BIG DATA

CURS 10

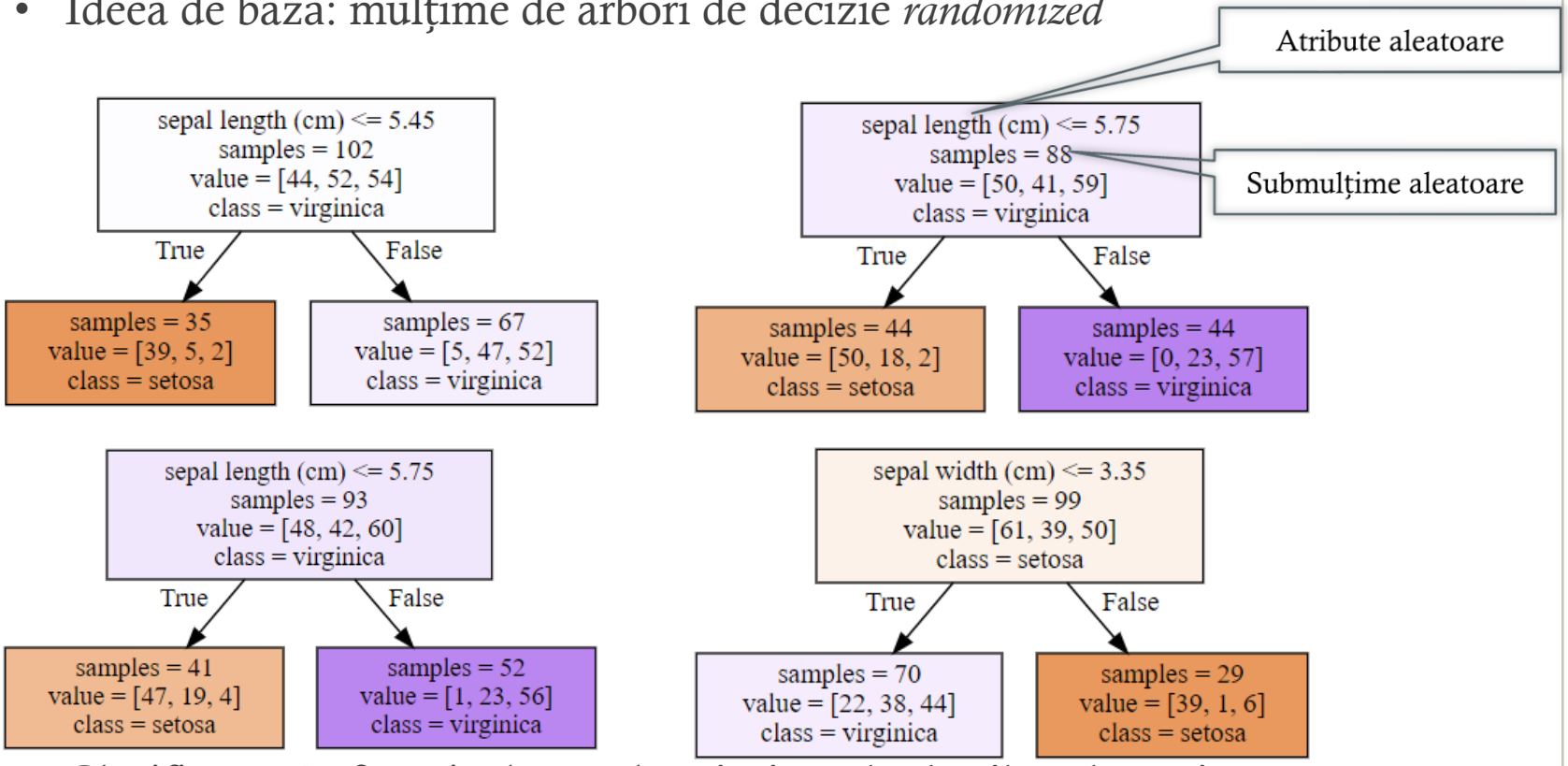
Concepte și metode fundamentale ale analizei datelor

1. Etapele unui proiect Data Science
2. Explorarea datelor
3. Analiza datelor
4. Descoperirea regulilor de asociere
5. Clustering
6. **Clasificare**
7. **Regresie**

6. Clasificare (continuare)

Random Forest

- Ideea de bază: mulțime de arbori de decizie *randomized*



- Clasificarea în funcție de votul majoritar al arborilor aleatori

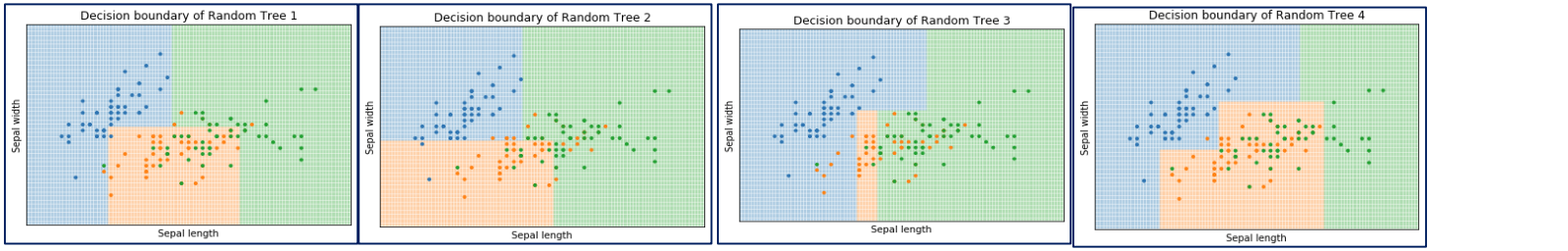
Agregarea bootstrap (Bagging)

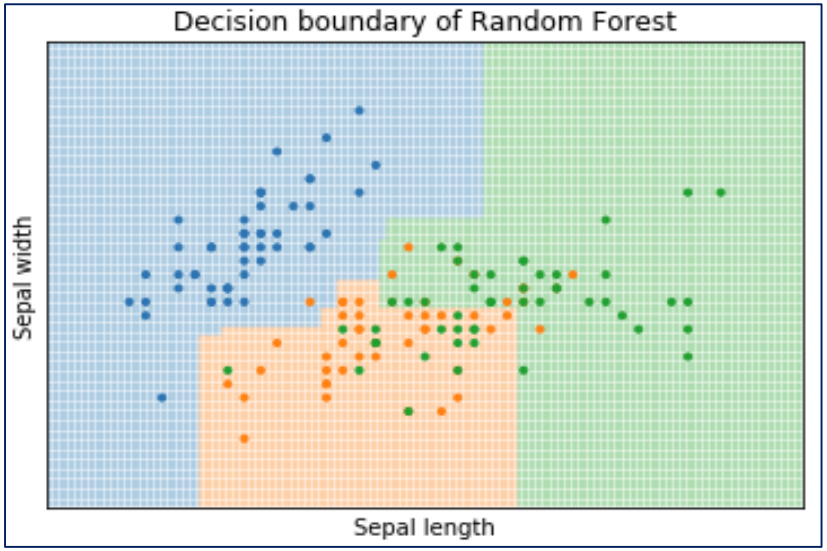
- Bagging – prescurtarea lui Bootstrap Aggregating

- Extrage submulțimi aleatoare (sub-eșantioane) de date de antrenament
- Construiește modelul pentru fiecare submulțime => ansamblu de modele
- Se aplică votarea pentru crearea claselor
  - Poate fi ponderata, adică să utilizeze calitatea ansamblului de modele

- Random Forests combină tehnica Bagging cu:
  - Arborii de decizie scurți (ce au adâncime mică)
  - Permit doar o submulțime aleatoare de caracteristici pentru fiecare decizie.

Suprafața de decizie a *Random Forest*





Regresia logistică

- Ideea de bază:
  - Model de regresie al probabilităţii ca un obiect să aparţină unei clase
  - Combină funcţia *logit* cu regresia liniară
- Regresia liniară
  - y este o combinaţie liniară de $x_1, \ldots, x_n$
  - $y = b_0 + b_1 x_1 + \ \ldots + b_n x_n$
- Funcţia logit
  - $logit\big(P(y = c)\big) = ln \frac{P(y=c)}{1-P(y=c)}$
- Regresia logistică
  - $logit\big(P(y = c)\big) = b_0 + b_1 x_1 + \ \ldots + b_n x_n$

Rapoarte de cote

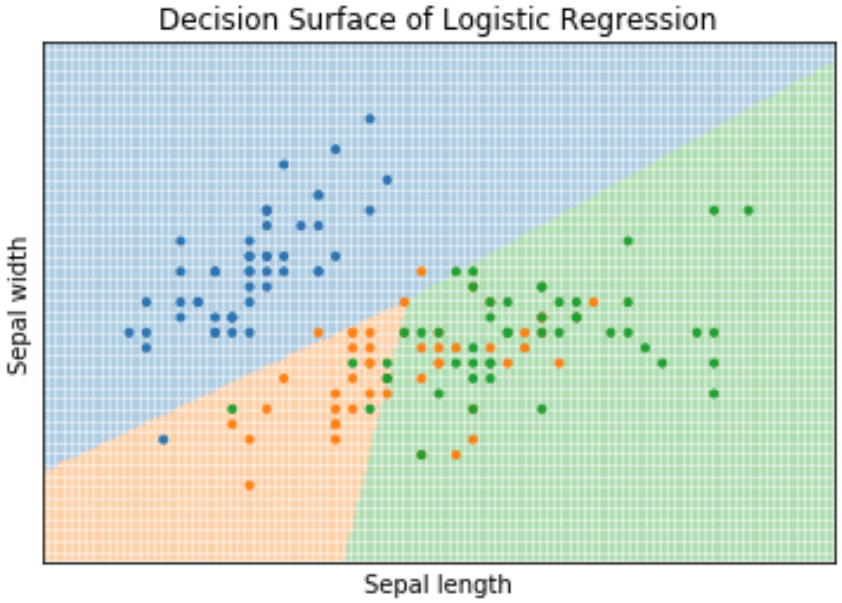
- Probabilități vs cote
  - Probabilitate: *P*(promovare_an) = 0.75
  - Cota de promovare a anului școlar: *odds*(promovare_an) = $\frac{0.75}{1-0.75} = 3$
    - Cota de promovare este de 3 la 1
- Dacă inversăm logaritmul natural, obținem:
  - $\frac{P(y=c)}{1-P(y=c)} = \exp(b_0 + b_1 x_1 + \ \ldots + b_n x_n) = \prod_{j=0}^{n} \exp(b_j x_j)$

