Machine Learning - Tests

Nicolas Bourgeois

Rappel des Hypothèses

Observations :

- Variable empirique cible $\tilde{Y}$
- Variables empiriques explicatives $\tilde{X}$

Rappel des Hypothèses

Observations :

- Variable empirique cible $\tilde{Y}$
- Variables empiriques explicatives $\tilde{X}$

Hypothèses :

- $\tilde{X}$ est un ensemble d'observations lié à un processus aléatoire X
- $\tilde{Y}$ est un ensemble d'observations lié à un processus aléatoire Y
- il existe une relation $Y = f(X)$

Rappel des Hypothèses

Observations :

- Variable empirique cible $\tilde{Y}$
- Variables empiriques explicatives $\tilde{X}$

Hypothèses :

- $\tilde{X}$ est un ensemble d'observations lié à un processus aléatoire X
- $\tilde{Y}$ est un ensemble d'observations lié à un processus aléatoire Y
- il existe une relation $Y = f(X)$

Objectifs :

- Produire une fonction $\tilde{f}$ à partir de $\tilde{X}$ et $\tilde{Y}$
- Telle que $\tilde{f}$ soit une approximation fiable de f
- On pourra ainsi prédire $\tilde{Y}' = \tilde{f}(\tilde{X}')$ sur un nouvel échantillon

Erreur du modèle

La bonne mesure serait de minimiser :

$$D(\tilde{f}) = \mathbb{E}(d(\tilde{f}(x), y))$$

Mais comme on ne connaît pas la loi de (X, Y) c'est impossible.

Erreur moyenne empirique

On dispose d'un échantillon de test $\tau = (X_j, Y_j)_{j \leq n}$.
Minimiser :

$$\tilde{D}(\tilde{f}, \tau) = \frac{1}{n} \sum_{j \leq m} d(\tilde{f}(x_j), y_j)$$

Erreur moyenne empirique

On dispose d'un échantillon de test $\tau = (X_j, Y_j)_{j \leq n}$.
Minimiser :

$$\tilde{D}(\tilde{f}, \tau) = \frac{1}{n} \sum_{j \leq m} d(\tilde{f}(x_j), y_j)$$

Ne pas confondre cette somme sur les données avec la somme sur les variables !
Ne pas confondre cette moyenne empirique avec la moyenne

Convergence

D’après la loi des grands nombres, si les observations de test sont indépendantes, la moyenne empirique converge vers l’erreur du modèle.

Pertinence du test

On cherche à évaluer la probabilité que l'écart entre les deux mesures soit faible.

$$P\left(\tilde{D}(\tilde{f},\tau) - D(\tilde{f}) > \epsilon\right) < 1 - \rho$$

Test de Student monovarié

On dispose d'une série d'observations $\tilde{X}$.

Test de Student monovarié

On dispose d'une série d'observations $\tilde{X}$.

On suppose que $\tilde{X}$ procède d'une loi inconnue $X \sim \mathcal{N}(\mu, \sigma^2)$.

Test de Student monovarié

On dispose d'une série d'observations $\tilde{X}$.

On suppose que $\tilde{X}$ procède d'une loi inconnue $X \sim \mathcal{N}(\mu, \sigma^2)$.

On veut évaluer l'espérance de cette loi μ.

Test de Student monovarié

On calcule la moyenne empirique sur un échantillon de taille n :

$$\bar{x} = \frac{1}{n} \sum_{x_i \in \tilde{X}} x_i.$$

Test de Student monovarié

On calcule la moyenne empirique sur un échantillon de taille n :

$$\bar{x} = \frac{1}{n} \sum_{x_i \in \tilde{X}} x_i.$$

Et l'écart-type empirique :

$$s = \sqrt{\frac{1}{n-1} \sum_{x_i \in \tilde{X}} (x_i - \bar{x})^2}$$

Test de Student monovarié

Hypothèse nulle : $\mu = \mu_0$ pour une certaine constante μ_0.

Test de Student monovarié

Hypothèse nulle : $\mu = \mu_0$ pour une certaine constante μ_0.

