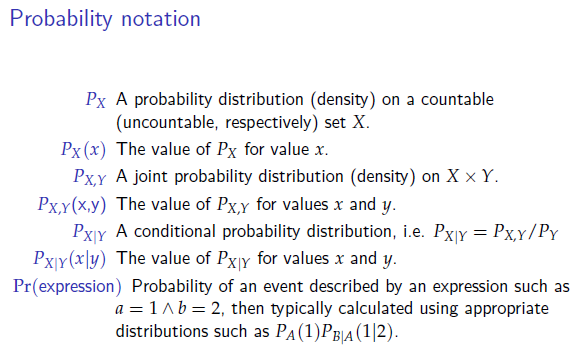
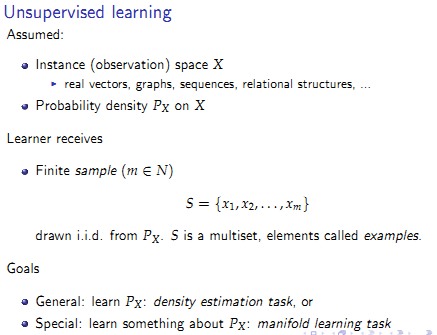
# 14. Výpočetní teorie učení: rozměr hypotézového prostoru, PAC-naučitelnost. Naučitelnost výrokových konjunkcí a disjunkcí.

**** 

**(3)** **unsepervised learning**

**unsupervised** - bez ucitele.. ukazuju vzorky a klasifikator si je nejak mapuje, shlukuje apod. aniz by vedel co to je

S je asi Sample, xi jsou vlastnosti

intuitivne bych rekl ze to bude neco jako shlukovani - proste analyza dat bez toho aniz bych vedel do ktere tridy patri

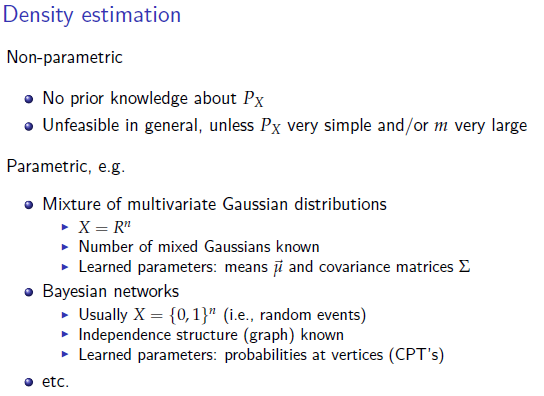
**manifold** – **různý, všelijaký, četný** (lerning task)

Unsupervised learning is closely related to the problem of density estimation in statistics. However unsupervised learning also encompasses (zahrnuje) many other techniques that seek to summarize and explain key features of the data.  
One form of unsupervised learning is clustering. Another example is blind source separation based on Independent Component Analysis (ICA).

**unfeasible – neproveditelný, nemožný**

proste bez ucitele (aniz by mi nekdo rikal co je co) si ty examply nejak strukturovat podle podobnosti

jj, to je poloucitel, protoze tady sice nerikas primo, ze tenhle vzorek je neco, ale jen rikas, ze tenhle shluk (mnozina vzorku je neco jineho nez jina mnozina vzorku nebo shodna)

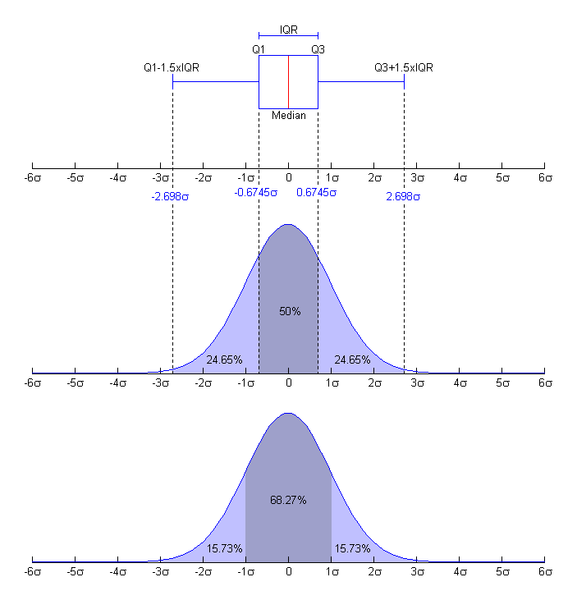


**(4) density estimation**

**prior – dřívější**

jde asi o to, ze treba na tom obrazku na wikipedii, je v miste -sigma ... sigma hustota vyssi nez treba jinde

takze treba u takoveho **k-means** bude hustota vyssi v centru shluku – vetsinou



takze muze to byt bez parametru nebo s parametry, bez parametru je to horsi, s parametry je to lepsi pokud vis jak data vypadaji, tak by to bylo asi presnejsi

**gaussian** bude asi takovy, ze se ti budou shlukovat data s nejakym tim gaussovskym shlukem

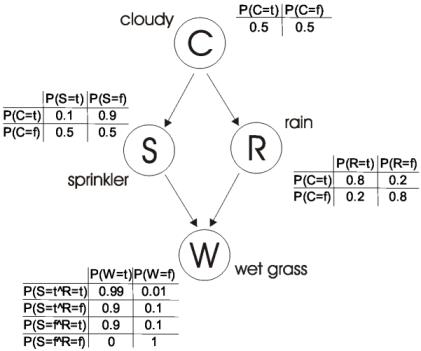
u k-means se ti pohybuji stredy shluku a prislusnost ke shluku je bud JE nebo NENI (0 / 1)

u gaussian to bude asi podle vzdalenosti od stredu, cim blize ke stredu tim vic ANO, cim dale, tim vic NE

jasný ... rozmezí 0 – 1

**bayesian networks** to jsou site (jako graf s sipkama) a uzly predstavuji mnozinu nejakych priznaku a pravdepodobnosti, treba koureni zpusobuje rakovinu tak budes mit uzel Koureni a z toho udelas sipku Rakovina

a pak tam nejak nastavis pravdepodobnosti, takze pak kdyz o tom cloveku neco vis, ze treba ma kasel, kouri, byl ockovany proti TBC, tak nejspis ma rakovinu



tak se koukni na tabulku S, prvni radek: kdyz je cloudy, tak: 10% je postrikovac zapnut a 90% je postrikovac vypnut

jasne, proc by ho zapinali, kdyz kazdou chvili zacne prset

kdyz neni zamraceno, tak je to 50 na 50

ted rain R, kdyz je zamraceno, tak z 80% zaroven i prsi a z 20% ne, kdyz neni zamraceno, tak je to obracene

a pak tabulka W, tam jsou vsechny moznosti dohromady a jejich pravdepodobnosti

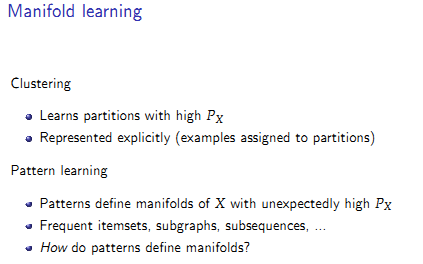
**OBECNE**:

density estimation muze byt bez parametru a s parametry, bez parametru to vetsinou dobre nejde

s parametry: priklady parametru: pocet shluku, u **gaussianovskych shluku** bych asi rekl sigma apod.

