Supervised learning
"učenie s učiteľom"
1. časť

Zuzana Rošťáková

4. november 2021

Seminár UM

Učenie s učiteľom - supervised learning

- vstupné údaje:
  - matica X o rozmere $n \times p$
  - **n** pozorovaných objektov
  - pre i-ty objekt pozorujeme **p** premenných v tzv. vektore čŕt

$$\mathbf{x}_i = \left(x_{i_\mathbf{1}}, x_{i_\mathbf{2}}, \ldots, x_{i_p}\right)^T$$

  - premenné môžu byť
    - kardinálne (číselné) - napr. vek, výška, krvný tlak, ...
    - kategorické - napr. krvná skupina, pohlavie, vzdelanie, ...

- výstupné údaje:
  - kardinálne - reálne čísla $\rightarrow y_1, \ldots, y_n$
    $\rightarrow$ **regresná analýza**
  - kategorické - označenie skupiny/triedy $\rightarrow c_1, c_2, \ldots, c_n$
    $\rightarrow$ **klasifikačná analýza**

Klasifikačná analýza

Klasifikácia

- manuálne zaraďovanie prvkov do tried $\rightarrow$ drahé, časovo náročné
- **cieľ:** natrénovanie modelu (klasifikátora) pomocou n pozorovaní

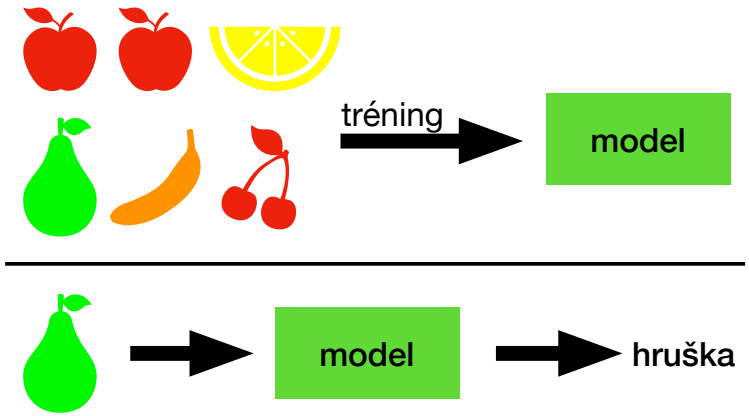
$$\mathbf{x_1}, \ldots, \mathbf{x_n} \in \mathbb{R}^p$$

so známym zaradením do L tried (skupín) $S_1, \ldots, S_L$

$$c_1, \ldots, c_n \in \{1, \ldots, L\}$$

$$S_l = \{\mathbf{x_i} : c_i = l\}, \quad l = 1, \ldots, L$$

a následné zaradenie nového pozorovania $\mathbf{x}^\star \in \mathbb{R}^p$ do jednej z tried



Klasifikácia - prehľad metód

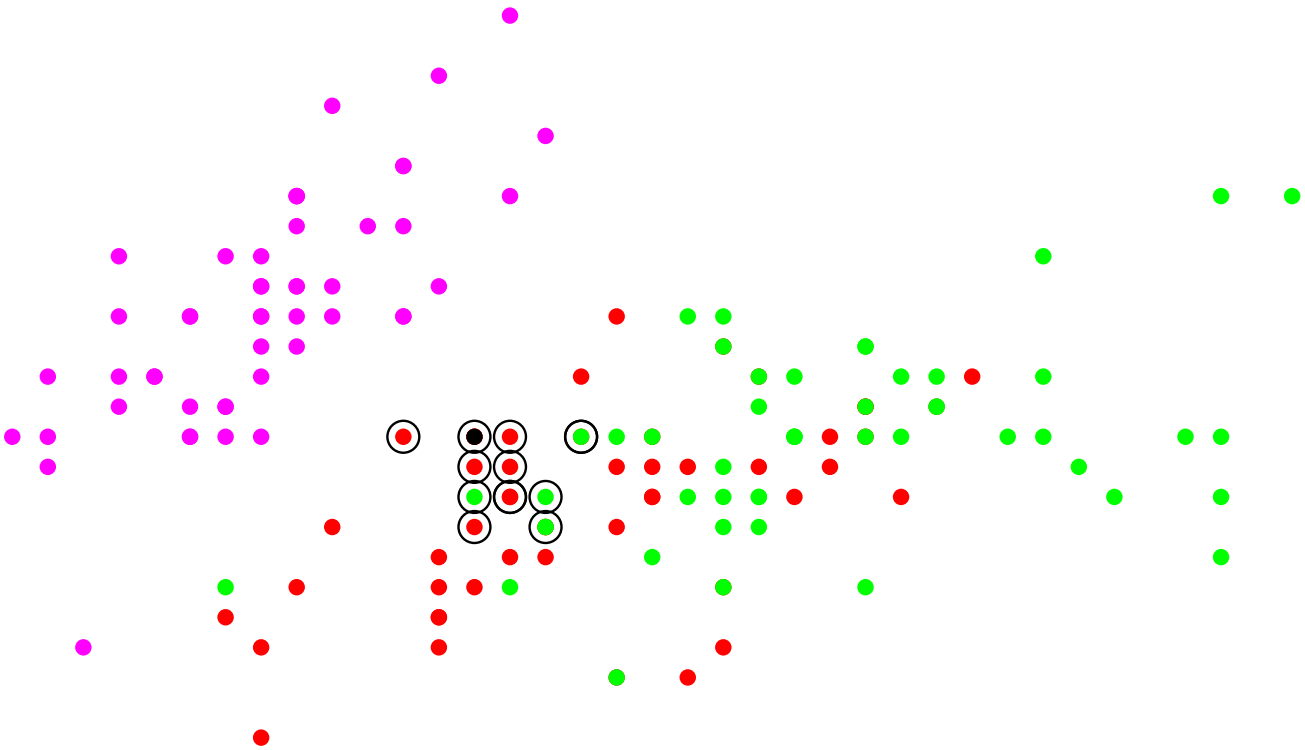
- k-najbližších susedov (k-nearest neighbours)
- klasifikačné stromy
- diskriminačná analýza - lineárna, kvadratická
- naivná Bayesovská klasifikácia (naïve Bayes)
- logistická regresia
- metóda nosného bodu (support vector machine)
- neurónové siete
- ...

k-najbližších susedov

k-nearest neighbours (KNN)

k-najbližších susedov

- trénovacia vzorka - $\mathbf{x_1}, \ldots, \mathbf{x_n} \in \mathbb{R}^p$ zaradené do L tried
- nové pozorovanie $x^\star$
  - nájdeme jeho $k < n$ najbližších susedov z trénovacej vzorky $\rightarrow$ $\mathbf{x_{i1}}, \ldots, \mathbf{x_{ik}}$
  - $c^\star = mode(\{c_{i1}, c_{i2}, \ldots, c_{ik}\}) \rightarrow$ najčastejšie sa vyskytujúca trieda



k-najbližších susedov v MATLAB-e

Mdl = fitcknn(X,C,Name,Value)

X - $n \times p$ matica dát

C - $n \times 1$ vektor, zaradenie n pozorovaní do tried

- voliteľné parametre
  - CategoricalPredictors - buď 'all' alebo [ ] (všetky sú kardinálne)
  - NumNeighbors - počet najbližších susedov, default 1
  - Distance - euclidean, cityblock, chebyshev, minkowski,...
    - ak sú premenné kategorické, **Distance = 'hamming'** alebo **'jaccard'**
  - NSMethod - ako hľadať najbližších susedov
    - 'exhaustive' - vzdialenosť x^* od všetkých n pozorovaní v $X \rightarrow$ časovo náročné
    - 'kdtree' $\rightarrow$ Distance = euclidean, cityblock, minkowski, chebyshev. Binárny strom, rýchlejšie.
  - BreakTies - smalest, nearest, random - v k susedoch má viac tried rovnaké zastúpenie
  - IncludeTies - true (presne k najbližích susedov), false (susedov je viac ako k, ale rôznych vzdialeností je k)

Mdl - natrénovaný model

k-najbližších susedov v MATLAB-e - predikcia

$C^\star$ = **predict(Mdl, Xnew)**

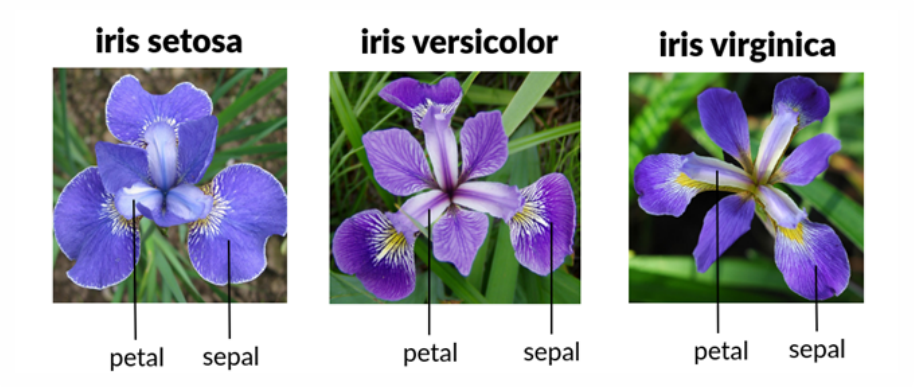
Mdl - natrénovaný model / klasifikátor (výstup funkcie napr. *fitcknn*)

Xnew - $n \times p$ matica nových pozorovaní

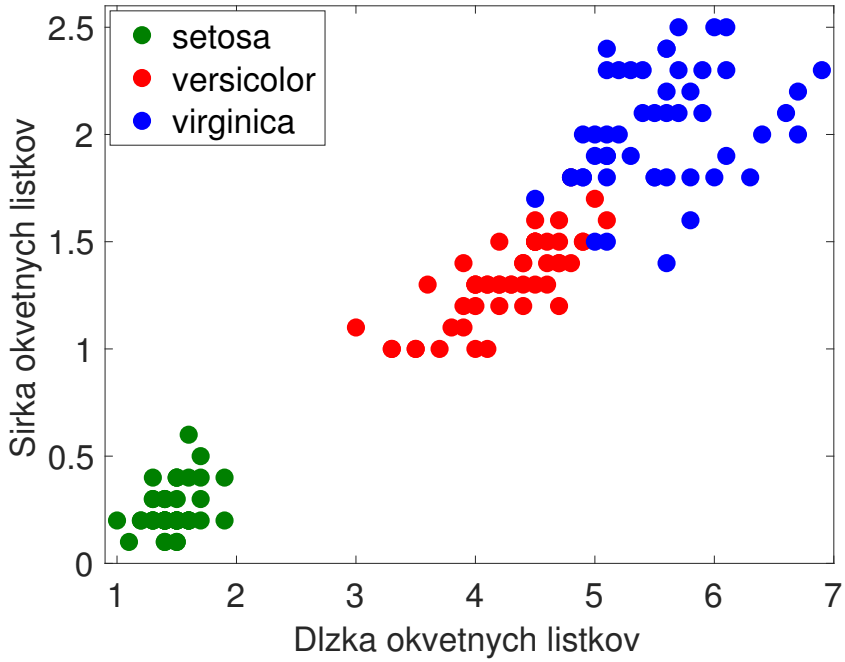
$C^\star$ - vektor zaradenia nových pozorovaní do tried

k-najbližších susedov - Príklad: Kosatce

- databáza **Fisher Iris** v MATLAB-e
- $n = 150$ meraní dĺžky a šírky okvetného lístka kosatcov ($p = 2$)
- **cieľ**: pomocou parametrov okvetného lístka určiť druh kosatca