Conséquence : $\bar{x} \sim \mathcal{N}(\mu_0, \sigma/\sqrt{n})$.

Test de Student monovarié

Hypothèse nulle : $\mu = \mu_0$ pour une certaine constante μ_0.

Conséquence : $\bar{x} \sim \mathcal{N}(\mu_0, \sigma/\sqrt{n})$.

Donc la variable

$$z = \sqrt{n}\frac{\bar{x} - \mu_0}{s}$$

suit une loi de Student à $n - 1$ DL.

Test de Student monovarié

Méthode : on compare les observations de z avec les valeurs attendues d'une loi de Student à $n - 1$ DL.

Test de Student monovarié

Méthode : on compare les observations de z avec les valeurs attendues d'une loi de Student à $n - 1$ DL.

Si l'écart est trop important, on rejette l'hypothèse faite.

Test de Student monovarié

Exercice

Prélevez un échantillon de 30 valeurs se longueur de pétales sur les données iris. Calculez leur moyenne m, et testez l'hypothèse que les longueurs de pétales sont distribuées selon une loi normale de moyenne m.

Exercice

Générez un ensemble de 1000 valeurs suivant une loi normale centrée réduite. Testez l'hypothèse que ses valeurs sont distribuées selon une loi normale de moyenne 0.

Test de Student monovarié

```
import numpy as np
from scipy.stats import ttest_1samp
from sklearn.datasets import load_iris
## exercice 1
petals = load_iris().data[:,0]
m = np.mean(petals[:30])
print(m,ttest_1samp(petals,m))
## exercice 2
a = np.random.normal(size=1000)
print(ttest_1samp(a,0))
```

Test du χ^2

On dispose de deux séries d'observations $\tilde{X}$, $\tilde{Y}$.

Test du χ^2

On dispose de deux séries d'observations $\tilde{X}, \tilde{Y}$.

On suppose que $\tilde{X}, \tilde{Y}$ procèdent de lois inconnues X, Y.

Test du χ^2

On dispose de deux séries d'observations $\tilde{X}, \tilde{Y}$.

On suppose que $\tilde{X}, \tilde{Y}$ procèdent de lois inconnues X, Y.

On veut déterminer si ces lois sont indépendantes ou corrélées.

Test du χ^2

On calcule les effectifs croisés espérés :

$$E_{i,j} = \frac{1}{n} \sum \sharp\{X = i\}\sharp\{Y = j\}$$

Test du χ^2

On calcule les effectifs croisés espérés :

$$E_{i,j} = \frac{1}{n} \sum \sharp\{X = i\}\sharp\{Y = j\}$$

Les effectifs croisés observés :

$$O_{i,j} = \sharp\{X = i \& Y = j\}$$

Test du χ^2

On calcule les effectifs croisés espérés :

$$E_{i,j} = \frac{1}{n} \sum \sharp\{X = i\}\sharp\{Y = j\}$$

Les effectifs croisés observés :

$$O_{i,j} = \sharp\{X = i \& Y = j\}$$

Et l'écart relatif entre les deux :

$$T = \sum_{i,j} \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}}$$

Test du χ^2

Hypothèse nulle : X, Y sont indépendantes.

Test du χ^2

Hypothèse nulle : X, Y sont indépendantes.

Conséquence : T suit une loi du χ^2 à $(I-1, J-1)$ DL.

Test du *chi*^{2}

Méthode : on compare les observations de T avec les valeurs attendues d'une loi du χ^2 à $(I-1, J-1)$ DL.

Test du *chi*^{2}

Méthode : on compare les observations de T avec les valeurs attendues d'une loi du χ^2 à $(I-1, J-1)$ DL.

Si l'écart est trop important, on rejette l'hypothèse faite.

Test du χ^2

Exercice

Construisez une table de contingence entre le genre et la classe sur les données du titanic. Testez l'hypothèse que la classe et le genre sont indépendants.

Exercice

Générez deux variables aléatoires indépendantes à deux modalités chacune. Construisez leur table de contingence et testez leur indépendance.

