



SVM

Support Vector Machines



Ce ne așteaptă?

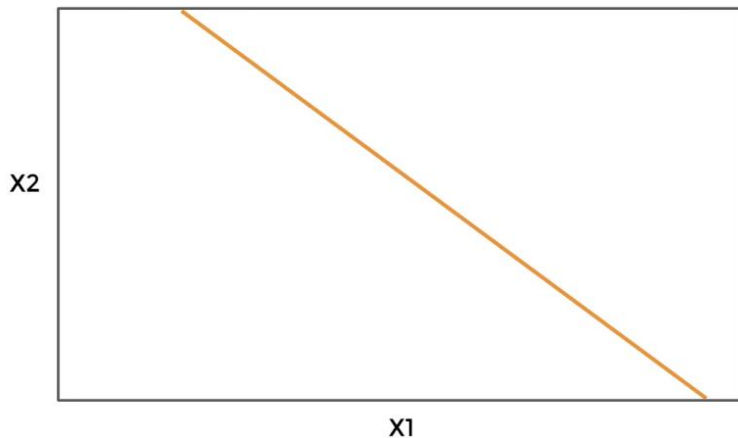
1. Hiperplanul
2. Maximal Margin Classifier
3. Support Vector Classifier
4. Support Vector Machines
5. Instrumente Scikit-Learn

1. Hiperplanul

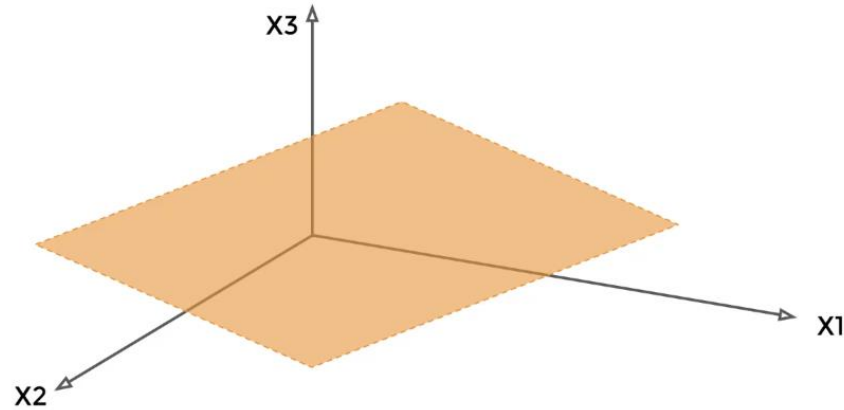
- Hiperplanul – un subspațiu plan de dimensiunea $N-1$ într-un spațiu de dimensiunea N
- Hiperplanul 1D este un punct pe axa unidimensională



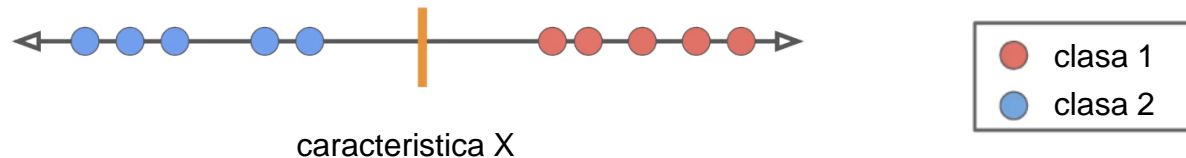
- Hiperplanul 2D este o line într-un plan bidirecțional



- **Hiperplanul 3D este un plan bidimensional într-un spațiu tridimensional**



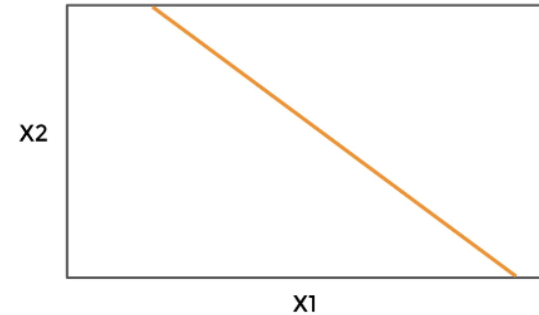
- **În SVM hiperplanul este utilizat pentru crearea unei separări între valorile claselor**