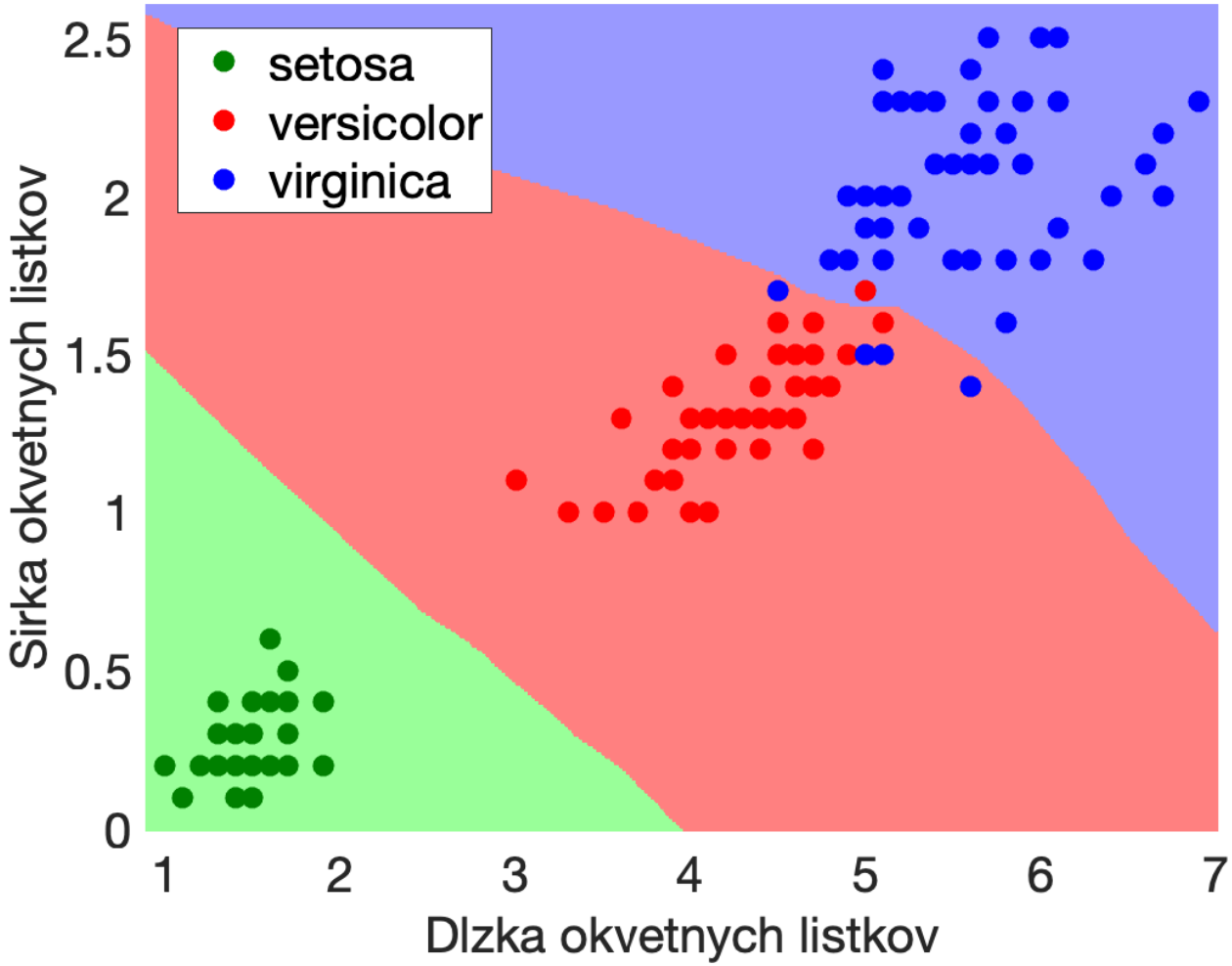


https://medium.com/@Nivitus./iris-flower-classification-machine-learning-d4e337140fa4



k-najbližších susedov - Príklad: Kosatce

- mdl = fitcknn(X,C,'NumNeibors',5,'Distance','seuclidean')
- $C^\star$ = predict(mdl,Xnew)



k-najbližších susedov - Príklad: Kosatce

Ako nastaviť vhodný počet susedov a typ vzdialenosti?

- mdlknn = fitcknn(X,C, 'OptimizeHyperparameters','auto',...
  'HyperparameterOptimizationOptions',...
  struct('AcquisitionFunctionName','expected-improvement-plus'))

→ optimálny počet susedov - NumNeighbors = 5

→ vzdialenosť - Distance = seuclidean

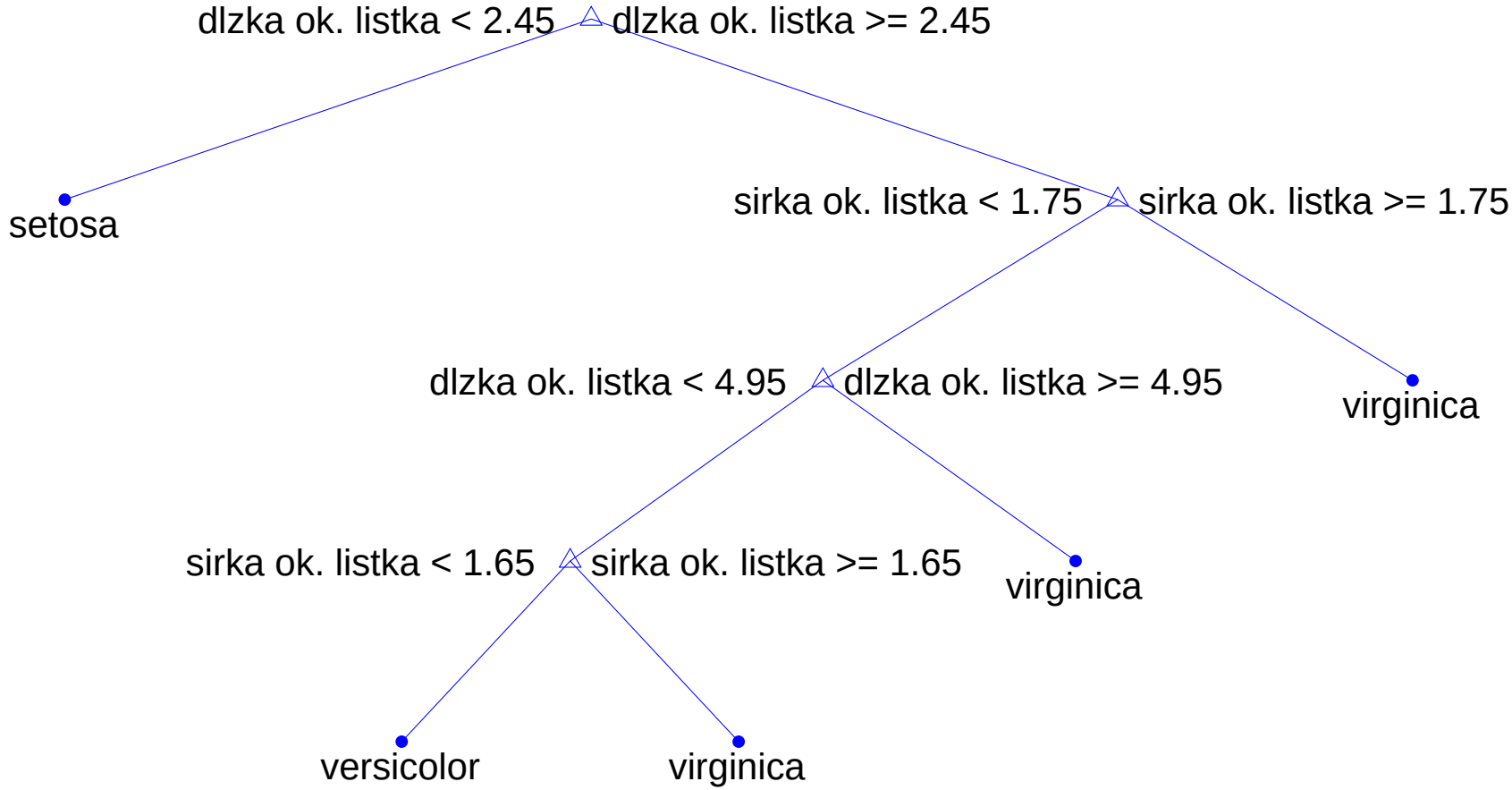
k-najbližších susedov - zhrnutie

- **výhody:**
  - jednoduchý algoritmus (implementácia, pochopenie)
  - priamočiara interpetácia
  - MATLAB-ovská verzia vie pracovať aj s kategorickými premennými
- **nevýhody:**
  - časovo náročné vyhľadávanie susedov, keď n je veľké
  - rôzne výsledky pre rôzne počty susedov k a rôzne typy vzdialeností
    - ALE vieme nájsť optimálny počet susedov, resp. vhodný typ vzdialenosti pre dáta
  - V MATLAB-ovskej verzií musia byť buď všetky premenné kardinálne alebo všetky kategorické

Klasifikačný strom

Classification tree (Ctree)

Klasifikačný strom



- koncový uzol - uzol, ktorý nemá potomkov
- nekoncový uzol
  - je rodičom presne dvoch potomkov
  - priradená podmienka $\rightarrow$ určí vetvu, ktorou sa ďalej vydáme

Klasifikačný strom v MATLAB-e

Mdl = fitctree(X,C,Name,Value)

X - $n \times p$ matica dát

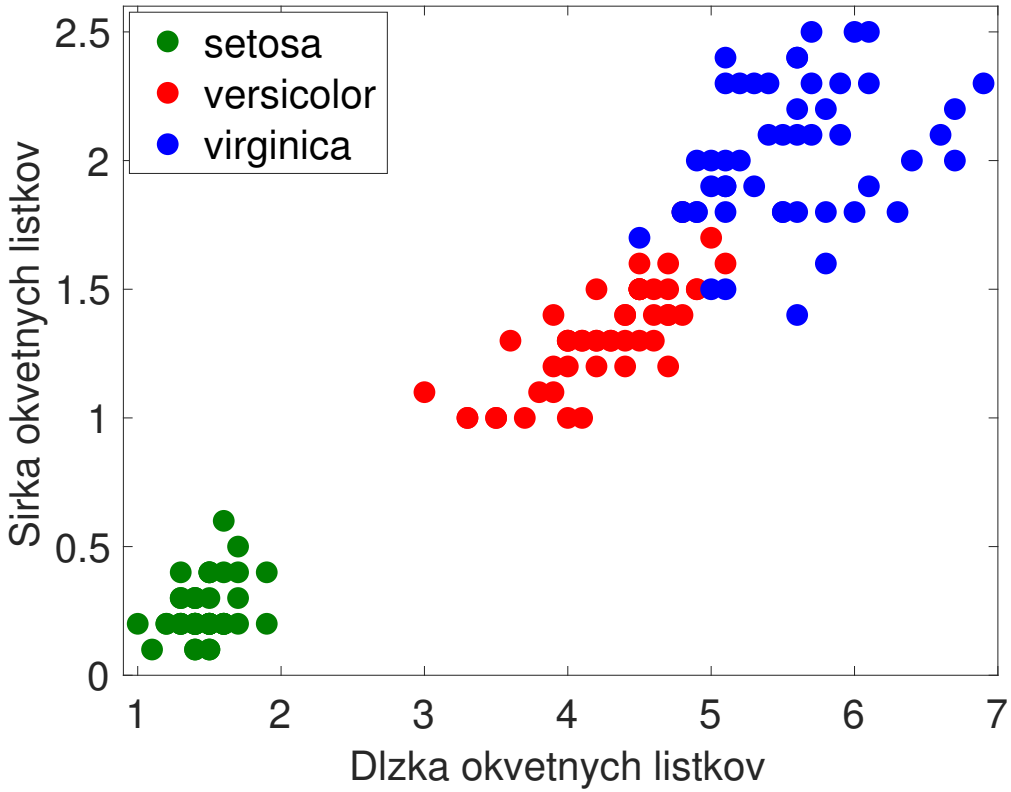
C - $n \times 1$ vektor, zaradenie n pozorovaní do tried

- voliteľné parametre
  - AlgorithmForCategorical - Exact, PullLeft, PCA, OVAbyClass
  - CategoricalPredictors - zoznam kategorických premenných
  - MaxDepth - maximálny počet úrovní stromu
  - MaxNumSplits - maximálny počet nekoncových uzlov
  - Prune - výstupom je najlepší orezaný strom (on) alebo kompletný strom (off)
  - PruneCriterion - error, impurity $\rightarrow$ kritérium na orezávanie

