ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ

Katedra technologií a měření

DIPLOMOVÁ PRÁCE

Možnosti využití statistických nástrojů pro analýzu infračervených spekter

vedoucí práce: **Ing. Pavel Prosr, Ph.D.**
autor: **Bc. Martin Vojáček**

2014

Anotace

Předkládaná diplomová práce je zaměřena na principy měření složení směsí, slitin, plynů, pomocí infračerveného záření a jeho následné měření infračervenými spektrometry, druhy spektrometrů a způsobů měření. V práci je rovněž zahrnuto několik statistických nástrojů, které se používají pro analýzu infračervených spekter, úpravy hodnot do jiných systémů nebo zařazování spekter do stejných tříd. V praktické části je navržen program pro analyzování dat pomocí analýzy hlavních komponent v programovacím jazyku C a jeho detailní popis.

Klíčová slova

Infračervená spektrometrie, infračervené záření, absorbance, transmitance, spektrometr, analýza hlavních komponent, diskriminační analýza, transmisní měření, reflexní měření

Abstract

The thesis is focused on analysis of infrared specters Firstly thesis deal with main principle of function of infrared spectrometers, propereties of infrared spectrum and measuring techniques with Fourier transform infrared spectrometry. Second part of thesis is focused on statistical tools for analysis infrared spectrum and description of more dimensional data. Third part of thesis deal with program written with C# programming language for analysis infrared spectrum by a principal components analysis.

Key words

Infrared spectrometry, Infrared radiation, absorbance, transmittance, spectrometer, principal component analysis, discriminant analysis, transmission measurement, reflectivemeasurement

Prohlášení

Předkládám tímto k posouzení a obhajobě diplomovou práci, zpracovanou na závěr studia na Fakultě elektrotechnické Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této diplomové práce.

Dále prohlašuji, že veškerý software, použitý při řešení této diplomové práce, je legální.

V Plzni dne 12.5.2014 Martin Vojáček

……..

Poděkování

Tímto bych rád poděkoval vedoucímu diplomové práce Ing. Pavlu Prosrovi, Ph.D. za cenné profesionální rady, připomínky a metodické vedení práce.

Obsah

Úvod

Předkládaná práce je zaměřena na funkci infračervené spektrometrie s užitím Fourierovy transformace a její využití pro následnou analýzu naměřených infračervených spekter. V současnosti je stále více kladen důraz na kvalitu materiálů, především kvůli zrychlování výroby, účinnosti, bezpečnosti a miniaturizaci, která je jedním z nejdiskutovanějších trendů.

Infračervená spektrometrie je v dnešní době jedna z nejpoužívanějších metod pro analýzu vzorků v laboratořích. Ve farmaceutickém průmyslu pro analýzu léků, v chemickém průmyslu pro analýzu rozpouštědel, plynů, past, vláken. Infračervená spektrometrie se rovněž využívá v elektronice, akademické sféře, ale velice pomohla i při vývoji biotechnologií. Fourierova transformace se využívala již od 50. let 20. století, nebyla však tak často využívána kvůli náročnosti na výpočet. Největší impuls ve vývoji infračervené spektrometrie přišel v 70. letech 20. století, kdy došlo ke spojení infračervených spektrometrů a osobních počítačů, což umožnilo využití Fourierovy transformace, zrychlení a zefektivnění výpočtů i měření hodnot. Od té doby začali spektroskopy s Fourierovou transformací nahrazovat do té doby používané disperzní spektroskopy, které byly mnohem pomalejší.

Infračervená spektrometrie je metoda zkoumající vibrace atomů v molekulách a jejich měření a následné analyzování. Atomy v molekulách materiálu začnou vibrovat při dodání energie infračerveným zářením. Energie, kterou absorbují atomy materiálu, se projeví na naměřeném spektru. Atomy různých materiálů vibrují při různých frekvencích infračerveného spektra. Na tomto jevu je založen princip infračervené spektrometrie.

Práce je rozdělena do třech částí, ve kterých je podrobněji popsána infračervená spektrometrie a nástroje pro analýzu. V první části práce je teorie o infračervené spektrometrii a infračerveném spektru obecně. Ve druhé části jsou podrobněji rozebrané nástroje pro analýzu a ve třetí části je rozebrán zdrojový kód programu pro PCA.

Seznam symbolů

FT-IR	Fourier transform infrared spektroskopy
PCA	Principal Component Analysis
DA	Disriminant Analysis
LDA	Linear Disriminant Analysis
QDA	Quadratic Discriminant Analysis
ATR	Attenuated Total Reflectance
DRIFTS	Difuse Refrection Spectroscopy
$W[cm^{-1}]$	Vlnočet
$\lambda[m]$	Vlnová délka
f[hz]	Frekvence
E[eV]	Energie fotonu
$h(6{,}626069 \cdot 10^{-34} J \cdot s)$	Planckova konstanta
c(299 792 458 m/s)	Rychlost světla
A[-]	Absorpce
T[%]	Transmitance
I_0	Intenzita pozadí
I	Intenzita absorbovaného světla
ε	Pohltivost
$\delta[cm]$	Optický dráhový rozdíl
$d_p[\mu m]$	Hloubka vniku záření
$n_v[-]$	Index lomu vzorku
$n_k[-]$	Index lomu krystalu
$\theta[°]$	Úhel lomu na fázovém rozhraní
d_E	Eukleidova metrika
d_H	Hammingova metrika
d_M	Minkowskiho metrika
d_{MA}	Mahalanobisova metrika
S_{SM}	Sokalův-Michenerův koeficient podobnosti
S_{RR}	Russelův-Raoův koeficient podobnosti
S_H	Hamannův koeficient podobnosti
$\overline{X}$	Aritmetický průměr množiny X
X_i	i-tý prvek množiny X

s	Výběrová směrodatná odchylka
Cov()	Kovariance
C	Matice kovariancí
C#	Programovací jazyk

1 Infračervená spektrometrie

Infračervená spektrometrie je analytický nástroj využívaný hojně v různých výrobních i nevýrobních odvětví. Některé podniky infračervenou spektrometrii využívají při kontrole dodávek materiálu od dodavatele. Zrychlí tak přejímku dodávek a zrychlí celý výrobní proces.[1]

Metoda infračervené spektrometrie je založena na interakci molekul s infračerveným zářením, kdy dojde k absorpci energie. Tuto změnu energie v odraženém spektru měříme pomocí spektrometru. [1]

1.1 Historie infračervené spektrometrie

Počátky infračervené spektrometrie sahají až do 18. století, když v roce 1800 bylo objeveno infračervené záření sirem Williamem Harschelem. Joseph Fourier učinil objev, díky kterému je možné vyjádřit časově závislý signál pomocí harmonických funkcí sinus a cosinus. Tím umožnil převod signálu z časové oblasti do oblasti frekvenční. Díky těmto objevům mohl mezi roky 1903 a 1905 William Coblentz naměřit stovky infračervených spekter mnoha organických i anorganických sloučenin. Začala se tím postupně vytvářet první knihovna vzorků infračervených spekter. Na konci 19. století byl zkonstruován první interferometr, jehož návrhářem a konstruktérem byl Albert Abraham Michelson, který byl v upravené verzi použit pro Michelson-Morleyův pokus roku 1887. Interferometr byl použit pro experiment vlivu éteru na rychlost světla. Žádné zpomalení však nebylo prokázáno a došlo k velké revizi fyziky a vytvoření speciální teorie relativity. První infračervený spektrometr byl sestrojen v 30. letech 20. století. Pro další vývoj infračervené spektrometrie přispěl především velký rozvoj průmyslu. V roce 1959 R.Barer se svými spolupracovníky spojili spektrometr s mikroskopem a tím opět vylepšili a zpřesnili analýzu infračerveným zářením. Největší využití infračervené spektrometrie přišlo v 80. letech 20. století, kdy se infračervený spektrometr spojil s osobním počítačem, a využila se rychlá Fourierova transformace, kterou objevili v roce 1959 James Cooley a John Turkey. [2]

1.2 Infračervené záření

V přírodě se můžeme setkat se všemi vlnovými délkami světla. Světlo je elektromagnetické vlnění. Vlnové délky, které můžeme vidět, jsou v oblasti od 400 do 800 nm. Na obrázku 1 je vidět celé spektrum světla rozdělené po určitých částech. Parametry

světelného záření jsou vlnová délka λ[m] (délka jedné periody vlnění), vlnočet W[cm^{-1}] (počet vln za určitou vzdálenost) a frekvence f[Hz] (počet period za jednotku času).

