Machine Learning and Pattern Recognition

A High Level Overview

Prof. Anderson Rocha

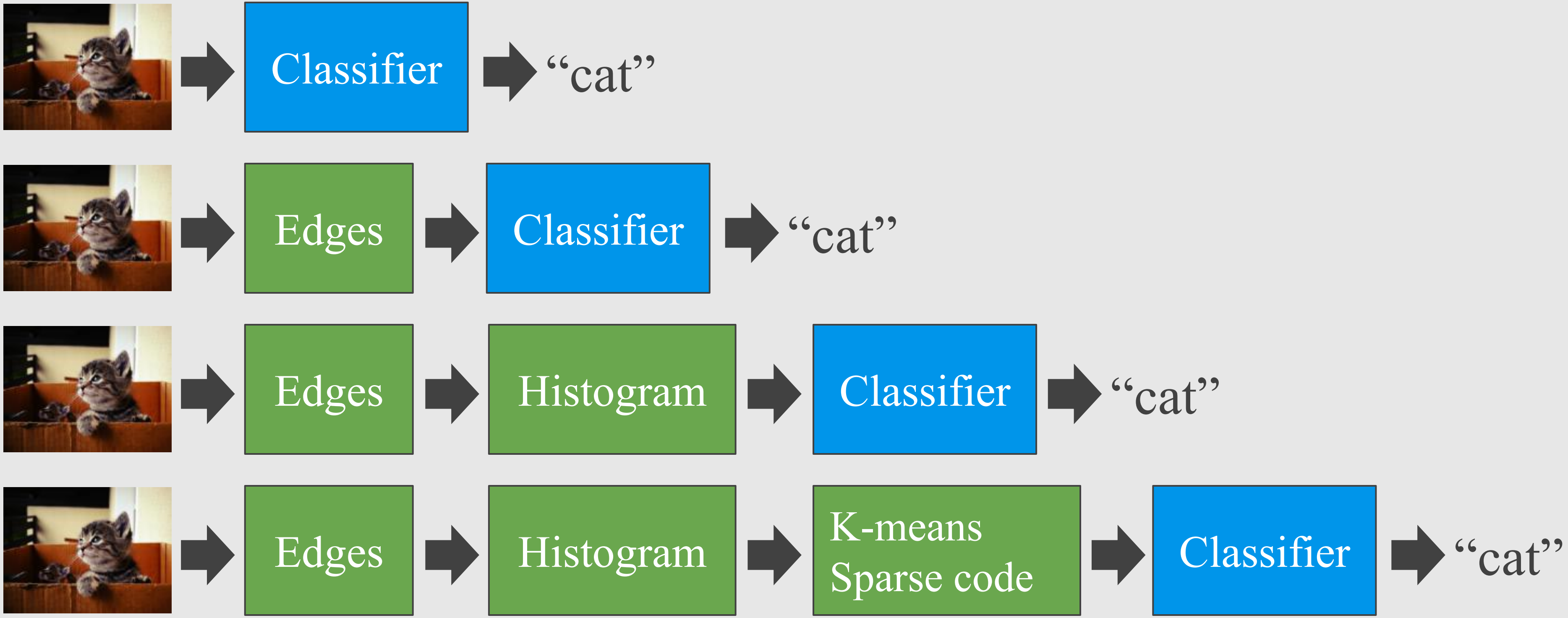
(Main bulk of slides kindly provided by **Prof. Sandra Avila**)

Institute of Computing (IC/Unicamp)

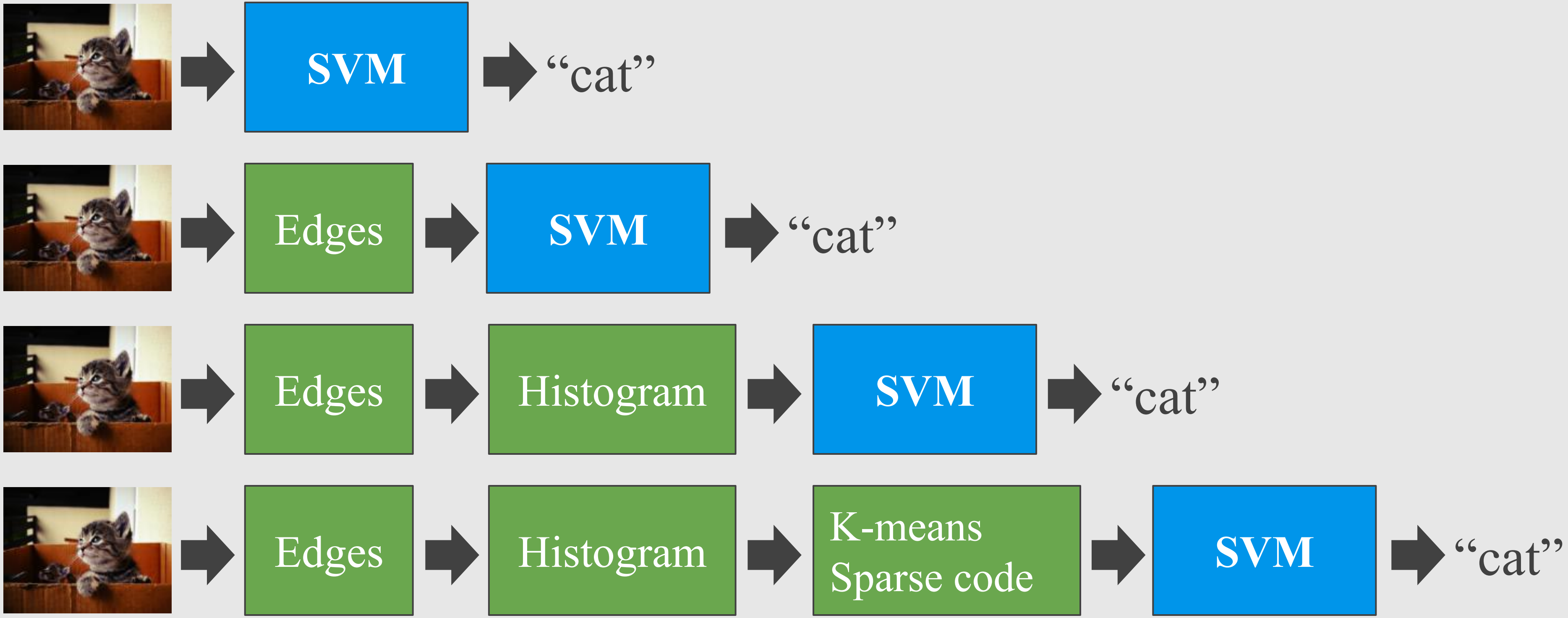
MC886/MO444

SVMs are among the best “off-the-shelf” supervised learning algorithm.

