

Strojové učení a rozpoznávání

Úvod do problematiky

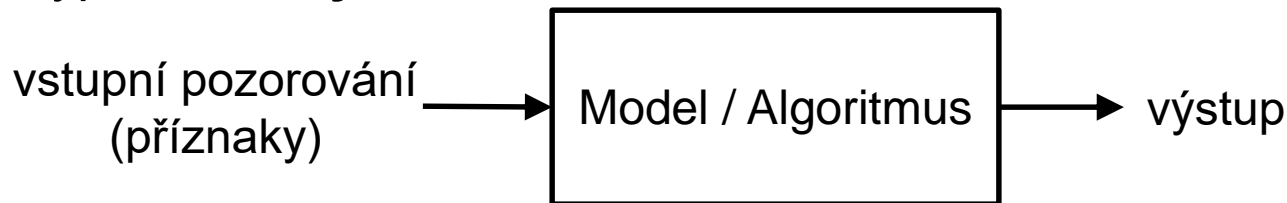
Lukáš Burget



Co je strojové učení?

- Tom M. Mitchell : „Říkáme, že počítačový program se učí ze zkušenosti E s ohledem na nějakou třídu úloh T a nějaké měřítko úspěšnosti P , pokud jeho se jeho úspěšnost měřená pomocí P , zlepší se zkušeností E .“

- Typické **úlohy**:

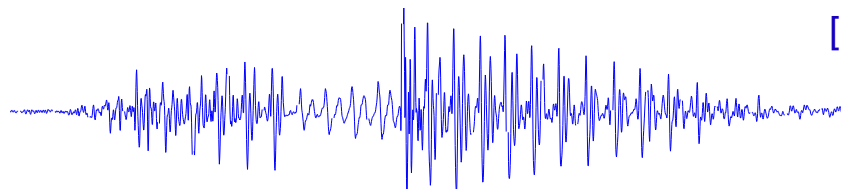


Příklady:

- Prozpoznej slova v řečové nahrávce
 - Rozpoznej identitu člověka z obrázku obličeje
 - Přelož český text do Korejštiny
 - Klasifikuj objekt na základě změřené velikosti a váhy
 - Predikuj cenu akcií z údajů o hospodaření firmy
- Typická reprezentace **zkušenosti**: Kolekce trenovacích vzorů (vstupy a/nebo výstupy).
 - Typický způsob **měření úspěšnosti**: Konrola jak dobře řešíme úlohu na nových (testovacích) vstupech a požadovaných výstupech.

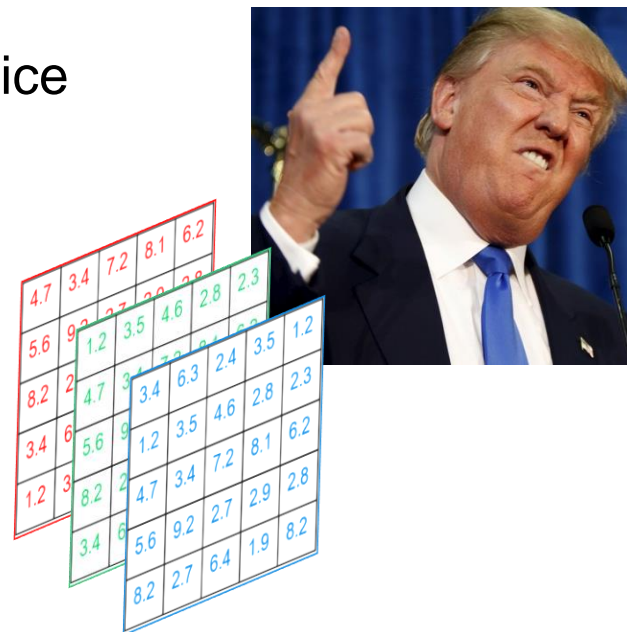
Příklady vstupních pozorování I.

- Řečová nahrávka – různě dlouhé sekvence numerických hodnot



[0.1, 1.5, 5.4, 5.2, 1.1, -2.3, -5.4, ..., 0.8]

- Obrázek obličeje 100x100 pixelů – 3D matice numerických hodnot (jedna dimenze pro barevný kanál)



- Sekvence slov – různě dlouhé sekvence diskretních symbolů

Příklady vstupních pozorování II.

- Vstupní pozorování (tzv. příznaký) mohou mít
 - Spojité nebo diskrétní hodnoty
 - Pevnou nebo proměnlivou velikost (např. vektor, matice vs. sekvence hodnot, vektoru, symbolu, ...)
- V našich příkladech budou pozorování často jen D -dimenzionální vektory reálných čísel (nebo jen reálné číslo):

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_D \end{bmatrix}$$

- Množina N pozorování (např. trénovací vzory) potom bude reprezentována maticí:

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N] = \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{N1} \\ x_{12} & x_{22} & \cdots & x_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1D} & x_{2D} & \cdots & x_{ND} \end{bmatrix}$$

Příklady vstupních pozorování III.

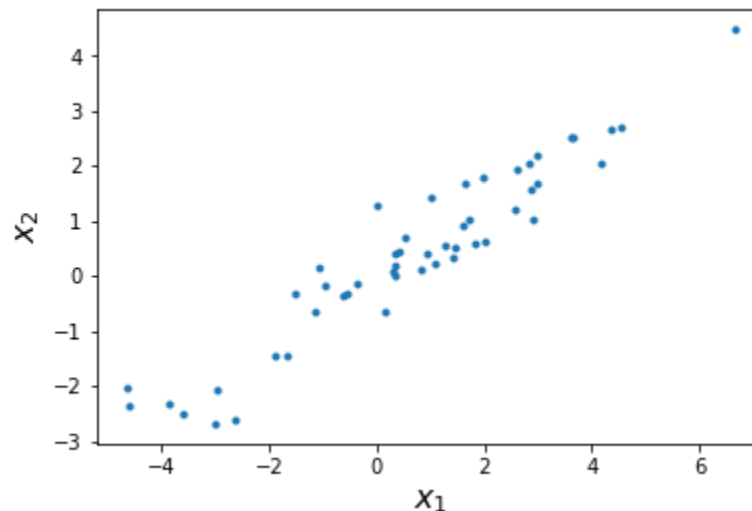
- Pro jednoduchost bude často vstupní pozorování jen 2D vektor:

Příklady:

- Vektor dvou hodnot odpovídající velikosti a váze vstupního objektu
- Triviální čenobílý (grayscale) “obrázek” o dvou pixelech

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

- Potom můžeme zobrazit vstupní pozorování jako bod ve 2D prostoru:



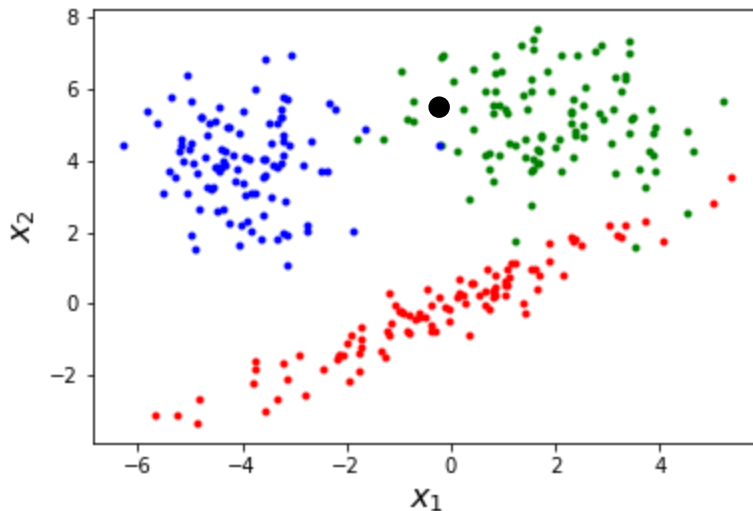
Co je strojové učení? II.

- Na daných trénovacích vzorech (vstupy a/nebo výstupy) se učíme zobrazit neviděné vstupy na požadované výstupy.
Příklady:
 - Na stovkách hodin řečových nahrávek s textovým přepsem se učíme automaticky přepsat nové řečové nahrávky.
 - Na datové sadě milionů obrázků lidských obličejů kde známe identitu člověka se učíme rozpoznat identitu člověka v nových obrázcích.
- Hlavní typy učících se algoritmů
 - Učení s učitelem (Supervised Learning)
 - Trénovací vzory jsou dvojice vstupů a požadovaných výstupů
 - Typické úkoly: klasifikace či obecně rozpoznávání vzoru, regrese, ...
 - Učení bez učitele (Unsupervised Learning)
 - Trénovací vzory jsou pouze „neanotovaná“ (vstupní) data
 - Typické úkoly: shlukování, detekce anomálií, odhad rozložení pravděpodobnosti, ...
 - Semi-supervizované učení (Semi-supervised learning)
 - Některé trénovací vzory jsou anotované dvojice vstup/výstup, ale pro některé (většinu) máme jen neanotované vstupy.
 - Posilované učení (Reinforcement learning)
 - Parametry modelu jsou upravovány na základě pozitivního či negativního zpětné vazby po tom co uděláme serii rozhodnutí/akcí (např. na konci vyhrané/prohrané hry, po (ne)úspěšné jízdě automaticky řízeným autem).
 - Příklady: naučit řídit autonomní vozidlo, naučit počítač hrát deskové či počítačové hry

Učení s učitelem

(Supervised Learning)

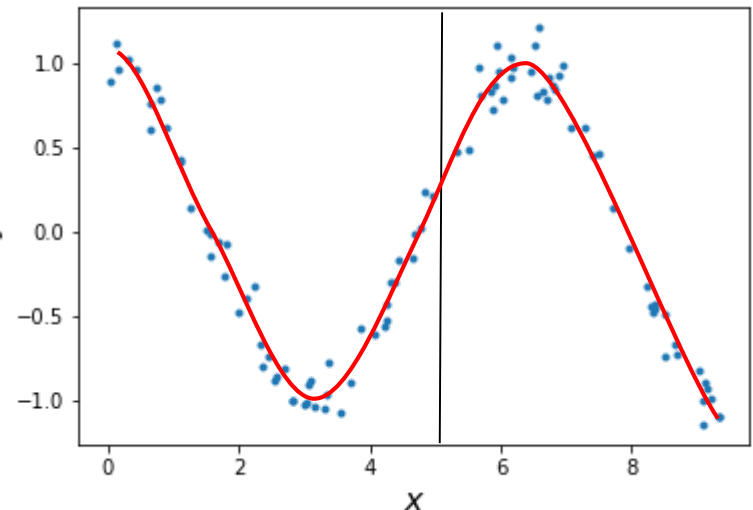
- Trénovací vzory jsou dvojice vstupů \mathbf{x} a požadovaných výstupů \mathbf{y}
Klasifikace:



$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad y = \{\text{red, green, blue}\}$$

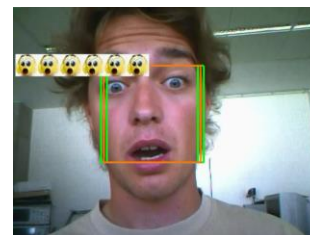
Na trénovacích vzorech (barevné tečky) se učíme přiřadit třídu (barvu) novému před tím neviděnému vzoru (černá tečka)

- Regrese:
Na trénovacích vzorech se učíme predikovat pravděpodobné hodnoty y pro nové vstupy x .
Tedy učíme se funkci $y = f(x)$



Učení s učitelem - příklady

- Všechny předchozí příklady odpovídaly učení s učitelem
- Classification:
 - Rozpoznej identitu člověka z obrázku obličeje
 - Klasifikuj objekt na základě změřené velikosti a váhy
 - Rozpoznej výraz v obličeji pro každý snímek videa
- Regression:
 - Predikuj cenu akcií z údajů o hospodaření firmy
 - Predikuj počasí (teplotu, vlhkost, pravděpodobnost deště, ...) z historie meteorologických měření
- More general pattern recognition problems
 - Prozpoznej slova v řečové nahrávce
 - Detekuj a klasifikuj všechny známe (>9k) objekty ve videu <https://youtu.be/MPU2Histivl>
 - Odhadni pózu každého člověka ve videu <https://youtu.be/pW6nZXeWIGM>
- Other supervised learning problem
 - Přelož český text do Korejštiny (Strojový překlad, Machine Translation)
 - Automatický popiš obrázek anglickým textem
 - Generuj realistické obrázky z textového popisu



Učení s učitelem - příklady

- Automatický popis obrázku anglickým textem
Kombinace konvoluční a rekurentní neuronové sítě
Andrej Karpathy, Li Fei-Fei: Deep Visual-Semantic Alignments for Generating Image Descriptions



man in black shirt is playing guitar.



construction worker in orange safety vest is working on road.



two young girls are playing with lego toy.



boy is doing backflip on wakeboard.

