University of Delaware

FSAN/ELEG815: Statistical Learning

Gonzalo R. Arce

Department of Electrical and Computer Engineering
University of Delaware

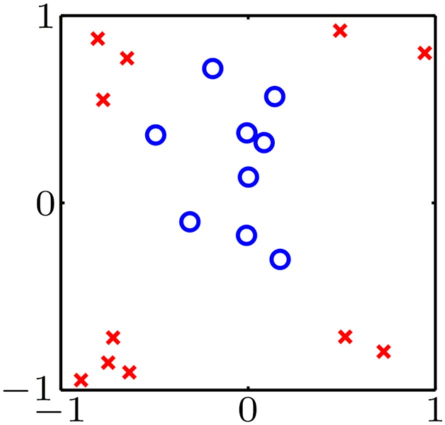
XII: Nonlinear Transformation and Logistic Regression

Outline of the Course

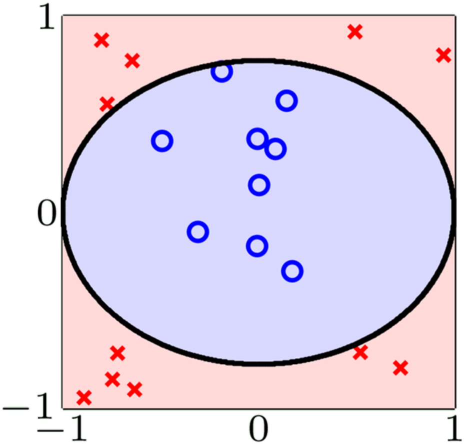
1. Review of Probability
2. Eigen Analysis, Singular Value Decomposition (SVD) and Principal Component Analysis (PCA)
3. The Learning Problem
4. Training vs Testing
5. Nonlinear Transformation and Logistic Regression
6. Overfitting and Regularization (Ridge Regression)
7. Lasso Regression
8. Neural Networks
9. Convolutional Neural Networks

Linear Model is Limited

Data:



Target Hypothesis:



Another Example

Credit line is affected by years in residence x_i

Does it affect the output linearly?

No! Stability might be achieved after about five years.

Define nonlinear features:

- $[\![x_i < 1]\!] \rightarrow$ credit limit is affected negatively.
- $[\![x_i > 5]\!] \rightarrow$ credit limit is affected positively.

Can we do this with linear models?

Linear in what?

Linear regression implements:

$$\sum_{i=0}^{d} w_i x_i$$

Linear classification implements

$$\text{sign}\left(\sum_{i=0}^{d} w_i x_i\right)$$

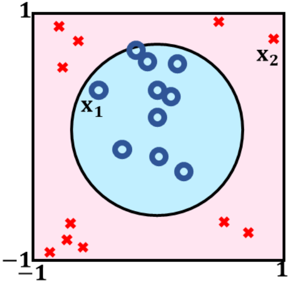
Algorithm works because of **linearity in the weights**.

Represent input by appropriate features and apply linear models.

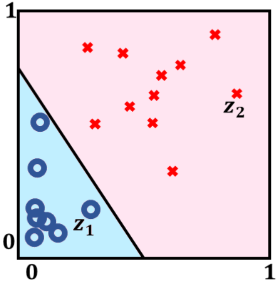
Example - Transform the Data Nonlinearly

- Data not linearly separable, but separable by a circle i.e. $x_1^2 + x_2^2 = 0.6$.
- A nonlinear hypothesis $h(\mathbf{x}) = \text{sign}(0.6 - x_1^2 - x_2^2)$ separates the data set.
- Hypotheses linear after applying a nonlinear transformation on $\mathbf{x}$:

$$h(\mathbf{x}) \quad = \quad \text{sign}\big[\underbrace{(0.6)}_{\tilde{w}_0} \cdot \underbrace{1}_{z_0} + \underbrace{(-1)}_{\tilde{w}_1} \cdot \underbrace{x_2^1}_{z_1} + \underbrace{(-1)}_{\tilde{w}_2^2} \cdot \underbrace{x_1^2}_{z_2}\big] = \text{sign}(\tilde{\mathbf{w}}^T \mathbf{z})$$