Definiția cotei

- Se obține că $\exp(b_j)$ este raportul de cote al caracteristicii *j*
  - Raportul de cote reprezintă modificarea cotei dacă mărim x_j cu 1
  - Dacă raportul este mai mare decât 1 => cota a crescut
  - Dacă raportul este mai mic decât 1 => cota s-a micșorat

Suprafața de decizie a regresiei logistice

- Frontierele de decizie sunt liniare



Naive Bayes

- Ideea de bază:
  - Presupunem că toate caracteristicile sunt independente
  - Obținem scoruri pentru clase folosind probabilitatea condițională

- Teorema lui Bayes
  - $P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$

- Probabilitatea condițională a unei clase:
  - $P(c|x_1, \dots x_n) = \frac{P(x_1, \dots x_n|c)P(c)}{P(x_1, \dots x_n)}$

De la teorema lui Bayes la *Naïve Bayes*

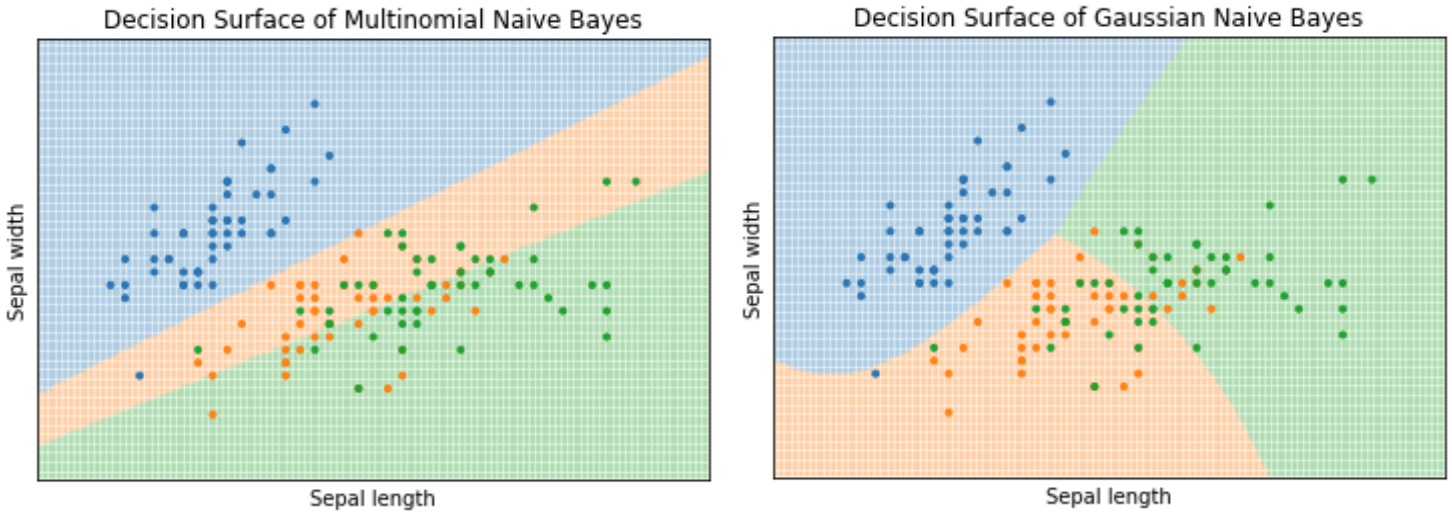
- Probabilitatea conform teoremei lui Bayes:
  - $P(c|x_1, \dots x_n) = \frac{P(x_1, \dots, x_n|c)P(c)}{P(x_1,\dots x_n)}$
- Presupunere „naivă”: $x_1, \dots x_n$ sunt independente condiţional pentru un c dat
  - $P(c|x_1, \dots x_n) = \frac{P(x_1|c)\dots P(x_n|c)P(c)}{P(x_1,\dots,x_n)} = \frac{\prod_{j=1}^{n} P(x_j|c)P(c)}{P(x_1,\dots,x_n)}$
- $P(x_1, \dots, x_n)$ este independentă de c şi nu se modifică
  - $score(c|x_1, \dots, x_n) = \prod_{j=1}^{n} P(x_j|c)P(c)$
- Se atribuie clasa cu cel mai mare scor

Naïve Bayes multinomial și Gaussian

- Variante diferite ale modului în care se estimează $P(x_j|c)$
- Multinomial
  - $P(x_j|c)$ este probabilitatea empirică obținută ca urmare a observării unei caracteristici
  - Are loc un *număr* de observări ale lui x_j în date
- Gaussian
  - Se presupune că valorile caracteristicilor urmează o distribuție Gaussiană (normală)
  - Se estimează probabilitatea condițională $P(x_j|c)$ folosind funcția de densitate gaussiană

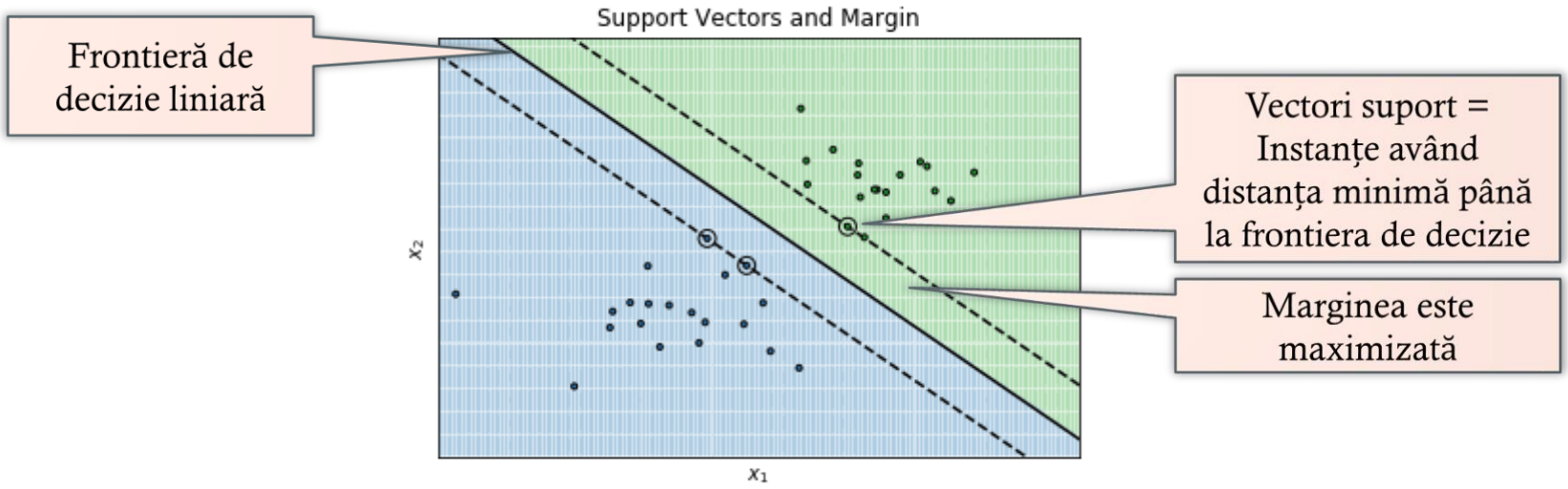
Suprafața de decizie a metodei Naive Bayes

- Varianta multinomială are frontiere de decizie liniare
- Varianta gaussiană are frontiere de decizie cuadratice pe porțiuni



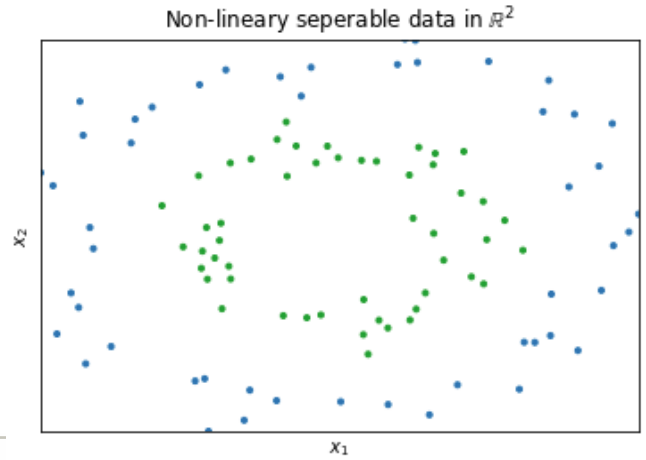
Support Vector Machines (SVM)

