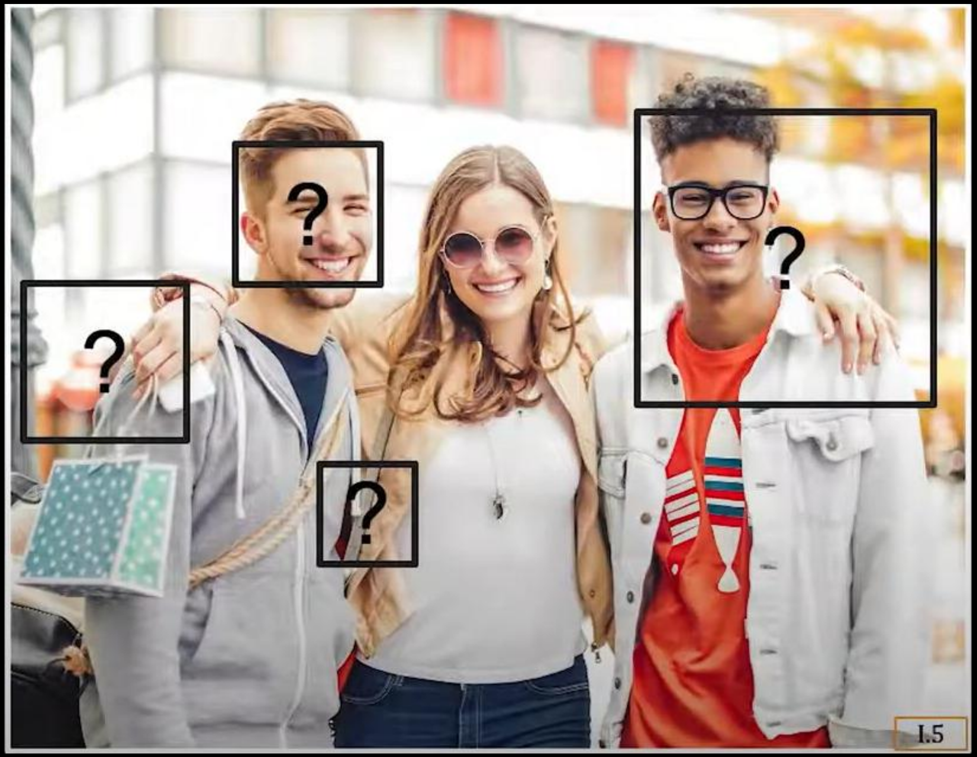
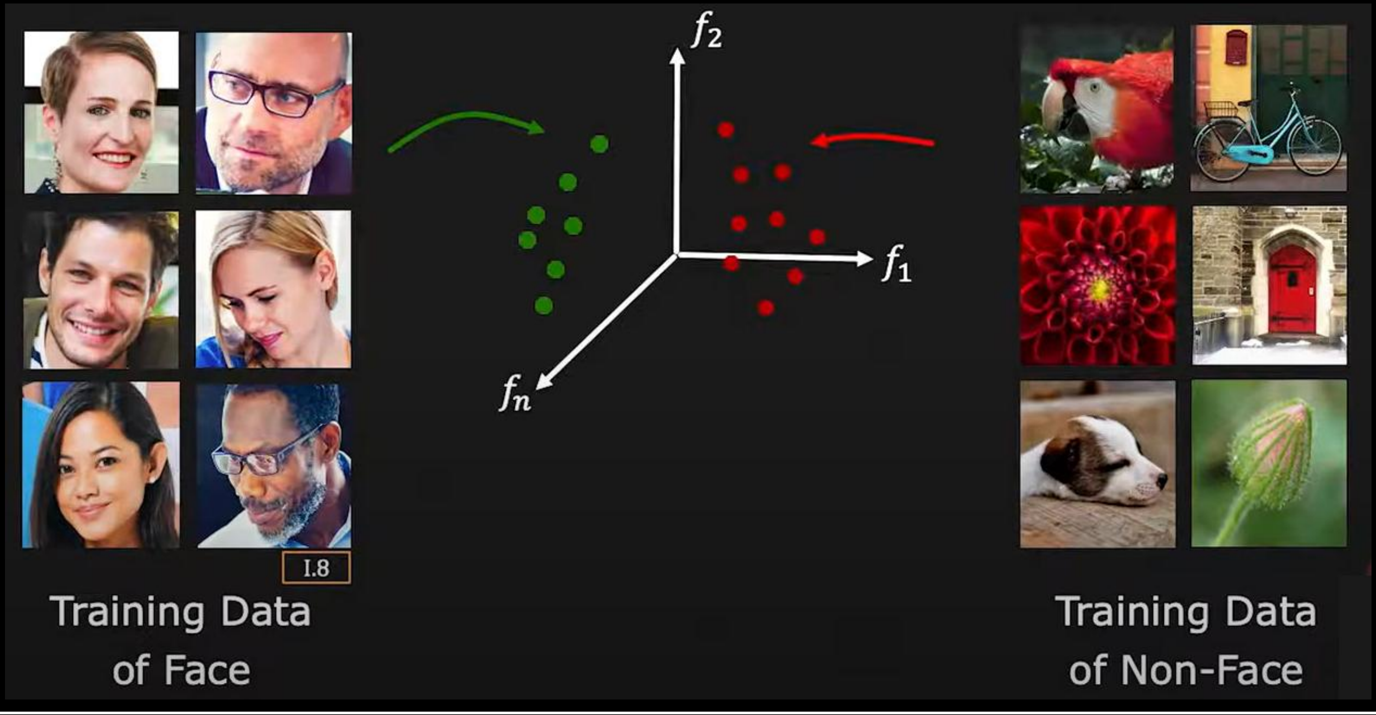
FACE DETECTION