■ Definiția matematică a hiperplanului

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 = 0 \quad - \text{ hiperplan 2-D}$$

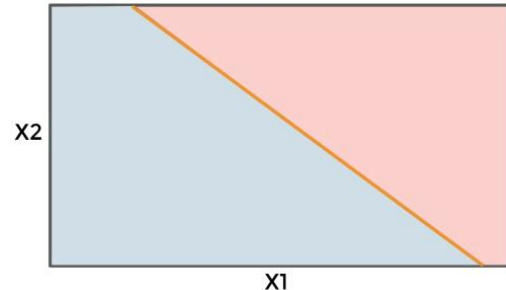
$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p = 0 \quad - \text{ hiperplan p-D}$$



■ Condițiile de separare

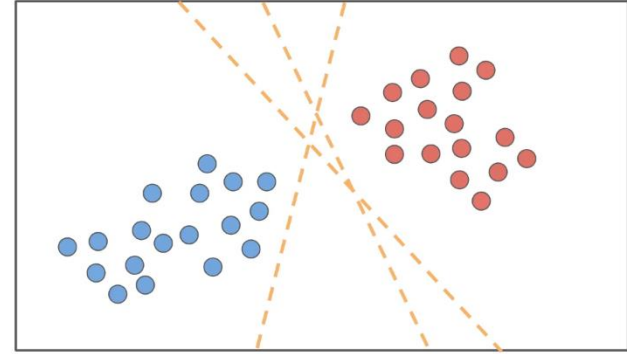
$$\beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p > 0 \quad - \text{ condiția clasei 1}$$

$$\beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p < 0 \quad - \text{ condiția clasei 2}$$

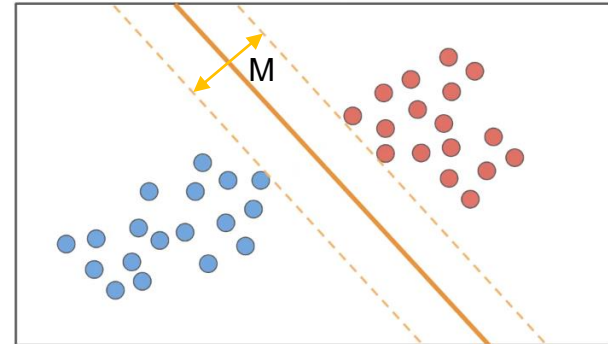
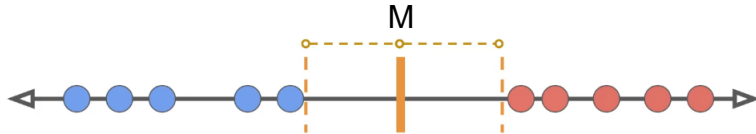


2. Maximal Margin Classifier

- Posibilități multiple de a poziționa hiperplanul



- Maximal Margin Classifier permite selecția prin maximizarea marginilor dintre clase