Data in $\mathcal{X}$-space



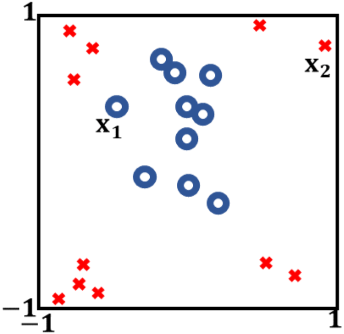
Transform data in $\mathcal{Z}$-space

Example - Transform the Data Nonlinearly

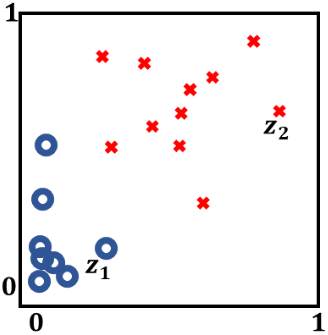
In feature space $\mathcal{Z}$, coordinates are higher-level features of raw input $\mathbf{x}$.

Let $\mathbf{z} = \Phi(\mathbf{x})$, where the transform $\Phi(\mathbf{x}) : \mathcal{X} \rightarrow \mathcal{Z}$ is defined as

$$(x_1, x_2) \xrightarrow{\Phi} (x_1^2, x_2^2)$$



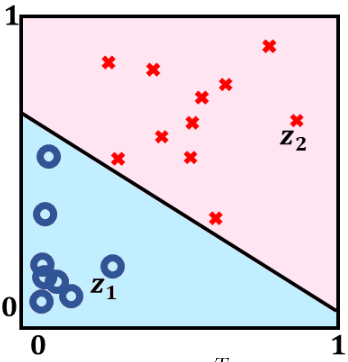
$\mathbf{x}_n \in \mathcal{X}$



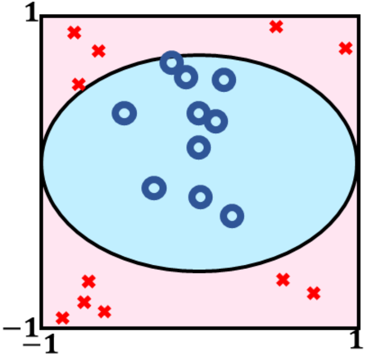
Transform data in $\mathbf{z}_n = \Phi(\mathbf{x}_n) \in \mathcal{Z}$

Example - Transform the Data Nonlinearly

Apply PLA on the transform data set to obtain $\tilde{\mathbf{w}}_{\text{PLA}}$ in space $\mathcal{Z}$



$\tilde{g}(\mathbf{z}) = \text{sign}(\tilde{\mathbf{w}}^T\mathbf{z})$



$g(\mathbf{x}) = \tilde{g}(\Phi(\mathbf{x})) = \text{sign}(\tilde{\mathbf{w}}^T\Phi(\mathbf{x}))$

Circular separator in $\mathcal{X}$ maps to linear separator in $\mathcal{Z}$ and vice versa

Nonlinear Transforms

In general:

$$\mathbf{x} = (x_0, x_1, \cdots, x_d) \xrightarrow{\Phi} \mathbf{z} = (z_0, z_1, \cdots\cdots\cdots, z_{\tilde{d}})$$

Each $z_i = \phi_i(\mathbf{x})$ and dimension $\tilde{d}$ of feature space $\mathcal{Z}$ can be any number.

$$\text{Example:} \qquad \mathbf{z} = (1, x_1, x_2, x_1x_2, x_1^2, x_2^2)$$

Final hypothesis $g(\mathbf{x})$ in $\mathcal{X}$ space:

- Linear classification:

$$\begin{aligned} h(\mathbf{x}) &= \text{sign}\left(\tilde{\mathbf{w}}^T \mathbf{z}\right) \\ h(\mathbf{x}) &= \text{sign}\left(\tilde{\mathbf{w}}^T \Phi(\mathbf{x})\right) \end{aligned}$$

- Linear Regression:

$$h(\mathbf{x}) = \tilde{\mathbf{w}}^T \Phi(\mathbf{x})$$

The Price to Pay

How does the feature transform affect the VC bound?

