

ارزیابی الگوریتم‌ها یادگیری ماشین (روش و معیار)

نادیه آرمین

گروه مهندسی کامپیوتر

دانشگاه فردوسی

پاییز ۹۹

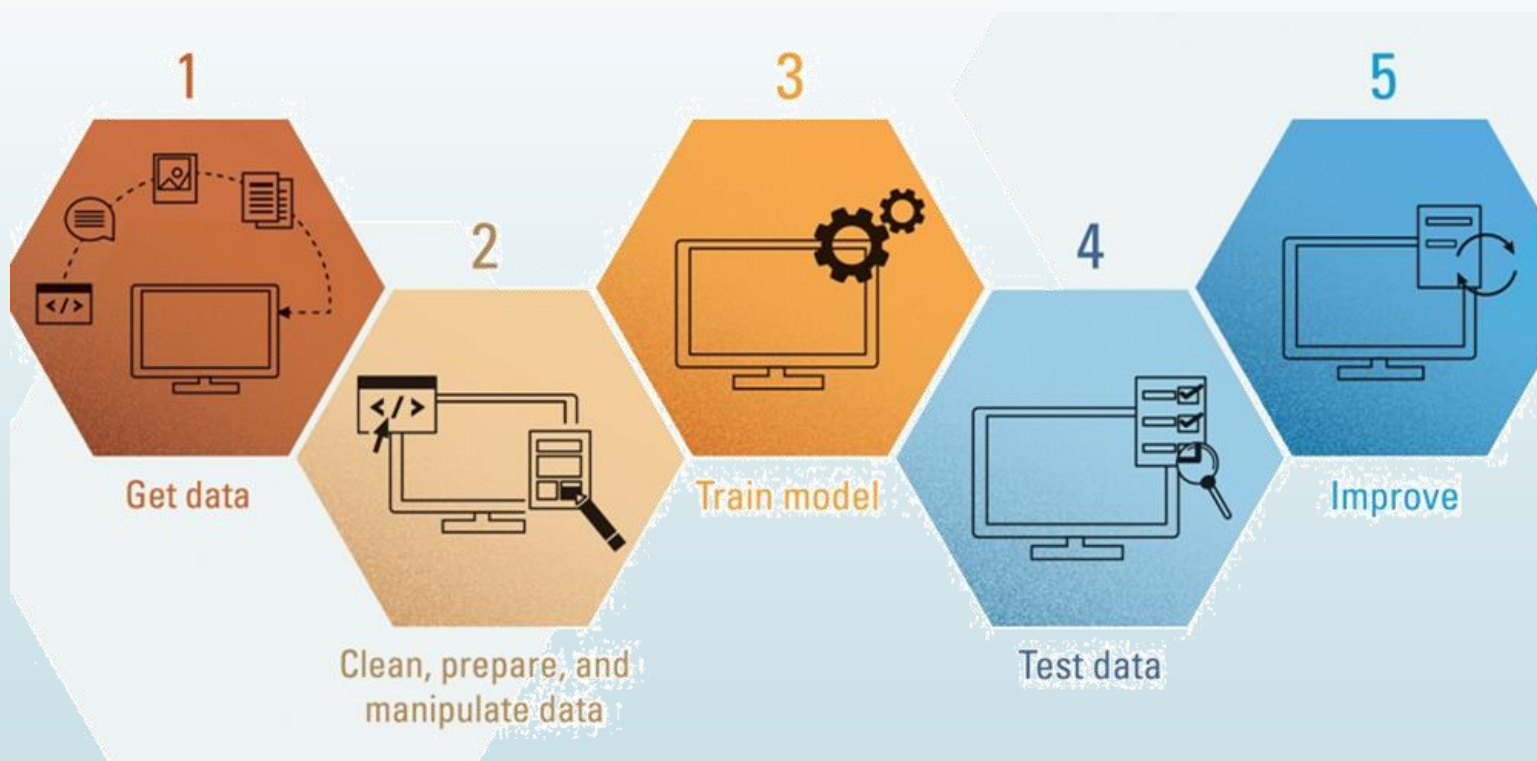
WT
laboratory



آزمایشگاه فناوری وب
Web Technology Lab

Web Technology Lab
آزمایشگاه فناوری وب

پروژه یادگیری ماشین



1. آماده‌سازی مسئله
2. شناخت داده‌ها
3. آماده‌سازی داده‌ها
4. ساخت و ارزیابی مدل
5. بهبود دقت
6. نهایی کردن مدل

