Logistic Regression for Classification

Bernoulli trials

lli trials are experiments with binary (two) outcome

lity of success = p, then probability of failure =q= 1-p

binary classification problem into probabilistic framework.

g upon the features, we assign objects to classes probabilistically.

$$Let\ \mathbf{x}_i = \begin{pmatrix} x_{i1} \\ \vdots \\ x_{in} \\ 1 \end{pmatrix}\ be\ the\ feature\ vector\ of\ ith\ object.$$

$$Corresponding\ class\ label\ be\ y_i \in \left[-1,1\right]$$

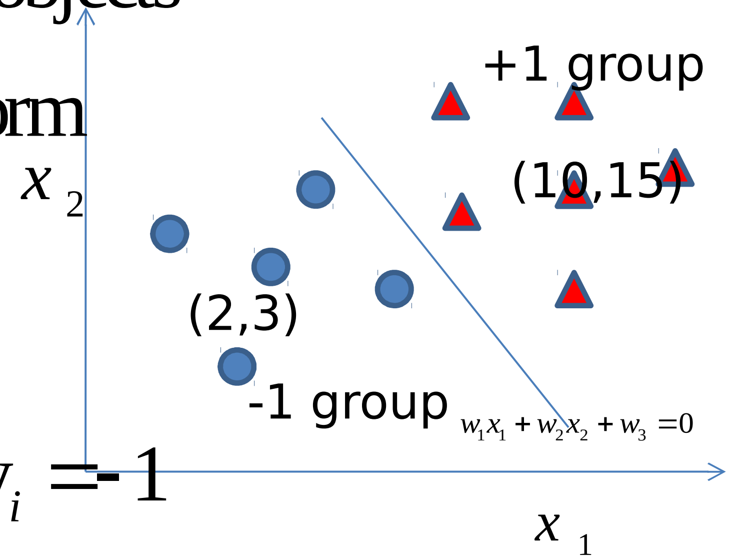
Here actual feature vector is $x_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{in} \end{pmatrix}$ 1 is appended at the end for the following re

Why append 1 to the feature vector

$Let\ (x_1, x_2)$ be the two feature values of objects

A binary linear classifier will be of the form

$$f(x|w) = w_1x_1 + w_2x_2 + w_3$$



$if\ f(x_i | w) > 0$, class label is $y_i = 1$ else $y_i = -1$

$or\ if\ w_1x_1 + w_2x_2 + w_3 > 0$, class label is $y_i = 1$ else $y_i = -1$

If we append 1 with data vector, we can write the classifier

in a more compact form: if $w^T x > 0$, class label is $y_i = 1$ else $y_i = -1$

How the data look like

$$A = \begin{bmatrix} x_{11} & x_{12} & 1 \\ x_{21} & x_{22} & 1 \\ x_{31} & x_{32} & 1 \\ \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & 1 \end{bmatrix}; y = \begin{bmatrix} 1 \\ -1 \\ 1 \\ \vdots \\ -1 \end{bmatrix}$$

Note, here data vectors are in rows

Some time, data can be in this format

$$A_1 = \begin{bmatrix} x_{11} & x_{21} & x_{31} & \cdots & x_{m1} \\ x_{12} & x_{22} & x_{32} & \cdots & x_{m2} \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix};$$

