Predavanje 5

Algoritam logističke regresije

Prof. dr. Adis Alihodžić

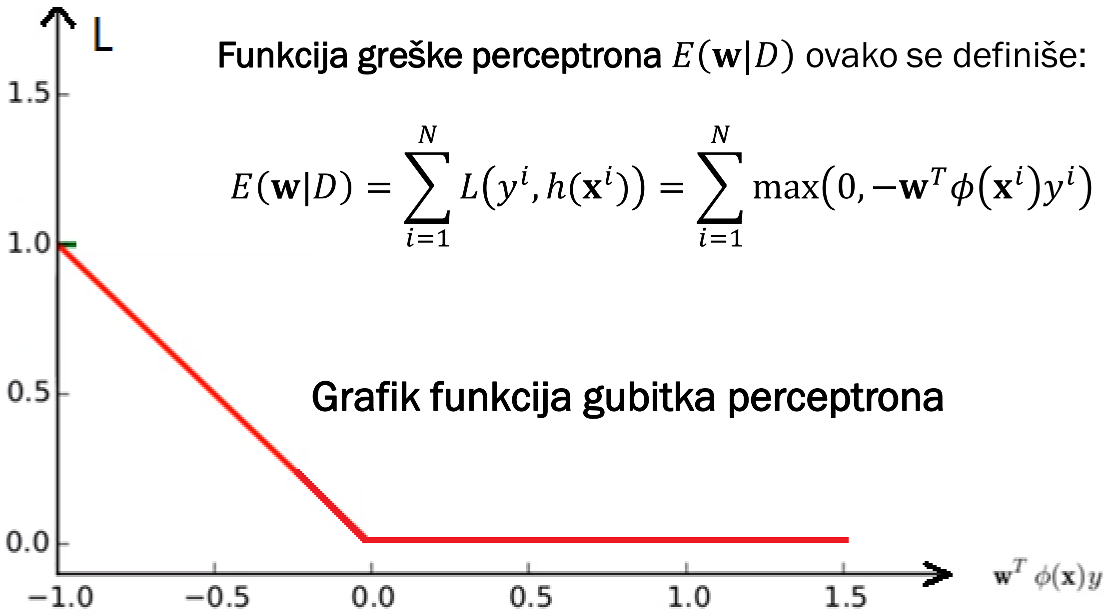
MODEL PERCEPTRONA

Funkcija gubitka perceptrona $L(y, h(\mathbf{x}))$ može se ovako zadati:

$$L(y, h(\mathbf{x})) = \max(0, -\mathbf{w}^T \phi(\mathbf{x}) y)$$

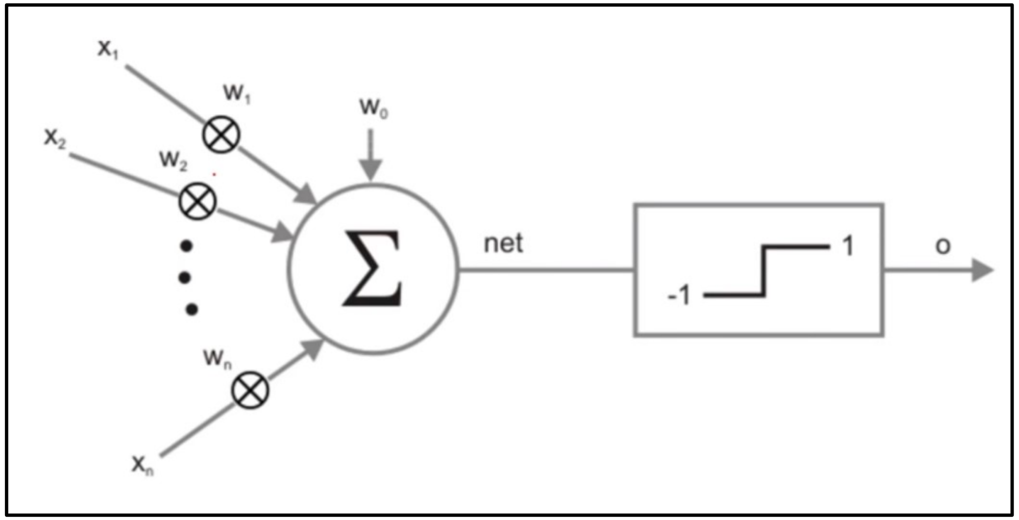
Funkcija greške perceptrona $E(\mathbf{w}|D)$ ovako se definiše:

$$E(\mathbf{w}|D) = \sum_{i=1}^{N} L\left(y^i, h(\mathbf{x}^i)\right) = \sum_{i=1}^{N} \max\left(0, -\mathbf{w}^T \phi\left(\mathbf{x}^i\right) y^i\right)$$



