

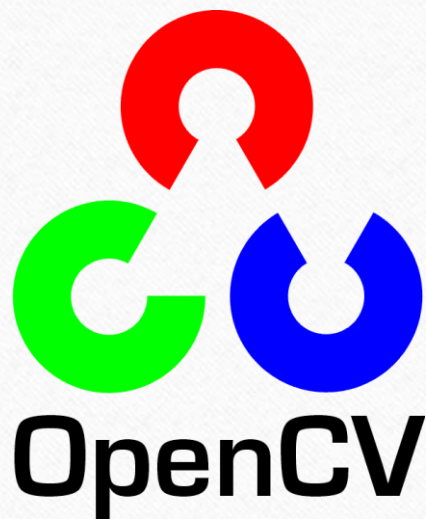
# دوره آموزشی بینایی ماشین کاربردی

آکادمی رباتک - آزمایشگاه تعامل انسان و ربات

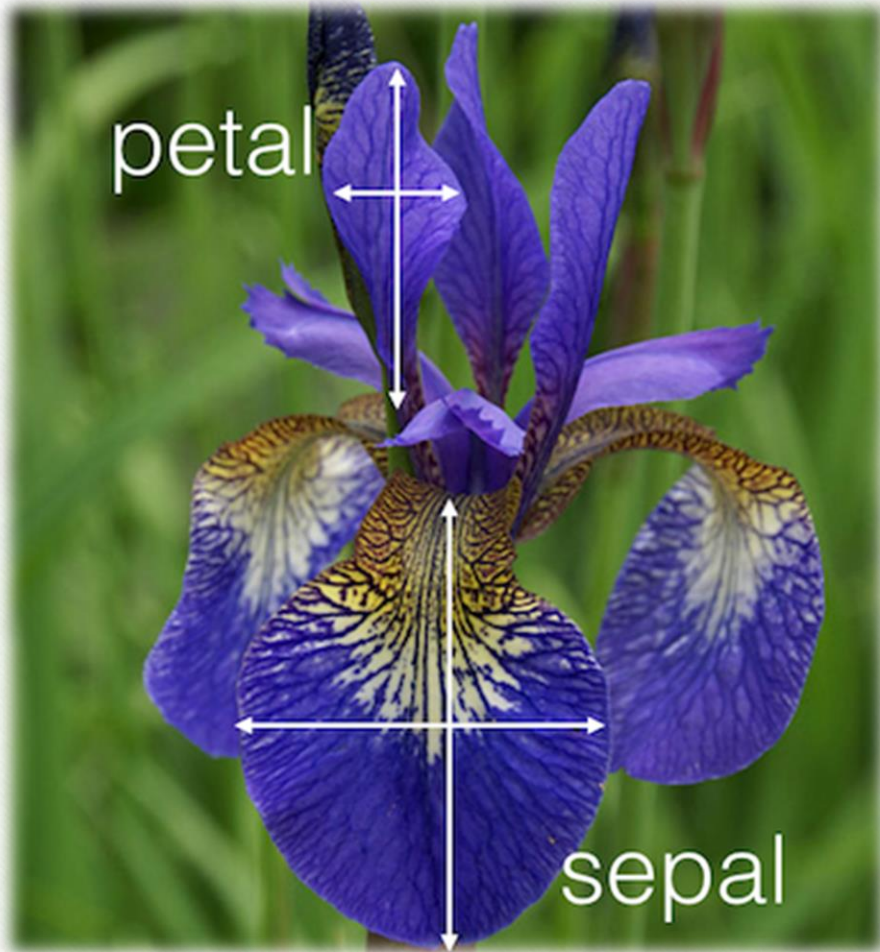
جلسه 6 - ادامه بحث Classification و حل چند مثال



رباتک



## دیتاست گل های IRIS



سه کلاس از هر کلاس 50 عدد

داده ها (ویژگی ها)      کلاس ها :

- |                    |              |
|--------------------|--------------|
| • Iris Setosa      | • طول گلبرگ  |
| • Iris Versicolour | • عرض گلبرگ  |
| • Iris Virginica   | • طول کاسبرگ |
|                    | • عرض کاسبرگ |

نمونه :

5.1,3.5,1.4,0.2,Iris-setosa



## دیتاست MNIST (اعداد دست نویس)



60 هزار داده Train و 10 هزار داده Test

داده ها

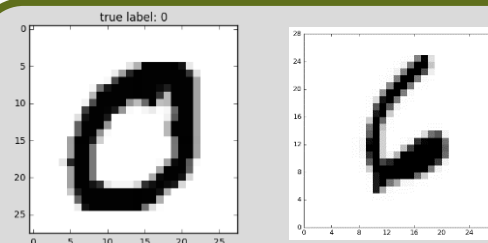
کلاس ها :

تصاویر 28 در 28

اعداد دست نویس 1 تا 10

تک کاناله

نمونه :



## دیتاست هدی (اعداد دست نویس فارسی)

۴	۵	۶	۷	۸	۹
۴	۵	۶	۷	۸	۹
۴	۵	۶	۷	۸	۹
۴	۵	۶	۷	۸	۹
۴	۵	۶	۷	۸	۹

شامل 60K داده Train و 20K داده تست

داده ها

کلاس ها :

تصاویر 28 در 28

ارقام 0 تا 9 فارسی

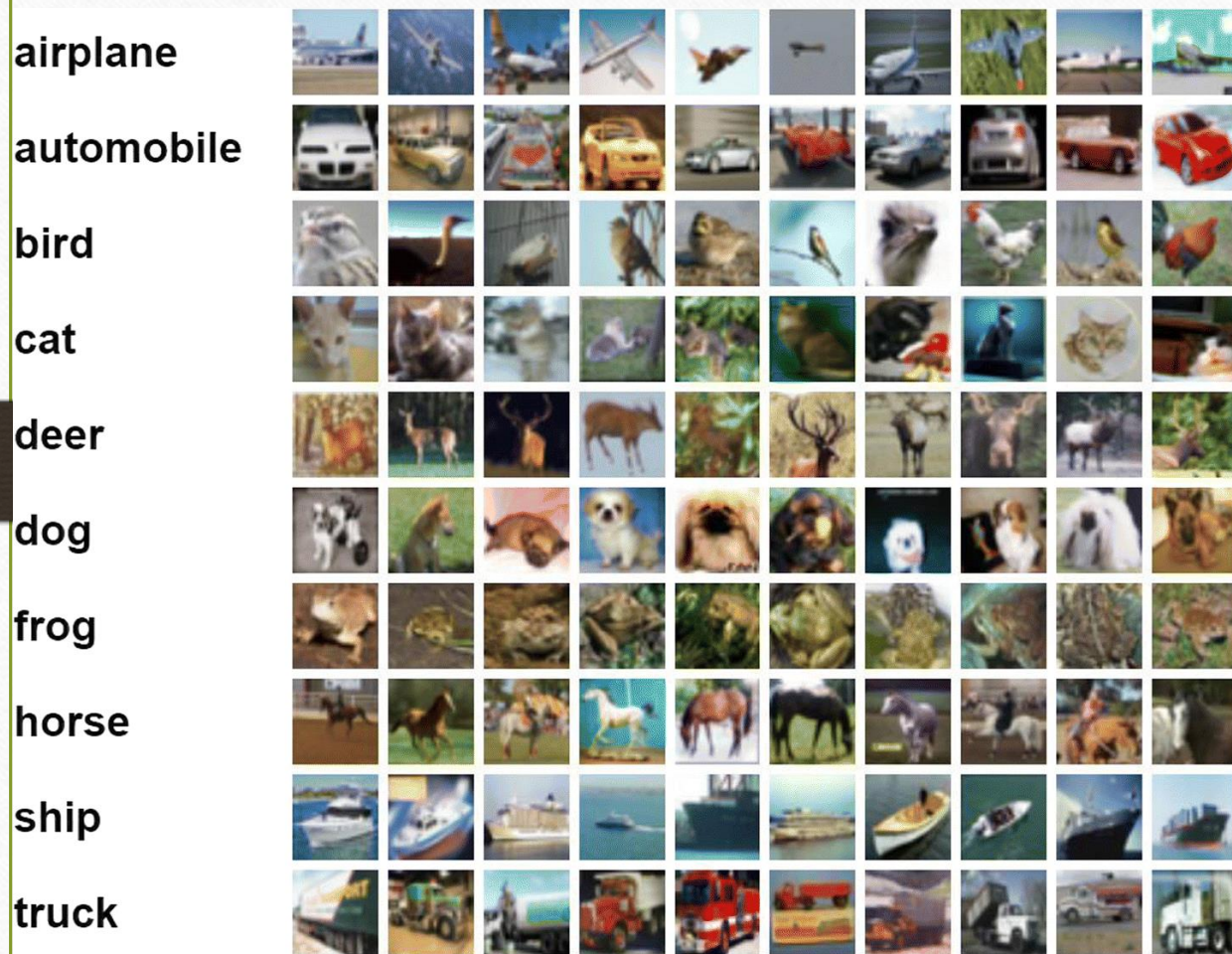
تک کاناله

نمونه :