Supervised learning - examples

- Generování realistických obrázků z textového popisu

Generativní kompetitivní neuronová síť (Generative Adversal Neural)

Han Zhang, et al.: StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks.



This bird has a yellow belly and tarsus, grey back, wings, and brown throat, nape with a black face



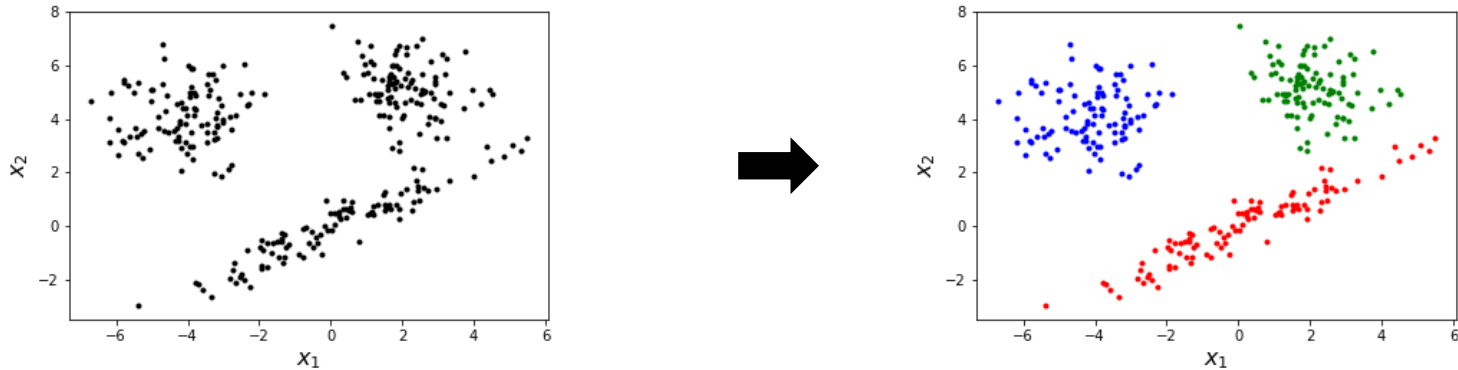
This bird is white with some black on its head and wings, and has a long orange beak



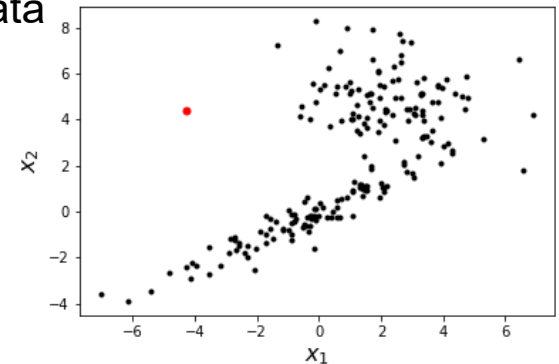
This flower has overlapping pink pointed petals surrounding a ring of short yellow filaments

Učení bez učitele I.

- Shlukování (Clustering): najdi shluky „podobných“ vstupních vzorů



- V kolekci dokumentů najdi podobné dokumenty (stejně téma)
- Zjistí kolik lidí mluví v nahrávce konverzace a kdo mluví kdy (diarizace)
- Detekce anomálií: detekuj neobvyklé vstupy (outliers)
 - pro zamítnutí dalšího zpracování
 - abychom na ně upozornili jako na zajímavá nová data



Učení bez učitele I.

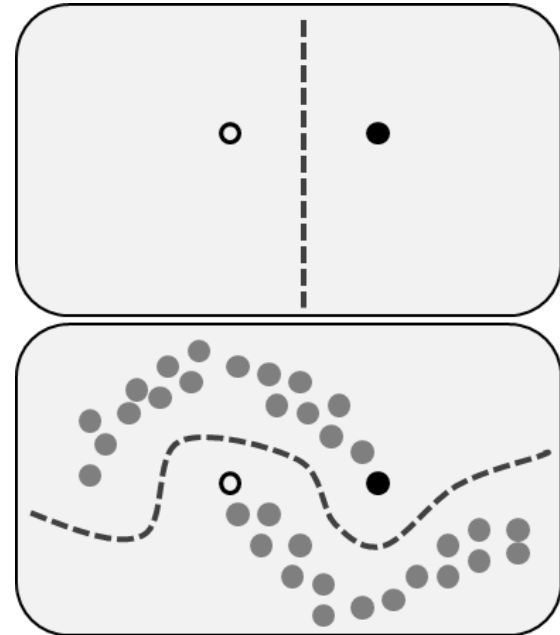
- Odhad rozložení pravděpodobnosti z dat
 - Uvidíme jak odhadnout parametry jednoduchých rozložení pravděpodobnosti (Gaussovské, Diskrétní, Směs Gaussovských rozložení)
 - S využívajícími hlubokých neuronových sítí můžeme modelovat (a generovat vzory ze) složitých rozložení (např. rozložení obrázků lidských tváří)

Diederik P. Kingma, Prafulla Dhariwal: Glow: Generative Flow with Invertible 1x1 Convolutions



Semisupervizované učení

- Unannotated examples can help to find better decision boundary between classes
- There is lots of unannotated data available on the internet
 - Text
 - Photos and other images
 - Speech and other recordings
 - ...



Jak postavi klasifikátor?

The task:



vs.



apple

**Marmelade
factory**

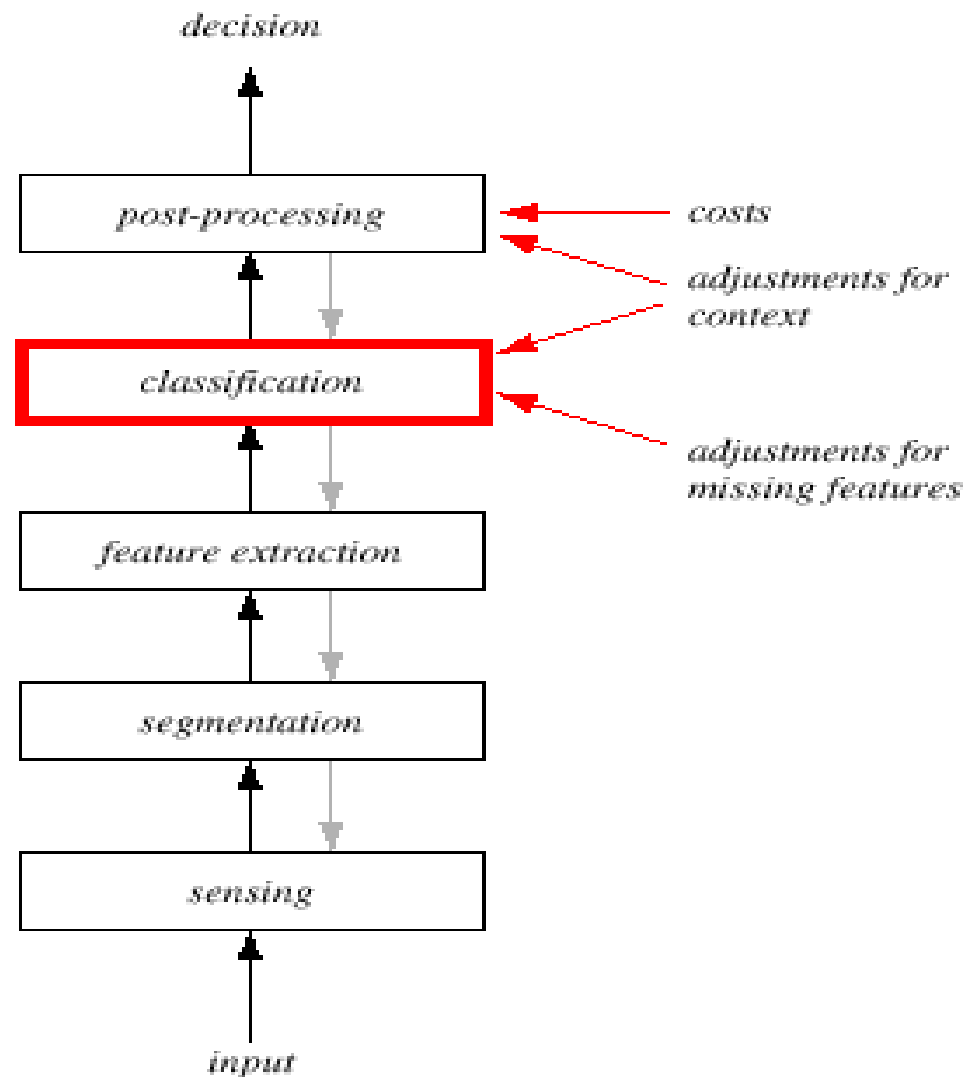


Pyrotechnician



grenade

Jak takový klasifikátor pracuje?