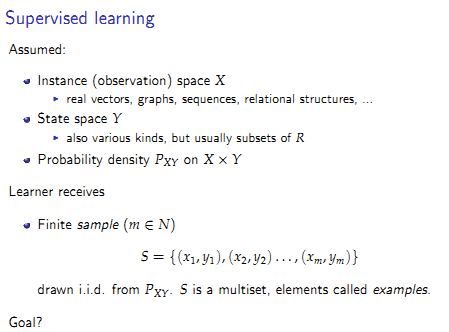
tady muze byt X = R^n

tedy sample je tvoren vektorem z realnych cisel nebo **bayess**, vektor 0 a 1



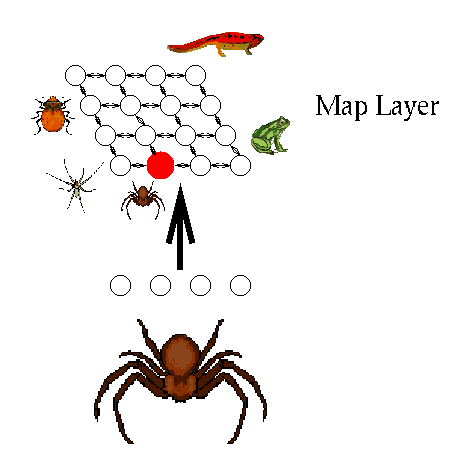
(5) **Manifold learning**

asi jsou 2 moznosti - clustering a pattern learning, bayesian network by mel byt asi ten pattern learning



(6) **Supervised learning**

supervised je s ucitelem - tedy nekdo musi ten algoritmus ucit, budu ukazovat vzorky a rikat co to je

**Unsupervised**:****

ukazuju vzorky a ten algoritmus si je na zaklade podobnosti dava do nejake struktury.. pritom nevi co je co, proste podle vzhledu

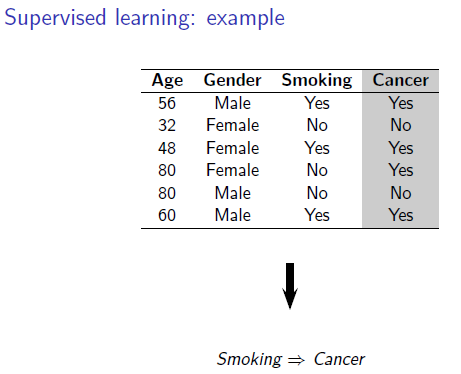
**supervised:** je ze mu ukazu vzorek a reknu - "Tohle je pavouk"

ukazujeme vzorky a rikame co to je, tim si ten klasifikator muze vypocitavat nejakou pravd. ze pokud je napr. sample v nejakem shluku, tak s jakou pravdepodobnosti bude patrit do nektere tridy

s **poloucitelem** bych mu mohl rict, ze vsechny vzorky, co maji vic jak 4 koncetiny urcite nejsou stejne jako ty co maji 4 koncetiny a ten alg. pak pozna ze hmyz je neco jineho nez savci treba

a supervized a poloučitel se nerovná? no u poloucitele nerikam, primo u vzorku co to je

**supervised** = s ucitelem  
**unsupervised** = bez ucitele  
a asi **semi-supervised** bude s poloucitelem

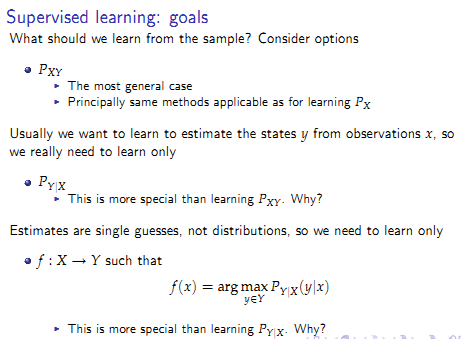


**(7)** **Supervised Lear. Example**

**cancer - rakovina**

neco o tom cloveku vim (treba z ankety) a reknu tomu alg. at se nauci, ze ten clovek ma rakovinu s temihle parametry, pak kdyz ten program uvidi cloveka, urci jeho vek, pohlavi a jestli kouri, tak odhadne pravd. ze ten clovek ma rakovinu

vlastnosti xi a naucena vlastnost: ma / nema rakovinu



**(8)**

Pxy = P(x ) \* P(y), takže y je stav a x je pozorování, tak bych se podival na Y (cancer) tedy stav, 4 ano, 2 ne, P(Y) = 4/6, P(Y) = 4/6

co je X? mel by to byt asi binarni priznak - plati / neplati

u veku je to celkem obtizne, jedine, ze bych rekl, ty co jsou starsi nez 50 treba a mladsi

tak at X je treba Gender, male = 1, female = 0, tak to bude P(x) = 3/6, takze P(xy) = 3/6 \* 4/6 = 0.333

jinak:  
P(x|y) = P(xy)/P(y)

a tady to mame obracene

P(y|x ) = P(yx)/P(x)

**Pxy = Px\*Py = Py\*Px = Pyx**

takze tady bych to vydelil jeste 4/6

takze to by vyslo 0.5

a zda se, ze je vyhodnejsi zjistovat P(y|x ) nez P(xy), zrejme vice vypovidajici

jj, ale hlavne P(xy) = P(x ) \* P(y) = P(y ) \* P(x ), takze X a Y jsou rovnocenne, ale **P(y | x) != P(x | y)**

co je to f:X -> Y?

je tam **argmax pres vsechny y**, takze bych to videl takhle**,** str. 7

y = ANO / NE, takze pro y = ANO, spocitam P(y |x ) = 0.5

pro Y = NE spocitam P(y | x ) = ... a vyberu z toho **maximum**

a ucit se tohle je pry jeste lepsi nez P(y | x )

takze potrebujeme zjistit P(Y|X)

tedy kdyz zname vlastnosti x\_i, tak abysme mohli urcit jaka je pravdepodobnost, ze bude Y

podminena pravdepodobnost

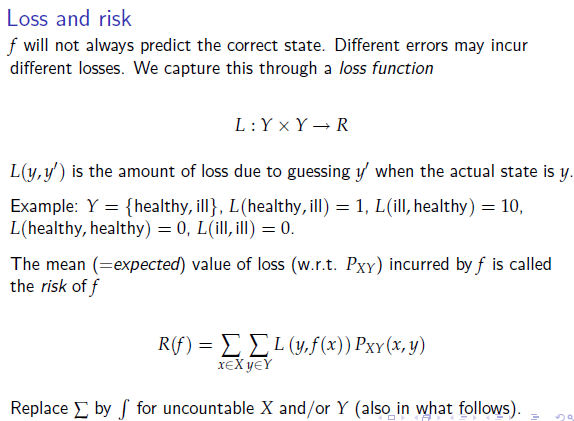
a jeste lepsi je asi maximalizovat to co je pod tim

ne, to je to same nejspis, ale jinak zapsane, obecneji

f(x ) je klasifikator, nejaka funkce, ktera dostane na vstup vlastnosti x\_i

a vrati odpoved Y a ta odpoved je klasickej argmax, tedy to co ma vyssi pravdepodobnost vyhrava