Classifier for face detection



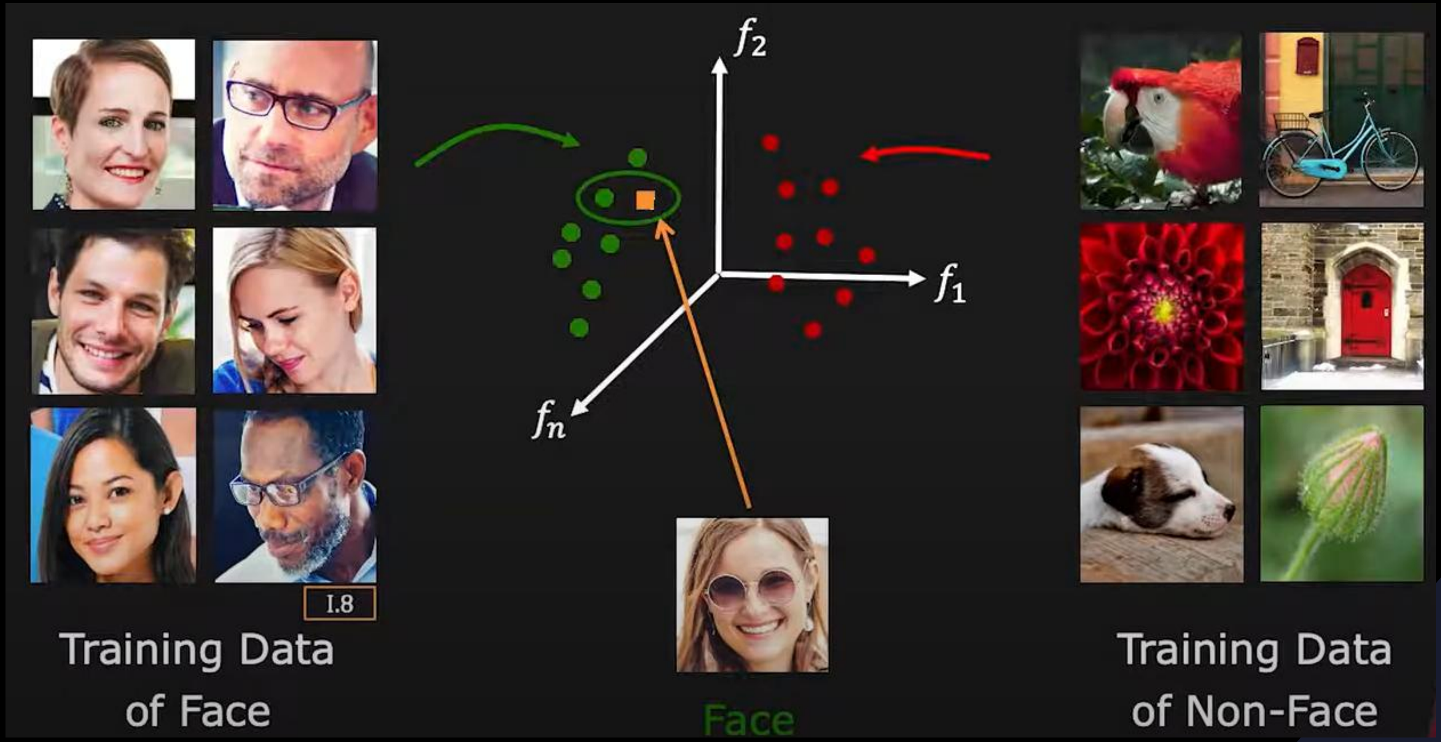
Feature Space

- Haar features f (a vector) at a pixel is a point in an n-D space, $f \in R^n$



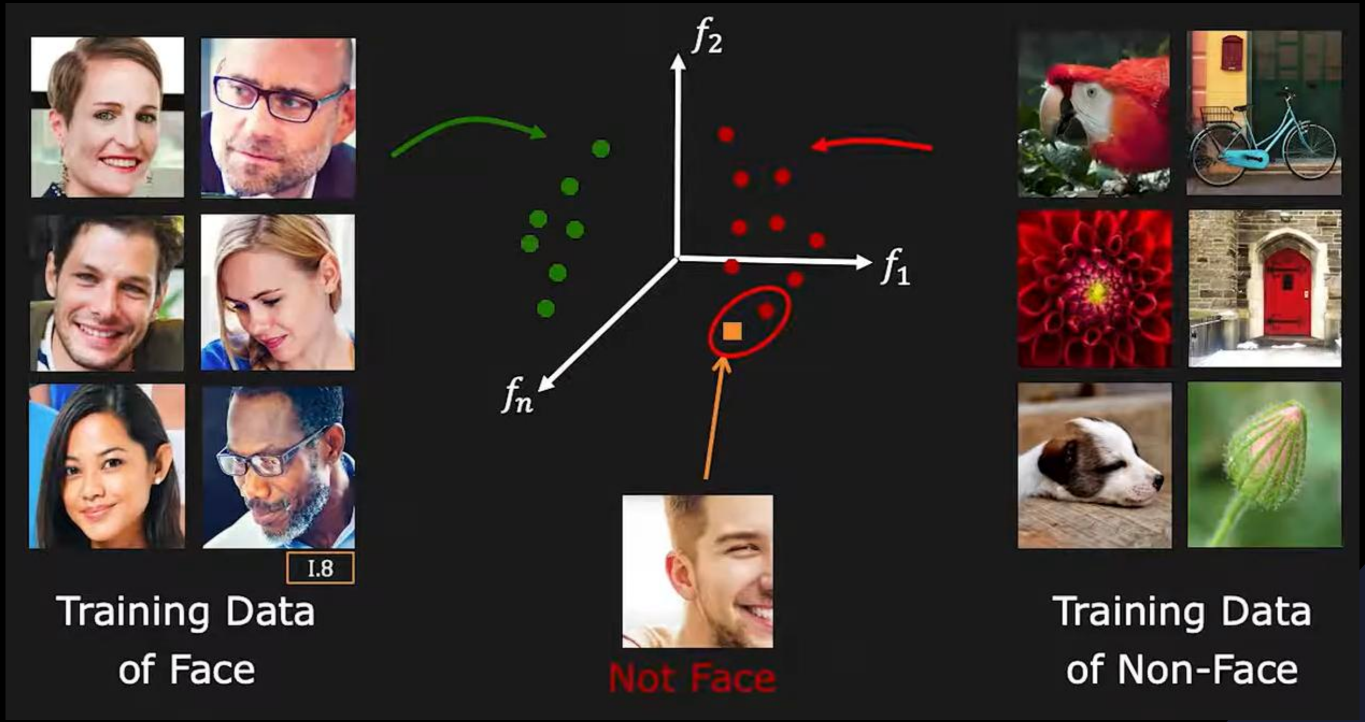
Nearest Neighbour Classifier

- Find the nearest training sample and assign the label