Nejčastější dělení v infračervené oblasti je podle vlnočtu W. [3]

$$W = \frac{1}{\lambda}[cm^{-1}] \quad (1)$$

Rovnice č.1 popisuje vztah mezi vlnovou délkou a vlnočtem.

Vlnovou délku jedné periody je možné spočítat ze vztahu:

$$\lambda = \frac{c}{f}[m] \quad (2)$$

Kvantová fyzika říká, že světlo je emitované zdrojem v jednotkách, které se nazývají fotony. Foton při kolizi s molekulou látky předá energii (absorpce energie molekulou) a tyto změny energie jsou popsány vztahy:

$$E = h * f\ [eV] \quad (3)$$

$$E = h * c * W\ [eV] \quad (4)$$

$$f = \frac{E_1 - E_0}{h}[Hz] \quad (5)$$

E – energie fotonu

h – Planckova konstanta (6,626 * 10^{-34}J.s)

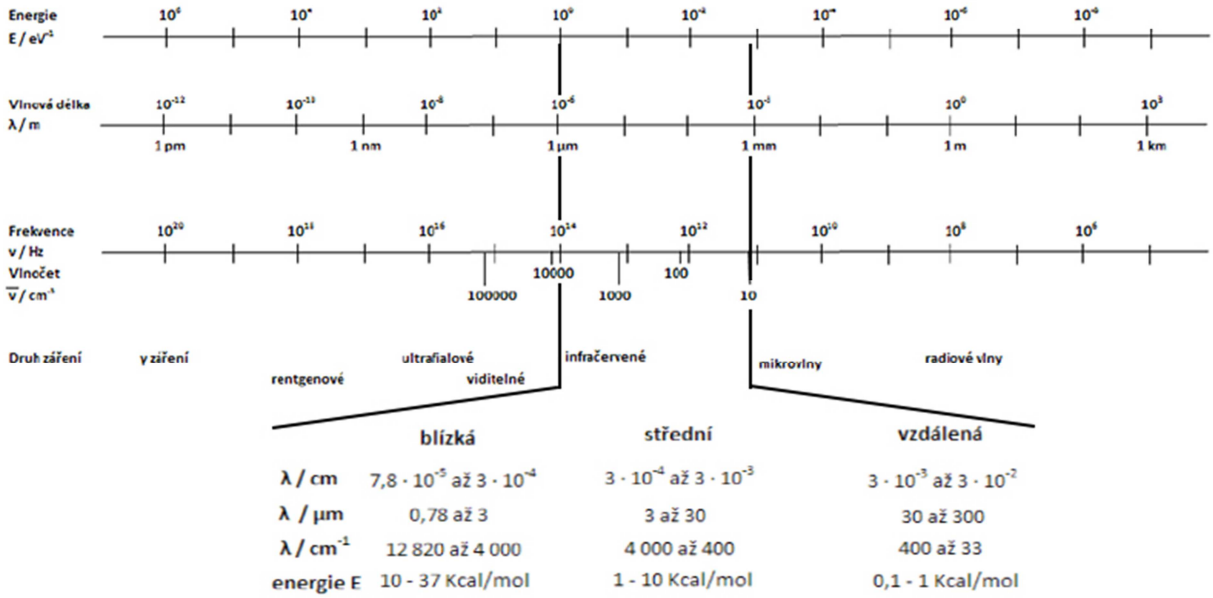
c – rychlost světla (299 792 458 m/s)

Rovnice č.3 popisuje energii potřebnou pro přestup na vyšší energetickou hladinu elektronu v molekule. V rovnici č.4 je výpočet energie pomocí vlnočtu, který je u infračervených spekter častější. Rovnice č.5 frekvence, při které molekula absorbuje energii pro přechod elektronu na vyšší energetickou hladinu.

Při dosazení do rovnice č.3 za f, dostáváme vztah:

$$E = \frac{h*c}{\lambda}[eV] \quad (6)$$

Zde je vidět, že energie je nepřímo úměrná vlnové délce a přímo úměrná frekvenci záření. [5]



Obr.1.:Elektromagnetické spektrum [1]

1.3 Infračervené spektrum

Celkové infračervené spektrum se dělí do třech základních oblastí, které se nazývají blízká, střední a vzdálená infračervená oblast. Rozdělení těchto oblastí je vidět na obrázku č.1.

Blízká infračervená oblast má vlnočet v rozsahu 12 820 až 4000 cm^{-1}. V tomto rozsahu má infračervené záření největší energii, což vyplívá z rovnice 4. V této oblasti jsou data nejméně kvalitní z důvodu překrývání spektrálních informací. Tato oblast se často používá při měření chemických reakcích.

Střední infračervená oblast se nachází v rozmezí 4 000 až 400 cm^{-1}. Tato oblast se využívá nejvíce, protože v tomto spektru nejvíce látek absorbuje energii infračerveného záření.

Vzdálená infračervená oblast má vlnočet 300 až 30 cm^{-1}. Oblast se nejčastěji využívá pro analýzu anorganických látek. Těžké molekuly absorbují nejvíce energie právě ve vzdálené oblasti. [1][2]

1.4 Teorie IR spektrometrie

Pro zobrazení naměřených výsledků se pro přehlednost využívá zobrazení do grafů. Na osu x se vynáší nejčastěji vlnočet, který se vynáší od nejvyšších hodnot. Někdy se místo vlnočtu vynáší vlnová délka. Na svislé ose y se pak vynáší intenzita absorpčních pásů, která se vynáší pomocí absorbance nebo transmitance. [2]

Absorpce se počítá podle vztahu:

$$A = \frac{I_0}{I}[-] \quad (7)$$

A – absorpce (bezrozměrná veličina)

I_0 – intenzita pozadí

I – Intenzita absorbovaného světla

Podle Beerova zákona je možné absobanci vztáhnout na koncentraci molekul ve vzorku, potom:

$$A = \varepsilon * l * c\,[-] \quad (8)$$

A – absorbance (bezrozměrná veličina)

ε – pohltivost (bezrozměrná veličina)

l – délka světelné dráhy skrz vzorek (cm)

c – koncentrace (bezrozměrná veličina)

Transmitance udává, kolik procent světla prošlo skrz vzorek.