P(JE RAKOVINA | X ) = 0.6  
P(NENI RAKOVINA | X ) = 0.4 tak vyhraje Y = JE RAKOVINA



(9) **Loss and Risk**

Loss je tedy funkce ktera pro spravne Y a odhadnute Y' priradi hodnotu,ktera rika, jak velke chyby jsem se dopustil

kdyz nekdo byl zdravy a program odhadnul ze je nemocny, tak je **chyba 1**

kdyz je to obracene a o nekom kdo je nemocny jsem rekl, ze je zdravy tak je **chyba treba 10** (protože to je horší než to předchozí za 1)

jinak je **chyba 0**, protoze se to odhadlo spravne

uz vidim jak to tam budem pocitat..dont forget calculator

tady je Y to cemu chceme naucit a X jsou nejake atributy, takze bych rekl, ze ne.. samozrejme pokud bys pak chtel misto rakoviny urcovat treba pohlavi, tak bys to obratil

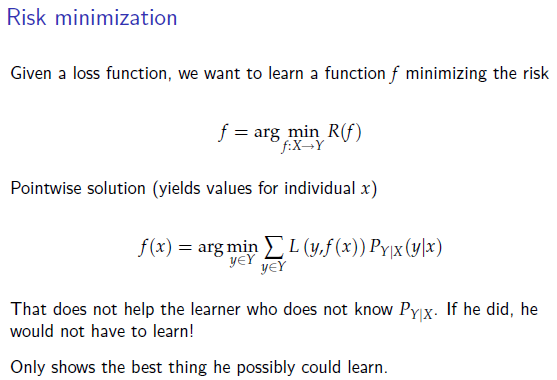
**jo a f tedy vraci y s maximalni pravdep.**

pak mame risk of **f** a to je proste ten vzorec a jsou tam dve sumy ..

mohou pri vyhodnoceni nastat chyby, takze potrebujeme chybu klasifikace zachytit

takze loss funktion muzeme prirazovat ruzna cisla, treba 0 kdyz se trefi a vetsi nez 0 kdyz se netrefi

Risk, funkci L vynasobujeme necim mezi 0 a 1 a scitame, R(f) bude nejaka chyba, ktera je dana tim, jak casto se chybne projevuje L, P je pravdepodobnost ze nastane jev x,y (oba zaroven) a L je chyba, kdyz nastane takovy jev, takze v podstate nascitas ty chyby pres vsechny jevy



(10) **Risk minimization**

takze minimalizujeme risk, tady je ta f nejaka upravena, predtim na 8 byla argmax P(...), tady argmin L\*P, takze P vynasobime tou ztratou a to chceme minimalizovat

takze chapal bych to tak, ze P predstavuje jakousi cetnost, jak casto k tomu jevu dochazi a vynasobime to tou ztratovou funkci L, aby jsme ziskali celkovou ztratu a tu minimalizujeme

risk tedy predstavuje chybu a my chceme takovy klasifikator, aby mel risk minimalni, to vyjadruje ten prvni argmin

**dejme tomu X je kouri / nekouri a Y je nemocny / zdravy**

zeptas se 100 lidi

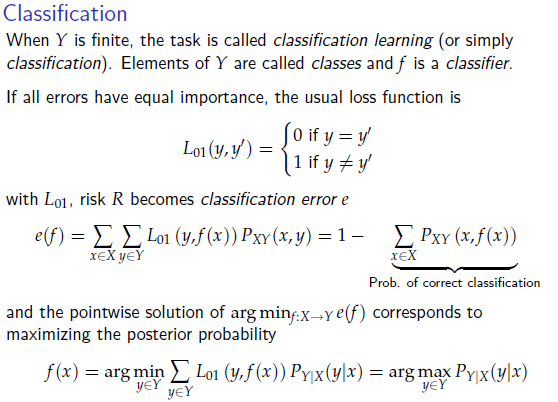
a 60% z nich kouri a je nemocnych  
10% kouri a je zdravejch  
20% nekouri a je zdravejch  
a 10% nekouri a je nemocnych

to jsou vsechny jevy, kazdy jev ohodnotim chybou L a urcim risk

pak mame tu sumu z Risku zapsanou trochu jinak jako f(x ) - klasifikator s minimalnim riskem a rekl bych, ze tady bude y|x misto xy proto, ze uz tu neni ta druha suma pres x

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**A supervised learning algorithm analyzes the training data and produces an inferred function, which is called a classifier** (if the output is discrete, see classification) **or a regression function** (if the output is continuous, see regression)**.**



**(11)** **Classification**

**finite – konečný**

y is finite = proste ze mas omezeny pocet trid, takze kdyz je **pocet trid** **omezeny**, tak je to klasifikace, tady je pocet trid 2 - **ma rakovinu / nema rakovinu**

f se nazyva klasifikator

pokud všechny chyby mají stejnou dulezitost pak plati tam to pod tim

R risk

jednou v tom vzorci maji Pxy a podruhe P(x|y), prostě jednou je to obecnější a pak specifičtější

ten vzorec 1-suma...tak bych ho chapal takhle, 1 je 100%,

x pres vsechny X bych chapal, ze prochazime vsechny vzorky v nejake sade a pocitame pravdepodobnost, ze klasifikace byla spravna ci spatna a pak to odectem od tech 100% a mame chybu

dalo by se totez spocitat asi i tak, ze bych prochazel vzorek po vzorku a pocital, kolikrat to dopadlo spatne

treba mame 1000 vzorku a napocital bych 50 chyb, tak to e = 50/1000, coz je chyba 5%

ne tady je chyba jenom 01, to je nad tim

ještě mi uniklo, proč se to odečítá od 100%

protoze ta pravdepodobnost v sume co tady maji neni chyba ale uspesnost, tedy 1 - uspesnost = chyba

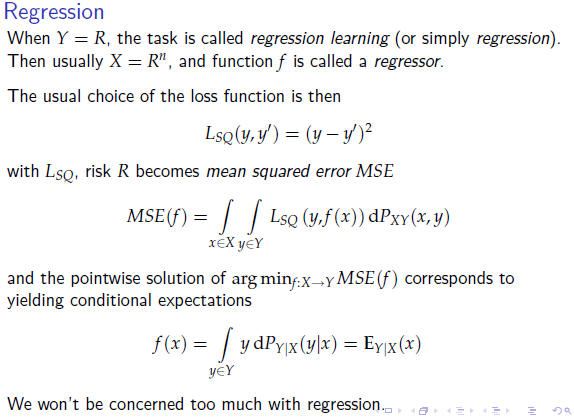
oni pocitaji spravne vzorky je jich treba 950 z 1000, tak udelaji 1 - 950/1000 = 50/1000 = 5%

tam to maji napsane pod sumou prob. of correct classific.

takze ten klasifikator f chceme takovy, ktery **minimalizuje** chybu apak = takovy, ktery **maximalizuje pravdepodobnost**

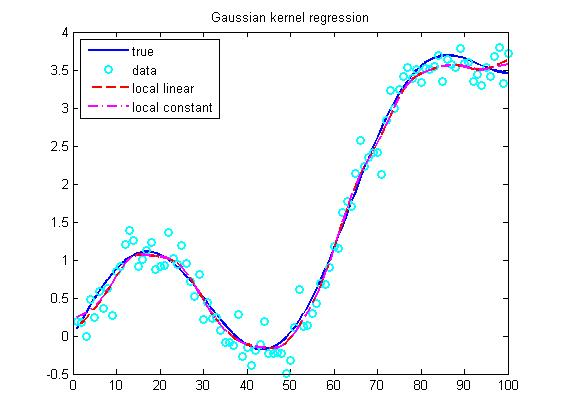
klasifikace - chybu L mame 0 a 1, error ma stejny vzorec jako risk a lze ho upravit na 1-..., kde ... predstavuje naopak uspesnost a dole zase napsali f(x ) podobne jak to bylo predtim jen ze tady uz je L jen 0 a 1

a pak tam mame argmax misto argmin, **protoze muzeme na to nahlizet jako na minimalizaci chyby, nebo jako na maximalizaci uspesnosti** jako treba na minimalizaci te funkce e(f) = 1- ..., nebo na maximalizaci toho ...