Grafik funkcija gubitka perceptrona

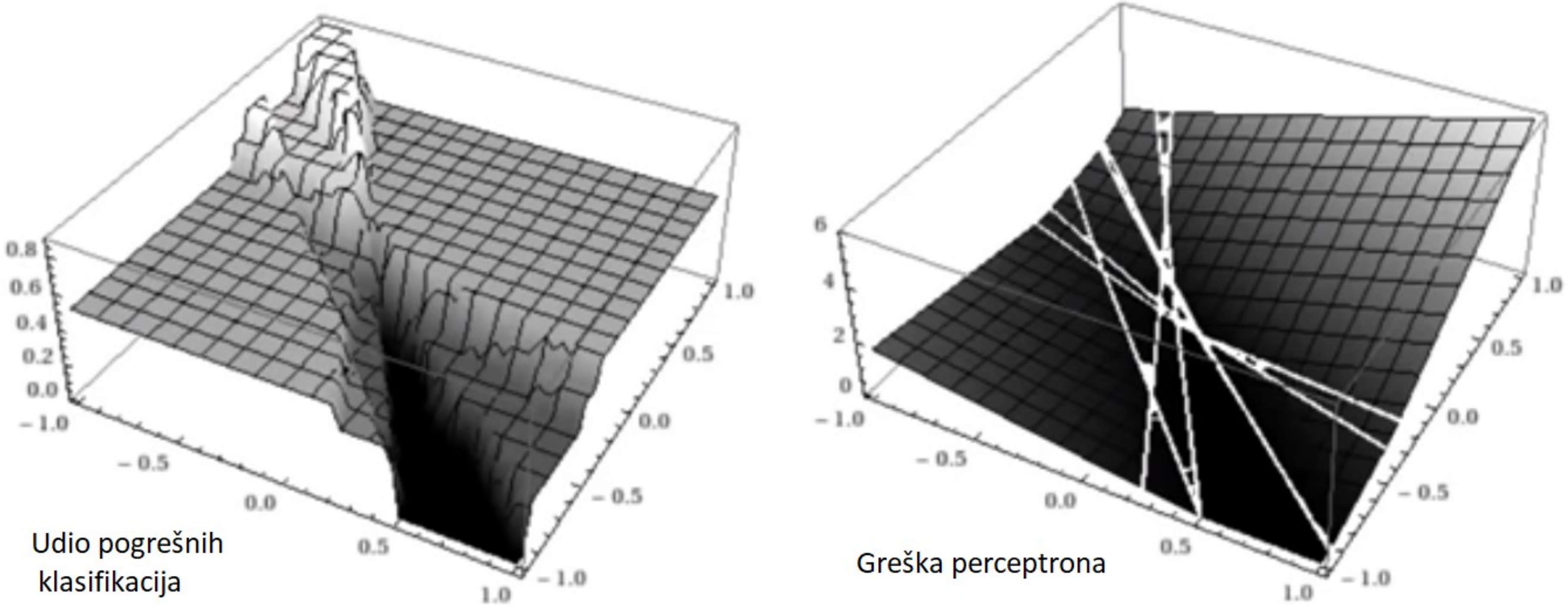
Vještački neuron, 1943.



Funkcija gubitka perceptrona $L(y, h(\mathbf{x}))$ za netačno klasificiranu instancu $\mathbf{x}$ jednaka je $-\mathbf{w}^T \phi(\mathbf{x}) y$.

MODEL PERCEPTRONA

Funkcija greške perceptrona u prostoru težinskih koeficijenata



OPTIMIZACIJA MODELA PERCEPTRONA

Izvod funkcije gubitka za netačno klasificiranu instancu ovako se zadaje:

$$\nabla_{\mathbf{w}} L(\mathbf{w}|D) = \nabla_{\mathbf{w}}(-\mathbf{w}^T \phi(\mathbf{x}) y) = -\phi(\mathbf{x}) y$$

Pravilo ažuriranja težinskog vektora $\boldsymbol{w}$ zvano Widrow-Hoftovo Delta pravilo ovako glasi:

$$\mathbf{w} = \mathbf{w} - \eta \nabla L(\mathbf{w}|D)$$

Dakle, ažuriranje se provodi u suprotnom smjeru od gradijenta ∇L funkcije gubitka L.

Korak ažuriranje se zadaje preko konstante η.

Perceptron ne kažnjava tačno klasificirane instance, dok to regresija može uraditi.

ALGORITAM PERCEPTRONA

Algoritam perceptrona

inicijaliziraj $\mathbf{w} \leftarrow (0, \ldots, 0)$

ponavljaj do konvergencije

  za $i = 1, \ldots, N$

    ako $f\left(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}^{(i)})\right) \neq y^{(i)}$ **onda** $\mathbf{w} \leftarrow \mathbf{w} + \eta\phi(\mathbf{x}^{(i)})y^{(i)}$

PITANJE: Da li će ovaj algoritam zakonvergirati ukoliko su instance dataseta D linearno separabilne?

PREDNOSTI I NEDOSTACI ALGORITMA PERCEPTRONA

Prednosti:

1. Robustniji od regresije
2. Jednostavan postupak

Nedostatci:

1. Izlazi modela nemaju vjerojatnosnu intepretaciju
2. Rezultat (hipoteza) ovisi o početnim težinama i redoslijedu korekcije
3. Ne konvergira ako primjeri nisu linearno odvojivi

ALGORITAM MAŠINSKOG UČENJA

Svaki algoritam mašinskog učenja je definiran sa tri komponente:

A. Model

B. Funkcija gubitka

C. Postupak optimizacije

NEOPHODNO IZVESTI KOD LOGISTIČKE REGRESIJE

Model:

$$h(\mathbf{x};\mathbf{w}) = \sigma(\mathbf{w}^\intercal \boldsymbol{\phi}(\mathbf{x})) = \frac{1}{1+\exp(-\mathbf{w}^\intercal \boldsymbol{\phi}(\mathbf{x}))}$$

Funkcija greške/gubitka:

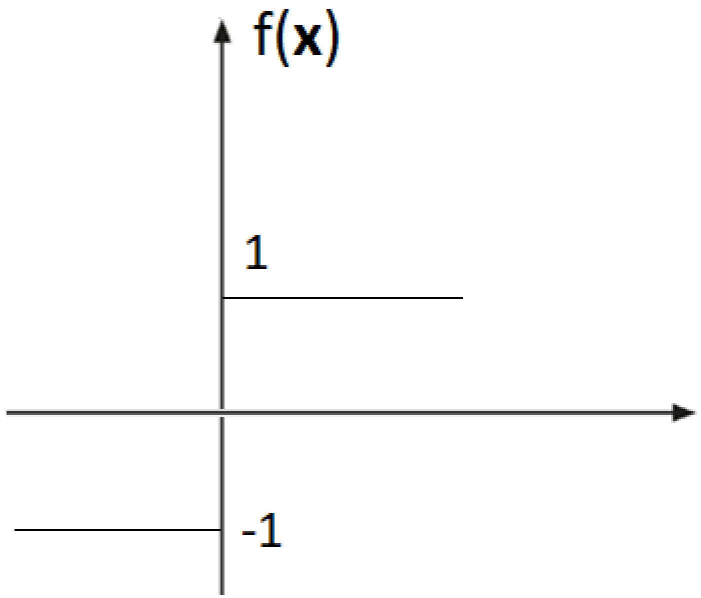
$$E(\mathbf{w}|\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N}\Big(-y^{(i)}\ln h(\mathbf{x}^{(i)};\mathbf{w}) - (1-y^{(i)})\ln\big(1-h(\mathbf{x}^{(i)};\mathbf{w})\big)\Big)$$

$$L(y,h(\mathbf{x};\mathbf{w})) = -y\ln h(\mathbf{x};\mathbf{w}) - (1-y)\ln\big(1-h(\mathbf{x};\mathbf{w})\big)$$