Snímání

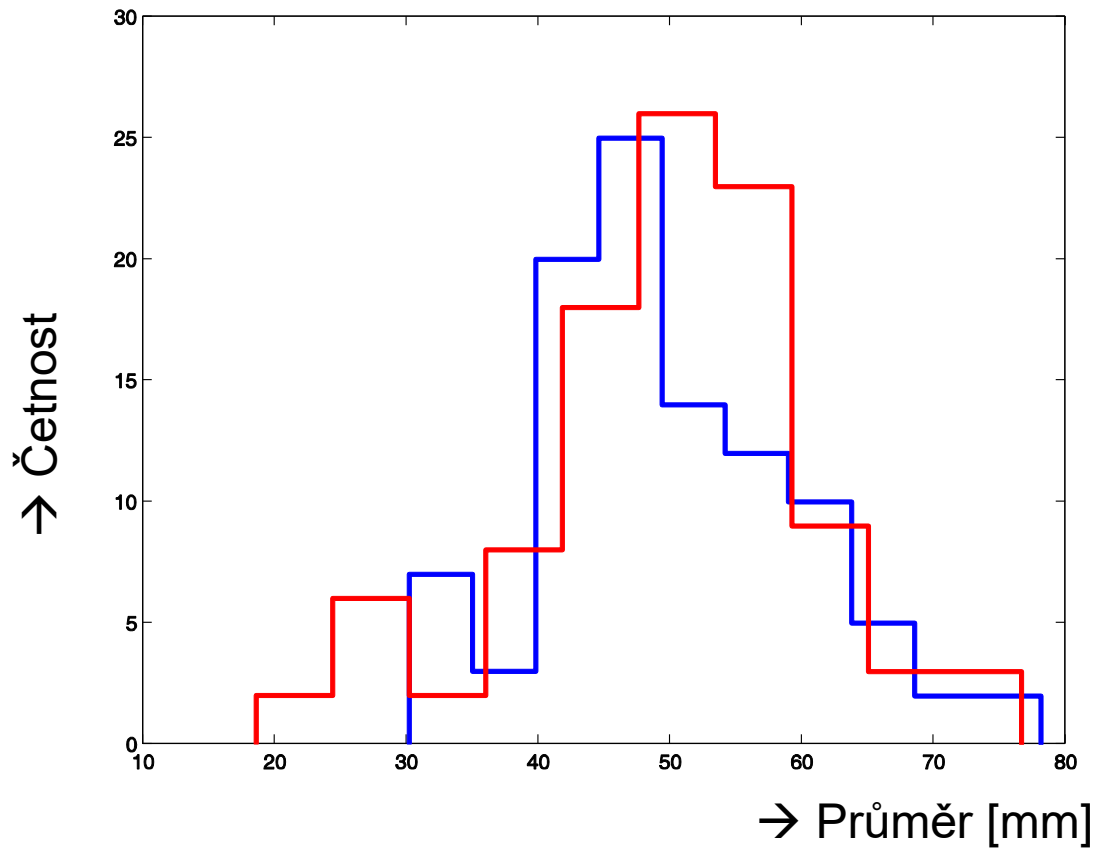
- Co se dá o rozpoznávaných předmětech zjistit?
 - obraz, tlak, teplota, hmotnost, zvuk, pach ?
 - jak tyto veličiny prakticky získat, jde to vůbec a kolik to bude stát ?
 - jaké vlastnosti bude mít snímač a převod veličina → číslo ?
 - šum
 - linearita
 - kalibrace
 - stárnutí
 - atd.

Extrakce příznaků

- Příznaky musí umožnit rozlišovat mezi třídami → musí být diskriminativní.
- Předzpracování vstupu do následujícího klasifikátoru
 - Redukce dimenzí
 - Invariance vůči:
 - translace (místo v obrázku, čas v řeči)
 - rotace
 - scale (velikost v obrázku, volume v řeči)
 - occlusion (zakrytí objektu vs. Maskování šumem)
 - projective distortion (úhel pohledu, optika)
 - rate (rychlost v řeči - intra- a inter-speaker variabilita)
 - deformace
 - atd.
 - Dekorelace...ale o tom ještě bude řeč v samostatné přednášce o příznacích.

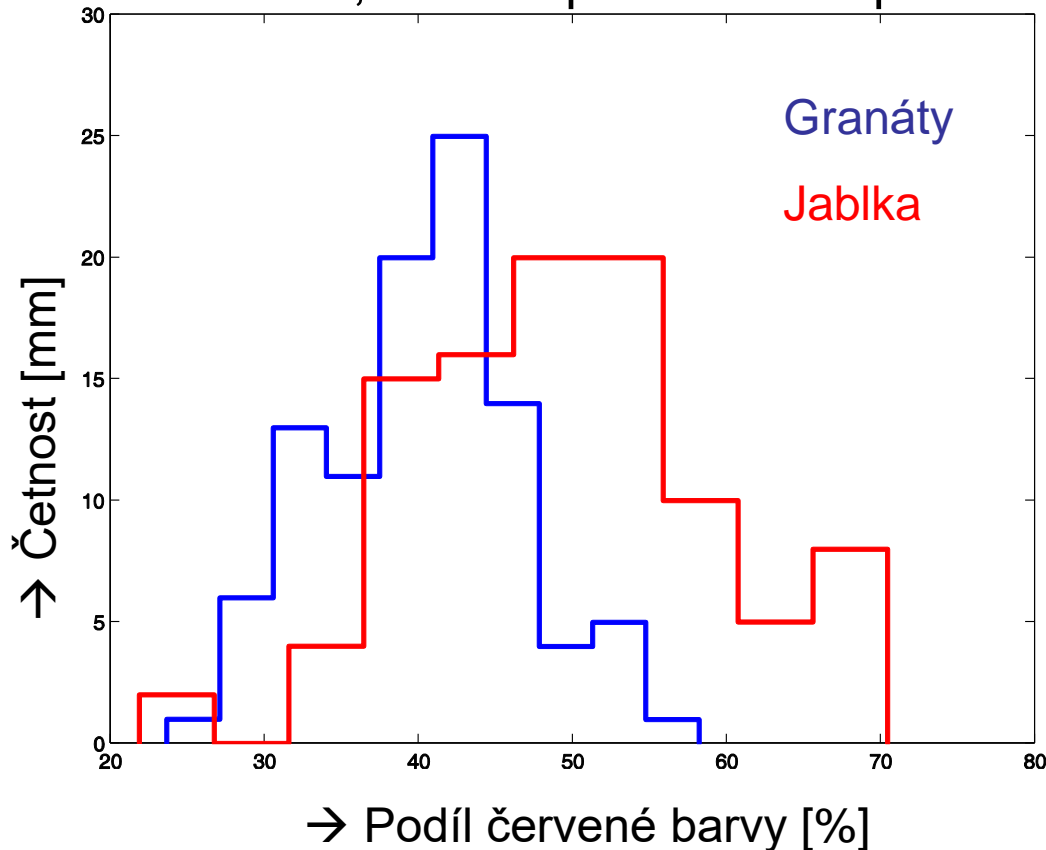
Extrakce příznaků

Bude průměr jablka / granátu dobrým příznakem?



Extrakce příznaků

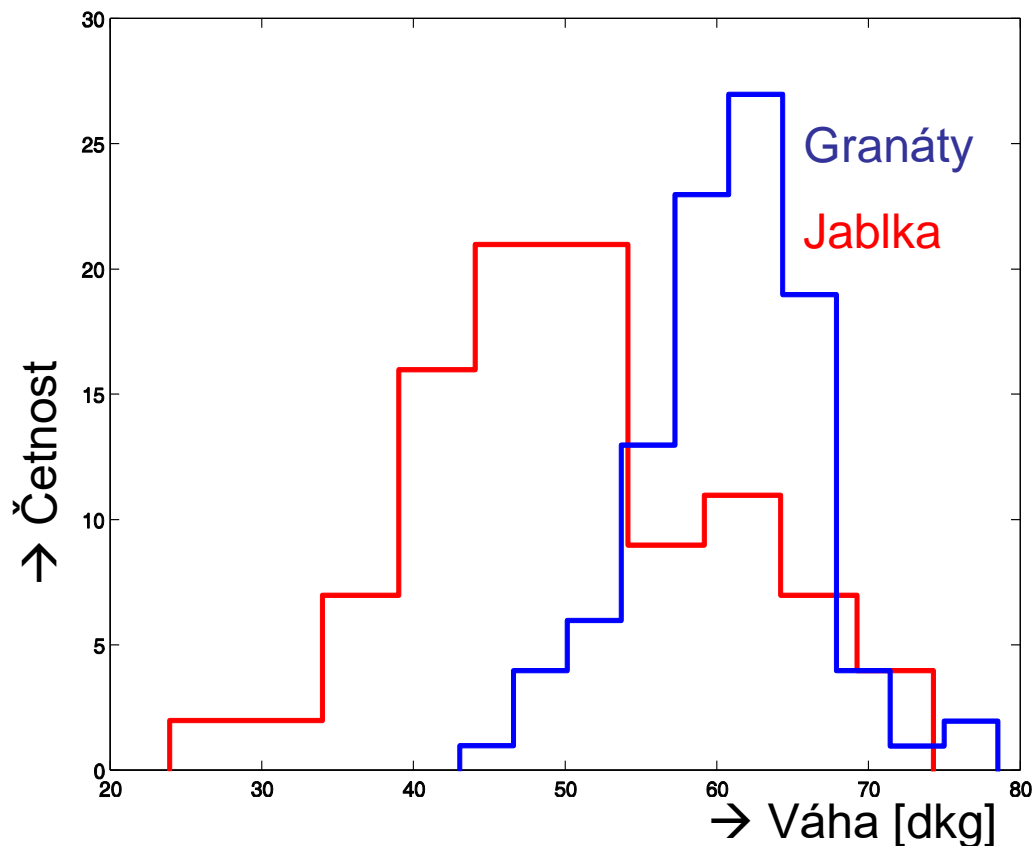
Nic moc, ale alespoň trochu lepší



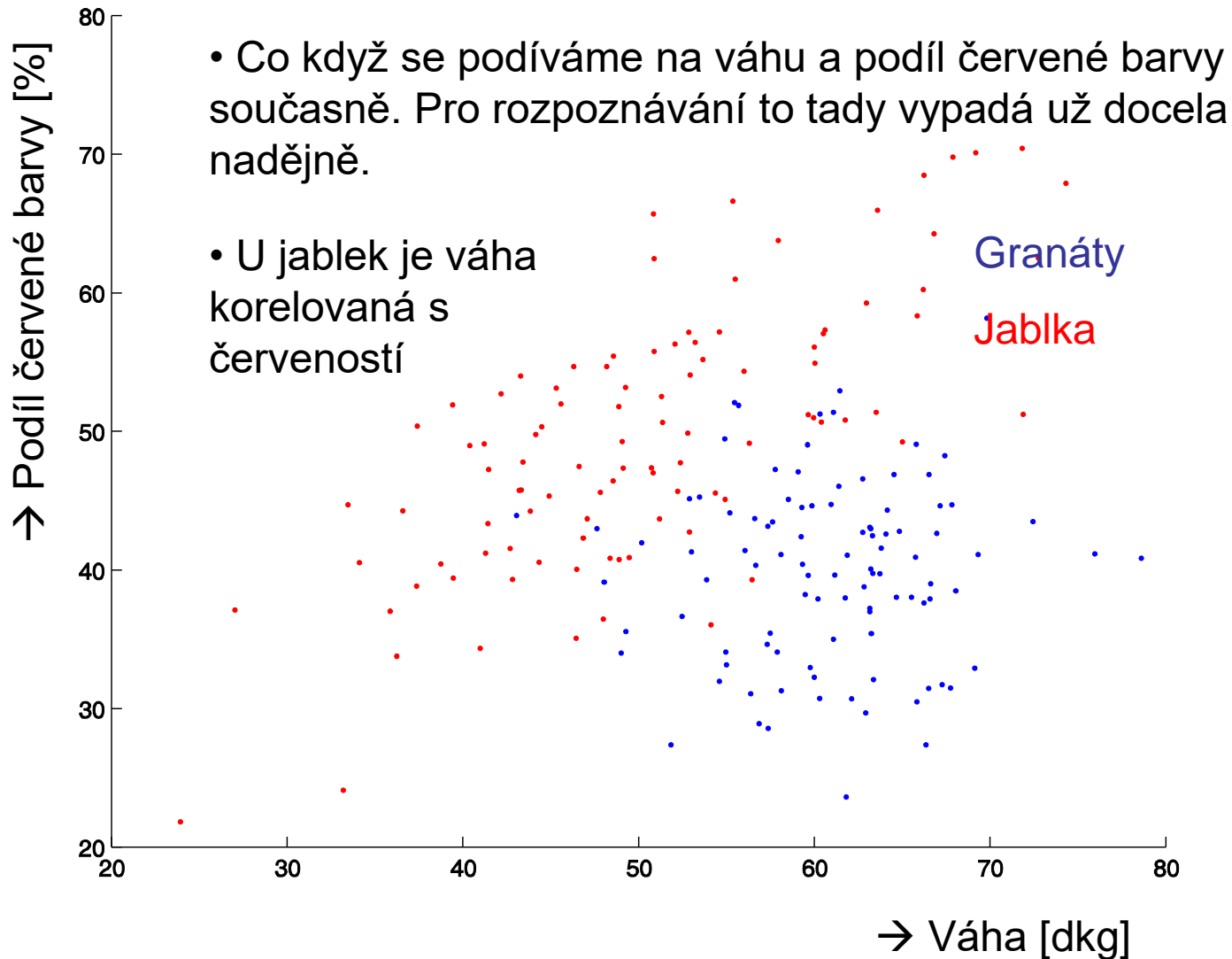
Intuitivně už bychom mohli začít rozpoznávat, nastavením prahu tak aby bylo co nejvíce vzorů, které jsme zatím viděli rozpoznáno správně