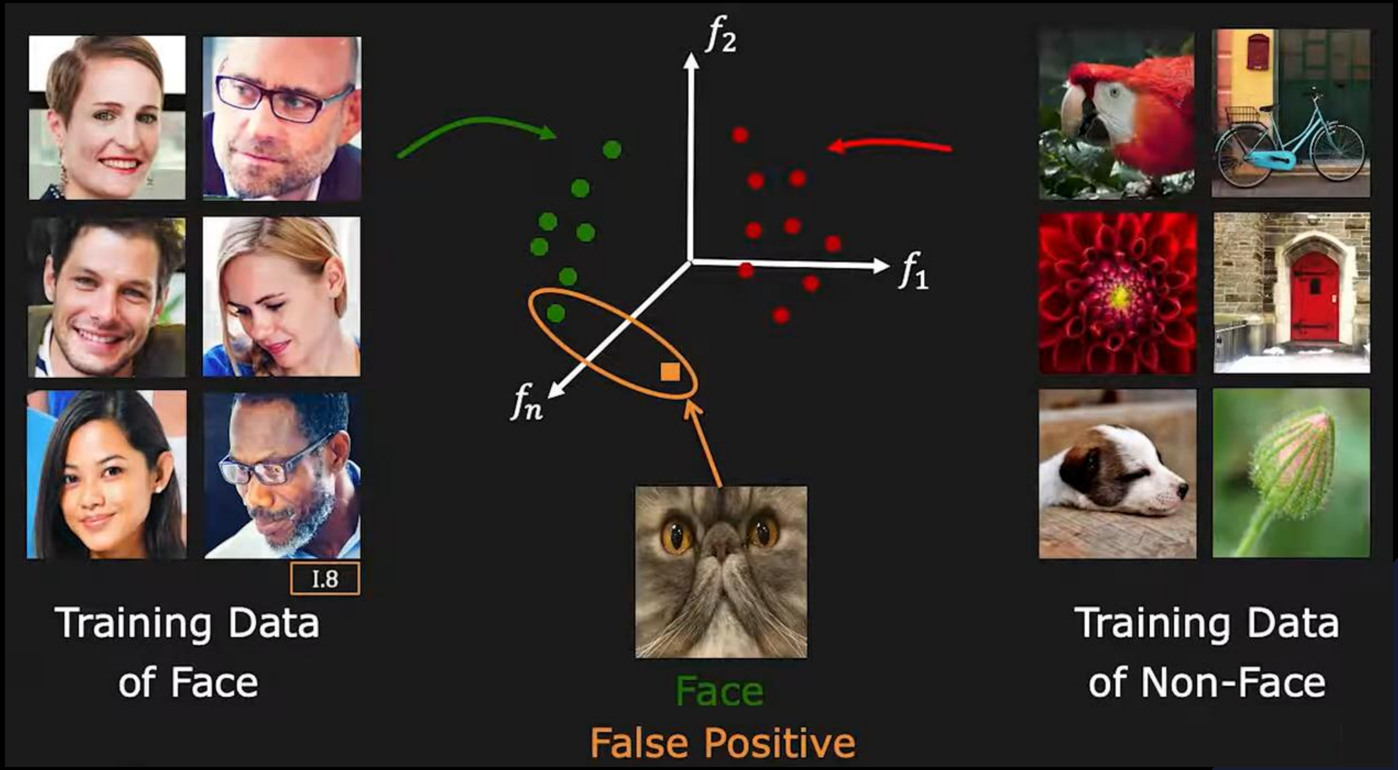
Nearest Neighbour Classifier

- Find the nearest training sample and assign the label



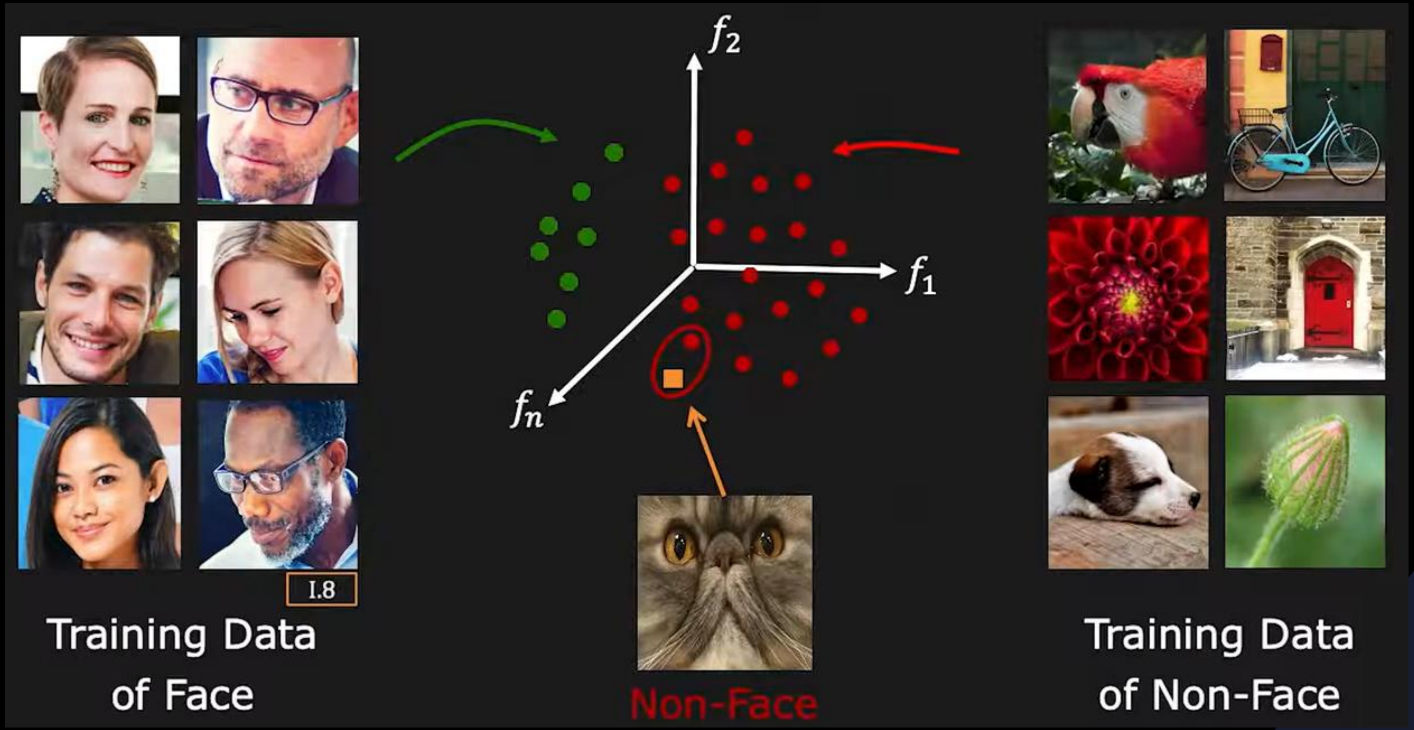
Nearest Neighbour Classifier

- Find the nearest training sample and assign the label