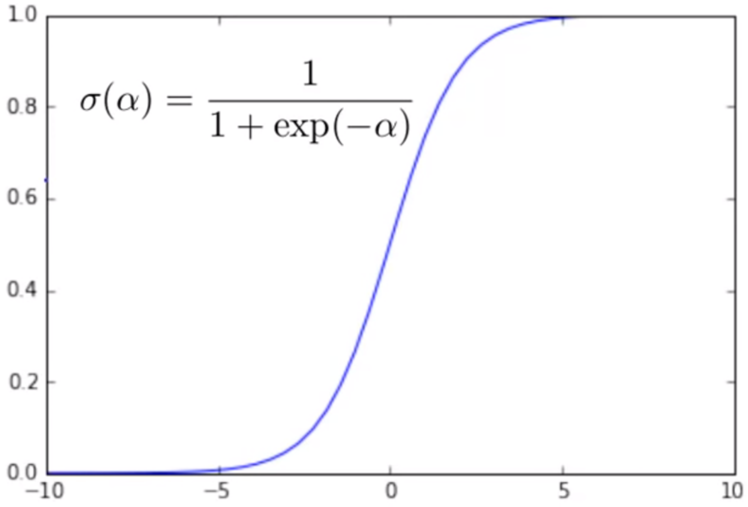
Za razliku od perceptrona, logistička regresija istovremeno posjeduje: robusnost, konvergenciju i probabilistički izlaz.

MODEL LOGISTIČKE REGRESIJE

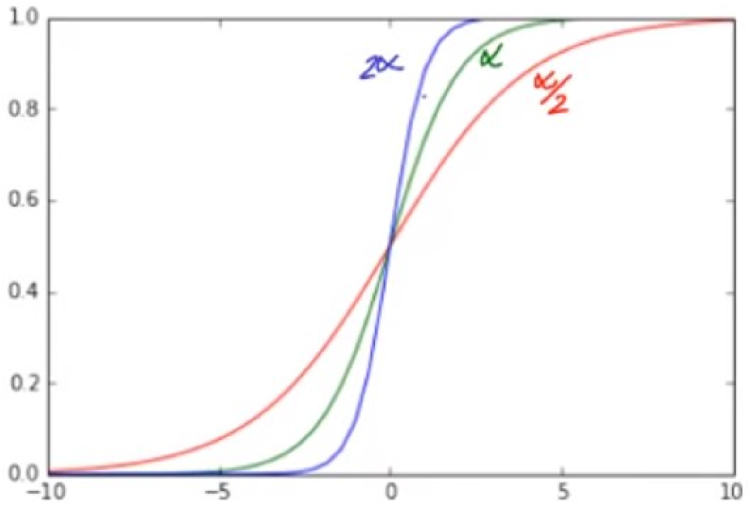
Poopćeni linearni model $h(\boldsymbol{x}, \boldsymbol{w}) = f(\boldsymbol{w}^T \phi(\boldsymbol{x}))$



Graf logističke funkcije



Sigmoidna f-ja za razne vrijed.



MODEL LOGISTIČKE REGRESIJE

Model logističke regresije se bazira na Bernulijevoj distribuciji neke ciljne varijable *y*.

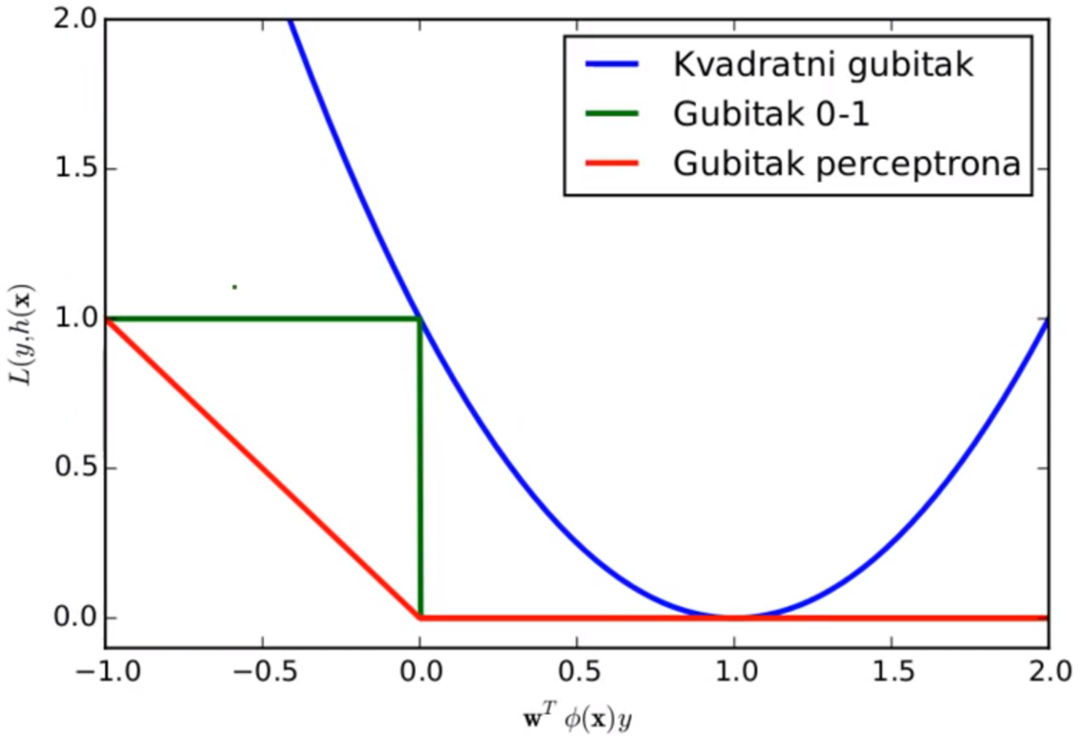
$$p(y|\mathbf{x}) = \begin{cases} \mu & , y=1 \\ 1-\mu & , y=0 \end{cases}$$

Model logističke regresije u općem obliku se definira kao:

$$h(\boldsymbol{x}, \boldsymbol{w}) = \mu = \sigma\left(\boldsymbol{w}^T \phi(\boldsymbol{x})\right) = \frac{1}{1 + e^{-\boldsymbol{w}^T \phi(\boldsymbol{x})}}$$

MODEL LOGISTIČKE REGRESIJE

Neki primjeri funkcija gubitaka *L*



IDEALNO: Funkcija gubitka bi trebala biti što sličnija funkciji gubitka 0-1.