**(12)**

**Regrese** - vzorkami prolozit krivku



chyba je tedy L(y,y') = (y-y')^2, tedy vzdalenost bodu od krivky a tu chceme minimalizovat

to bude nějaká plocha pod tou křivkou protoležneou ... a ta se bude rovnat tomu E

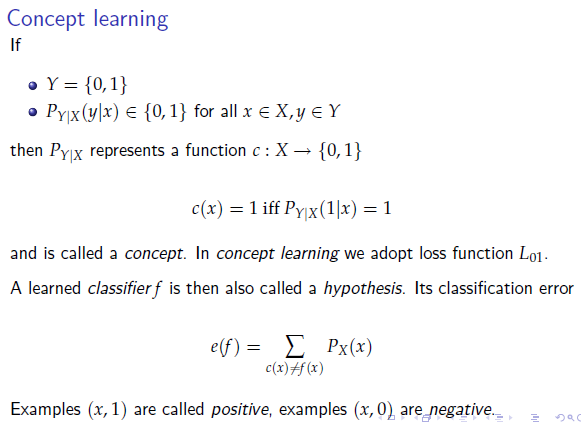
regrese - prolozeni krivky mezi body, y je tedy realne cislo (napr. v 2D grafu y-souradnice)

chyba je vzdalenost odhadnuteho y' a znameho y, pak par integralu, ktere odpovidaji tem sumam

protoze to **neni v diskretnich, ale realnych cislech**

chyba je vzdalenost odhadnuteho y' a znameho y, pak par integralu, ktere odpovidaji tem sumam

protoze to **neni v diskretnich, ale realnych cislech**



**(13)** **Concept Learning**

koncepty jsou asi nejake skupiny prvku s podobnymi vlastnostmi, treba obdelniky, nebo savci apod.

jsou nejake atributy, ktere je charakterizuji, no jasne, proste nam jde o tvar a ne o barvu, rotaci ci zvetseni



tak mame treba ze maji backbone (asi pater), a u tech zvirat se zvysi nebo snizi neco (asi nejaka pravdepodobnost nebo tak neco), ale jak z obrázku poznají ... že má páteř? to je asi v tech atributech

mas vzorek a o tom vis: ma rohy - ano, ne / plave - ano / ne, ma pater - ano/ne, ...

a pak mas asi koncepty - ryba, hmyz, ..

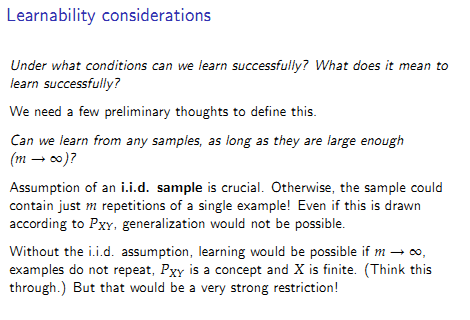
Obr. na dalším slide:

nema to nahodou souvislost s tim, jak jsme neco delali na cviceni, myslim ze se to taky jmenovalo koncept

ze jsi mel nejaky seznam: 1, -1, a ?

a bud neco platilo, neplatilo nebo to nezalezelo

Y je 0 nebo 1, **P(y|x ) je take 0 nebo 1** - tohle je zajimave, takze kdyz zname x, mame 100% jistotu? ne, jo bude to tak asi, proste to jsou ty koncepty, co jsme delali na cviceni a tam to tak bylo ne? ze koncept musi vyhovovat vsem vzorkum



**(15)** **Learnability considerations**

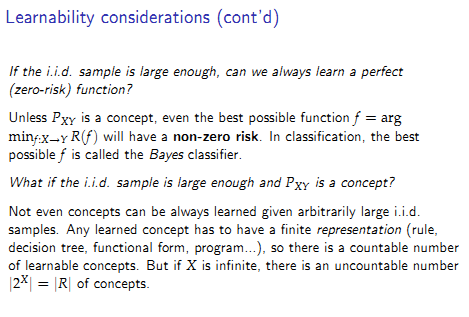
**i.i.d.** - Independent and identically distributed ( např. random variables ), nezávislé a identické rozložení

In probability theory and [statistics](http://en.wikipedia.org/wiki/Statistics), a [sequence](http://en.wikipedia.org/wiki/Sequence) or other collection of [random variables](http://en.wikipedia.org/wiki/Random_variable) is **independent**(nezávisle) **and identically distributed** (**i.i.d.**) if each random variable has the same probability distribution as the others and all are mutually independent.

**mutually - vzájemně**

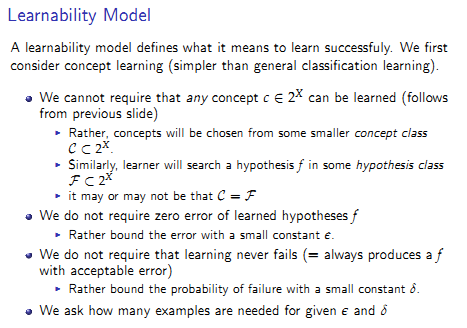
Předpokládejme, že i.i.d. vzorek je klíčový. Jinak by vzorek mohl obsahovat právě m representací jednoho příkladu! Dokonce i když je toto koncipováno v závislosti na Pxy, generalizace (zobecnění) nebude možný. :-D

Bez i.i.d. předpokladu, učení bude možné při m jdoucí k nekonečnu, přiklady se neopakují, Pxy je koncept a X je konečný. (Popřemýšlejte.) Ale tohle je velkmi silné omezení!



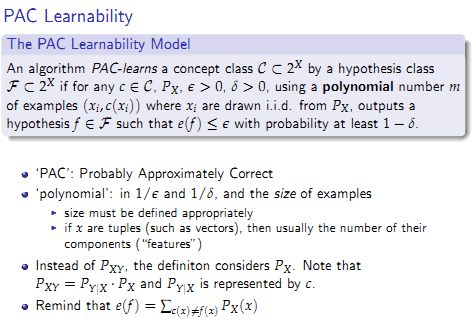
**(16)**

Pokud i.i.d. vzorek je dostatečně velký, můžeme vždy naučit perfektní (zero-riziko) funkce?  
Pokud PXY je koncept, který i té nejlepší možné funkci f = arg minf: X -> Y R (f) bude mít **non-nulové riziko**. V klasifikaci, nejlepší možné f se nazývá **klasifikátor Bayes**.  
Co když i.i.d. vzorek je dostatečně velký a PXY je koncept?  
Ani koncepty nemohou být vždy naučeny daným libovolně velkým iid vzorků. Jakýkoliv naučili koncept musí mít konečnou reprezentaci (pravidlo, rozhodovací strom, funkční podobě, program ...), tak tam je nepočitatelné číslo  
z learnable konceptů. Ale pokud X je nekonečný, je nespočetná číslo | 2x | = | R | konceptů.