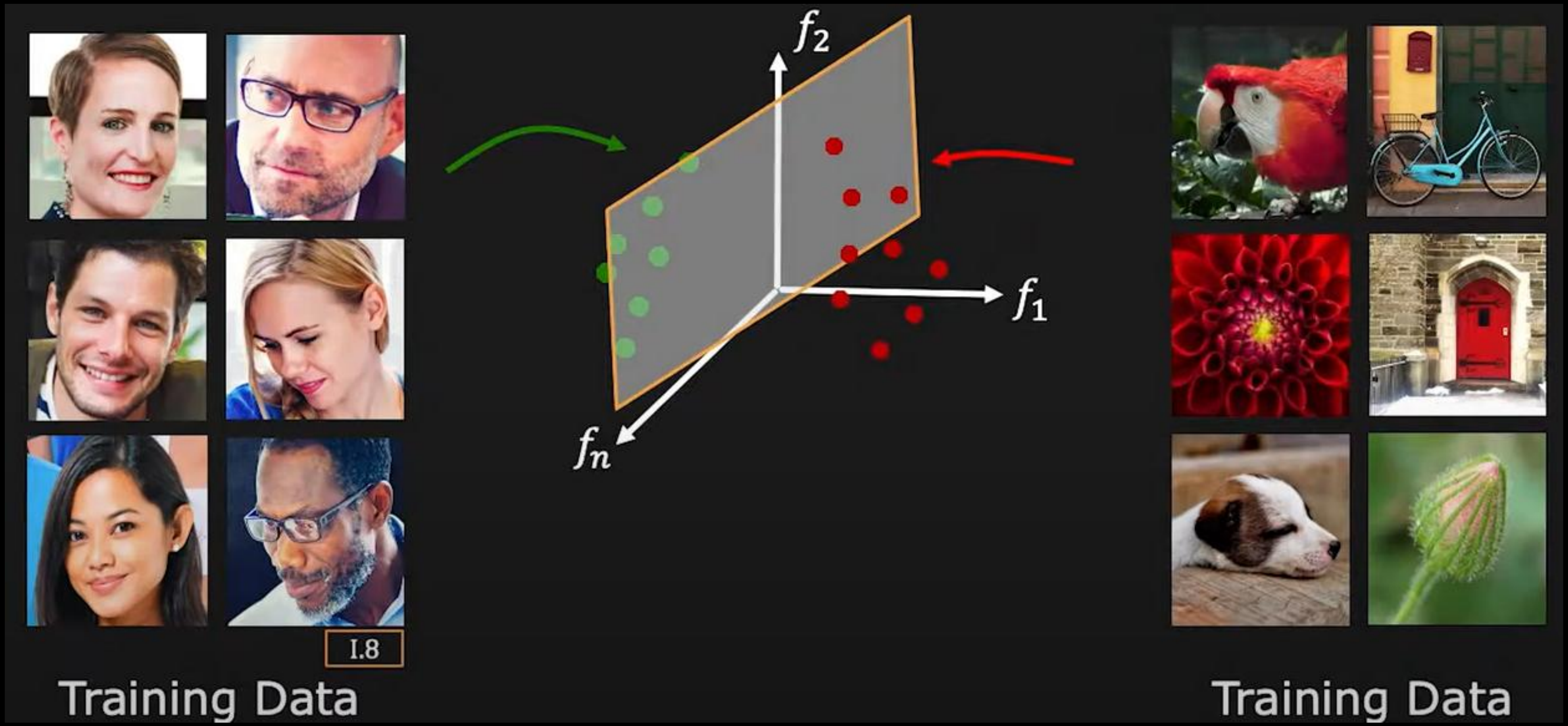


Nearest Neighbour Classifier

- Large training set
- Robust NN classifier
- Slower the NN classifier

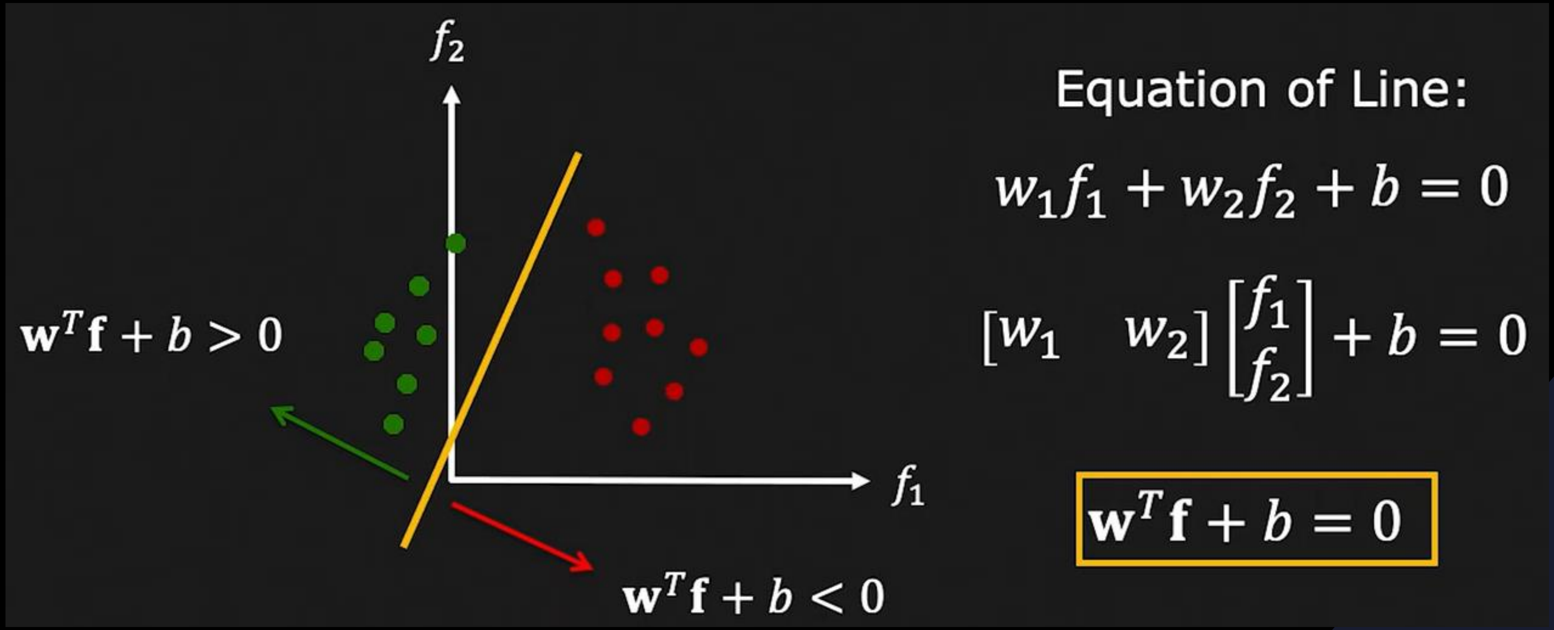


Decision boundary



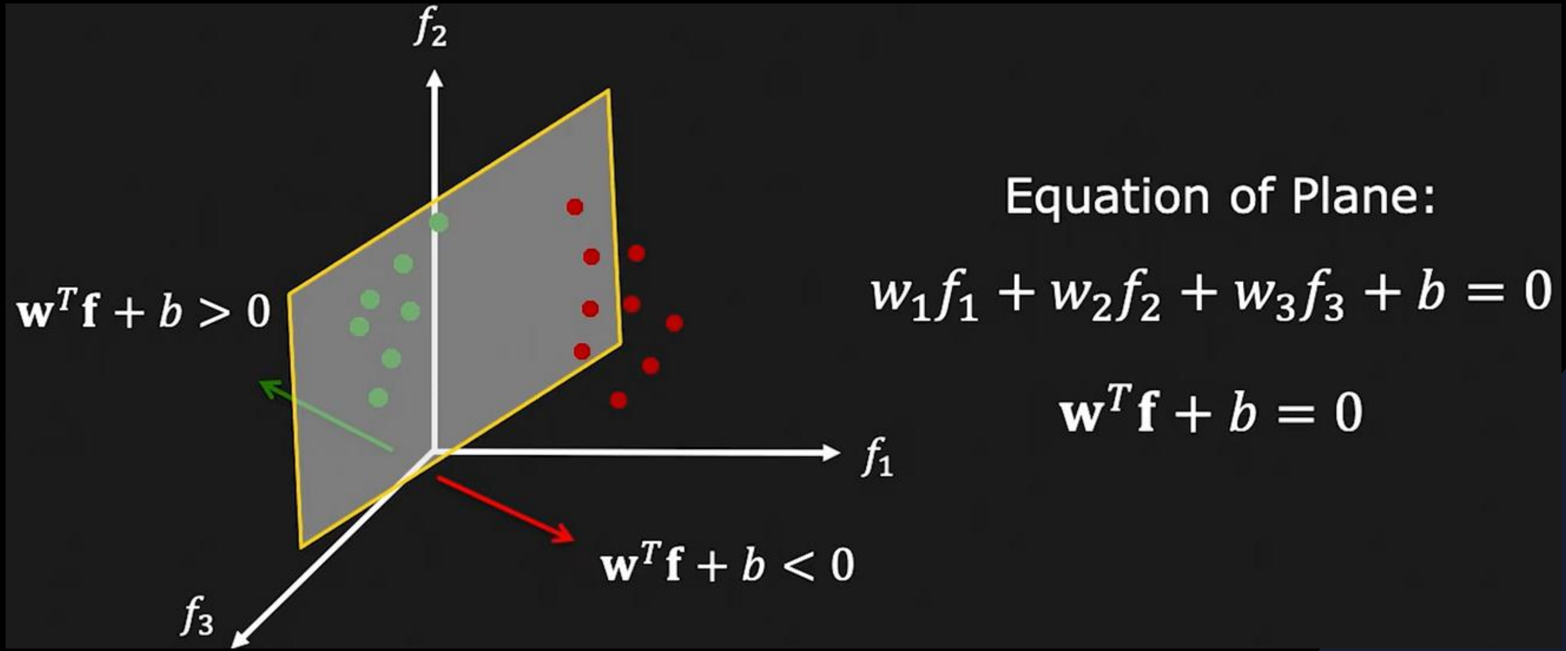
Linear decision boundaries

