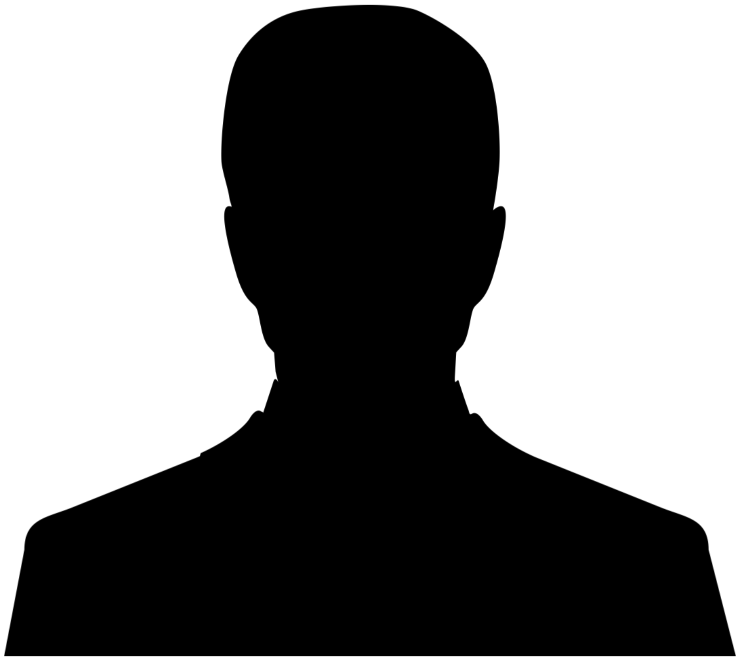
# Logistic Regression
## Lesson Preview

- One of the **most widely used linear classifier** is logistic regression



# Logistic Regression

**When** to use Logistic Regression

- there is a **binary (or nominal) outcome**
- there is **one or more measureable variable**
- when **predictions** about the nominal variable can be made

# Binary Classification

Ad Placement

Feed Ranking

Recommendation Systems

# Binary Classification Quiz

**Check those tasks** that lend themselves to binary classification.

- [x] Spam detection and filtering
- [x] Credit card fraudulent transaction detection
- [x] Medical testing to determine if a patient has a given illness or not

# Definitions

$x$ and $\theta$ have the same dimensions.

$$x = (x_1, x_2, x_3, x_4, \ldots x_d) \qquad \theta = (\theta_1, \theta_2, \theta_3, \ldots, \theta_d)$$

$$y = +1 \text{ or } y = -1$$

The cross product:

$$< x, \theta > = \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_d x_d$$

# Definitions

Given a vector of features x, assign a label `y` of `+1` or `-1`

$(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), (x^{(3)}, y^{(3)}), \ldots, (x^{(n)}, y^{(n)})$

- Each $x^{(i)}$ is a vector
- Each $y^{(i)}$ is its +1 or -1 label.

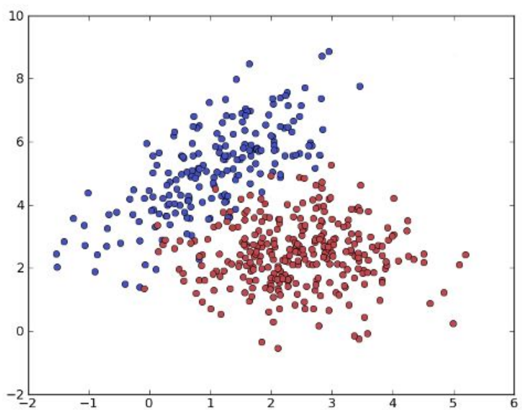
# LinkedIn Example

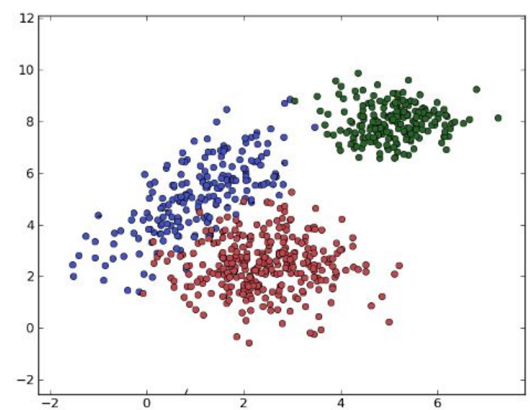
| | User Characteristics | | | | |
|---|---|---|---|---|---|
| | $x_1$ | $x_2$ | $x_3$ | $x_4$ | y |
| | | | | | **click** |
| | | | | | **no click** |

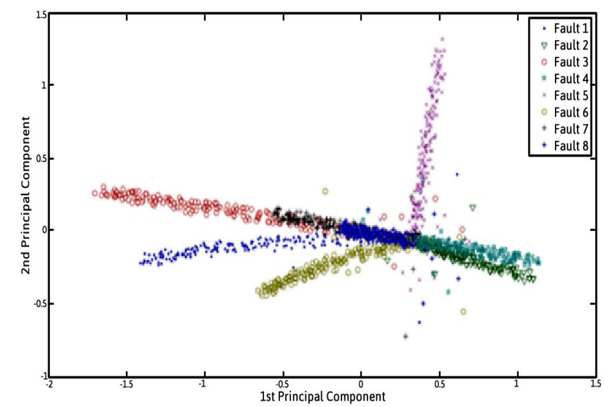
Training data = table (dataframe) of rows representing training set examples and labels

# Visual Binary Classification Quiz

**Check the plots** whose data can be binary classified.







# Linear Classifiers

Predicted label of x

Linear classifiers have the form: $\mathbf{y} = \mathbf{sign}(\langle \theta, x \rangle)$

Parameter vector of x

# Linear Classifiers

| Inner product | Predicted Class |
| --- | --- |
| $<\theta, x>$ positive | +1 |
| $<\theta, x>$ negative | -1 |

# Predicted Classes Quiz

**Fill in the blanks.** Assume a classification goal of predicting whether a patient has cancer or not

What would be the vector of characteristics?

levels of radiation exposure, age, gender, BMI

What would be the vector of parameters?