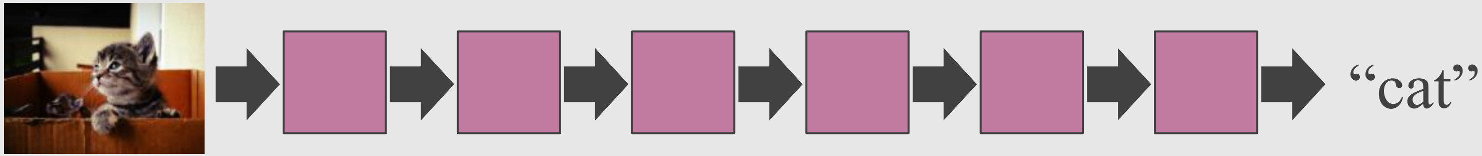
Traditional Recognition



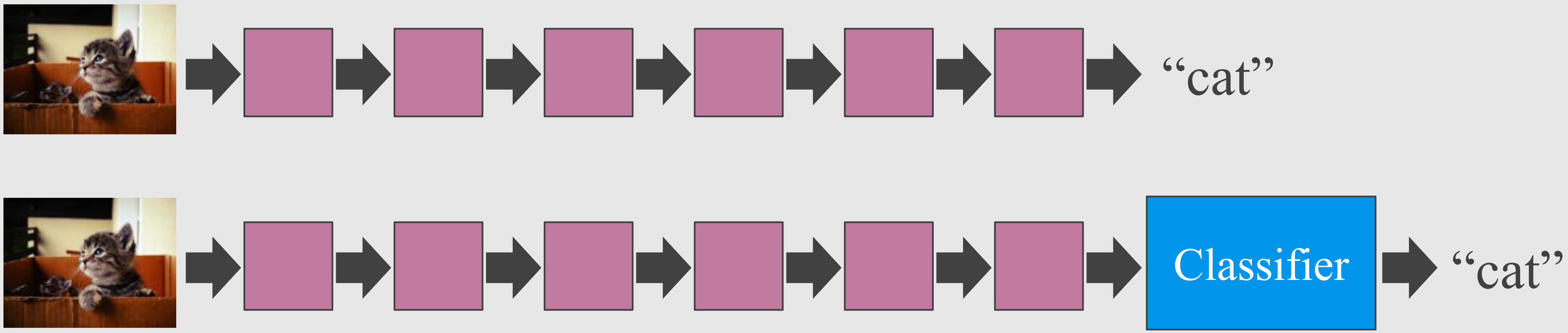
Traditional Recognition



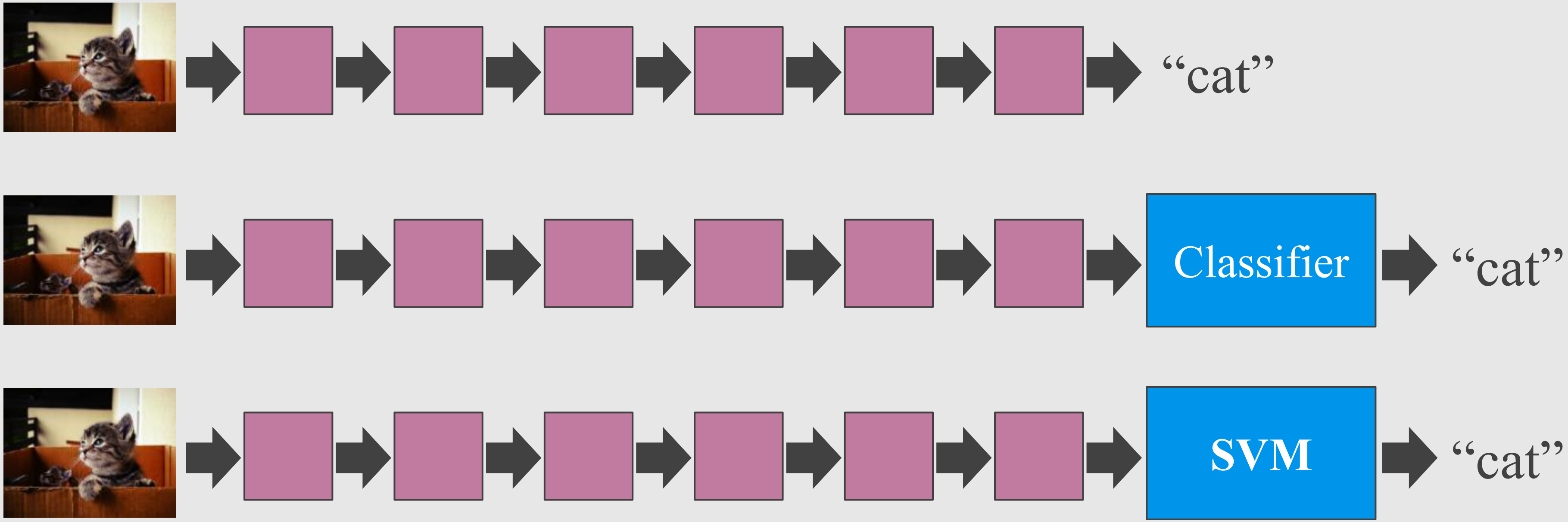
Deep Learning



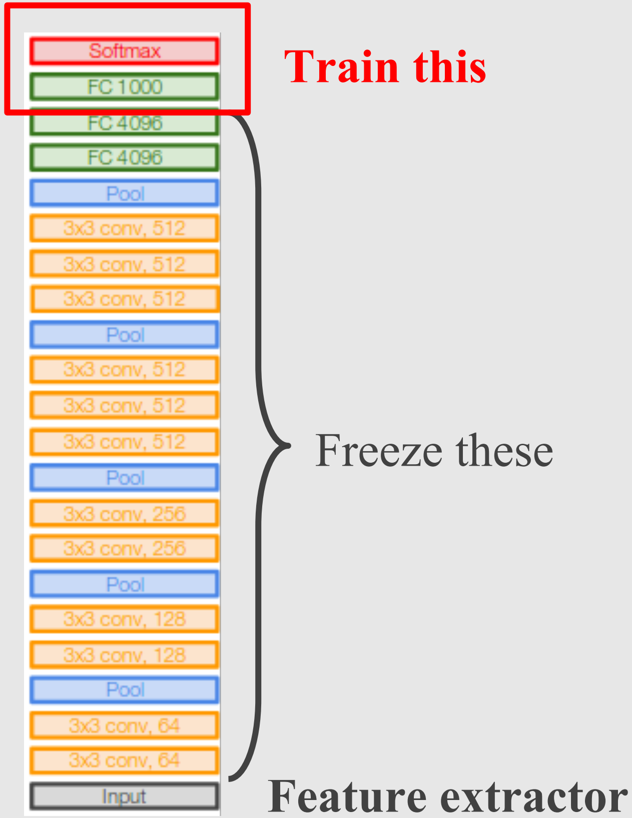
Deep Learning



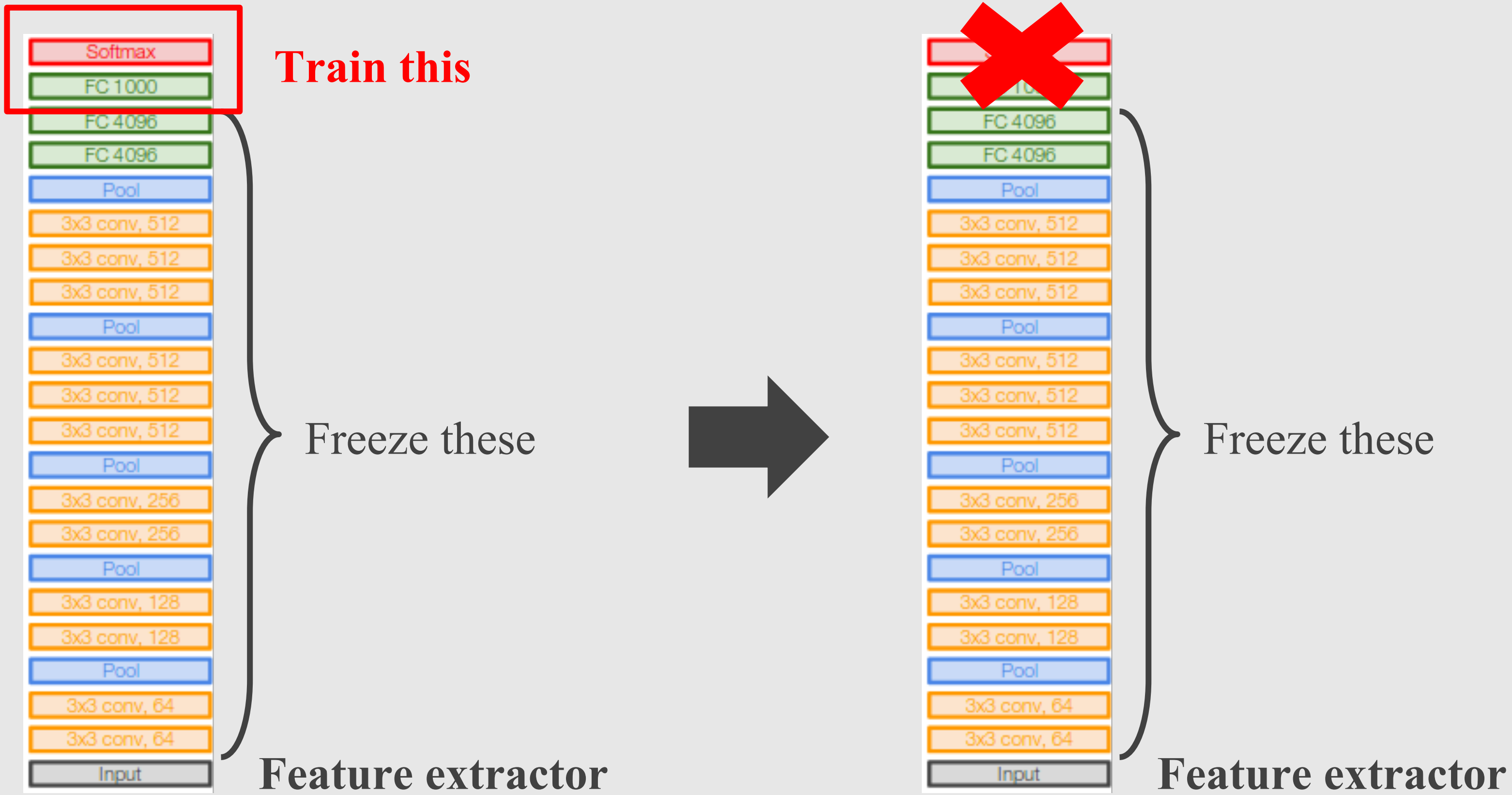