$$T = 100 * \frac{I}{I_0}[\%] \quad (9)$$

T – transmitance

I_0 – intenzita pozadí

I – Intenzita absorbovaného světla

Mezi transmitancí a absorbancí je matematická závislost. Pomocí softwaru lze mezi absorbancí a transmitancí hodnoty převádět. [5]

1.4.1 Absorbování energie molekulou

Při dodání dostatečné energie molekule ve formě elektromagnetického záření, může přejít na vyšší energetickou hladinu a konat pohyb rotační (otáčení kolem své osy), translační (dojde k posouvání), vibrační (pohyb jednotlivých atomů) nebo elektronový. Během absorpce zároveň nastává změna dipólového momentu, nicméně u symetrických molekul, jako např. O_2, N_2, CO_2, ke změně momentu nedochází. Například u CO_2 se vazba C=O kompenzuje, protože jejich momenty působí proti sobě. Proto jsou tyto molekuly neměřitelné infračervenou spektrometrií. [2]

Infračervené záření nejvíce ovlivňuje vibrační pohyb, který se rozděluje na deformační, při kterém se mění úhel mezi atomy, a valenční, mění délku vazby. Translační a rotační pohyby se obtížně měří, protože jsou několikanásobně menší než deformační a valenční. Ve výsledném spektru se také projevují méně. Frekvence pro absorbování musí mít stejnou frekvenci jako je frekvence atomů. Během absorpce se pak nezmění frekvence atomů, ale amplituda.[2]

1.5 Základní druhy infračervených spektrometrů

Infračervené spektrometry můžeme rozdělit do dvou hlavních skupin podle konstrukčního uspořádání a principu na disperzní a nedisperzní. Disperzní spektrometry se používali jako první a položili základy infračervené spektrometrii. Až s odstupem času se začaly rozšiřovat spektrometry nedisperzní. Nejznámější nedisperzní infračervený spektrometr je FT-IR (infračervený spektrometr s Fourierovo transformací).

1.5.1 Disperzní spektrometry

Dispersní spektrometry mají 3 základní části a to zdroj záření, monochromátor a detektor záření. Zdrojem záření nejčastěji používaný v dispersních spektrometrech je keramická tyčinka, na jejímž povrchu je navinut odporový drát, který se průchodem proudu zahřívá na teploty 1000°C až 1400°C. Některé zdroje záření mají odporový drát vinutý uvnitř keramické tyčinky. Detektory používané v infračervené spektroskopii jsou termoelektrický detektor a pyroelektrický detektor. Princip termoelektrického detektoru je takový, že infračervené záření dopadající na spoj dvou různých kovů vytvoří proud, který jsme schopni změřit. Pyroelektrický detektor mění kapacitu dielektrika mezi dvěma elektrodami, změnu vlastností dielektrika způsobuje dopadající infračervené záření. Funkcí monochromátoru je rozklad infračerveného záření na difrakční mřížce. Otáčením difrakční mřížky se mění vlnočet záření, který následně dopadá na detektor. Princip měření je takový, že ze zdroje infračerveného záření přivedeme paprsek na kyvetu, ve které je zkoumaný vzorek a následně je paprsek veden na vstupní štěrbinu monochromátoru, ze kterého vychází infračervené světlo určitého vlnočtu. Kyveta musí být vyrobena z materiálu schopného propustit infračervené záření (např.: NaCl, KBr, CsI). [10]

1.5.2 Infračervené spektrometry s Fourierovou transformací

V dnešní době se využívají infračervené spektrometry s Fourierovou transformací kvůli rychlosti výpočtů a možnosti měřit celé spektrum současně. S užitím rychlé Fourierovy transformace se výpočty ještě více zrychlily. Spektrometry s Fourierovo transformací, lze použít i v takových případech, kde by bylo možné disperzní spektrometry použít jen velmi složitě nebo vůbec. [7]

Albert Abraham Michelson sestrojil první interferometr, který se dnes používá jako základ spektrometrů s Fourierovo transformací. Michelsonův interferometr se skládá z pohyblivého zrcadla, pevného zrcadla a polopropustného děliče záření. Ze zdroje záření dopadá paprsek na polopropustný dělič pod úhlem 45° a dělí se na polovinu, která se odrazí na pevné zrcadlo a polovinu která projde skrz polopropustný dělič a dopadne na pohyblivé

zrcadlo. Od zrcadel se paprsek odrazí zpět k polopropustnému děliči, na kterém dojde k interferenci odražených paprsků. Interference může být konstruktivní nebo destruktivní, to záleží na poloze pohyblivého zrcadla a vlnové délce procházejícího záření. Konstruktivní interference vznikne, je-li dráhový rozdíl dopadajících paprsků násobkem vlnové délky procházejícího záření.[8][9][10]

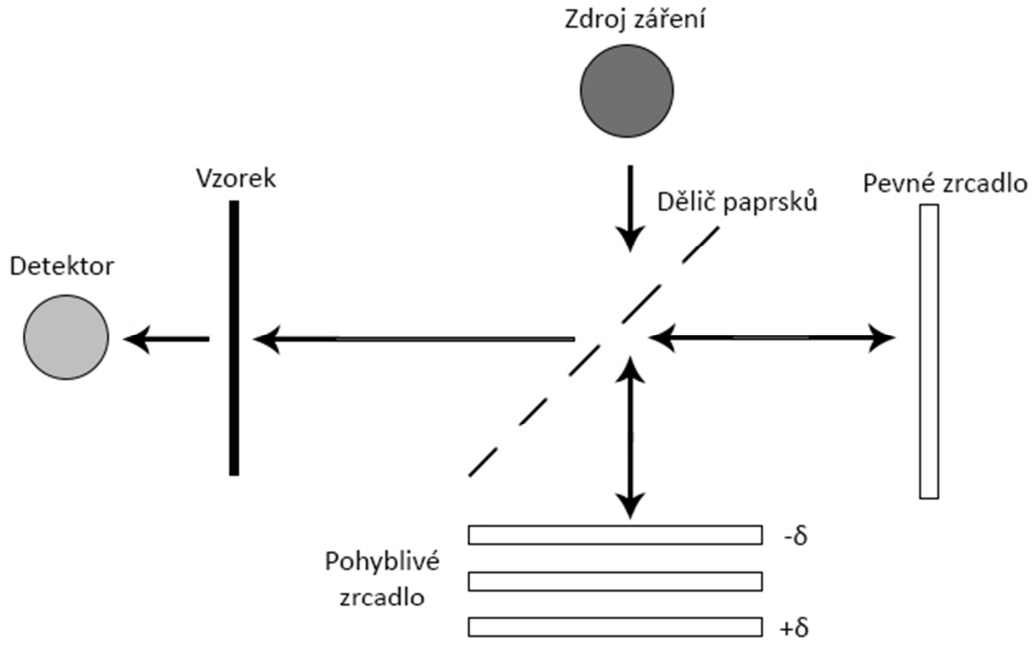
Platí:

$$\delta = n * \lambda \qquad (10)$$

δ – optický dráhový rozdíl

n – 0, 1, 2, …

λ – vlnová délka



Obr. 3:Schéma Michelsonova interferometru [1]

Paprsek, který rekombinoval na polopropustném děliči, je odražen do kyvetového prostoru, kde je umístěn vzorek a následně dopadá na detektor. Paprsek dopadající na detektor záření je snímán v závislosti na pohybu zrcadla (od -δ do +δ). Proto musí být rychlost pohybu zrcadla úměrná rychlosti snímání použitého detektoru. [10]