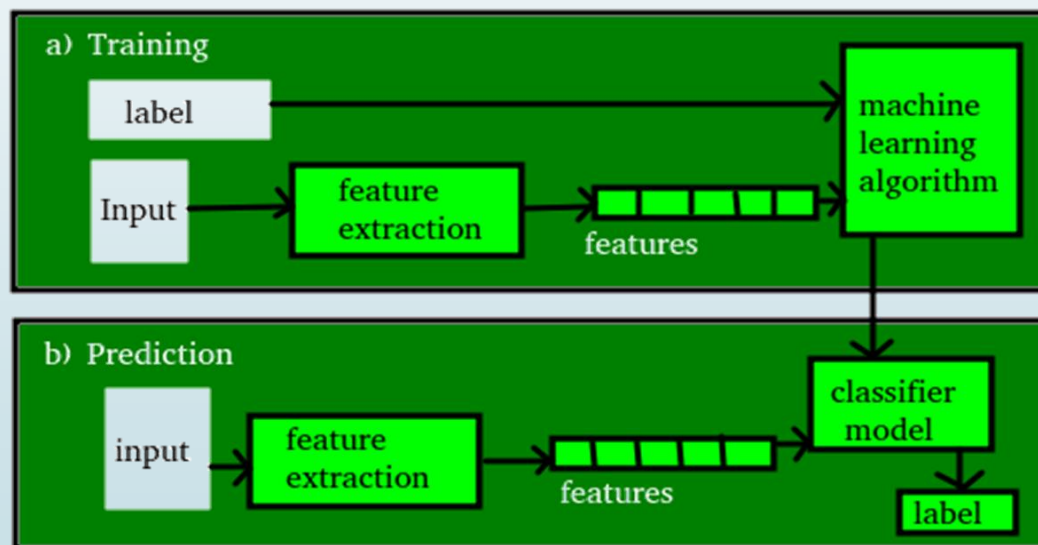
سر فصل

- روش ارزیابی
 - داده آموزشی و داده آزمایشی
 - اعتبارسنجی متقابل K تایی
 - اعتبارسنجی متقابل Leave One Out
 - تقسیم‌بندی داده تصادفی و تکراری
- معیار ارزیابی
 - معیارهای دسته‌بندی
 - معیارهای رگرسیون

✓ هدف ارزیابی:

مدل به چه میزان روی داده‌های ناشناخته خوب عمل می‌کند؟

□ بیش برآزش



بیش برآزش

overfitting

| | Underfitting | Just right | Overfitting |
|--------------------|---|--|---|
| علام | <ul style="list-style-type: none"> خطای بالای آموزش خطای آموزش نزدیک به خطای آزمایش پیش‌قدر زیاد | <ul style="list-style-type: none"> خطای آموزش کمی کمتر از خطای آزمایش | <ul style="list-style-type: none"> خطای آموزش بسیار کم خطای آموزش بسیار کمتر از خطای آزمایش واریانس بالا |
| نمایش وایزش | | | |
| نمایش دسته‌بندی | | | |
| نمایش یادگیری عمیق | | | |
| اصلاحات احتمالی | <ul style="list-style-type: none"> مدل را پیچیده‌تر کنید ویژگی‌های بیشتری اضافه کنید مدت طولانی‌تری آموزش دهید | | <ul style="list-style-type: none"> نظام‌بخشی انجام دهید داده‌های بیشتری گردآوری کنید |

ارزیابی: روش

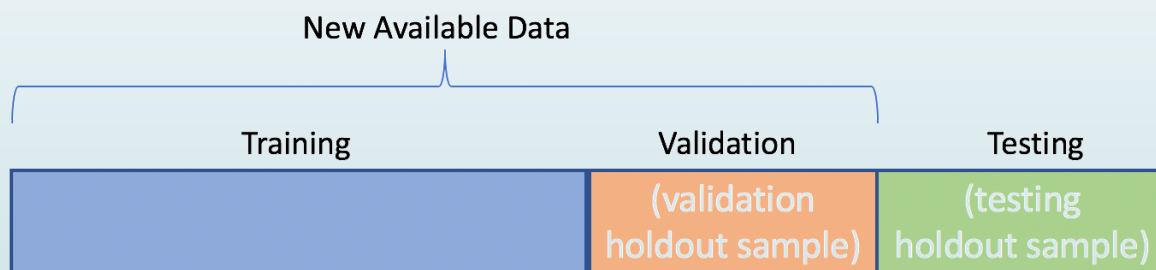
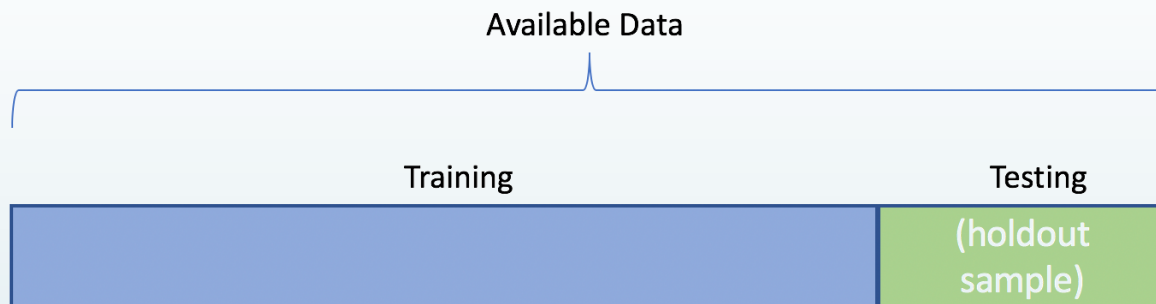
روش ارزیابی