- The bound remains true by using $d_{\text{VC}}(\mathcal{H}_\Phi)$ if we decide on Φ before seeing the data.
- Denote $\mathcal{H}_\Phi$ to be the hypothesis set in $\mathcal{Z}$

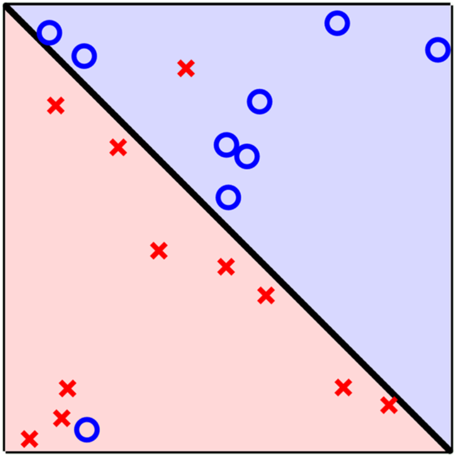
$$\begin{tikzcd}
\mathbf{x} = (x_0, x_1, \cdots, x_d) \arrow[r, "\Phi"] \arrow[d] & \mathbf{z} = (z_0, z_1, \cdots\cdots\cdots, z_{\tilde{d}}) \arrow[d] \\
\mathbf{w} & \tilde{\mathbf{w}}
\end{tikzcd}$$

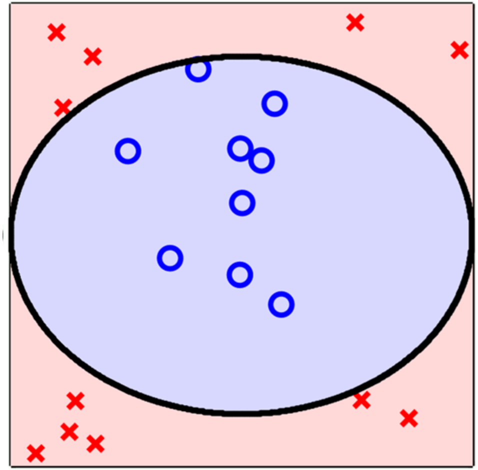
In general, $\tilde{d} > d$

$d_{\text{VC}} = d + 1$ $\qquad$ $d_{\text{VC}} \leq \tilde{d} + 1$

The $\leq$ is because some points $\mathbf{z} \in \mathcal{Z}$ may not be valid transforms of any $\mathbf{x}$ (some dichotomies are not realizable).

Two non-separable cases

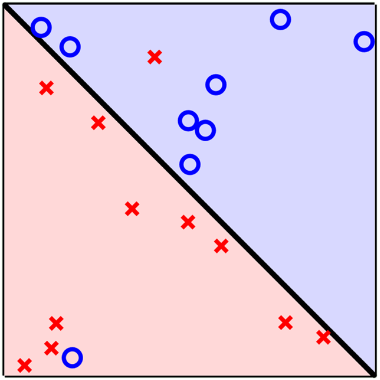




First Case

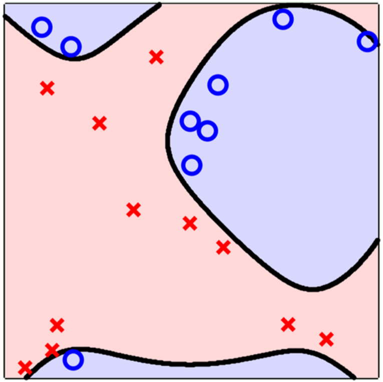
We have two outliers. Two possibilities:

Use a linear model in $\mathcal{X}$;
accept $E_{in} > 0$



Better option: ignore the two outliers.

Insist on $E_{in} = 0$;
go to high-dimensional $\mathcal{Z}$

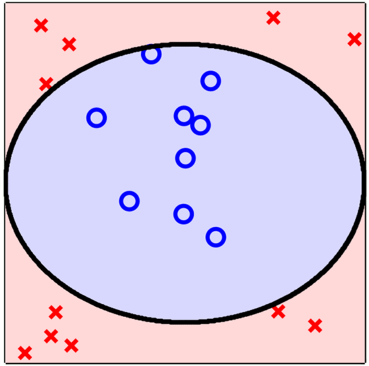


Not a good generalization!
(4th order polynomial fit)

Second Case

There is no chance to approximate the target using a linear model.
Apply: $\mathbf{z} = (1, x_1, x_2, x_1 x_2, x_1^2, x_2^2)$

6 degrees of freedom vs 3 using linear.



Why not: $\mathbf{z} = (1, x_1^2, x_2^2)$
3 degrees of freedom?

or better yet: $\mathbf{z} = (1, x_1^2 + x_2^2)$
2 degrees of freedom?

or even: $\mathbf{z} = (x_1^2 + x_2^2 - 0.6)$
1 degrees of freedom?
No!