(2, 1, 0, 0.5)

Assign the scalar values to the outcomes:

| | | |
|---|---|---|
| y = +1 when cancer is | ☑ present | ☐ not present |
| y = -1 when cancer is | ☐ present | ☑ not present |

# Why Linear Classifiers?

Easy to **train**

**Predict labels** very quickly at serve time

**Well known statistical theory** of linear classifiers leading to effective modeling strategies

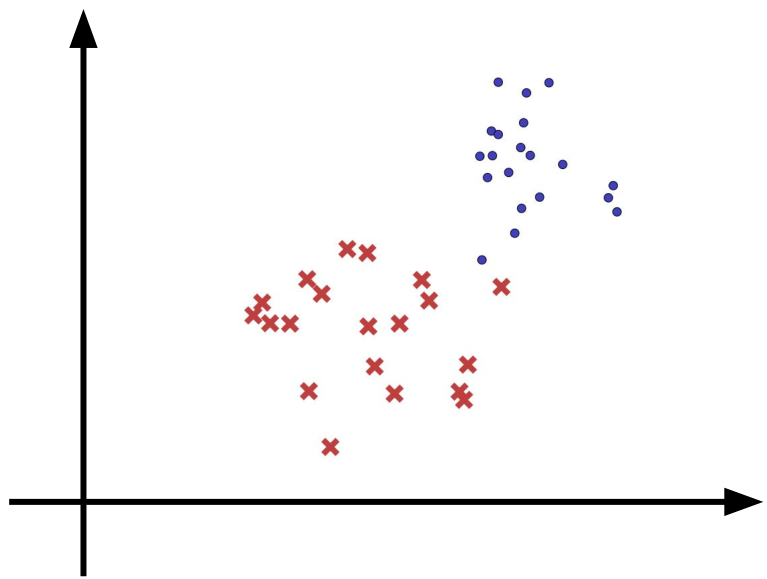
# Why Linear Classifiers?

Linear classifiers **excel at high dimensions** due to:

- their simplicity
- attractive computational load
- nice statistical properties

**For data visualization:** assume dimension = 2

# The Linear Plane

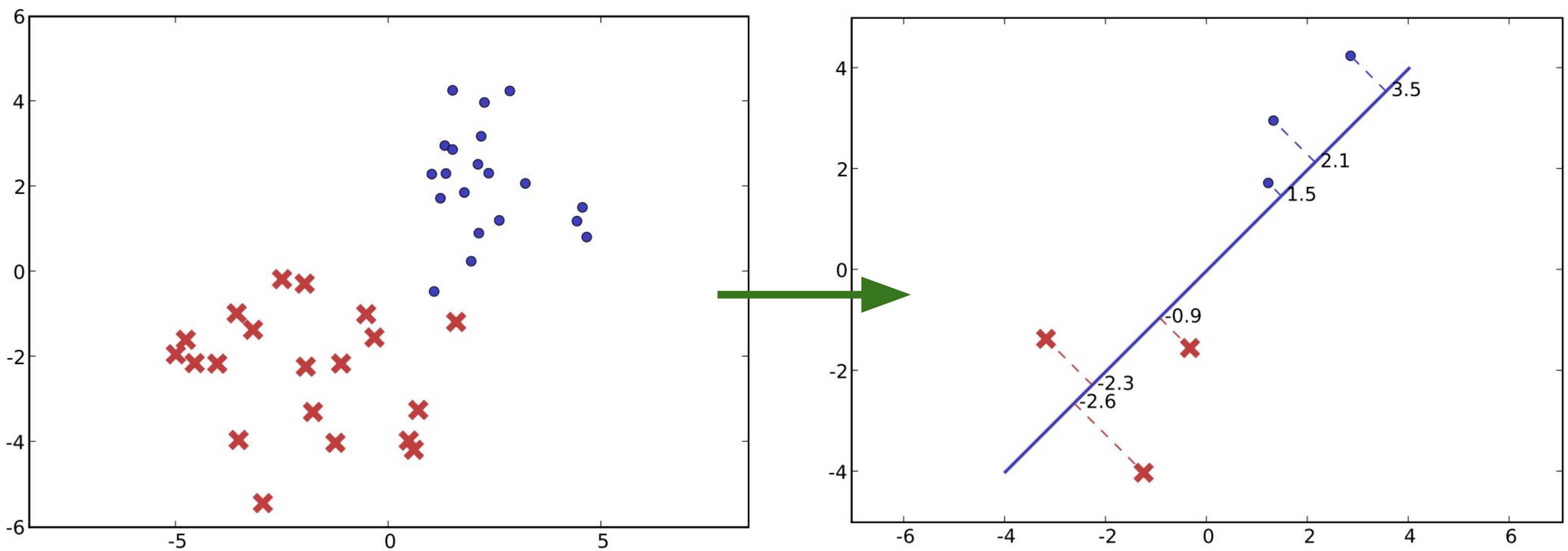


# The Linear Plane

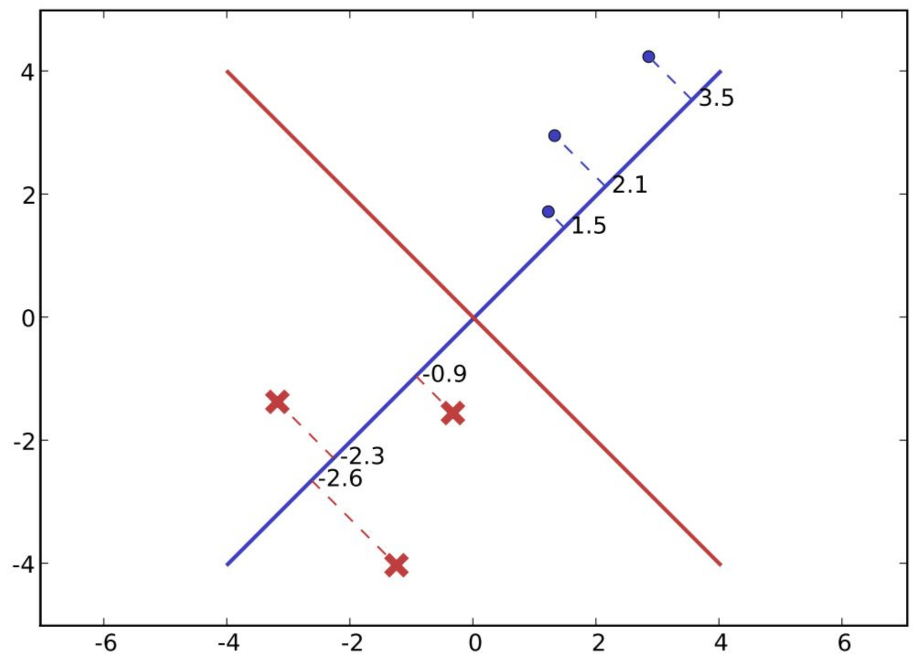
- Given $x = (x_1, x_2)$
- The classification is: $\text{sign}(ax_1 + bx_2 + c)$