- داده آموزشی و داده آزمایشی
- اعتبار سنجی متقابل K تایی
- اعتبار سنجی متقابل Leave One Out
- تقسیم‌بندی داده تصادفی و تکراری

روش ارزیابی: داده آموزشی و داده آزمایشی

تقسیم مجموعه داده:

- بخش آموزش (۶۷٪)
- بخش تست (۳۳٪)



- ✓ ساده ترین روش
- ✓ سریع ترین روش
- ✓ مناسب برای داده های بزرگ

- تفاوت در دو مجموعه منجر به تفاوت در دقت

■ بذر تصادفی (Random seed)

روش ارزیابی: داده آموزشی و داده آزمایشی

```
from pandas import read_csv
```

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

```
from sklearn.linear_model import LogisticRegression
```

```
filename = 'pima-indians-diabetes.csv'
```

```
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
```

```
dataframe = read_csv(filename, names=names)
```

```
array = dataframe.values
```

```
X = array[:, 0:8]
```

```
Y = array[:, 8]
```

```
test_size = 0.33
```

```
seed = 7
```

```
X_train,X_test,Y_train,Y_test = train_test_split(X,Y,test_size=test_size ,random_state=seed)
```

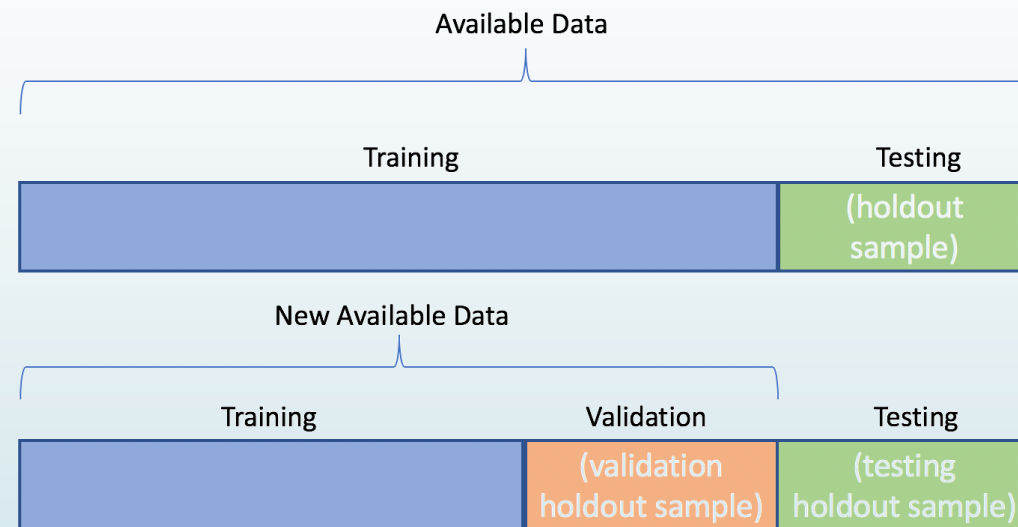
```
model =LogisticRegression(max_iter=1000)
```

```
model.fit(X_train,Y_train)
```

```
result = model.score(X_test,Y_test)
```

```
print("Accuracy: %.3f" %(result*100.0))
```

