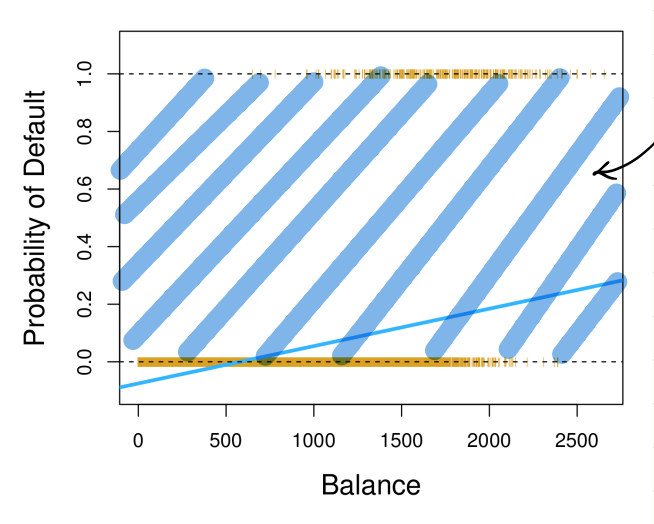
Topic 7: Logistic regression

Section 1: logistic regression

One of the issues we have encountered with MSE is the way it estimates our missclassifications, indeed if $\tilde{p}_{\hat{w}}(x) > 1$ then it contributes to the error term, but shouldn't as it just shows that we are confident in our prediction even more than for a general case $\tilde{p}_{\hat{w}}(x) \geq \frac{1}{2}$.

Also, the outcome of regression analysis is rather continuous than discrete, and the resulting MSE error penalises this, but we should allow this and maybe interpret this differently.

We need to apply a different result's digestion!



The object function should take values in $[0,1]$.

We should also consider values of $\tilde{p}_{\hat{w}}(x)$ as a probability of belonging to one of the classes.

Transformation of linear regression

Below we consider the way of transforming the linear regression problem to adapt it to binary classification needs, but the method works for any regression problem. Let

$$f(x, w) = \langle x, w \rangle$$

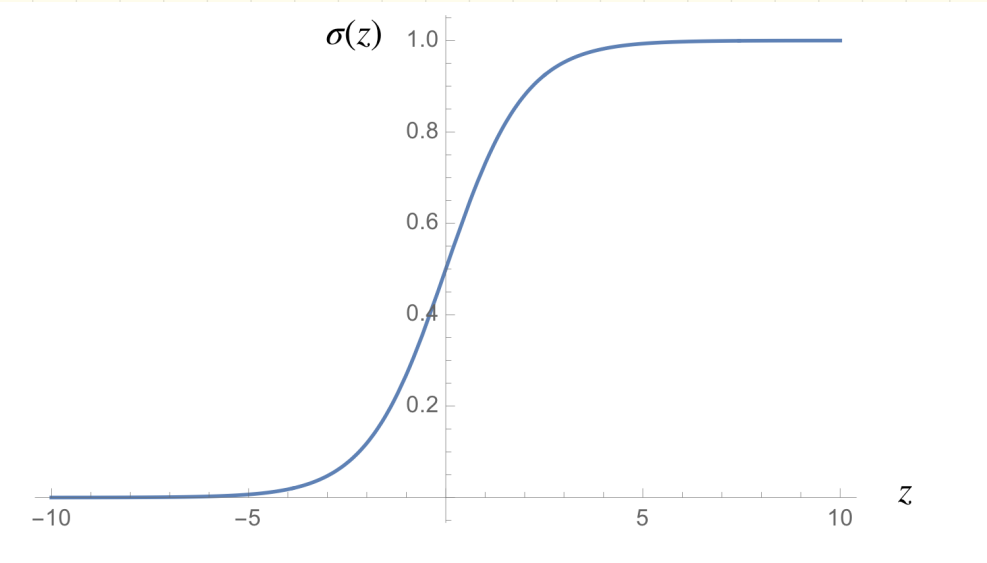
(w: weights; x: augmented x)

We will change our object/prediction function now with a composition (like in DL) of a function $\sigma: \mathbb{R} \to [0,1]$ applied to f.