| Sign( $ax_1 + bx_2 + c$) | Classification is |
|---|---|
| positive | +1 |
| negative | -1 |

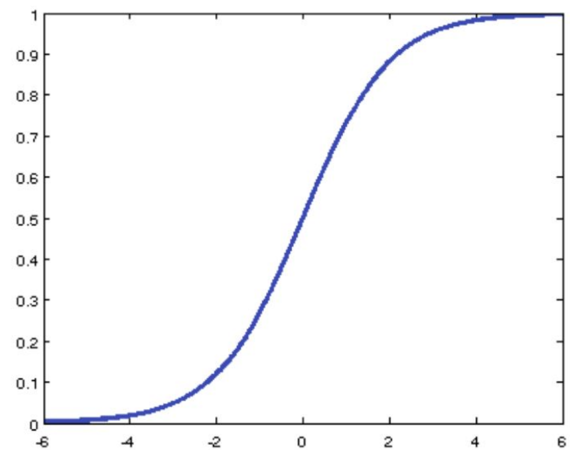
# The Linear Plane



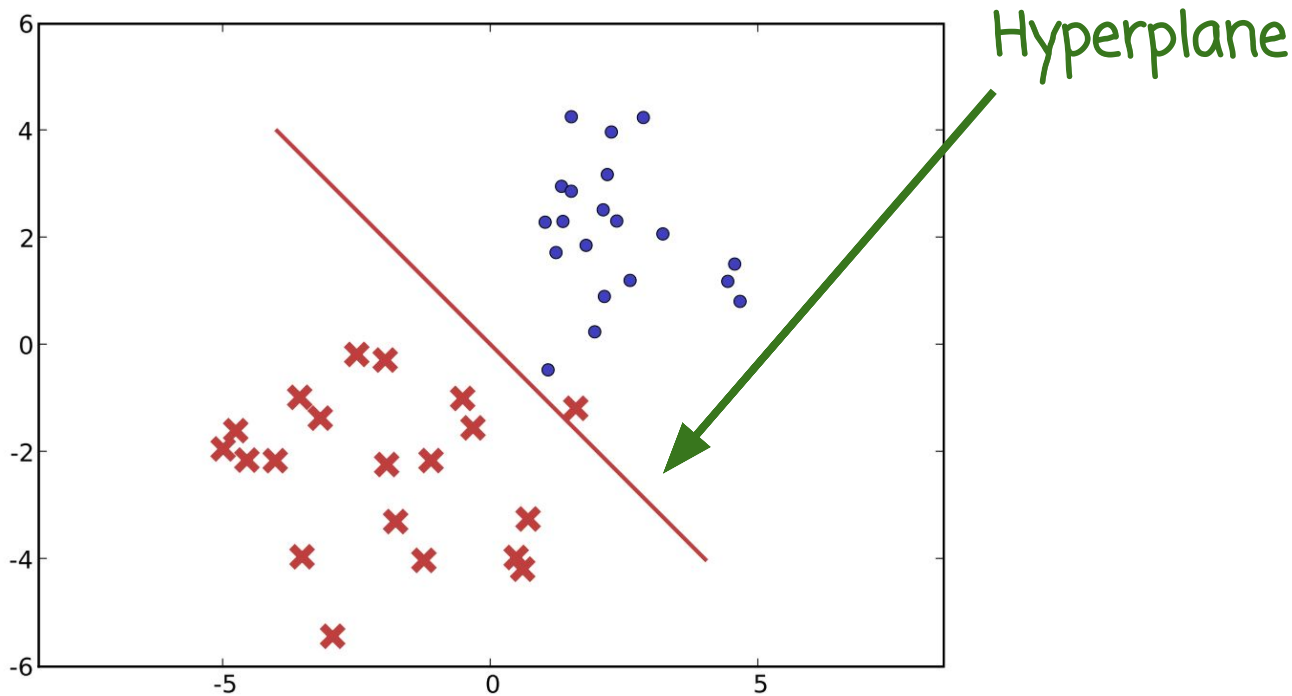
# The Linear Plane



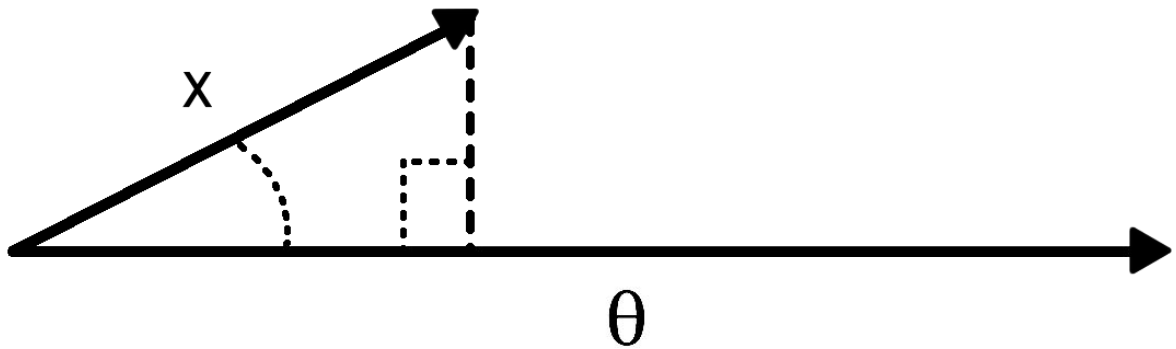
# The Linear Plane



# The Linear Plane



The Linear Plane



# Decision Boundary Quiz

Should the decision boundary pass through the origin?

