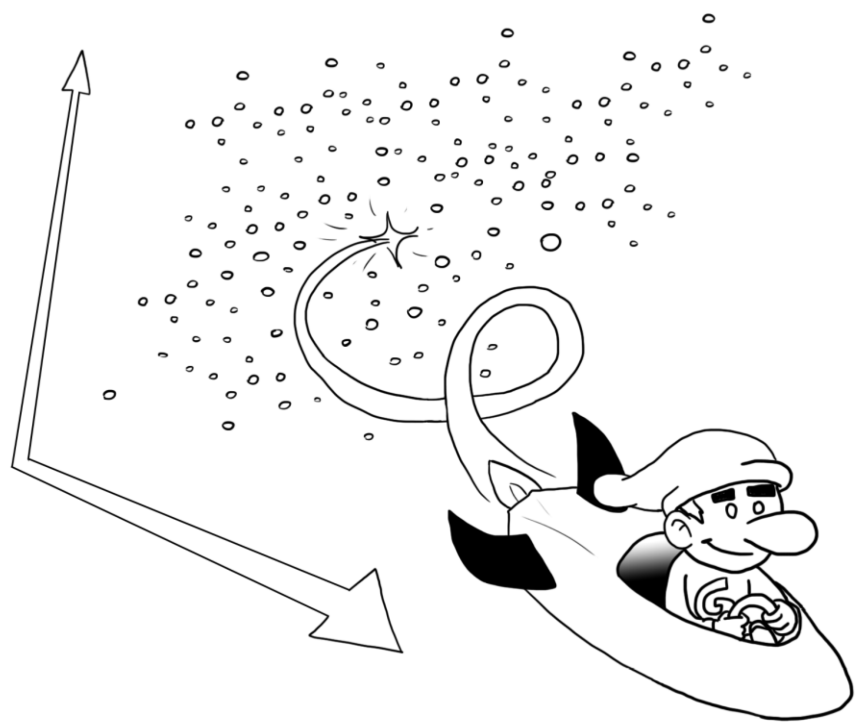


ROBERTO BATTITI, MAURO BRUNATO.
The LION Way: Machine Learning plus *Intelligent Optimization*.
LIONlab, University of Trento, Italy,
Apr 2015

http://intelligent-optimization.org/LIONbook

Chap. 8 Specific nonlinear models

He who would learn to fly one day must first learn to stand and walk and run and climb and dance; one cannot fly into flying.
(Friedrich Nietzsche)