Test du χ^2

```
import pandas as pd
from scipy.stats import chi2_contingency
from random import randint
## exercice 1
titanic = pd.read_csv('./C1/data1.csv')
cont = [[titanic.loc[(titanic.sex==j) & (titanic.pclass==i)].shape[0]
for i in range(1,4)] for j in ['male','female']]
print(cont, chi2_contingency(cont))
## exercice 2
data = [(randint(0,1),randint(0,1)) for i in range(100)]
cont = [[len([x for x in data if x[0]==i and x[1] ==j])
for i in range(2)] for j in range(2)]
print(cont, chi2_contingency(cont))
```

Biais-Variance

2 parmi les grandes sources d'erreurs dans l'apprentissage :

Biais-Variance

2 parmi les grandes sources d'erreurs dans l'apprentissage :

Le biais correspond au sous-apprentissage (modèle trop simple, trop peu de données)

Biais-Variance

2 parmi les grandes sources d'erreurs dans l'apprentissage :

Le biais correspond au sous-apprentissage (modèle trop simple, trop peu de données)

La variance correspond au surapprentissage (modèle apprend aussi le bruit)

Biais-Variance

Supposons une distribution bruitée $Y = f(X) + \epsilon$ avec ϵ de moyenne 0 et de variance σ^2.

Biais-Variance

Supposons une distribution bruitée $Y = f(X) + \epsilon$ avec ϵ de moyenne 0 et de variance σ^2.

$$E[(Y - \tilde{f}(X))^2] = E[(f(X) - \tilde{f}(X))]^2 + E[(\tilde{f}(X) - E[\tilde{f}(X)])^2] + \sigma^2$$

Biais-Variance

Exercice

Générez un vecteur X quelconque et un vecteur Y=3X+e où e est un bruit gaussien centré de variance s. Testez les variations du score d'une régression linéaire entre X et Y en fonction de s.

Biais-Variance

```
from sklearn.linear_model import LinearRegression
from numpy.random import normal,randint
X = randint(0,100,100)
Y1 = 3*X + normal(0,10,100)
Y2 = 3*X + normal(0,50,100)
l = LinearRegression()
for s in [1,50,200]:
    for n in [40,60,90]:
        l = LinearRegression()
        Y = 3*X + normal(0,s,100)
        l.fit(X[:n].reshape(-1,1),Y[:n])
        print(s,n,l.coef_,l.score(X[n:].reshape(-1,1),Y[
```

Cross Validation

On veut diviser efficacement nos données entre échantillons d'apprentissage et de test.

La taille et la nature de ces divisions peuvent varier selon le classifieur employés, et scikit-learn en fournit en conséquence.

Paramétrisation

La cross validation peut être utilisée pour optimiser les paramètres d'un algorithme, soit entre eux, soit contre la complexité (ex. degré d'un kernel polynomial)