Not necessarily. If we do not require that (by assigning one features to the value 1) the classifier becomes considerably more powerful.

# Bias Term

$$x_1\theta_1 + \dots + x_d\theta_d + c = \langle x, \theta \rangle \cancel{+ c}$$

$$\langle x, \theta \rangle$$

# Increasing Data Dimensionality

**Two Dimensional Data Vector**

$$x=(x_1,x_2)$$

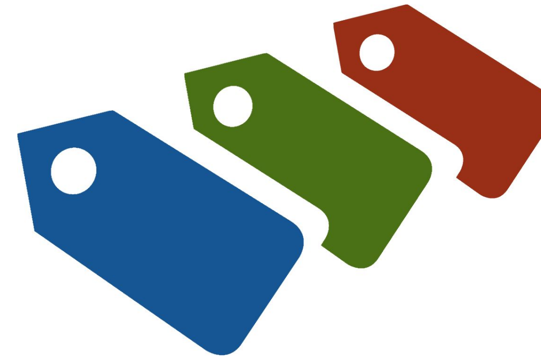
**Six Dimensional Data Vector**

$$\hat{x}=(1,x_1,x_2,x_1^2,x_2^2,x_1x_2)$$

# Increasing Data Dimensionality

| | | |
|---|---|---|
| **Transformed vector** | $(\hat{x}^{(1)}, y^{(1)}), \ldots, (\hat{x}^{(d)}, y^{(n)})$ | **Classifier is linear** in the coordinate system of $\hat{x}$ |
| **Original data** | $(x^{(1)}, y^{(1)}), \ldots, (x^{(d)}, y^{(n)})$ | But **classifier is non-linear** in the original coordinate system of x |

# Classifiers

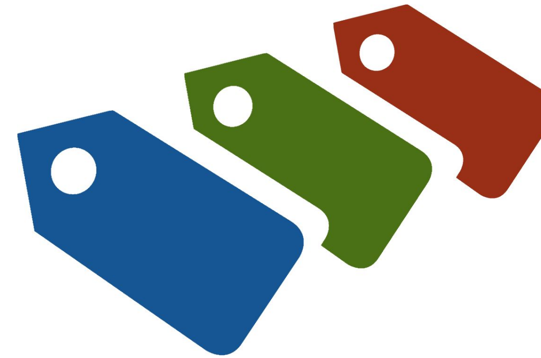


**Classifiers define a map from a vector of features x to a label.**
Sometimes we get a confidence, and sometimes that confidence is also the probability that the label is 1.

# Classifiers

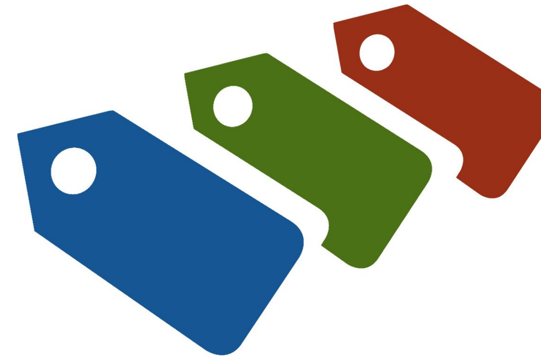
**Probabilistic classifiers** provide that tool by defining the probabilities of the labels +1 and -1 given the feature vector x..

# Classifiers



$$p(Y = +1 | X=x) + p(Y= -1 | X=x) = 1$$
$$p(Y = +1 | X=x) > 0$$
$$p(Y = -1 | X=x) > 0$$

# Classifiers



The probability that a given element of vector x will be classified as '1':

$$p_\theta(Y=1 \mid X=x)$$

The probability that a given element of vector x will be classified as '-1':

$$p_\theta(Y=-1 \mid X=x)$$

# Label Probability Quiz

Type the letter that corresponds to the correct answer in the textbox.

A. 1

B. 0

C. $p_\theta(Y=-1 \mid X=x)$

D. $p_\theta(Y=1 \mid X=x)$

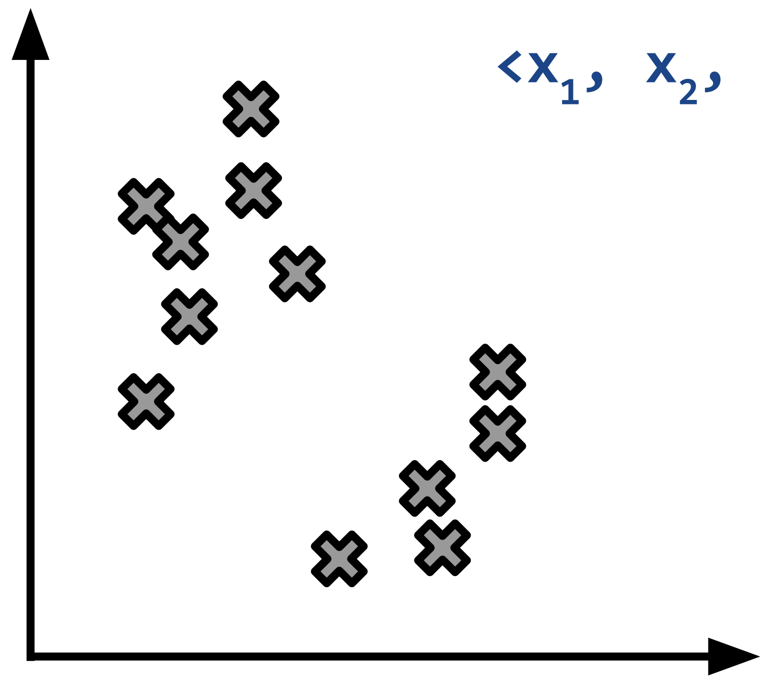
$1 - p_\theta(Y=1 \mid X=x) = \boxed{C}$