Problem

$$Find\ a\ \mathbf{w}\ such\ that\ \mathbf{w}^T\mathbf{x}_i > 0\ if\ \ y_i = 1$$

$$also\ \mathbf{w}^T\mathbf{x}_j < 0\ if\ \ y_j = -1$$

$\mathbf{w}$ *shouls be such that for most* **training** *data,*

the above relation is satisfied

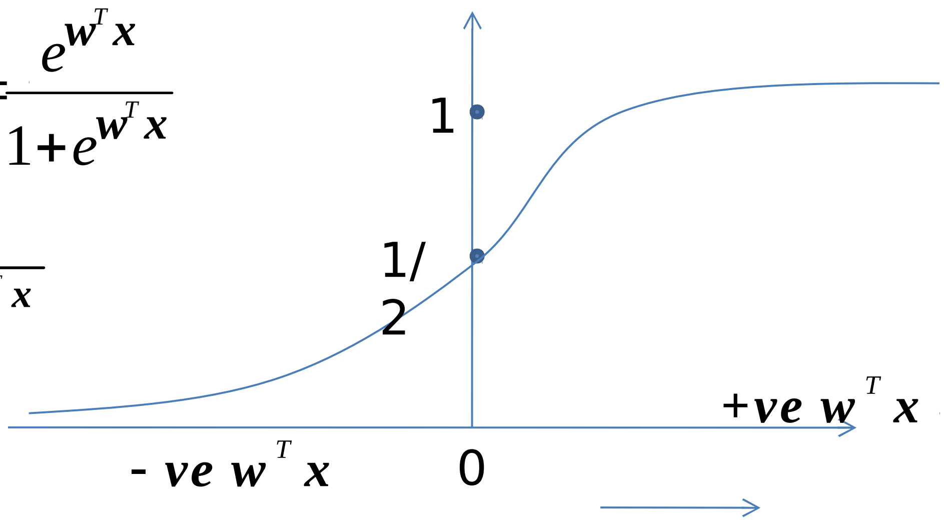
Once we obtain **w,** *we give probabilistic sataement for class / label prediction*

using the formula

$$\Pr ob(y = 1 \mid \mathbf{x}; \mathbf{w}) = p = f\left(\mathbf{w}^T\mathbf{x}\right) = \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}} = \frac{e^{\mathbf{w}^T\mathbf{x}}}{1 + e^{\mathbf{w}^T\mathbf{x}}}$$

$$\Pr ob(y = -1 \mid \mathbf{x}; \mathbf{w}) = 1 - p = 1 - \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}}$$

$$= \frac{e^{-\mathbf{w}^T\mathbf{x}}}{1 + e^{-\mathbf{w}^T\mathbf{x}}} = \frac{1}{1 + e^{\mathbf{w}^T\mathbf{x}}}$$



Training the Classifier

We assume we are given labeled data of the form

$$A = \begin{bmatrix} x_{11} & x_{12} & 1 \\ x_{21} & x_{22} & 1 \\ x_{31} & x_{32} & 1 \\ \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & 1 \end{bmatrix}; \; y = \begin{bmatrix} 1 \\ -1 \\ 1 \\ \vdots \\ -1 \end{bmatrix}$$

Training **involves** *Finding a* $\mathbf{w}$ *such that* $\mathbf{w}^T \mathbf{x}_i > 0$ *if* $y_i = 1$

also $\mathbf{w}^T \mathbf{x}_j < 0$ *if* $y_j = -1$

$\mathbf{w}$ *should be such that for most data the above relation is satisfied*

Objective function

*To obtain **w**, we need an objective function that connects our data and **w***

*We use the concept of **likelyhood function** from probability theory.*

$$\Pr ob(y = y_i = 1 \mid \boldsymbol{x}_i; \boldsymbol{w}) = p_i = \frac{1}{1+e^{-\boldsymbol{w}^T\boldsymbol{x}_i}} = \frac{1}{1+e^{-y_i\boldsymbol{w}^T\boldsymbol{x}_i}}$$

$$\Pr ob(y = y_i = -1 \mid \boldsymbol{x}_i; \boldsymbol{w}) = 1 - p_i = \frac{1}{1+e^{\boldsymbol{w}^T\boldsymbol{x}_i}} = \frac{1}{1+e^{-y_i\boldsymbol{w}^T\boldsymbol{x}_i}}$$

$$\therefore \Pr ob(y = y_i \mid \boldsymbol{x}_i; \boldsymbol{w}) = \frac{1}{1+e^{-y_i\boldsymbol{w}^T\boldsymbol{x}_i}}$$

This is for unknown data assuming w is known

For training data, data _label is already known

$$likelyhood\ of\ the\ data\ L(\boldsymbol{w}) = \prod_{i=1}^{m} \frac{1}{1+e^{-y_i\boldsymbol{w}^T\boldsymbol{x}_i}}$$

It is product of probabilities for the given data in terms of **w**

To simplify we take log

$$Log\ likelyhood\ of\ the\ data\ LogL(\boldsymbol{w}) = \prod_{i=1}^{m} \frac{1}{1+e^{-y_i\boldsymbol{w}^T\boldsymbol{x}_i}}$$

$$-\sum_{i=}^{m} \log\left(1+e^{-y_i\boldsymbol{w}^T\boldsymbol{x}_i}\right)$$

What is likelyhood function

nction that gives joint probability of occurrence of the data und
on that data is drawn independently from a given distribution.