Mdl - natrénovaný model

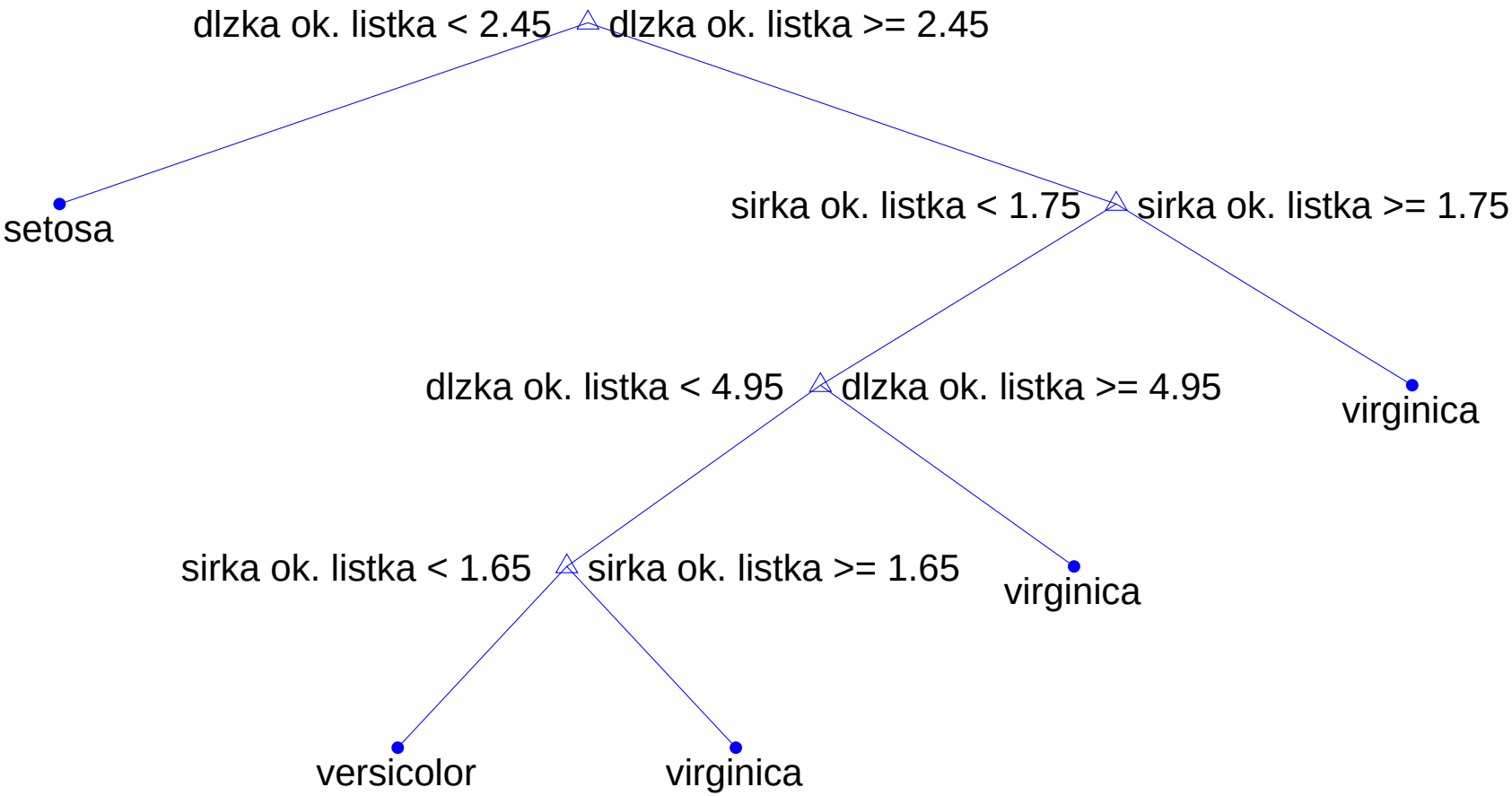
Klasifikačný strom - Príklad: Kosatce

- databáza **Fisher Iris** v MATLAB-e
- $n = 150$ meraní dĺžky a šírky okvetného lístka kosatcov ($p = 2$)
- 3 triedy kosatcov - versicolor, virginica, setosa

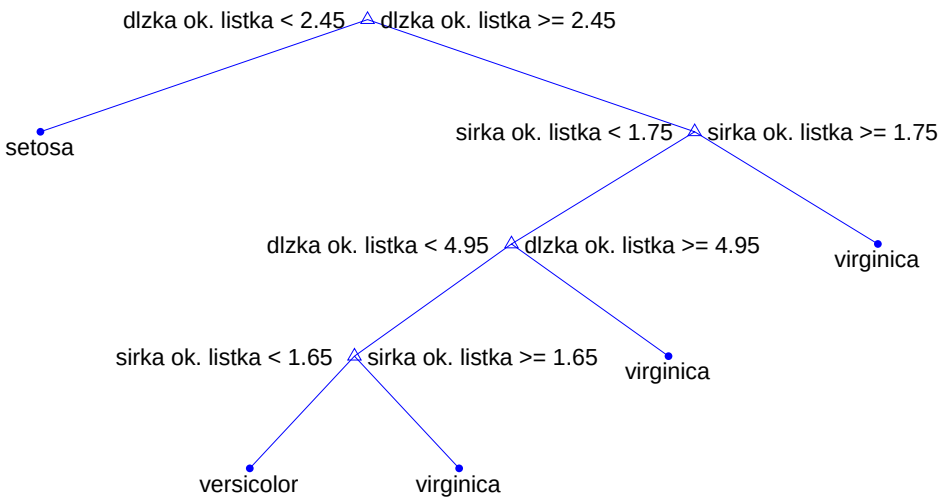


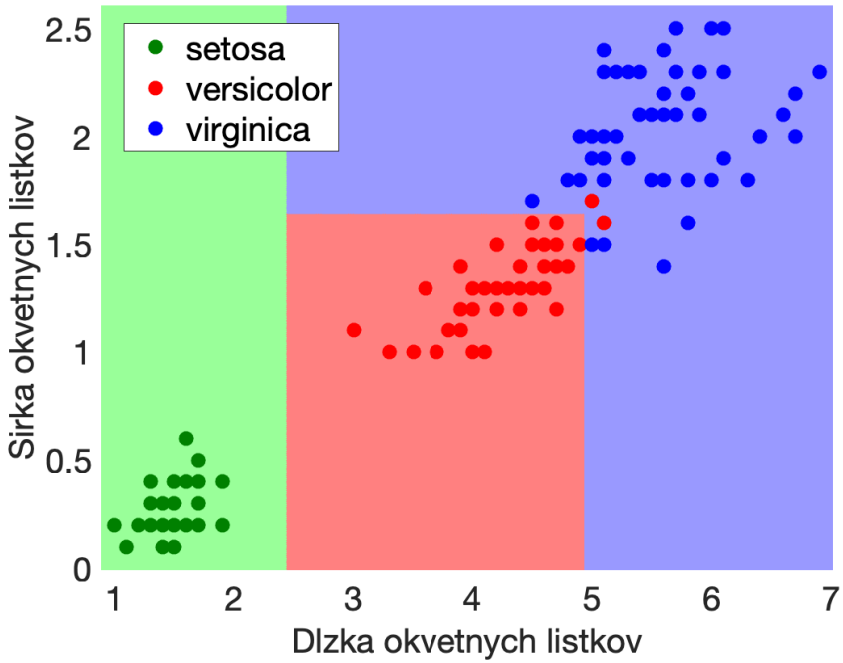
Klasifikačný strom - Príklad: Kosatce

- mdltree = fitctree(X,C,'CategoricalPredictors',[ ])



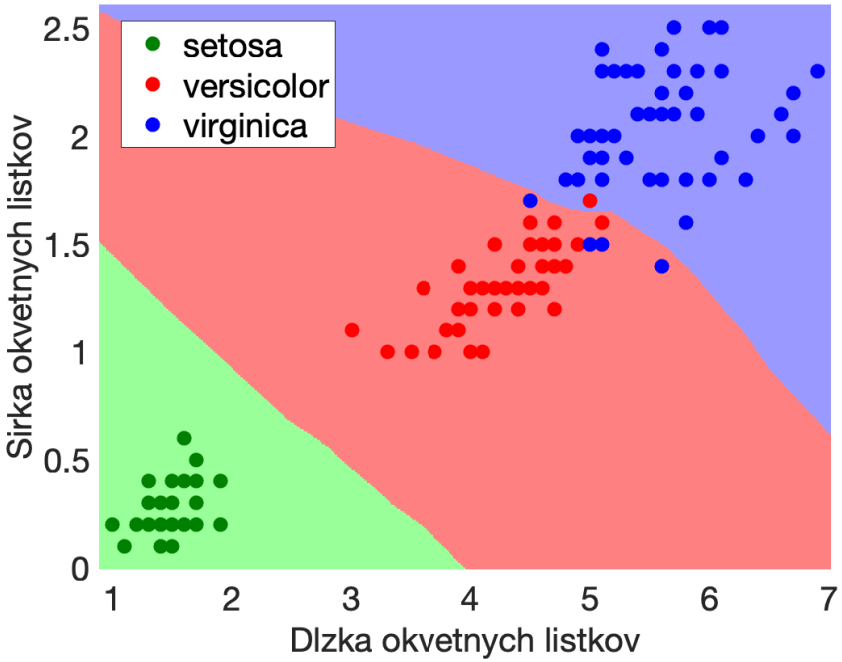
Klasifikačný strom - Príklad: Kosatce



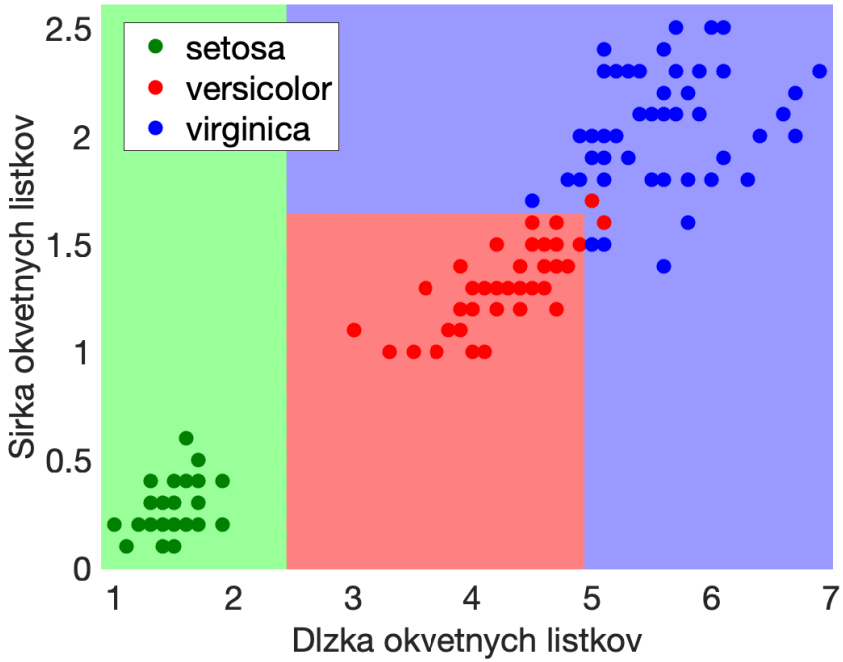


Klasifikačné stromy - Príklad: Kosatce

KNN



Ctree



Klasifikačný strom - Príklad: Cukrovka

- dáta dostupné na

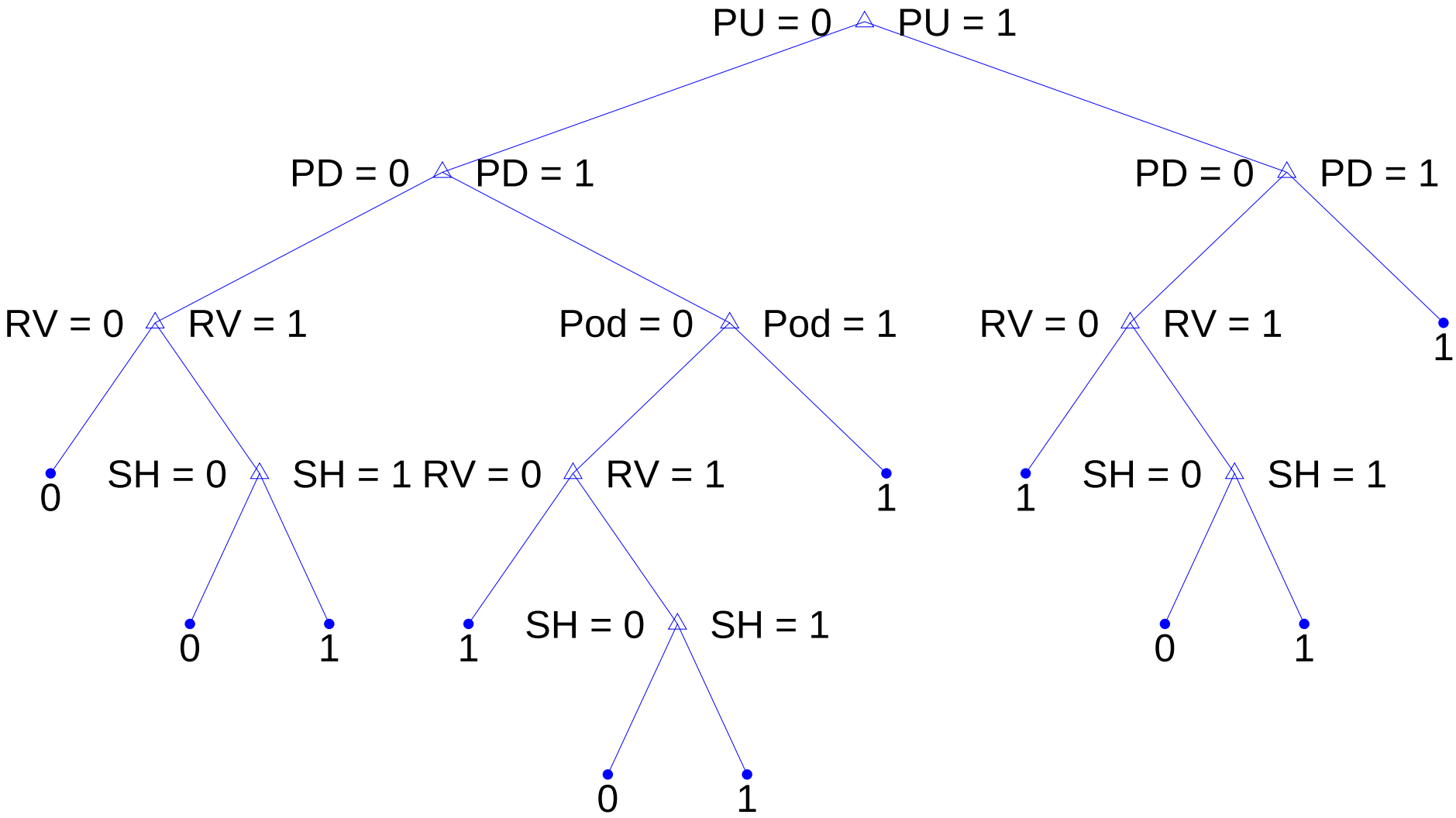
  https://archive.ics.uci.edu/ml/datasets/Early+stage+diabetes+risk+prediction+dataset.