The standard way of defining σ is

$$\sigma(z) = \frac{e^z}{1+e^z}$$

(sigmoid function)

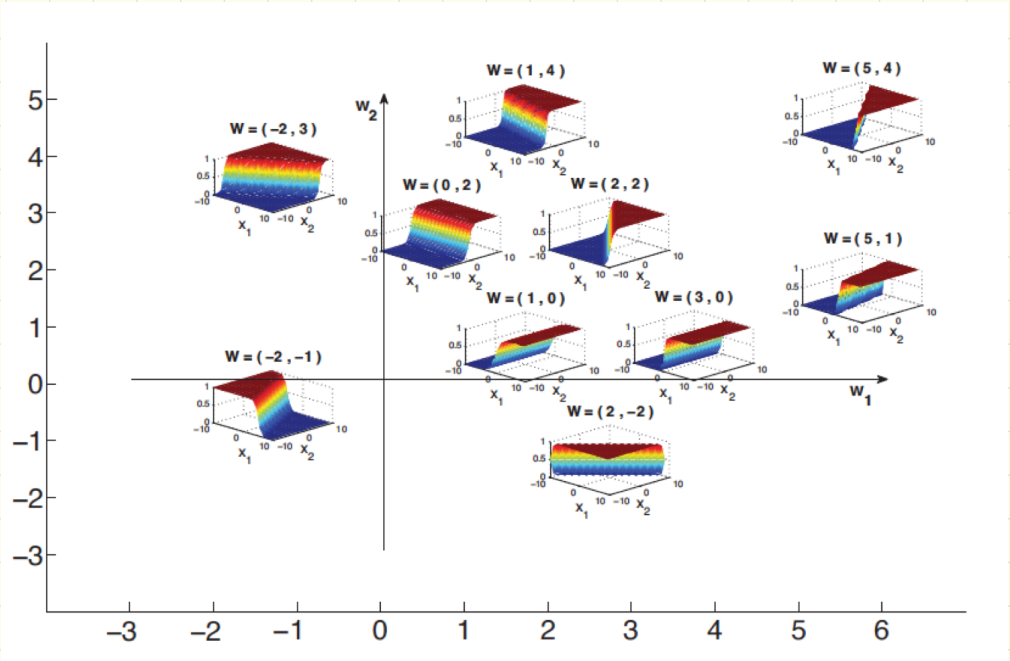


Then for any sample input x we define posterior probability of having corresponding class labels as

$$p(1|x) = \sigma\left(f(x,w)\right) = \sigma(\langle x, w\rangle)$$

tendency to class 1.

$$p(0|x) = 1 - \sigma\left(f(x,w)\right) = 1 - \sigma(\langle x, w\rangle)$$



How do we obtain the optimal weights $\hat{w}$?

Similar to all our previous problems, we expect optimal weights to maximise the likelihood of sampling given data sample from the distribution of class labels given by logistic function.

Optimal weights/parameters derivation

$\{x^{(i)}, y_i\}_{i=1}^{s}$ — i.i.d.

$x^{(i)} \in \mathbb{R}^d$, $y_i \in \{0, 1\}$

$$y := \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_s \end{pmatrix}, \quad X = \begin{pmatrix} x^{(1)T} \\ x^{(2)T} \\ \vdots \\ x^{(s)T} \end{pmatrix} = \begin{pmatrix} 1 & x_1^{(1)} & \dots & x_d^{(1)} \\ 1 & x_1^{(2)} & \dots & x_d^{(2)} \\ \vdots & & & \\ 1 & x_1^{(s)} & \dots & x_d^{(s)} \end{pmatrix}$$

↑ augmented

$$p(y \mid X, w) \overset{\text{independence}}{=} \prod_{i=1}^{s} p(y_i \mid x^{(i)}, w)$$

$$= \prod_{i:\, y_i = 0} p(y_i \mid x^{(i)}, w) \prod_{i:\, y_i = 1} p(y_i \mid x^{(i)}, w)$$

$$\text{"}1-\sigma\text{"} \qquad \text{"}\sigma\text{"}$$

Trick $\quad a^{\alpha} b^{1-\alpha} = \begin{cases} a, & \alpha = 1 \\ b, & \alpha = 0 \end{cases}$

$$= \prod_{i=1}^{s} \sigma(\langle x^{(i)}, w \rangle)^{y_i} \left(1 - \sigma(\langle x^{(i)}, w \rangle)\right)^{1-y_i}$$

maximise

Let us calculate minus log-likelihood

$$-\log p(y \mid X, w) = -\sum_{i=1}^{s} y_i \log \sigma_i + (1-y_i) \log(1 - \sigma_i)$$