**(17) Learnability Model**

L.M. definuje co to znamená se učit úspěšně. Prvně zvažujeme konceptové učení (jednodušší než obecné klasifikační učení).



**(18) PAC Learnability**

In [computational learning theory](http://en.wikipedia.org/wiki/Computational_learning_theory), **probably approximately correct learning** (**PAC learning**) is a framework for mathematical analysis of [machine learning](http://en.wikipedia.org/wiki/Machine_learning).

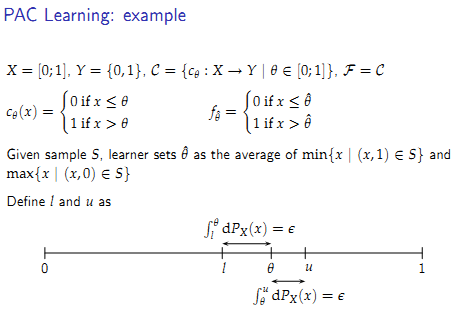
In this framework, the learner receives **samples** and must select **a generalization function** (called the ***hypothesis***) from a certain class of possible functions. The goal is that, with high probability (the "probably" part), the selected function will have low [generalization error](http://en.wikipedia.org/wiki/Generalization_error) (the "approximately correct" part). The learner must be able to learn the concept given any arbitrary approximation ratio, probability of success, or [distribution of the samples](http://en.wikipedia.org/wiki/Empirical_distribution).

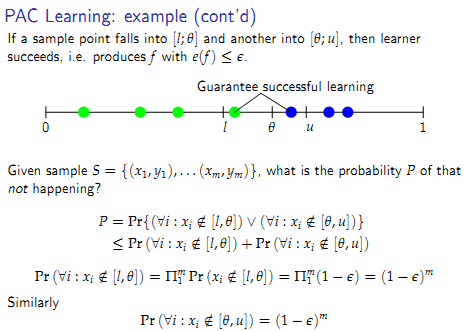
2^x

co to znamena? vektor nul a jednicek? nebo asi mnozina 0 a 1, R^n jsou vektory (R,R,R,...), 2^n jsou vektory (0-1,0-1,0-1,...), mnozina vsech techto vektoru

koncepty jsou ty vektory 0,1, plati neplati pro ruzna x, popr. jeste ten otaznik, kdyz nezalezi,

takze C je nejakapodmnozina nul a jednicek delky X, to same F, c z mnoziny C je koncept





**(19)** priklad

X = [0; 1]

Y muze byt bud 0 nebo 1

no bude to tak s tim l a u

ten integral tam, kdyz si to predstavis jako plochu, dejme tomu u Theta bude maximum, pak P smerem k okrajum klesa, integral bude plocha (suma vsech pravdepodobnosti) na tom intervalu a oni si to tady nazvali epsilon

ten integral tedy vyjadruje asi pravdepodobnost, ze bod spadne do toho intervalu

takze my si treba zvolime nejake epsilon, tedy rozsah kolem thety, a podle 20, rikame, ze pokud e(f) < epsilon tak je to spravne, ne epsilon, ale ze ten klasifikator rozpoznal spravne

takze C je nejakapodmnozina nul a jednicek delky X, to same F, c z mnoziny C je koncept

a delame asi nasledujici, udelame si samply (xi, c(xi))a naucime tim f

**PAC**

<http://www.cis.temple.edu/~ingargio/cis587/readings/pac.html>

ucime alg. a chceme aby dobre rozpoznaval

kdy zajistime, ze dobre rozpoznava?

kdyz bude nizka chyba

chyba je dana pravdepodobnosti jevu, ze odhadnuty vysledek je jiny nez ocekavany vysledek

jo? pokud bude pravdepodobnost, ze to dopadne jinak 100%, tak alg. je uplne spatne, protoze vsechno dela blbe

potrebujeme aby pravdepodobnost, ze se vysledek nebude shodovat byla mala

takze to ze shora omezime nejakym malym epsilon

treba epsilon = 5%

pokud je error < 5% tak rekneme, ze alg. se uci dobre

bude-li error > 5%, tak alg. se uci spatne

neni ta error nejaka krivka, ktera s vyšším poctem dat klesá?

jo s vyssi trenovaci mnozinou by mela klesat

ok, tak bude-li error > 5%, tak alg. se uci spatne

chceme, aby se neucil spatne

takze P(err > 5%) < delta

jeste podruhe to omezime nejakym delta, coz nevim proc, ale aspon si muzem pod tim epsilon a delta neco predstavit

no neni to ten upper a lower bound jak bylo v přednáškách? to bude ono

takze tohle je asi nejdulezitejsi poznatek:  
**Probability[error(h) > epsilon] < delta**

We are now in a position to say when a learned concept is good:  
When the probability that its error is greater than the accuracy epsilon is less than the confidence delta.

jasné

cim mensi epsilon a delta, tim lepe se alg. uci

protoze je nizzsi chyba

a problemem je urcit velikost trenovaci mnoziny

jakoze, kolik minimálně bude stačit dat, aby se to dobře naučilo?

jj, asi

asi taky zaleží na jakých datech trénujeme, neboť trenovat na jednom vzorku (stejnem) a mit jich jakoze nekolik by bylo zbytecne ...

na nahodnych

We would like to know how good is this learning algorithm. We choose epsilon and delta and determine a value for m that will satisfy the PAC learning condition.

takze je to presne tak jak pises, urcuji m - pocet vzorku v trenovaci mnozine

ok ... přičemž se snažím držet to m co nejmenší ...

****

oni tady resi problem prumerneho cloveka

vsichni lide jsou v tom velkem obdelniku

a rekli si, ze prumerni lide, budou ten mensi obdelnik uprostred

takze pokud alg. nejakeho cloveka mimo oblast h vyhodnoti jako prumerneho, tak je to chybne

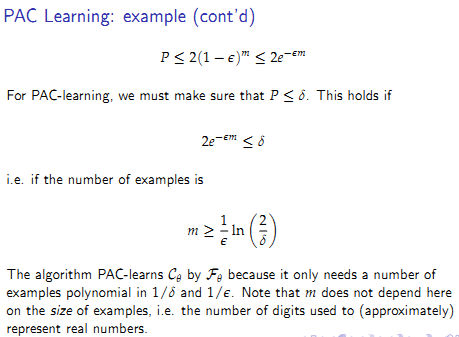
1 - epsilon/4 je pravdepodobnost, ze clovek bude mimo oblast h

tedy ze dojde k chybe

(1 - epsilon/4)^m - je pravdep. ze vsech m lidi bude mimo h

a pak to jeste omezi tim delta

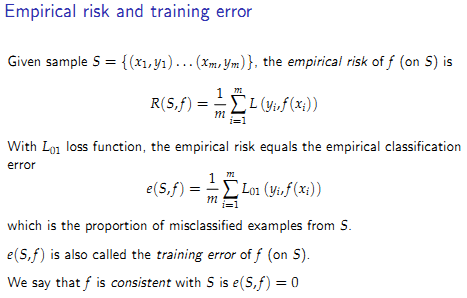
z toho pak vypoctou m



**(22)** prednaska colt

tam je taky takovahle podminka pro m vypoctena

takze chci-li aby se to dobre ucilo, zvolim nejake nizke epsilon a delta a vypoctu z toho kolik potrebuju trenovacich dat, tedy to m ..., jj