## دیتاست CIFAR-10



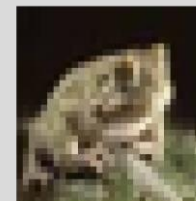
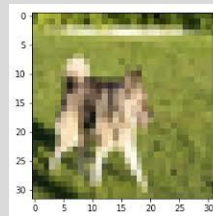
50 هزار داده Train و 10 هزار داده Test

کلاس ها :

داده ها

10 دسته از حیوانات و  
وسایل نقلیه

تصاویر 32 در 32  
سه کاناله



نمونه :



## دیتاست SMILES



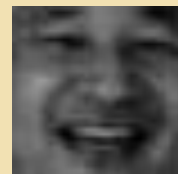
شامل 13165 تصویر

کلاس ها :

داده ها

خندیدن و نخندیدن

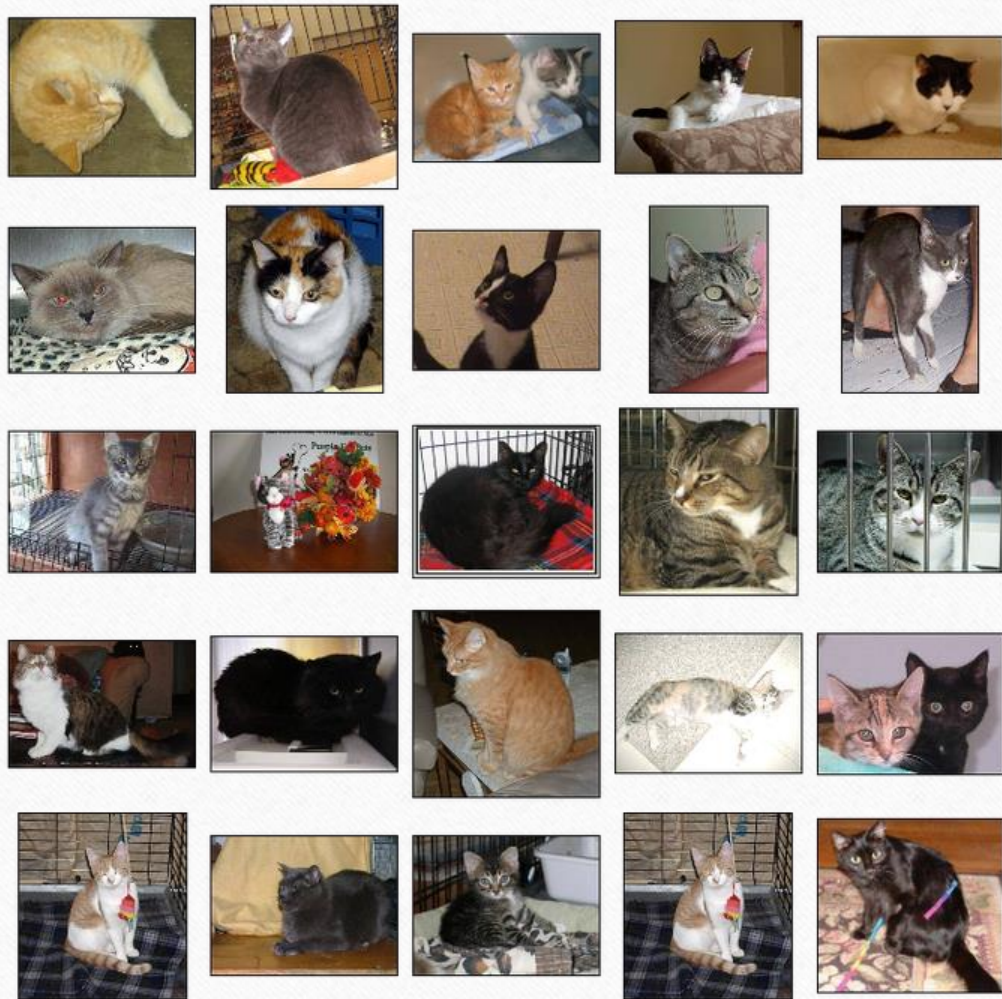
تصاویر 64 در 64  
gray



نمونه :



## دیتاست Kaggle: Dogs vs. Cats



شامل 25000 تصویر

کلاس ها :

گربه و سگ

داده ها

بدون پیش پردازش

نمونه :





## دیتاست 17-Flowers

شامل 1360 تصویر

کلاس ها :

داده ها

17 نوع گل

بدون پیش پردازش

نمونه :





# دیتاست CALTECH-101

baseball-bat



dog



basketball-hoop



Kayac



traffic light



شامل 8677 تصویر

کلاس ها :

داده ها

101 نوع شی

بدون پیش پردازش

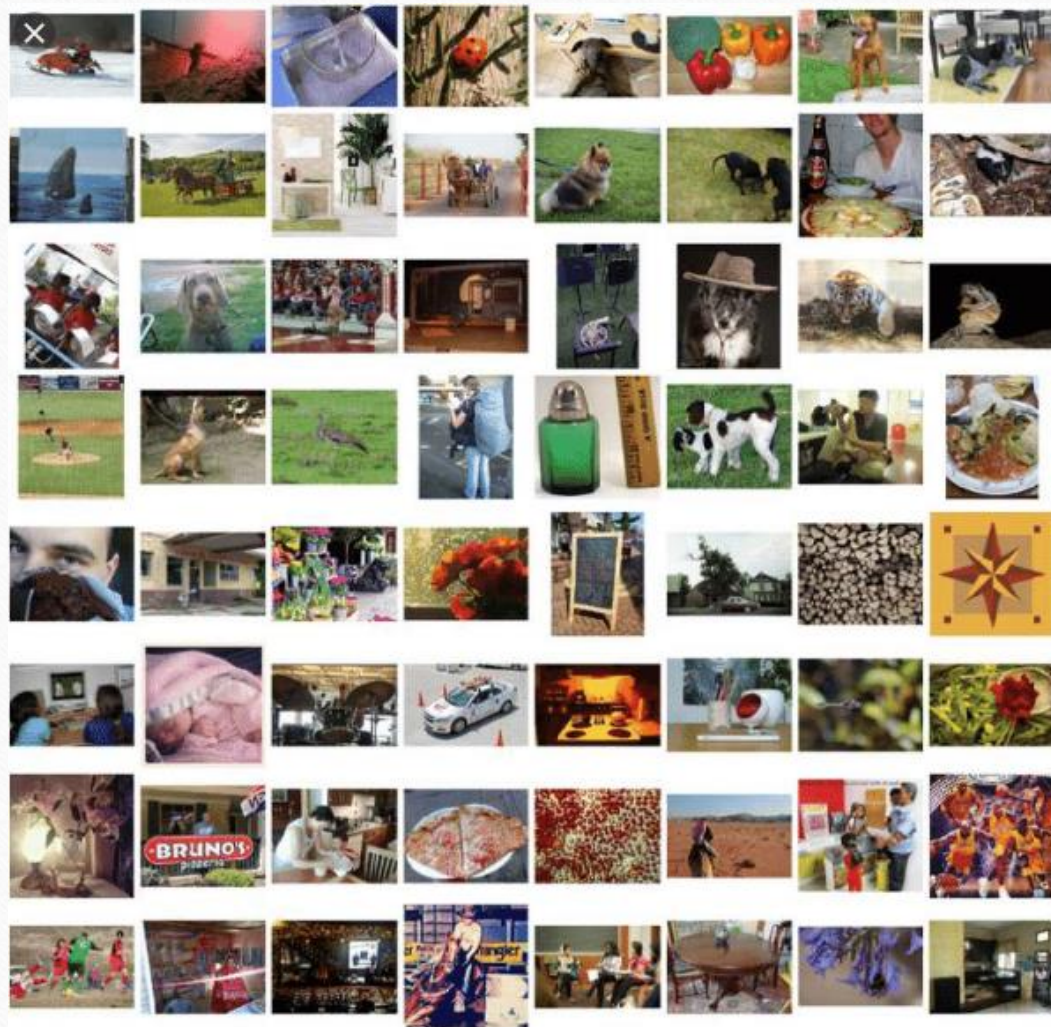
یک دیتاست معروف برای Object Detection

نمونه :





## دیتاست Tiny ImageNet 200



شامل 120000 تصویر

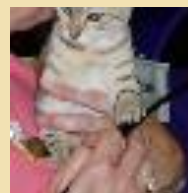
کلاس ها :