1.6 Techniky měření

Jelikož cílem každého měření je získání co nejkvalitnějších výsledů, je základem vybrat co nejvhodnější metodu, která nebude příliš zdlouhavá a náročná, ale dá nám správné výsledky. Základní metody používané v FT-IR jsou transmisní a odrazové metody.[5]

1.6.1 Transmisní metoda

Jedná se o nejstarší používanou metodu v infračervené spektroskopii, při které prochází paprsek infračerveného záření skrz měřený vzorek na detektor záření a je možné takto měřit plynné, kapalné i pevné vzorky. Nevýhodou této metody je, že paprsek musí projít vzorkem a v případě, že vzorek pohltí příliš mnoho infračerveného záření, na detektor se nemusí dostat žádné infračervené záření. Je tedy potřeba vzorky dělat co nejtenčí. Naopak výhodou je doba používání, se kterou jsou spjaty velké zkušenosti.[2]

a) Plynné vzorky

Kvůli velmi nízké hustotě plynů je třeba zajistit delší dráhu paprsku. Při měření se používá plyn pod správným tlakem stlačený ve speciální kyvetě. Kyveta má stěny vytvořené ze skla nebo kovu a okýnko z materiálu schopného propustit IČ záření. Tímto okýnkem do vzorku vstupuje infračervené záření a následně vystupuje. Pokud je potřeba ještě prodloužit dráhu paprsku, používají se plynové kyvety, které mají na koncích zrcadla, která odrážejí paprsek a výsledná dráha paprsku může být až 120m. [2]

b) Kapalné vzorky

Před měřením kapalin není třeba provádět mnoho příprav vzorku. Pokud se jedná o stálou kapalinu, můžeme ji nanést na materiál propouštějící infračervené záření a abychom dostali co nejtenčí film, stačí na kapalinu umístit další destičku z IČ propustného materiálu. Kapaliny však často pohlcují velké množství infračerveného záření a proto se musí rozpouštět (ředit). Jako rozpouštědlo se nejčastěji používá oxid těžkého uhlíku (D_2O) a dříve se používali uhlíkaté látky (CS_2 nebo CCl_4). Pro měření kapalných vzorků je více druhů kyvet. Kyvety, které mají pevnou délku dráhy paprsku, jsou vhodné spíše pro nestabilní kapaliny, avšak jejich čištění velice náročné, protože není možné je rozebrat. Snadnější práce je s polopropustnými kyvetami, které je možné rozebrat. [2]

c) Pevné vzorky

Na měření pevných vzorků můžeme aplikovat tři možné postupy. Můžeme měřený materiál namlít na jemný prášek, smísit s IČ propustným materiálem a slisovat d tenkých tablet. Další možnost je namletý materiál smíchat s olejem anebo materiál rozpustit a nanést jako tenký film. [2]

Pokud se z materiálu lisuje tenká tableta, namletý materiál se smísí s materiálem, který propustí infračervené záření (např. KBr) a při vysokém tlaku se slisuje. Při výrobě tablety z materiálu KBr je třeba postupovat co nejrychleji nebo pracovat ve vhodném prostředí, protože KBr pohlcuje vlhkost z okolní atmosféry. Při výrobě záleží na důkladném promísení základního materiálu a materiálu propouštějící IČ záření, protože dochází k nehomogenitě tablety. [2]

Při smíchání materiálu s olejem (vytvoření suspenze) se napřed zkoumaný materiál rozemele na jemný prach, který se smíchá s minerálním olejem (nujol) nebo fluorizovaným olejem (fluorob). Oba tyto oleje mají však své absorpční pásy, a proto je vhodné měřený vzorek měřit v obou olejích a následné porovnat naměřená spektra, ve kterých můžeme vyloučit absorpční pásy olejů. [2]

Zkoumaný materiál je také možné rozpustit vhodným rozpouštědlem. Některé polymery a vosky je možné vyšší teplotou rozehřát a nanést na IČ propustný materiál jako tenký film, kterým infračervené záření projde.[2]

1.6.2 Reflexní metody

Reflexní metody se využívají pro vzorky, u kterých se transmisní metoda dá využít jen velmi obtížně nebo ji není možné použít vůbec. Reflexní metody jsou založené na odrazu světla od povrchu vzorku. Mezi nejčastěji využívané metody patří metoda spektrální reflexe, metoda zeslabené totální reflexe (ATR – attenuated total reflectance), metoda difuzní reflexe (DRIFTS – diffuse refrection spectroscopy). [2]

a) **Spektrální reflexe**

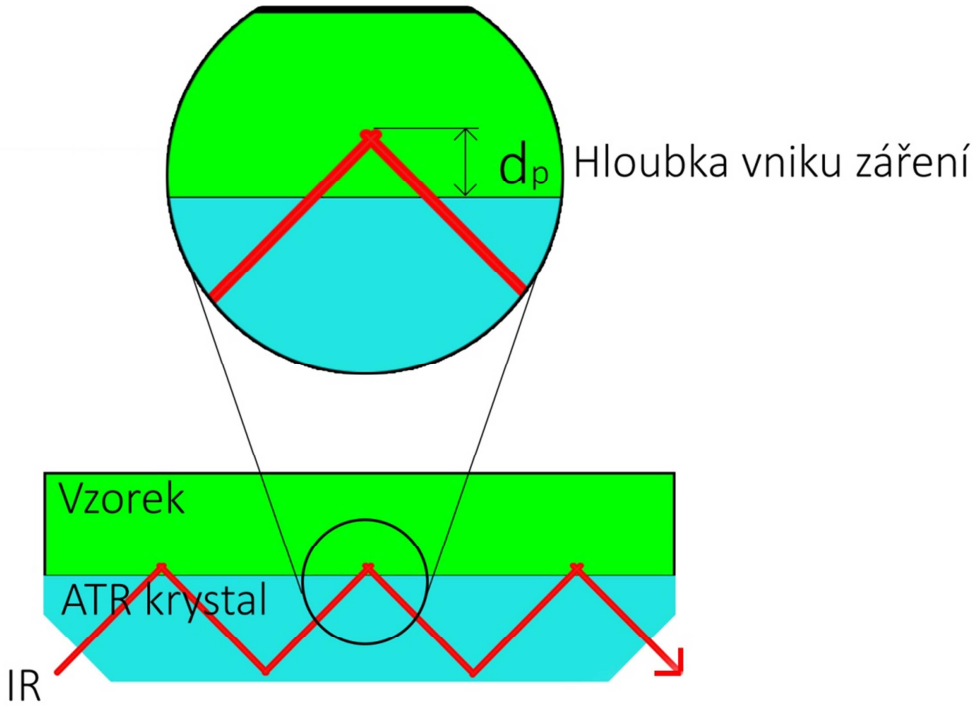
Spektrální reflexe se využívá, jelikož se jedná o nedestruktivní metodu a vzorek je tedy možné použít znova. Metoda pracuje na principu odrazu světla od povrchu vzorku. Úhel dopadajícího záření by měl být totožný s úhlem odraženého světla. Je třeba, aby při dopadu došlo k úplnému odrazu na rozhraní dvou ploch, jinak by intenzita odraženého záření klesla. Déle je třeba vzít v úvahu index lomu světla materiálu a absorpční schopnost. Všechny tyto parametry ovlivní intenzitu odraženého světla.[2]