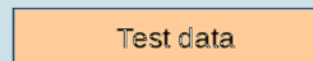
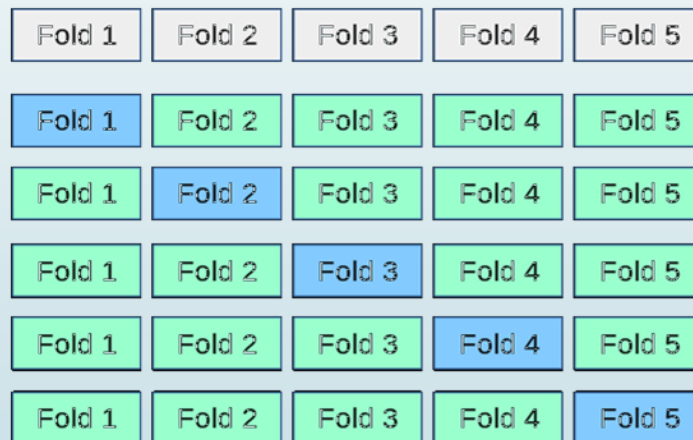
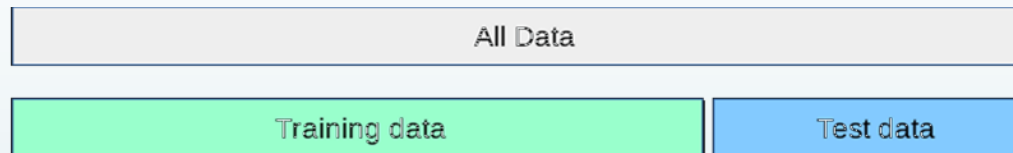
یادگیری ماشین با پایتون



روش ارزیابی: اعتبارسنجی متقابل K تایی

تقسیم مجموعه داده به K بخش:

- بخش آموزش (K-1 بخش)
- بخش تست (1 بخش)



✓ قابل اطمینان

✓ برای مجموعه داده با اندازه متوسط (هزاران یا ده ها هزار رکورد) استاندارد طلایی ($K=3,5,10$)

■ اندازه k باید هر بخش مجموعه تست را به اندازه کافی بزرگ بگیرد تا نمونه معقولی از مساله باشد

روش ارزیابی: اعتبارسنجی متقابل K تایی

```
from pandas import read_csv

from sklearn.model_selection import KFold

from sklearn.model_selection import cross_val_score

from sklearn.linear_model import LogisticRegression

filename = 'pima-indians-diabetes.csv'

names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']

dataframe = read_csv(filename, names=names)

array = dataframe.values

X = array[:, 0:8]

Y = array[:, 8]

num_folds = 10

seed = 7

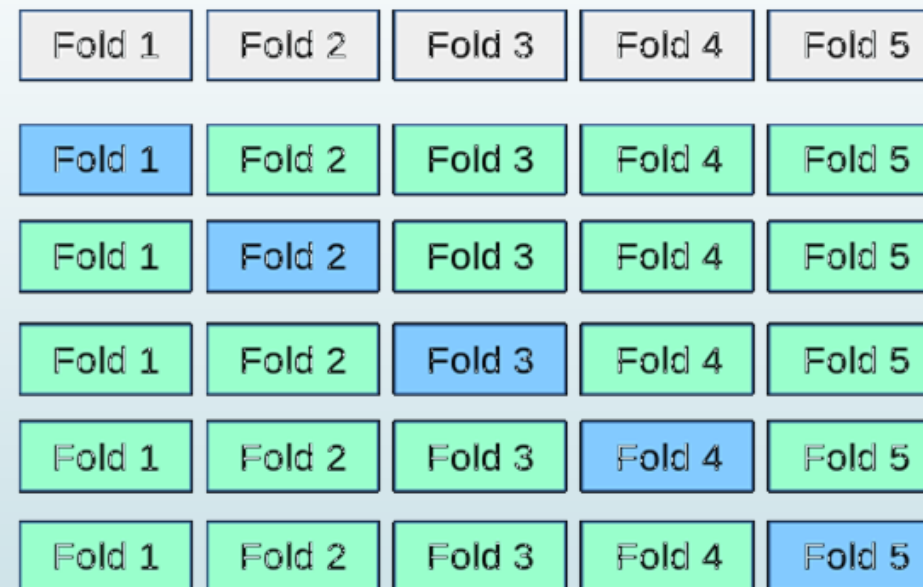
Kfold = KFold(n_splits=num_folds, random_state=seed, shuffle=True)

model = LogisticRegression(max_iter=1000)

result = cross_val_score(model, X, Y, cv=Kfold)

print("Accuracy: %.3f (%.3f)" % (result.mean()*100.0, result.std()*100.0))
```

یادگیری ماشین با پایتون



روش ارزیابی: اعتبارسنجی متقابل Leave One Out

اعتبارسنجی متقابل با $K=1$

```
from pandas import read_csv
```

```
from sklearn.model_selection import LeaveOneOut
```

```
from sklearn.model_selection import cross_val_score
```

```
from sklearn.linear_model import LogisticRegression
```

```
filename = 'pima-indians-diabetes.csv'
```

```
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
```

```
dataframe = read_csv(filename, names=names)
```

```
array = dataframe.values
```

```
# separate array into input and output component
```

```
X = array[:, 0:8]
```

```
Y = array[:, 8]
```