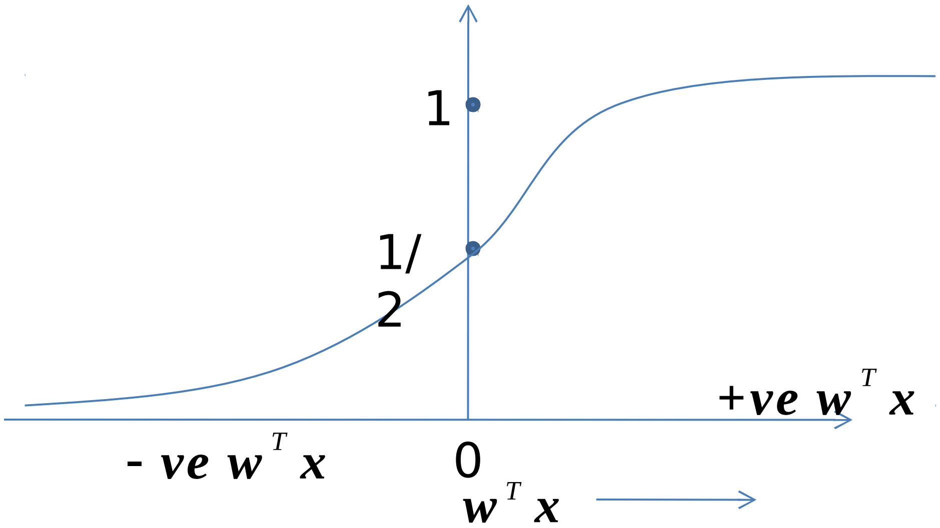
r a given **w**, we can assign a probability to each data vector
elong to the given category (already given label).

 to maximize the joint probability (for whole data)
g a proper **w,** for which, we use optimization theory

Logistic Regression

$$\mathrm{Prob}(y=1\,|\,\boldsymbol{x};\boldsymbol{w}) = p = f\left(\boldsymbol{w}^T\boldsymbol{x}\right) = \frac{1}{1+e^{-\boldsymbol{w}^T\boldsymbol{x}}} = \frac{e^{\boldsymbol{w}^T\boldsymbol{x}}}{1+e^{\boldsymbol{w}^T\boldsymbol{x}}}$$

$$\mathrm{Prob}(y=-1\,|\,\boldsymbol{x};\boldsymbol{w}) = 1-p = 1-\frac{1}{1+e^{-\boldsymbol{w}^T\boldsymbol{x}}} = \frac{e^{-\boldsymbol{w}^T\boldsymbol{x}}}{1+e^{-\boldsymbol{w}^T\boldsymbol{x}}} = \frac{1}{1+e^{\boldsymbol{w}^T\boldsymbol{x}}}$$



By optimization theory we will find a $\boldsymbol{w}$ *such that*

$if\ y_i = 1\ then\ \boldsymbol{w}^T\boldsymbol{x}_i > 0$

$\boldsymbol{else}\ \ if\ y_j = -1\ \ then\ \boldsymbol{w}^T\boldsymbol{x}_j < 0$

formulation

$$class\ label\ be\ y_i \in [1, -1]$$

$$\Pr ob(y = y_i = 1 \mid \boldsymbol{x}_i; \boldsymbol{w}) = p_i = \frac{1}{1 + e^{-\boldsymbol{w}^T \boldsymbol{x}_i}} = \frac{1}{1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}$$

$$\Pr ob(y = y_i = -1 \mid \boldsymbol{x}_i; \boldsymbol{w}) = 1 - p_i = \frac{1}{1 + e^{\boldsymbol{w}^T \boldsymbol{x}_i}} = \frac{1}{1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}$$

$$\therefore \Pr ob(y = y_i \mid \boldsymbol{x}_i; \boldsymbol{w}) = \frac{1}{1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}$$

$$likelyhood\ of\ the\ data = \prod_{i=1}^{m} \frac{1}{1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}$$

$$Log\ likelyhood = \log \prod_{i=1}^{m} \frac{1}{1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}} = -\sum_{i=}^{m} \log\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)$$