$\sigma_i = \sigma(\langle x^{(i)}, w \rangle)$

$e_i = e^{\langle x^{(i)}, w \rangle}$

$$= -\sum_{i=1}^{s} y_i \log \frac{e_i}{1+e_i} + (1-y_i) \log \frac{1}{1+e_i}$$

$$= -\sum_{i=1}^{s} y_i \log e_i + \log \frac{1}{1+e_i}$$

So the negative log-likelihood takes the form

$$-\log p(y \mid X, w) = \sum_{i=1}^{s} \log\left(1 + \langle x^{(i)}, w \rangle\right) - y_i \langle x^{(i)}, w \rangle$$

Remark: for general regression with object function f one would obtain

$$-\log p(y \mid X, w) = \sum_{i=1}^{s} \log\left(1 + f(x^{(i)}, w)\right) - y_i f(x^{(i)}, w)$$

Optimal parameters/weights are then found as

$$\hat{w} = \arg\min_w -\log p(y \mid X, w)$$
$$= \arg\min_w \sum_{i=1}^{s} \log\left(1 + \langle x^{(i)}, w \rangle\right) - y_i \langle x^{(i)}, w \rangle$$

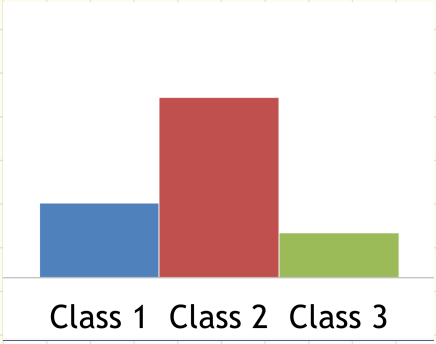
Further steps:

1. Extension of the method to multiclass classification problems
2. Ways of solving the optimisation problem

Multinomial logistic regression

Idea: instead of two probabilities $p(0|x,w)$ $p(1|x,w)$ one needs to define n probabilities $p(j|x,w)$, $j=\overline{1,n}$ such that

$$\sum_{j=1}^{n} p(j|x,w) = 1$$



Binary case: $z \rightarrow \frac{e^z}{1+e^z}, \frac{1}{1+e^z} = \frac{e^z}{e^0+e^z}, \frac{e^0}{e^0+e^z}$

Multinomial case: $(z_1, z_2, \ldots, z_n)$ ← n "independent" values

$\rightarrow$ soft max $(z_1, \ldots, z_n)$

What is softmax function?

$\mathrm{softmax}(\bar{x} = (x_1, x_2, \ldots, x_n))$

should satisfy

- $\mathrm{softmax} : \mathbb{R}^n \to$ n-dim. probability simplex
- softmax is monotonic in all coordinates

Def.

$$\sigma(\bar{x} = (x_1, x_2, \ldots, x_n)) = \left(\frac{e^{x_1}}{\sum_{j=1}^{n} e^{x_j}}, \frac{e^{x_2}}{\sum_{j=1}^{n} e^{x_j}}, \ldots, \frac{e^{x_n}}{\sum_{j=1}^{n} e^{x_j}} \right)$$

Remark: Component-wise the softmax function is

$$\sigma(\bar{x})_k = \frac{e^{x_k}}{\sum_{j=1}^{n} e^{x_j}}$$

Remark: The softmax name comes from the fact that softmax function is a continuous approximation of an argmax function

$$\mathrm{argmax}(x_1, x_2, \ldots, x_n) = x_p$$

$$= (0\ 0\ \ldots\ 0\ \underbrace{1}_{p}\ 0\ \ldots\ 0)$$

$$\mathrm{soft\,max}(x_1, \ldots, x_n) = \left(\frac{e^{x_1}}{\sum_{j=1}^{n} e^{x_j}}, \ldots, \underbrace{\frac{e^{x_p}}{\sum_{j=1}^{n} e^{x_j}}}_{\text{largest and} \to 1 \text{ if } x_p \to \infty}, \ldots, \frac{e^{x_n}}{\sum_{j=1}^{n} e^{x_j}} \right)$$