Theory of d_{VC} valid if Φ decided before seeing data or trying any algorithm.
VC dimension is charged for previously explored models.

Lesson Learned

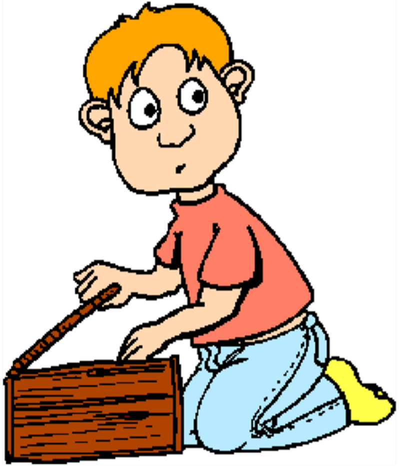
Looking at the data before choosing the model can be hazardous to your E_{out}.

Data snooping:

Decide how to perform after looking at the data
You must account for all of the data snooping you engage in.

However, deciding on Φ based on understanding of the problem does not affect generalization.

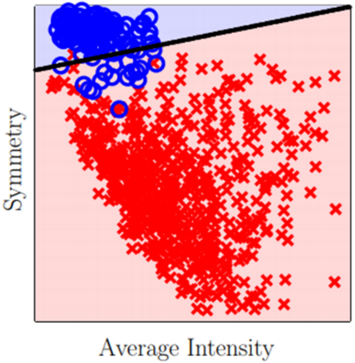
E.g. suggest nonlinear transformation for the 'years in residence'.



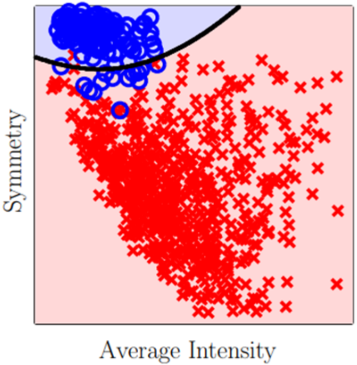
Example - Handwritten Digit Recognition

- Separate digit 1 from all the other digits, using intensity and symmetry.
- A line can roughly separate digit 1 from the rest.

Classification of the digits data using linear and third order polynomial models:



Linear model
$E_{\text{in}} = 2.13\%$ $E_{\text{out}} = 2.38\%$



3rd order polynomial model
$E_{\text{in}} = 1.75\%$ $E_{\text{out}} = 1.87\%$

Maximum Likelihood and Bayes Estimation

Estimation

Estimation is the inference of unknown quantities. Two cases are considered:

1. Quantity is fixed, but unknown – parameter estimation
2. Quantity is random and unknown – random variable estimator

Parameter Estimation

Consider a set of observations forming a vector

$$\mathbf{x} = [x_1, x_2, \cdots, x_N]^T$$

Assumption: The x_i *RV*s come from a known density governed by unknown (but fixed) parameter θ

Objective: Estimate θ. What optimality criteria should be used?

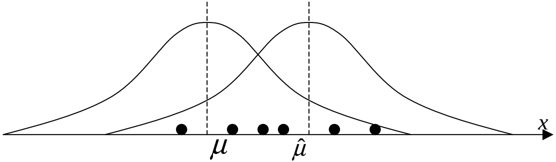
Definition (Maximum Likelihood Estimation)

The maximum likelihood estimate of θ is the value $\hat{\theta}_{\text{ML}}(\mathbf{x})$ which makes the $\mathbf{x}$ observations most likely

$$\hat{\theta}_{\text{ML}}(\mathbf{x}) = \underset{\theta}{\text{argmax}}\, f_{\mathbf{x}|\theta}(\mathbf{x}|\theta)$$

Example

Let $x_i \sim N(\mu, \sigma^2)$. Given N observations, find the ML estimate of μ.



For i.i.d. samples

$$\begin{aligned} f_{\mathbf{x}|\mu}(\mathbf{x}|\mu) &= \prod_{i=1}^{N} f_{x_i|\mu}(x_i|\mu) \\ &= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \qquad \text{[Gaussian case]} \\ &\triangleq \text{likelihood function} \end{aligned}$$

Thus the estimate of the mean it is set as

$$\hat{\mu} = \underset{\mu}{\operatorname{argmax}}\, f_{\mathbf{x}|\mu}(\mathbf{x}|\mu)$$

Interpretation: Set the distribution mean to the value that makes obtaining the observed samples most likely.