A Linear decision boundary in 2-D space is a 1-D line



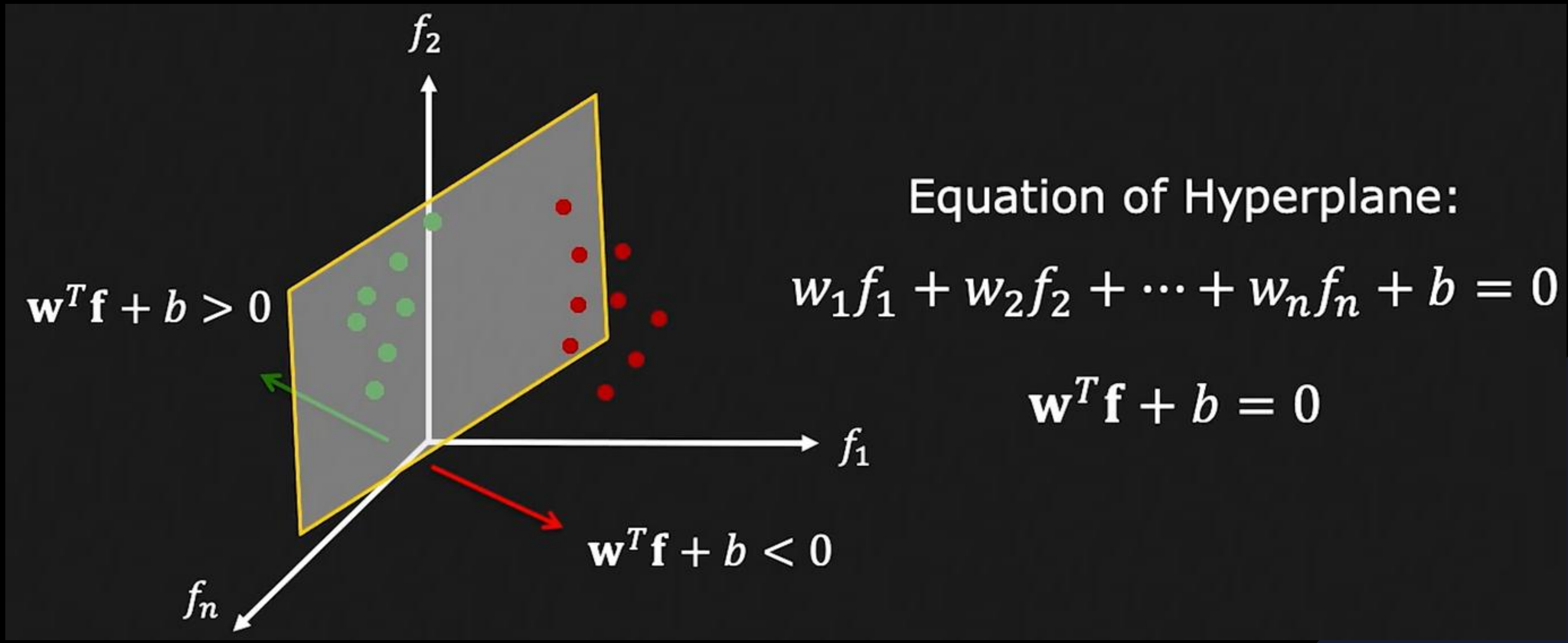
Linear decision boundaries

A Linear decision boundary in 3-D space is a 2-D plane

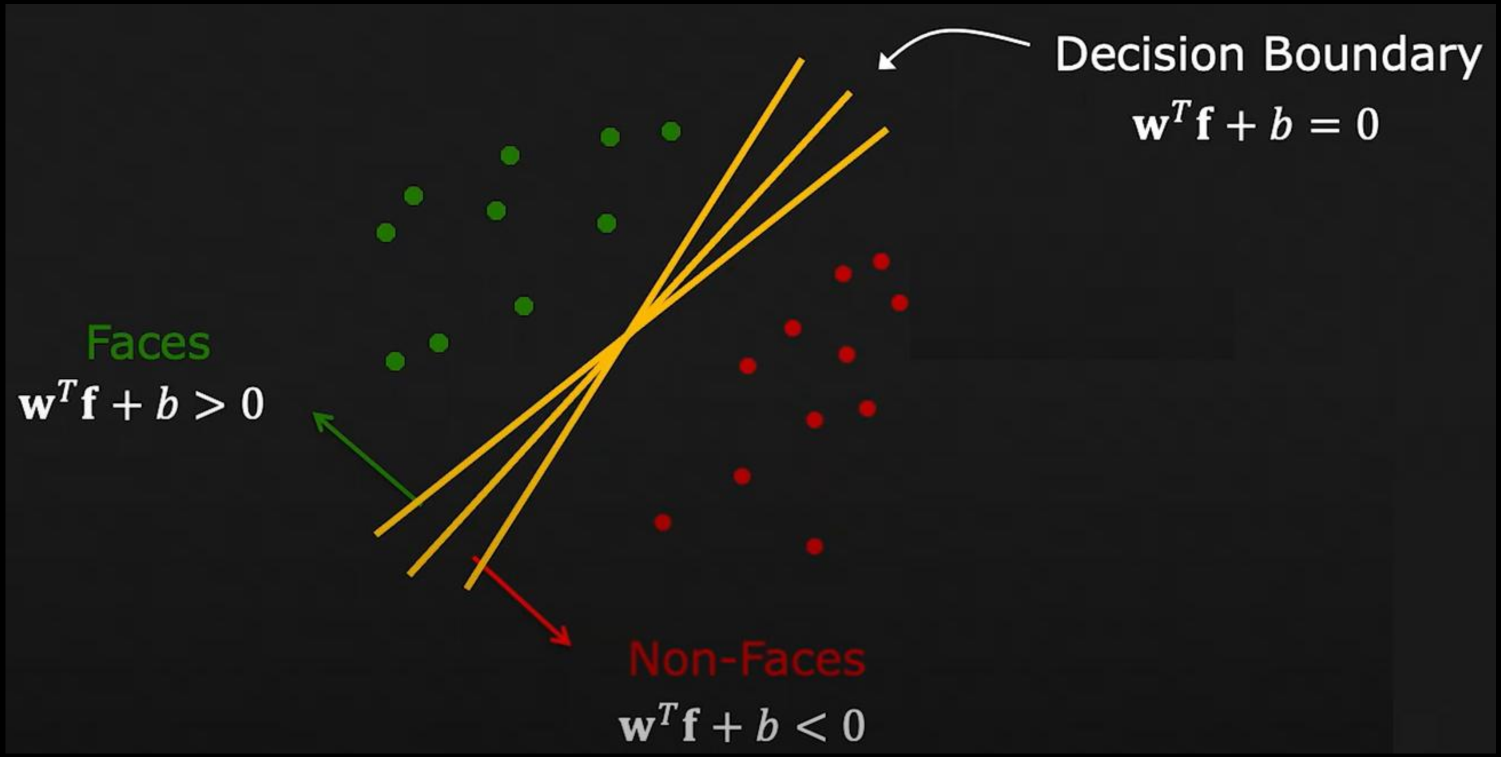


Linear decision boundaries