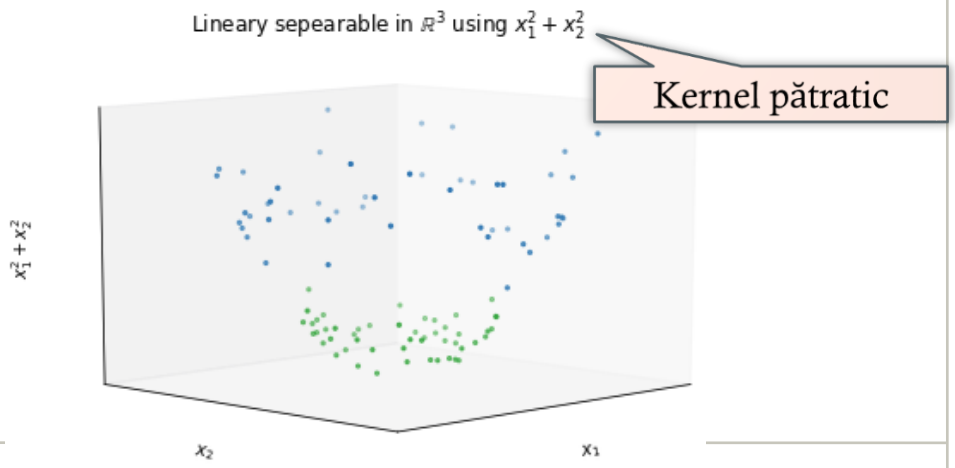
- Ideea de bază:
  - Determinarea frontierei de decizie astfel încât aceasta să se afle „departe” de date



SVM-uri neliniare

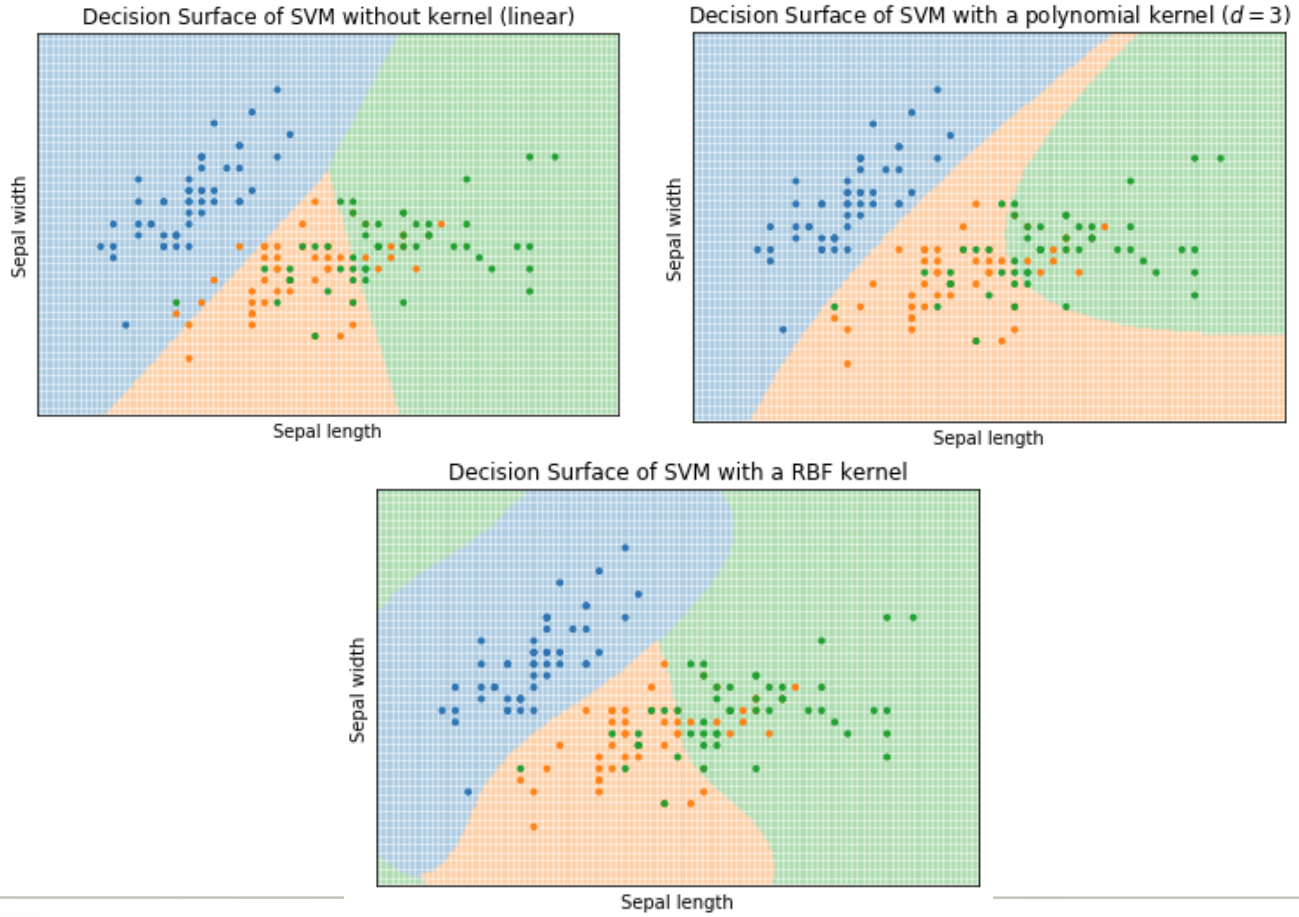
- Caracteristicile pot fi expandate folosind nuclee (*kernel*-uri) pentru separarea datelor neliniare
  - Transformare într-un spațiu nucleu cu număr mare de dimensiuni
    - Poate fi infinit (de exemplu: nucleul gaussian, nucleul RBF – *Radial Basis Function*)
  - Se determină separarea liniară în spațiul nucleu
  - Se aplică *kernel trick* (https://towardsdatascience.com/understanding-the-kernel-trick-e0bc6112ef78) pentru evitarea expansiunii





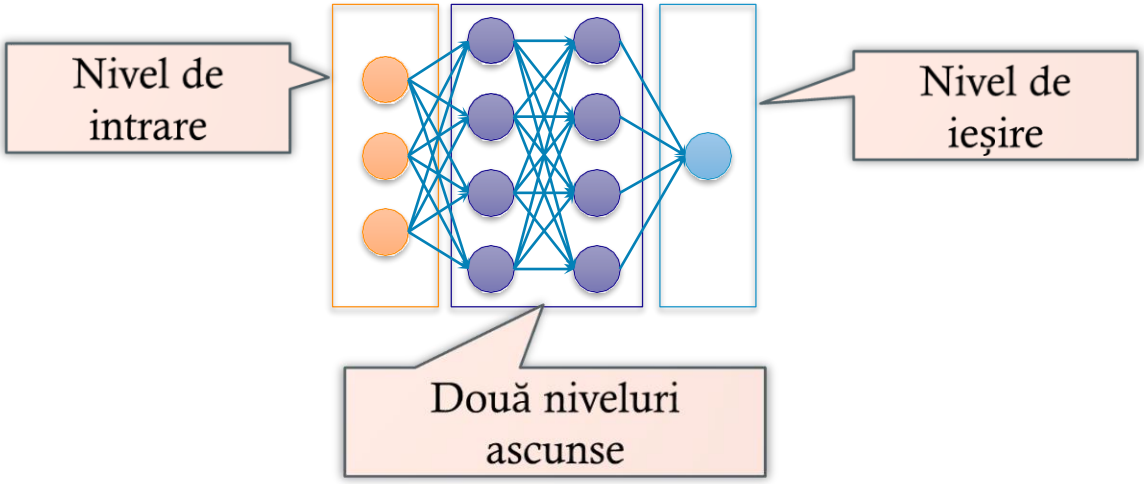
Suprafața de decizie a SVM-urilor

- Forma suprafeței de decizie depinde de nucleu



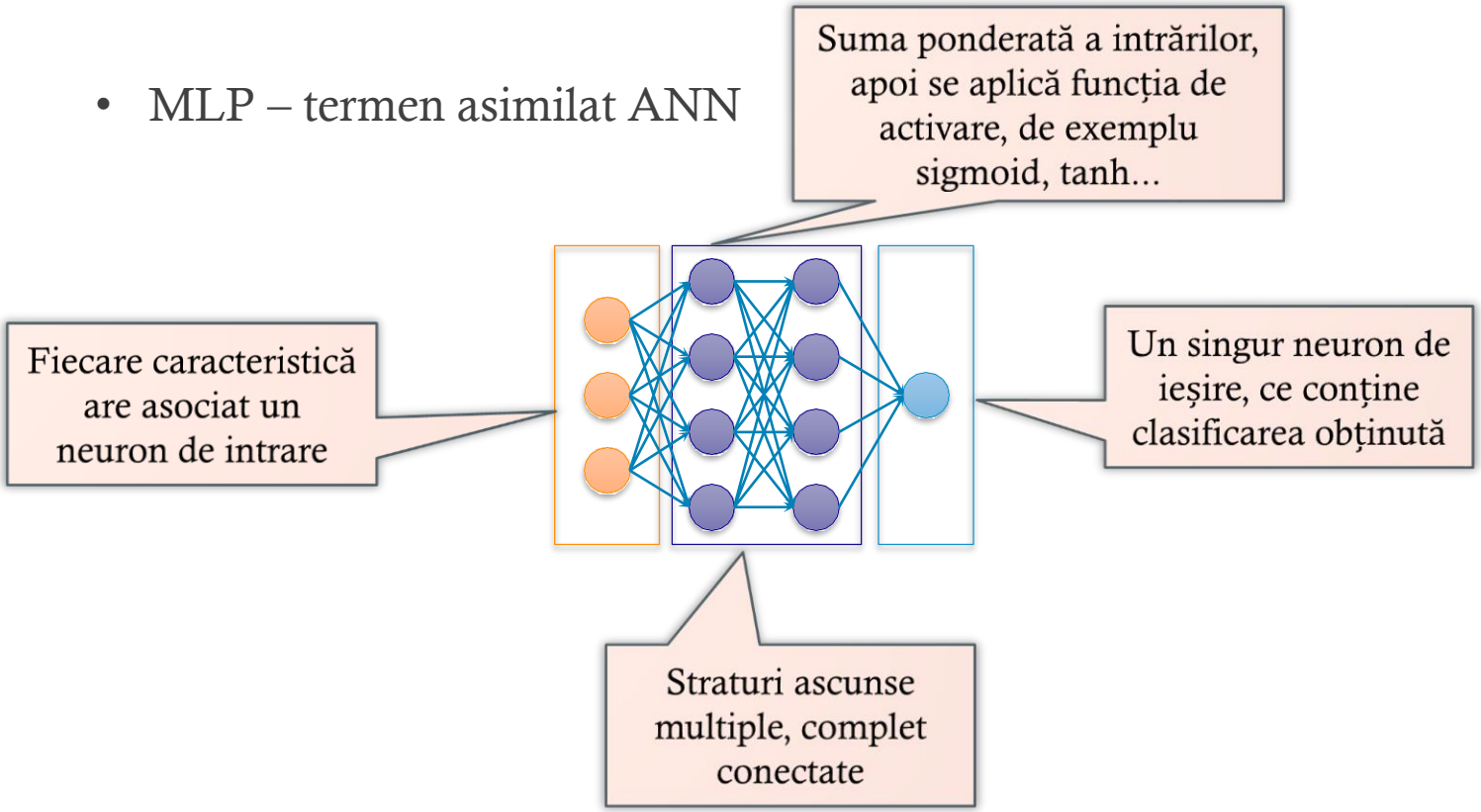
Rețele neuronale

- Ideea de bază:
  - Rețele de neuroni cu diferite straturi și comunicare între neuroni
  - Stratul de intrare aduce datele în rețea (este alcătuit din neuroni ale căror intrări se află în date)
  - Nivelurile ascunse corelează datele
  - Nivelul de ieșire furnizează rezultatul