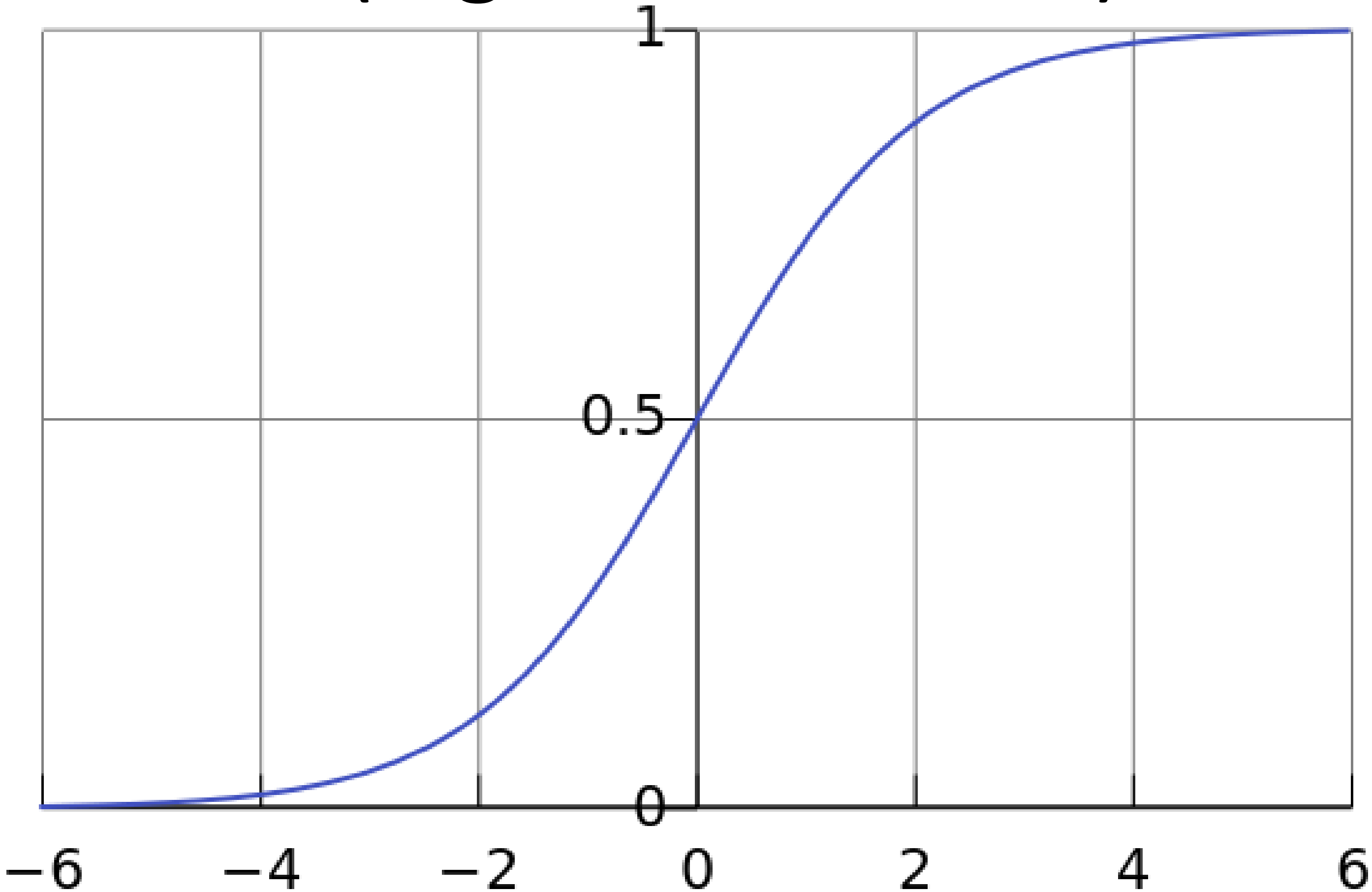


Logistic regression (1)

- Goal: predicting the probability of the outcome of a categorical variable
- It is a technique for classification, obtained through an estimate of the probability

- Problem with linear models: the output value is **not bounded**. We need to bound it between zero and one
- Solution: use a logistic function to transform the output of a linear predictor in a value between zero and one, which can be interpreted as a probability

Logistic regression (2) (logistic functions)



A logistic function transforms input values into an output value in the range 0-1, in a smooth manner. The output of the function can be interpreted as a probability.

Logistic regression (3) (logistic functions)

A logistic function a.k.a. sigmoid function is

$$P(t) = \frac{1}{1 + e^{-t}},$$

When applied to the output of the linear model

$$P(\boldsymbol{x}) = \frac{1}{1 + e^{-(\boldsymbol{w}^T \boldsymbol{x})}}.$$

The weights in the linear model need to be learnt

Logistic regression (4) (maximum likelihood estimation)

The best values of the weights in the linear model can be determined via by maximum likelihood estimation:

Maximize the (model) probability of getting the output values which were **actually** obtained on the given labeled examples.

Logistic regression (5) (maximum likelihood estimation)

- Let y_i be the observed output (1 or 0) for the corresponding input x_i
- let $\Pr(y = 1|x_i)$ be the probability obtained by the model.

the probability of obtaining the measured output value y_i is

- $\Pr(y = 1|x_i)$ if $y_i = 1$,
- $\Pr(y = 0|x_i) = 1 - \Pr(y = 1|x_i)$ if $y_i = 0$.