Měření vzorku je možné provést přímým odrazem od povrchu vzorku, při kterém materiály odráží přibližně 5 – 10% dopadajícího záření. V oblasti, ve které materiál vykazuje velkou absorpci záření, odráží vzorek větší množství záření. Na naměřené spektrum se pak musí použít Kramers – Kronig transformace, která převede spektrum na transmitanční nebo absorbanční vyjádření spektra.[2]

Druhou možností jak provést měření spektrální reflexe je, vyslat světlo, které projde tenkým filmem, odrazí se od neabsorbující podkladové destičky. Nejvhodnější pro měření kapalin. Za předpokladu, že dokážeme z materiálu vytvořit tenký film, který umístíme na podkladovou neabsorbující destičku, můžeme použít i pevné materiály. [2]

b) Zeslabená totální reflexe

Zeslabený úplný odraz (ATR) je metoda využívající úplného odrazu na rozhraní krystalu a měřeného vzorku. Při měření se využívá jednonásobného nebo vícenásobného odrazu pro prodloužení průchodu vzorkem. Záření se vyšle krystalem pod takovým úhlem, aby došlo k úplnému odrazu na rozhraní krystal – vzorek. Záření pronikne do vzorku několik mikrometrů hluboko. Záření dopadající na vzorek může být vzorkem i zcela absorbováno. [2]



Obr.4: Metoda totální reflexe[10]

Vztah pro výpočet hloubky vniku do měřeného vzorku:

$$d_p = \frac{\lambda}{2\pi n_k \sqrt{\sin^2\theta - (n_v / n_k)^2}} \qquad (10)$$

d_p – hloubka vniku [µm]

λ – vlnová délka [m]

n_v – index lomu vzorku [-]

n_k – index lomu krystalu [-]

θ – úhel lomu na fázovém rozhraní [°]

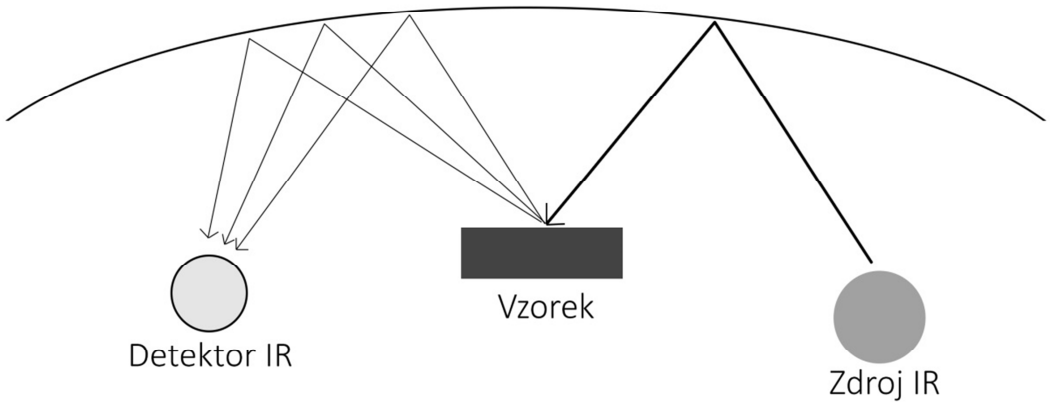
Ze vzorce (10) je vidět, že s větší vlnovou délkou záření je i hloubka vniku větší. Vlnové délky, které vzorek absorbuje, se projeví v naměřeném spektru. Pokud materiál absobruje málo, použije se metoda s několikanásobným odrazem. Používají se krystaly z germania, silikonu, bromoididu, diamantu. Krystaly musí mít vysokou čistotu a po každém měření se musí očistit, aby při dalším měření nedošlo ke zkreslení naměřeného spektra nečistotami.[2]

Pomocí metody zeslabené totální reflexe je možné měřit obrovské množství druhů vzorků. Nejrůznější kapaliny, pasty, polymery, tkaniny, lepidla a mnoho dalšího, naopak pro měření směsí s nerovnoměrnou koncentrací materiálů není příliš vhodná. [2][10]

c) **Difuzní reflexe**

Metoda difuzní reflexe (DRIFTS) měří odražené záření od povrchu vzorku, které se šíří do více směrů. Při použití této metody je možné měřit některé materiály přímo, ale většinou se materiály rozemelou a smíchají se s IR propustným materiálem. Nejčastěji využívaný je materiál KBr. Koncentrace namletého zkoumaného materiálu by se měla pohybovat mezi 5% až 15%. Směs se před měřením slisuje. Metoda je založena na difuzním odrazu od povrchu materiálu, a proto když je třeba měřit nějaký materiál s lesklým povrchem přímo, je třeba povrch zdrsnit. [2][10]

Měření se vyšle paprsek ze zdroje záření na měřený materiál. Záření pronikne pod povrch a dojde k difuznímu odrazu do více směrů. Odražené paprsky se pomocí eliptické odrazné plochy soustředí do ohniska, kde je umístěn detektor IR. Získané hodnoty při měření jsou ve spektru slabé, a proto se používají Kubelka – Munkovy jednotky na svislé ose místo absorbance, respektive transmitance. [2][10]



Obr.5: Metoda DRIFTS [2]

2 Statistické nástroje pro analýzu infračervených spekter

Myšlenka statistického zpracování dat je založena na analyzování vztahů mezi individuálními body z velkého souboru naměřených dat, analyzování odchylek mezi po sobě jdoucích stejných měření, někdy je užitečné znát i aritmetický průměr souboru dat. [11]

V oborech jako je technika, biologie nebo i lékařství se kromě informací, které jsou obsaženy ve skaláru (jednorozměrná data), také zkoumají informace, které nese vektor o *m* prvcích (vícerozměrná data). Vícerozměrná data jsou například:[12]

a) Vlastnosti produktů (potraviny, oleje, slitiny,...) vyjádřené různými analytickými metodami
b) Určování spekter absorpčními pásy, polohami a plochami, pro identifikování chemických sloučenin
c) Sledování složení surovin, odpadů a jiných směsí v závislosti na čase či jiných parametrech
d) Sledování parametrů výstupních produktů výroby a řízení jakosti

2.1 Popis vícerozměrných dat

Zdrojová matice dat (obsahuje data, ze kterých vycházíme) popisuje proměnné v *n* řádcích (například typy procesorů, automobily) a v *m* sloupcích, ve kterých jsou měřená data (takt procesorů, L1 a L2 cache, spotřeba vozů, objem motoru, spotřeba). Data zdrojové matice však často mívají různé jednotky, je zdrojová matice ještě před zpracováním upravena. Této úpravě se říká škálování a to tak, že se od prvků sloupce odečte jejich průměr (centrování), anebo se centrované prvky ve sloupcích ještě vydělí jejich směrodatnou odchylkou (standardizace).[12]

Statistická analýza je založena na předpokladech, že x_{ij} tvoří náhodný výběr (naměřená nebo jinak získaná data). Výběr obsahuje *n* řádkových vektorů, které je možné prát jako řádky zdrojové matice, anebo jako souřadnice *n* bodů v *m* rozměrném prostoru původních proměnných. Zdrojová matice může vypadat takto:[12]

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & \cdots & x_{1,m} \\ \vdots & & \vdots & & \vdots \\ x_{i,1} & \cdots & x_{i,j} & \cdots & x_{i,m} \\ \vdots & & \vdots & & \vdots \\ x_{n,1} & \cdots & x_{n,j} & \cdots & x_{n,m} \end{bmatrix} \tag{11}$$