Extrakce příznaků

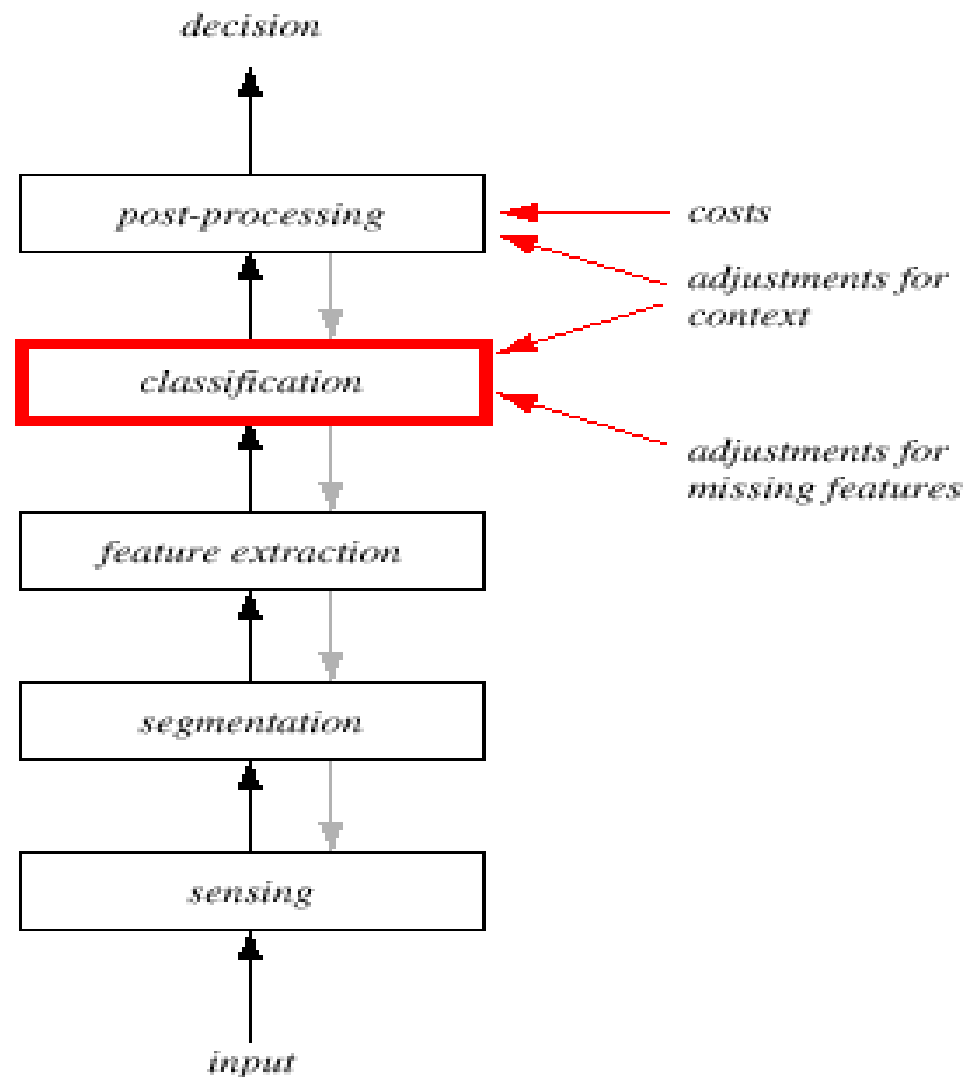
Když se tak díváme na histogramy příznaků, asi nás budou pro rozpoznávání zajímat jejich pravděpodobnostní rozložení ...ale to už zase předbíháme.



Vícerozměrné příznaky

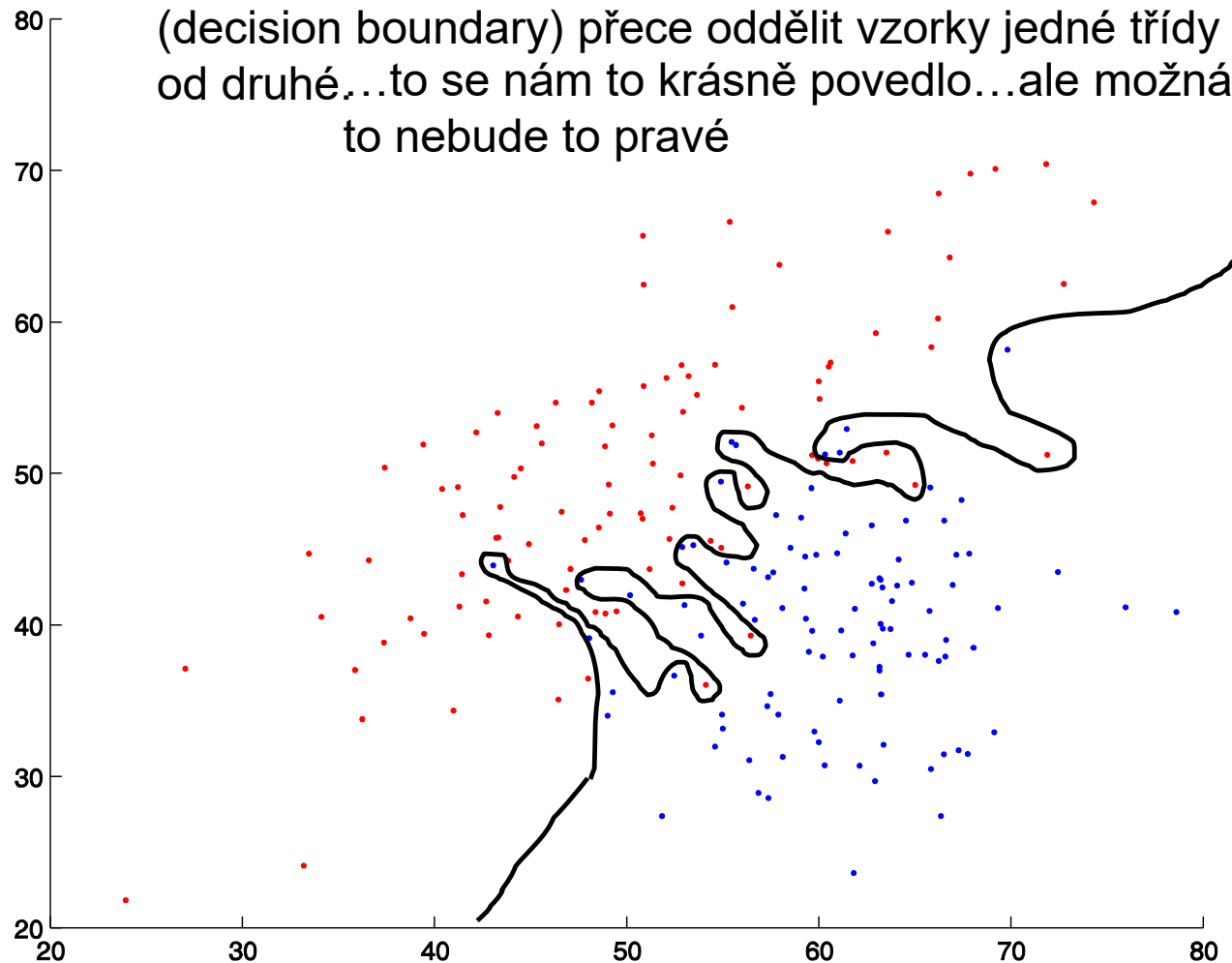


Klasifikace



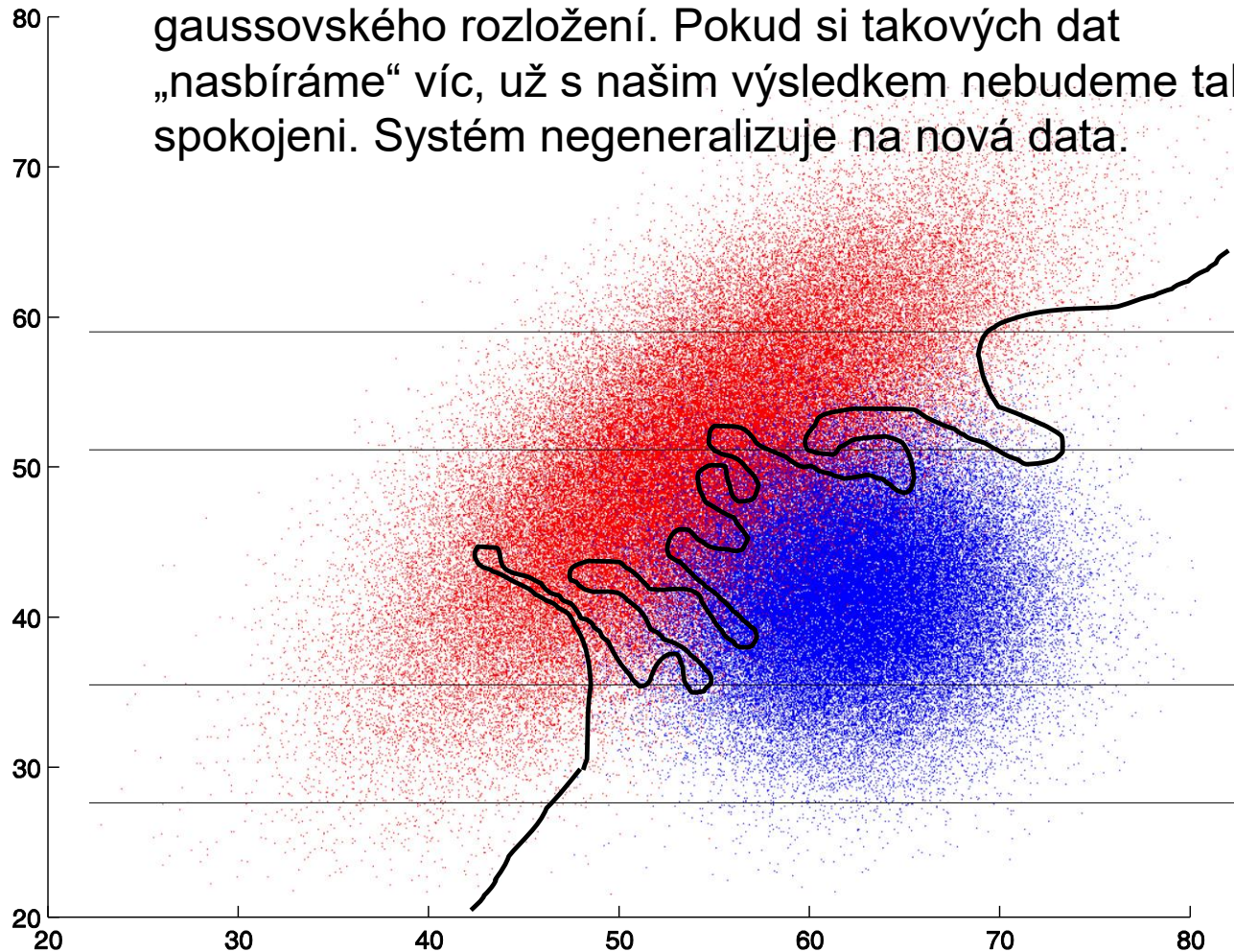
Klasifikace

Jde nám o to je najít vhodnou **rozhodovací hranici** (decision boundary) přece oddělit vzorky jedné třídy od druhé...to se nám to krásně povedlo...ale možná to nebude to pravé