Note: Maximizing $f_{\mathbf{x}|\mu}(\mathbf{x}|\mu)$ is equivalent to maximizing any monotonic function of $f_{\mathbf{x}|\mu}(\mathbf{x}|\mu)$. Choosing $\ln(\cdot)$

$$\begin{aligned}
\ln(f_{\mathbf{x}|\mu}(\mathbf{x}|\mu)) &= \ln\left(\prod_{i=1}^{N}\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}\right)\\
&= -N\ln(\sqrt{2\pi\sigma^2}) - \sum_{i=1}^{N}\frac{(x_i-\mu)^2}{2\sigma^2}\\
&= -N\ln(\sqrt{2\pi\sigma^2}) - \sum_{i=1}^{N}\frac{x_i^2}{2\sigma^2} + \mu\sum_{i=1}^{N}\frac{x_i}{\sigma^2} - \sum_{i=1}^{N}\frac{\mu^2}{2\sigma^2}
\end{aligned}$$

Taking the derivative and equating to 0,

$$\frac{\partial \ln(f_{\mathbf{x}|\mu}(\mathbf{x}|\mu))}{\partial\mu} = \sum_{i=1}^{N}\frac{x_i}{\sigma^2} - \frac{N\mu}{\sigma^2} = 0$$

$$\Rightarrow \hat{\mu} = \frac{1}{N}\sum_{i=1}^{N}x_i \overset{\triangle}{=} \text{sample mean}$$

General Maximum Likelihood Result

General Statement: The ML estimate of θ is

$$\hat{\theta}_{\mathrm{ML}}(\mathbf{x}) = \underset{\theta}{\operatorname{argmax}}\, f_{\mathbf{x}|\theta}(\mathbf{x}|\theta)$$

Solution: The ML estimate of θ is obtained as the solution to

$$\left.\frac{\partial}{\partial\theta} f_{\mathbf{x}|\theta}(\mathbf{x}|\theta)\right|_{\theta=\theta_{\mathrm{ML}}} = 0$$

or

$$\left.\frac{\partial}{\partial\theta} \ln[f_{\mathbf{x}|\theta}(\mathbf{x}|\theta)]\right|_{\theta=\theta_{\mathrm{ML}}} = 0$$

- $f_{\mathbf{x}|\theta}(\mathbf{x}|\theta)$ is the likelihood function of θ.
- $\hat{\theta}_{\mathrm{ML}}$ is a *RV* since it is a function of the *RVs* $x_1, x_2, \cdots, x_N$

Historical Note: ML estimation was pioneered by geneticist and statistician Sir R. A. Fisher between 1912 and 1922

Example

The time between customer arrivals at a bar is a *RV* with distribution

$$f_T(T) = \alpha e^{-\alpha T} U(T)$$

Objective: Estimate the arrival rate α based on N measured arrival intervals $T_1, T_2, \cdots, T_N$.

Assuming that the arrivals are independent,

$$\begin{aligned} f(T_1, T_2, \cdots, T_N) &= \prod_{i=1}^{N} f_T(T_i) \\ &= \prod_{i=1}^{N} \alpha e^{-\alpha T_i} = \alpha^N e^{-\alpha \sum\limits_{i=1}^{N} T_i} \\ \Rightarrow \ln[f(T_1, T_2, \cdots, T_N)] &= [N \ln(\alpha) - \alpha \sum_{i=1}^{N} T_i] \end{aligned}$$

Taking the derivative and equating to 0,

$$\begin{aligned}\frac{\partial}{\partial\alpha}\ln[f(T_1,T_2,\cdots,T_N)] &= \frac{\partial}{\partial\alpha}[N\ln(\alpha)-\alpha\sum_{i=1}^{N}T_i] \\ &= \frac{N}{\alpha}-\sum_{i=1}^{N}T_i=0\end{aligned}$$

Solving for α gives the ML estimate

$$\Rightarrow \hat{\alpha}_{\mathrm{ML}}=\frac{1}{\frac{1}{N}\sum_{i=1}^{N}T_i}=\frac{1}{\overline{\overline{T}}}$$

Result: The ML estimate of arival rate for exponentially distributed samples is the reciprocal of the sample mean arrival

Logistic Regression - Outline

Popular method to predict the probability of a binary outcome. Logistic regression measures the relationship between the y "Label" and the $\mathbf{x}$ "Features". The probability is used to predict the label class.