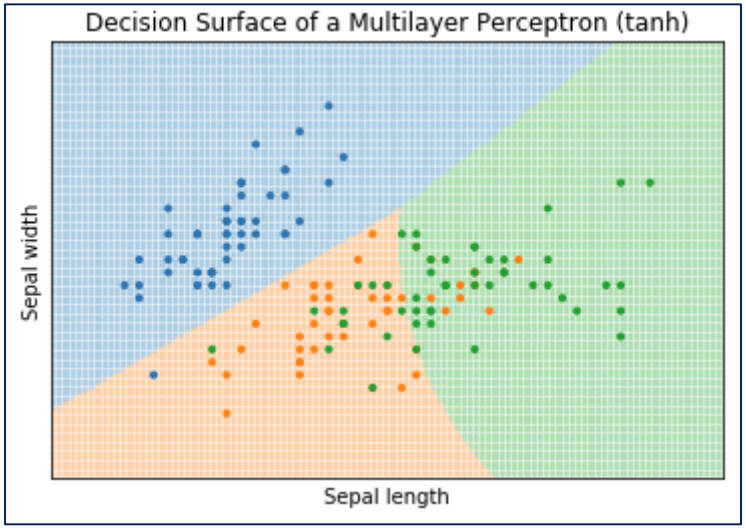
Multilayer Perceptron (MLP)

- MLP – termen asimilat ANN



Suprafața de decizie a MLP

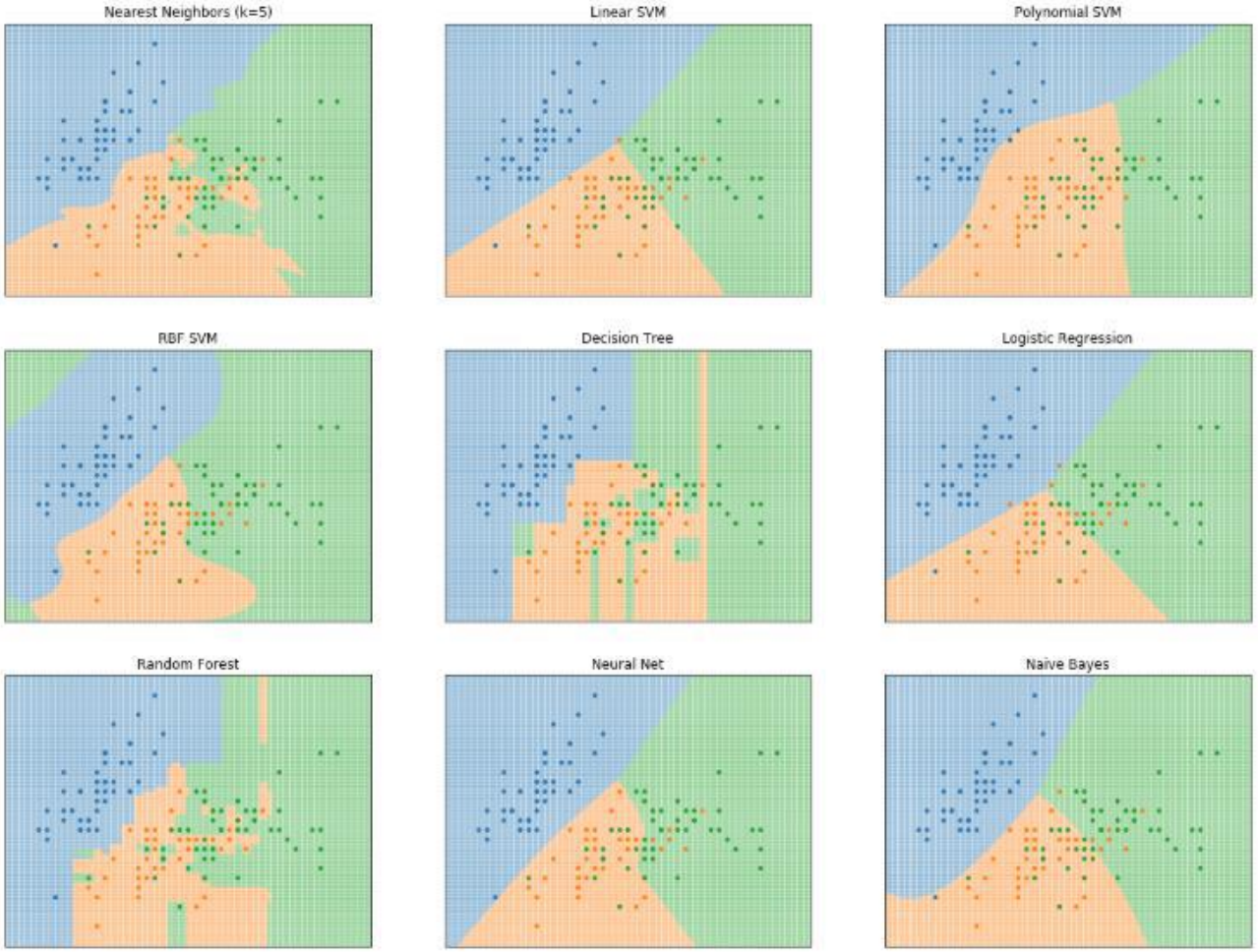
- Forma frontierei de decizie depinde de:
  - Funcția de activare
  - Numărul de niveluri ascunse
  - Numărul de neuroni din nivelurile ascunse



Comparație între modelele de clasificare

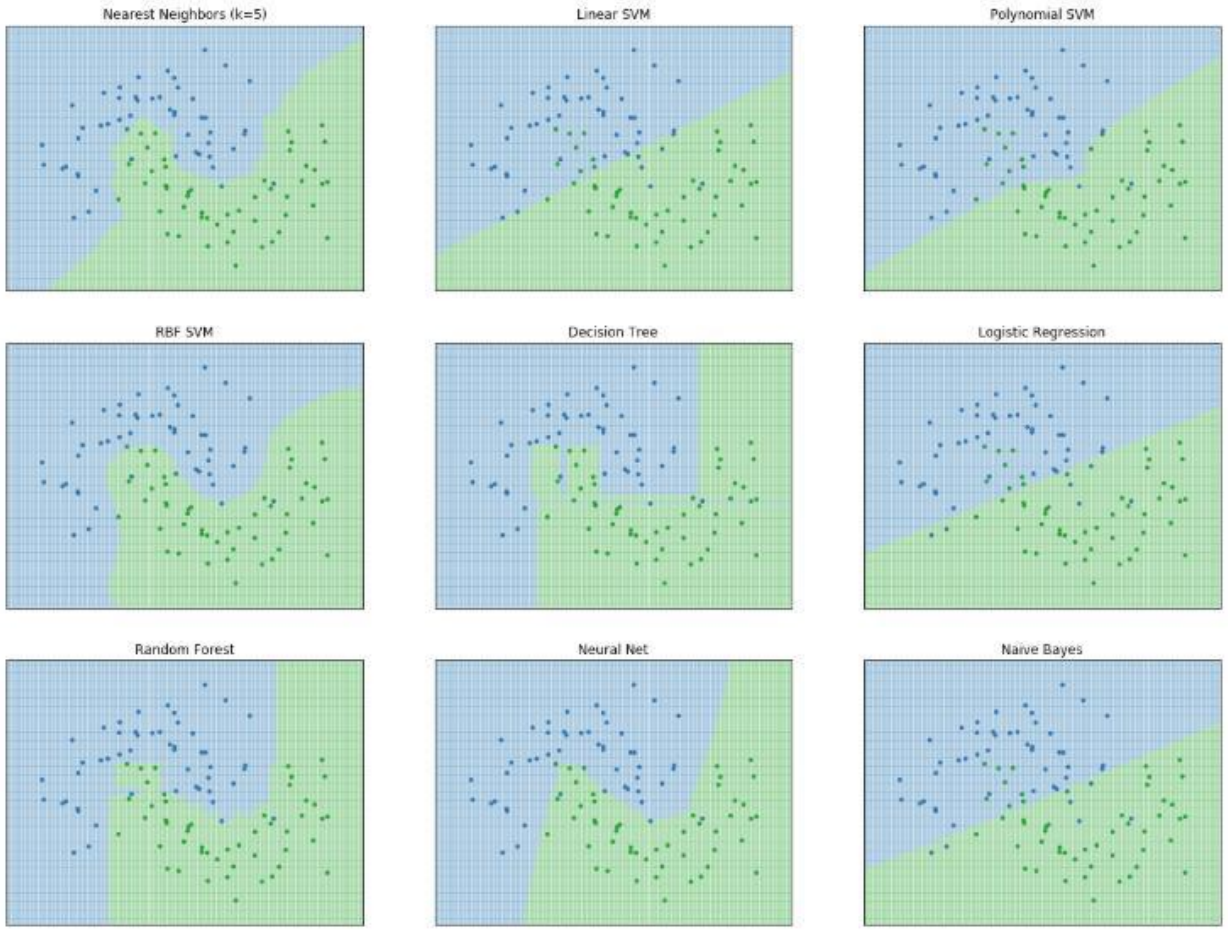
- Există abordări diferite în cadrul clasificatorilor
  - *k-nearest neighbor* – este bazat pe instanță
  - Arbori de decizie – bazați pe reguli + teoria informației
  - Random Forests – ansambluri *randomized*
  - Regresia logistică – regresie
  - Naive Bayes – Probabilitatea condițională
  - Support Vector Machines – Maximizarea marginilor + nuclee
  - Rețele neurale – Regresie foarte complexă