OPIS: Funkcija gubitka 0-1 kažnjava **neatačno klasifikovane instance**, ostale ostavlja na miru.

Netačno klasifikovane instance su one kod kojih vrijedi:

$$\boldsymbol{w}^T \phi(\boldsymbol{x}) \leq 0$$

NEDOSTATAK: Funkcija gubitka 0-1 nije derivabilna.

GREŠKA UNAKRSNE ENTROPIJE

Funkcija vjerodostojnosti $L(\mathbf{w})$ ovako se definira:

$$L(\mathbf{w}) = \prod_{i=1}^{N} p_{\mathbf{w}}(y_i|\mathbf{x}_i)$$

Empirijska greška se definira kao **negativna vrijednost logaritamske funkcije vjerodostojnosti**, tj. ovako:

$$E(h|D) = E(\boldsymbol{w}|D) = -\ln L(w)$$

Nakon raspisivanja se dobija:

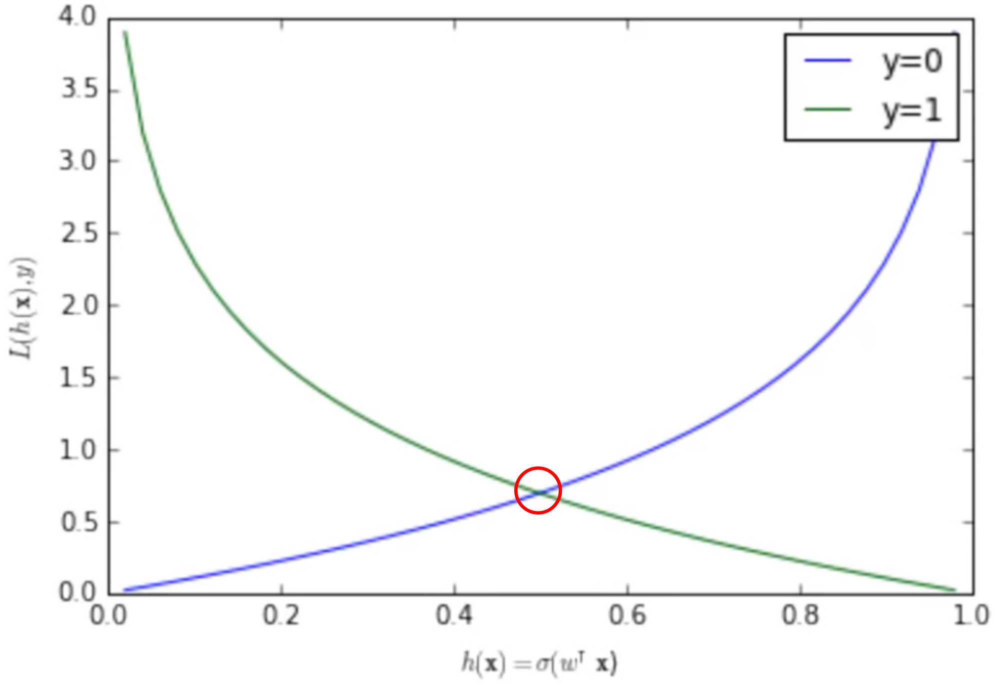
$$E(\boldsymbol{w}|D) = -\ln \prod_{i=1}^{N} p_{\mathbf{w}}(y_i|\mathbf{x}_i) = -\sum_{i=1}^{N} \ln\left(p_{\mathbf{w}}(y_i|\mathbf{x}_i)\right) = -\sum_{i=1}^{N} \ln\left(h(\boldsymbol{x}_i, \boldsymbol{w})^{y^i} \cdot \left(1 - h(\boldsymbol{x}_i, \boldsymbol{w})^{1-y^i}\right)\right) =$$

$$= \sum_{i=1}^{N} \left(-\mathrm{y}^i \ln\left(h(\boldsymbol{x}_i, \boldsymbol{w})\right) - \left(1 - y^i\right) \ln\left(1 - h(\boldsymbol{x}_i, \boldsymbol{w})\right)\right)$$

FUNKCIJA GUBITKA

Funkcija gubitka logističke regresije u oznaci $L(h(\mathbf{x}), y)$ definira se na sljedeći način:

$$L(h(\boldsymbol{x}), y) = -y \ln\big(h(\mathbf{x})\big) - (1 - y) \ln\big(1 - h(\mathbf{x})\big)$$



Kako funkcioniše kažnjavanje koristeći funkciju gubitka L(h(**x**),y)?

Da li je funkcija $L(h(\mathbf{x}), y)$ konveksna?

Šta se dešava kada je izlaz modela jednak 0.5, $h(\boldsymbol{x}) = 0.5$?

Kada su gubici najmanji za obje klase, tj. y=0 odnosno y=1?