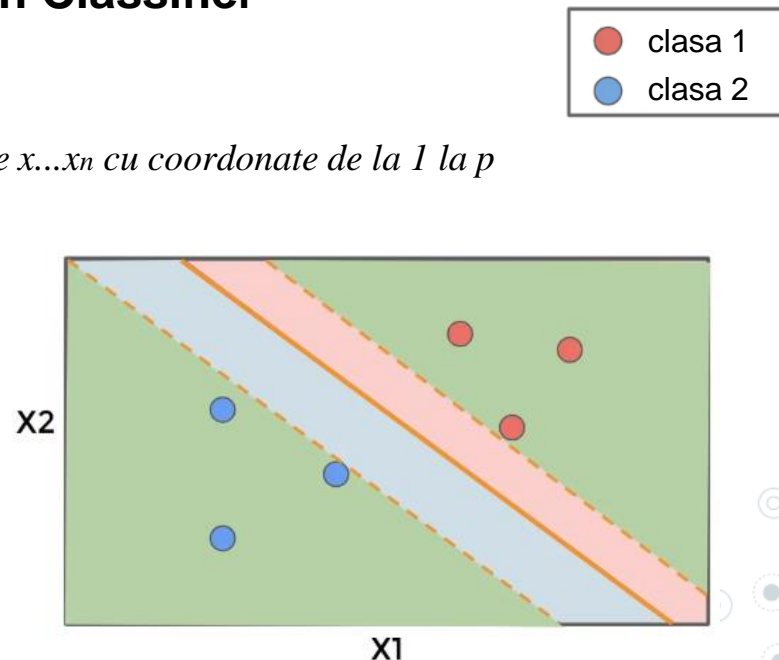


■ Aparatul matematic al Maximal Margin Classifier

$$x_1 = \begin{pmatrix} x_{11} \\ \vdots \\ x_{1p} \end{pmatrix}, \dots, x_n = \begin{pmatrix} x_{n1} \\ \vdots \\ x_{np} \end{pmatrix} \quad - \text{punctele } x \dots x_n \text{ cu coordonate de la 1 la } p$$

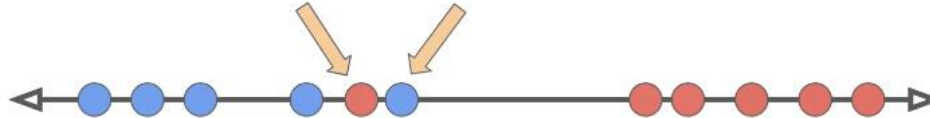
$$\underset{\beta_0, \beta_1, \dots, \beta_p, M}{\text{maximize}} \quad M \quad - \text{maximizarea valorii lui } M$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}) \geq M \quad - \text{condiția ca toate punctele să fie în exteriorul spațiului } M$$

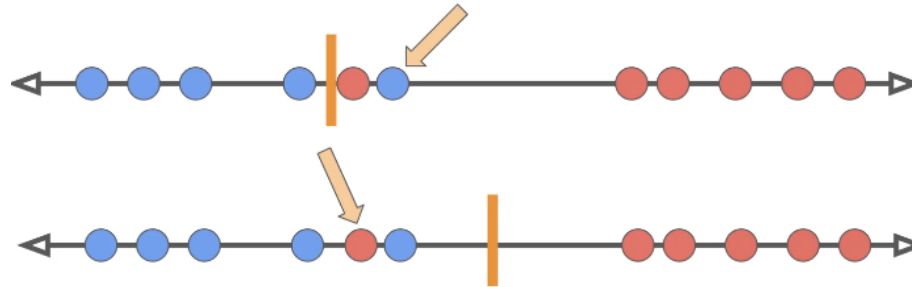


3. Support Vector Classifier

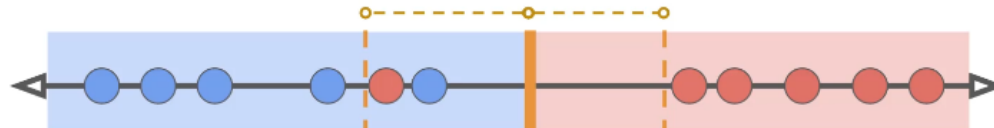
- **Cazul claselor cu separare imperfectă**



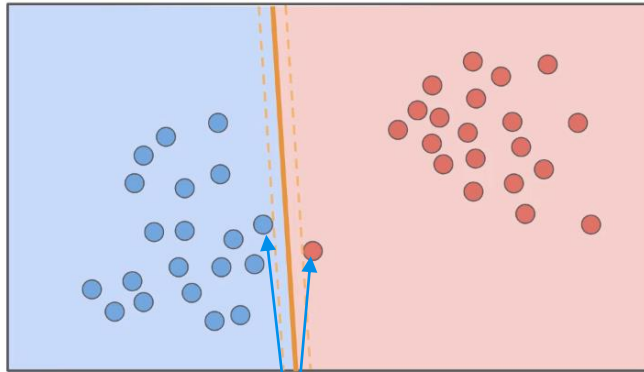
- **Poziționarea hiperplanului în cazul claselor cu separare imperfectă**