$1 - p_\theta(Y=-1 \mid X=x) = \boxed{D}$

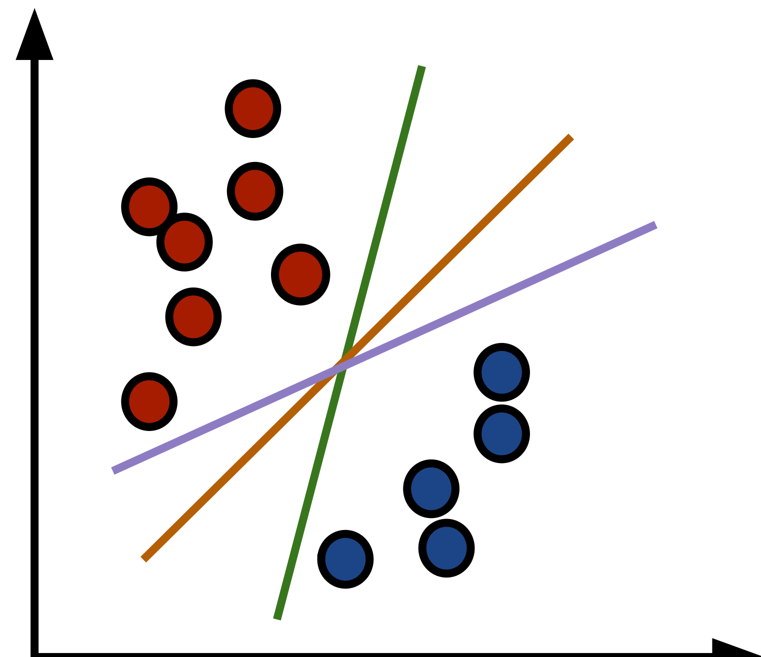
$p_\theta(Y=1 \mid X=x) + p_\theta(Y=-1 \mid X=x) = \boxed{A}$

# Maximum Likelihood Estimator (MLE)

$\langle x_1, x_2, x_3, \ldots, x_d \rangle$



# Maximum Likelihood Estimator (MLE)



# Maximum Likelihood Estimator (MLE)

# Maximum Likelihood Estimator (MLE)

**Frequentists:** MLE uses pairs of feature vectors and labels $(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(n)}, y^{(n)})$ **usually from historic data** to estimate correct classifier or nature

**Bayesians:** A single classifier cannot represent the "truth". **Estimate the revised probability** that each classifier is correct and use them all

# Maximum Likelihood Estimator (MLE)

**Frequentists:** MLE uses pairs of feature vectors and labels $(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(n)}, y^{(n)})$ **usually from historic data** to estimate correct classifier or nature

# MLE Defined

$$
\begin{aligned}
\hat{\theta}_{\mathrm{MLE}} &= \arg\max_{\theta} \quad p_\theta(Y = y^{(1)} | X = x^{(1)}) \cdots p_\theta(Y = y^{(n)} | X = x^{(n)}) \\
&= \arg\max_{\theta} \quad \log p_\theta(Y = y^{(1)} | X = x^{(1)}) + \cdots + \log p_\theta(Y = y^{(n)} | X = x^{(n)})
\end{aligned}
$$

# MLE Defined

**Justifications** for using MLE:

It converges to the optimal solution in the limit of large data **(consistency)**

Data is generated based on the **logistic regression model family and n → infinity while d is fixed**

# MLE Defined

Justifications for using MLE:

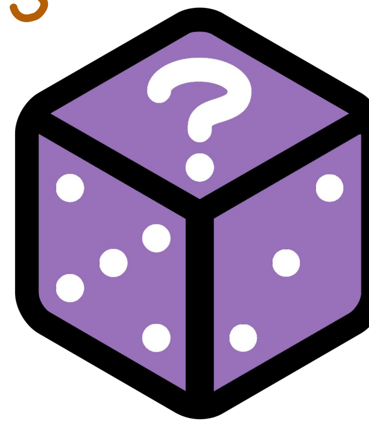
- It converges to the optimal solution in the limit of large data (consistency)
- The convergence occurs at the fastest possible rate of convergence (statistical efficiency)

# MLE Quiz

**Describe a computational procedure** for reaching the value x for which f(x) is at a maximum. Does it scale to high dimensions x?

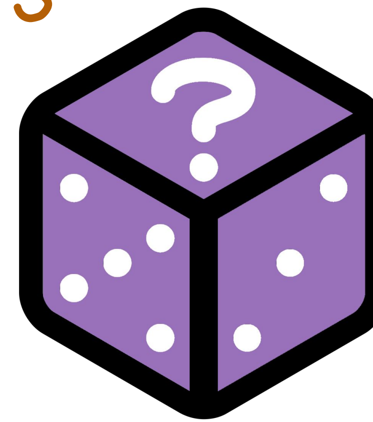
- Compute f(x) on a grid of all possible values and find the maximum.
- Do this for scalars x or low dimensional vectors x.
- (It does not scale to higher dimensions. An alternative technique that does scale is gradient ascent.)

# Probabilistic Classifiers



Logistic regression is the **most popular probabilistic classifier.**

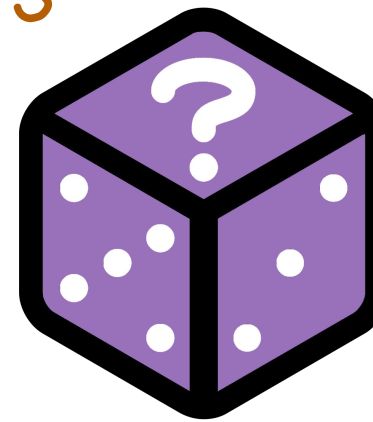
# Probabilistic Classifiers



$$p(Y = y | X = x) = \frac{1}{1 + \exp(y\langle\theta, x\rangle)}$$

y = +1 or y= -1

# Probabilistic Classifiers



$p(Y = 1|x) + p(Y = -1|x) = 1$:

$$\begin{aligned}\frac{1}{1+\exp(\langle\theta,x\rangle)} + \frac{1}{1+\exp(-\langle\theta,x\rangle)} &= \frac{1+\exp(\langle\theta,x\rangle)+1+\exp(-\langle\theta,x\rangle)}{(1+\exp(\langle\theta,x\rangle))(1+\exp(-\langle\theta,x\rangle))} \\ &= \frac{1+\exp(\langle\theta,x\rangle)+1+\exp(-\langle\theta,x\rangle)}{1+\exp(\langle\theta,x\rangle)+\exp(-\langle\theta,x\rangle)+1} = 1\end{aligned}$$