داده ها

200 کلاس مختلف

تصاویر 64 در 64 رنگی

نمونه :





## Adience دیتاست



شامل 26580 تصویر

داده ها

کلاس ها :

بدون پیش پردازش

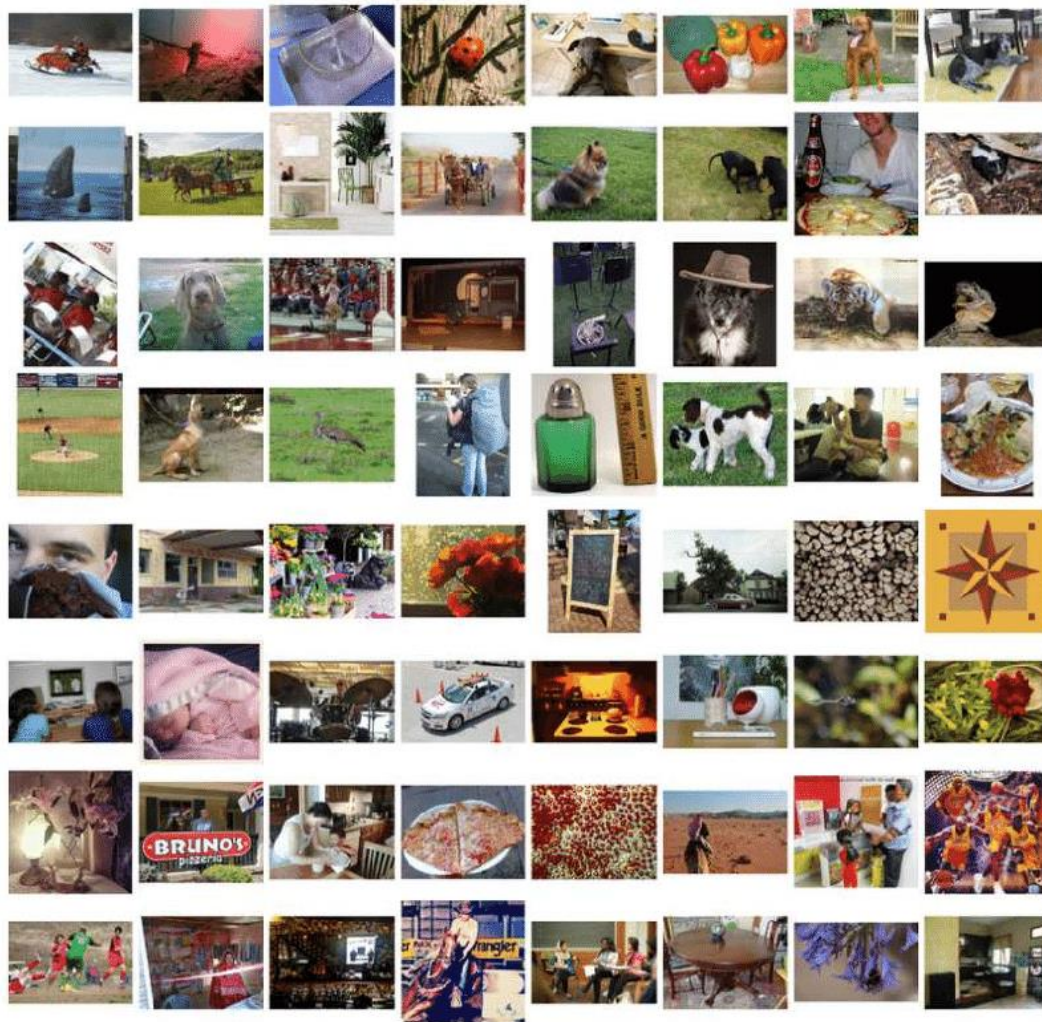
جنسیت و سن (0-60)

نمونه :





## دیتاست ImageNet



بیش از 14 میلیون تصویر

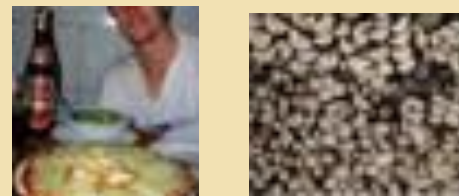
داده ها

کلاس ها :

بدون پیش پردازش

22000 جسم مختلف

نمونه :





## دیتاست Kaggle: Facial Expression Recognition Challenge



شامل 35888 تصویر

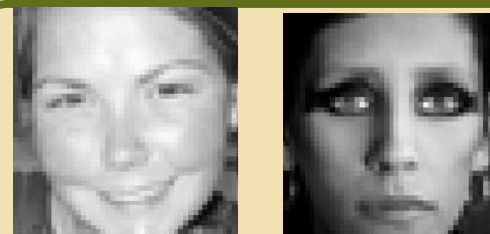
کلاس ها :

داده ها

7 حالت چهره

48 در 48 تک کاناله

نمونه :



## دیتاست Indoor CVPR



Hospital



KinderGarden



ClassRoom



RestaurantKitchen



ArtStudio



Office



MusicStudio



Labratory



OpratingRoom



WareHouse



TVStudio



MeetingRoom

شامل 15620 تصویر

کلاس ها :

داده ها

67 محل مختلف

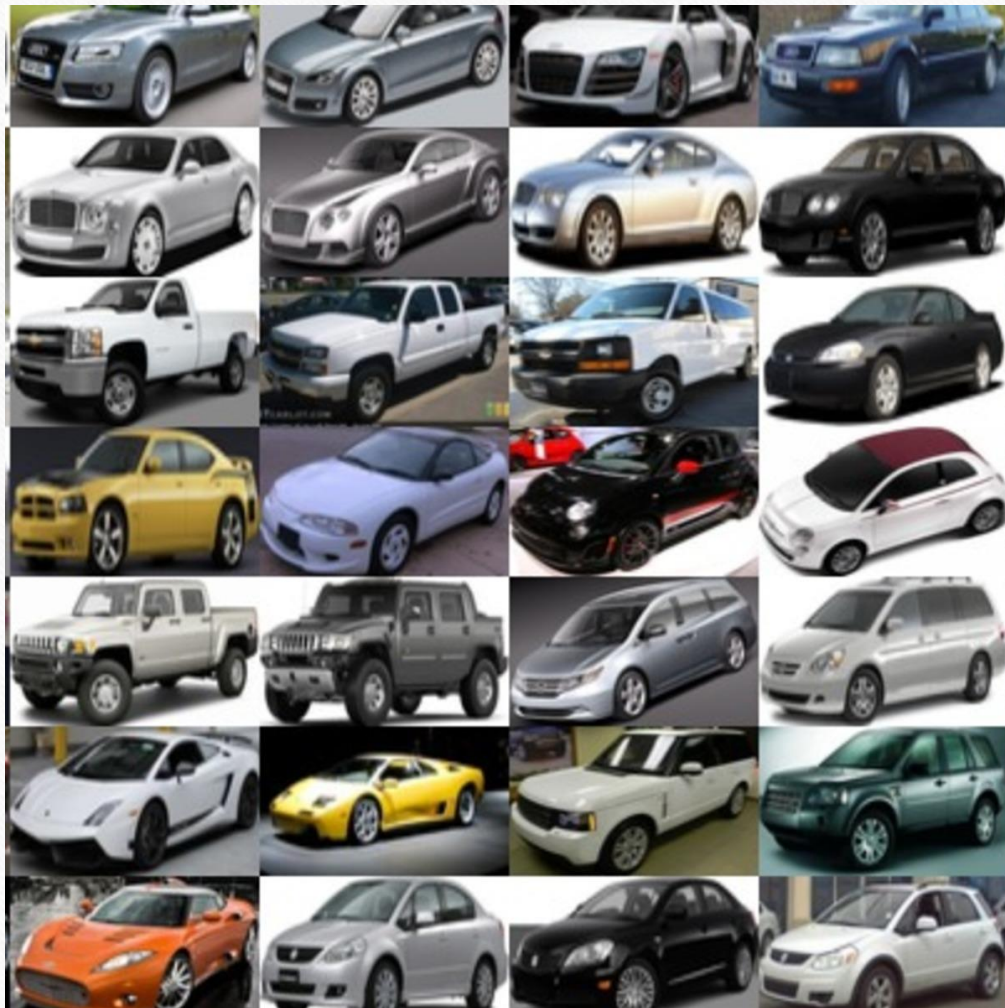
بدون پیش پردازش

نمونه :





## دیتاست Stanford Cars



شامل 16185 تصویر

داده ها

کلاس ها :

بدون پیش پردازش

196 ماشین مختلف

نمونه :