Logistic regression (6) (maximum likelihood estimation)

- After multiplying all the probabilities (independence) and applying logarithms → log likelihood

$$\mathrm{LogLikelihood}(\boldsymbol{w}) = \sum_{i=1}^{\ell} \left\{ y_i \ln \Pr(y = y_i | \boldsymbol{x}_i, \boldsymbol{w}) + (1 - y_i) \ln(1 - \Pr(y = y_i | \boldsymbol{x}_i, \boldsymbol{w})) \right\}.$$

- The log likelihood depends on the weights of the linear regression

- No closed form expression for the weights that maximize the likelihood function: an iterative process can be used for maximizing (for example gradient descent)

Locally-weighted regression

- Motivation: similar cases are usually deemed more relevant than very distant ones to determine the output (*natura non facit saltus*)

- Determine the output of an evaluation point as a weighted linear combination of the outputs of the neighbors (with bigger weights for closer neighbors)

Locally-weighted regression (2)

- The linear regression depends on the query point (it is local)

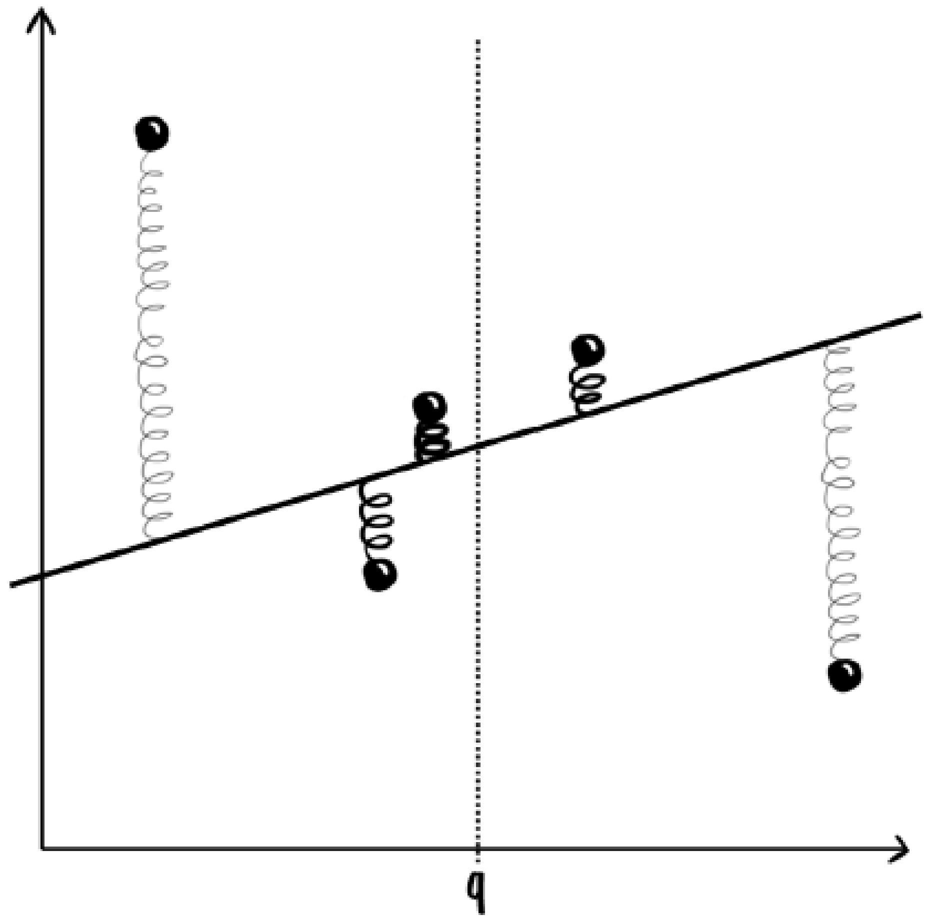
- Linear regression on the training points with a significance that decreases with its distance from the query point.