We should maximize this by adjusting

$$negative\ Log\ likelyhood = \log \prod_{i=1}^{m} \frac{1}{1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}} = \sum_{i=1}^{m} \log\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)$$

We should minimize this by adjusting w

$gradient:$

$$\nabla J(\boldsymbol{w}) = \sum_{i=1}^{m} \frac{-y_i e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}{\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)} \boldsymbol{x}_i$$

Programming

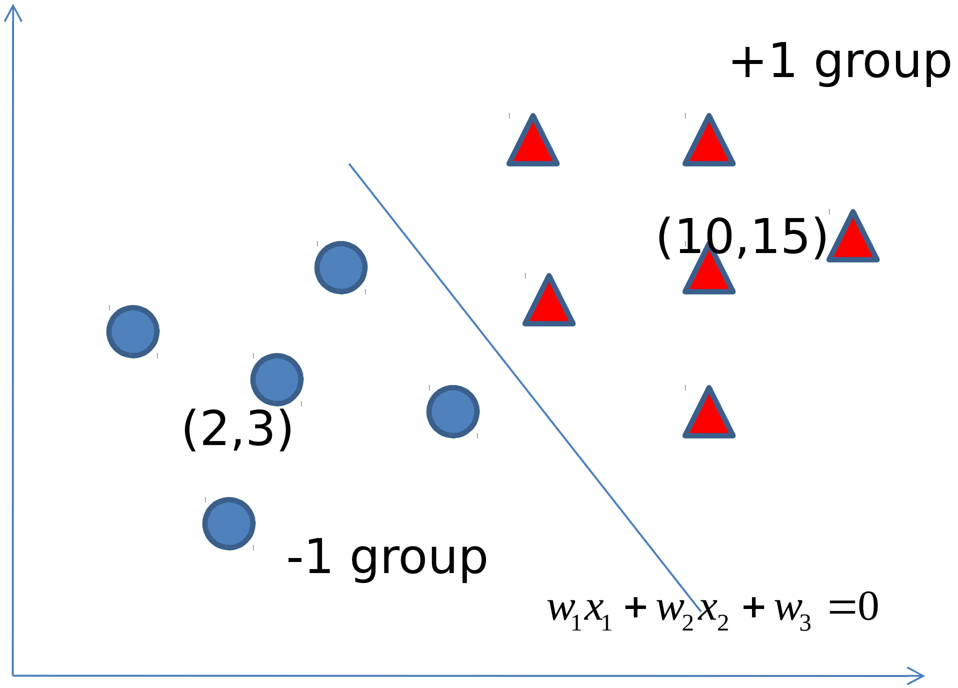
$$J(\boldsymbol{w}) = \sum_{i=1}^{m} \log\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)$$

gradient :

$$\nabla J(\boldsymbol{w}) = \sum_{i=1}^{m} \frac{-y_i e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}{\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)} \boldsymbol{x}_i$$

$$= \sum_i a_i \boldsymbol{x}_i = \textbf{linear combination of data\_vectors}$$

$$A=\begin{bmatrix} x_{11} & x_{12} & 1 \\ x_{21} & x_{22} & 1 \\ x_{31} & x_{32} & 1 \\ \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & 1 \end{bmatrix};\; y=\begin{bmatrix} 1 \\ -1 \\ 1 \\ \vdots \\ -1 \end{bmatrix}$$