Řádek zdrojové matice, nebo i-tý vektor matice, se nazývá objekt (například typ procesoru nebo automobilu). Objekt je pak možné charakterizovat proměnnými a to buď kvantitativními (číselnými) hodnotami, nebo kvalitativními (nečíselnými) proměnnými.[12]

Kvantitativní (číselné) proměnné jsou v typech:

a) Proměnné s absolutní stupnicí, které mají přirozený počátek a jeden parametr měřítka například KNO_3
b) Proměnné s poměrovou stupnicí, zachovávají poměr hodnot charakteristik
c) Proměnné s intervalovou stupnicí zachovávají podíl rozdílů charakteristik. Mají přirozený počátek pro obě srovnávané veličiny (např. poměr absorbancí detektoru)
d) Proměnné s rozdílovou stupnicí, vztahují se k různým počátkům

Kvalitativní proměnné mají typy:

a) Proměnné v nominálním měřítku, udávají nejméně informací. Navíc informace vypovídá pouze o rovnosti nebo různosti tříd.
b) Proměnné v ordinální škále, jsou definovány relace mezi třídami větší nebo menší
c) Proměnné v alternativním (binárním) měřítku, které popisují jestli jsou třídy rovné či nerovné, v porovnání s nějakým zadaným kritériem (binární soustava je obvykle popsána jako 1 – ano, 0 – ne)

Shluk dat se nazývá třídou. Třídou pak rozumíme objekty, které mají stejné proměnné, anebo alespoň velmi podobné (například procesory AMD, ATMEL, Intel). Vzdálenost mezi objekty, pro rozdělení do tříd, posuzujeme podle míry vzdálenosti objektů v *m*-rozměrném prostoru.[12]

Míru vzdálenosti objektů pro číselné proměnné se používají metriky:

Eukleidova metrika, nebo-li geometrická vzdálenost se počítá ze vztahu

$$d_E(x_k, x_l) = \sqrt{\sum_{j=1}^{m} (x_{kj} - x_{lj})^2} \tag{12}$$

Hammingova metrika

$$d_H(x_k, x_l) = \sum_{j=1}^{m} |x_{kj} - x_{lj}| \tag{13}$$

Zobecněná Minkowskiho metrika

$$d_M(x_k, x_l) = \sqrt[n]{\sum_{j=1}^{m} |x_{kj} - x_{lj}|^n} \tag{14}$$

Ve vztahu 14 je vidět, že pro n = 1 je vztah Hammingovou metrikou a pro n = 2 je vztah Eukleidova metrika. Tyto metriky vzdáleností však neuvažují závislosti mezi proměnnými. Pokud zakomponujeme do vztahu pro vzdálenost ještě závislosti mezi proměnnými, které vyjádříme kovarianční maticí C, získáme Mahalanobisovu metriku 15.[12]

$$d_{MA}(x_k, x_l) = \sqrt{(x_k - x_l)^T C^{-1} (x_k - x_l)} \tag{15}$$

S Eikleidovou metrikou je Mahalanobisova metrika nejpoužívanější v praxi. Objekty, které zařazujeme do tříd, jsou si bližší, pokud je vzdálenost mezi nimi menší.[12]

Pro míru podobnosti objektů při klasifikaci do tříd je možné využít Pearsonův párový korelační koeficient r. Čím více se párový koeficient blíží k jedničce, tím více jsou si objekty podobné. Za předpokladu, že máme ordinální měřítko, používá se míra podobnosti pomocí Spearmanův korelační koeficient. Pro podobnost binárních proměnných můžou nastat případy podobnosti 0 – 0, 0 – 1, 1 – 0, 1 – 1. Ty to případy si označíme pro snadnější práci jako 0 – 0 písmenem a, 1 – 0 písmenem b, 0 – 1 písmenem c a případ 1 – 1 písmenem d, můžeme pak definovat případy podobnosti:[12]

a) Sokalův-Michenerův koeficient podobnosti

$$S_{SM} = \frac{a+d}{a+b+c+d} \tag{16}$$

b) Russelův-raoův koeficient podobnosti

$$S_{RR} = \frac{d}{a+b+c+d} \tag{17}$$

c) Hamannův koeficient podobnosti

$$S_{H} = \frac{a+d-b-c}{a+b+c+d} \tag{18}$$

2.2 Analýza hlavních komponent PCA

Analýza hlavních komponent je často využívaná statistická metoda, která nalezla uplatnění nejen ve výzkumu, ale i komprimaci při přenosech signálů či při forenzním zkoumání. Požívá se pro získání významných vzorků z velkého množství dat. Nejvýznamnější vzorky souboru jsou právě ty, mezi kterými dochází k velkým změnám. Velké množství po sobě jdoucích vzorků s podobnými hodnotami nenesou významné informace o měřené veličině.

2.2.1 Nástroje pro analýzu hlavních komponent:

a) Výběrová směrodatná odchylka

Výběrová směrodatná odchylka vypovídá o tom, jak se od sebe liší prvky souboru dat. Pokud je směrodatná odchylka malá, znamená to, že prvky souboru dat jsou si velmi podobné a naopak pokud je směrodatná odchylka velká, prvky souboru jsou více odlišné. [1]

Pro výpočet výběrové směrodatné odchylky je zapotřebí znát aritmetický průměr souboru dat, který získáme ze vztahu:

$$\overline{X} = \sqrt{\frac{\sum_{i=1}^{n} X_i}{n}} \tag{19}$$

X_i – i-tý prvek souboru dat

n – počet prvků souboru dat

$\overline{X}$ - aritmetický průměr souboru dat

Pro výpočet výběrové směrodatné odchylky se pak použije vztah:

$$s = \sqrt{\frac{\sum_{i=1}^{n} \left(X_i - \overline{X}\right)^2}{n-1}} \tag{20}$$

s – výběrová směrodatná odchylka souboru dat

X_i – i-tý prvek souboru dat

n – počet prvků souboru dat

$\overline{X}$ - aritmetický průměr souboru dat

Například pro soubor dat X = [2, 8, 12, 18] vychází výběrová směrodatná odchylka s = 5,831 a pro výběr Y = [10, 10, 10, 10] vyjde s = 0, protože všechny hodnoty jsou stejné a nemají žádnou odchylku.[11]

b) Kovariance

Předchozí způsob výpočtu počítá vztahy pouze mezi jednorozměrnými daty. Nicméně většina měření má výstupní data vícerozměrná a úkolem statistické analýzy je zjistit vztahy mezi těmito dimenzemi dat. Například můžeme naměřit spektrum materiálu a různé koncentrace materiálu ve vzorcích. Následně nás bude zajímat, jestli koncentrace materiálu ve vzorcích měla vliv na měřené spektrum.[11]

Výběrová směrodatná odchylka zkoumala změny pouze v jedné dimenzi. Kovariance zkoumá změny v souboru dat, přičemž bere v úvahu více dimenzí. Pokud by se udělala

kovariance jedné dimenze, výsledkem by byla výběrová směrodatná odchylka. Za předpokladu že budeme mít tří rozměrná data (x, y, z), můžeme provést kovarianci rozměrů x a y, x a z, nebo y a z. Vzorec kovariance pak je:

$$\mathrm{cov}(X,Y) = \frac{\sum_{i=1}^{n}\left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right)}{n-1} \tag{21}$$