Locally-weighted regression (3)

- Given a query-point q, let s_i be the significance level assigned to the i-th labelled example.
- The weighted version of least squares fit aims at minimizing the following weighted error

(1) $$\mathrm{error}(w; s_1, \ldots, s_n) = \sum_{i=1}^{\ell} s_i (w^T \cdot x_i - y_i)^2.$$

Locally-weighted regression (4)



Locally-weighted regression (5)

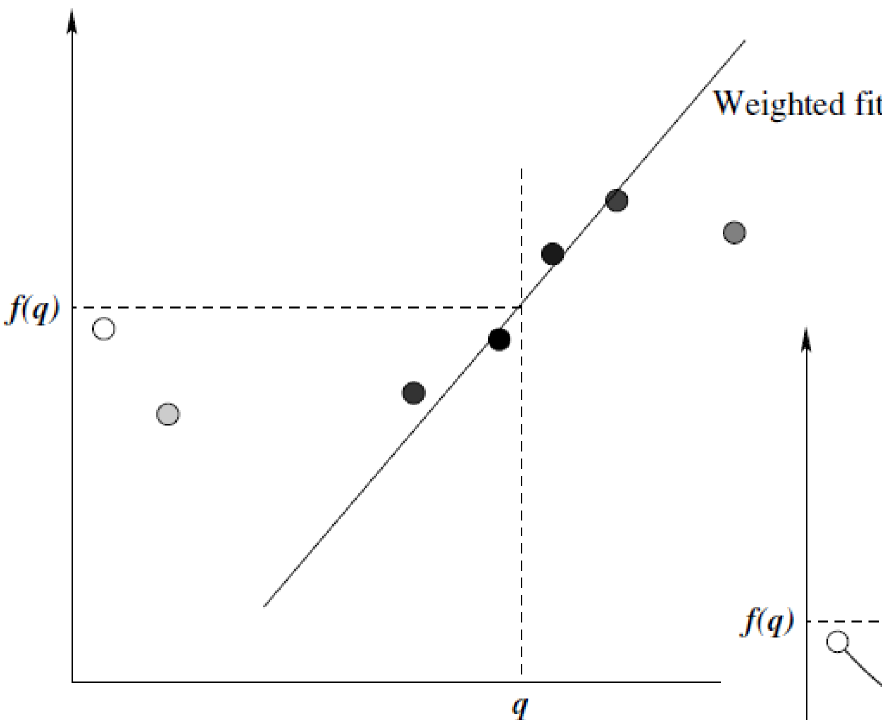
- Minimization of equation (1) yields the solution