Deep Learning



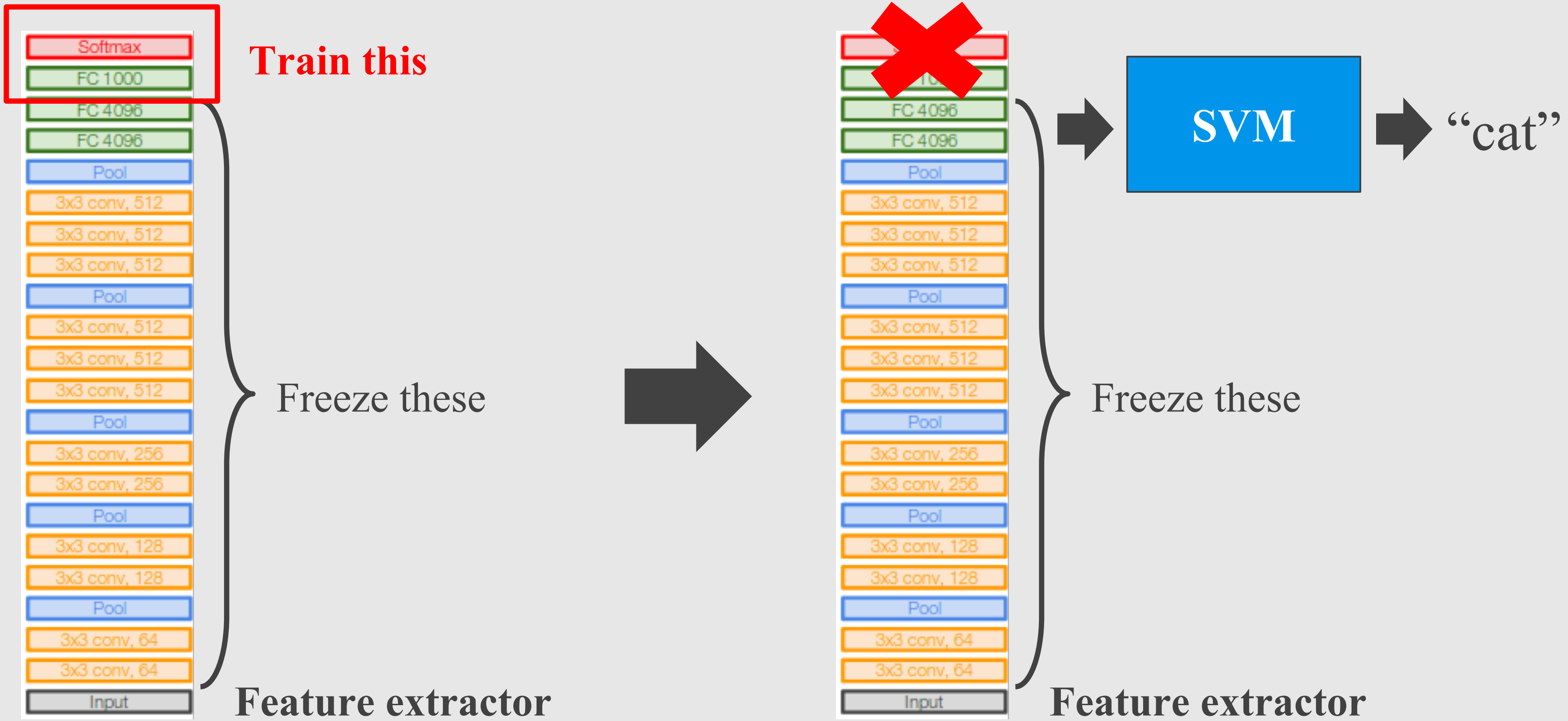
Transfer Learning



Transfer Learning



Transfer Learning

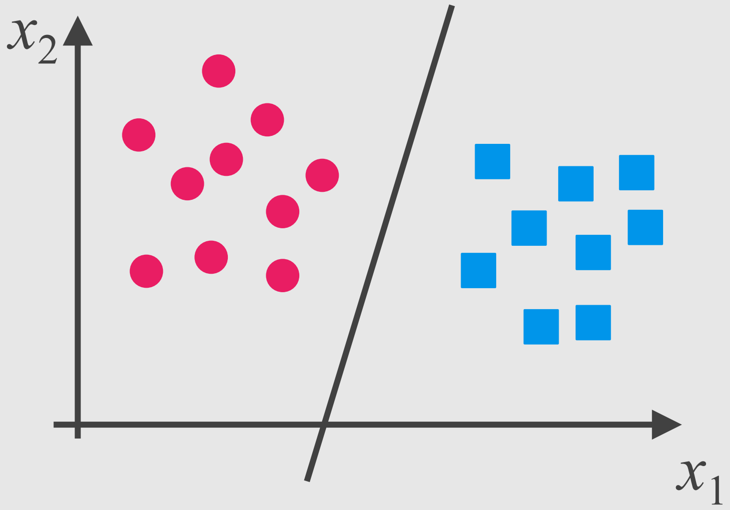


What is Support Vector Machine?

Support Vector Machine

[Vapnik and Chervonenkis, 1964; Vapnik, 1982; Vapnik, 1995]