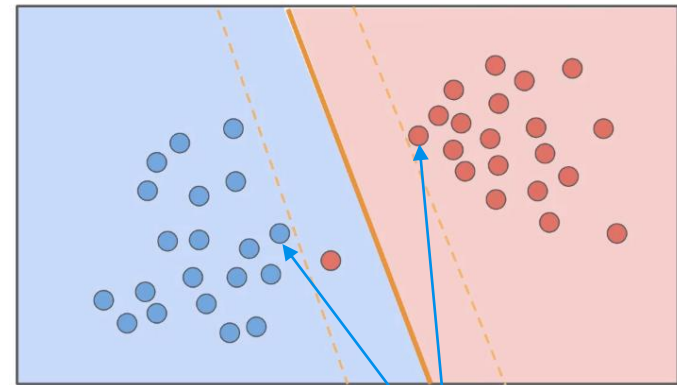
- **Support Vector Classifier (Soft Margins) permite poziționarea hiperplanului cu introducerea de erori**



■ Maximal Margin Classifier vs Support Vector Classifier



support vectors



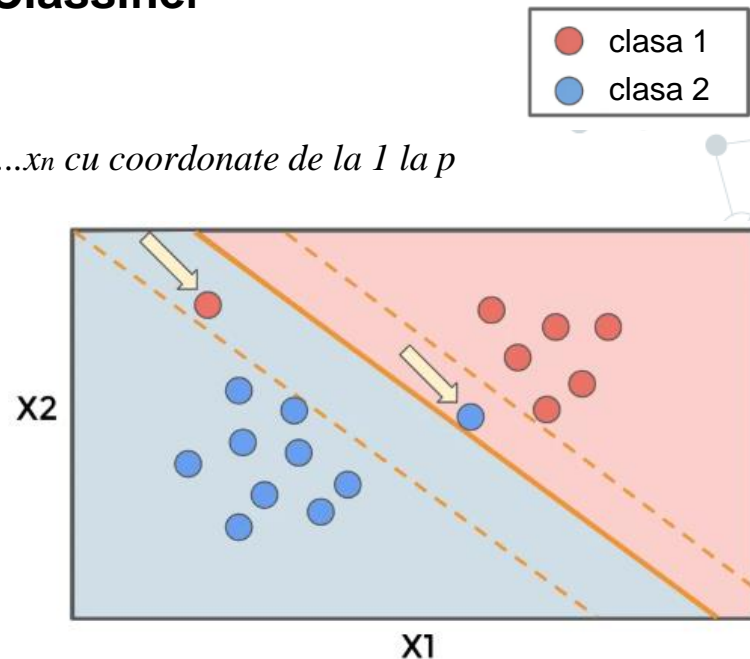
support vectors

■ Aparatul matematic al Support Vector Classifier

$$x_1 = \begin{pmatrix} x_{11} \\ \vdots \\ x_{1p} \end{pmatrix}, \dots, x_n = \begin{pmatrix} x_{n1} \\ \vdots \\ x_{np} \end{pmatrix} \quad - \text{ punctele } x \dots x_n \text{ cu coordonate de la 1 la } p$$

$$\text{maximize}_{\beta_0, \beta_1, \dots, \beta_p, M} M \quad - \text{ maximizarea valorii lui } M$$

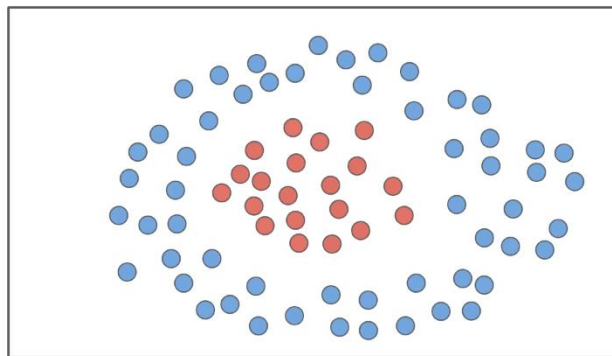
$$\epsilon_i \geq 0, \quad \sum_{i=1}^n \epsilon_i \leq C \quad - \text{ introducerea unei erori ce poate lua valoarea maxima } C$$