$\mathrm{argmax}(1.5 \quad 0.3 \quad -3.7) = 0 = (1\ 0\ 0)$

$\mathrm{softmax}(1.5 \quad 0.3 \quad -3.7) = (0.765 \quad 0.231 \quad 0.004)$

Setting up the multinomial regression

$\{x^{(i)}, y_i\}_{i=1}^{s}$ – data samples

$x^{(i)} \in \mathbb{R}^d$, $y_i \in \{1, 2, \dots, K\}$

Linear regression assumption is not essential but makes the notations clearer.

$$f(\bar{x}, W) = \left(\langle \bar{x}, w^{(1)} \rangle \quad \langle \bar{x}, w^{(2)} \rangle \quad \dots \quad \langle \bar{x}, w^{(k)} \rangle\right)$$

$$\sigma(\bar{x}, W) = \left(\frac{e^{\langle \bar{x}, w^{(1)} \rangle}}{\sum_{j=1}^{k} e^{\langle \bar{x}, w^{(j)} \rangle}}, \dots, \frac{e^{\langle \bar{x}, w^{(k)} \rangle}}{\sum_{j=1}^{k} e^{\langle \bar{x}, w^{(j)} \rangle}} \right)$$

$$p(y_i = k \mid x^{(i)}, W) = \sigma(\bar{x}, W)_k = \frac{e^{\langle \bar{x}, w^{(k)} \rangle}}{\sum_{j=1}^{k} e^{\langle \bar{x}, w^{(j)} \rangle}}$$

The likelihood maximisation

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_s \end{pmatrix} \qquad X = \begin{pmatrix} x^{(1)T} \\ x^{(2)T} \\ \vdots \\ x^{(s)T} \end{pmatrix} \qquad x^{(i)} = (1 \; x_1^{(i)} \dots x_d^{(i)})^T$$

$$W = \left(w^{(1)} \; w^{(2)} \dots w^{(k)}\right) \qquad w^{(i)} = \left(w_0^{(i)} \; w_1^{(i)} \dots w_d^{(i)}\right)$$

$$p(y \mid X, W) \overset{\text{independence}}{=} \prod_{i=1}^{s} p(y_i \mid x^{(i)}, W)$$

split into groups

$$= \prod_{i: y_i = 1} p(1 \mid x^{(i)}, W) \cdot \prod_{i: y_i = 2} p(2 \mid x^{(i)}, W) \cdot \dots \cdot \prod_{i: y_i = k} p(k \mid x^{(i)}, W)$$

Trick: $\prod_{p=1}^{K} \alpha_p^{1\{p=j\}} = \alpha_j$

$$p(y \mid X, W) = \prod_{i=1}^{s} \prod_{k=1}^{K} p(k \mid x^{(i)}, W)^{1_{y_i=k}}$$

$$\hat{W} = \arg\min_W -\log p(y \mid X, W)$$

$$= \arg\min_W -\sum_{i=1}^{s} \sum_{k=1}^{K} \log p(k \mid x^{(i)}, W)^{1_{y_i=k}}$$

$$= \arg\min_W -\sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i=k} \cdot \log p(k \mid x^{(i)}, W)$$

$$= \arg\min_W -\sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i=k} \cdot \log \frac{e^{\langle x^{(i)}, w^{(k)} \rangle}}{\sum_{j=1}^{K} e^{\langle x^{(i)}, w^{(j)} \rangle}}$$

$$= \arg\min_W -\sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i=k} \left[\langle x^{(i)}, w^{(k)} \rangle - \log\left(\sum_{j=1}^{K} e^{\langle x^{(i)}, w^{(j)} \rangle} \right) \right]$$

$$= \arg\min_W \sum_{i=1}^{s} \underbrace{\sum_{k=1}^{K} 1_{y_i=k}}_{=1} \cdot \log\left(\sum_{j=1}^{K} e^{\langle x^{(i)}, w^{(j)} \rangle} \right) - \sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i=k} \langle x^{(i)}, w^{(k)} \rangle$$

$$\hat{W} = \arg\min_W \sum_{i=1}^{s} \log\left(\sum_{j=1}^{K} e^{\langle x^{(i)}, w^{(j)} \rangle} \right) - \sum_{i=1}^{s} \sum_{k=1}^{K} 1_{y_i=k} \langle x^{(i)}, w^{(k)} \rangle$$

Section 2: Solving Logistic regression

Previously we have derived the optimisation/ minimisation problems that one needs to solve to classify the data using logistic regression.