  Islam, M.M. Faniqul, et al. 'Likelihood prediction of diabetes at early stage using data mining techniques.' Computer Vision and Machine Intelligence in Medical Image Analysis. Springer, Singapore, 2020. 113-125.

- 520 pozorovaní - 328 mužov, 192 žien, priemerný vek 48.03 ± 12.15
- 5 kategorických (Y/N) premenných
  - polyuria (PU) - nadmerné vylučovanie moču
  - polydipsia (PD) - nadmerný smäd
  - náhla strata hmotnosti (SH)
  - rozmazané videnie (RV)
  - podráždenosť (Pod)
- skoré štádium cukrovky - dve triedy 0 (negatívny) a 1 (pozitívny)

Klasifikačný strom - Príklad: Cukrovka

- mdltree1 = fitctree(X,C,'CategoricalPredictors',1:size(X,2))



Klasifikačný strom zhrnutie

- **výhody:**
  - jednoduchá interpretácia
  - možné grafické znázornenie v prípade nižšieho počtu premenných
  - vie pracovať s kardinálnymi aj s kategorickými premennými
- **nevýhody:**
  - neprehľadnosť v prípade väčšieho počtu premenných
    - ťažšia interpretácia
    - nutné orezávanie $\rightarrow$ strata presnosti / kvality klasifikátora

Lineárna a kvadratická diskriminačná analýza

Linear and quadratic discriminant analysis

(LDA and QDA)

Lineárna diskriminačná analýza (LDA)

- uvažujme len dve triedy (0 a 1) (funguje aj pre viac tried)
- **predpoklad:**

$$\mathbf{x}|c = 0 \sim \mathcal{N}_p(\mu_0, \Sigma) \text{ a } \mathbf{x}|c = 1 \sim \mathcal{N}_p(\mu_1, \Sigma)$$

→ $\mathbf{x}$ musia byť reálne vektory (predpoklad normality)

→ triedy sa líšia len v charakteristike strednej hodnoty, nie v kovariančnej matici

- **trénovanie** = odhad charakteristík tried μ_0, μ_1, Σ
- zaradenie nového pozorovania $\mathbf{x}^\star$

$$c^\star = \begin{cases} 1, & \text{ak } (\mu_1 - \mu_0)^T \Sigma^{-1} \mathbf{x}^\star > T + \frac{1}{2}(\mu_1 - \mu_0)^T \Sigma^{-1}(\mu_1 + \mu_0) \\ & \mathbf{w}^T \mathbf{x}^\star > \theta \\ 0, & \text{inak} \end{cases}$$

→ ide o lineárny klasifikátor

Kvadratická diskriminačná analýza (QDA)

- uvažujme len dve triedy (0 a 1) (funguje aj pre viac tried)
- **predpoklad:**

$$\mathbf{x}|c = 0 \sim \mathcal{N}_p(\mu_0, \Sigma_0) \text{ a } \mathbf{x}|c = 1 \sim \mathcal{N}_p(\mu_1, \Sigma_1)$$

$\rightarrow$ $\mathbf{x}$ musia byť reálne vektory (predpoklad normality)

- **trénovanie** = odhad charakteristík tried μ_0, Σ_0, μ_1, Σ_1
- zaradenie nového pozorovania $\mathbf{x}^\star$

$$c^\star = \begin{cases} 1, & \text{ak } Q(x^\star) > T, \\ 0, & \text{inak} \end{cases}$$

$$\begin{aligned} Q(x^\star) &= (x^\star - \mu_0)^T \Sigma_0^{-1} (x^\star - \mu_0) + \log|\Sigma_0| - \\ &\quad - (x^\star - \mu_1)^T \Sigma_1^{-1} (x^\star - \mu_1) - \log|\Sigma_1| \\ &\approx \mathbf{x}^{\star T} \Omega \mathbf{x}^\star + \mathbf{w}^T \mathbf{x}^\star + \delta \end{aligned}$$

LDA a QDA v MATLAB-e

Mdl = fitcdiscr(X,C,Name,Value)

X - $n \times p$ matica pozorovaní

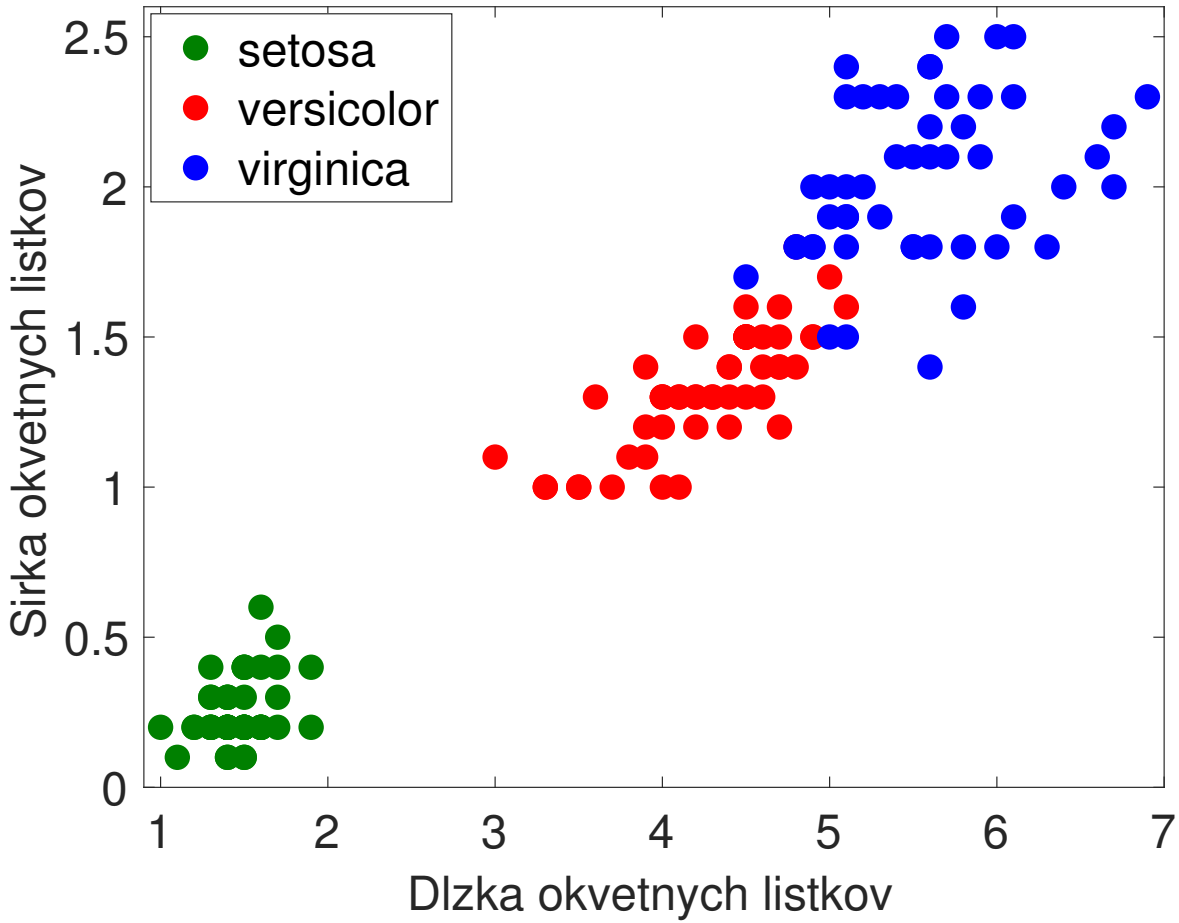
C - vektor $n \times 1$, c_i je zaradenie / označenie triedy i-teho pozorovania

- voliteľné parametre
  - DiscrimType - linear, diaglinear, quadratic, diagquadratic, ...
  - Prior - apriórne pravdepodob. tried ("empirical", "uniform", vektor)
  - ...

Mdl - natrénovaný model, MATLAB-ovská štruktúra

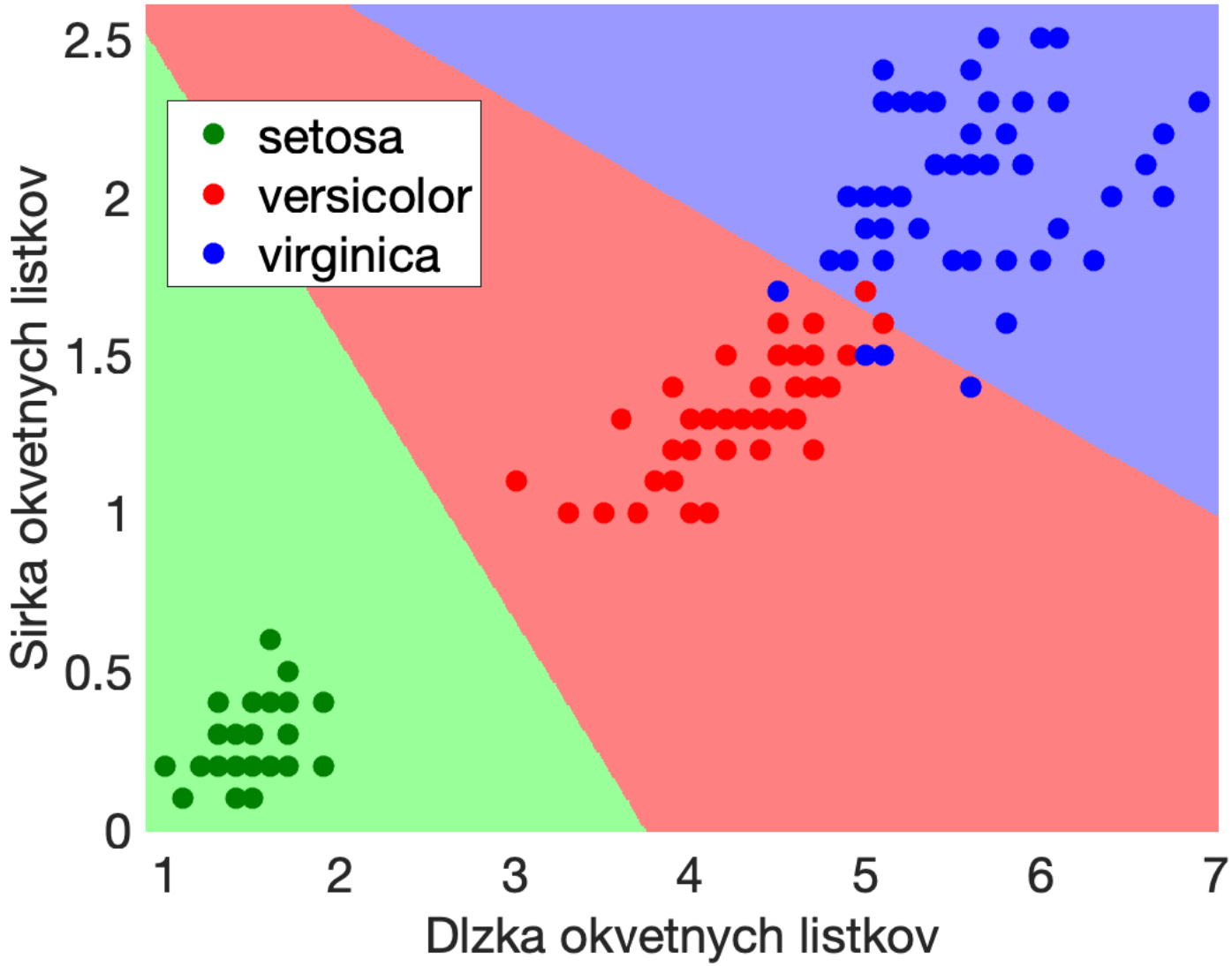
LDA a QDA - Príklad: Kosatce

- databáza **Fisher Iris** v MATLAB-e
- $n = 150$ meraní dĺžky a šírky okvetného lístka kosatcov ($p = 2$)
- 3 triedy kosatcov - versicolor, virginica, setosa