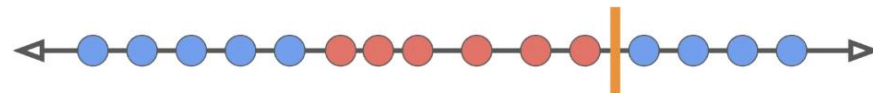
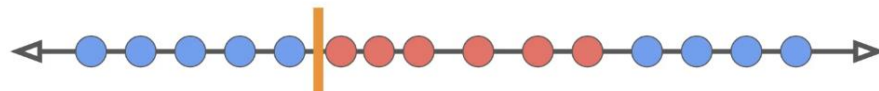
$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}) \geq M(1 - \epsilon_i) \quad - \text{ condiția poziționării punctelor față de spațiul } M$$

4. Support Vector Machines

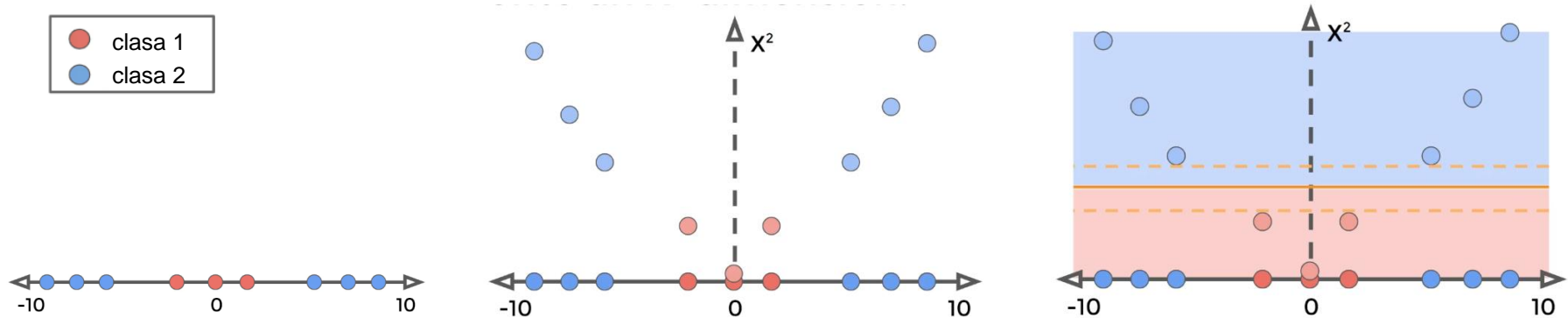
■ Cazul claselor cu separare imposibilă



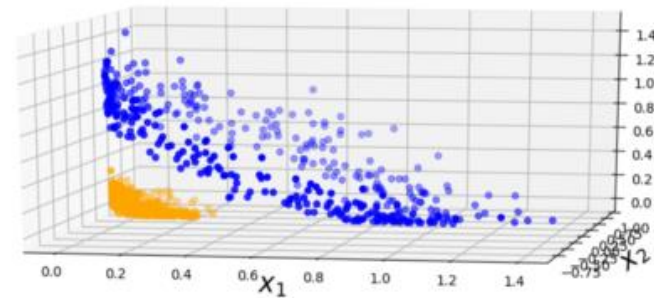
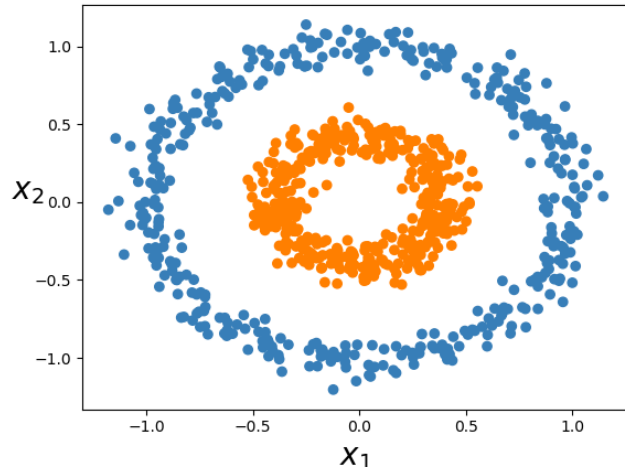
■ Poziționarea hiperplanului în cazul claselor cu separare imposibilă