A Linear decision boundary in n-D space is a (n-1)-D Hyperplane

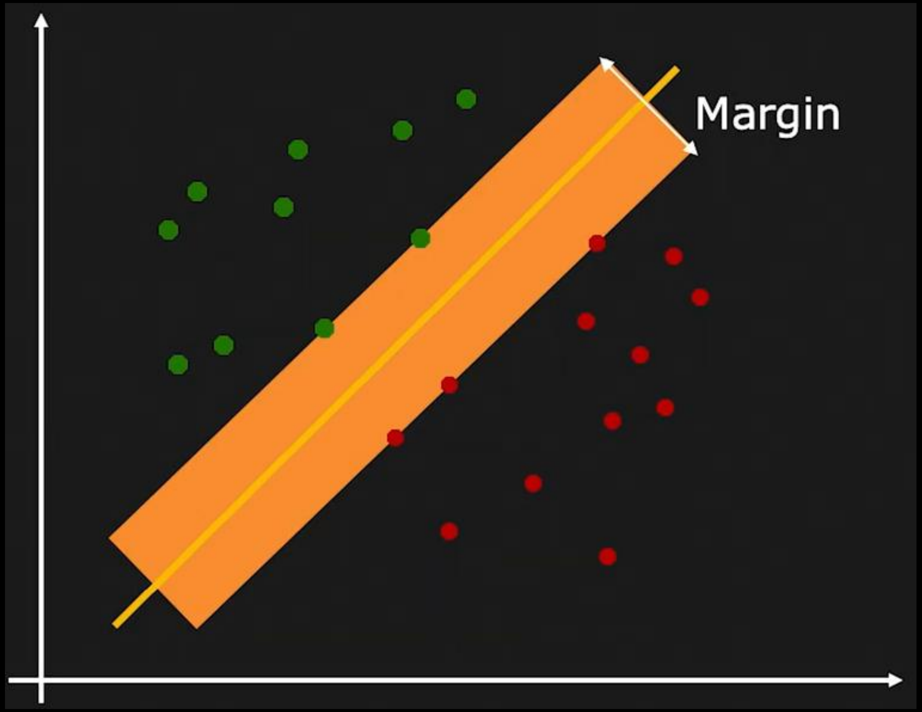


Decision boundary (w,b)

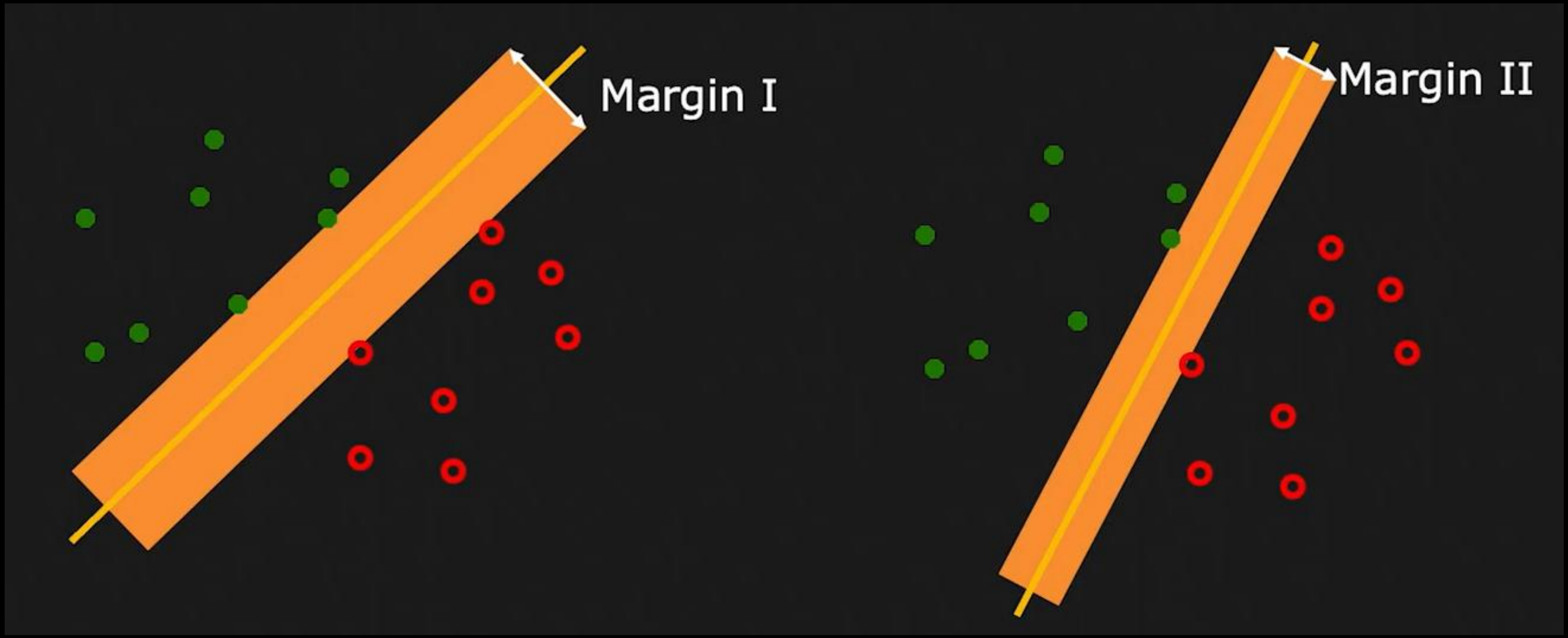


Evaluating a decision boundary

Margin or Safe Zone: The width that the boundary could be increased, before hitting a feature point