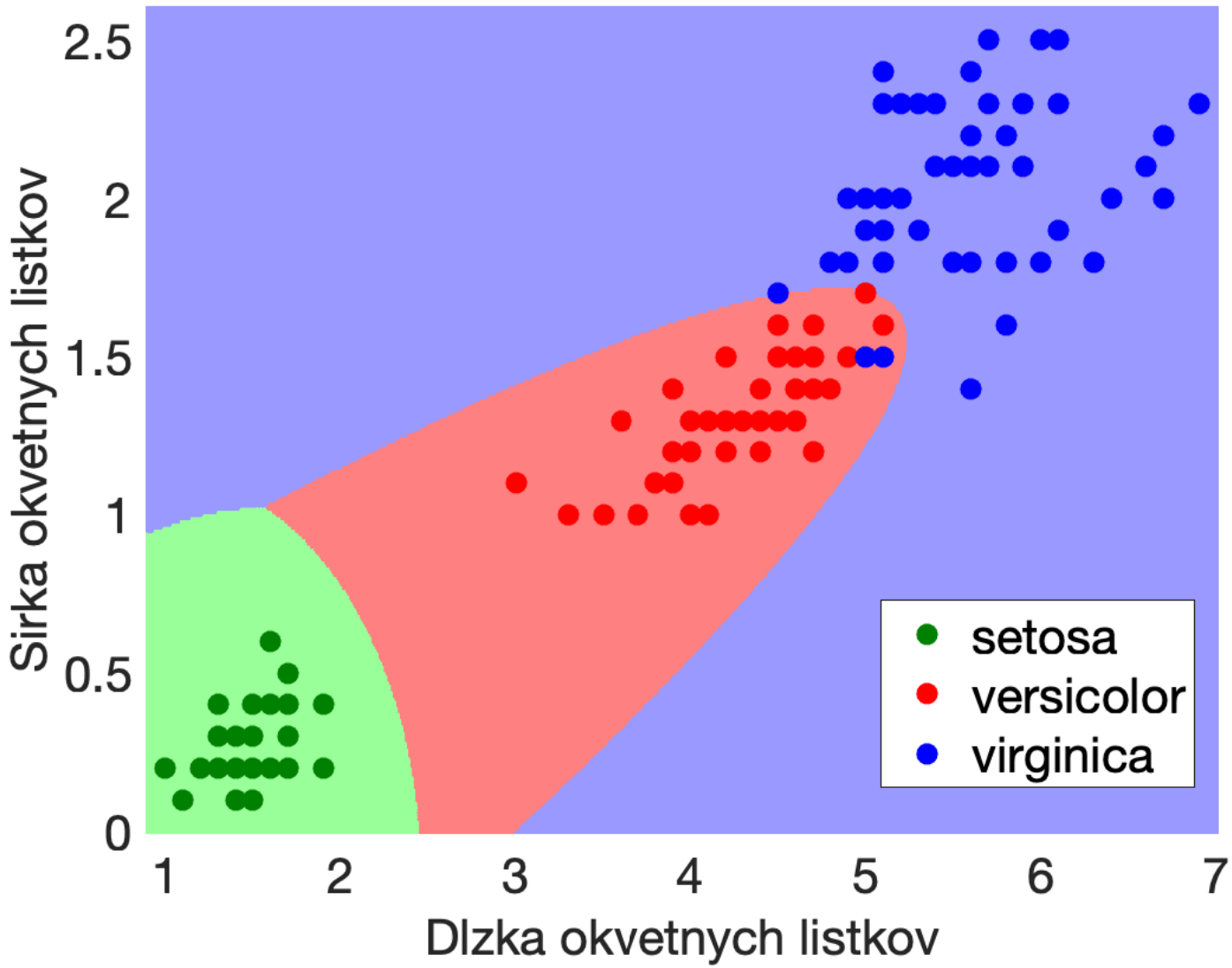
LDA - Príklad: Kosatce

$$MdlLinear = fitcdiscr(X, C)$$



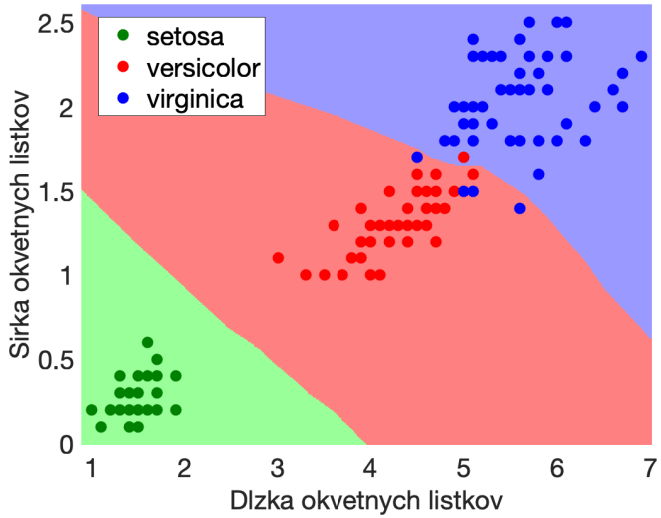
QDA - Príklad: Kosatce

$$MdlQuad = fitcdiscr(X, C, 'DiscrimType', 'quadratic')$$

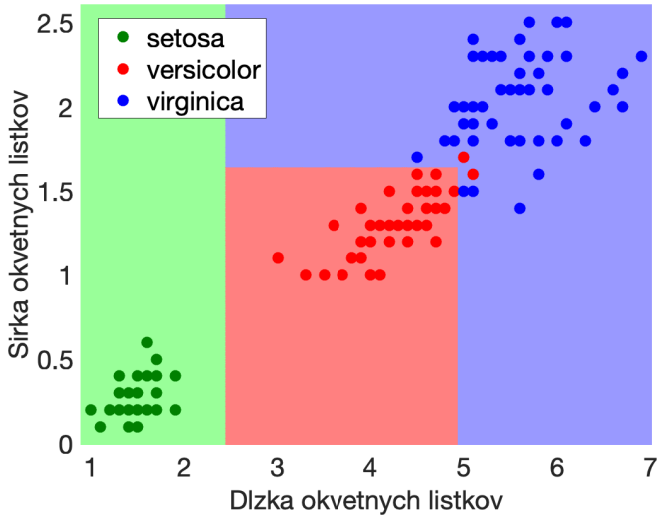


LDA a QDA - Príklad: Kosatce

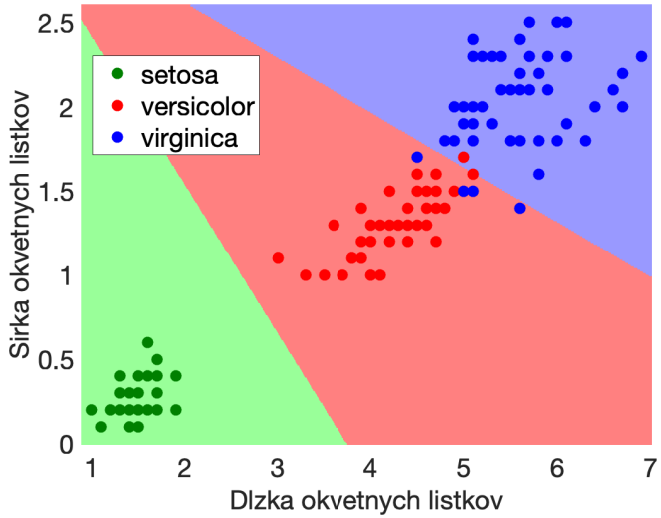
KNN



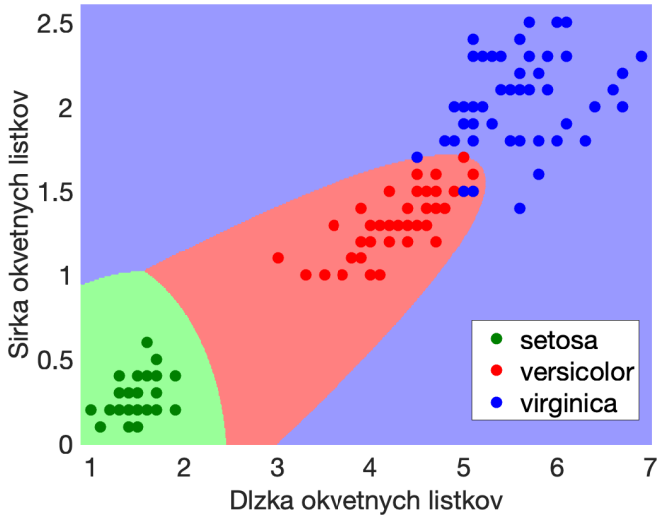
Ctree



LDA



QDA



LDA a QDA zhrnutie

- **výhody:**
  - jednoduché na pochopenie a implementáciu
  - jednoduchá interpretácia klasifikátora
  - pomerne rýchle natrénovanie klasifikátora
- **nevýhody:**
  - vektor čŕt nemôže obsahovať kategorické premenné
  - predpoklad normality premenných (ale môže byť porušený)
  - pri LDA nutný predpoklad lineárnej separovateľnosti tried

Bayesovský klasifikátor

Naive Bayes classification (NBC)

Bayesovský klasifikátor (NBC)

- n pozorovaní zaradených do L skupín $S_1, \ldots, S_L$
- premenné vo vektore čŕt $\mathbf{x} = (x_1, \ldots, x_p)^T$ sú navzájom nezávislé
  $\rightarrow$ silný predpoklad

$$P(S_l|\mathbf{x}) = \frac{P(S_l)P(x_1, \ldots, x_p|S_l)}{P(x_1, \ldots, x_p)} \propto P(S_l) \prod_{i=1}^{p} P(x_i|S_l)$$

  - $P(S_l)$ - apriórna pravdepod. triedy S_l ($\frac{1}{L}$ pre rovnocenné triedy)
  - $P(x|S_l)$ - rozdelenie pravdepod. x za podmienky zaradenia do S_l
    - **normálne** rozdelenie $\rightarrow$ spojité premenné
    - **multinomické, Bernoulliho** rozdelenie $\rightarrow$ diskrétne premenné

- rozhodovacie pravidlo

$$c^\star = \text{argmax}_{l=1,\ldots,L} P(S_l) \prod_{i=1}^{p} P(x_i|S_l)$$

Bayesovský klasifikátor v MATLAB-e

Mdl = fitcnb(X,C,Name,Value)