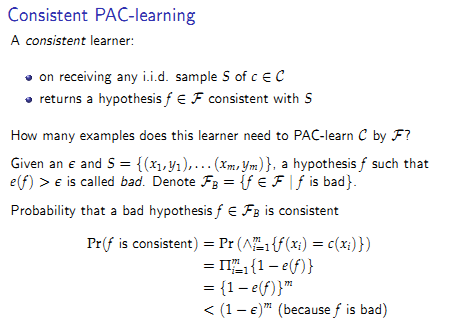
**(23)**

predtim byl Risk - vazeny prumer pres pravdepodobnost P(x,y)

nyni je risk obycejny prumer

protoze asi je to nahodne a kazdy vzorek ma stejnou pravdepodobnost vyskytu

We say that f is **consistent** with S if e(S, f ) = 0



**(24)**

tady se opiraji o to, ze hypoteza musi byt konzistentni s S

ale co znamena to konzistentni?

aha, to je asi na 23

zde ziskaji tu podminku konzistence P < (1-epsilon)^m

takze ta podminka dole rika, kdy je spatna hypoteza konzistentni

pro vetsi m ta pravdep. klesa, takze asi mame vyssi jistotu, ze je to dobre

nebo nevim, moc mi samotnemu neni jasne, co jsem ted napsal :D

no klesa horni omezeni pravdepodobnosti

epsilon je vetsi nez 0

takze (1-e) < 1

takze klesa s rosotoucim m

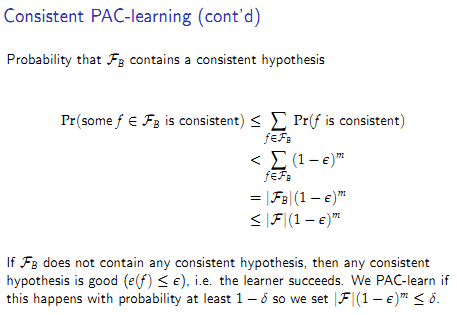
a co znamena ze to je konzistentni ??? to e(S,f) = 0

chyba fce f na semplech (test. datech) S je nula? jj

tak chyba klesa a kdyz pak budes mit treba 99% spravne a 1% chybu tak to presto **neni konzistentni**

a aby to bylo konzistentni tedy 100%, tak je pravdep. uz hodne nizka s velkou trenovaci mnozinou

no ja nevim.. mozna to tak je, mozna ne



**(25)**

F\_B je mnozina spatnych hypotez

tedy takovych f, ze e(f) > epsilon

tady se resi pravdepodobnost, ze mezi spatnymi hypotezami je nejaka konzistentni

a ta **hypoteza**, vlastnimi slovy, o co jde?

rekl bych, ze to je ten **klasifikator**

klasifikator vlastnimi slovy :-)

nejaky **matematicky model (funkce)**, ktery provadi klasifikaci

koukni na 23

tam je risk a je tam f(x )

**xi** jsou **vlastnosti** toho co chceme klasifikovat

**f** provede klasifikaci a vrati vysledek, ktery porovnavame s y a bud L nam vrati 0 nebo 1

predstavil bych si to tak, ze za f se skryva nejaka funkce, ktera na vstup dostane promenne x a vrati y

dostane priznaky a vrati vysledek klasifikace

y je treba - ma rakovinu / nema

takze v xi mu dam: kouri / nekouri, ockovanej / neockovanej, ...

a f tohle zhodnoti a vrati y = zdravy / nemocny

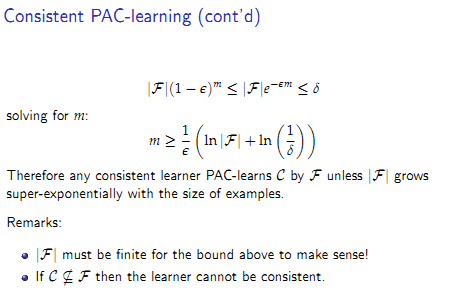
takhle mohu mit ruzna f, nektera to zhodnoceni provadi lepe, jina hure

nektera jsou zcela bezchybna a tem se rika asi konzistentni (na trenovacich datech !)

velke F je nejaka mnozina techto f

FB jsou spatne f, ktere maji chybu horsi nez epislon e(f) > epsilon

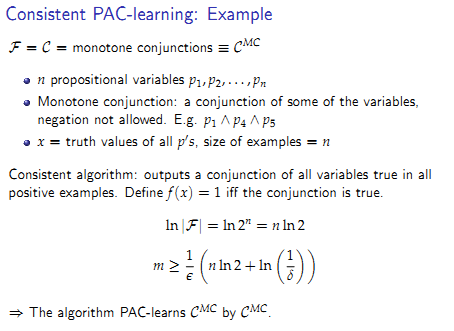
a na 25 tady odvodi zase jinou nerovnici

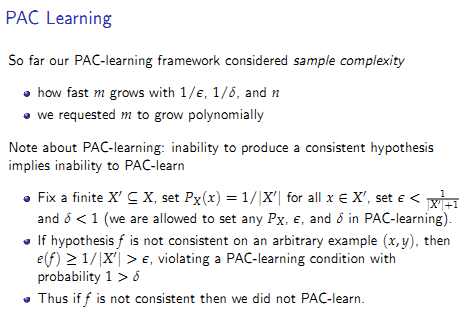


**(26)**

|F| roste super exponencialne, s poctem examplu, neni |F| neco jako to stirlingovo cislo?

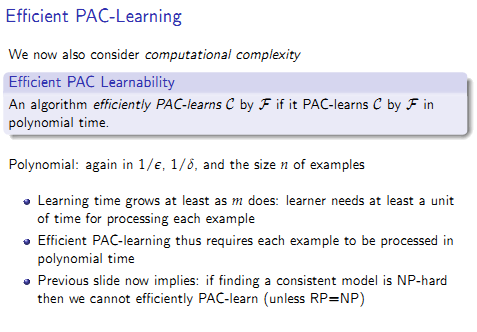
tedy ruzne moznosti jak rozdelit data do shluku, jako neni to to same, ale intuitivne to asi bude neco podobneho





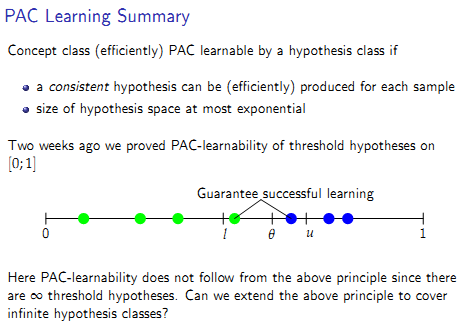
**(2)**

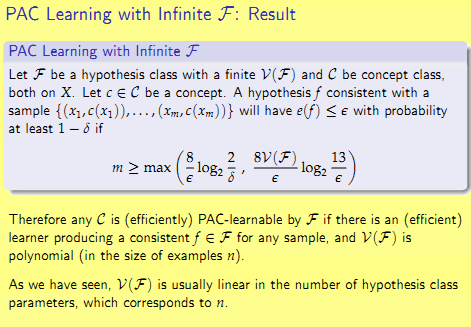
chceme aby m rostlo polynomialne, tedy pomale asypmtoticky ... no asi