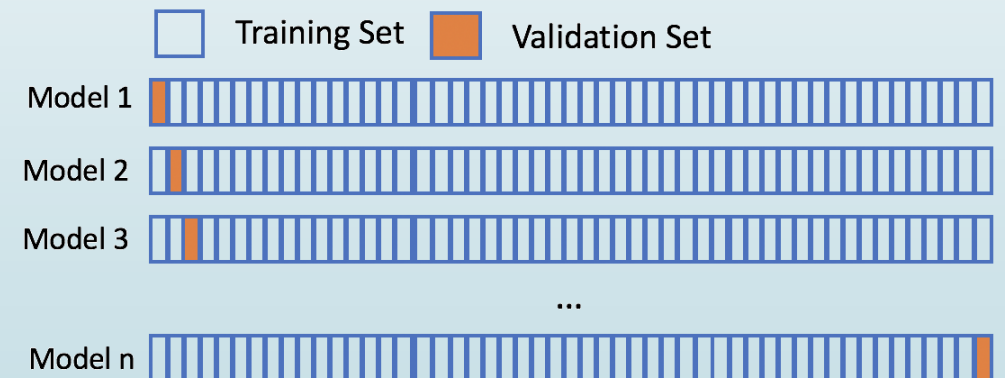
```
loocv=LeaveOneOut()
```

```
model=LogisticRegression(max_iter=1000)
```

```
result=cross_val_score(model,X,Y,cv=loocv)
```

```
print("Accuracy: %.3f (%.3f)" %(result.mean()*100.0 ,result.std()*100.0))
```

- نتایج متغیرتری نسبت به K های بالاتر دارد.
(واریانس نتایج بیشتر است)



روش ارزیابی: تقسیم‌بندی داده تصادفی و تکراری

```
from pandas import read_csv
```

```
from sklearn.model_selection import ShuffleSplit
```

```
from sklearn.model_selection import cross_val_score
```

```
from sklearn.linear_model import LogisticRegression
```

```
filename = 'pima-indians-diabetes.csv'
```

```
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
```

```
dataframe = read_csv(filename, names=names)
```

```
array = dataframe.values
```

```
X = array[:, 0:8]
```

```
Y = array[:, 8]
```

```
n_splits=10
```

```
test_size=0.33
```

```
kfold=ShuffleSplit(n_splits=n_splits, test_size=test_size, random_state=7)
```

```
model = LogisticRegression(max_iter=1000)
```

```
result = cross_val_score(model, X, Y, cv=kfold)
```

```
print("Accuracy: %.3f (%.3f)" % (result.mean()*100.0, result.std()*100.0))
```

یادگیری ماشین با پایتون

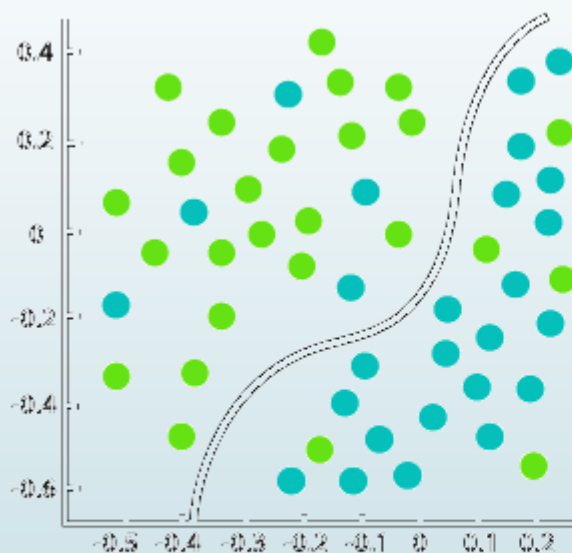
اعتبارسنجی متقابل K تایی به صورت تصادفی:

- بخش آموزش (۶۷٪)

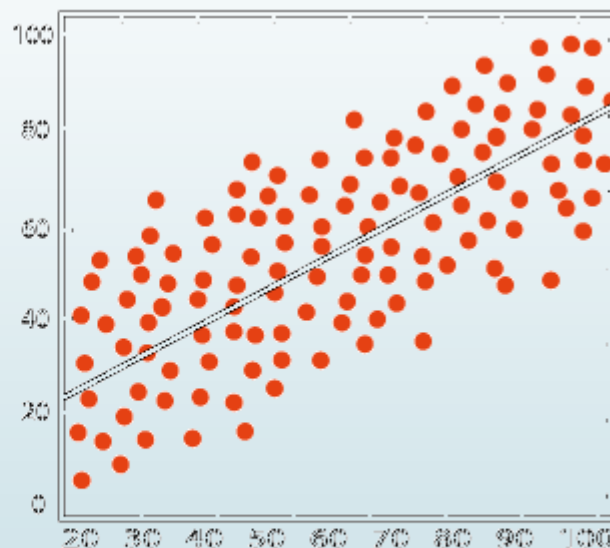
- بخش تست (۳۳٪)