Comparație a suprafețelor de decizie (setul de date Iris)



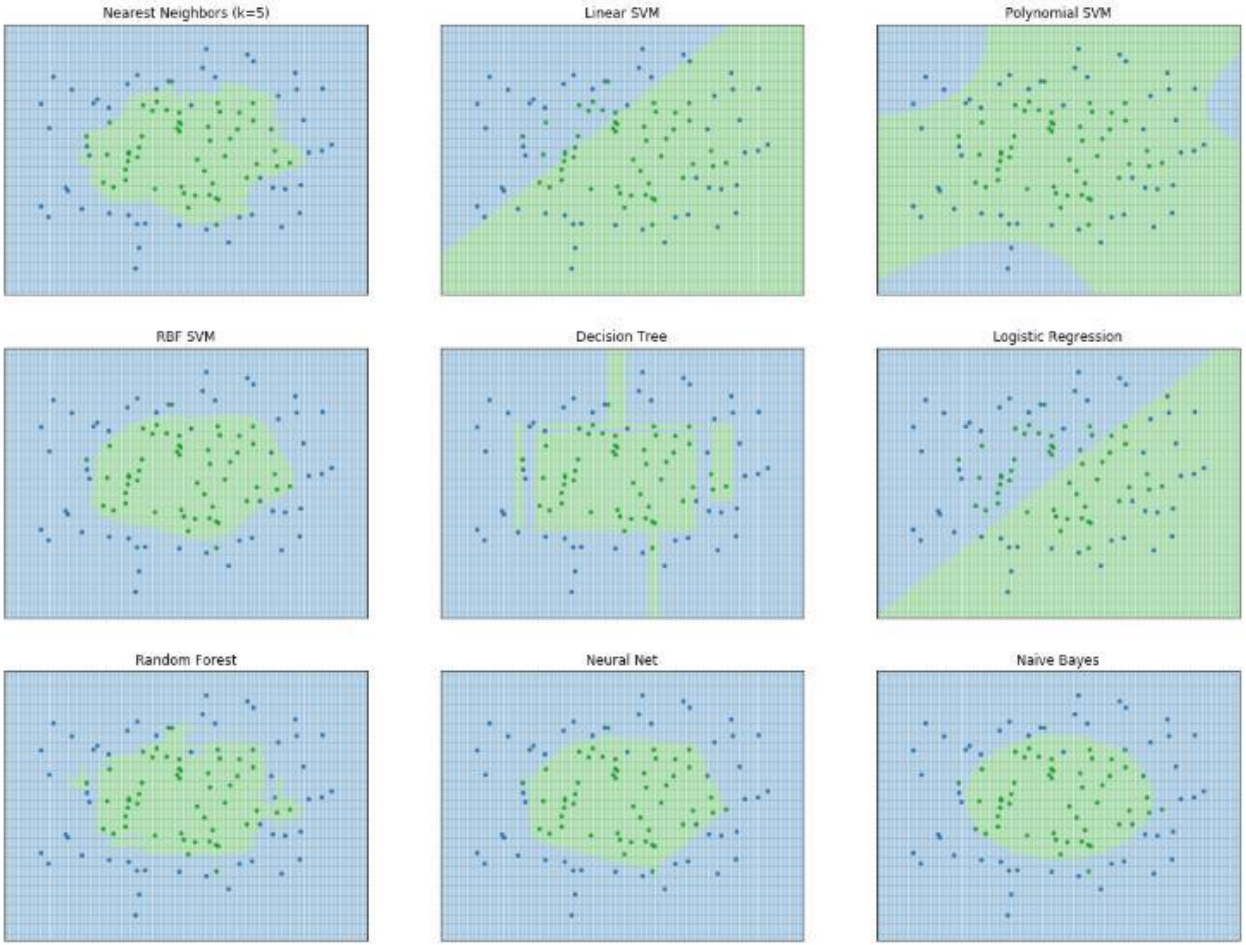
Rezultatele pot fi diferite în funcție de valorile hiperparametrilor.

Comparație a suprafețelor de decizie (cazul separabil non-liniar)



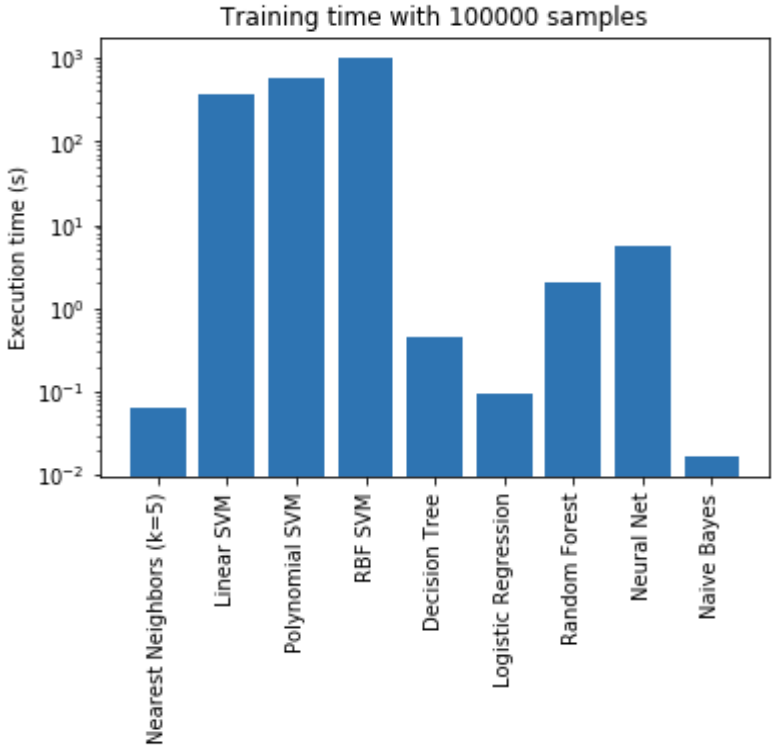
Rezultatele pot fi diferite în funcție de valorile hiperparametrilor.

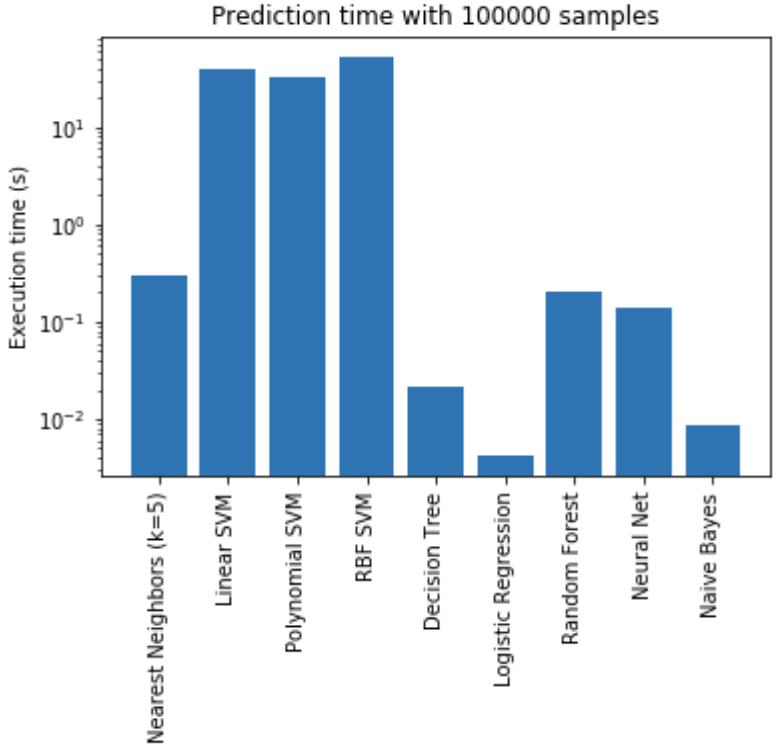
Comparație a suprafețelor de decizie (cazul datelor clasificate în „cercuri”)



Rezultatele pot fi diferite în funcţie de valorile hiperparametrilor.