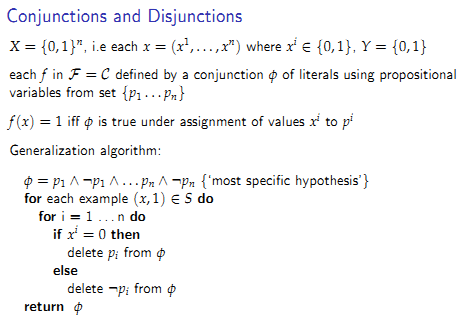
**(3)**

takze alg. se uci efektivne pokud se uci polynomialne, learn time roste aspon tak rychle jako m (pocet vzorku v trenovaci mnozine) a potrebujeme naucit na kazdem vzorku, takze nauceni jednoho vzorku chceme aby bylo taky polynomialni, no a pokud by se zjistilo, ze nalezeni konzistentniho modelu je NP hard, tak nemuzeme se ucit efektivne





## Naucitelnost conj+disj



**(4)**

to jsou kojunkce jako na cviceni

fi je nejake pravidlo z konjunkci a negaci

co dela generalization alg. nechapu

nejak upravuje fi

fi je nejaka kontradikce

nemuze nikdy byt true

je vzdy false na zacatku

mas tam treba:  
p1 a non p1

to je vzdycky false

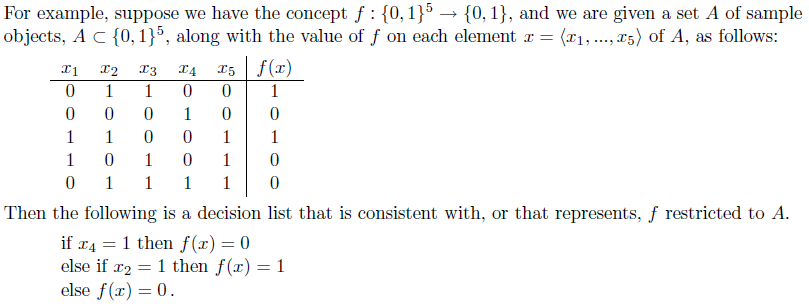
a pak pro kazdy sample delam:  
pro kazde n, neco kontroluju a odstranuju nektere casti z fi, aby to pak davalo smysl

kdyz xi = 0, tak odstranim pi a necham tedy non pi

kdyz xi = 1, tak odstranim non pi a necham jen pi

jj, to je neco podobneho jen jinak zapsane

tak fajn ... abych si to dovedl představit víš jak ...

****

na netu jsou proste podminky if else, pomoci kterych to mohu vyhodnocovat spravne

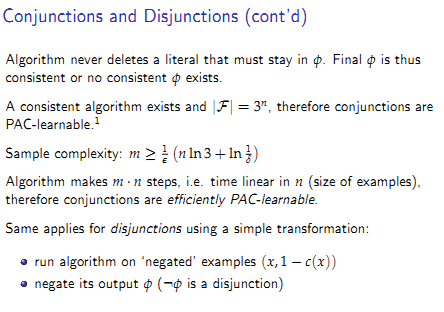
a v prednaskach nejaka pravidla, ktera kdyz jsou splnena tak je asi na vystupu 1 nebo 0.. nejsem si jeste jisty jak to funguje v prednaskach

pomoci generalizacniho algoritmu tedy sestavujeme tato pravidla fi

ty konjunkce jsou jedna jenom kdyz 1 conj 1 jinak nula

jako AND

jinak fi je asi neco jako: most specific hypothesis

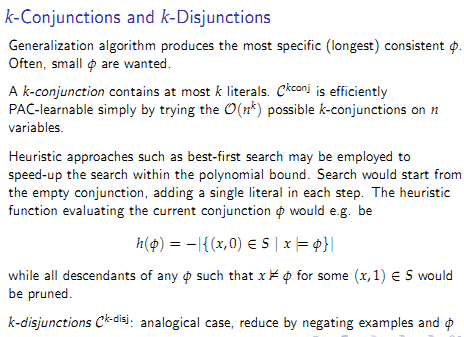


**(5)**

3^n je zrejme kvuli tomu, za v tom pravidle muze byt true, false a nezalezi

takhle:  
bud je tam pi  
nebo je tam non pi  
nebo tam neni vubec nic s tim pi

sample complexity je vzorec, ktery uz byl predtim v predchozi prednasce, jen tam bylo 2 misto 3



**(6)**

general. alg. vytvari nejdelsi konzistentni fi, čím delší, tím vice specifičtější .. jj, asi jo

k konjunkce obsahuje maximalne k literalu

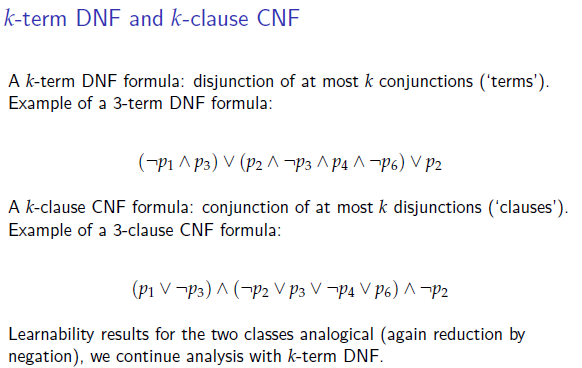
asi misto odebirani literalu je tady pridavame, hm?

asi ano...

ono by to mělo pak mít jakože stejný dopad na věc jako ta generalizace kerá mazala ... co myslíš?

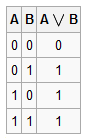
bud smazes a nechas to jedno a nebo naopak pridas a tim to prebijes ...

jj

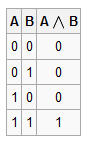


**(7)**

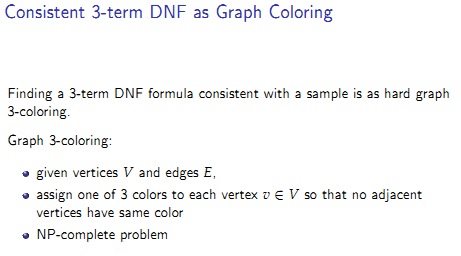
**Disjunkce, or**

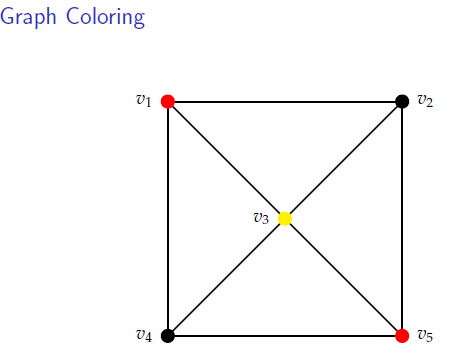
****

**Konjunkce, and**

****

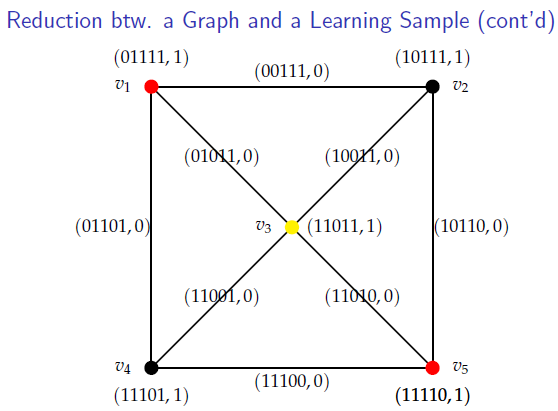
tak tady je priklad zapisu v DNF a CNF, tehoz vyrazu