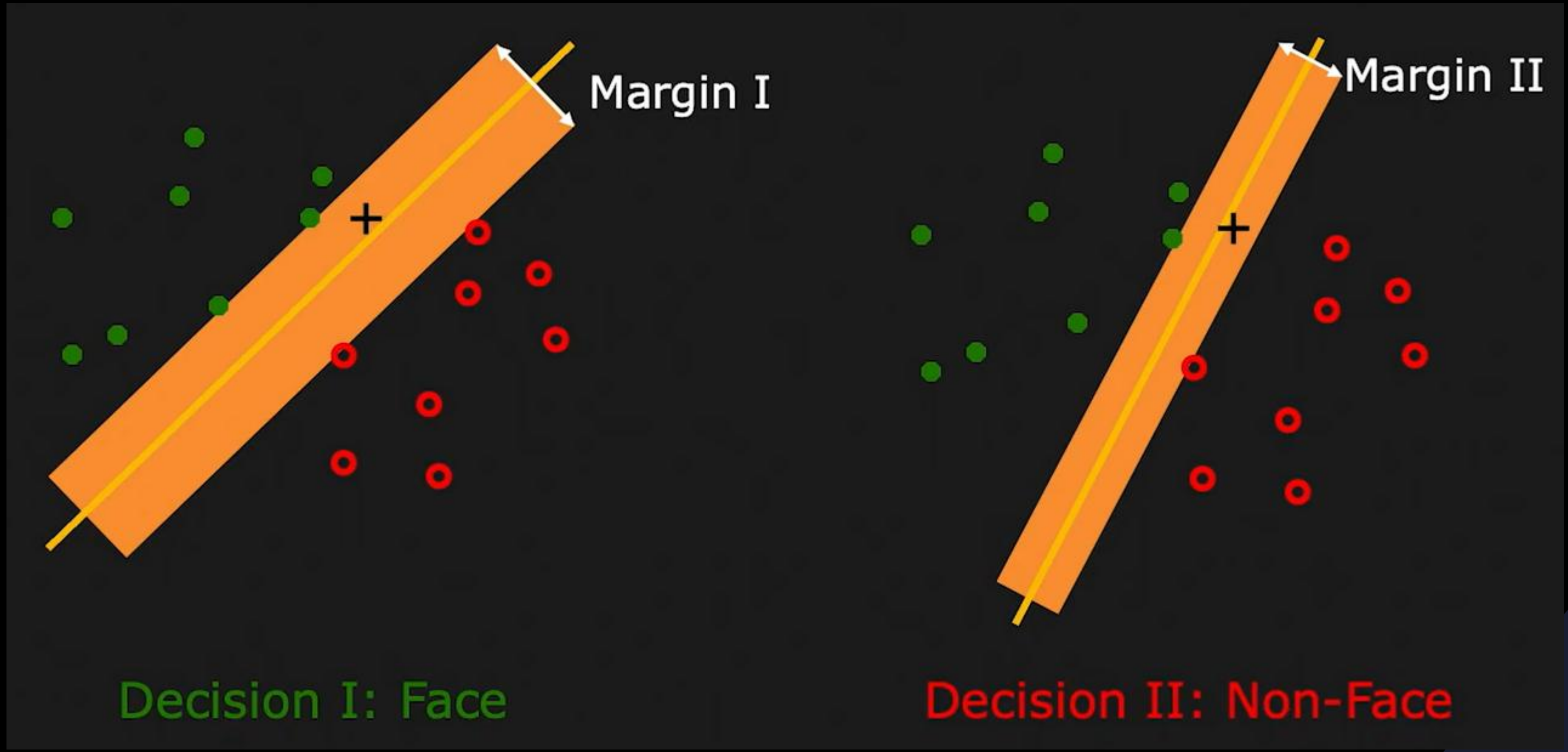


Evaluating a decision boundary



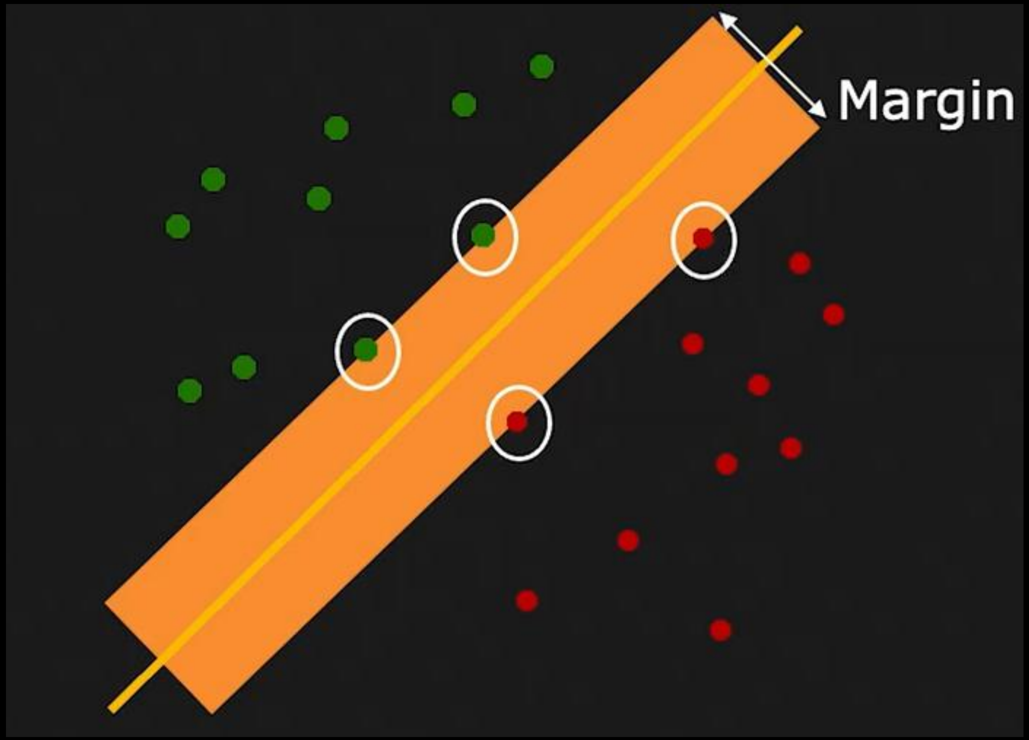
Evaluating a decision boundary

Choose decision boundary with Maximum margin



Support Vector Machine (SVM)

Support Vectors: Closest data samples to the boundary



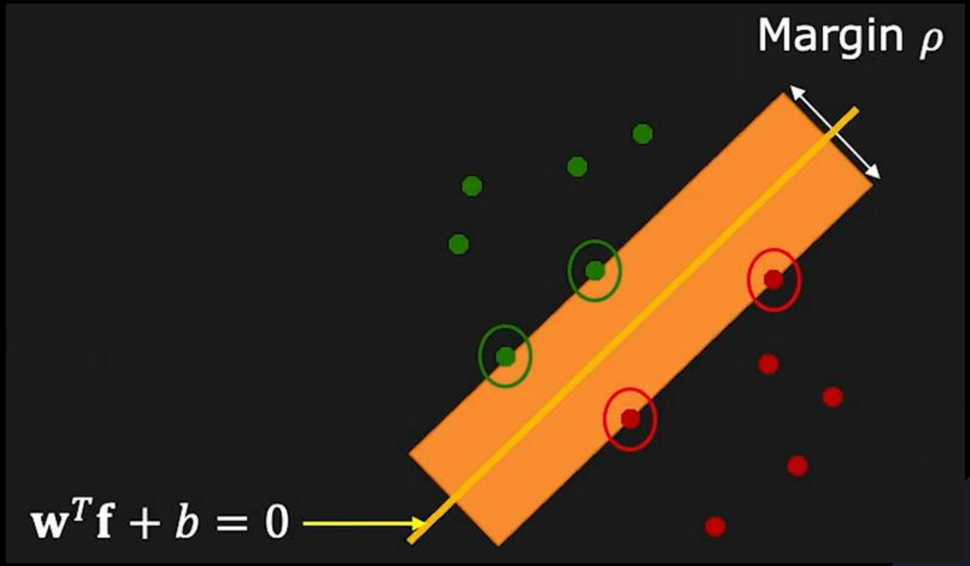
Support Vector Machine (SVM)

Given

- k training images $\{I_1, I_2, \ldots, I_k\}$ and their Haar features $\{f_1, f_2, \ldots, f_k\}$
- K corresponding labels $\{\lambda_1, \lambda_2, \ldots, \lambda_k\}$, where $\lambda_j = +1$ if I_j is a face and $\lambda_j = -1$ if I_j is not a face