X_i – i-tý prvek souboru dat X

$\overline{X}$ - aritmetický průměr souboru dat X

Y_i – i-tý prvek souboru dat Y

$\overline{Y}$ - aritmetický průměr souboru dat Y

n – počet prvků souborů dat

Ze vzorce 13 je patrný důvod proč kovariance jednoho rozměru vychází stejně jako výběrová směrodatná odchylka. Když se namísto rozměru Y dosadí rozměr X, vzorec pak je totožný jako pro výpočet výběrové směrodatné odchylky.[11]

c) Matice kovariance

Při výpočtu kovariancí mezi všemi dimenzemi se pro přehlednost používá matice, která pro 3 dimenze vypadá následovně:

$$C = \begin{pmatrix} \mathrm{cov}(X,X) & \mathrm{cov}(X,Y) & \mathrm{cov}(X,Z) \\ \mathrm{cov}(Y,X) & \mathrm{cov}(Y,Y) & \mathrm{cov}(Y,Z) \\ \mathrm{cov}(Z,X) & \mathrm{cov}(Z,Y) & \mathrm{cov}(Z,Z) \end{pmatrix} \tag{22}$$

Na hlavní diagonále matice 14 jsou kovariance stejné dimenze, tedy výběrová směrodatná odchylka. Protože cov(X,Y) je stejné jako cov(Y,X), je patrné, že matice je symetrická podle hlavní diagonály.[11]

d) Vlastní čísla a vlastní vektory matice

Vlastní vektory a čísla matice se využívají pro transformaci matic. Při transformaci se pouze mění délka vlastního vektoru, směr zůstává stejný. Vlastní vektory je možné nalézt pouze pro čtvercové matice a ne zcela všechny čtvercové matice mají vlastní vektor. Pro větší rozměry matic je výpočet vlastních čísel matice a následně i výpočet vlastních vektorů matice velmi obtížný, a proto se využívají programy, které tyto výpočty provedou za nás. [11]

2.2.2 Postup metody PCA

Při řešení metody se využijí nástroje uvedené v bodech a) až d), které se aplikují na soubor naměřených nebo jinak získaných dat. Úpravy souboru dat probíhají následujícími kroky:

Krok č.1: Odečtení průměru

Pro každou dimenzi dat je zapotřebí vypočítat průměr a odečíst ho od každé hodnoty v souboru dat. Tak získáme soubor dat, kterého střední hodnota je rovna nule. [11]

Krok č2: Výpočet matice kovariance

Výpočet probíhá stejně, jako bylo popsáno v bodě c).

Krok č.3: Výpočet vlastních čísel a vlastních vektorů matice

Protože matice kovariance je čtvercová, je možné vypočítat její vlastní čísla a vlastní vektory. Tyto hodnoty dávají další důležitá data o souboru naměřených dat. Hodnoty se pohybují kolem os vektorů. Je zapotřebí, aby vlastní vektory byly jednotkové, tedy jejich velikost byla 1. Pro PCA analýzu je to důležité, ale většina matematických softwarů vlastní vektory počítá automaticky jako jednotkové. [11]

Krok č.4: Výběr komponent a vytváření charakteristického vektoru

V této fázi dochází ke snižování počtu rozměrů dat a kompresi dat. Vlastní vektor s nejvyšším vlastním číslem je hlavní komponenta souboru dat. Čím je vlastní číslo větší, o to důležitější komponentu souboru dat se jedná. [11]

Když jsou vlastní vektory kovarianční matice nalezeny, musí se seřadit podle váhy vlastních čísel od nejvyšší po nejnižší. Jinak řečeno seřadí se tak vlastní vektory od nejdůležitější hlavní komponenty. Nyní je možné některé hlavní komponenty vypustit. Vypuštěním některé z komponent se dopouštíme chyby při výpočtu, ale pokud je vlastní číslo malé, dopouštíme se malé chyby, kterou si můžeme dovolit. Důležité ale je, že vypuštěním komponent se sníží i počet dimenzí a velikost souboru dat. V případě, že z původních dat vyjde n vlastních čísel a tedy i n vlastních vektorů, vybere se jen p prvních vlastních vektorů, výsledný soubor dat bude mít p dimenzí. [11]

Vlastní vektor je ve své podstatě matice složená z vlastních vektorů, tak že každý vektor se zapíše do jednoho sloupce matice. Hlavní komponenty s malým vlastním číslem, které je možné vypustit, se do této matice již nezapisují. [11]

Krok č.5: Získání upraveného souboru dat

Provede se úprava charakteristického vektoru (zde jsou vybrané vlastní vektory), tak že ze sloupců budou řádky (vlastní vektor s nejvyšší důležitostí bude v prvním řádku) a soubor

dat se zapíše do řádků, tak že v každém řádku budou data jedné dimenze. A provede se jednoduchý výpočet:

Upravená data = Upravený char. Vektor · Data v řádcích (23)

Takto upravená data se již nemění kolem os (např. X a Y), ale mění se kolem vlastních vektorů, které jsme vybraly do charakteristického vektoru (matice vlastních vektorů). V případě, že zůstane pouze jeden vlastní vektor, výsledná data souboru jsou pouze jednorozměrná a soubor dat leží na přímce.[11]

Výsledkem transformace jsou tedy data, která již nejsou závislá na osách *x* a *y*, ale dávají lepší přehled o vztazích mezi daty souboru. Pokud se nevypustí žádný vlastní vektor, nedojde ke ztrátě dat, ale nesníží se počet dimenzí dat. [11]

2.3 Diskriminační analýza DA

Diskriminační analýza slouží k hledání struktury a vzájemných vazeb v objektech a byla zavedena v roce 1936 Ronaldem Fisherem. Pomocí klasifikačních metod, mezi které patří i diskriminační analýza, se objekt zařadí do existující třídy, anebo neuspořádaná množina objektů se seřadí do tříd s podobnými objekty. Diskriminační analýza je jednou z metod, která zkoumá závislosti mezi skupinou *p* nezávisle proměnných, které se nazývají diskriminátory. Skupina *p* nezávisle proměnných jsou sloupce zdrojové matice na straně jedné a na straně druhé závisle proměnná. Objekt se zařazuje do třídy na základě nejvyšší podobnosti. [12]

Nejjednodušším výstupem může být binární proměnná, která nabývá hodnoty 0 pro zařazení do první třídy a hodnoty 1 pro zařazení do třídy druhé. [13]

2.3.1 Zařazovací pravidla DA

Pro zařazování do tříd obvykle bývá větší množství tříd s mnoha objekty, pro vysvětlení principů však postačí dvě třídy A a B, klasifikace se provede na základě jednoho znaku *x* s normálním rozdělením. Ve třídě A se jedná o normální rozdělení $N(\mu_A, \sigma_A^2)$ a skupina B má normální rozdělení $N(\mu_B, \sigma_B^2)$ a nový objekt má hodnotu *x*. Logicky se pak snažíme zařadit objekt do třídy, k jejíž střední hodnotě má *x* nejblíže. Prahový bod pro rozhodování do jaké třídy *x* zařadíme pak je $C = (\mu_A + \mu_B)/2$. V případě, že $x < C$ pak objekt patří do skupiny A a pokud je $x \geq C$, zařadíme objekt do skupiny B. Pravděpodobnost zařazení je pro obě skupiny stejná. Pokud se pravidlo aplikuje na rozdělení, které nemá stejné rozptyly, například $\sigma_A^2 < \sigma_B^2$, došlo by k tomu, že pravděpodobnost nesprávného zařazení pro třídu A by byla větší než pro třídu B. [12]