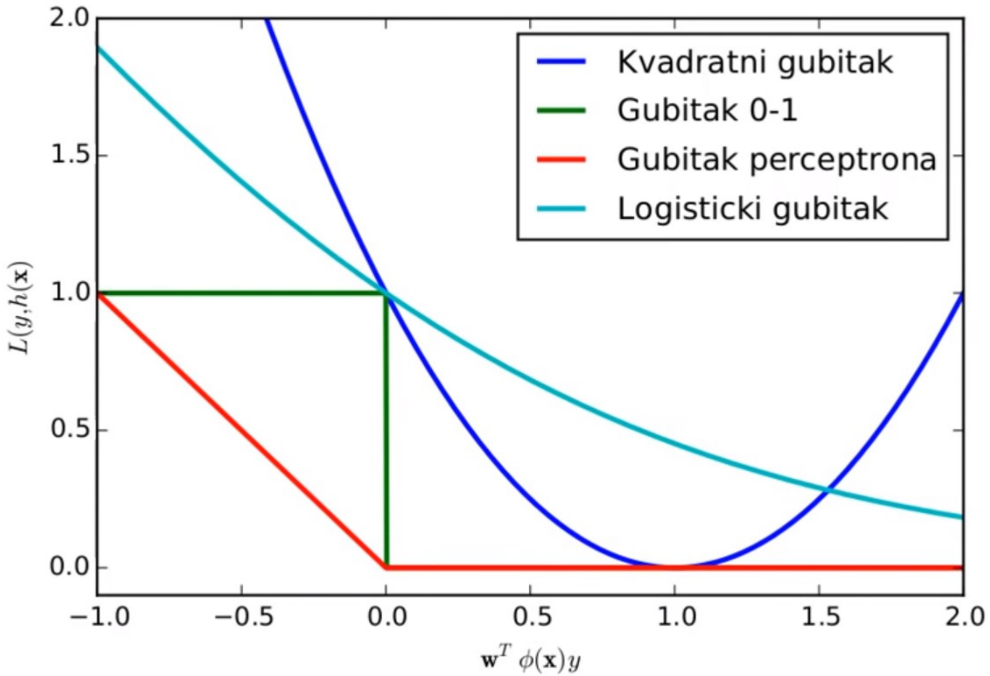
Kada nema gubitaka?

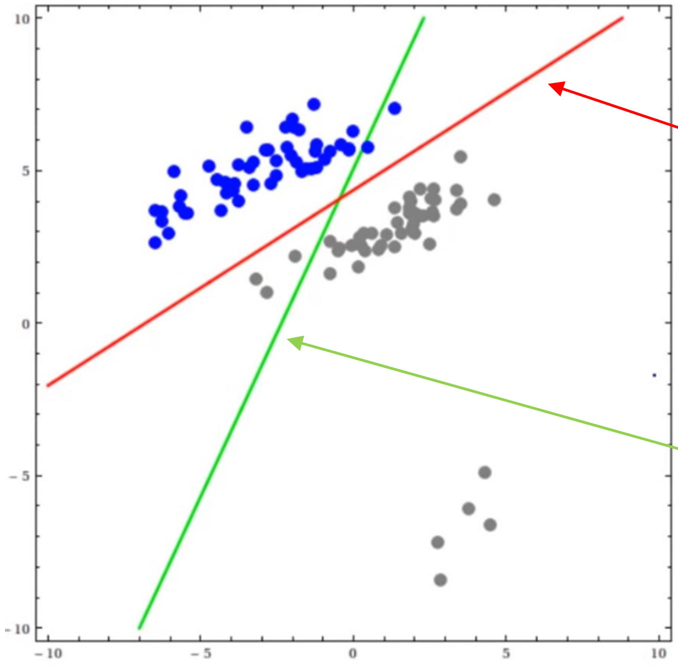
LOGISTIČKA FUNKCIJA GUBITKA

Logistička funkcija gubitka definira se ovako:

$$L(h(\boldsymbol{x}), y) = \frac{1}{\ln 2} \ln\left(1 + e^{-y\boldsymbol{w}^T\phi(\boldsymbol{x})}\right)$$

Šta se može zaključiti?





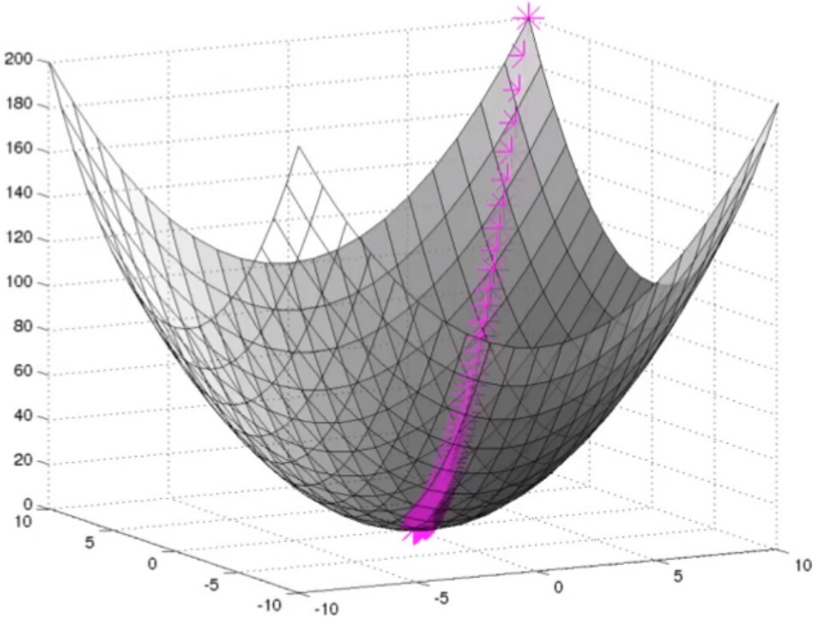
Klasifikacija logističkom regresijom

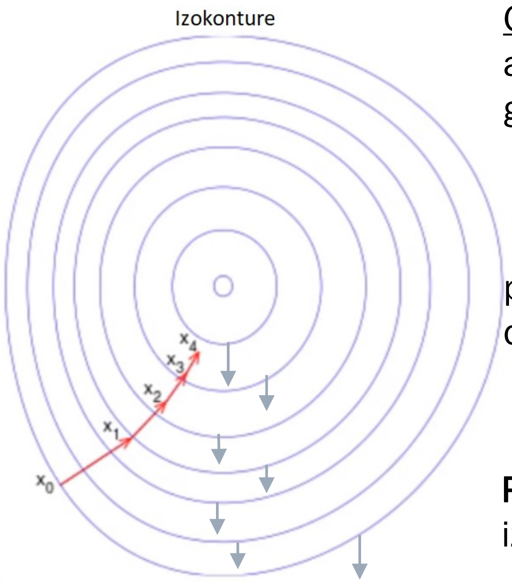
Klasifikacija linearnom regresijom odnosno linearnom hipotezom

OPĆENITO O GRADIJENTNOM SPUSTU

PITANJE: Da li se kod logističke regresije odnosno kod perceptrona može pronaći težinski vektor $\boldsymbol{w}^*$u zatvorenoj formi koji predstavlja njeno rješenje?