Let us create 10 in the category +1 and another 10 in category -1

```
c1=[10;15];
c2=[2;3];
B=[randn(2,10)+repmat(c1,1,10)
randn(2,10)+repmat(c2,1,10);ones(1,20) ]
A=B';
y=[ones(10,1); -1*ones(10,1)];
```

Computing J(w)

$$A=\begin{bmatrix} x_{11} & x_{12} & 1 \\ x_{21} & x_{22} & 1 \\ x_{31} & x_{32} & 1 \\ \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & 1 \end{bmatrix}; y=\begin{bmatrix} 1 \\ -1 \\ 1 \\ \vdots \\ -1 \end{bmatrix} \quad \boldsymbol{w}=\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}$$

$$J(\boldsymbol{w}) = \sum_{i=1}^{m} \log\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)$$

$$J(\boldsymbol{w}) = \sum_{i=1}^{m} \log\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right) = sum\left(\log\left(1+\exp\left(-1 * y.* Aw\right)\right)\right)$$

$$A*w=\begin{bmatrix} x_{11} & x_{12} & 1 \\ x_{21} & x_{22} & 1 \\ x_{31} & x_{32} & 1 \\ \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & 1 \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}=\begin{bmatrix} \boldsymbol{w}^T\boldsymbol{x}_1 \\ \boldsymbol{w}^T\boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{w}^T\boldsymbol{x}_{m-1} \\ \boldsymbol{w}^T\boldsymbol{x}_m \end{bmatrix}; \quad \exp\left(-1 * y.*(A*w)\right) = \begin{bmatrix} e^{-y_1 \boldsymbol{w}^T \boldsymbol{x}_1} \\ e^{-y_2 \boldsymbol{w}^T \boldsymbol{x}_2} \\ \vdots \\ e^{-y_{m-1} \boldsymbol{w}^T \boldsymbol{x}_{m-1}} \\ e^{-y_m \boldsymbol{w}^T \boldsymbol{x}_m} \end{bmatrix}$$

$$1+\exp\left(y.*Aw\right) = \begin{bmatrix} 1+e^{-y_1 \boldsymbol{w}^T \boldsymbol{x}_1} \\ 1+e^{-y_2 \boldsymbol{w}^T \boldsymbol{x}_2} \\ \vdots \\ 1+e^{-y_{m-1} \boldsymbol{w}^T \boldsymbol{x}_{m-1}} \\ 1+e^{-y_m \boldsymbol{w}^T \boldsymbol{x}_m} \end{bmatrix} \quad \log(1+y.*Aw) = \log\begin{bmatrix} 1+e^{-y_1 \boldsymbol{w}^T \boldsymbol{x}_1} \\ 1+e^{-y_2 \boldsymbol{w}^T \boldsymbol{x}_2} \\ \vdots \\ 1+e^{-y_{m-1} \boldsymbol{w}^T \boldsymbol{x}_{m-1}} \\ 1+e^{-y_m \boldsymbol{w}^T \boldsymbol{x}_m} \end{bmatrix} = \begin{bmatrix} \log\left(1+e^{-y_1 \boldsymbol{w}^T \boldsymbol{x}_1}\right) \\ \log\left(1+e^{-y_2 \boldsymbol{w}^T \boldsymbol{x}_2}\right) \\ \vdots \\ \log\left(1+e^{-y_{m-1} \boldsymbol{w}^T \boldsymbol{x}_{m-1}}\right) \\ \left(1+e^{-y_m \boldsymbol{w}^T \boldsymbol{x}_m}\right) \end{bmatrix}$$

Computing J(w)

$$J(\boldsymbol{w}) = \sum_{i=1}^{m} \log\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)$$

$$J(\boldsymbol{w}) = \sum_{i=1}^{m} \log\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right) = sum\left(\log\left(1 + \exp\left(-1 * y .* A * w\right)\right)\right)$$