Generalizace

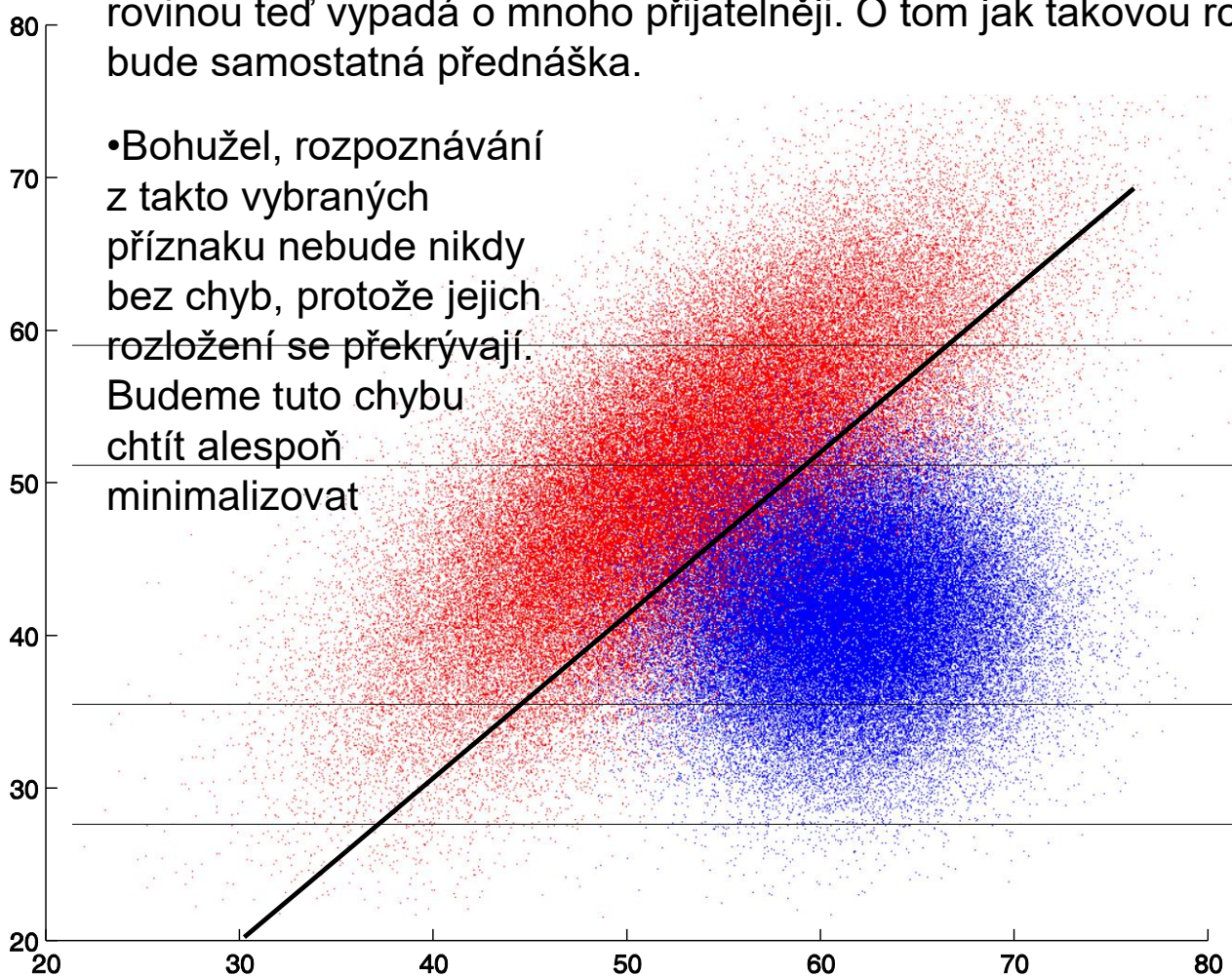
V našem příkladu byly data náhodně vygenerovány z gaussovského rozložení. Pokud si takových dat „nasbíráme“ víc, už s našim výsledkem nebudeme tak spokojeni. Systém negeneralizuje na nová data.



Lineární klasifikátor

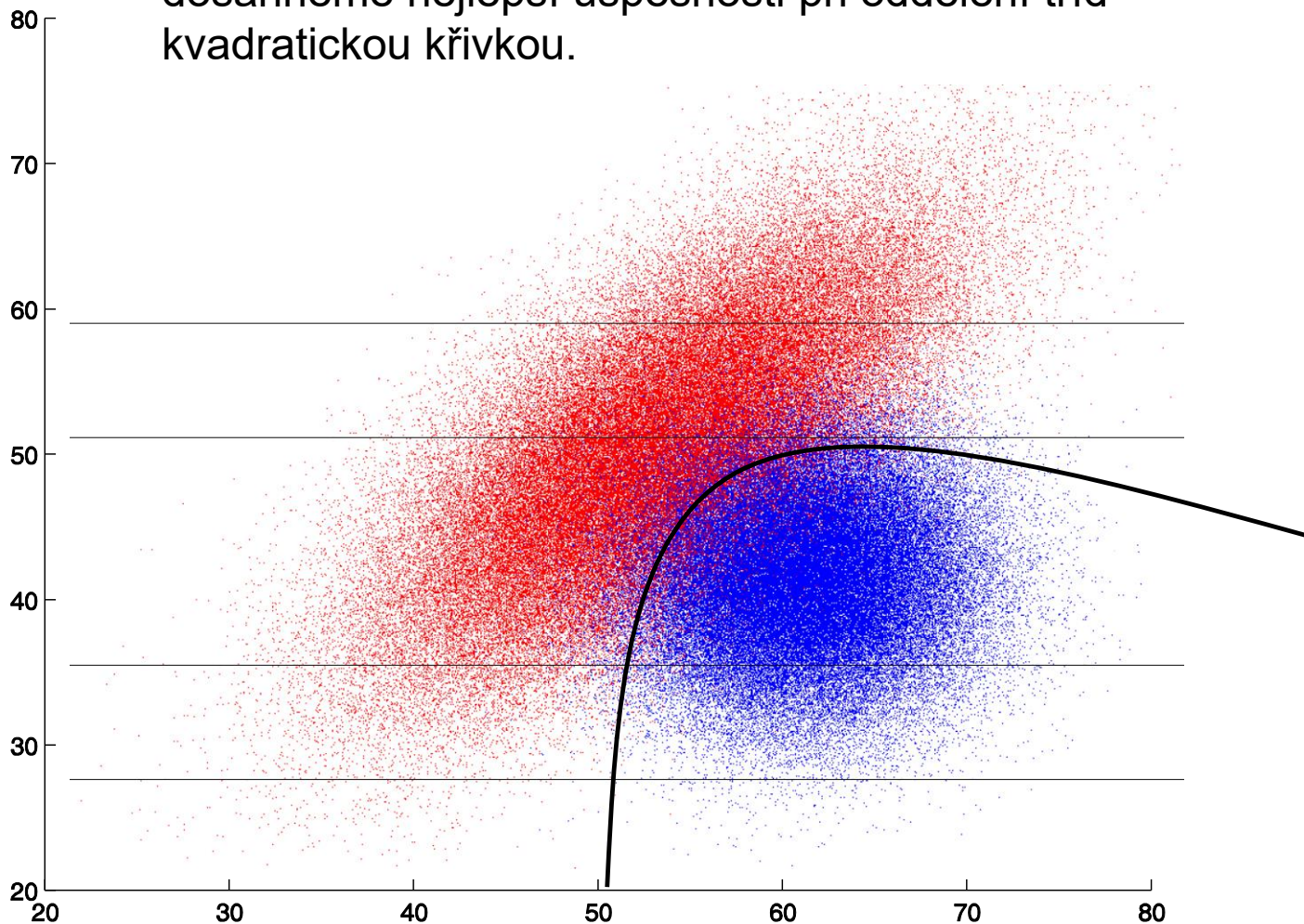
- Omezení schopnosti detailně modelovat rozhodovací linii vedlo ke zlepšení generalizace. Oddělení tříd prostou přímkou nebo obecně hyperrovinou teď vypadá o mnoho přijatelněji. O tom jak takovou rovinu určit bude samostatná přednáška.

- Bohužel, rozpoznávání z takto vybraných příznaku nebude nikdy bez chyb, protože jejich rozložení se překrývají. Budeme tuto chybu chtít alespoň minimalizovat



Kvadratická rozhodovací hranice

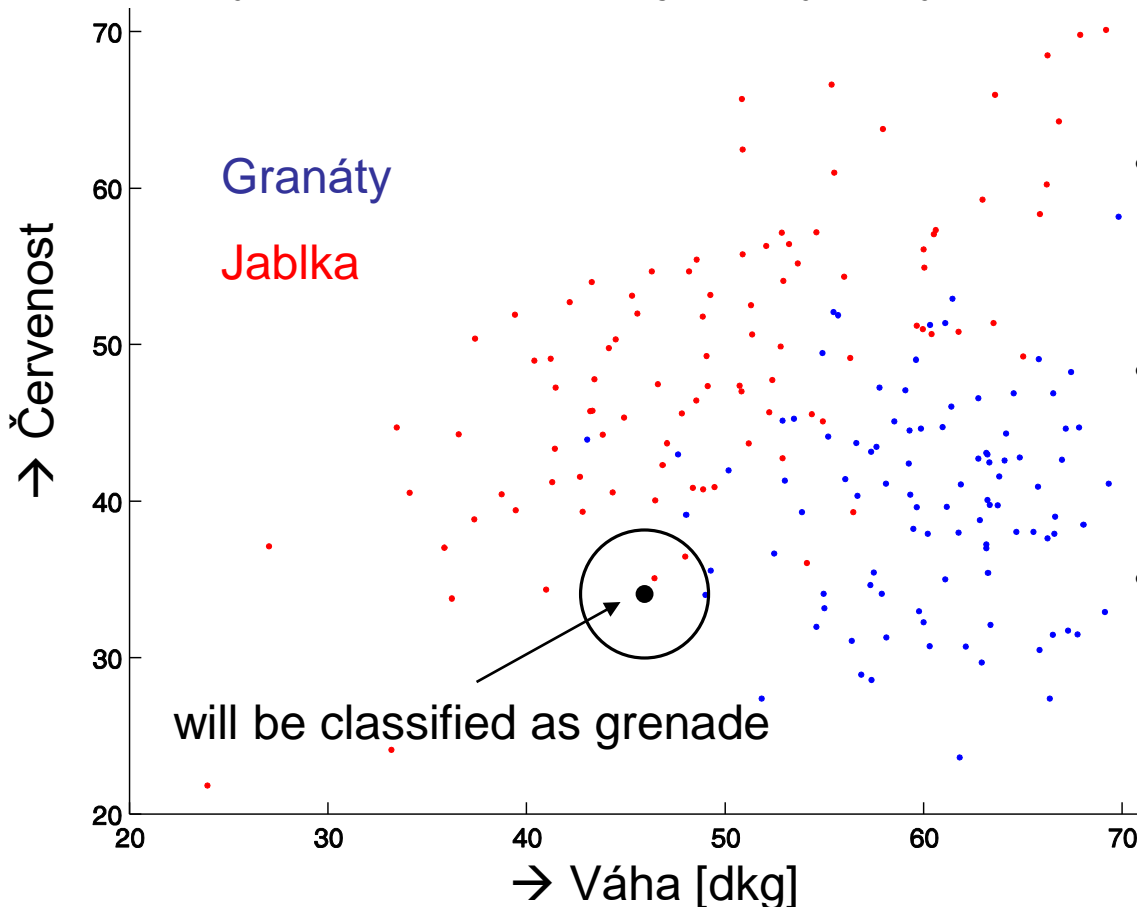
- V příští přednášce si ukážeme, že pro tento případ, kdy mají jednotlivé třídy gaussovské rozložení, dosáhneme nejlepší úspěšnosti při oddělení tříd kvadratickou křivkou.