E.g. prediction of heart attacks. There is not certainty, probability fits better than a binary decision.

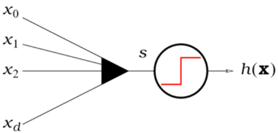
- The model
- Error measure
- Learning algorithm

A Third Linear Model

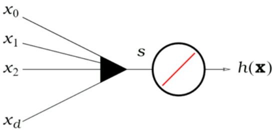
Let

$$s = \sum_{i=0}^{d} w_i x_i$$

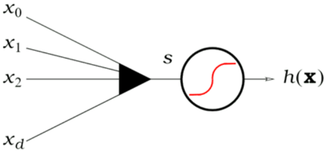
Linear classification	Linear regression	Logistic regression
$h(\mathbf{x}) = \text{sign}(s)$	$h(\mathbf{x}) = s$	$h(\mathbf{x}) = \theta(s)$



Threshold



Identity



θ is a nonlinear function.
Something in between

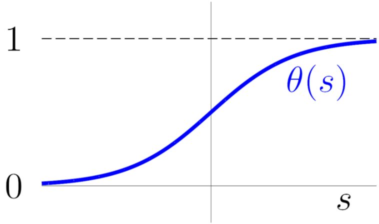
In logistic regression, output is real (like regression) but bounded (like classification)

The Logistic Function θ

The sigmoid function:

$$\theta(s) = \frac{e^s}{1+e^s}$$

- Restricts the output to probability range $[0, 1]$.
- Interpreted as a probability for a binary event (e.g. digit '1' vs digit '5').
- Allows to be uncertain.
- $\theta(s)$ offer analytical and computational advantages.



Soft threshold: uncertainty

There are other popular soft threshold functions.

Probability Interpretation

$h(\mathbf{x}) = \theta(s)$ is interpreted as a probability

Example: Prediction of heart attacks.

- Input $\mathbf{x}$: cholesterol level, age, weight, etc.
- $\theta(s)$: probability of a heart attack
  Predict how likely is to occur given these factors.
- The signal $s = \mathbf{w}^T\mathbf{x}$ "risk score"

Genuine Probability

$$f(\mathbf{x}) = \mathbb{P}[y = +1|\mathbf{x}]$$

Data does not give the value of f. Gives samples generated by this probability. E.g. patients who had heart attacks and who didn't.

Consider data $(\mathbf{x}, y)$ with binary y, generated by a noisy target:

$$\mathbb{P}(y|\mathbf{x}) = \begin{cases} f(\mathbf{x}) & \text{for } y = +1; \\ 1 - f(\mathbf{x}) & \text{for } y = -1. \end{cases}$$

The target $f : \mathbb{R}^d \rightarrow [0, 1]$ is the probability

Goal: Learn $g(\mathbf{x}) = \theta(\mathbf{w}^T\mathbf{x}) \approx f(\mathbf{x})$. How do I choose $\mathbf{w}$?

How close is hypothesis h to f in terms of noisy examples?

Error Measure

For each $(\mathbf{x}, y)$, y is generated by probability $f(\mathbf{x})$

$$y = \begin{cases} +1 & \text{with probability } f(\mathbf{x}); \\ -1 & \text{with probability } 1 - f(\mathbf{x}). \end{cases}$$

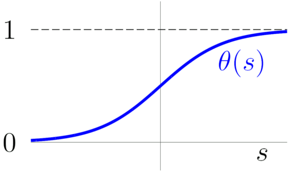
Logistic regression uses a plausible error measure based on **likelihood**:

If $h = f$, how likely to get y from $\mathbf{x}$?

$$P(y|\mathbf{x}) = \begin{cases} h(\mathbf{x}) & \text{for } y = +1; \\ 1 - h(\mathbf{x}) & \text{for } y = -1. \end{cases}$$

Formula for Likelihood

$$P(y|\mathbf{x}) = \begin{cases} h(\mathbf{x}) & \text{for } y = +1; \\ 1 - h(\mathbf{x}) & \text{for } y = -1. \end{cases}$$



Substitute $h(\mathbf{x}) = \theta(\mathbf{w}^T\mathbf{x})$, use the fact $\theta(-s) = 1 - \theta(s)$

$$P(y|\mathbf{x}) = \begin{cases} \theta(\mathbf{w}^T\mathbf{x}) & \text{for } y = +1; \\ \theta(-\mathbf{w}^T\mathbf{x}) & \text{for } y = -1. \end{cases}$$

$$P(y|\mathbf{x}) = \theta(y\mathbf{w}^T\mathbf{x})$$

Likelihood of data set $\mathcal{D} = (\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)$ is

$$\prod_{n=1}^{N} P(y_n|\mathbf{x}_n) = \prod_{n=1}^{N} \theta(y_n\mathbf{w}^T\mathbf{x}_n)$$

since the data points are independently generated.