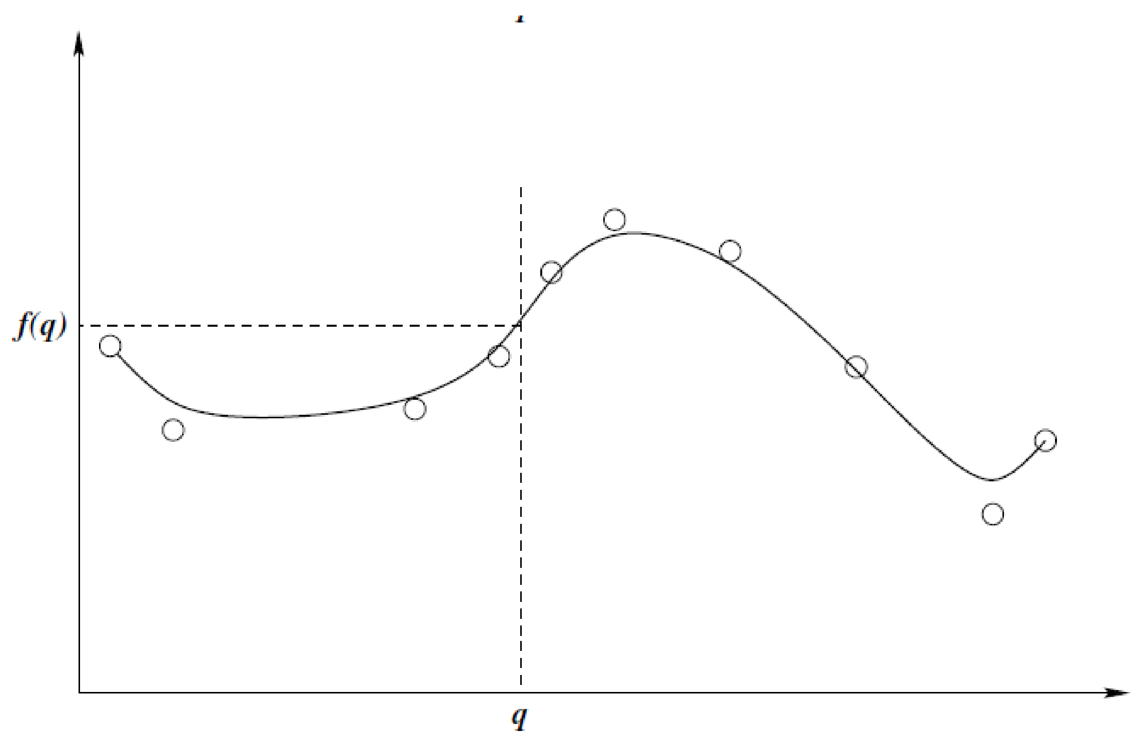
$$\boldsymbol{w}^* = (X^T S^2 X)^{-1} X^T S^2 \boldsymbol{y};$$

- where S = diag($s_1, \ldots, s_d$)
- A common function used to set the relationship between significance and distance is

$$s_i = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{q}\|^2}{W_K}\right);$$

Locally-weighted regression (6)





Evaluation of LWR model at query point q, sample point significance is represented by the interior shade.
Right: Evaluation over all points, each point requires a different linear fit.

Bayesian Locally-weighted regression

- B-LWR, is used if prior information about what values the coefficients should have can be specified when there is not enough data to determine them.
- Advantage of Bayesian techniques: the possibility to model not only the expected values but entire probability distributions (and to derive “error bars”)

Bayesian Locally-weighted regression(2)

- Prior assumption: $w=N(0,\Sigma)$
- $\Sigma=diag(\sigma_1,\ldots,\sigma_l)$
- $1/\sigma_i=Gamma(k,\vartheta)$

Let $S=diag(s_1,\ldots,s_l)$ be the matrix of the significance levels prescribed to each point

Bayesian Locally-weighted regression (3)

- The local model for the query point q is predicted by using the distribution of w whose mean is estimated as

$$\bar{\boldsymbol{w}} = (\Sigma^{-1} + X^T S^2 X)^{-1}(X^T S^2 \boldsymbol{y}).$$

- The variance of the Gaussian noise is estimated as

$$\sigma^2 = \frac{2\theta + (\boldsymbol{y}^T - \boldsymbol{w}^T X^T) S^2 \boldsymbol{y}}{2k + \sum_{i=1}^{\ell} s_i^2}.$$

Bayesian Locally-weighted regression (4)

- The estimated covariance matrix of the w distribution is then calculated as

$$\sigma^2 V_w = \frac{(2\theta + (\boldsymbol{y}^T - \boldsymbol{w}^T X^T) S^2 \boldsymbol{y})(\Sigma^{-1} + X^T S^2 X)}{2k + \sum_{i=1}^{\ell} s_i^2}.$$

- The predicted output response for the query point q is

$$\hat{y}(\boldsymbol{q}) = \boldsymbol{q}^T \bar{\boldsymbol{w}}$$

- The variance of the mean predicted output is:

$$Var(\hat{y}(\boldsymbol{q})) = \boldsymbol{q}^T V_w \boldsymbol{q} \sigma^2$$

LASSO to shrink and select inputs

- With a large number of input variables, we would like to determine a smaller subset that exhibits the strongest effects.