Find

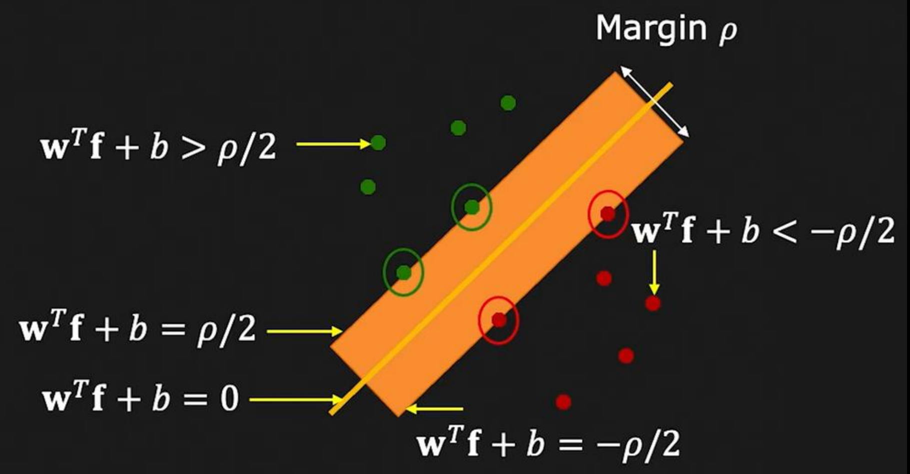
- Decision boundary $w^T f + b = 0$ with maximum margin ρ



Finding decision boundary (w, b)

For each training sample $(\mathbf{f}_i, \lambda_i)$:

$$\left.\begin{aligned} &\text{If } \lambda_i = +1: \quad \mathbf{w}^T\mathbf{f}_i + b \geq \rho/2 \\ &\text{If } \lambda_i = -1: \quad \mathbf{w}^T\mathbf{f}_i + b \leq -\rho/2 \end{aligned}\right\} \quad \boxed{\lambda_i(\mathbf{w}^T\mathbf{f}_i + b) \geq \rho/2}$$



Finding decision boundary (w, b)

For each training sample $(\mathbf{f}_i, \lambda_i)$:

$$\left.\begin{array}{ll} \text{If } \lambda_i = +1: & \mathbf{w}^T\mathbf{f}_i + b \geq \rho/2 \\ \text{If } \lambda_i = -1: & \mathbf{w}^T\mathbf{f}_i + b \leq -\rho/2 \end{array}\right\} \boxed{\lambda_i(\mathbf{w}^T\mathbf{f}_i + b) \geq \rho/2}$$

If $\mathcal{S}$ is the set of support vectors,

Then for every support vector $s \in \mathcal{S}$: $\boxed{\lambda_s(\mathbf{w}^T\mathbf{f}_s + b) = \rho/2}$

> Numerical methods exist to find $\mathbf{w}, b$ and $\mathcal{S}$ that maximize ρ

Classification using SVM

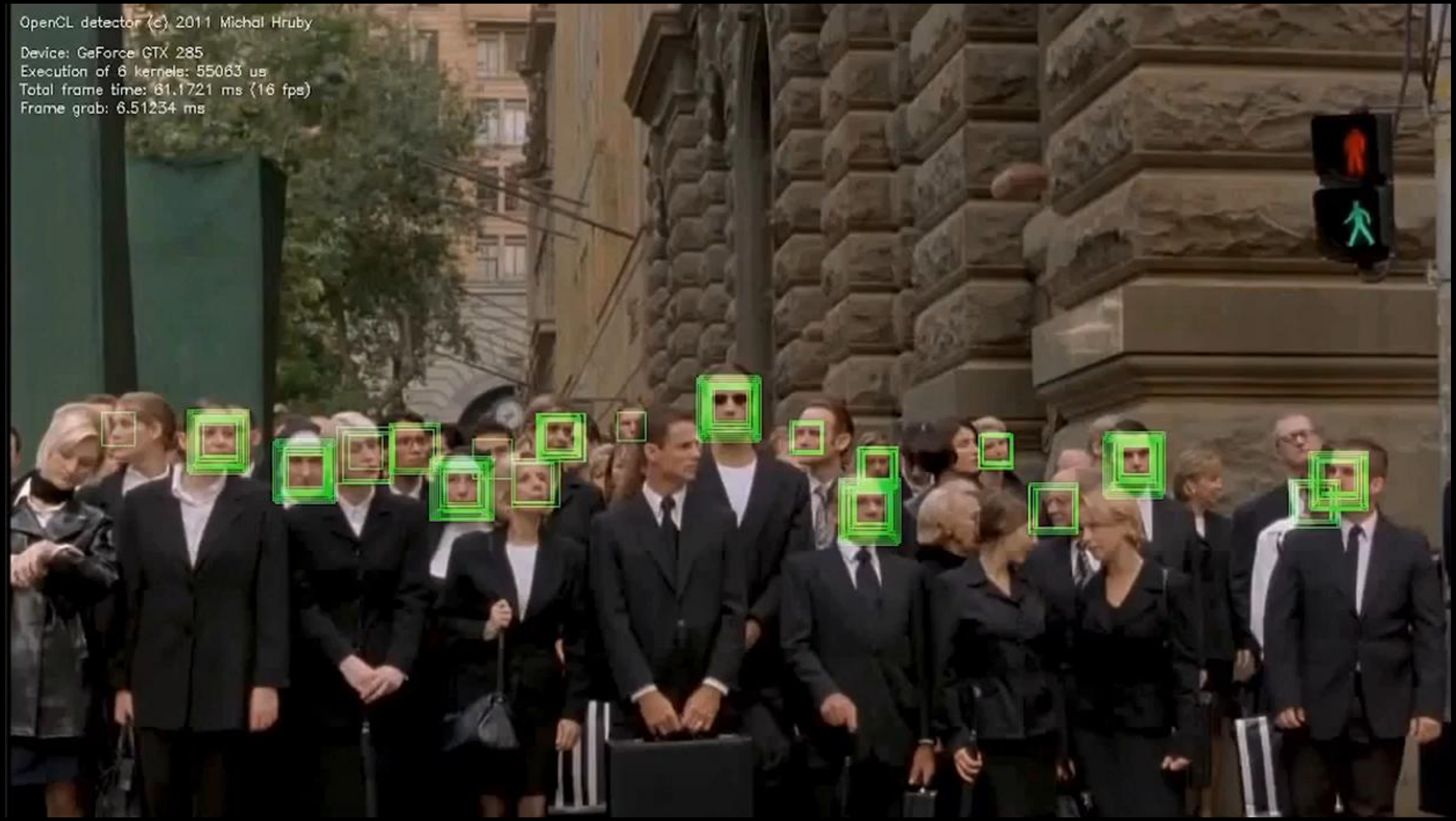
Given: Haar features $\mathbf{f}$ for an image window and SVM parameters $\mathbf{w}, b, \rho, \mathcal{S}$

Classification:

Compute $d = \mathbf{w}^T\mathbf{f} + b$

$$\text{If: } \begin{cases} d \geq \rho/2 & \text{Face} \\ d > 0 \; and \; d < \rho/2 & \text{Probably Face} \\ d < 0 \; and \; d > -\rho/2 & \text{Probably Not-Face} \\ d \leq -\rho/2 & \text{Not-Face} \end{cases}$$

Results


OpenCL detector (c) 2011 Michal Hruby
Device: GeForce GTX 285
Execution of 6 kernels: 55063 us
Total frame time: 61.1721 ms (16 fps)
Frame grab: 6.51234 ms

Programs

- Nearest Neighbour Classifier
- Support Vector Machine