Comparația timpilor de execuție





Setul de date „half moons”

Avantaje și dezavantaje

	Valoare explicativă	Reprezentare concisă	Scor	Caracteristici categoriale	Caracteristici necunoscute	Caracteristici corelate
k-nearest Neighbor	o	-	-	-	+	-
Decision Tree	+	+	+	+	o	+
Random Forest	-	-	+	+	o	+
Logistic Regression	+	+	+	o	-	o
Naive Bayes	o	-	+	+	-	-
SVM	-	-	-	o	-	-
Neural Network	-	-	+	o	-	+

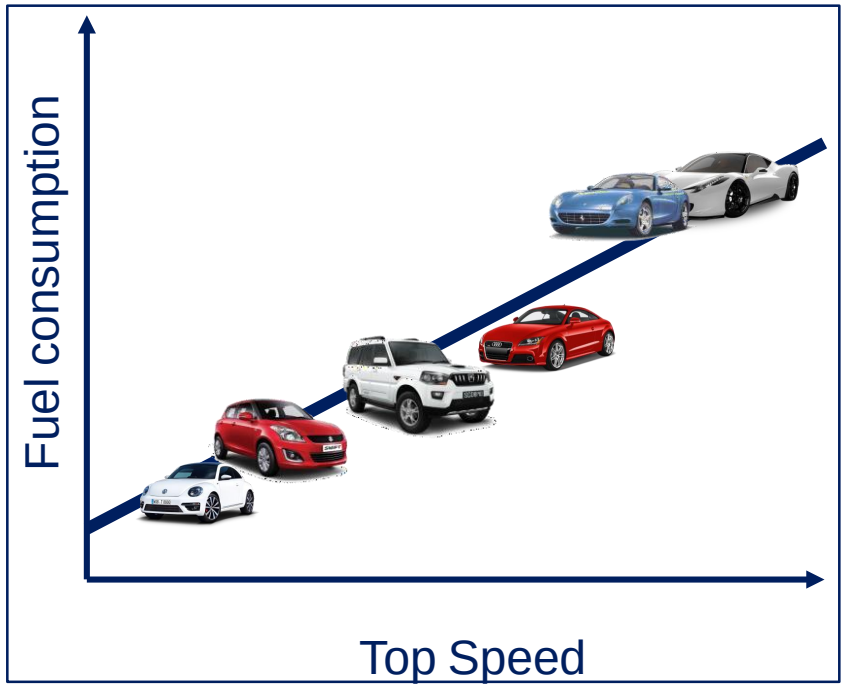
Concluzii

- Clasificarea este acţiunea de a asocia „etichete” obiectelor
- Există multe criterii de evaluare
  - Matricea de confuzie este utilizată în mod obişnuit
- Există mulţi algoritmi de clasificare
  - Bazați pe reguli, instanță, mulţimi, regresie etc.
- Algoritmi diferiţi pot fi potriviţi în situaţii diferite

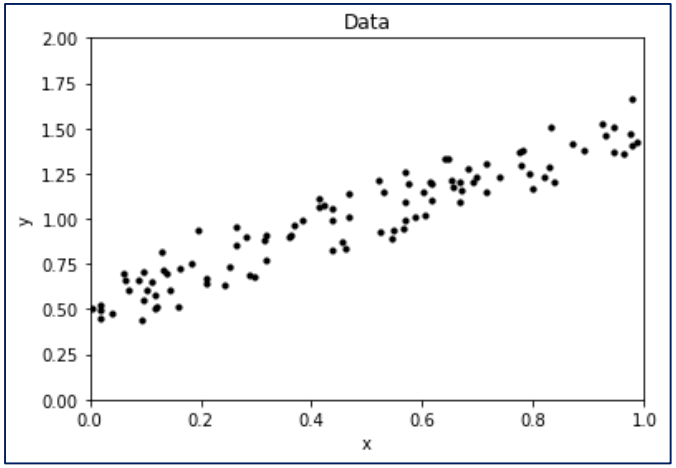
7. Regresie

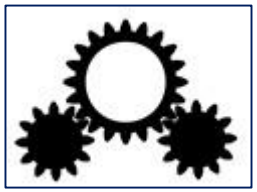
Exemplu de regresie

Regresie

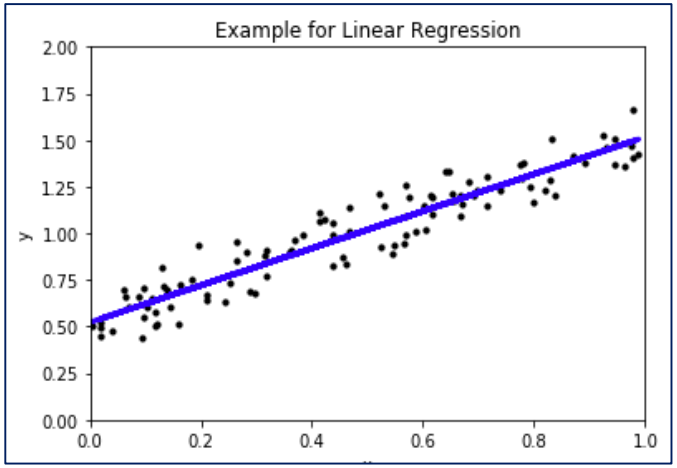


Problema generală



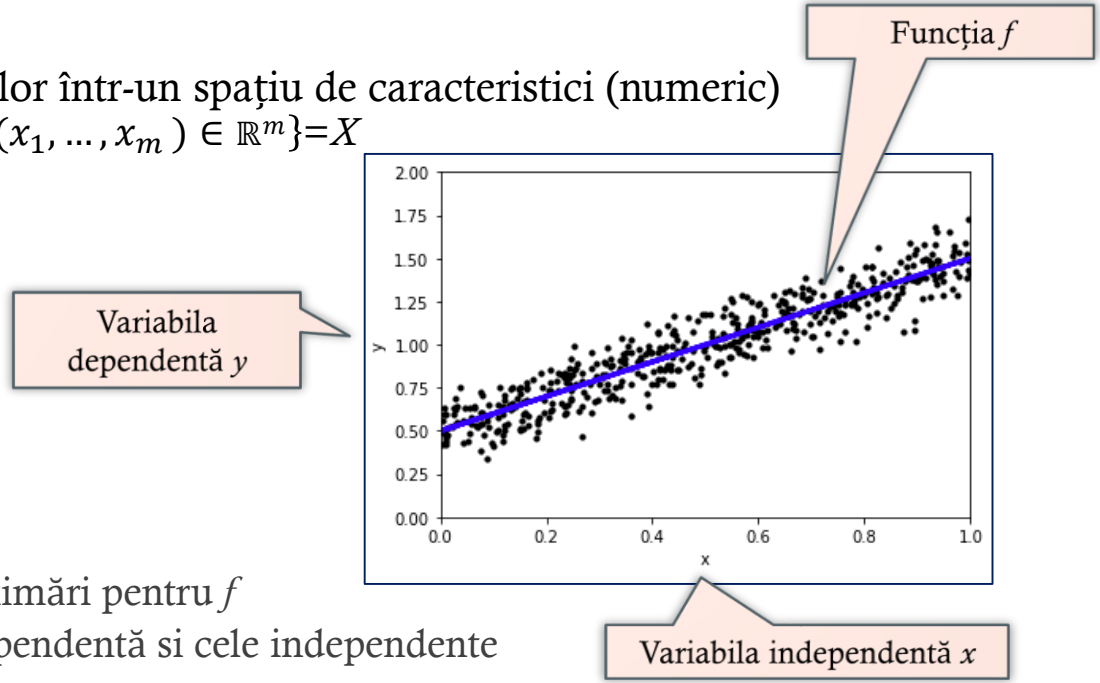


Regresie



Problema formală

- Spațiu de obiecte
  - $O = \{object_1, object_2, \dots\}$
  - Poate fi infinit
- Reprezentări ale obiectelor într-un spațiu de caracteristici (numeric)
  - $\mathcal{F} = \{\phi(o)\ ,\ o \in O\} = \{(x_1, \dots, x_m) \in \mathbb{R}^m\} = X$
  - Variabile independente
- Variabilă dependentă:
  - $f^*(o) = y \in \mathbb{R}$
- O funcție de regresie:
  - $f: \mathbb{R}^m \to \mathbb{R}$
- Regresie:
  - Determinarea unei aproximări pentru f
  - Relație între variabila dependentă si cele independente



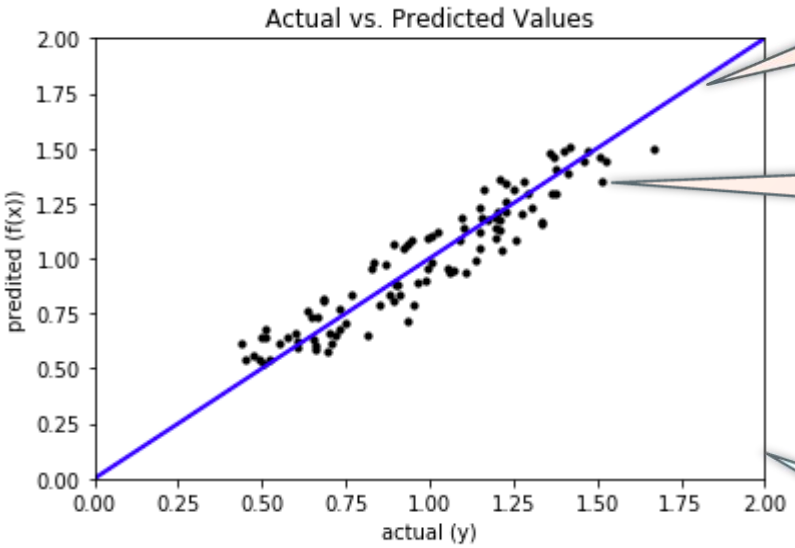
Calitatea regresiilor

Cum evaluăm $f^*(o)$?