- افزونگی در ارزیابی

ارزیابی : معیار



Classification



Regression

- معیارهای دسته‌بندی
- ماتریس درهم ریختگی
- صحت دسته بندی
- گزارش دسته بندی
- اتلاف لگاریتمی
- مساحت زیر منحنی ROC
- معیارهای رگرسیون
- میانگین قدر مطلق خطا
- میانگین مربعات خطا
- R^2

ارزیابی : معیار

معیارهای دسته‌بندی

- ماتریس درهم ریختگی
- صحت دسته بندی
- گزارش دسته بندی
- اتلاف لگاریتمی
- مساحت زیر منحنی ROC

معیار ارزیابی (دسته بندی): ماتریس درهم ریختگی

پیش بینی ها روی محور X
مقادیر واقعی روی محور Y

| دسته پیش بینی شده | | | |
|-------------------|---|--|---|
| | | + | - |
| دسته واقعی | + | TP True Positives | FN False Negatives Type II error |
| | - | FP False Positives Type I error | TN True Negatives |

معیار ارزیابی (دسته بندی): ماتریس درهم ریختگی

```
from pandas import read_csv
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix

filename = 'pima-indians-diabetes.csv'
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
dataframe = read_csv(filename, names=names)
array = dataframe.values
X = array[:, 0:8]
Y = array[:, 8]

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.33, random_state=7)
model = LogisticRegression(max_iter=1000)
model.fit(X_train, Y_train)
predicted = model.predict(X_test)
matrix = confusion_matrix(Y_test, predicted)
print(matrix)
```

یادگیری ماشین با پایتون

■ مقادیر روی قطر اصلی: تعداد پیش بینی های درست

دسته پیش بینی شده

| | + | - |
|------------|--|---|
| دسته واقعی | <p>TP True Positives</p> | <p>FN False Negatives Type II error</p> |
| | <p>FP False Positives Type I error</p> | <p>TN True Negatives</p> |

معیار ارزیابی (دسته بندی): صحت دسته بندی

```
# Cross Validation Classification Accuracy
```

```
from pandas import read_csv
```

```
from sklearn.model_selection import KFold
```

```
from sklearn.model_selection import cross_val_score
```

```
from sklearn.linear_model import LogisticRegression
```

```
filename = 'pima-indians-diabetes.csv'
```

```
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
```

```
dataframe = read_csv(filename, names=names)
```

```
array = dataframe.values
```

```
X = array[:, 0:8]
```

```
Y = array[:, 8]
```

```
Kfold=KFold(n_splits=10,random_state=7 ,shuffle=True)
```

```
model = LogisticRegression(max_iter=1000)
```

```
scoring='accuracy'
```

```
result=cross_val_score(model, X ,Y ,cv=Kfold, scoring=scoring)
```

```
print("Accuracy: %.3f (%.3f)" %(result.mean()*100.0 ,result.std()*100.0))
```

یادگیری ماشین با پایتون

تعداد پیش بینی های صحیح به نسبت
همه پیش بینی ها

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

✓ متداول ترین معیار کارایی برای دسته بندی

- مناسب است اگر تعداد مشاهدات مساوی در
هر دسته باشند.

- تمام پیش بینی ها و خطاهای پیش بینی به
یک اندازه مهم است

معیار ارزیابی (دسته بندی): گزارش دسته بندی

گزارش سریعی از دقت و
فراخوانی و امتیاز f

```
from pandas import read_csv  
from sklearn.model_selection import train_test_split  
from sklearn.linear_model import LogisticRegression  
from sklearn.metrics import classification_report  
  
filename = 'pima-indians-diabetes.csv'  
  
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']  
  
dataframe = read_csv(filename, names=names)  
  
array = dataframe.values  
  
X = array[:, 0:8]  
  
Y = array[:, 8]  
  
X_train,X_test,Y_train,Y_test = train_test_split(X,Y,test_size=0.33 ,random_state=7)  
  
model =LogisticRegression(max_iter=1000)  
  
model.fit(X_train,Y_train)  
  
predicted = model.predict(X_test)  
  
report = classification_report(Y_test,predicted)  
  
print(report)  
یادگیری ماشین با پایتون
```