Postupak iterativne optimizacije – gradijentni spust





Gradijentni spust kaže da se težine uvijek trebaju ažurirati u suprotnom smjeru od gradijenta funkcije greške, tj. ažuriranje se ovako obavlja:

$$\mathbf{w} = \mathbf{w} - \eta \nabla E(\mathbf{w}|D),$$

pri čemu konstanta η govori koliki se koraci prave dok se vrši spuštanje gradijentnim spustom.

PITANJE: Da li će se gradijentni spust svesti na traženje izvoda funkcije greške E(**w**|D)?

DVA TIPA ALGORITMA GRADIJENTNOG SPUSTA

U zavisnosti od izvođenje deriviranja funkcije greške $E(\mathbf{w}|D)$, predlažu se dva algoritma gradijentnog spusta:

A. Grupni gradijentni spust

Ažuriranje se ovdje obavlja ovako:

$$\mathbf{w} = \mathbf{w} - \eta \sum_{i=1}^{N} \nabla L(y^i, h(\mathbf{x}^i, \mathbf{w}))$$

B. Stohastički gradijentni spust

Ažuriranje se ovdje obavlja odmah:

$$\mathbf{w} = \mathbf{w} - \eta \, \nabla L$$

DVA TIPA ALGORITMA GRADIJENTNOG SPUSTA

Algoritam grupnog gradijentnog spusta

Gradijentni spust (*batch*)

inicijaliziraj $\mathbf{w} \leftarrow (0, \ldots, 0)$
ponavljaj do konvergencije
 $\Delta\mathbf{w} = (0, \ldots, 0)$
 za $i = 1, \ldots, N$
  $\Delta\mathbf{w} \leftarrow \Delta\mathbf{w} - \nabla L\big(y^{(i)}, h(\mathbf{x}^{(i)}; \mathbf{w})\big)$
 $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru $\Delta\mathbf{w}$
 $\mathbf{w} \leftarrow \mathbf{w} + \eta\Delta\mathbf{w}$

Algoritam pojedinačnog gradijentnog spusta

Stohastički gradijentni spust – SGD

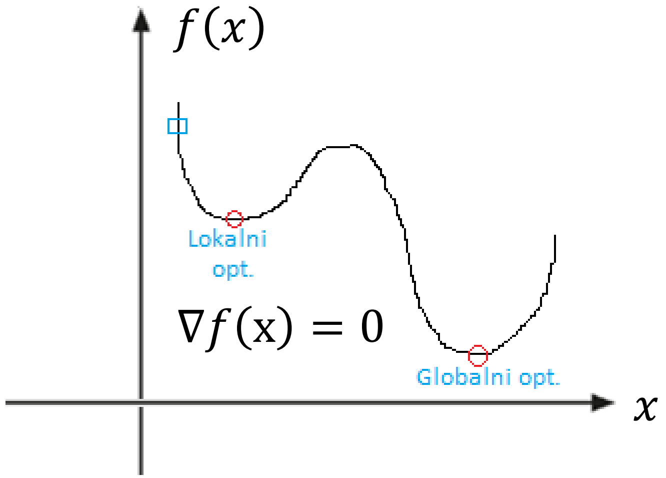
inicijaliziraj $\mathbf{w} \leftarrow (0, \ldots, 0)$
ponavljaj do konvergencije
 slučajno permutiraj primjere u $\mathcal{D}$
 za $i = 1, \ldots, N$
  $\Delta\mathbf{w} \leftarrow -\nabla L\big(y^{(i)}, h(\mathbf{x}^{(i)}; \mathbf{w})\big)$
  $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru $\Delta\mathbf{w}$
  $\mathbf{w} \leftarrow \mathbf{w} + \eta\Delta\mathbf{w}$

Algoritam linijskog pretraživanja garantuje svojstvo konvergencije, tj. on kaže da će postupak gradijentnog spusta sigurno konvergirati, iako se ne kaže prema kojem optimumu.

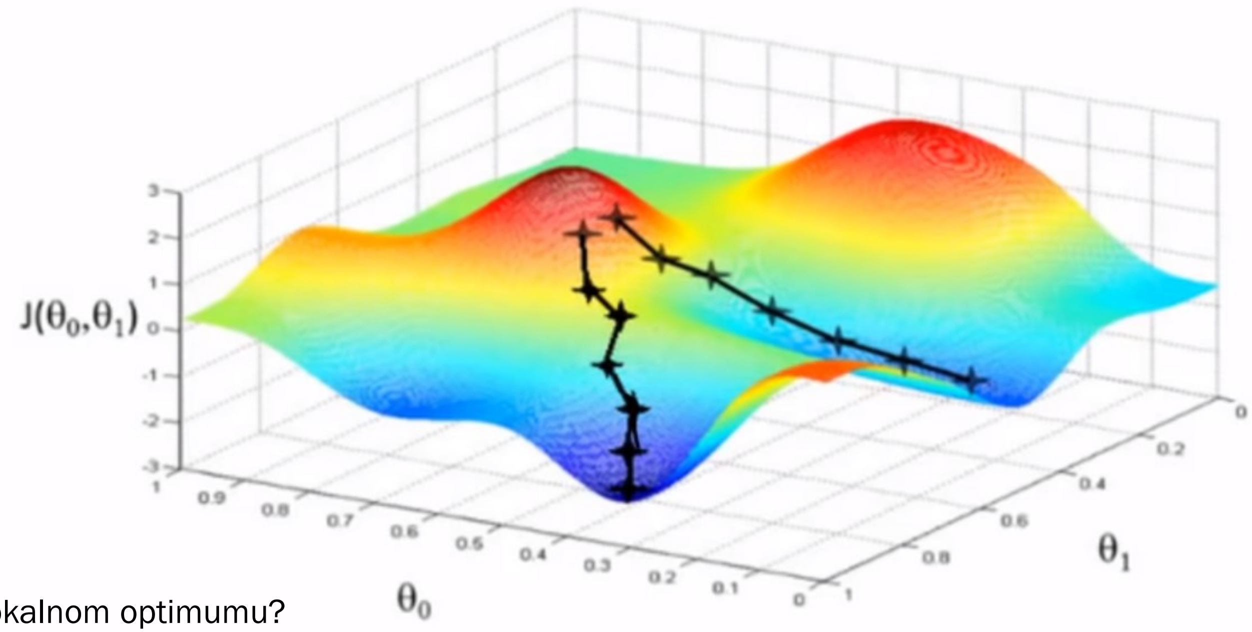
Šta se može zaključiti na osnovu pseudokodova ovih algoritama?