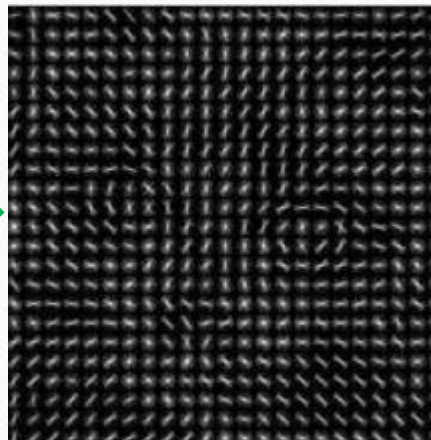
## مروری بر استفاده از الگوریتم های تشخیص ویژگی



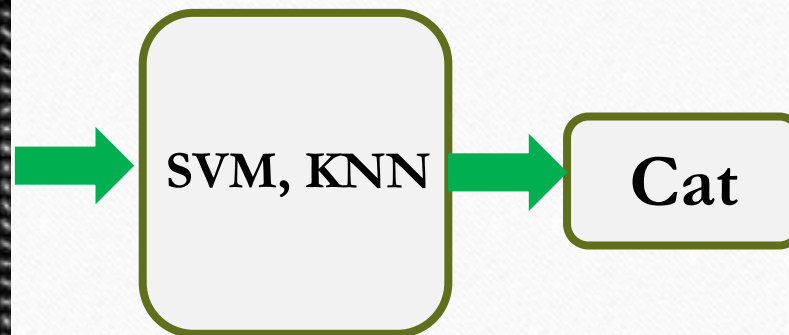
تصویر ورودی



پیش پردازش



استخراج ویژگی  
(SIFT, HOG, ...)



الگوریتم  
Classification

تشخیص  
شکل



حل یک مساله Classification ساده به کمک  
دیتاست IRIS



**Setosa**

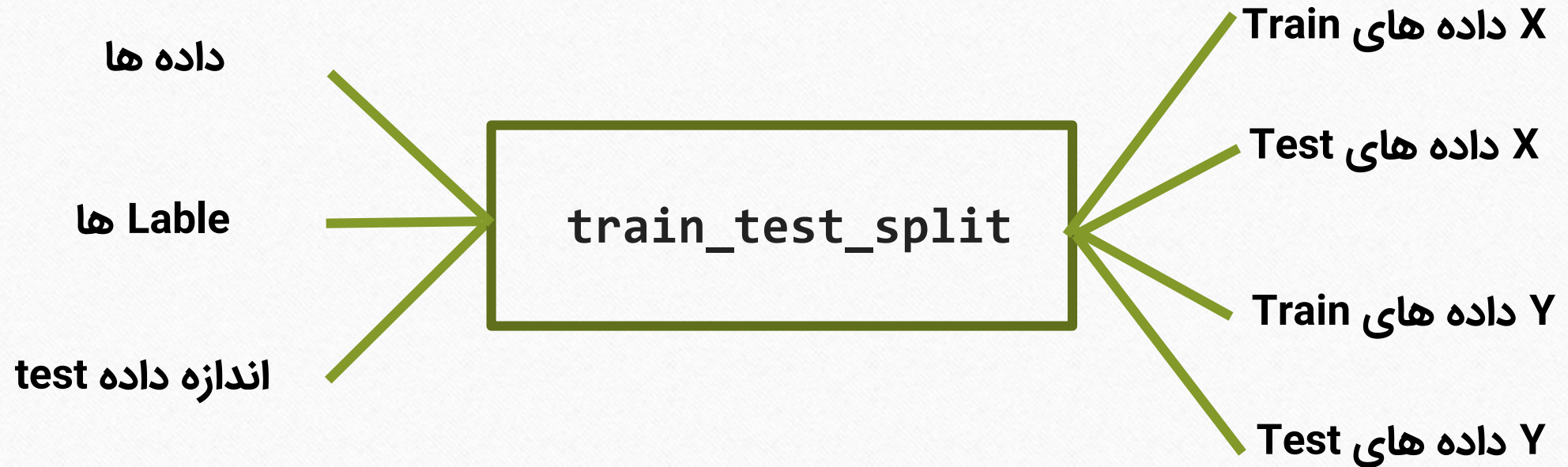


**Versicolor**



**Verginica**

## دستور 32 : جدا کردن داده های Train و Test



```
from sklearn.model_selection import train_test_split
```

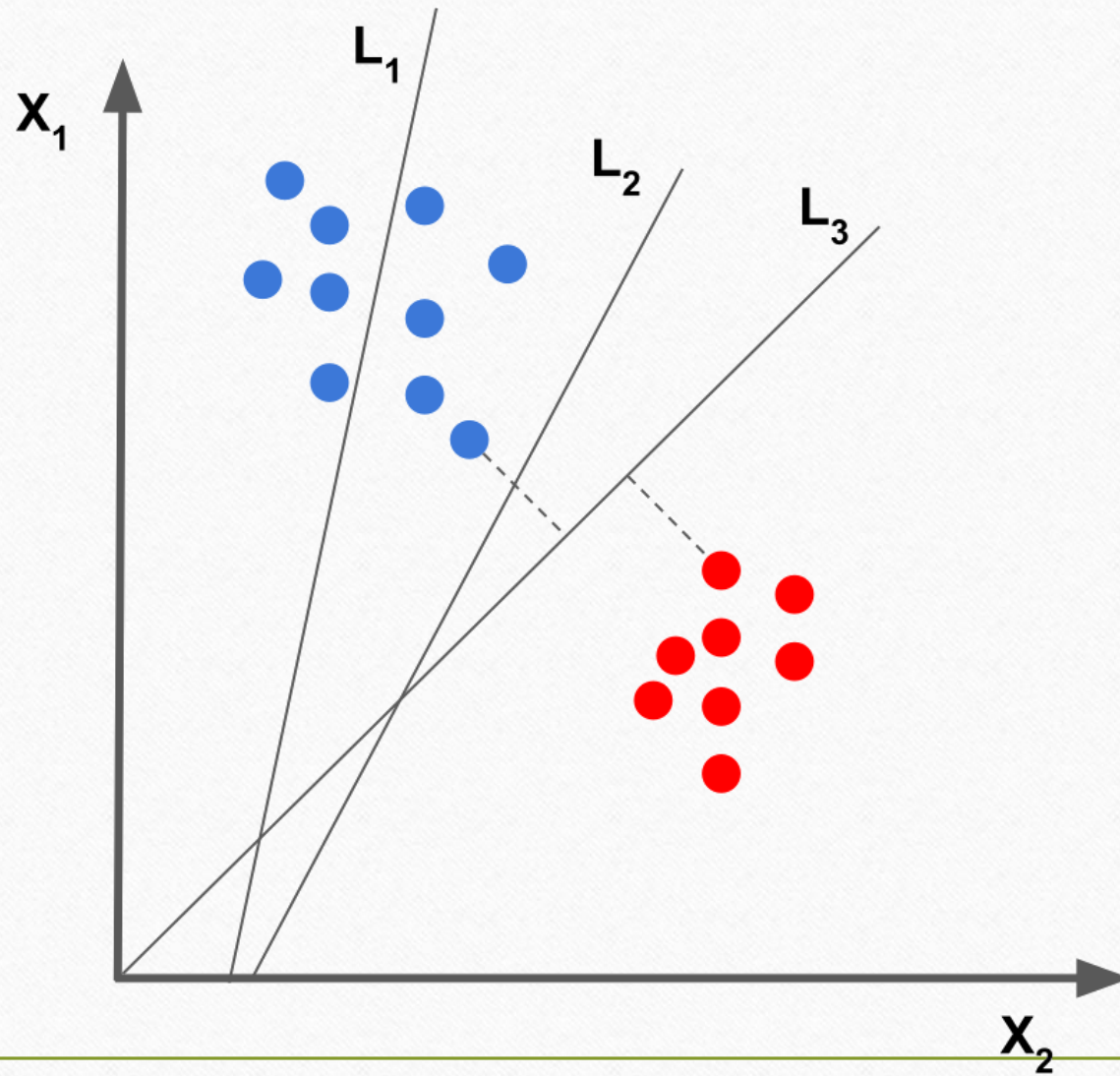
مثال :

```
X, x, y, y= train_test_split(train_data, lable, test_size = 0.33)
```