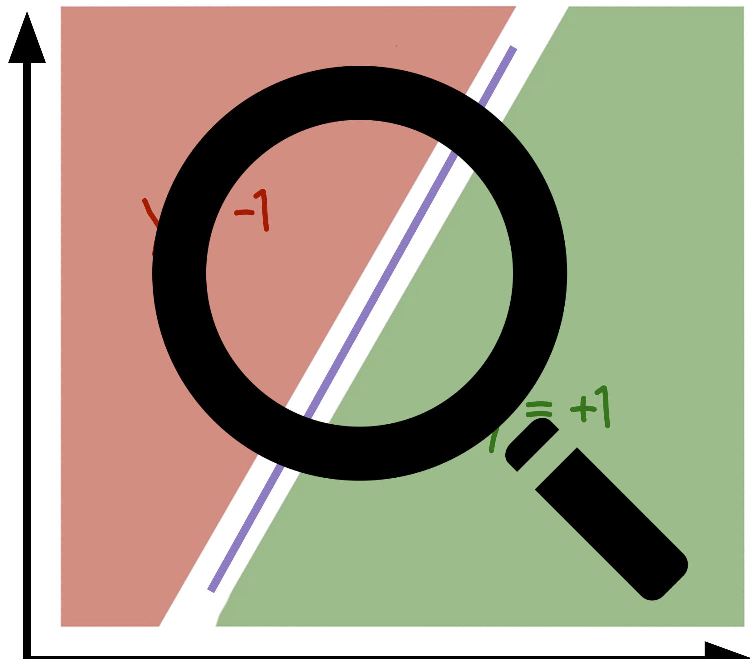
# Decision Boundary Quiz

Where should the **decision boundary be placed**?

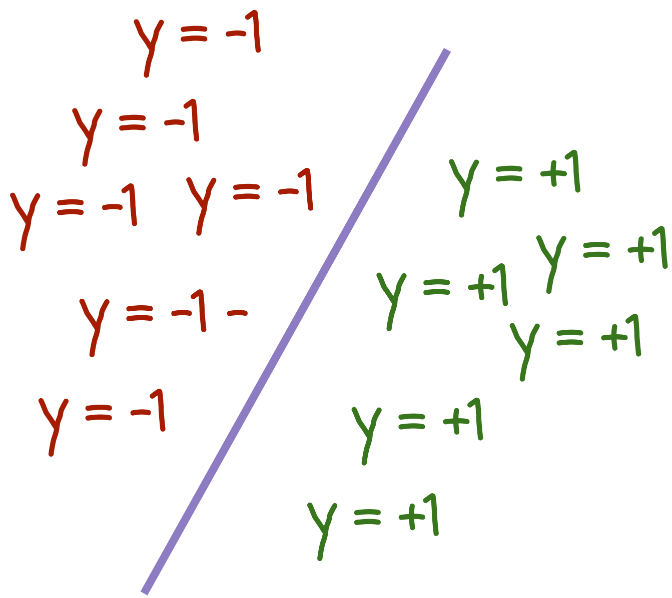
- ☐ It is the set of points where $p(Y = 1|x) < p(Y = -1|x)$
- ☐ It is the set of points where $p(Y = 1|x) > p(Y = -1|x)$
- ☑ It is the set of points where $p(Y = 1|x) = p(Y = -1|x) = 0.5$

$$
\begin{aligned}
\frac{1}{2} &= \frac{1}{1 + \exp(\langle \theta, x \rangle)} \\
1 + \exp(\langle \theta, x \rangle) &= 2 \\
\langle \theta, x \rangle &= \log(1) = 0.
\end{aligned}
$$

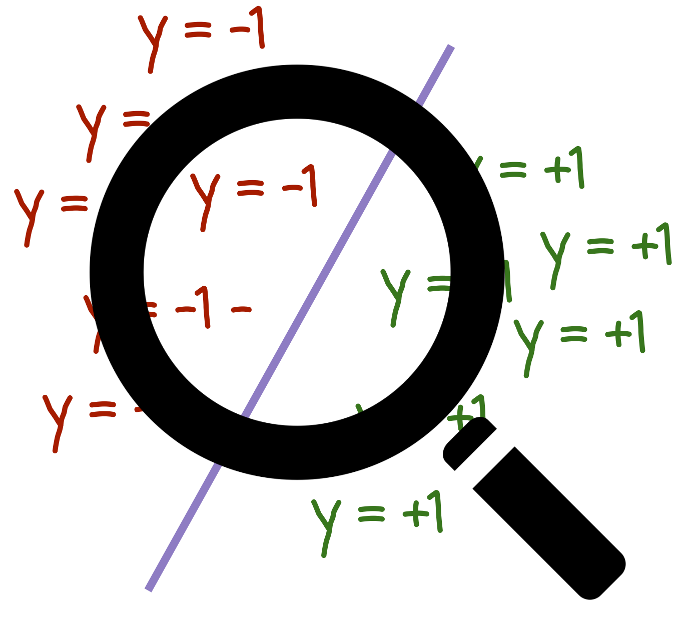
# The Decision Boundary & the Hyperplane