ZAGLAVLJIVANJE GRADIJENTNOG SPUSTA U LOKALNI OPTIMUM

Zaglavljivanje u ravni



Zaglavljivanje u prostoru



PITANJE: Kako riješiti to da se algoritam gradijentnog spusta ne zaglavljuje u lokalnom optimumu?

KONVEKSNA OPTIMIZACIJA

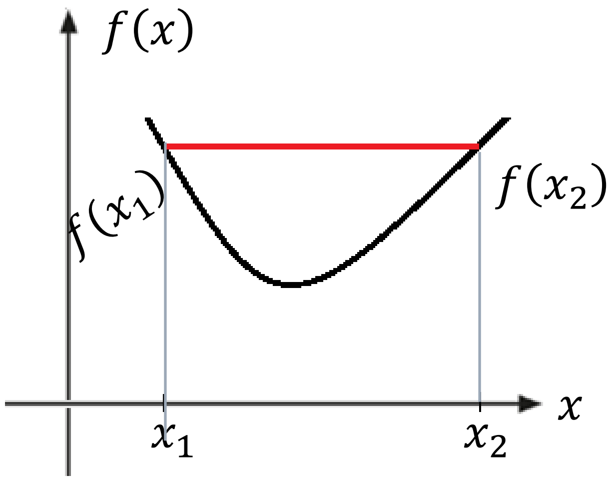
Definicija. Funkcija $f : C^1 \to \mathbb{R}$ je **konveksna** ako za bilo koja dva vektora odluke $\mathbf{x}_1$, $\mathbf{x}_2 \in C$ vrijedi

$$f(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) \leq \lambda f(\mathbf{x}_1) + (1-\lambda) f(\mathbf{x}_2), \ \lambda \in [0,1]$$

U slučaju da za bilo koja dva međusobno različita vektora odluke $\mathbf{x}_1$ i $\mathbf{x}_2$ iz C i za $0 < \lambda < 1$ vrijedi

$$f(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) < \lambda f(\mathbf{x}_1) + (1-\lambda) f(\mathbf{x}_2)$$

onda se kaže da je funkcija f **striktno konveksna**. Skup C^1 predstavlja skup svih neprekidno diferencijabilnih funkcija.



PITANJE: Ako je funkcija gubitka konveksna, da li će biti konveksna i njena funkcija empirijske greška?

GRADIJENT LOGISTIČKE REGRESIJE

$$E(\mathbf{w}|\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N}\Big(-y^{(i)}\ln h(\mathbf{x}^{(i)};\mathbf{w}) - (1-y^{(i)})\ln\big(1-h(\mathbf{x}^{(i)};\mathbf{w})\big)\Big)$$

$$\nabla_{\mathbf{w}}E(\mathbf{w}|\mathcal{D}) = \frac{1}{N}\sum_{i=1}^{N}\nabla L\big(y^{(i)}, h(\mathbf{x}^{(i)};\mathbf{w})\big)$$

$$\nabla L\big(y, h(\mathbf{x})\big) = \Big(-\frac{y}{h(\mathbf{x})} + \frac{1-y}{1-h(\mathbf{x})}\Big)h(\mathbf{x})\big(1-h(\mathbf{x})\big)\boldsymbol{\phi}(\mathbf{x}) = \big(h(\mathbf{x})-y\big)\boldsymbol{\phi}(\mathbf{x})$$

Gradijent funkcije gubitka

$$\boxed{\nabla E(\mathbf{w}|\mathcal{D}) = \sum_{i=1}^{N}\big(h(\mathbf{x}^{(i)}) - y^{(i)}\big)\boldsymbol{\phi}(\mathbf{x}^{(i)})}$$

Gradijent empirijske greške

ALGORITAM GRADIJENTNOG SPUSTA LOGISTIČKE REGRESIJE

Algoritam grupnog gradijentnog spusta

Logistička regresija (gradijentni spust)

$\mathbf{w} \leftarrow (0, 0, \ldots, 0)$
ponavljaj do konvergencije
  $\Delta\mathbf{w} \leftarrow (0, 0, \ldots, 0)$
  za $i = 1, \ldots, N$
    $h \leftarrow \sigma(\mathbf{w}^{\mathrm{T}}\mathbf{x}^{(i)})$
    $\Delta\mathbf{w} \leftarrow \Delta\mathbf{w} - (h - y^{(i)})\,\mathbf{x}^{(i)}$
  $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru $\Delta\mathbf{w}$
  $\mathbf{w} \leftarrow \mathbf{w} + \eta\Delta\mathbf{w}$