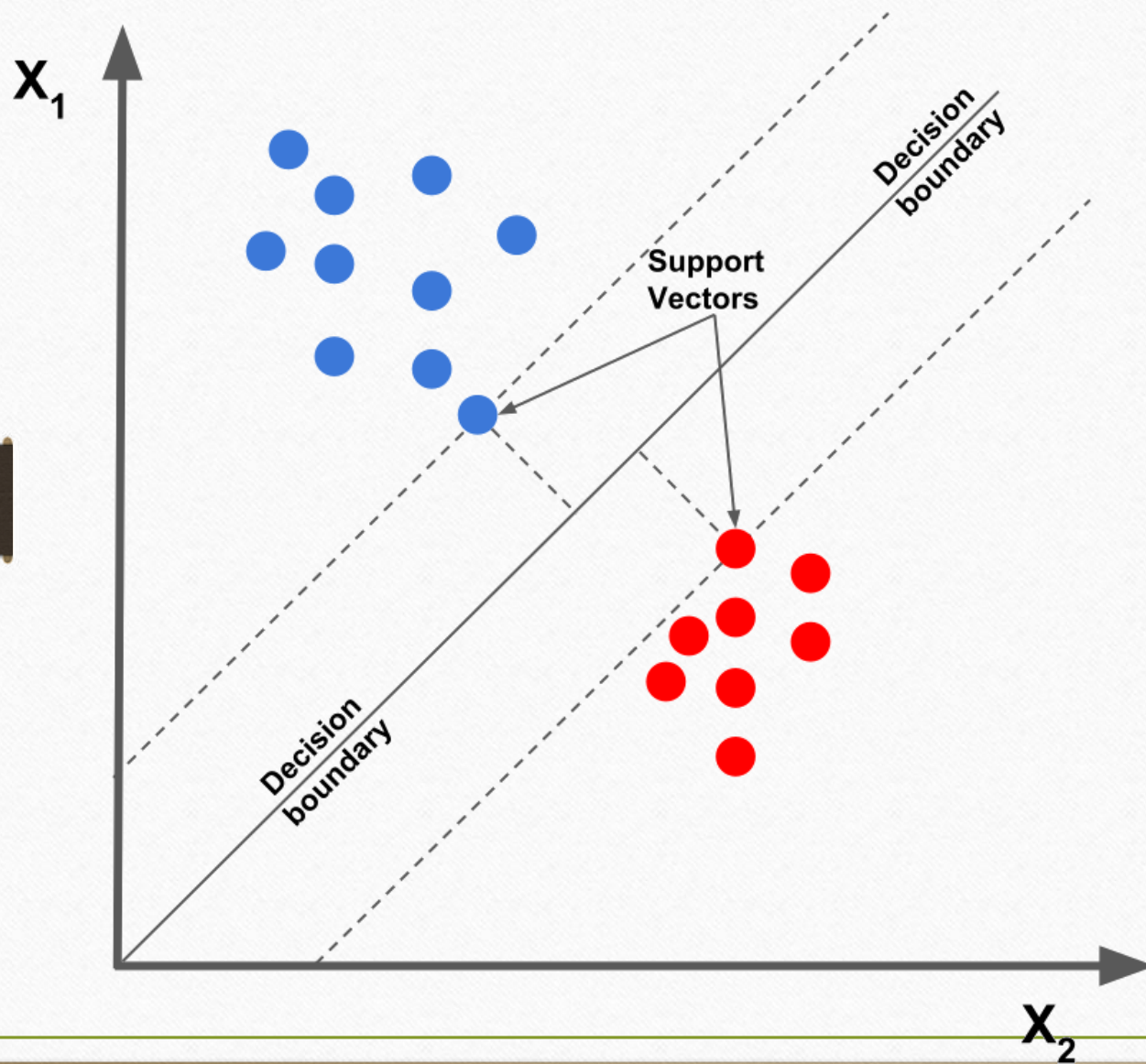
**SVM**

**Support Vector Machine**



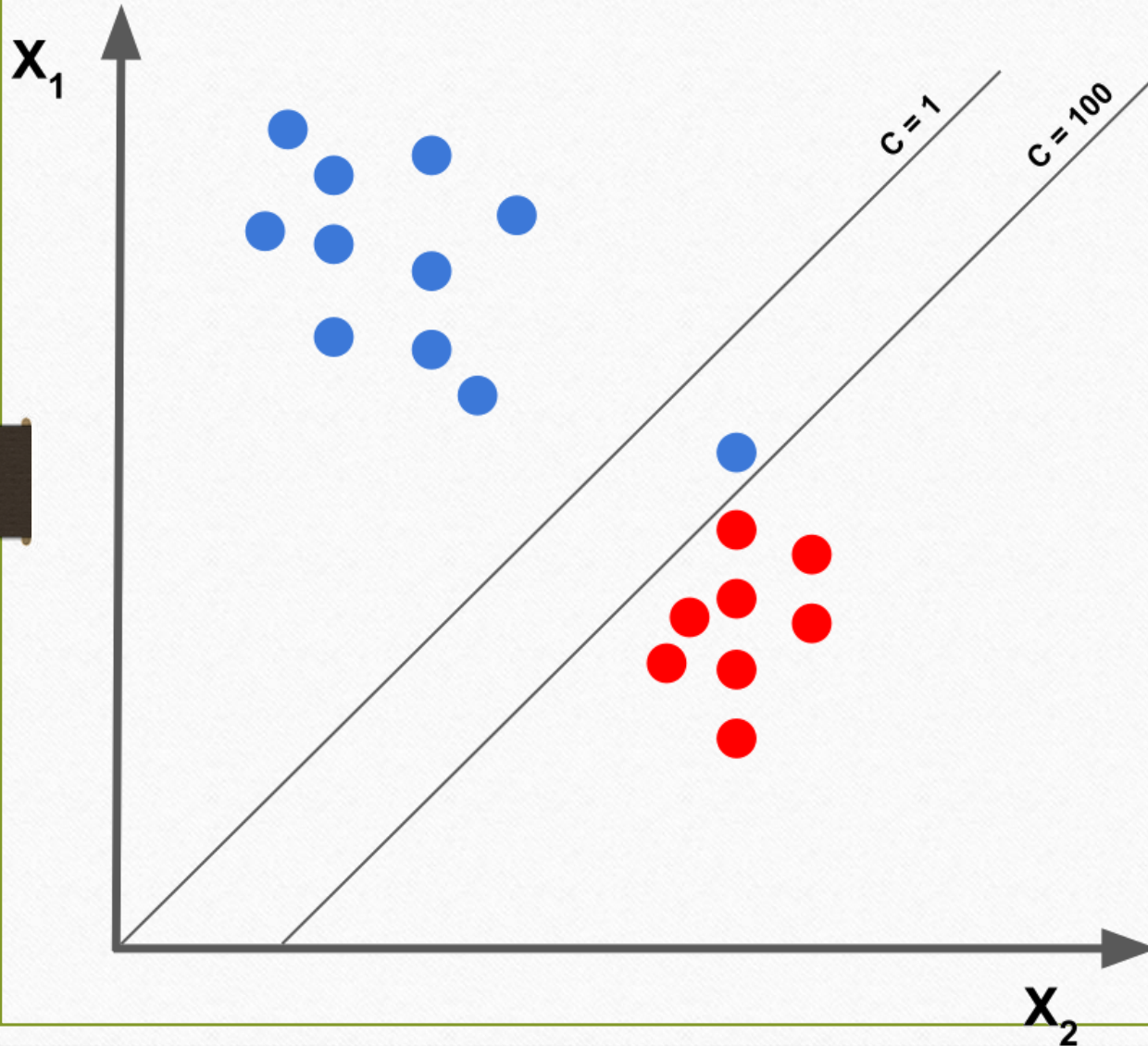
در SVM تلاش می کنیم تا عبارت زیر را  
ماکزیمم کنیم