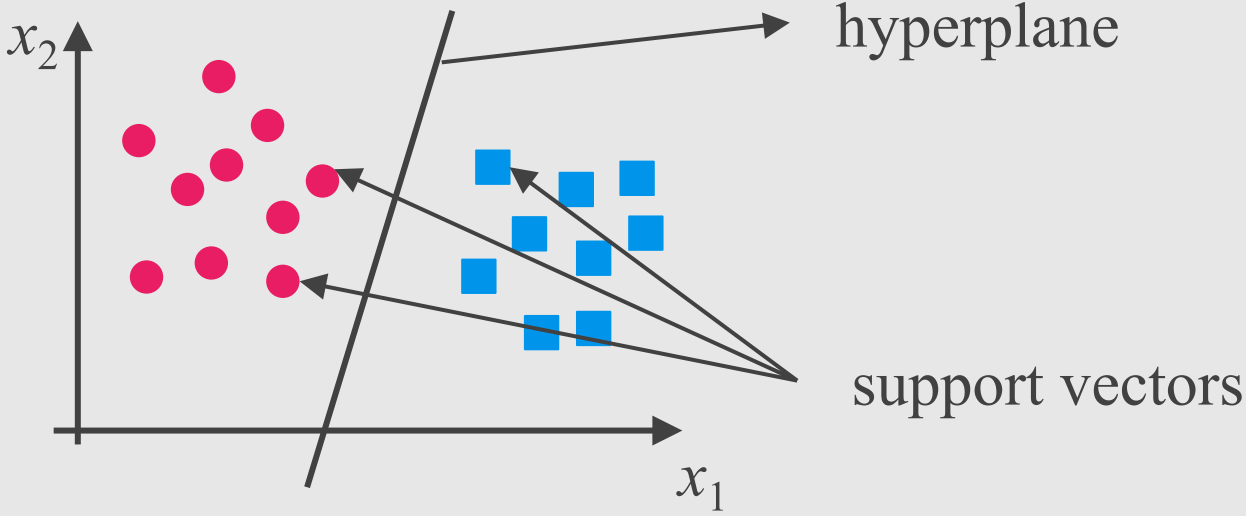
Idea of separating data with **a large "gap."**



Support Vector Machine

[Vapnik and Chervonenkis, 1964; Vapnik, 1982; Vapnik, 1995]

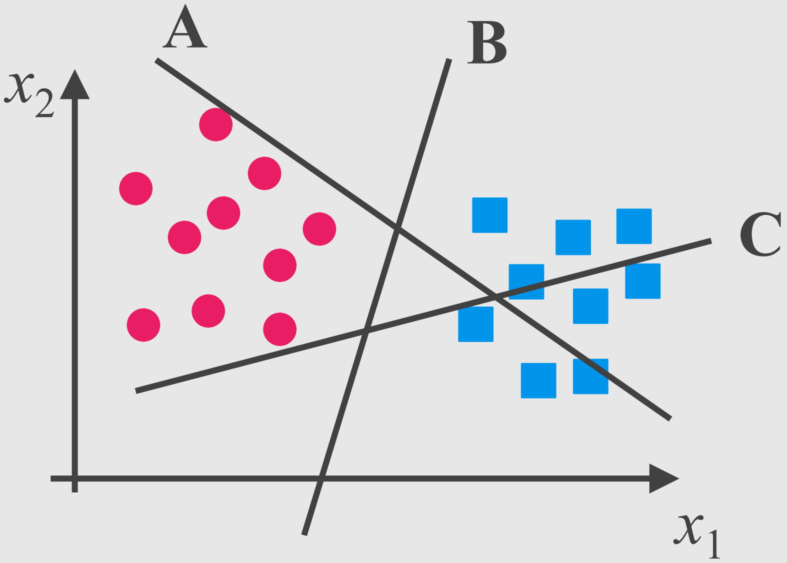
Idea of separating data with **a large "gap."**



How does SVM work?