- Scop: Aproximarea variabilei dependente
  - $f^*(o) \approx f(\phi(o))$
- Se folosesc date de test
  - Aceeași structură ca datele de antrenare
  - Se aplică funcția de regresie aproximată

$\phi(o)$					$f^*(o)$	$f(\phi\, o\,)$
Top Speed	Engine Size	Horse Power	Weigth	Year	value	prediction
250	1.4	130	1254	2003	**7.8**	**7.5**
280	1.8	185	1430	2010	**6.3**	**6.9**
…	…	…	…	…	…	

Comparație vizuală



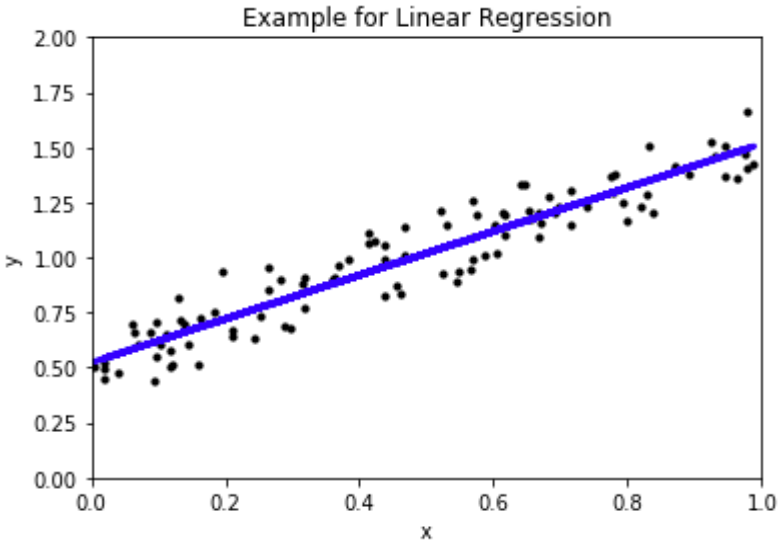
Predicție perfectă → Deviația de la această linie este un indicator vizual al calității predicției

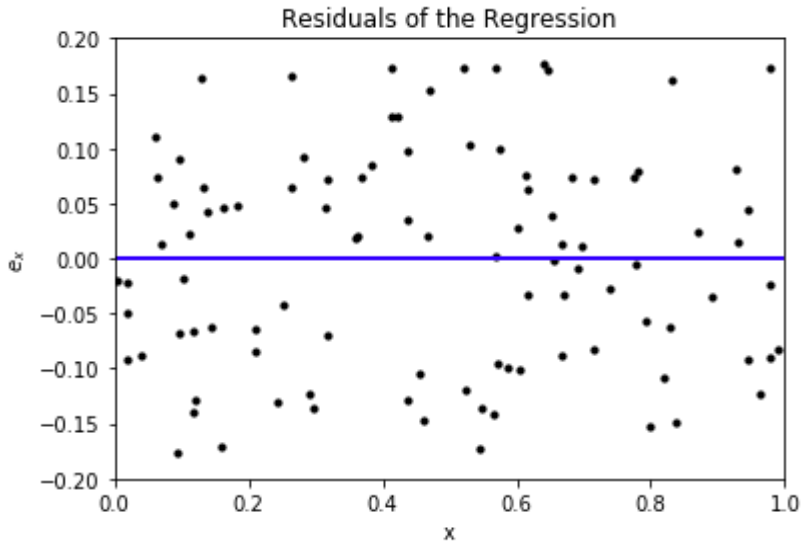
Predicție bună → Date apropiate de linia albastră, deviația are un pattern regulat

Permite inspectarea calității rezultatului (dacă predicția este corectă sau nu)

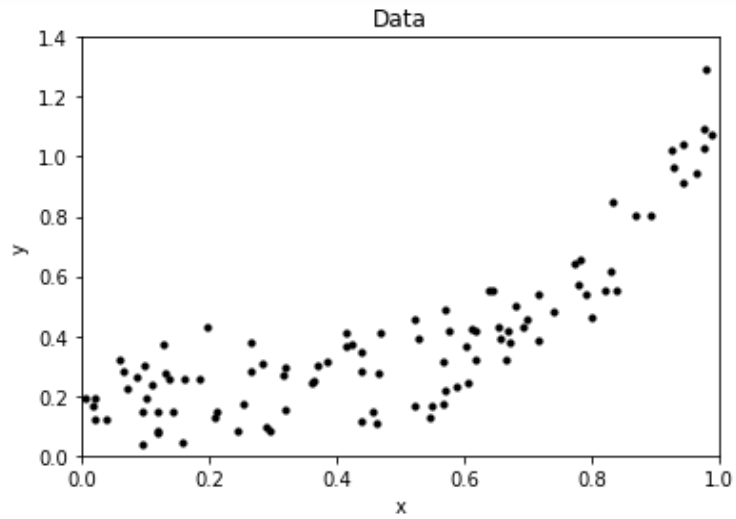
Valori reziduale

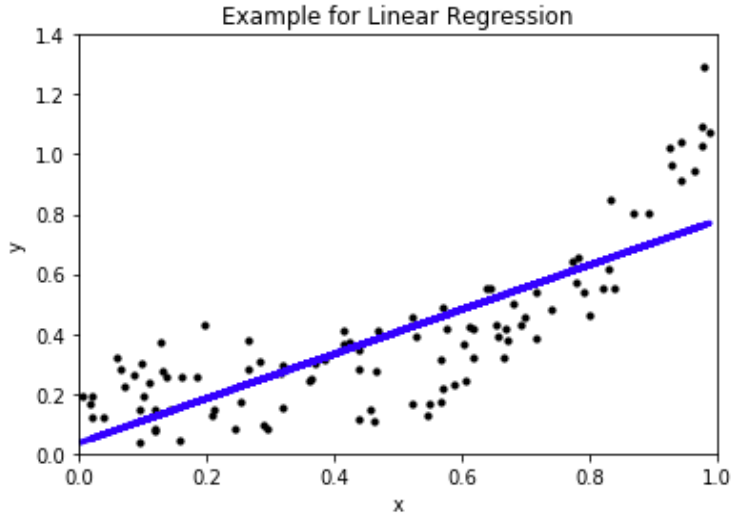
- Diferențele dintre predicții și valorile reale
  - $e_x = y - f(x)$

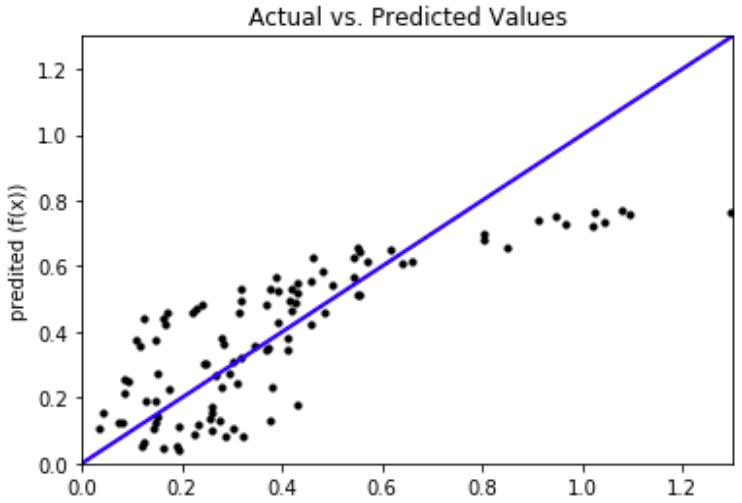


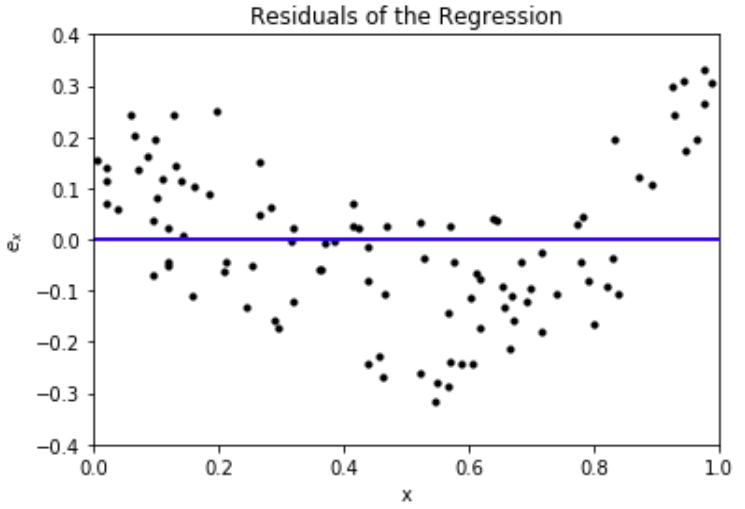


Comparația vizuală a unei predicții incorecte (*bad fit*)







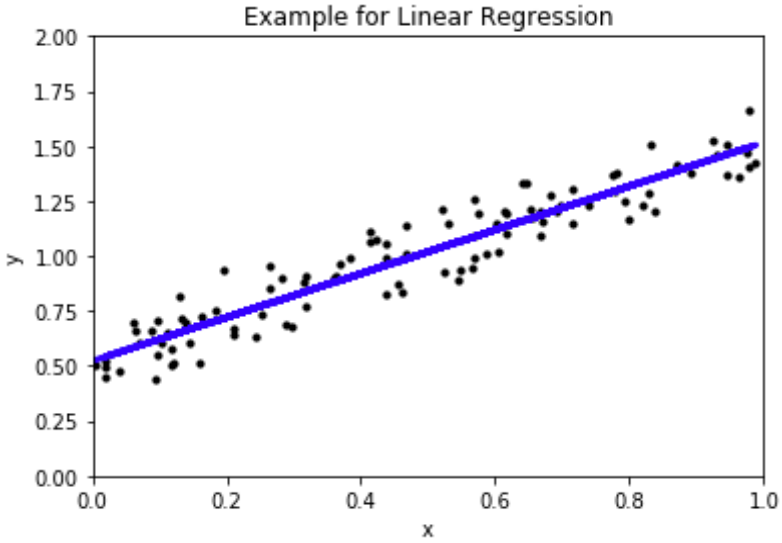


Măsuri ale calității regresiei

- Eroarea medie absolută (MAE – *Mean Absolute Error*)
  - $MAE = \frac{1}{|X|}\sum_{x \in X} e_x$
- Eroarea medie pătratică (MSE – *Mean Squared Error*)
  - $MSE = \frac{1}{|X|}\sum_{x \in X} (e_x)^2$
- Coeficientul de determinare (R^2)
  - Proporția varianței în variabila dependentă (fracția din varianță care este explicată de model)
  - $R^2 = 1 - \frac{\sum_{x \in X}(y - f(x))^2}{\sum_{x \in X}(y - mean(y))^2}$
- Coeficientul de determinare ajustat ($\bar{R}^2$)
  - Ia în considerare numărul de caracteristici
  - $\bar{R}^2 = 1 - (1 - R^2)\frac{|X| - 1}{|X| - m - 1}$