X - $n \times p$ matica pozorovaní

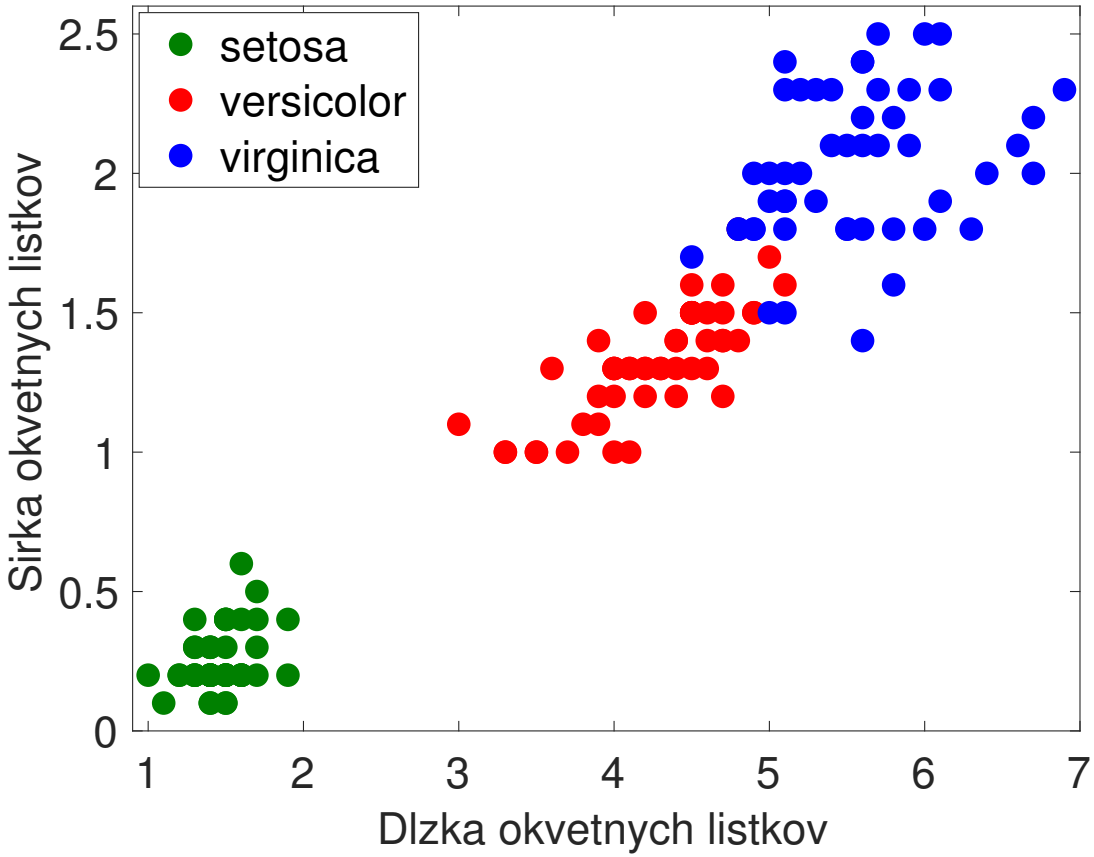
C - $n \times 1$ vektor zaradenia do tried

- voliteľné parametre
  - DistributionNames - normal, mn (multinomial), mvmn, kernel
    $\rightarrow$ pre každú premennú môže byť iné rozdelenie pravdepodobnosti
  - CategoricalPredictors - zoznam kategorických premenných
    $\rightarrow$ automaticky priradené rozdelenie *mvmn*
  - ...

Mdl - natrénovaný klasifikátor

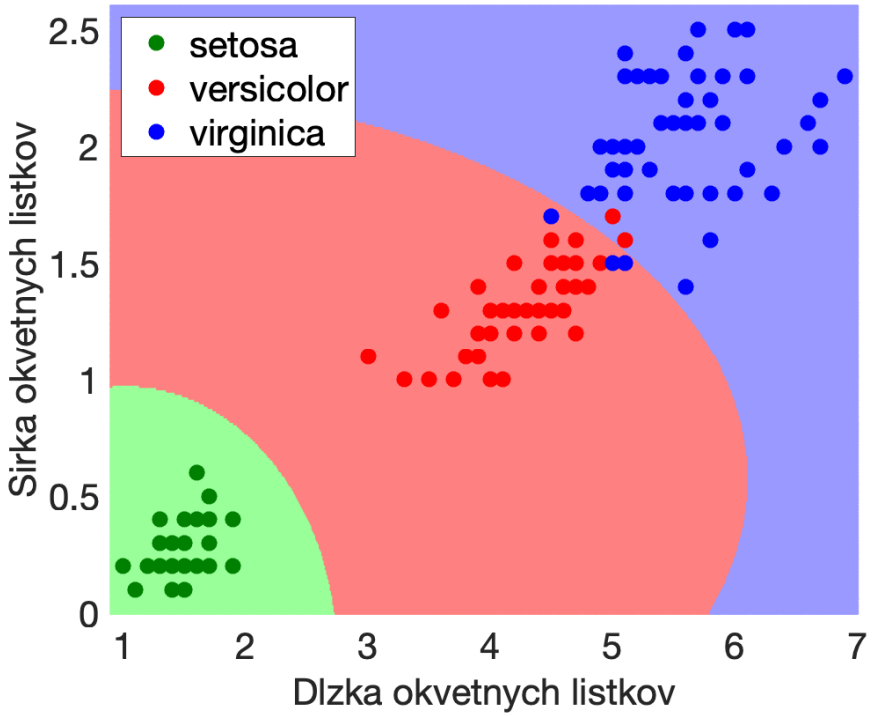
Bayesovský klasifikátor - Príklad: Kosatce

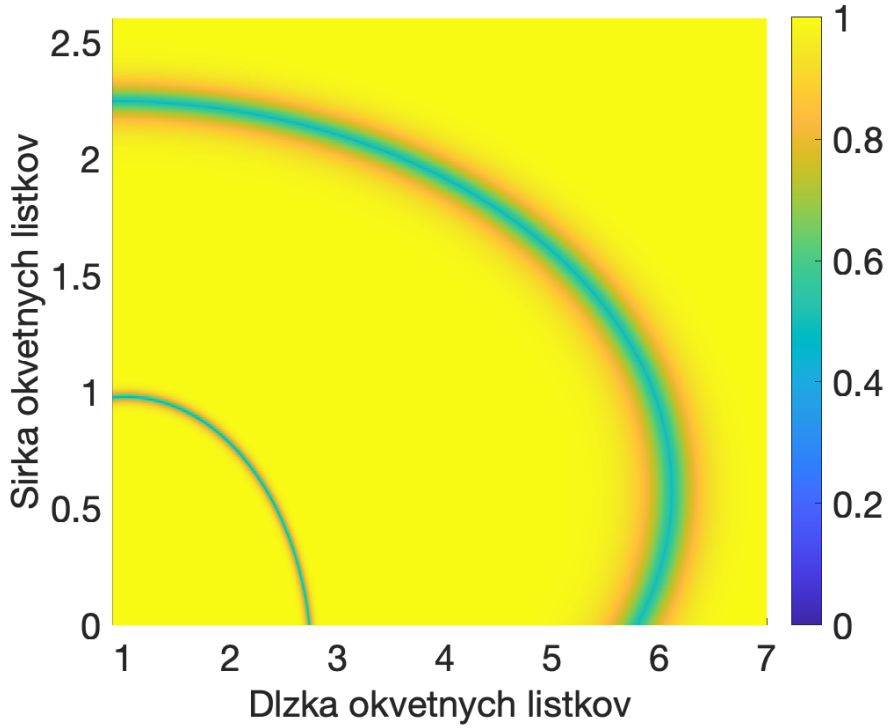
- databáza **Fisher Iris** v MATLAB-e
- 150 pozorovaní, 3 triedy - versicolor, virginica, setosa
- premenné - dĺžka a šírka **okvetného** lístka
  $\rightarrow$ premenné (opticky aj logicky) nie sú úplne nezávislé



Bayesovský klasifikátor - Príklad: Kosatce

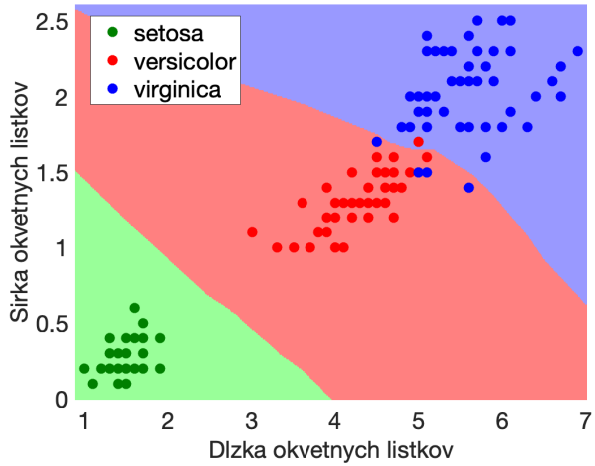
- $mdl = fitcnb(X, C)$
  $\rightarrow$ normálne rozdelenie je predvolené, netreba ho nastavovať



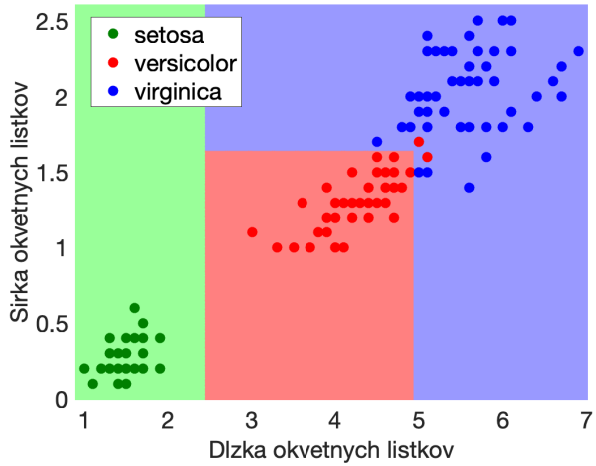


Bayesovský klasifikátor - Príklad: Kosatce

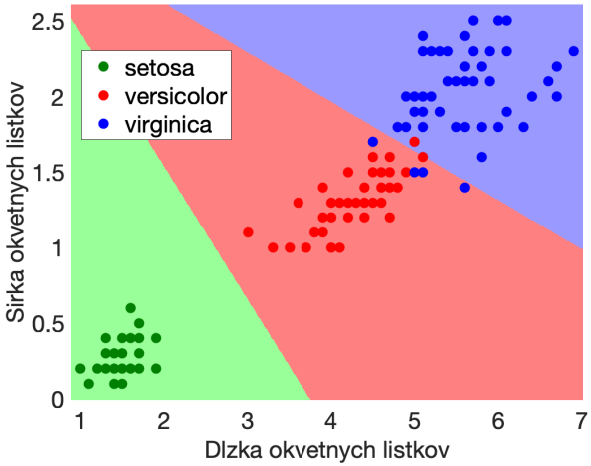
KNN



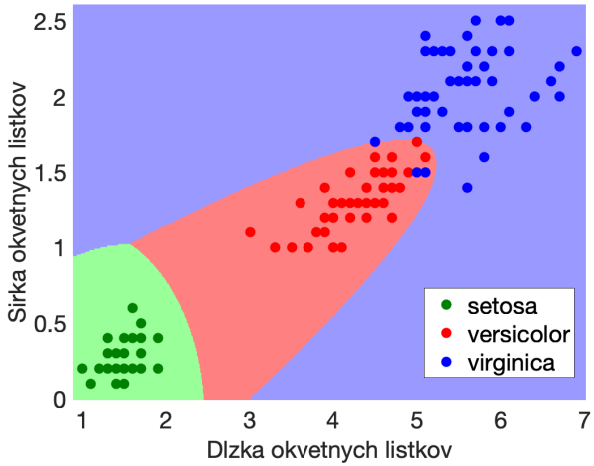
Ctree



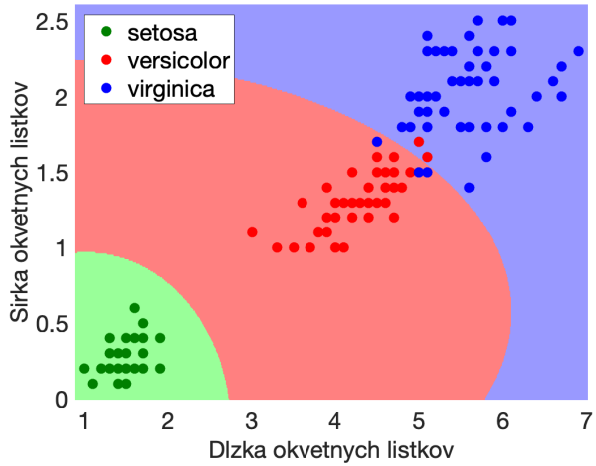
LDA



QDA



NBC



Bayesovský klasifikátor zhrnutie

- **výhody:**
  - vie pracovať aj s kategorickými premennými
    - → MATLAB im automaticky priradí viacrozmerné multinomické rozdelenie (mvmn)
  - predpoklad nezávislosti premenných, ale funguje dobre aj v prípade, že je tento predpoklad čiastočne porušený
- **nevýhody:**
  - náročnejšie na pochopenie a implementáciu v porovnaní s predchádzajúcimi metódami

Logistická regresia

Logistic regression (LogReg)