- Feature subset selection: can be very variable
- Ridge regression: more stable, but does not reduce the number of input variables

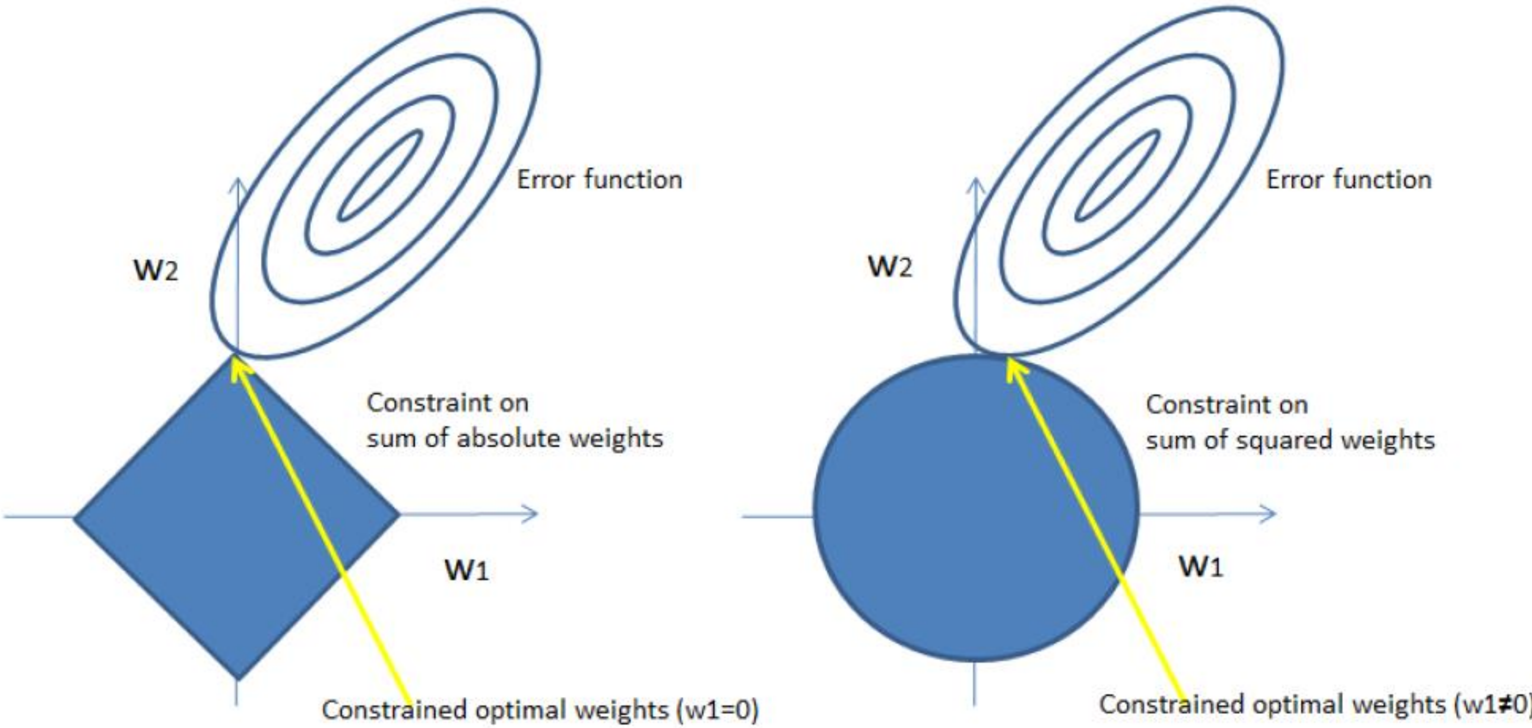
LASSO ("least absolute shrinkage and selection operator") retains the good features of *both* subset selection *and* ridge regression. It **shrinks** some coefficients and sets other ones to **zero**.