Regresia liniară

- Regresia ca funcţie liniară:
  - $y = b_0 + b_1 x_1 + \cdots + b_m x_m$
  - b_0 este intercepţia (valoarea lui y pentru care linia intersectează axa de coordonate *Oy*)
  - $b_1, \ldots, b_m$ sunt coeficienţii liniari
- Se calculează cu Metoda celor mai mici pătrate ordinară (OLS – *Ordinary Least Squares*)
  - Se optimizează MSE
  - $\min \|b_0 + Xb - y\|_2^2$



Pătratul distanţei euclidiene

  - $X = \begin{pmatrix} x_{1,1} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,m} \end{pmatrix}$

n = numărul de instanţe din datele de antrenament

  - $b = (b_1, \ldots, b_m)$
  - $y = (y_1, \ldots, y_n)$

Regresia Ridge

- Este tot o funcție liniară
- Metoda celor mai mici pătrate (OLS) permite soluții multiple pentru $n > m$
- Regresia Ridge penalizează soluțiile cu coeficienți mari
- Se calculează cu regularizarea lui Tihonov:
  - $\min\|b_0 + Xb - y\|_2^2 + \|\Gamma b\|_2^2$
  - Se folosește $\Gamma = \alpha I$
- Se folosește α pentru a ajusta puterea de regularizare:
  - $\min\|b_0 + Xb - y\|_2^2 + \alpha\|b\|_2^2$

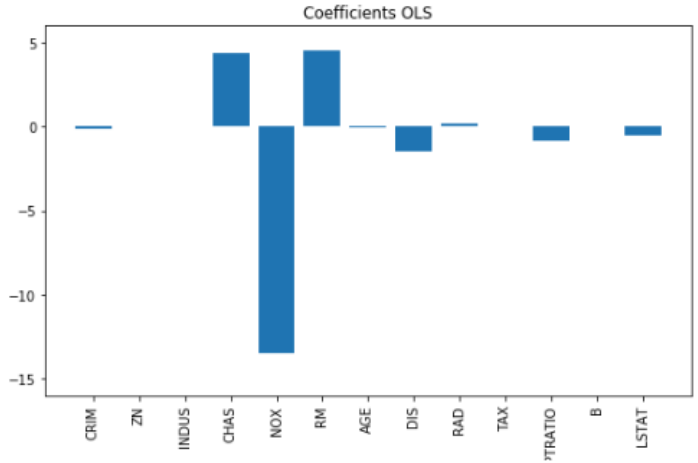
Regresia Lasso

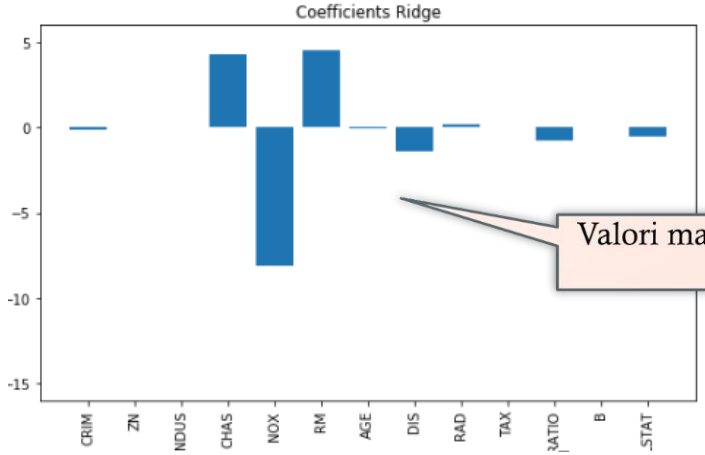
- Este tot o funcție liniară
- Penalizarea coeficienților mari nu reduce redundanța
  - Exemplu extrem: caracteristici identice a căror predicție este perfectă
    - $y = x_1 = x_2$
  - Ridge:
    - $b_1 = b_2 = 0.5$
  - Ar fi mai potrivit ca unul dintre coeficienți să fie 0:
    - $b_1 = 1, b_2 = 0$
- Regresia Lasso: regresie Ridge cu norma Manhatten
  - $\min\|b_0 + Xb - y\|_2^2 + \alpha\|b\|_1$
- Mărește probabilitatea coeficienților egali cu 0
  - Selecție mai bună a caracteristicilor relevante

Regresia *Elastic net*

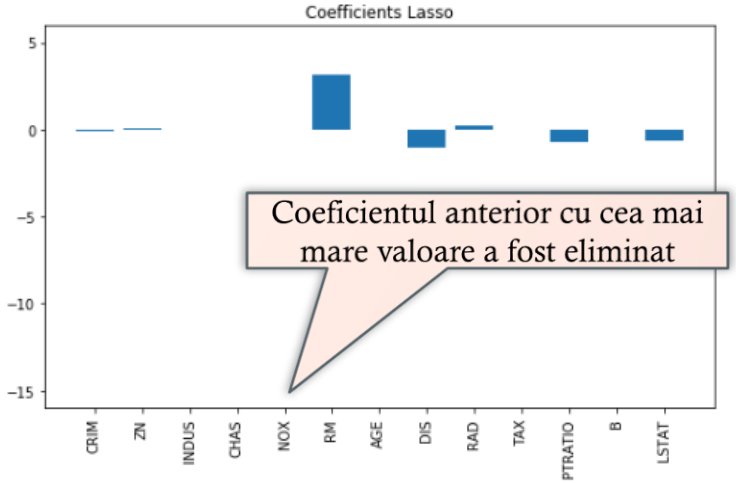
- Este tot o funcție liniară
- Regresia Lasso tinde să selecteze aleator una dintre mai multe caracteristici corelate
  - Se poate pierde informație
- *Elastic Net* combină regresiile Ridge și Lasso
  - Reține doar caracteristicile corelate relevante și minimizează coeficienții
- Se folosește un raport ρ pentru a da mai multă pondere regresiei Ridge, respectiv Lasso
  - $\min\|b_0 + Xb - y\|_2^2 + \rho\alpha\|b\|_1 + \frac{(1-\rho)}{2}\alpha\|b\|_2^2$

Comparație a modelelor de regresie

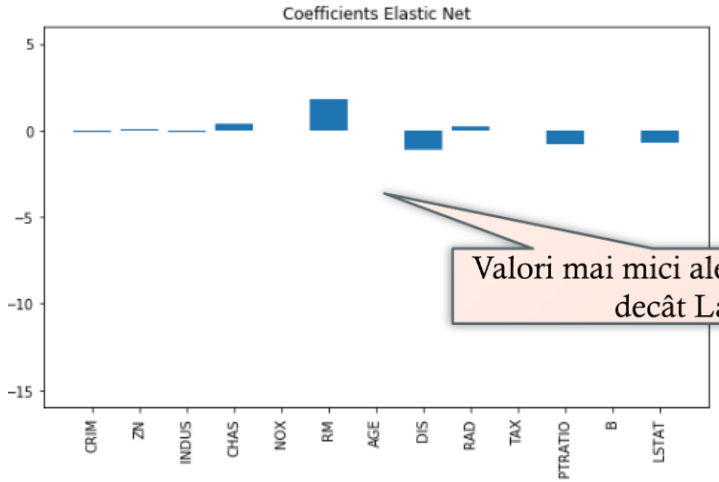




Valori mai mici ale coeficienților decât OLS



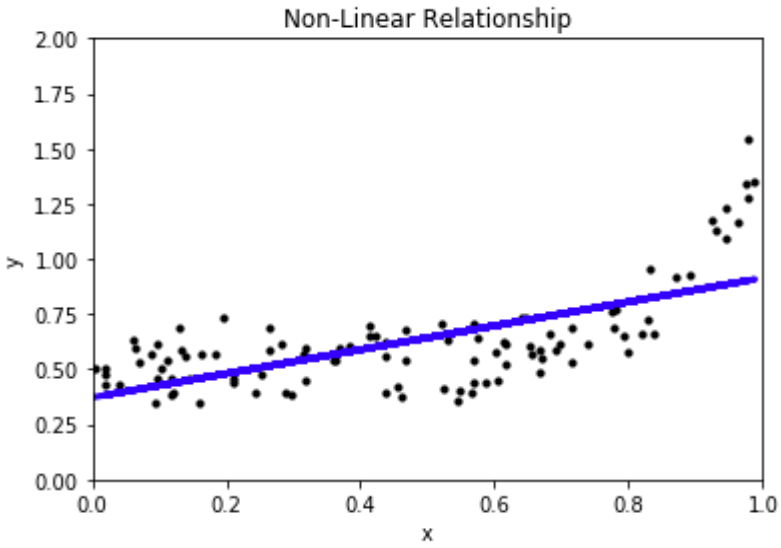
Coeficientul anterior cu cea mai mare valoare a fost eliminat



Valori mai mici ale coeficienților decât Lasso

Regresie neliniară

- Multe relații nu sunt liniare



- Regresie polinomială
- Support Vector Regression
- Rețele neuronale

Concluzii

- Regresia determină relațiile dintre variabilele independente și cele dependente
- Regresia liniară este un model simplu, adeseori eficient
- Regularizarea poate îmbunătăți soluțiile (Lasso, Ridge, Elastic net etc)
- Există și multe abordări neliniare
  - Necesită atenție atunci când sunt aplicate
  - Se poate ajunge ușor la overfitting