- Nucleul (kernel) reprezintă o funcție ce permite proiectarea datelor într-un spațiu cu o dimensiune superioară pentru a putea delimita clasele
- Nucleul funcție polinomială x^2 de proiectare a datelor din 1D în 2D



- Nucleul ce permite proiectarea datelor din 2D în 3D



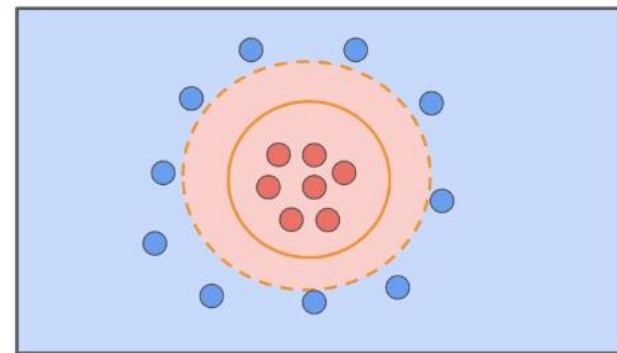
■ Aparatul matematic al Support Vector Machines

$X_1, X_2, \dots, X_p, \rightarrow X_1, X_1^2, X_2, X_2^2, \dots, X_p, X_p^2$ - crearea caracteristicilor polinomiale

$\underset{\beta_0, \beta_{11}, \beta_{12}, \dots, \beta_{p1}, \beta_{p2}, \epsilon_1, \dots, \epsilon_n, M}{\text{maximize}} M$ - maximizarea valorii lui M

$\sum_{i=1}^n \epsilon_i \leq C, \epsilon_i \geq 0,$ - introducerea unei erori ce poate lua valoarea maxima C

$y_i \left(\beta_0 + \sum_{j=1}^p \beta_{j1} x_{ij} + \sum_{j=1}^p \beta_{j2} x_{ij}^2 \right) \geq M(1 - \epsilon_i)$ - condiția poziționării punctelor față de spațiul M



5. Instrumente Scikit-Learn

■ Modelul Support Vector Machines pentru sarcini de clasificare

○ Se importa clasa algoritmului SVC

```
from sklearn.svm import SVC
```

○ Se creează un model cu fixarea valorilor hiper-parametrilor:

- C – coeficientul invers de includere a erorii (penalizării) (valori posibile = float, implicit =1.0)
- kernel - funcția de proiectare a datelor într-o dimensiune mai mare (valori posibile = {'linear', 'poly', 'rbf', 'sigmoid', 'precomputed'}, implicit = 'rbf')
- gamma – coeficientul kernel pentru funcțiile 'poly', 'rbf' și 'sigmoid' (valori posibile = {'scale', 'auto'} sau float, implicit='scale')
- degree – gradul polinomului funcției 'poly' (valori posibile = int, implicit =3)

```
model = SVC(C=1, kernel = 'poly', gamma = 'scale', degree=3)
```

- Se realizează trainingul modelului pe datele de training
`model.fit(X_train, y_train)`
- Se vizualizează coeficienții modelului
`model.coef_`
- Se realizează predicția pe datele de test
`y_pred = model.predict(X_test)`
- Se realizează predicția pe datele de test cu afișarea probabilităților
`y_pred = model.predict_proba(X_test)`

■ Modelul Support Vector Machines pentru sarcini de regresie

- Se importa clasa algoritmului SVR

```
from sklearn.svm import SVR
```

- Se creează un model cu fixarea valorilor hiper-parametrilor:

- C – coeficientul invers de includere a erorii (penalizării) (valori posibile = float, implicit =1.0)
- kernel - funcția de proiectare a datelor într-o dimensiune mai mare (valori posibile = {'linear', 'poly', 'rbf', 'sigmoid', 'precomputed'}, implicit = 'rbf')
- gamma – coeficientul kernel pentru funcțiile 'poly', 'rbf' și 'sigmoid' (valori posibile = {'scale', 'auto'} sau float, implicit='scale')
- degree – gradul polinomului funcției 'poly' (valori posibile = int, implicit =3)
- epsilon - (valori posibile = float, implicit =0.01)

```
model = SVR(C=1, kernel = 'poly', gamma = 'scale', degree=3,  
epsilon=0.1)
```