LASSO to shrink and select inputs (2)

- LASSO minimizes the residual sum of squares subject to the sum of the **absolute value** of the coefficients being less than a constant.
- Using Lagrange multipliers, it is equivalent to the following unconstrained minimization:

$$\text{LASSOerror}(\boldsymbol{w}; \lambda) = \sum_{i=1}^{\ell} (\boldsymbol{w}^T \cdot \boldsymbol{x}_i - y_i)^2 + \lambda \sum_{j=0}^{d} |w_j|.$$

LASSO to shrink and select inputs (3)



LASSO vs. Ridge regression

- In ridge regression, as the penalty is increased, all parameters are reduced while still remaining non-zero
- In LASSO, increasing the penalty will cause more and more of the parameters to be driven to **zero**.
- The inputs corresponding to weights equal to zero can be **eliminated**
- LASSO is an embedded method to perform **feature selection as part of the model construction**

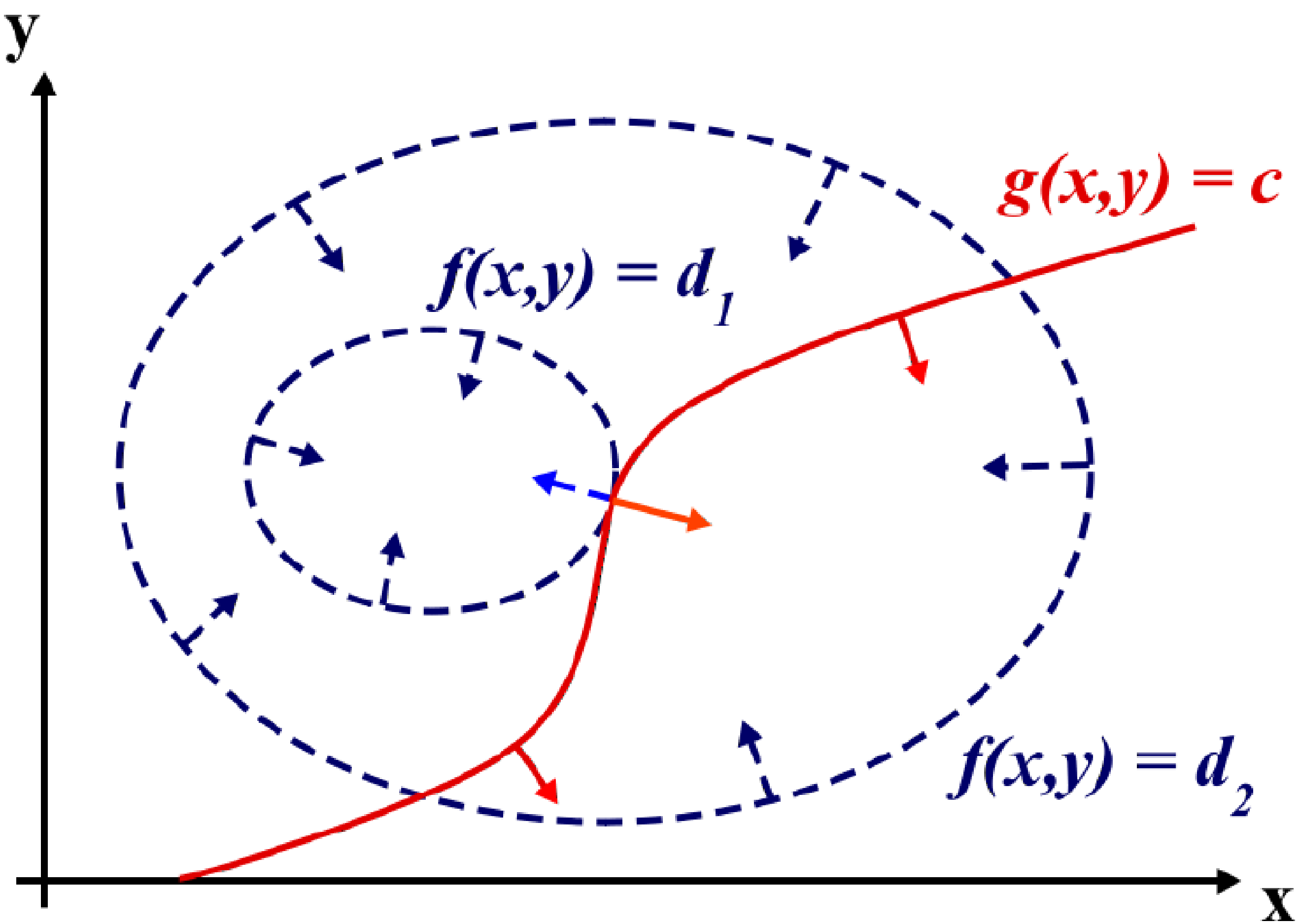
Lagrange multipliers

- The method of Lagrange multipliers is a strategy for finding the local maxima and minima of a function **subject to constraints**

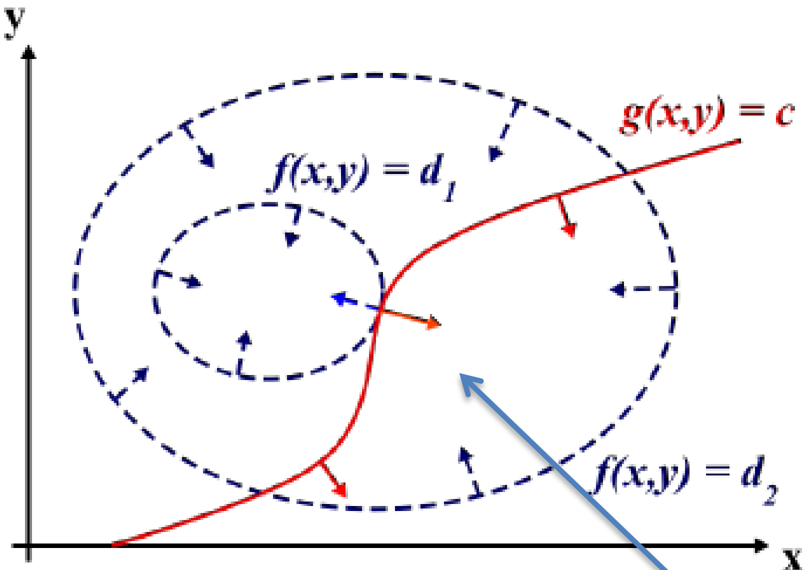
- The problem is transformed into an unconstrained one by adding each constraint multiplied by a parameter (Lagrange multiplier)

Lagrange multipliers(2)

Two-dimensional problem: Max f(x,y) Subject to g(x,y)=c



Lagrange multipliers(3)



Suppose we walk along the contour line with g = c. while moving along the contour line for g = c the value of f can vary.

Only when the contour line for g = c meets contour lines of f tangentially, f is approx. constant.

This is the same as saying that the gradients of f and g are parallel, thus we want points (x, y) where g(x, y) = c and

$$\nabla f(x, y) = \lambda \nabla g(x, y).$$

The Lagrange multiplier specifies how one gradient needs to be multiplied to obtain the other one.

Gist

- Linear models are widely used but insufficient in many cases
- **logistic regression** can be used if the output needs to have a limited range of possible values (e.g., if it is a probability),
- **locally-weighted regression** can be used if a linear model needs to be localized, by giving more significance to input points which are closer to a given input sample to be predicted.

Gist(2)

- **LASSO** reduces the number of weights different from zero, and therefore the number of inputs which influence the output.