معیار ارزیابی (دسته بندی): گزارش دسته بندی

معیارهای متداول برای الگوریتمهای دسته بندی

| معیار | فرمول |
|------------------------|-------------------------------------|
| صحت (Accuracy) | $\frac{TP + TN}{TP + TN + FP + FN}$ |
| دقت (Precision) | $\frac{TP}{TP + FP}$ |
| فراخوانی (Recall) | $\frac{TP}{TP + FN}$ |
| ویژگی (Specificity) | $\frac{TN}{TN + FP}$ |
| F1 score | $\frac{2TP}{2TP + FP + FN}$ |

| | | Predicted Class | | |
|--------------|----------|--|--|--|
| | | Positive | Negative | |
| Actual Class | Positive | True Positive (TP) | False Negative (FN) Type II Error | Sensitivity $\frac{TP}{(TP + FN)}$ |
| | Negative | False Positive (FP) Type I Error | True Negative (TN) | Specificity $\frac{TN}{(TN + FP)}$ |
| | | Precision $\frac{TP}{(TP + FP)}$ | Negative Predictive Value $\frac{TN}{(TN + FN)}$ | Accuracy $\frac{TP + TN}{(TP + TN + FP + FN)}$ |

معیار ارزیابی (دسته بندی): اتلاف لگاریتمی

```
from pandas import read_csv
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LogisticRegression
filename = 'pima-indians-diabetes.csv'
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
dataframe = read_csv(filename, names=names)
array = dataframe.values
X = array[:, 0:8]
Y = array[:, 8]
Kfold=KFold(n_splits=10,random_state=7 ,shuffle=True)
model = LogisticRegression(max_iter=1000)
scoring='neg_log_loss'
result=cross_val_score(model, X ,Y ,cv=Kfold, scoring=scoring)
print("Logloss: %.3f (%.3f)" %(result.mean()*100.0 ,result.std()*100.0))
```

یادگیری ماشین با پایتون

Logloss

برای ارزیابی پیش بینی های احتمالات
عضویت در یک کلاس معین

میزان عدم اطمینان از پیش بینی بر
اساس میزان تفاوت آن از برچسب اصلی

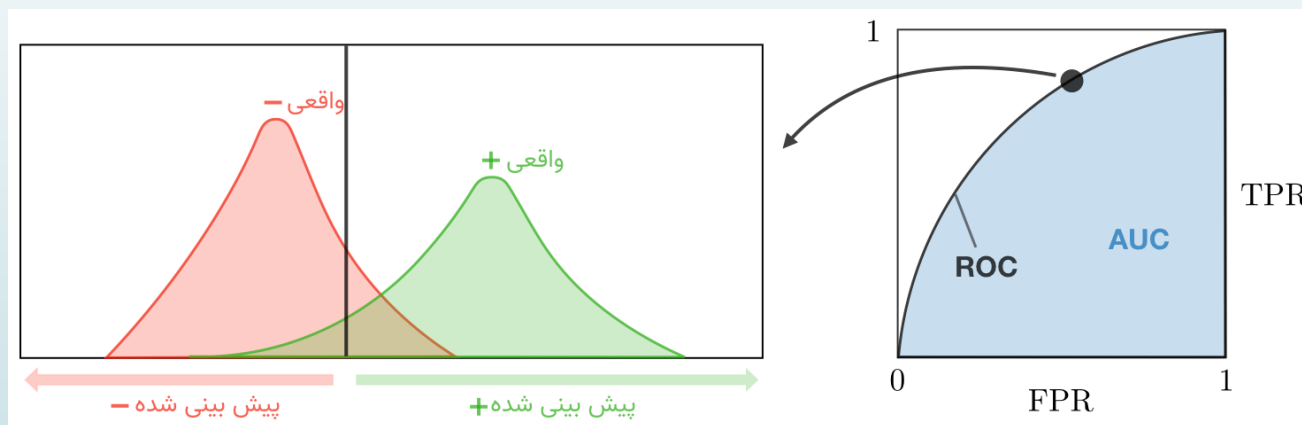
$$LogLoss = -\frac{1}{n} \sum_{i=0}^n [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

✓ هرچه کمتر بهتر

معیار ارزیابی (دسته بندی): مساحت زیر منحنی ROC

AUC (Area Under Curve)– ROC(Receiver Operating Characteristic)

منحنی ROC: نسبت «نرخ مثبت صحیح» TPR (True Positive Rate) بر حسب «نرخ مثبت کاذب» FPR (False Positive Rate)