Algoritmus k-nejbližších sousedů

(K-nearest neighbors classifier)

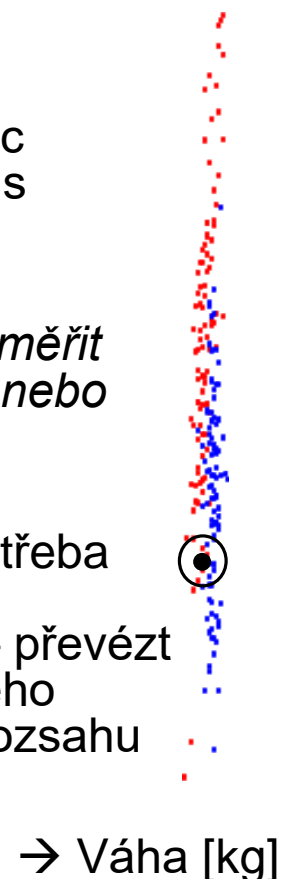
- “Neparametrický klasifikátor” → nemá žádné parametry, které by bylo potřeba trénovat či odhadovat.
- Klasifikátor si pamatuje všechna “trénovací data”.
- K nově klasifikovanému vzoru (černá tečka) najde K nejbližších vzorů z trénovacích dat a vybere tu třídu, která je ve vybraných vzorech nejčastěji zastoupena.



- Můžu ale vůbec rovnávat váhu s červeností?

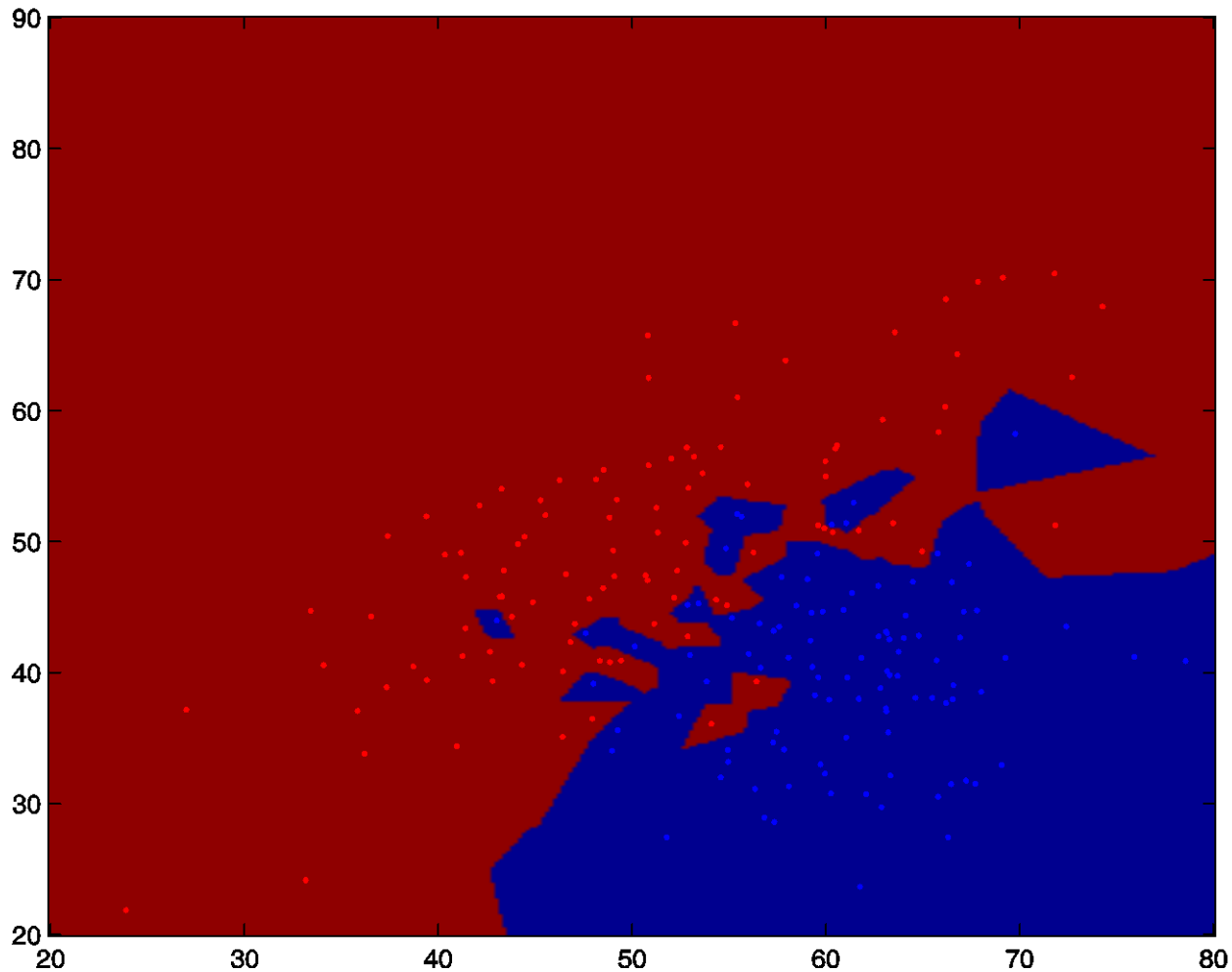
- *Co když budu měřit váhu v tunách nebo miligramech*

- Prvně bude potřeba obě veličiny normalizovat – převést do srovnatelného dynamického rozsahu



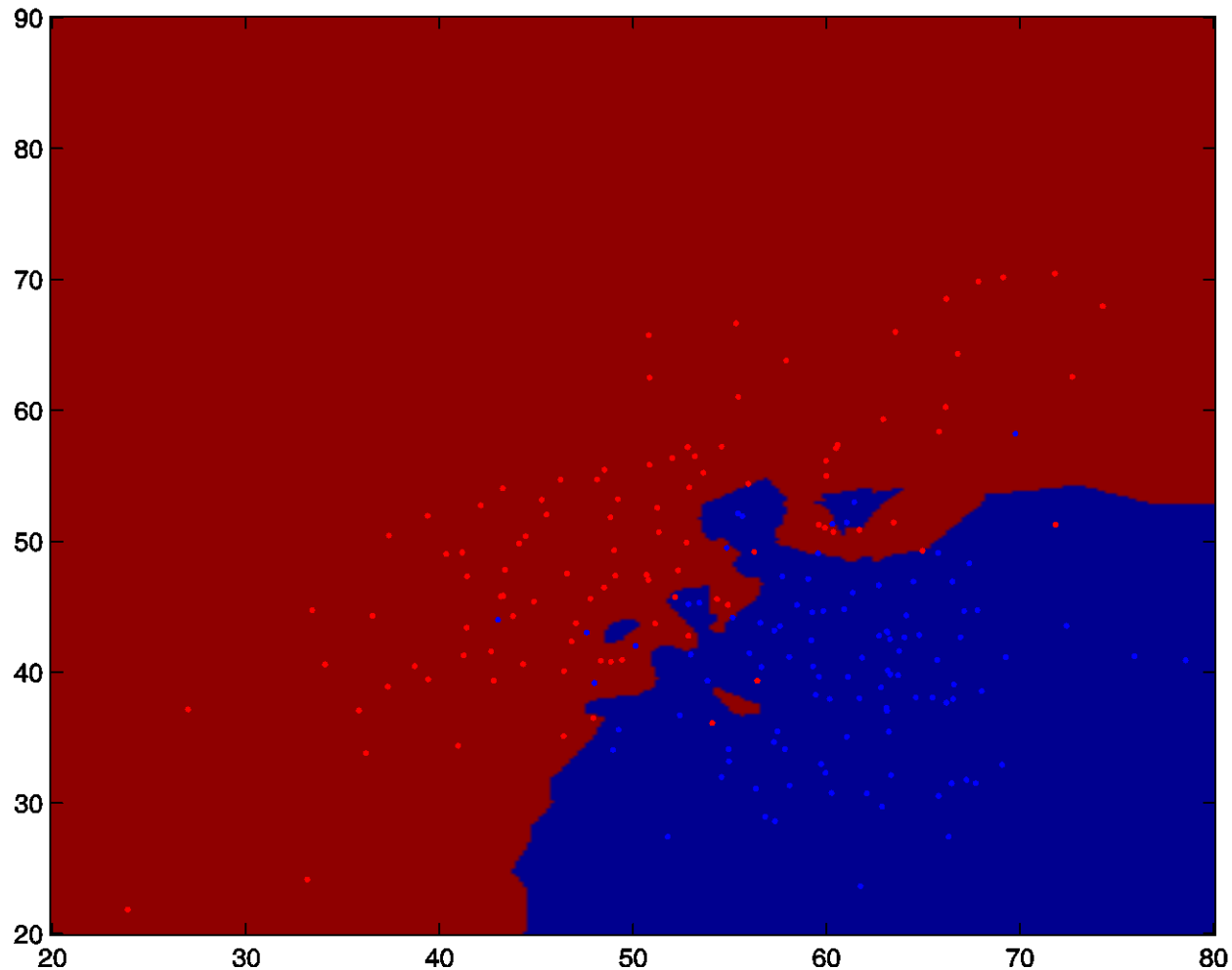
1-nejbližší soused

Opět problém z generalizací – podobná klikatá rozhodovací hranice.