Úvod do logistickej regresie

- Príklad: infarkty (inšpirované prednáškou J. Somorčíka z FMFI UK)
  - nezávislé premenné:
    - **vek** - v porovnaní s hodnotou 50
      (t.j. -2 znamená 48-ročného človeka)
    - **cholesterol** - v porovnaní s hodnotou 5
      (t.j. -1 znamená cholesterol na úrovni 4)
    - **pohlavie** - 0 = muž, 1 = žena
  - závislá premenná - čo sa stalo s osobou 10 rokov od experimentu
    - 0 = nezomrel na infarkt
    - 1 = zomrel na infarkt

Úvod do logistickej regresie

- **model 1 - (klasická) lineárna regresia**

$$y = \beta_0 + \beta_1 x_{vek} + \beta_2 x_{cholesterol} + \beta_3 x_{pohlavie} + \varepsilon$$

- **PROBLÉM:**
  - $\varepsilon \sim \mathcal{N}(0, \sigma^2) \rightarrow y$ by bola spojitá premenná, ale my máme diskrétnu

$\rightarrow$ zlý model

Úvod do logistickej regresie

- **model 2 - zovšeobecnený lineárny model (GLM)**

$$y \sim Bin(1, p_x)$$

p_x = pravdepodobnosť úmrtia na infarkt do 10 rokov

$$p_x = \beta_0 + \beta_1 x_{vek} + \beta_2 x_{cholesterol} + \beta_3 x_{pohlavie} + \varepsilon$$

- **PROBLÉM:**
  - $p_x \in [0, 1]$
  - ALE $\beta_0 + \beta_1 x_{vek} + \beta_2 x_{cholesterol} + \beta_3 x_{pohlavie} \in (-\infty, \infty)$

$\rightarrow$ zlý model

Úvod do logistickej regresie

- **model 3 - logistická regresia**

$$y \sim Bin(1, p_x)$$

p_x = pravdepodobnosť úmrtia na infarkt do 10 rokov

$$ln\frac{p_x}{1-p_x} = \beta_0 + \beta_1 x_{vek} + \beta_2 x_{cholesterol} + \beta_3 x_{pohlavie} + \varepsilon$$

$$p_x = \frac{e^{\beta_0+\beta_1 x_{vek}+\beta_2 x_{cholesterol}+\beta_3 x_{pohlavie}}}{1 + e^{\beta_0+\beta_1 x_{vek}+\beta_2 x_{cholesterol}+\beta_3 x_{pohlavie}}}$$

- $ln\frac{p_x}{1-p_x} = logit(p_x) \in (-\infty, \infty)$
- $\beta_0 + \beta_1 x_{vek} + \beta_2 x_{cholesterol} + \beta_3 x_{pohlavie} \in (-\infty, \infty)$

Logistická regresia pre dve triedy

- n pozorovaní, dve triedy - 0 a 1 (vysvetľujúca premenná je binárna)
- model logistickej regresie

$$p_{\mathbf{x}} = P(c = 1|\mathbf{x}) = \frac{e^{\beta_0 + \sum_{i=1}^{p} \beta_i x_i}}{1 + e^{\beta_0 + \sum_{i=1}^{p} \beta_i x_i}} \quad \rightarrow \quad P(c = 0|\mathbf{x}) = 1 - p_{\mathbf{x}}$$

$$logit(p_{\mathbf{x}}) = log \frac{p_x}{1 - p_x} = \beta_0 + \sum_{i=1}^{p} \beta_i x_i$$

- **trénovanie:** odhad $\beta_0, \beta_1, \ldots, \beta_p$ pomocou metódy maximálnej vierohodnosti z trénovacích dát

- nové pozorovanie $\mathbf{x}^\star \rightarrow$ vypočítame $p_{\mathbf{x}^\star}$

$$p_{\mathbf{x}^\star} = \frac{e^{\widehat{\beta_0} + \sum_{i=1}^{p} \widehat{\beta}_i x_i^\star}}{1 + e^{\widehat{\beta_0} + \sum_{i=1}^{p} \widehat{\beta}_i x_i^\star}} \in (0, 1) \quad \rightarrow \quad c^\star = \begin{cases} 1, & \text{ak } p_{\mathbf{x}^\star} \geq 0.5 \\ 0, & \text{ak } p_{\mathbf{x}^\star} < 0.5 \end{cases}$$

Logistická regresia v MATLAB-e - dve triedy

Mdl =
fitglm(X,C, 'Distribution','binomial','link','logit' ,Name,Value)

X - $n \times p$ matica pozorovaní

C - $n \times 1$ vektor zaradenia do tried

- Distribution - rozdelenie pravdepodobnosti závislej premennej, binomial $\rightarrow$ 2 triedy
- Link = logit $\rightarrow$ logistická regresia
- voliteľné parametre:
  - *CategoricalVars* - zoznam kategorických premenných, napr. $[2, 3, 5, 8]$
  - *Varnames* - názvy premenných (vysvetľujúcich + závislej), napr. $\{'x1', 'x2', 'y'\}$

Mdl - model logistickej regresie

- *mdl.Coefficients.Estimate* - odhad parametrov $\beta_0, \beta_1, \ldots, \beta_p$

Log. regresia v MATLAB-e - dve triedy, predikcia

$p^\star$ = **predict(Mdl,Xnew)**
$c^\star$ = **zeros(1,size(Xnew,1))**
$c^\star(p^\star >= 0.5) = 1$
$c^\star(p^\star < 0.5) = 0$;

Mdl - výstup funkcie *fitglm*

Xnew - matica nových pozorovaní

$p^\star$ - pravdepodobnosť zaradenia nových pozorovaní do triedy 1

$c^\star$ - zaradenie nových pozorovaní do tried

Logistická regresia pre viac tried

- n pozorovaní, L tried - 1,. . . ,L
  $\rightarrow$ jednu triedu si zvolíme ako "referenčnú", napr. poslednú L
- model logistickej regresie

$$log\frac{p_x^l}{p_x^L} = log\frac{P(c = l|x)}{P(c = L|x)} = \beta_{l0} + \sum_{i=1}^{p}\beta_{li}x_i, \qquad l = 1, \ldots, L-1$$

$$p_x^1 + p_x^2 + \cdots + p_x^L = 1$$

- **trénovanie:** odhad $\beta_{l0}, \beta_{l1}, \ldots, \beta_{lp} \rightarrow (p+1) \times (L-1)$ parametrov
- nové pozorovanie $\mathbf{x}^\star$:

$$P(c^\star = l|\mathbf{x}^\star) = p_{\mathbf{x}^\star}^l = p_{\mathbf{x}^\star}^L e^{\beta_{l0} + \sum_{i=1}^{p}\beta_{li}x_i^\star}, \quad l = 1, \ldots, L-1$$

$$p_x^1 + p_x^2 + \cdots + p_x^L = 1$$

$$\rightarrow \text{ systém rovníc}$$

$$c^\star \in argmax_{l \in \{1,\ldots,L\}} p_{\mathbf{x}^\star}^l$$

Logistická regresia v MATLAB-e - viac tried

[β,dev,stats] = mnrfit(X,C,Name,Value)

X - $n \times p$ matica pozorovaní

C - $n \times 1$ vektor zaradenia do tried

- voliteľné parametre:
  - *Model* - nominal (kategórie sa nedajú usporiadať), ordinal (prirodzené usporiadanie kategórií)
  - *Link* - 'logit' pre logistickú regresiu (default)

β - $(p+1) \times (L-1)$ matica odhadov koeficentov (L je počet tried)

dev - chyba modelu

stats - napr. $stats.p \rightarrow$ p-hodnoty o signifikantnosti koeficientov,...

Log. regresia v MATLAB-e - viac tried, predikcia

$p^\star = \textbf{mnrval}(\beta, Xnew)$
$[\sim, c^\star] = \textbf{max}(p^\star, [], 2)$

β - výstup z *mnrfit*

$Xnew$ - $n^\star \times p$ matica nových pozorovaní

$p^\star$ - predikcia pravdepodobností zaradenia do tried pre nové pozorovania

$c^\star$ - zaradenie do tried pre nové pozorovania

Logistická regresia - Príklad: Cukrovka

- 520 pozorovaní - 328 mužov a 192 žien (vek 48.03 ± 12.15)
- 2 triedy reprezentujúce skoré štádium cukrovky (0 a 1)

- 15 premenných
  - kardinálne - vek
  - kategorické (Y/N) - pohlavie, polyúria (PU), polydipsia (PD), strata hmotnosti, slabosť, polyfágia, rozmazané videnie, svrbenie, podráždenosť, oneskorené hojenie, čiastočné ochrnutie, stuhnutosť svalov, alopécia, obezita

- $mdl = fitglm(X,'Distribution','binomial','link','logit', ...$
  $...'CategoricalVars', 2:15)$
- $Beta = mdl.Coefficients.Estimate$

Logistická regresia - Príklad: Cukrovka

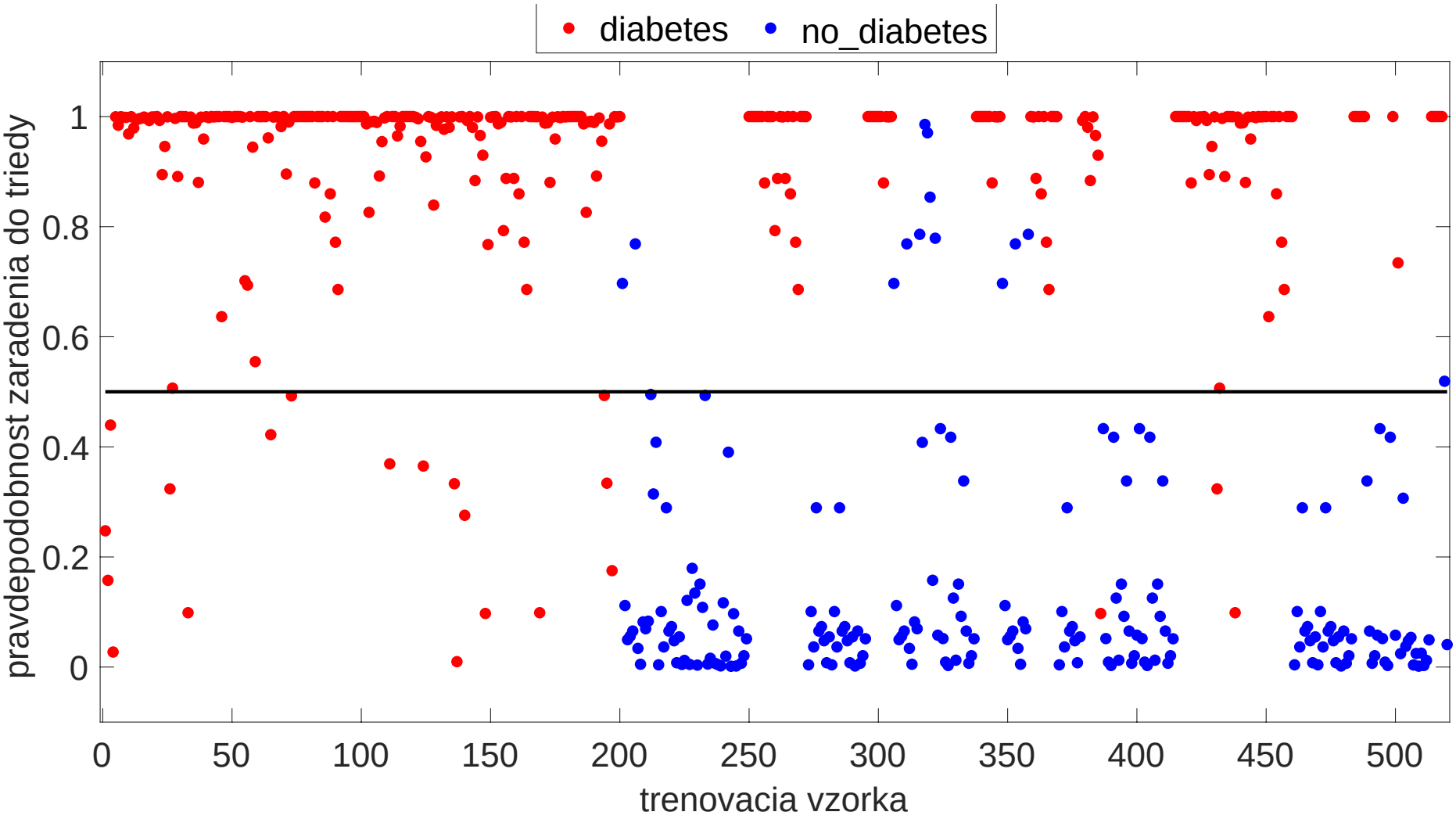
- výstup funkcie *fitglm*

```
>> mdl

mdl =


Generalized linear regression model:
    druh ~ [Linear formula with 16 terms in 15 predictors]
    Distribution = Binomial

Estimated Coefficients:
```