$$\gamma = \min y_i \frac{w^T x_i + b}{\|w\|}$$





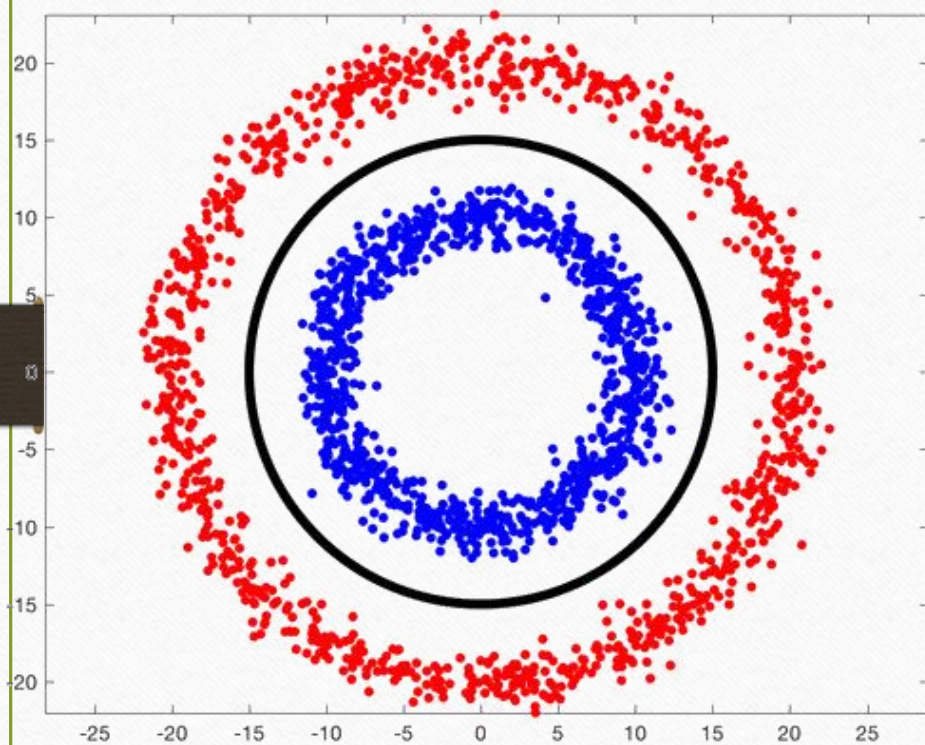
## پارامتر $C$ در SVM



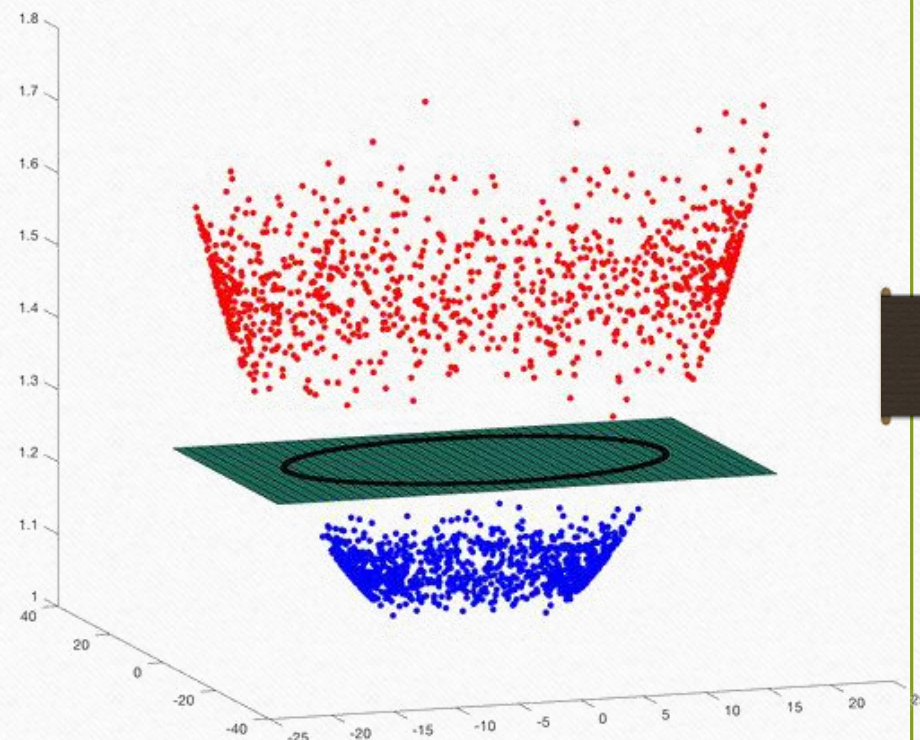
Trade off بین دقت و maximum margin

سوال : کدام خط بهتر است ؟

اگر داده ها به صورت خطی نباشند ؟



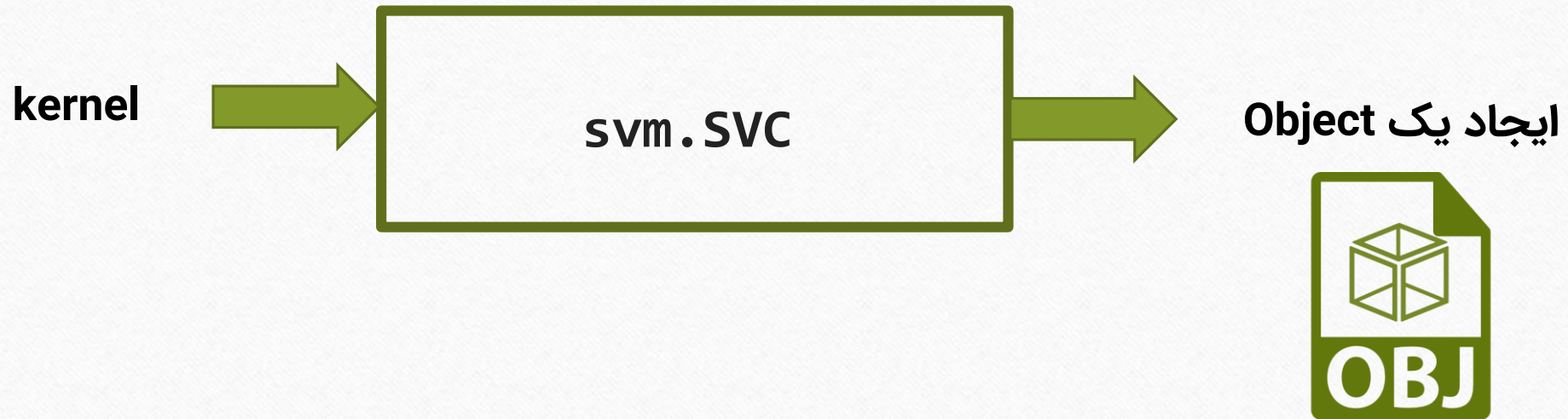
$$z = e^{-\gamma(x^2 + y^2)}$$





## دستور 32 : تعریف یک Object

---



---

```
from sklearn import svm
```

```
clf = svm.SVC()
```

مثال :

## دستور 32 : train ، Classifier ساخته شده

داده های ویژگی

Label ها

`clf.fit`

ایجاد مدل



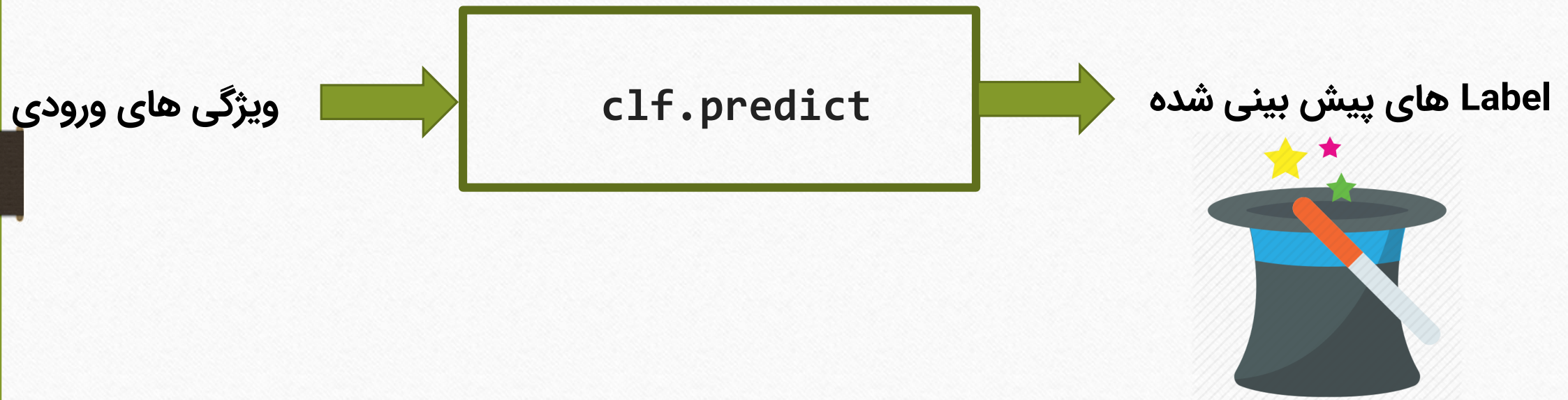
`clf.fit(X, y)`

مثال :