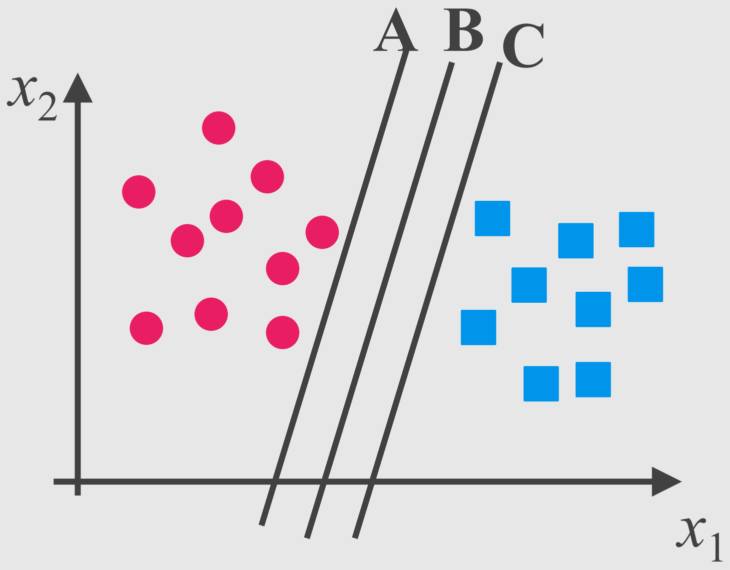
How can we identify the right hyperplane?

Scenario 1



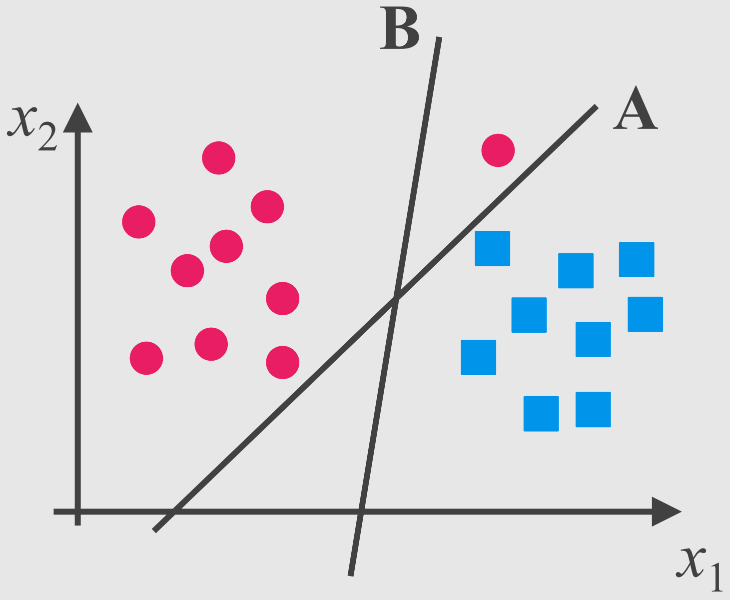
How can we identify the right hyperplane?

Scenario 2



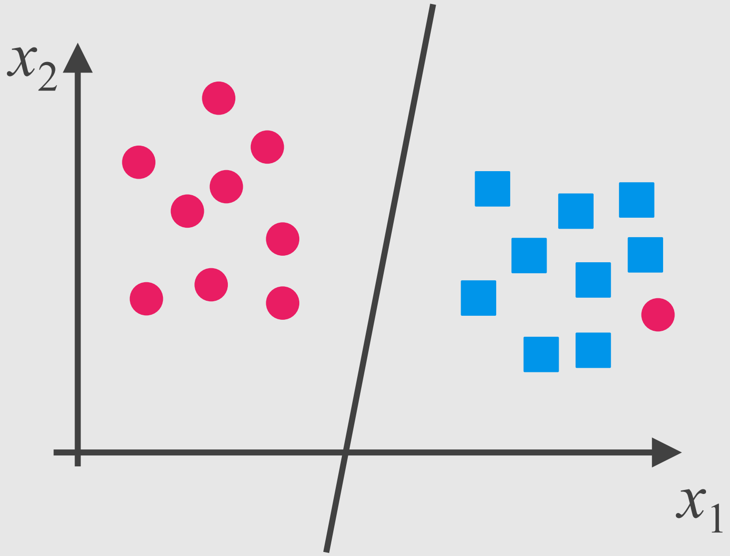
How can we identify the right hyperplane?

Scenario 3



How can we identify the right hyperplane?

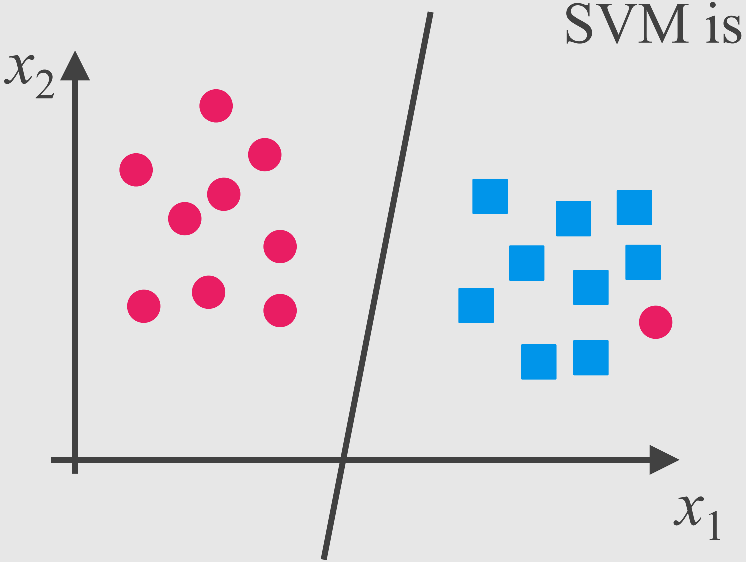
Scenario 4



How can we identify the right hyperplane?