Exercice

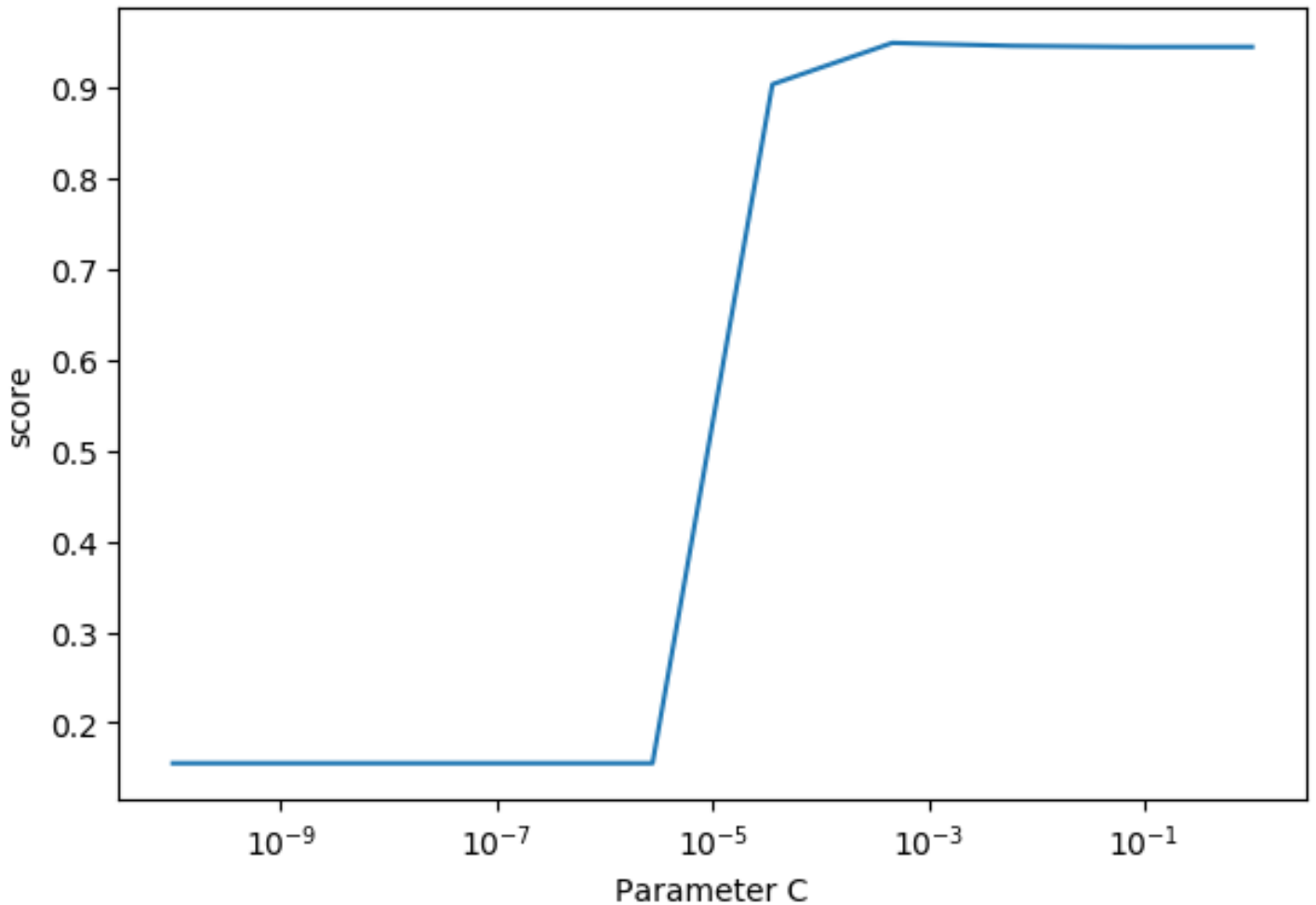
Exercice

Calculez et affichez le cross-validation score d'un svm linéaire sur le jeu de données `digits` en fonction de différentes valeurs du paramètre de régularisation C.