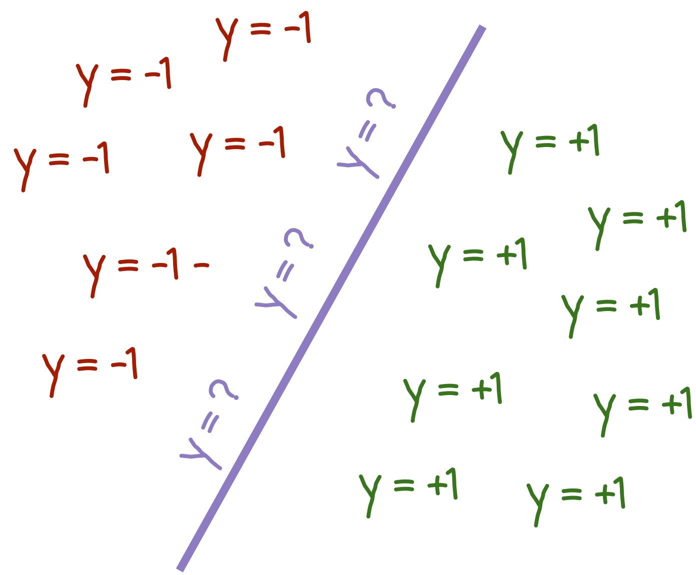
# The Decision Boundary & the Hyperplane



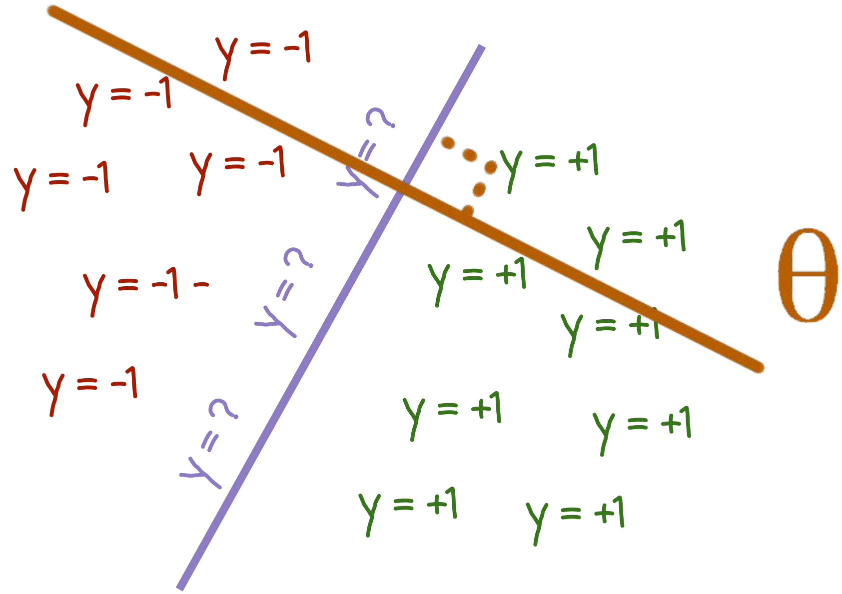
# The Decision Boundary & the Hyperplane



# The Decision Boundary & the Hyperplane



# The Decision Boundary & the Hyperplane



# Prediction Confidence

| Task | Use |
|---|---|
| **Predict the label** associated with a feature vector x | prediction rule sign $(\theta, x)$ |
| **Measure of confidence** of that prediction | $p(Y = y \mid X = x) = 1/(1 + \exp(y\langle\theta, x\rangle))$ |

# MLE and Iterative Optimization

$$p(Y = y | X = x) = \frac{1}{1 + \exp(y\langle \theta, x \rangle)}$$

$$\begin{aligned}
\hat{\theta}_{\mathrm{MLE}} &= \arg\max_{\theta} \quad p_\theta(Y = y^{(1)} | X = x^{(1)}) \cdots p_\theta(Y = y^{(n)} | X = x^{(n)}) \\
&= \arg\max_{\theta} \quad \log p_\theta(Y = y^{(1)} | X = x^{(1)}) + \cdots + \log p_\theta(Y = y^{(n)} | X = x^{(n)})
\end{aligned}$$

# MLE and Iterative Optimization

$$
\begin{aligned}
\hat{\theta}_{\mathrm{MLE}} &= \arg\max_{\theta} \sum_{i=1}^{n} \log \frac{1}{1+\exp(y^{(i)}\langle \theta, x^{(i)} \rangle)} = \arg\max_{\theta} \sum_{i=1}^{n} -\log\left(1+\exp(y^{(i)}\langle \theta, x^{(i)} \rangle)\right) \\
&= \arg\min_{\theta} \sum_{i=1}^{n} \log\left(1+\exp(y^{(i)}\langle \theta, x^{(i)} \rangle)\right)
\end{aligned}
$$

# Gradient Descent

a. initialize the dimensions of $\theta$ to random values

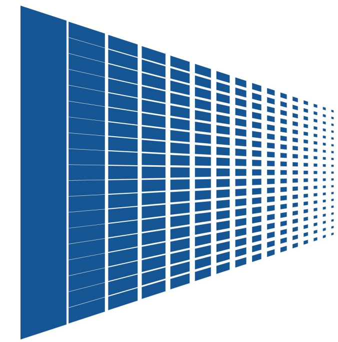
b. for `j = 1, ..., d` update

$$\theta_j \leftarrow \theta_j - \alpha \frac{\partial \sum_{i=1}^{n} \log\left(1+\exp(y^{(i)} \langle \theta, x^{(i)} \rangle)\right)}{\partial \theta_j}$$

c. repeat the update (step b) until the updates becomes smaller than a threshold

Let $\alpha$ decay as the gradient descent iterations increase

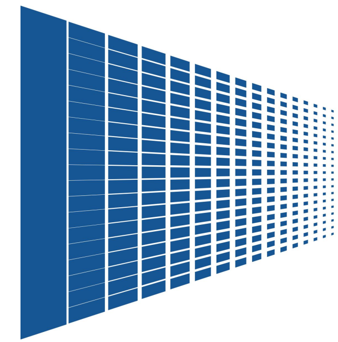
# Gradient Descent



$$\theta_j \leftarrow \theta_j - \alpha \frac{\partial \sum_{i=1}^{n} \log\left(1+\exp(y^{(i)} \langle \theta, x^{(i)} \rangle)\right)}{\partial \theta_j}$$

When the data vectors $x^{(i)}$ are sparse, the computation of **the partial derivative can be made particularly fast**.