	Estimate	SE	tStat	pValue
(Intercept)	-1.4117	0.85788	-1.6455	0.099857
vek	-0.041593	0.022335	-1.8622	0.062569
pohlavie_1	4.0855	0.5546	7.3665	1.7523e-13
polyuria_1	4.3912	0.66888	6.565	5.205e-11
polydipsia_1	4.8511	0.7532	6.4407	1.1892e-10
strataHmot_1	0.50327	0.51244	0.9821	0.32605
slabost_1	0.63528	0.5078	1.251	0.21092
polyfagia_1	1.095	0.53261	2.0559	0.039795
rozmazaneVidenie_1	0.3803	0.59692	0.63711	0.52405
svrbenie_1	-2.3851	0.64079	-3.7221	0.00019754
podrazdenost_1	2.4206	0.57713	4.1943	2.7372e-05
oneskoreneHojenie_1	-0.13945	0.5376	-0.25939	0.79533
ciastocneOchrnutie_1	0.88854	0.49071	1.8107	0.070184
stuhnutostSvalov_1	-0.96566	0.54945	-1.7575	0.078835
alopecia_1	0.24282	0.5876	0.41323	0.67944
obezita_1	-0.27569	0.52539	-0.52474	0.59976

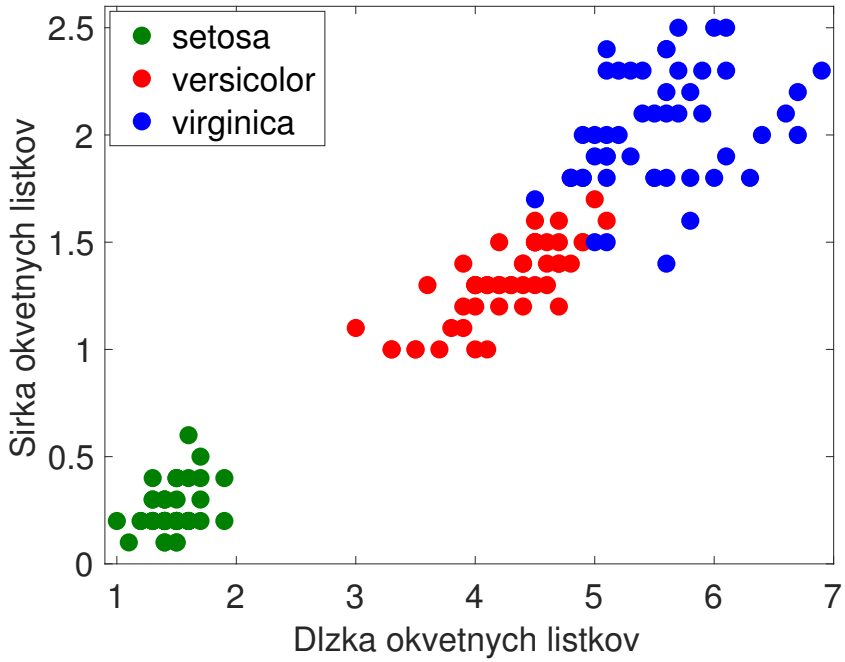
```
520 observations, 504 error degrees of freedom
Dispersion: 1
Chi^2-statistic vs. constant model: 509, p-value = 7.46e-99
```

Logistická regresia - Príklad: Cukrovka



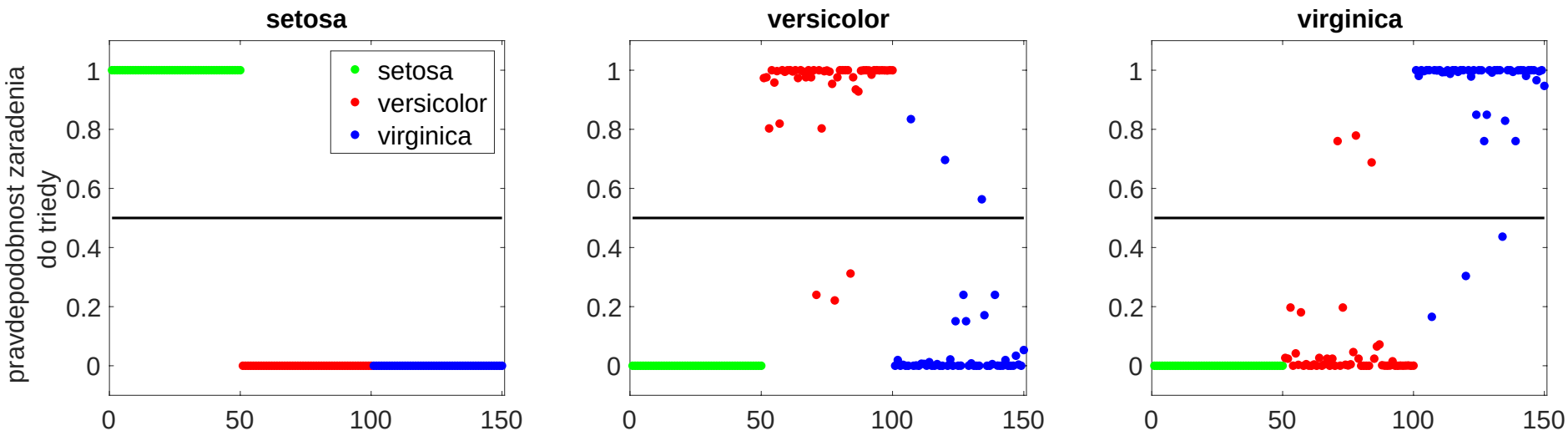
Logistická regresia - Príklad: Kosatce

- databáza **Fisher Iris** v MATLAB-e
- 150 pozorovaní, 3 triedy $\rightarrow$ musíme ísť cez funkciu *mnrfit*
- 2 premenné - dĺžka a šírka okvetného lístka
- odhad parametrov modelu
  $[B, dev, stats] = mnrfit(X, C)$



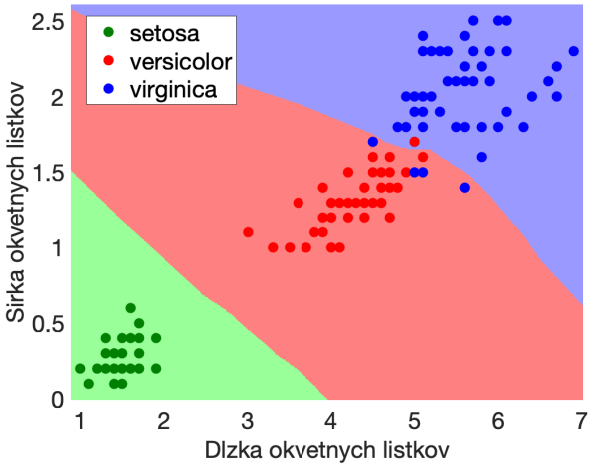
Logistická regresia - Príklad: Kosatce

- MATLAB automaticky zvolil ako referenčnú kategóriu *virginica*
- niektoré pozorovania sú "neistejšie" z hľadiska pravdepodobnosti

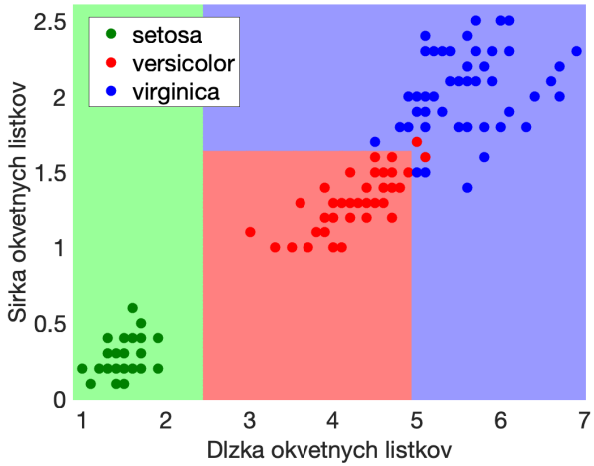


Logistická regresia - Príklad: Kosatce

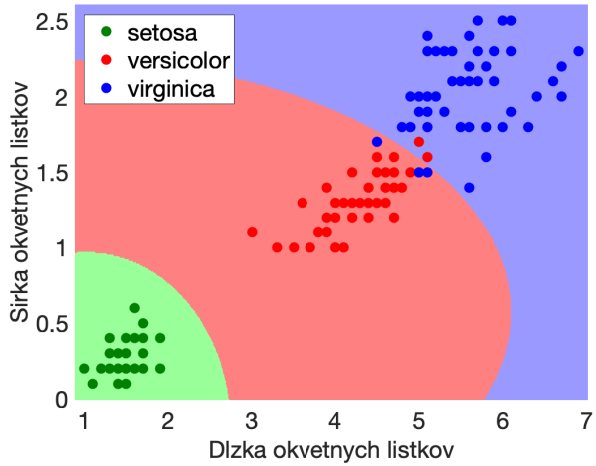
KNN



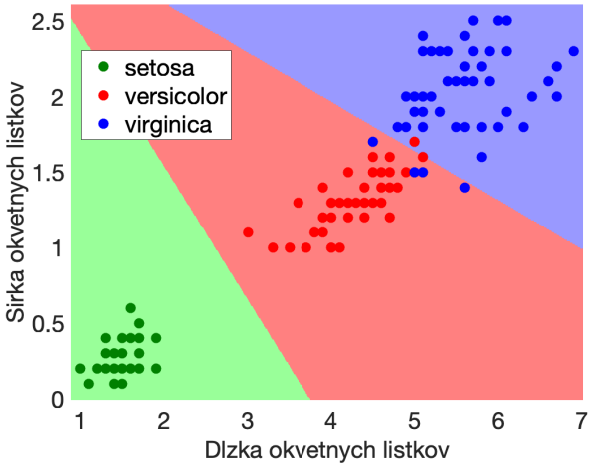
Ctree



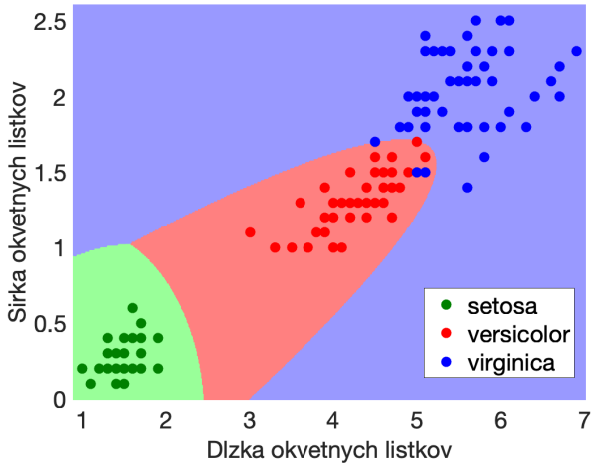
NBC



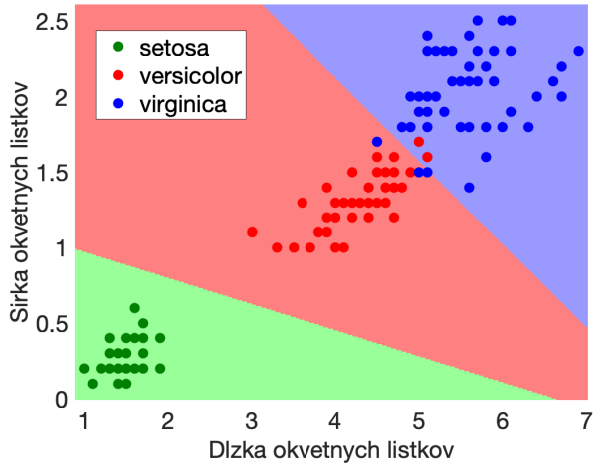
LDA



QDA



LogReg



Logistická regresia zhrnutie

- **výhody:**
  - jednoduchá na pochopenie, interpretáciu
  - vie pracovať s kardinálnymi aj kategorickými premennými
  - výstupom je aj pravdepodobnosť zaradenia do tried $\rightarrow$ vidíme, ktoré pozorovania sú "na hranici"
  - vo výstupe modelu logistickej regresie vidíme, ktoré premenné sú signifikantné $\rightarrow$ najviac prispievajú k odlíšeniu tried
- **nevýhody:**
  - ak $n << p$, môže viesť k nadhodnoteným výsledkom
  - v porovnaní s inými metódami trochu "ťažšia" implementácia v MATLAB-e

Otázky ?