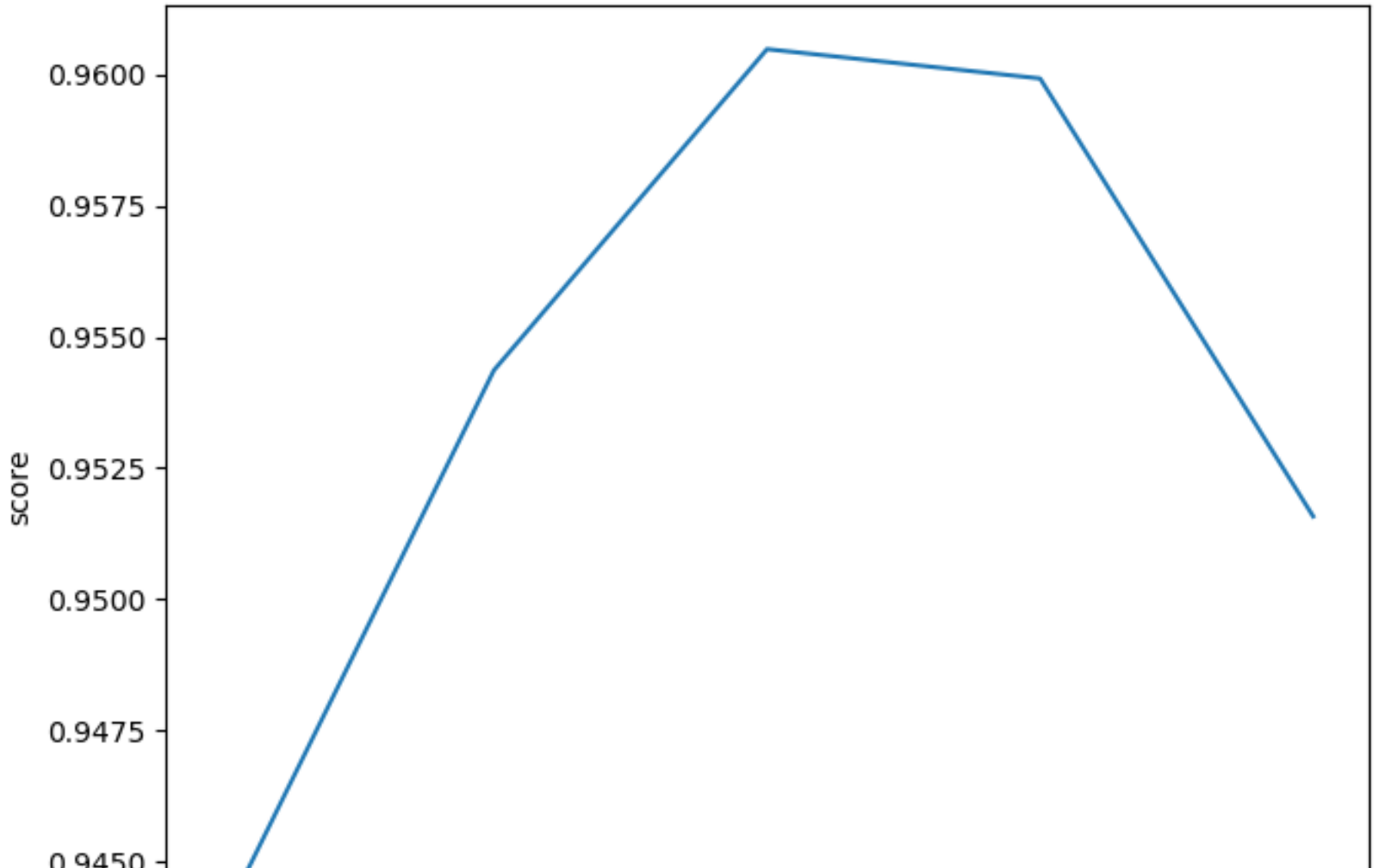
Exercice

Calculez et affichez le cross-validation score d'un svm polynomial sur le jeu de données `digits` en fonction de différentes valeurs du degré du polynome.

Résultat attendu (1)



Résultat attendu (2)



solution

```
import numpy as np
from sklearn.model_selection import cross_val_score
from sklearn import datasets, svm
import matplotlib.pyplot as plt
digits = datasets.load_digits()
X,Y = digits.data, digits.target
svc = svm.SVC(kernel='linear')
C_s = np.logspace(-10,0,10)
scores = []
for C in C_s:
    svc.C = C
    this_scores = cross_val_score(svc, X, Y)
    scores.append(np.mean(this_scores))
plt.semilogx(C_s, scores)
plt.ylabel('score')
plt.xlabel('Parameter C')
plt.show()
```

solution

```
import numpy as np
from sklearn.model_selection import cross_val_score
from sklearn import datasets, svm
import matplotlib.pyplot as plt
digits = datasets.load_digits()
X,Y = digits.data, digits.target
Delta = np.arange(1,6)
print(Delta)
scores = []
for d in Delta:
    svc = svm.SVC(kernel='poly',degree=d)
    this_scores = cross_val_score(svc, X, Y)
    scores.append(np.mean(this_scores))
plt.plot(Delta, scores)
plt.ylabel('score')
plt.xlabel('degre')
plt.show()
```