# Gradient Descent



$$\arg\min_{\theta} \sum_{i=1}^{n} \log\left(1 + \exp(y^{(i)} \langle \theta, x^{(i)} \rangle)\right)$$

The single global maximum will be reached regardless of the starting point.

Although the maximum could be at $\theta_j \to \pm\infty$ for some $j$

# Stochastic Gradient Descent

| **Amount of Data** | **Preferred Technique** |
|---|---|
| **non-massive data** | Gradient descent |
| **massive data** | Stochastic gradient descent |

# Stochastic Gradient Descent

a. initialize the dimensions of $\theta$ vector to random values

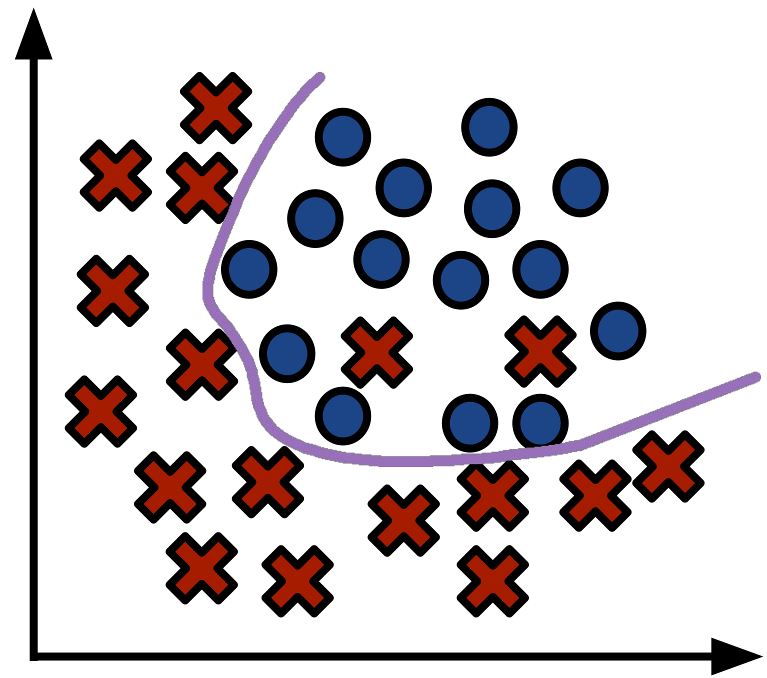
b. pick one labeled data vector $(x^{(i)}, y^{(i)})$ randomly, and update each

$$j = 1, \ldots, d:\ \theta_i \leftarrow \theta_j - \alpha \frac{\partial \log\left(1+\exp(y^{(i)}\langle\theta, x^{(i)}\rangle)\right)}{\partial \theta_j}$$

c. repeat step b until the updates of the dimensions of become too small (reducing alpha as the number of iteration increases)

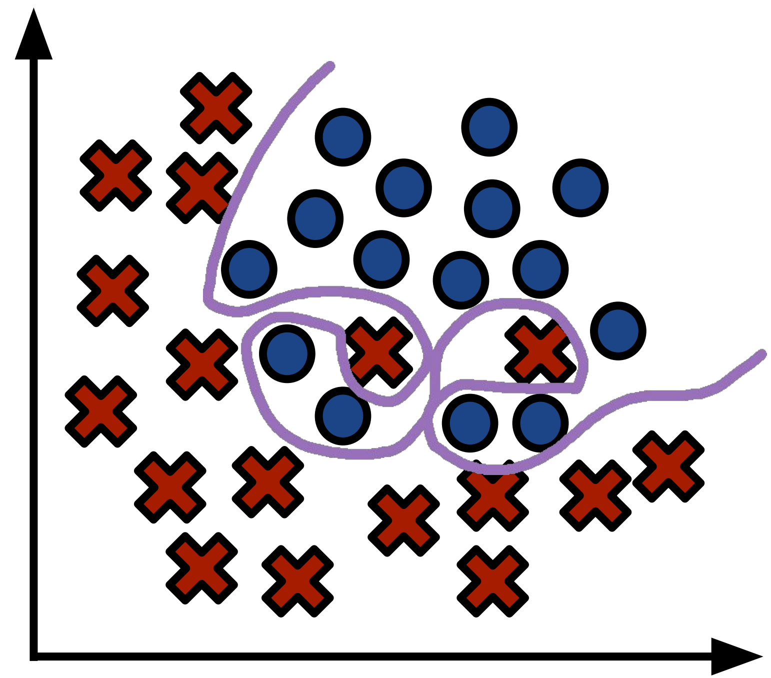
# Overfitting

A **good model selection** may not be perfect on the training data, but it generalizes to new data



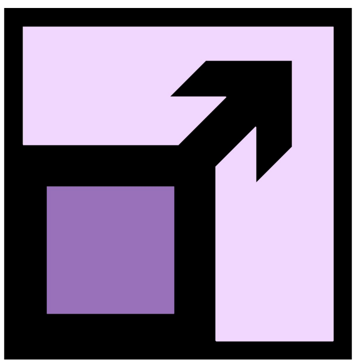
# Overfitting

**Overfitting:** fitting a model too aggressively to the training data **may not generalize well**



# Overfitting

When x is high dimensional **overfitting becomes a problem**:



**Too many parameters** to estimate from the available labeled data

The classifier **fits random noise patterns** that exist in the data

## Overfitting

**Example:**

$d = 10^6$ and $n = 2$

(d = dimension of x vector,
n = dimension of the training data)

## Overfitting

Exan

d = 0⁶ d = 2

(d = dimension of x vector,
n = dimension of the training data)

# Regularization

Regularization terms **added to the maximum likelihood** cost function

$$\beta\theta_1^2 + \cdots + \beta\theta_d^2 \quad \text{or} \quad \beta|\theta_1| + \cdots + \beta|\theta_d|$$

High values of $\theta$ are **penalized**
**MLE is tempered** from achieving high values of $\theta_1, \theta_2, \ldots, \theta_d$

$\beta$ is selected through **experimentation**
$\beta$ should be **close to the optimal value**