Maximizing the Likelihood

Use the method of maximum likelihood to select the hypothesis h which maximizes the probability of a given data set.

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\mathrm{argmax}} \quad \prod_{n=1}^{N} \theta(y_n \mathbf{w}^T \mathbf{x}_n)$$

Note: Maximizing a positive function q is equivalent to maximizing any monotonic function of q.

Conveniently choosing $\frac{1}{N}\ln(q)$ to get an error:

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\mathrm{argmax}} \quad \frac{1}{N}\ln\left(\prod_{n=1}^{N} \theta(y_n \mathbf{w}^T \mathbf{x}_n)\right)$$

This is equivalent to

$$\hat{\mathbf{w}} = \underset{\mathbf{w}}{\mathrm{argmin}} \quad -\frac{1}{N}\ln\left(\prod_{n=1}^{N} \theta(y_n \mathbf{w}^T \mathbf{x}_n)\right)$$

Maximizing the Likelihood

$$\begin{aligned}\hat{\mathbf{w}} &= \underset{\mathbf{w}}{\operatorname{argmin}} \quad -\frac{1}{N}\ln\left(\prod_{n=1}^{N}\theta(y_n\mathbf{w}^T\mathbf{x}_n)\right)\\ &= \underset{\mathbf{w}}{\operatorname{argmin}} \quad \frac{1}{N}\sum_{n=1}^{N}\ln\left(\frac{1}{\theta(y_n\mathbf{w}^T\mathbf{x}_n)}\right) \qquad (*)\end{aligned}$$

substituting $\theta(s)=\frac{1}{1+e^{-s}}$ in $(*)$ and treating the cost function in $(*)$ as the ‘in-sample error measure’

$$E_{in}(\mathbf{w})=\frac{1}{N}\sum_{n=1}^{N}\underbrace{\ln\left(1+e^{-y_n\mathbf{w}^T\mathbf{x}_n}\right)}_{e(h(\mathbf{x}_n),y_n)} \qquad \text{“cross-entropy” error}$$

Maximizing the likelihood is equivalent to minimizing E_{in}

Learning Algorithm - How to Minimize E_{in}

For linear regression:

$$E_{in}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{w}^T\mathbf{x}_n - y_n)^2 \qquad \leftarrow \text{closed form solution}$$

Compare to logistic regression,

$$E_{in}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N}\ln\left(1+e^{-y_n\mathbf{w}^T\mathbf{x}_n}\right) \qquad \leftarrow \text{iterative solution}$$

Note: Error measure is small when $y_n\mathbf{w}^T\mathbf{x}_n$ is positive. Encourages **w** to 'classify' each $\mathbf{x}_n$ correctly (i.e. $\text{sign}(\mathbf{w}^T\mathbf{x}_n) = y_n$).

Iterative Method: Gradient Descent

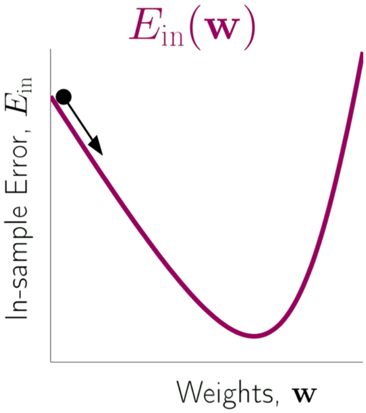
Start at $\mathbf{w}(0)$; take a step along steepest slope.

Fixed step size:

$$\mathbf{w}(1) = \mathbf{w}(0) + \eta(-\nabla E_{in})$$

In logistic regression, cross-error entropy error is a convex function of **w**.

It has unique global minimum.