$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN} = 1 - FNR$$

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN} = 1 - TNR$$

معیار ارزیابی (دسته بندی): مساحت زیر منحنی ROC

```
from pandas import read_csv
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LogisticRegression
filename = 'pima-indians-diabetes.csv'
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
dataframe = read_csv(filename, names=names)
array = dataframe.values
X = array[:, 0:8]
Y = array[:, 8]
Kfold=KFold(n_splits=10,random_state=7 ,shuffle=True)
model = LogisticRegression(max_iter=1000)
scoring='roc_auc'
result=cross_val_score(model, X ,Y ,cv=Kfold, scoring=scoring)
print('AUC: %.3f (%.3f)' %(result.mean()*100.0 ,result.std()*100.0))
```

معیار عملکرد برای دسته بندی های دودویی

مقدار ۱ : تمام پیش بینی ها صحیح

مقدار ۰/۵ : پیش بینی ها تصادفی

ارزیابی : معیار

معیارهای رگرسیون

- میانگین قدر مطلق خطا
- میانگین مربعات خطا
- R^2

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}|$$

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y})^2$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y})^2}$$

$$R^2 = 1 - \frac{\sum (y_i - \hat{y})^2}{\sum (y_i - \bar{y})^2}$$

Where,

\hat{y} - predicted value of y

\bar{y} - mean value of y

معیار ارزیابی (رگرسیون): میانگین قدر مطلق خطا

```
from pandas import read_csv
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LinearRegression

filename = 'california_housing_train.csv'
dataframe = read_csv(filename,header=0)
array = dataframe.values
X = array[:, 0:8]
Y = array[:, 8]

Kfold=KFold(n_splits=10,random_state=7,shuffle=True)
model = LinearRegression ()

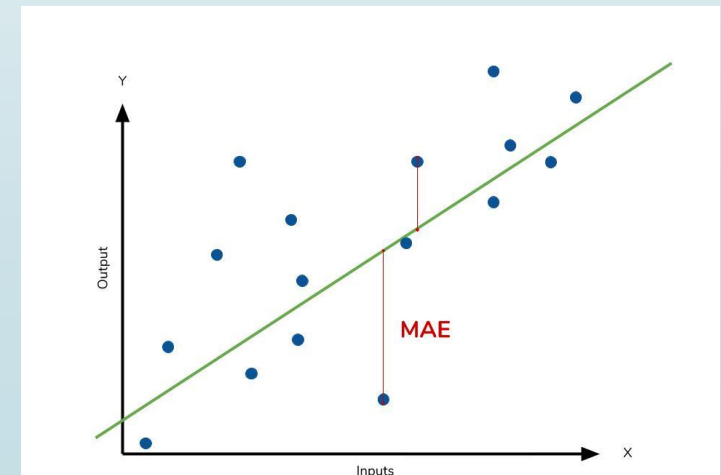
scoring='neg_mean_absolute_error'

result=cross_val_score(model, X ,Y ,cv=Kfold, scoring=scoring)
print("MAE: %.3f (%.3f)" %(result.mean()*100.0 ,result.std()*100.0))
```

MAE: مجموع قدر مطلق تفاضل
بین پیش بینی ها و مقادیر واقعی (خطا)

$$MAE = \frac{1}{n} \sum_{i=1}^n \underbrace{|y_i - \hat{y}_i|}_{\text{predicted value} - \text{actual value}}$$

test set



معیار ارزیابی (رگرسیون): میانگین مربعات خطا

```
from pandas import read_csv
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LinearRegression

filename = 'california_housing_train.csv'
dataframe = read_csv(filename,header=0)
array = dataframe.values
X = array[:, 0:8]
Y = array[:, 8]

Kfold=KFold(n_splits=10,random_state=7,shuffle=True)
model = LinearRegression ()

scoring='neg_mean_squared_error'

result=cross_val_score(model, X ,Y ,cv=Kfold, scoring=scoring)

print("MSE: %.3f (%.3f)" %(result.mean()*100.0 ,result.std()*100.0))
```

MSE : مربع تفاضل بین پیش بینی ها و مقادیر واقعی (خطا)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

test set
predicted vaue
actual value

معیار ارزیابی (رگرسیون): R^2

ضریب دترمینان
مقداری بین ۰ و ۱

```
from pandas import read_csv
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LinearRegression

filename = 'california_housing_train.csv'
dataframe = read_csv(filename,header=0)
array = dataframe.values
X = array[:, 0:8]
Y = array[:, 8]
Kfold=KFold(n_splits=10,random_state=7,shuffle=True)
model = LinearRegression ()
scoring='r2'
result=cross_val_score(model, X ,Y ,cv=Kfold, scoring=scoring)
print("R^2: %.3f (%.3f)" %(result.mean()*100.0,result.std()*100.0))
```

$$\hat{R}^2 = 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} = 1 - \frac{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\frac{1}{n} \sum_{i=1}^n (Y_i - \bar{Y})^2}$$

۰: عدم برازش
۱: برازش کامل