**(8, 9)**

tak tady je problem barveni grafu 3 barvami, asi vime o co jde, ne? dál je příklad



**(11)**

pocet cislic je stejny jako pocet uzlu

tedy 5 cislic v tom prvnim cisle

nula je tam, kde je vrchol cisla

jo

hrana je taky jasna

dve nuly na mistech uzlu

vrchol ma nakonci 1 a hrana 0, jj

za tou čárkou až myslím ...

takze 0 znamena hrana a 1 znamena vrchol?

takže tohle je asi nějake struktura a nebo spíš před příprava pro ten alg, kerý to bude řešit. ... řel bych ..

jj, asi jo

a mozna jsem na neco prisel, ale nemam jistoti ze to tak je

vrcholy jsou jednotlive tridy klasifikace: napr: jablko, hruska, svestka, ...

kdyz klasifikator odhadne napr. ovoce spravne, tak skonci v nejakem vrcholu

treba jablko bude v2, a rozpozna ho spravne tak ma (10111,1) - pozitivni example

a treba v3 bude hruska

a pak mame negativni example

spletl si hrusku s jablkem

a tomu odpovida hrana (10011,0)

- negativni example

a proč 3 barvy?

takze by to popisovalo co s cim plete.. co myslis? odpovidalo by to tem nazvum negaticni a pozitivni

pro více barev, tedy tříd, je třeba větší graf ...

jako hezký ... to by mohlo být, hrana by spojovala tu záměnu ...

to je dobré ...

přemýšlím, proč proto využívat obarvující se graf ...

barevnost mi nejak popisuje ty konjunkce

kolik konjunkci potrebuju treba abych to mohl popsat

tak tady 3

pro větší grafy, vice barev ...

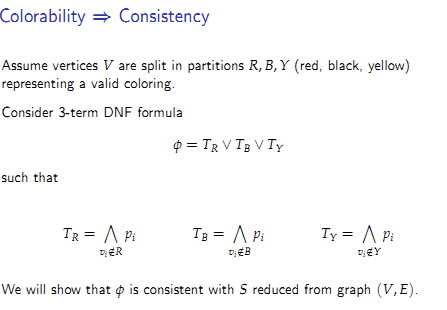
a vždy se bude jednat o uplne grafy, ze? jj

aby tam mohla byt hrana mezi zamenami ... asi jo

pokud si alg. muze poplest vsechno se vsim

pokud treba jablka a svestky si neplete, tak by tam treba hrana byt nemusela

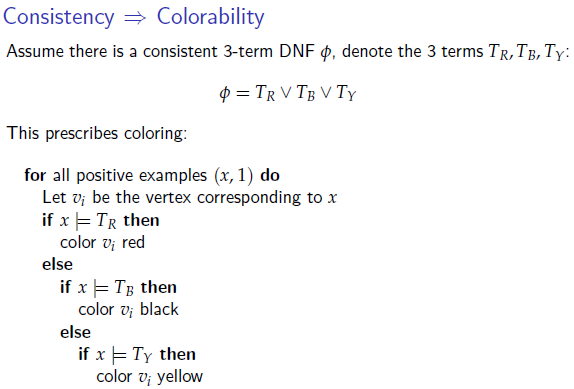
tak to zkusme znova projet to negaticni a pozitivni, jestli to na to bude pasovat



**(13)**

tady nejdriv ukazuji: Colorability => Consistency a dal na slajdech to ukazuji obracene

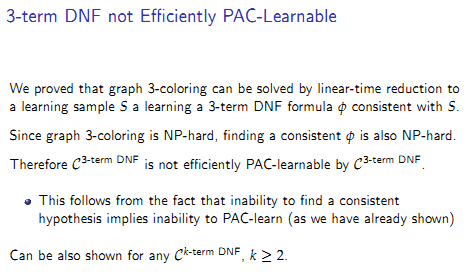
takze tady je co zname, zname nejake fi v DNF forme a ze mame graf obarveny 3 barvami



**(16)**

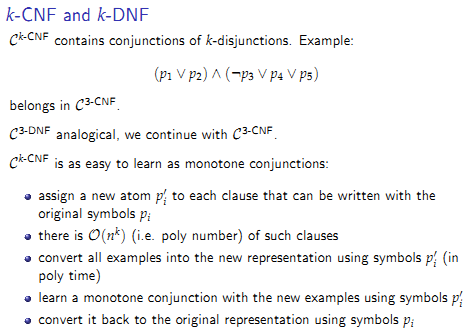
takže to fi, bude 1, aspon jeden z T bude 1

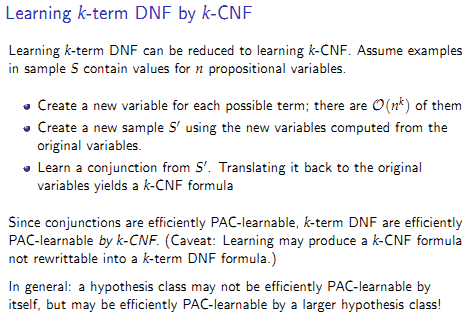
tak tady se jedná o barvičky a ty připadaj vrcholum, proto ta jednička ... alg jasnej



**(18)**

takze konzistentni uceni je stejne jako barveni grafu Nphard

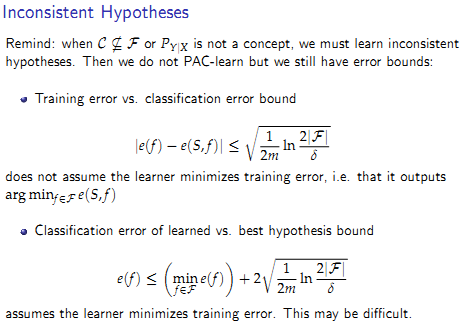


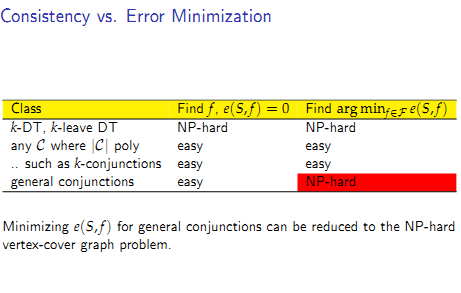


**(20)**

CNF a DNF muzes mezi sebou libovolne prevadet

to je blbost nejaka ne? ty dve vety si odporuji nahore a dole





**(36)**

takze tady je asi srovna ni nalezeni konzistentniho f a nalezeni nejlepsiho f, ktere nemusi byt konzistentni