Algoritam pojedinačnog gradijentnog spusta

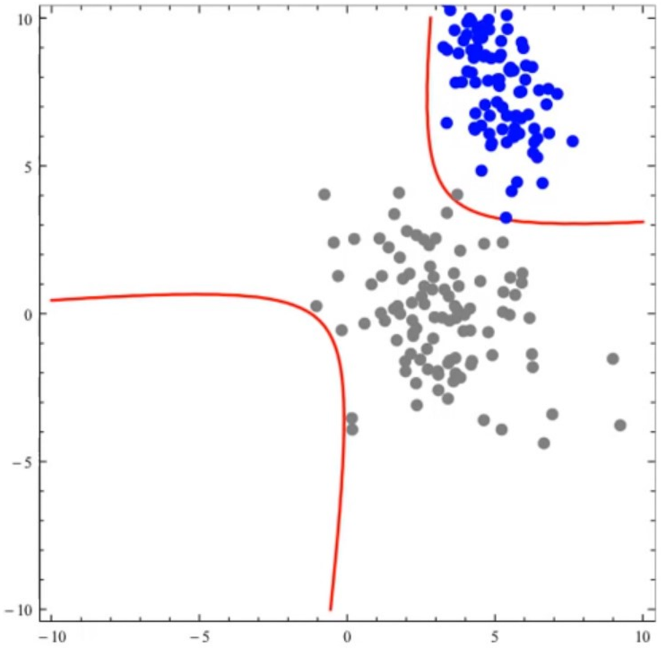
Logistička regresija (stohastički gradijentni spust)

$\mathbf{w} \leftarrow (0, 0, \ldots, 0)$
ponavljaj do konvergencije
  slučajno permutiraj primjere u $\mathcal{D}$
  za $i = 1, \ldots, N$
    $h \leftarrow \sigma(\mathbf{w}^{\mathrm{T}}\mathbf{x}^{(i)})$
    $\Delta\mathbf{w} \leftarrow -(h - y^{(i)})\mathbf{x}$
    $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru $\Delta\mathbf{w}$
    $\mathbf{w} \leftarrow \mathbf{w} + \eta\Delta\mathbf{w}$

REGULARIZACIJA LOGISTIČKA REGRESIJE

Primjer modela logističke regresije bez regularizacije

$$\phi(\mathbf{x}) = (1, x_1, x_2, x_1x_2, x_1^2, x_2^2)$$



Upotreba regularizacije kod logističke regresije postiže da sigmoida bude što strmija, čime se gubici smjanjuju, pa je ukupna greška hipoteze manja **.**

PITANJE: Da li strmost odnosno nagib hipoteze zavisi od težinskog vektora $\mathbf{w}$?

PITANJE: Ukoliko l_2 norma vektora $\mathbf{w}$ raste, da li se dobija strmija sigmoida?

PITANJE: Da li se model logističke regresije previše prilagodio šumu?

L2 REGULARIZACIJA LOGISTIČKA REGRESIJE

L2 regularizacija logističke regresije može se ovako izvesti:

$$E_R(\boldsymbol{w}|D)=\sum_{i=1}^{N}\left(y^i \ln h\left(\mathbf{x}^i\right) - \left(1-y^i\right)\ln\left(1-h\left(\mathbf{x}^i\right)\right)\right) + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$$

Sada se ažuriranje težinskog vektora $\mathbf{w}$ ovako izvodi:

$$\mathbf{w} = \mathbf{w} - \eta\sum_{i=1}^{N}\nabla L\left(y^i, h\left(\mathbf{x}^i\right)\right) = \mathbf{w} - \eta\left(\sum_{i=1}^{N}\left(h\left(\mathbf{x}^i\right) - y^i\right)\mathbf{x}^i + \lambda\mathbf{w}\right) =$$

$$= \mathbf{w}(1+\eta\lambda) - \eta\sum_{i=1}^{N}\left(h\left(\mathbf{x}^i\right) - y^i\right)\mathbf{x}^i$$

pri čemu se izraz $1+\eta\lambda$ zove prigušenje težina.

PSEUDO KOD ALGORITMA GRADIJENTNOG SPUSTA U SLUČAJU REGULARIZIRANE LOGISTIČKA REGRESIJE

L2-regularizirana logistička regresija (gradijentni spust)

$\tilde{\mathbf{w}} \leftarrow (0, 0, \ldots, 0)$
ponavljaj do konvergencije
  $\Delta w_0 \leftarrow 0$
  $\Delta \mathbf{w} \leftarrow (0, 0, \ldots, 0)$
  za $i = 1, \ldots, N$
    $h \leftarrow \sigma(\tilde{\mathbf{w}}^{\mathrm{T}} \tilde{\mathbf{x}}^{(i)})$
    $\Delta w_0 \leftarrow \Delta \mathbf{w}_0 - (h - y^{(i)})$
    $\Delta \mathbf{w} \leftarrow \Delta \mathbf{w} - (h - y^{(i)}) \mathbf{x}^{(i)}$
  $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru $\Delta \tilde{\mathbf{w}}$
  $w_0 \leftarrow w_0 + \eta \Delta w_0$
  $\mathbf{w} \leftarrow \mathbf{w}(1 - \eta\lambda) + \eta \Delta \mathbf{w}$

PSEUDO KOD ALGORITMA STOHASTIČKOG GRADIJENTNOG SPUSTA U SLUČAJU REGULARIZIRANE LOGISTIČKA REGRESIJE

L2-regularizirana logistička regresija (stohastički gradijentni spust)

$\tilde{\mathbf{w}} \leftarrow (0, 0, \ldots, 0)$
ponavljaj do konvergencije:
  slučajno permutiraj primjere u $\mathcal{D}$
  za $i = 1, \ldots, N$
    $h \leftarrow \sigma(\tilde{\mathbf{w}}^{\mathrm{T}}\tilde{\mathbf{x}}^{(i)})$
    $\Delta w_0 \leftarrow -(h - y^{(i)})$
    $\Delta \mathbf{w} \leftarrow -(h - y^{(i)})\mathbf{x}$
    $\eta \leftarrow$ optimum linijskim pretraživanjem u smjeru $\Delta\tilde{\mathbf{w}}$
    $w_0 \leftarrow w_0 + \eta \Delta w_0$
    $\mathbf{w} \leftarrow \mathbf{w}(1 - \eta\lambda) + \eta\Delta\mathbf{w}$

HVALA NA PAŽNJI ☺