Iterative Method: Gradient Descent

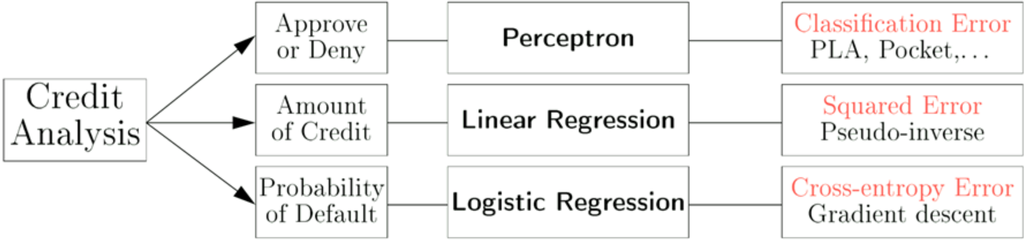
Computing the gradient:

$$\begin{aligned}\nabla E_{in} &= \frac{1}{N}\sum_{n=1}^{N}\frac{1}{1+e^{-y_n\mathbf{w}^T(t)\mathbf{x}_n}}\nabla_{\mathbf{w}}(1+e^{-y_n\mathbf{w}^T(t)\mathbf{x}_n})\\ &= \frac{1}{N}\sum_{n=1}^{N}\frac{e^{-y_n\mathbf{w}^T(t)\mathbf{x}_n}}{1+e^{-y_n\mathbf{w}^T(t)\mathbf{x}_n}}\nabla_{\mathbf{w}}(-y_n\mathbf{w}^T(t)\mathbf{x}_n)\\ &= -\frac{1}{N}\sum_{n=1}^{N}\frac{y_n\mathbf{x}_n}{1+e^{y_n\mathbf{w}^T(t)\mathbf{x}_n}}\end{aligned}$$

Update the weights

$$\mathbf{w}(t+1)=\mathbf{w}(t)-\eta\nabla E_{in}$$

Summary of Linear Models



Example - South African Coronary Heart Disease (CHD)

Data set: A sample of males in a heart-disease high-risk region of the Western Cape, South Africa. Data are taken from a larger dataset, described in Rousseauw et al, 1983, South African Medical Journal.

Risk Factors:

- Tobacco: cumulative tobacco (kg)
- LDL: Low Densiity Lipoprotein cholesterol adiposity
- Famhist: family history of heart disease (Present 1, Absent 0)
- Age: age at onset

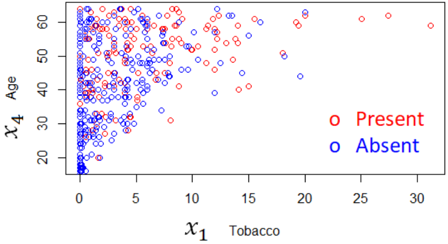
Each data example:

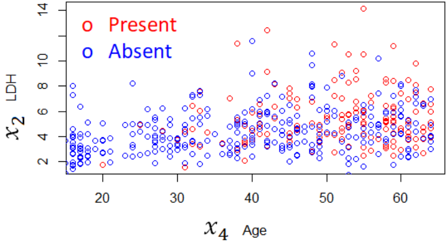
$$\mathbf{x} = [x_0, x_1, x_2, x_3, x_4]^T$$

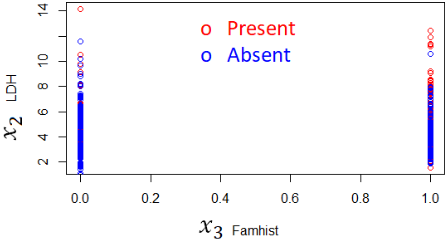
Output Label:

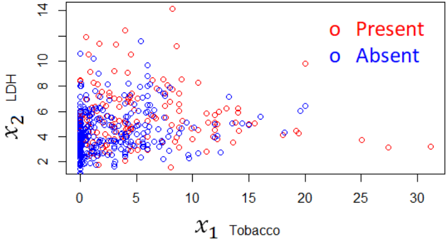
CHD: response (Present 1, Absent 0), Coronary Heart Disease

Analyzing Features









Results from Logistic Regression Fit

Weights:

$$\mathbf{w} = [-4.204, 0.081, 0.168, 0.924, 0.044]^T$$

Given a data point $\mathbf{x} = [1, 12, 5.73, 1, 52]^T$.

The probability of Coronary Heart Disease is:

$$g(\mathbf{x}) = \theta(\mathbf{w}^T\mathbf{x}) = 0.719$$