Binary $$\hat{w} = \arg\min_w \left\{ \sum_{i=1}^{S} \log\left(1 + e^{\langle x^{(i)}, w \rangle}\right) - y_i \langle x^{(i)}, w \rangle \right\}$$

Multinomial

$$\hat{W} = \arg\min_W \left\{ \sum_{i=1}^{S} \log\left(\sum_{j=1}^{K} e^{\langle x^{(i)}, w^{(j)} \rangle} \right) - \sum_{i=1}^{S} \sum_{j=1}^{K} 1_{y_i = j} \langle x^{(i)}, w^{(j)} \rangle \right\}$$

Gradient descent?

$$L_2(w) = \sum_{i=1}^{S} \left\{ \log\left(1 + e^{\langle x^{(i)}, w \rangle}\right) - y_i \langle x^{(i)}, w \rangle \right\}$$

$$L_n(W) = \sum_{i=1}^{S} \log\left(\sum_{j=1}^{K} e^{\langle x^{(i)}, w^{(j)} \rangle} \right) - \sum_{i=1}^{S} \sum_{j=1}^{K} 1_{y_i = j} \langle x^{(i)}, w^{(j)} \rangle$$

$$W^{(k+1)} = W^{(k)} - \tau \nabla L(W^{(k)})$$

matrix or vector

We solely concentrate on a binary logistic regression

$$L_2(w) = \sum_{i=1}^{s} \log\left(1 + e^{\langle x^{(i)}, w\rangle}\right) - y_i \langle x_i^{(i)} w\rangle$$

To calculate the gradient we will need to calculate two sorts of derivatives:

- $$l(z) = \log\left(1 + e^{z}\right)$$

$$l'(z) = \frac{e^z}{1+e^z} = \sigma(z)$$

- $$\frac{\partial}{\partial w_p} \langle x^{(i)}, w\rangle = \frac{\partial}{\partial w_p} \sum_{j=0}^{d} x_j^{(i)} w_j$$

$$= x_p^{(i)}$$

$$\frac{\partial}{\partial w_p} L_2(w) = \sum_{i=1}^{s} \frac{\partial}{\partial w_p} l\left(\langle x^{(i)}, w\rangle\right) - y_i \langle x_i^{(i)} w\rangle$$

$$= \sum_{i=1}^{s} \frac{\partial}{\partial w_p} \langle x^{(i)}, w\rangle \cdot \frac{d}{dz_i}\left(l(z_i) - y_i \cdot z_i\right)$$

where $z_i = \langle x^{(i)}, w\rangle$

$$= \sum_{i=1}^{s} x_p^{(i)} \left(\sigma(\langle x^{(i)}, w\rangle) - y_i\right)$$

$$\nabla L_2(w) = X^T(\sigma(Xw) - y),$$

where $\sigma(u) = (\sigma(u_1)\ \sigma(u_2)\ \dots\ \sigma(u_s))$

Remark: The above is very similar to the classical regression gradient descent result.

Will the gradient descent converge to a global minimiser?

There are two parts:
1. Would the result be a minimiser?
2. would the method converge?

Convexity of L

L is a sum of identical functions applied to vectors $x^{(i)}$. So let us check every single piece individually.

L can be considered as a composition of a function applied to linear function, thus we need to check convexity of an external one only.

$$\ell(z) = \log(1 + e^z) \Rightarrow \ell'(z) = \sigma(z)$$

$$\sigma'(z) = \sigma(z)(1 - \sigma(z)) > 0$$

Remark: We can also consider regularisation of logistic regression

$$\hat{w} = \arg\min_{w} \{ L_2(w) + \alpha R(w) \}$$

If R is differentiable, then one can use the gradient descent directly. If R is not differentiable, then one can use the proximal gradient descent.