دستور 32 predict: به کمک مدل ساخته شده

---



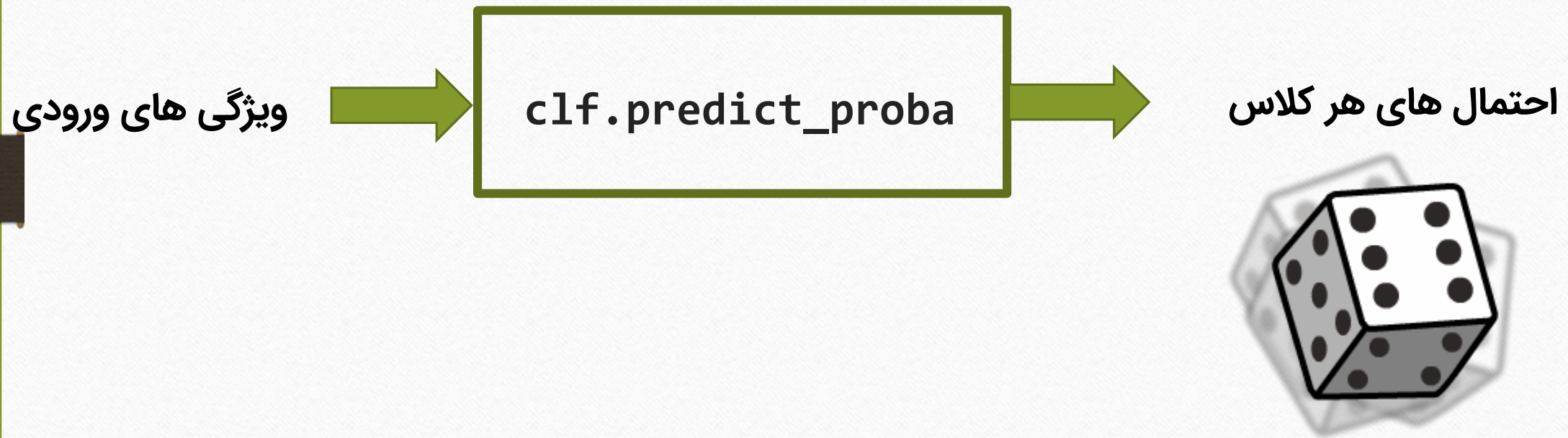
---

`clf.predict(X_test)`

مثال :

دستور 32 predict: با محاسبه احتمال هر کلاس

---



---

`clf.predict_proba(X_test)`

مثال :



## دستور 32 : محاسبه دقت Classifier

Label های واقعی

Label های پیش  
بینی شده

`clf.accuracy_score`

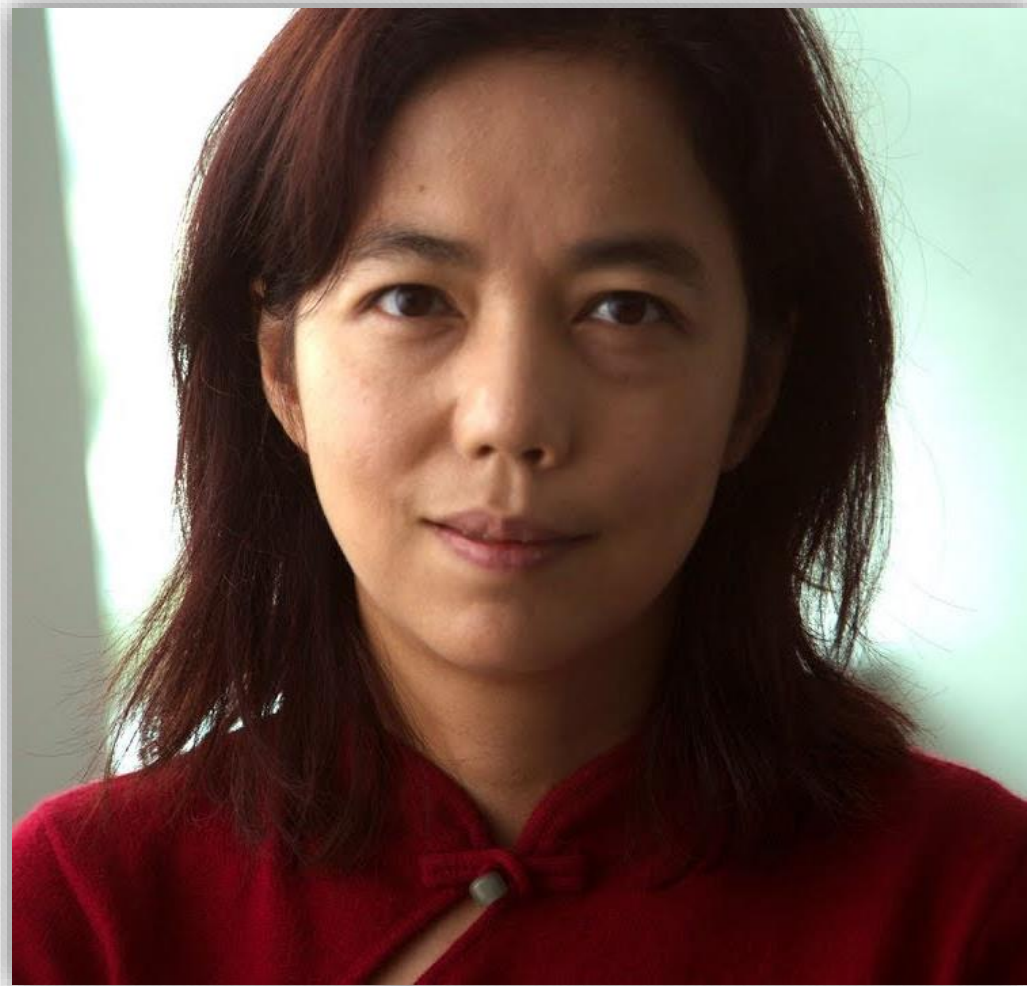
محاسبه دقت بر حسب  
درصد



```
acc = clf.accuracy_score(y_true, y_pred)
```

مثال :