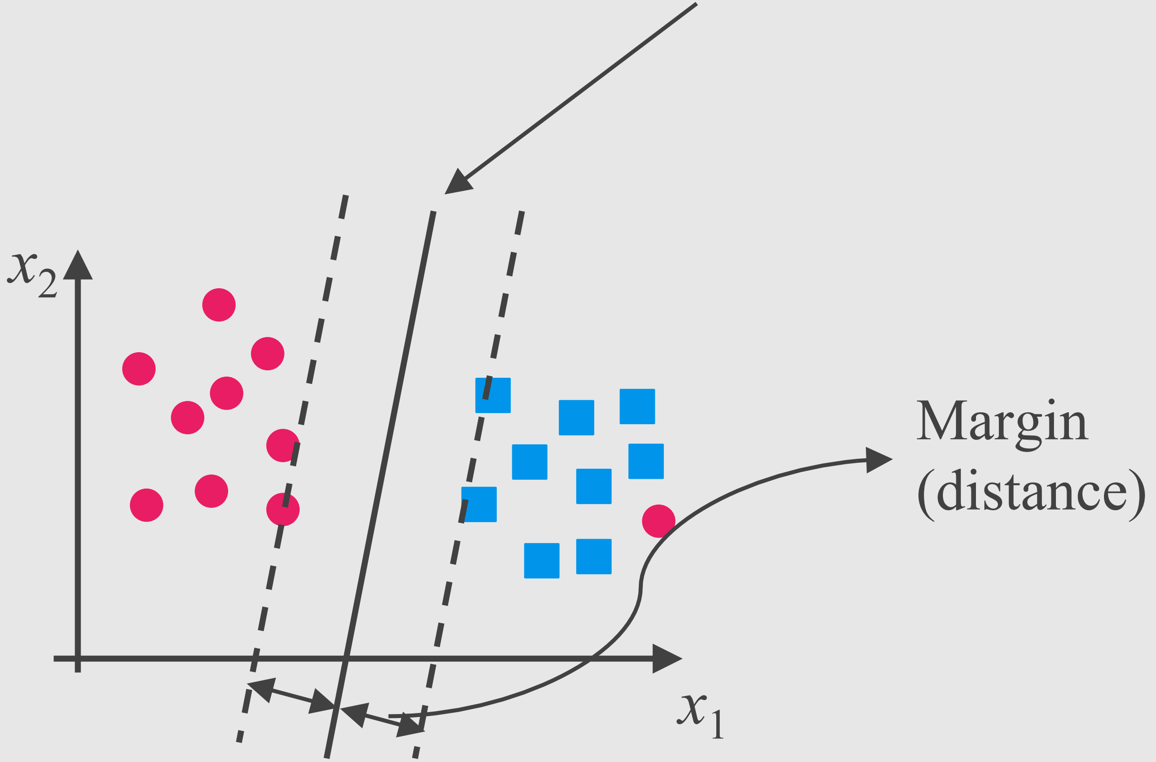
Scenario 4

SVM is robust to outliers



How can we identify the right hyperplane?

Scenario 4



SVM: Notation

We will be considering a **linear classifier for a binary classification** problem with labels y and features x.

SVM: Notation

We will be considering a **linear classifier for a binary classification** problem with labels y and features x.

- Class labels: $y \in \{-1,1\}$ (instead of $\{0,1\}$)
- Parameters: w, b (instead of vector θ)

SVM: Notation

We will be considering a **linear classifier for a binary classification** problem with labels y and features x.

- Class labels: $y \in \{-1,1\}$ (instead of $\{0,1\}$)
- Parameters: w, b (instead of vector θ)
- Classifier: $h_{w,b}(x) = g(w^T x + b)$
  - $g(z) = 1$ if $z \geq 0$, and $g(z) = -1$ otherwise

SVM: The Optimal Hyperplane

Given a training example $(x^{(i)}, y^{(i)})$, we define the margin of (w, b) with respect to the training example:

$$y^{(i)}(w^T x + b) \geq 1, i = \{1, ..., m\}.$$

SVM: The Optimal Hyperplane

Let $P(x^{(1)}, y^{(1)})$ be a point and l be a line defined by $ax + by + c = 0$. The distance d from P to l is defined by:

$$d(a,b,P) = \frac{|ax^{(1)} + by^{(1)} + c|}{\sqrt{a^2 + b^2}}$$

SVM: The Optimal Hyperplane

Let $P(x^{(1)}, y^{(1)})$ be a point and l be a line defined by $ax + by + c = 0$. The distance d from P to l is defined by:

$$d(a,b,P) = \frac{|ax^{(1)} + by^{(1)} + c|}{\sqrt{a^2 + b^2}}$$

$$\Downarrow$$

$$d(w,b,x) = \frac{|w^T x + b|}{||w||}$$

SVM: The Optimal Hyperplane

$$d(w,b,x) = \frac{|w^T x + b|}{||w||}$$

$$\downarrow$$

$$\min_{w,b} \ \tfrac{1}{2}||w||^2$$

$$\text{s.t. } y^{(i)}(w^T x + b) \geq 1,\ i = \{1, ..., m\}$$

SVM: The Optimal Hyperplane

$$d(w,b,x) = \frac{|w^T x + b|}{||w||}$$

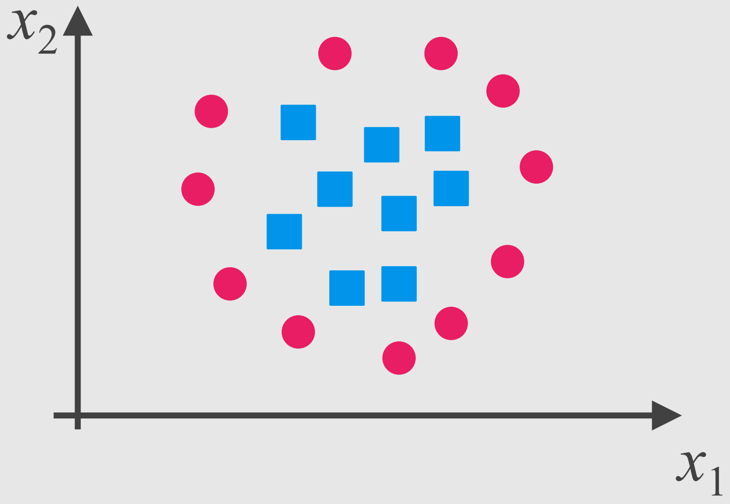
http://cs229.stanford.edu/notes/cs229-notes3.pdf