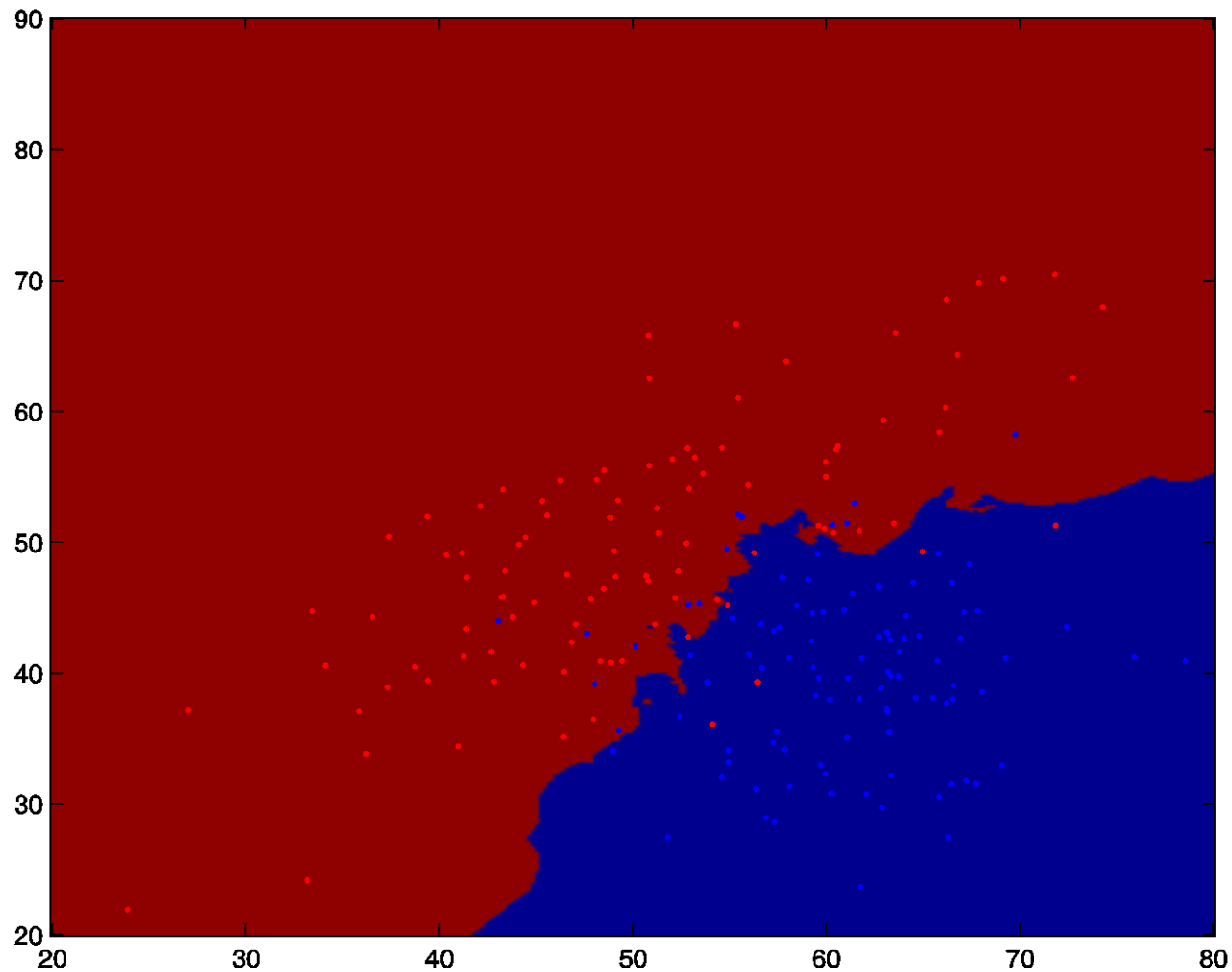
3-nejbližší sousedé

O něco lepší výsledek. Zvýšení „počtu sousedů“ vede k vyhlazení rozhodovací hranice, přestože jsme model nijak nezjednodušili. Zde by se dalo mluvit o období **regularizace** (viz další přednášky)