Computing $\nabla J(\boldsymbol{w})$

gradient:

$$\nabla J(\boldsymbol{w}) = \sum_{i=1}^{m} \frac{-y_i e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}{\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)} \boldsymbol{x}_i$$

$$1+\exp\left(-y.*\left(A*w\right)\right) = \begin{bmatrix} 1+e^{-y_1\boldsymbol{w}^T\boldsymbol{x}_1} \\ 1+e^{-y_2\boldsymbol{w}^T\boldsymbol{x}_2} \\ \vdots \\ 1+e^{-y_{m-1}\boldsymbol{w}^T\boldsymbol{x}_{m-1}} \\ 1+e^{-y_m\boldsymbol{w}^T\boldsymbol{x}_m} \end{bmatrix} \qquad -y.*\exp\left(y.*\left(A*w\right)\right) = \begin{bmatrix} -y_1e^{-y_1\boldsymbol{w}^T\boldsymbol{x}_1} \\ -y_2e^{-y_2\boldsymbol{w}^T\boldsymbol{x}_2} \\ \vdots \\ -y_{m-1}e^{-y_{m-1}\boldsymbol{w}^T\boldsymbol{x}_{m-1}} \\ -y_me^{-y_m\boldsymbol{w}^T\boldsymbol{x}_m} \end{bmatrix}$$

$Computing \quad \nabla J(\mathbf{w})$

$$\left(-y.*\exp\left(y.*\left(A*w\right)\right)\right)./\,(1+y.*A*w) = \begin{bmatrix} \dfrac{-y_1 e^{-y_1 \mathbf{w}^T \mathbf{x}_1}}{\left(1+e^{-y_1 \mathbf{w}^T \mathbf{x}_1}\right)} \\ \dfrac{-y_2 e^{-y_2 \mathbf{w}^T \mathbf{x}_2}}{\left(1+e^{-y_2 \mathbf{w}^T \mathbf{x}_2}\right)} \\ \vdots \\ \dfrac{-y_{m-1} e^{-y_{m-1} \mathbf{w}^T \mathbf{x}_{m-1}}}{\left(1+e^{-y_{m-1} \mathbf{w}^T \mathbf{x}_{m-1}}\right)} \\ \dfrac{-y_m e^{-y_m \mathbf{w}^T \mathbf{x}_m}}{\left(1+e^{-y_m \mathbf{w}^T \mathbf{x}_m}\right)} \end{bmatrix}$$

gradient:

$$\nabla J(\mathbf{w}) = \sum_{i=1}^{m} \frac{-y_i e^{-y_i \mathbf{w}^T \mathbf{x}_i}}{\left(1+e^{-y_i \mathbf{w}^T \mathbf{x}_i}\right)} \mathbf{x}_i$$

Computing $\nabla J(\boldsymbol{w})$

$$\nabla J(\boldsymbol{w}) = \nabla J(\boldsymbol{w}) = \sum_{i=1}^{m} \frac{-y_i e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}{\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)} \boldsymbol{x}_i$$

$= linear$ **combination of** $\boldsymbol{x}_i$

$= linear$ **combination of columns of** A^T

$$= A^T * \left(-y.*\exp\left(-y.*(A*w)\right)\right)./\left(1+\exp\left(-y.*A*w\right)\right)$$

Matlab Function to evaluate

$$J(\boldsymbol{w});\ \nabla J(\boldsymbol{w})$$

```
[Jfun, Jgrad] =Mylogistic(A, y,
w)
com=exp(-1*y.*A*w;
Jfun=sum(log(1+com);
GradJ=A'*(-y.*com)./(1+com);
end
```

Now apply gradient method to find the solution

$$A^T * \left(-y.*\exp\left(-y.*\left(A*w\right)\right)\right)./\left(1+\exp\left(-y.*A*w\right)\right)$$

New formulation

$L2-$ *regularized logistic regression*

$$\min_{\boldsymbol{w}} J(\boldsymbol{w}) = \frac{\lambda}{2}\boldsymbol{w}^T\boldsymbol{w} + \sum_{i=1}^{n} \log\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)$$

gradient :

$$\nabla J(\boldsymbol{w}) = \lambda \boldsymbol{w} + \sum_{i=1}^{n} \frac{-y_i e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}}{\left(1 + e^{-y_i \boldsymbol{w}^T \boldsymbol{x}_i}\right)} \boldsymbol{x}_i$$