$$\min_{w,b} \ \tfrac{1}{2}||w||^2$$

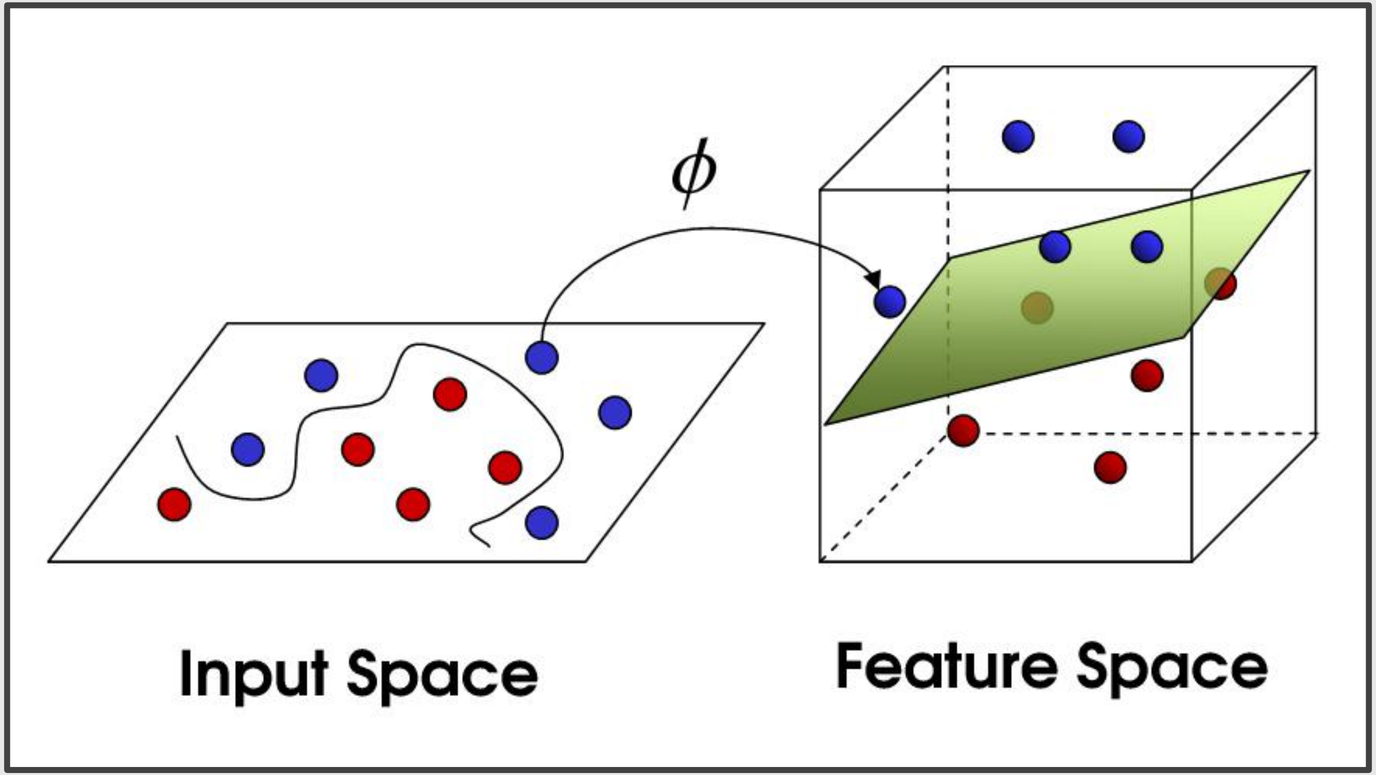
$$\text{s.t. } y^{(i)}(w^T x + b) \geq 1, \ i = \{1, \ldots, m\}$$

How can we identify the right hyperplane?

Scenario 5



Kernel Trick



Kernel Trick

- Linear SVM: $x_i \cdot x_j$

Kernel Trick

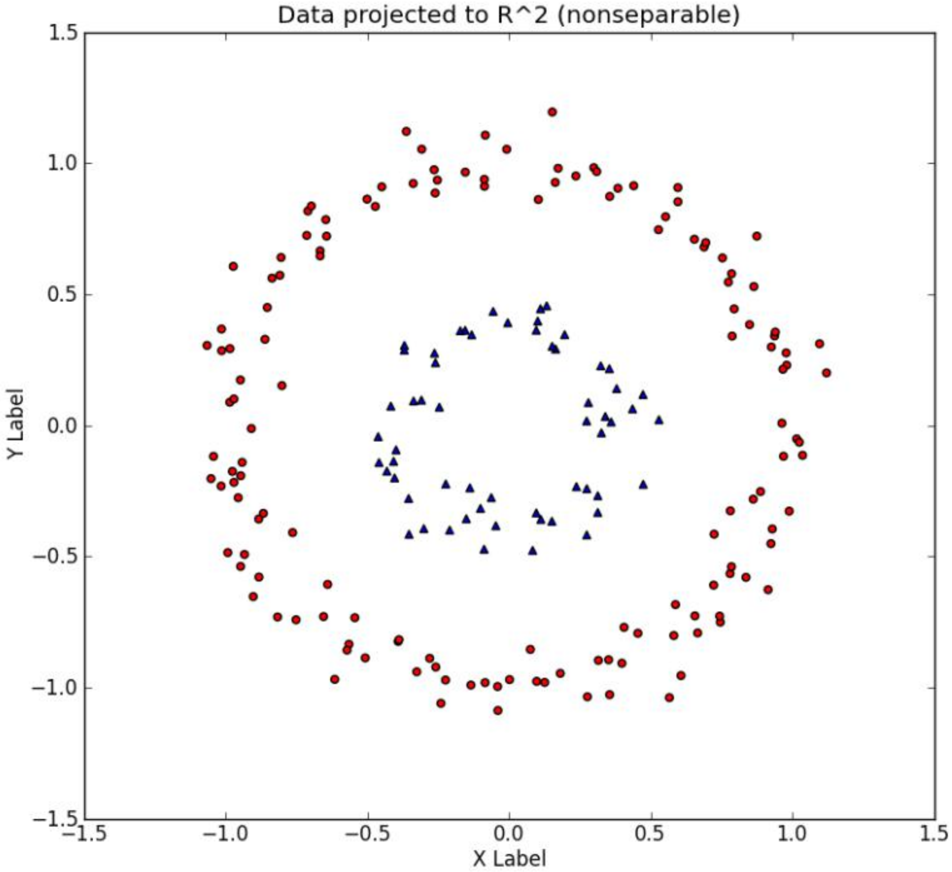
- Linear SVM: $x_i \cdot x_j$
- Nonlinear SVM: $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$, feature mapping ϕ