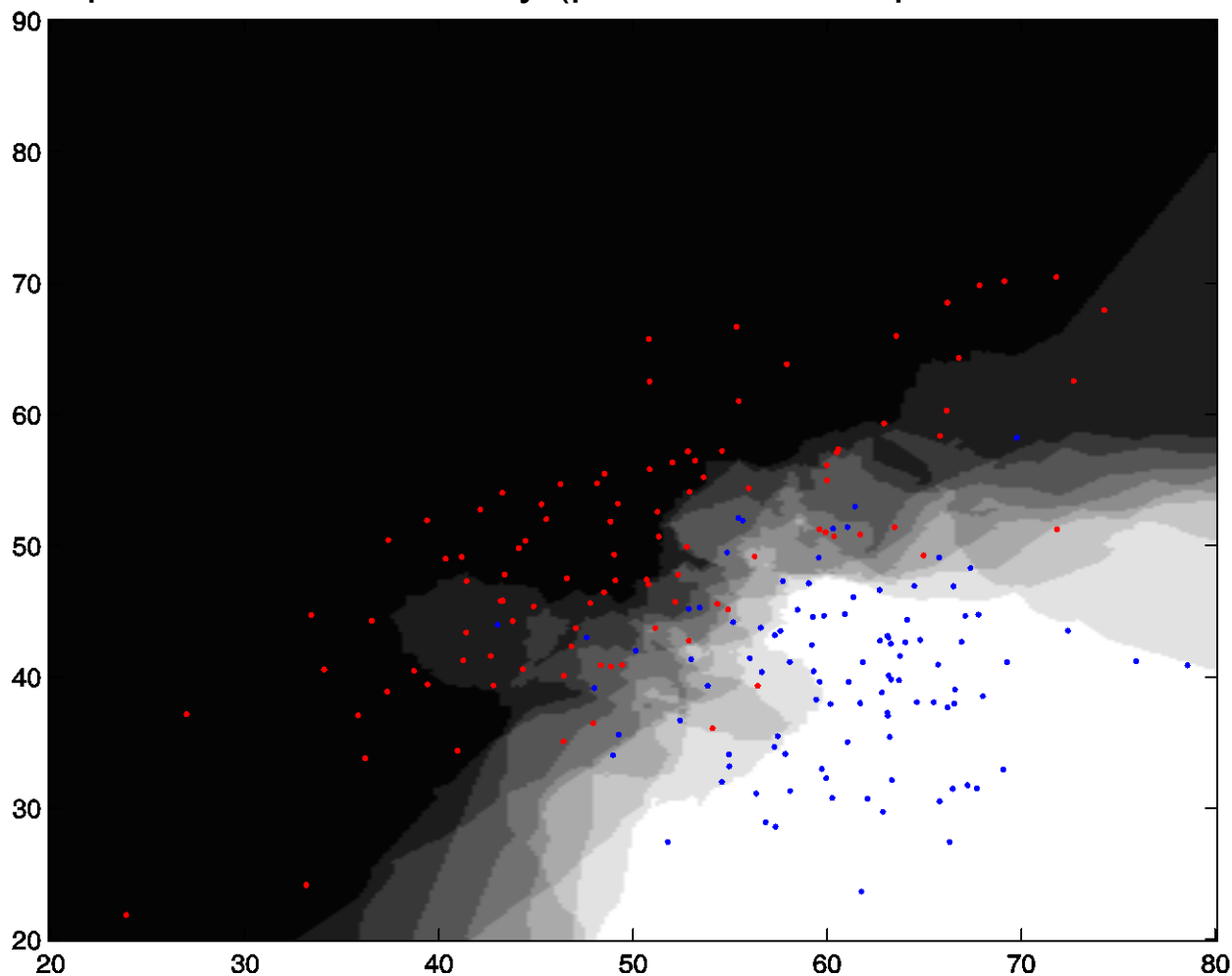
9-nejbližších sousedů

Rozhodovací linie už je dosti podobná optimální kvadratické křivce

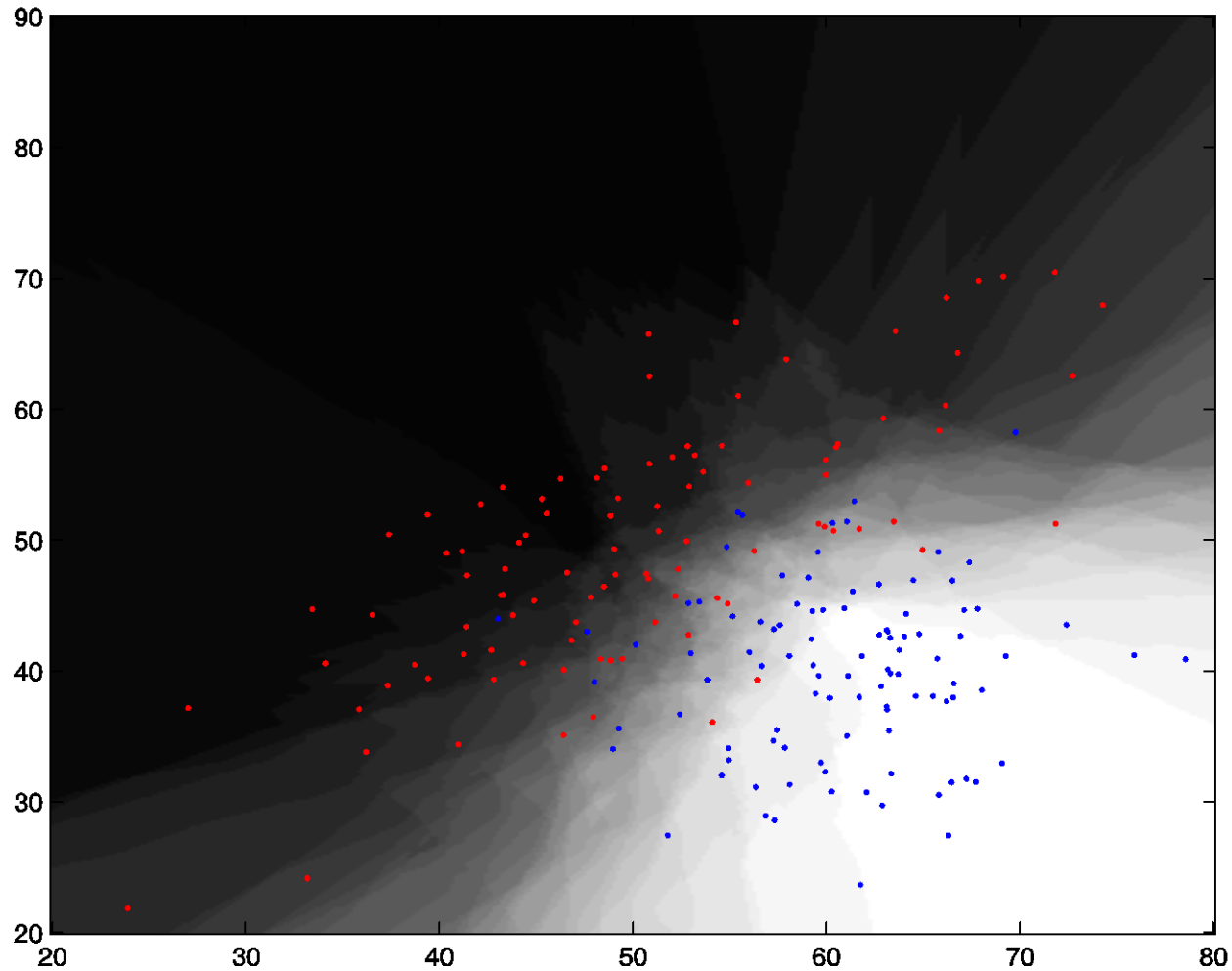


9-nejbližších sousedů měkké rozhodnutí

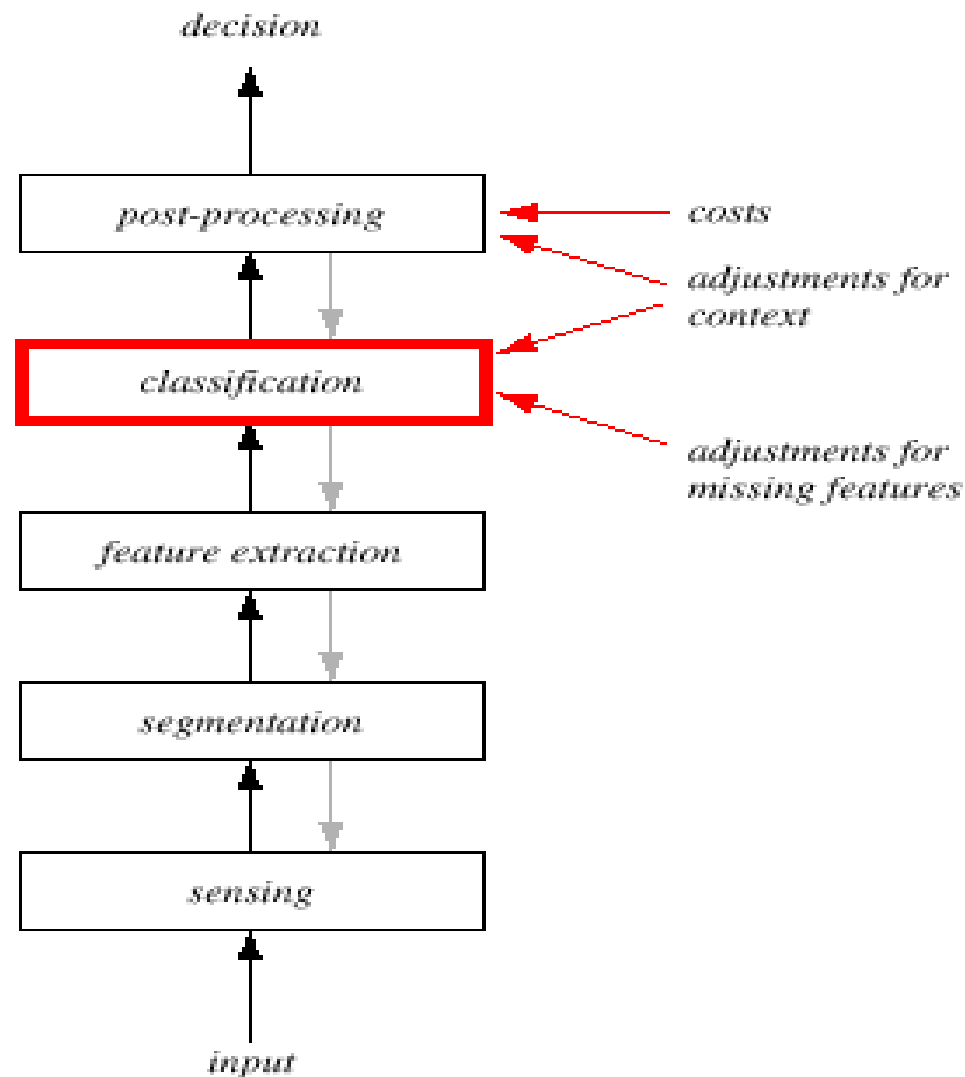
Místo tvrdého rozhodnutí můžeme použít poměr mezi počtem sousedů z různých tříd jako „měkké“ měřítko důvěry (confidence), že vzor patří do té či oné třídy (pro $K=9$ máme pouze 10 úrovní).



31-nejbližších sousedů měkké rozhodnutí



Post-processing



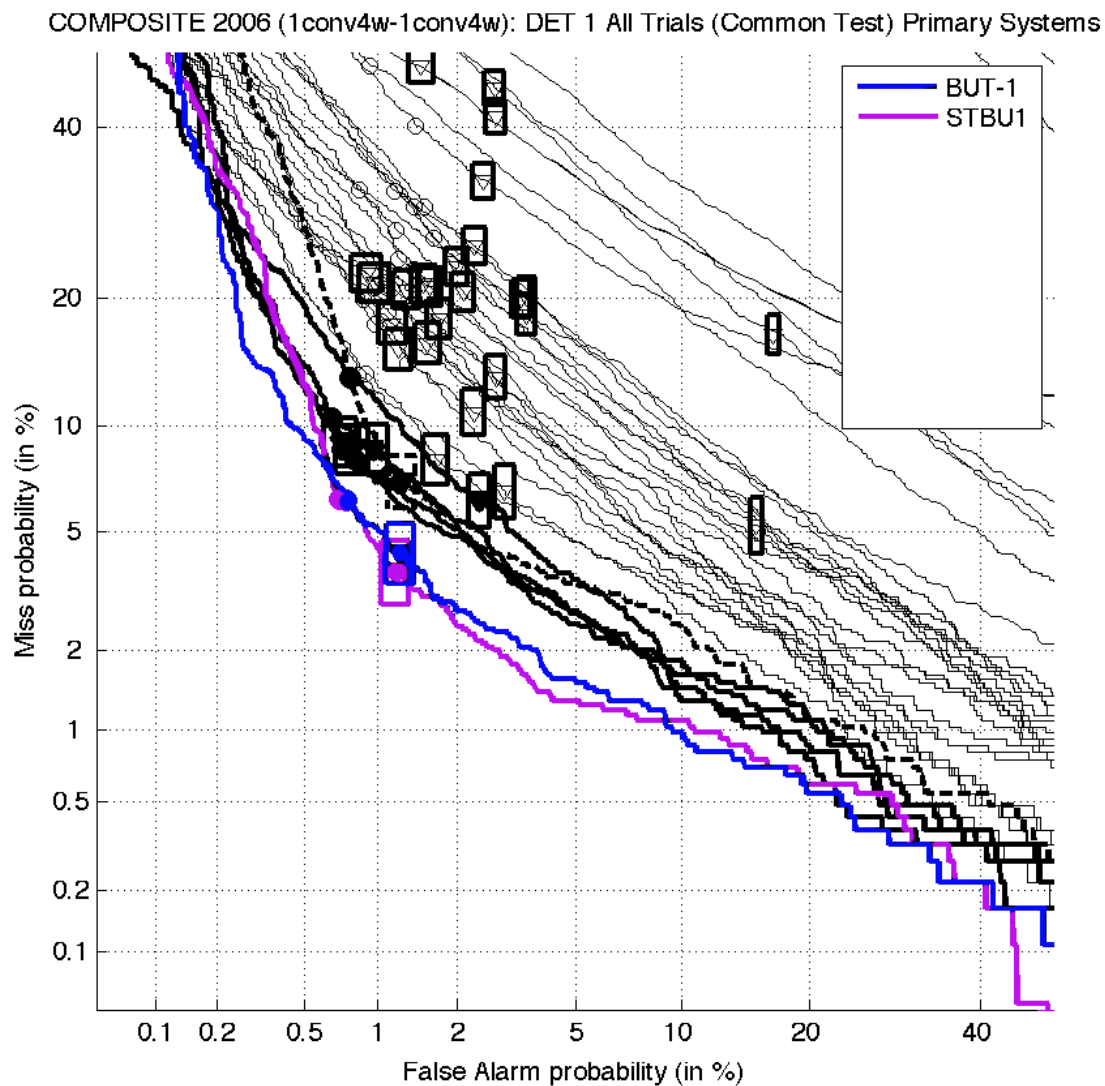
Post-processing

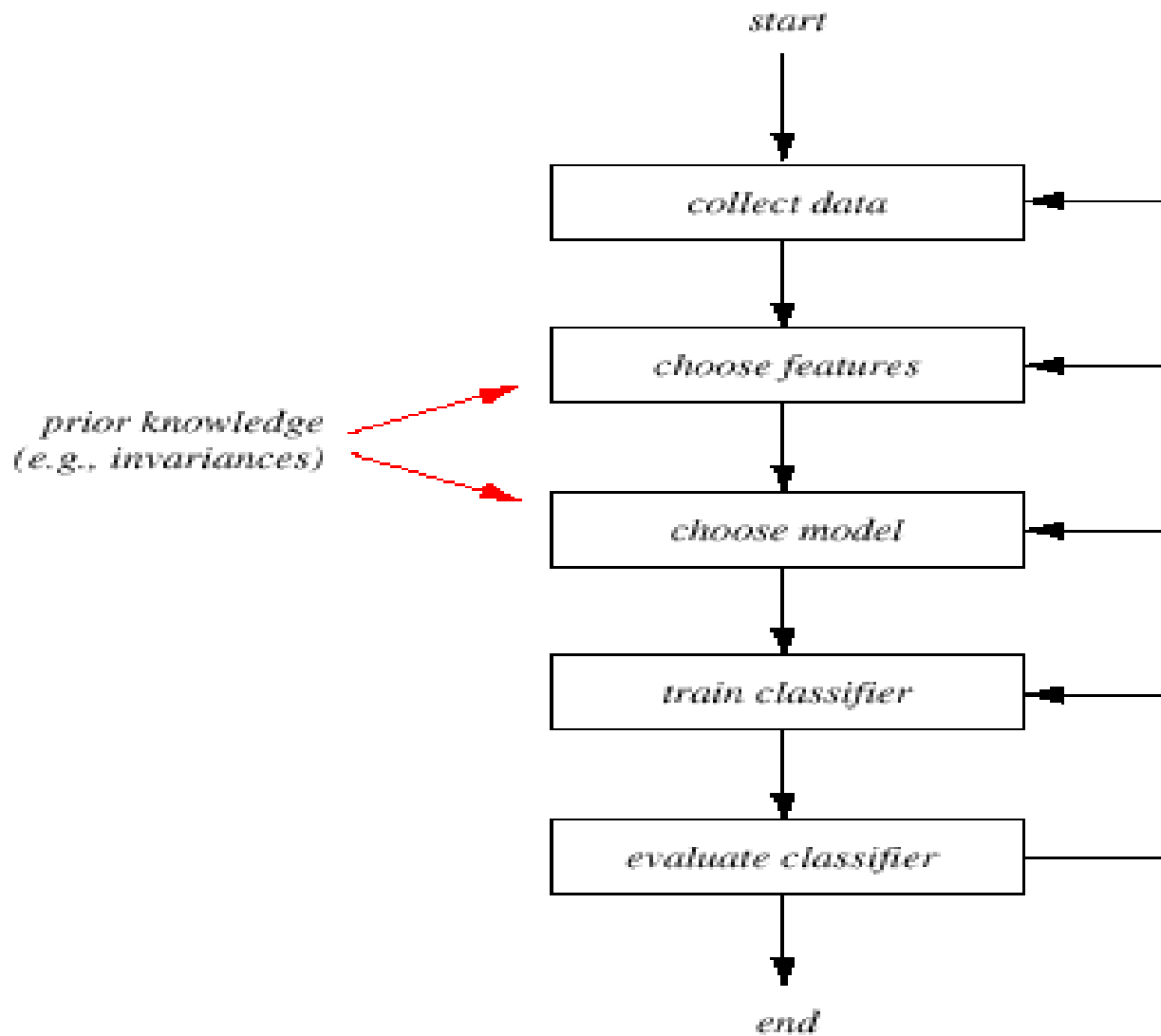
- Závislé na konkrétním úkolu.
- Využití dalších kontextových informací jiných než je samotný vzor
- Pokud je výstupem klasifikátoru měkké rozhodnutí, post-processing se může přiklonit jedné variantě než té s nejlepším skóre:
 - např. integrace apriorní pravděpodobnosti (viz další přednáška)
 - Můžeme brát v úvahu **ceny** jednotlivých rozhodnutí. Co nás bolí víc? Poslat jablko pyrotechnikovi nebo granát do marmeládovny.
 - Rozhodnutí pro konkrétní třídu pokud její skóre překročí jistý práh → Detekční úloha

Identifikace vs. detekce

- Identifikace → vyber jednu z N možných tříd
- V příchozích vzorech detekuj ty, které paří do třídy, kterou hledáme.
 - Vzory, které detekovat nechceme nemusí patřit do omezeného tříd (např. v telefonních hovorech hledáme hlas konkrétního mluvčího mezi hlasy všech možný mluvčích)
 - Detekci proved' na základě **měkkého rozhodnutí** – **skóre** – a nastaveného **prahu**.
 - Detekční práh je možné měnit podle požadované aplikace:
 - Práh nastavený nízko → Hodně detekcí ale také hodně planých poplachů
 - Práh nastavený vysoko → opačný problém

Detection tradeoff (DET) křivka





The Design Cycle

- Data Collection
 - How do we know when we have collected an adequately large and representative set of examples for training and testing the system?
- Feature Choice
 - Depends on the characteristics of the problem domain. Simple to extract, invariant to irrelevant transformation insensitive to noise.
- Model Choice
 - Unsatisfied with the performance of our fish classifier and want to jump to another class of model
- Training
 - Use data to determine the classifier. Many different procedures for training classifiers and choosing models
- Evaluation
 - Measure the error rate (or performance and switch from one set of features to another one