## شخصیت هفته : Fei-Fei Li



➤ استاد دانشگاه Stanford

➤ مدیر آزمایشگاه هوش مصنوعی استنفورد

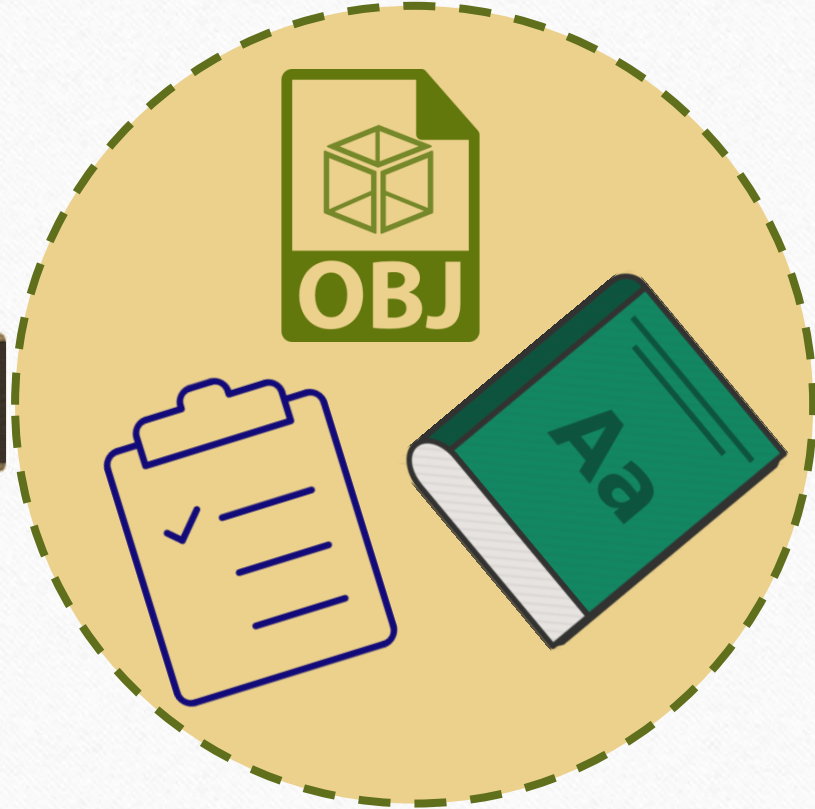
➤ مدرس دوره معروف CS231n



## تشخیص ماشین بودن یا نبودن یک تصویر



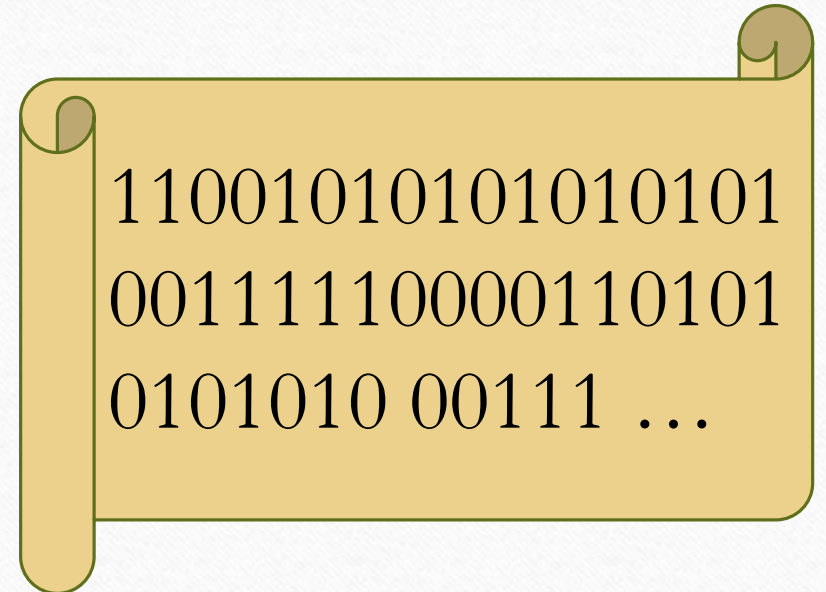
## ماژول Pickle



`pickle.dump ()`

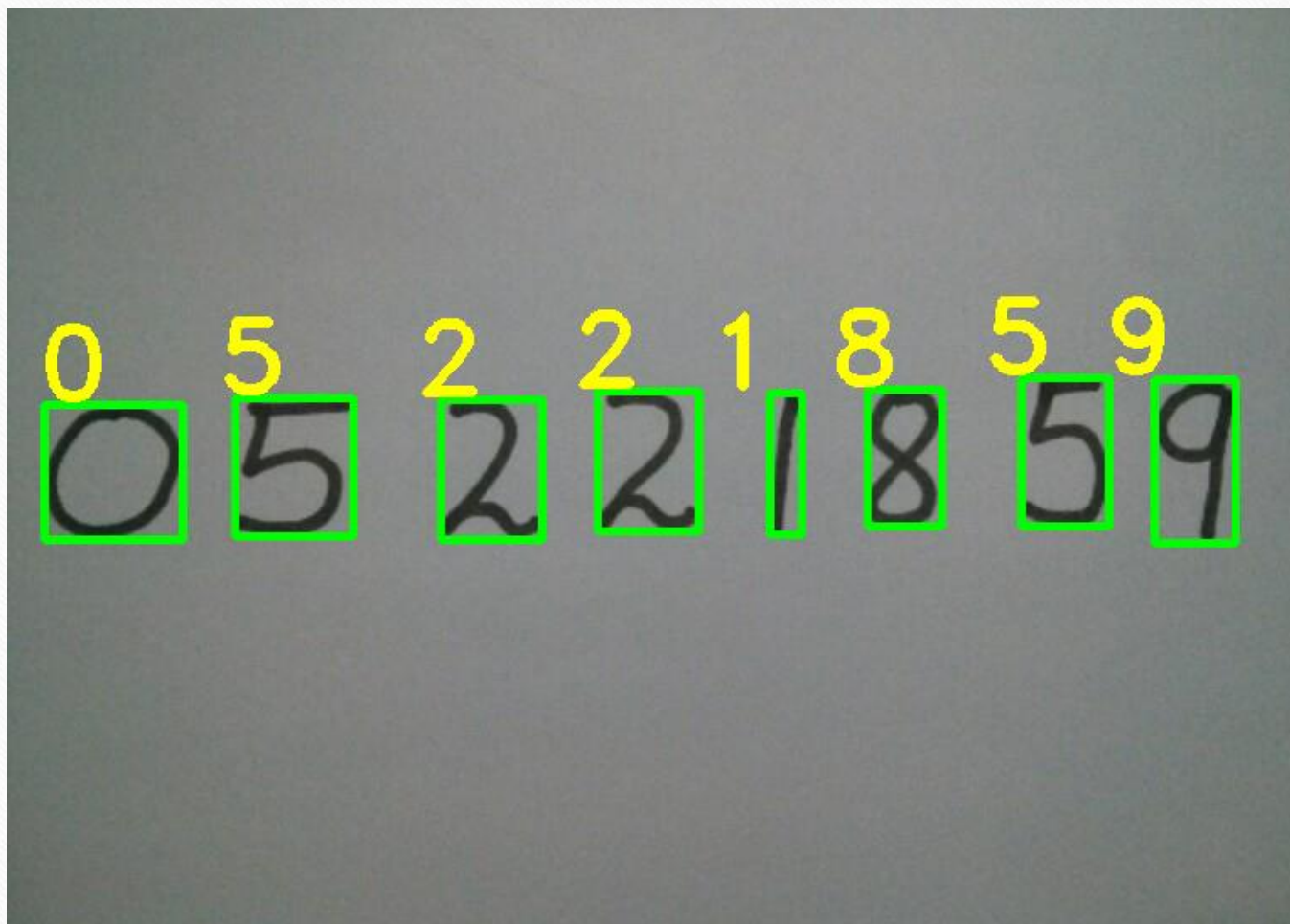


`pickle.load ()`





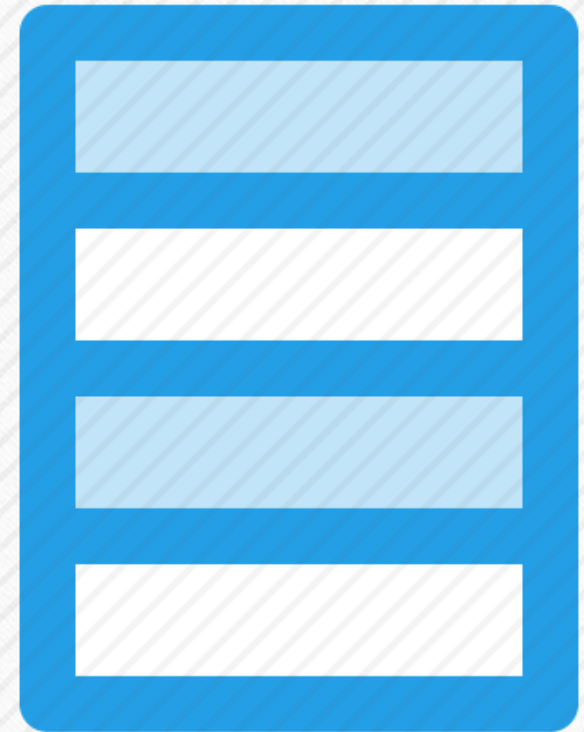
## تشخیص اعداد به کمک SVM و HOG



## گام 1 : اضافه کردن ماژول های مورد نیاز

```
from sklearn import datasets  
from skimage.feature import hog  
from sklearn.svm import LinearSVC  
import cv2
```

از ماژول dataset به منظور دانلود کتابخانه MNIST استفاده میکنیم.





## گام 2 : خواندن داده ها و استخراج ویژگی ها و lable ها

```
dataset = datasets.fetch_openml("mnist_784")  
features = np.array(dataset.data, 'int16')  
labels = np.array(dataset.target, 'int')
```

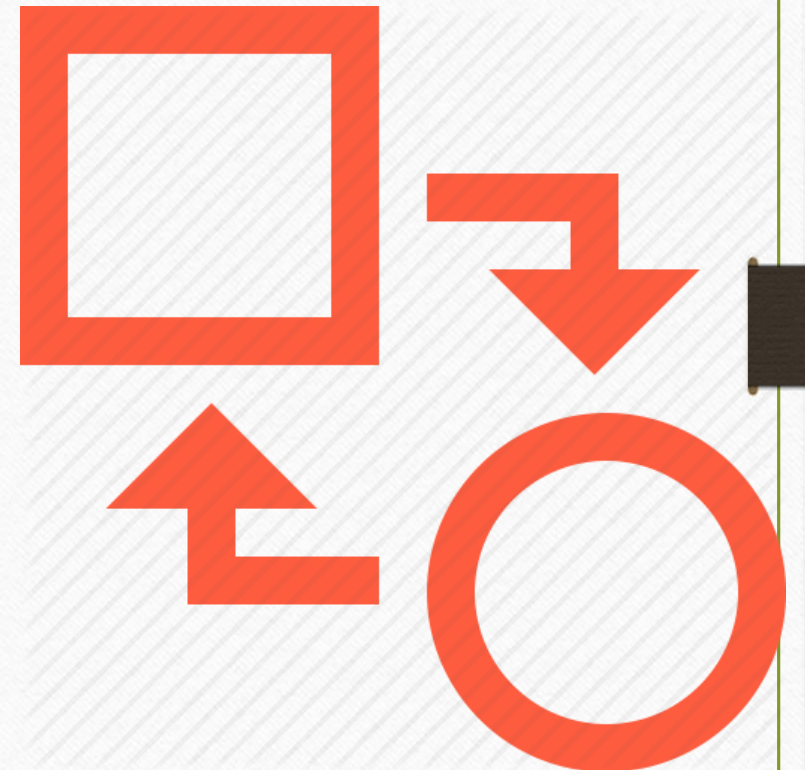
داده ها را باید به آرایه ای از numpy تبدیل کنیم.



### گام 3 : استخراج ویژگی به کمک HOG

```
list_hog_fd = []  
for feature in features:  
    fd = hog( . . . , visualise=False)  
    list_hog_fd.append(fd)  
hog_features = np.array(list_hog_fd, 'float64')
```

لازم است تا داده ها به تصاویر 28 در 28 تبدیل شوند.





## حل مساله Parkinson به کمک HOG و Classification