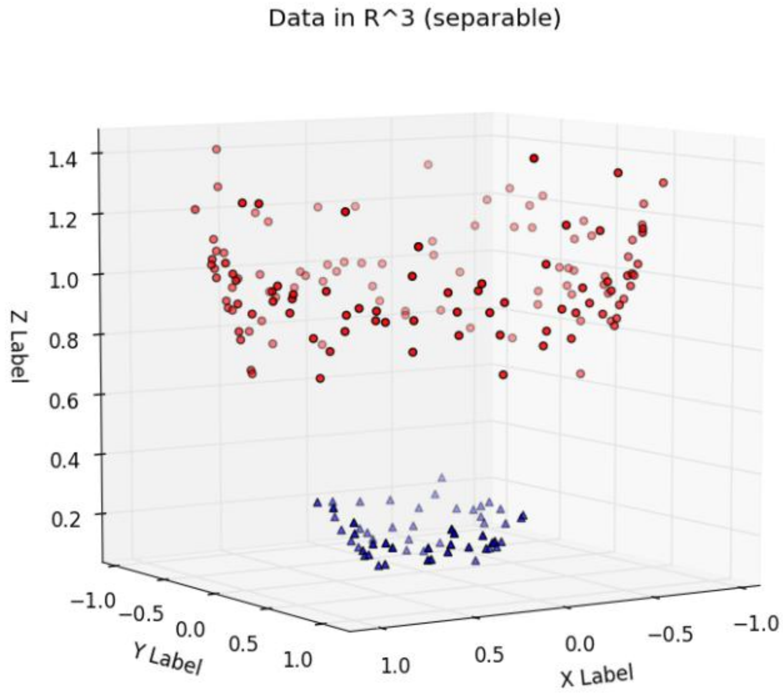
Kernel Trick

- Linear SVM: $x_i \cdot x_j$
- Nonlinear SVM: $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$, feature mapping ϕ
- Kernel matrix $K_{ij} = K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) = \phi(x_j) \cdot \phi(x_i) = K_{ji}$

Kernel Trick

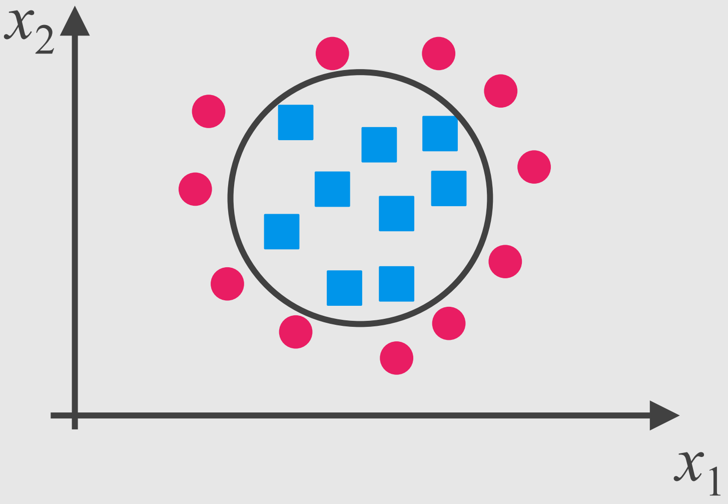
- Linear SVM: $x_i \cdot x_j$
- Nonlinear SVM: $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$, feature mapping ϕ
- Kernel matrix $K_{ij} = K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) = \phi(x_j) \cdot \phi(x_i) = K_{ji}$
- Gaussian kernel: $K(x_i, x_j) = \exp(-\|x_i - x_j\|^2/(2\sigma^2))$
- Polynomial kernel: $K(x_i, x_j) = (x_i \cdot x_j + 1)^{\mathrm{d}}$, d degree
- Chi-square kernel, histogram intersection kernel, string kernel, ...


Data projected to R^2 (nonseparable)
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
X Label
Y Label


Data in R^3 (separable)
1.4
1.2
1.0
0.8
0.6
0.4
0.2
Z Label
−1.0
−0.5
0.0
0.5
1.0
Y Label
1.0
0.5
0.0
−0.5
−1.0
X Label

How can we identify the right hyperplane?

Scenario 5



SVM & scikit-learn

SVM is also available in scikit-learn library and follow the same structure : import library, object creation, fitting model and prediction.

SVM & scikit-learn

```
#Import Library
from sklearn import svm
#Assumed you have, X (predictor) and Y (target) for training data set and x_test(predictor)
of test_dataset
# Create SVM classification object
model = svm.svc(kernel='linear', c=1, gamma=1)
# there is various option associated with it, like changing kernel, gamma and C value. Will
discuss more # about it in next section.Train the model using the training sets and check s
core
model.fit(X, y)
model.score(X, y)
#Predict Output
predicted= model.predict(x_test)
```

SVM & scikit-learn

```
#Import Library
from sklearn import svm
#Assumed you have, X (predictor) and Y (target) for training data set and x_test(predictor)
of test_dataset
# Create SVM classification object
model = svm.svc(kernel='linear', c=1, gamma=1)
# there is various option associated with it, like changing kernel, gamma and C value. Will
discuss more # about it in next section.Train the model using the training sets and check s
core
model.fit(X, y)
model.score(X, y)
#Predict Output
predicted= model.predict(x_test)
```

object creation

SVM & scikit-learn

```
#Import Library
from sklearn import svm
#Assumed you have, X (predictor) and Y (target) for training data set and x_test(predictor)
of test_dataset
# Create SVM classification object
model = svm.svc(kernel='linear', c=1, gamma=1)
# there is various option associated with it, like changing kernel, gamma and C value. Will
discuss more # about it in next section.Train the model using the training sets and check s
core
model.fit(X, y)
model.score(X, y)
#Predict Output
predicted= model.predict(x_test)
```

fitting model

SVM & scikit-learn

```
#Import Library
from sklearn import svm
#Assumed you have, X (predictor) and Y (target) for training data set and x_test(predictor)
of test_dataset
# Create SVM classification object
model = svm.svc(kernel='linear', c=1, gamma=1)
# there is various option associated with it, like changing kernel, gamma and C value. Will
discuss more # about it in next section.Train the model using the training sets and check s
core
model.fit(X, y)
model.score(X, y)
#Predict Output
predicted= model.predict(x_test)
```

prediction

Important Parameters

Important parameters having higher impact on model performance, “kernel”, “gamma” and “C”.

Important Parameters

Important parameters having higher impact on model performance, “kernel”, “gamma” and “C”.

C: Penalty parameter C of the error term. It also controls the trade off between smooth decision boundary and classifying the training points correctly.

Important Parameters

Important parameters having higher impact on model performance, “kernel”, “gamma” and “C”.

C: Penalty parameter C of the error term. It also controls the trade off between smooth decision boundary and classifying the training points correctly.

The parameters can be tuned using grid-search.

References

Machine Learning Books

- Hands-On Machine Learning with Scikit-Learn and TensorFlow, Chap. 5
- Pattern Recognition and Machine Learning, Chap. 6 & 7

Machine Learning Courses

- https://www.coursera.org/learn/machine-learning, Week 7
- http://cs229.stanford.edu/syllabus.html, http://cs229.stanford.edu/notes/cs229-notes3.pdf