2.3.2 Lineární (LDA) a kvadratická (QDA) diskriminační funkce

Pro odvození obecného tvaru diskriminačních funkcí se často vychází z rovnice pro aposteriorní pravděpodobnost (pravděpodobnost podmíněnou zkušeností) příslušnosti k j-té skupině a následně nalézt maximum. Podle typu pravděpodobností pro $f_1(x)$ a $f_2(x)$ se liší jednotlivé diskriminační metody v tom, jak jsou určeny dělící oblasti a jaký je jejich tvar. V případě, že máme normální rozdělení, které se odlišuje pouze středními hodnotami tříd, získáme lineární diskriminační analýzu. Pokud máme normální rozdělení, které se liší středními hodnotami tříd a navíc také kovariančními maticemi tříd, získáme kvadratickou diskriminační funkci. V případě, že se jedná o neparametrické hustoty rozdělení, získáme flexibilní diskriminační funkce. Bayesův přístup předpokládá, že hustota pravděpodobnosti ve třídě se dostane jako součin marginálních hustot znaků (Bayesův přístup považuje znaky za závisle proměnné). Pro vícerozměrné Gaussovo rozdělení v i-té třídě má hustota pravděpodobnosti tvar [14][15]

$$f_i(x) = \frac{1}{(2\pi)^{m/2}\det(C_i)^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^T C_i^{-1}(x-\mu_i)\right) \tag{24}$$

2.3.3 Úprava prahového bodu

Zatím byl rozhodovací bod C uvažován jako bod, pro který je pravděpodobnost chybného zařazení pro obě třídy stejná. Někdy však je užitečné, aby každá třída měla trochu jiné pravděpodobnosti pro zařazení do třídy, podle apriorních pravděpodobností π_1 a π_2. Vzorec pro optimální volbu prahového bodu C, pro vícerozměrné normální rozdělení je[14][15]

$$C = \frac{\overline{Z_1} + \overline{Z_2}}{2} + \ln\frac{\pi_1}{\pi_2} \tag{25}$$

Ve vztahu 25 je vidět, že pro $\pi_1 = \pi_2 = 0{,}5$ bude poměr π_1 a π_2 roven jedné a ze vztahu vypadne přirozený logaritmus a vzorec se tak zjednoduší. [14][15]

2.3.4 Volba diskriminátorů

Nastává otázka, zda má volba diskriminátoru x dostatečnou přesnost pro rozdělování do tříd. Byly proto navrženy metody a postupy jak vybírat znaky pro rozdělení do tříd. Účelem každé metody či postupu, je dostatečná rozlišitelnost, do jaké třídy objekt patří. Některé metody začínají se všemi diskriminačními znaky a postupně se vynechávají ty, které neposkytují dostatečnou rozlišitelnost. Často je diskriminační analýza používána, jako explorativní nástroj. Při hledání vhodného modelu, se do dat zahrnuje velké množství potenciálně užitečných znaků. Ne všechny znaky se však ukáží jako účinné a takové znaky je třeba vypustit. Dopředu ovšem není známo, které znaky budou účinné pro zařazování do tříd,

a proto jedním z možných výsledků diskriminační analýzy je nalezení účinných znaků pro roztřídění, tedy účinných diskriminátorů. Při tomto výběru se jedná o analogii s vícenásobnou regresní analýzou. Rozdíl je takový, že v diskriminační analýze se netestuje, zda se změní hodnota čtvercového vícenásobného korelačního koeficientu R^2 přidáním nebo odebráním proměnné, ale testuje, zda se změní velikost Mahalanobisovy vzdálenosti d_{MA}^2. Nejčastějším obecně užívaným algoritmem je krokový výběr diskriminátorů, který kombinuje jak přidávání diskriminátorů, tak jejich odebírání. Při krokové metodě má první užitý diskriminátor, ve výběrové metodě, nejvyšší hodnotu. Provede se přepočtení hodnot diskriminátorů v modelu a nejvyšší se přidá ke skupině diskriminátorů, které se budou využívat pro zařazování do tříd. První znak, který byl v modelu využit se přepočítá, jestli nesplňuje vyřazovací podmínku. V případě, kdy splní podmínku, je diskriminátor vyřazen. Vše se opakuje, dokud jsou diskriminátory, které můžeme testovat na zařazení do souboru diskriminátoru pro roztřídění.[14][15]

2.3.5 Kritéria pro vybírání diskriminátorů

Pro vybírání diskriminátorů je více kritérií. Jedno z nich je Wilkovo kritérium λ, u kterého platí, že pokud diskriminátor v diskriminační funkci má nejnižší hodnotu Wilkova kritéria λ, bude tento diskriminátor zahrnut do modelu. K zavedení nebo odstranění diskriminátoru z modelu se dovoluje jeden krok. Maximální počet kroků k vybírání diskriminátorů je roven dvojnásobku jejich celkového počtu. Aby se předešlo potížím při provádění výpočtů, stanovuje se tolerance pro zavedení diskriminátoru do modelu. Tolerance je mírou lineární asociace mezi diskriminátory a počítá se pro i-tý prvek jako $1-R_i^2$, kde R_i^2 je čtverec vícenásobného korelačního koeficientu, když je brán v úvahu i-tý diskriminátor, jako závisle proměnná a je uvažována i regresní rovnice mezi tímto i-tým diskriminátorem a ostatními diskriminátory. Malé hodnoty tolerance naznačují že i-tý diskriminátor je lineární kombinací ostatních diskriminátorů. Diskriminátory s velice malou tolerancí není vhodné do modelu zařazovat. [14][15]

Významnost změny Wilkova kritéria λ pro zavedení diskriminátoru do modelu nebo jeho odstranění z modelu se zakládá na testovacím kritériu F. Hodnota testovacího kritéria F či vypočtená statistická významnost α se použije jako kritérium pro zavedení nebo odstranění diskriminátoru z modelu. Avšak obě kritéria se nemusí shodovat, protože pevné hodnoty kvantilu F, mění svou hodnotu pravděpodobnosti v závislosti na počtu diskriminátorů. Skutečná hladina významnosti α je obtížně spočítatelná, protože je závislá na mnoha faktorech, včetně testovacího kritéria F.[14][15]

Dříve než se aplikuje krokový algoritmus, jsou na začátku všechny tolerance a minimum tolerance rovné 1, z důvodu že na začátku v modelu není žádný diskriminátor. Spolu s Wilkovým kritériem λ se také spočte pro statistickou významnost každého diskriminátoru také testovací kritérium F. Hodnota F pro změnu Wilkova kritéria po přidání diskriminátoru do modelu s celkovým počtem *p* diskriminátorů se spočte[14][15]

$$F_{změmě} = \frac{n-g-p}{g-1}\left(\frac{\frac{1-\lambda_{p+1}}{\lambda_p}}{\frac{\lambda_{p+1}}{\lambda_p}}\right) \tag{26}$$

Pro který n je celkový počet objektů, g je počet tříd, λ_p je Wilkovo kritérium před přidáním diskriminátorů. Při každém kroku je jeden z diskriminátorů, který má nejnižší Wilkovo kritérium λ zařazen do souboru diskriminátorů. Dalším používaným kritériem Mahalanobisova vzdálenost.[12][14][15]

2.4 Principal Component Regression PCR

Regrese hlavních komponent je technika kvantitativní analýzy, která zkoumá specifický region nebo regiony spektra, aby určila, které oblasti se mění stejně jako funkce koncentrace komponent. [16]

PCR metoda pracuje ve dvou krocích. V prvním kroku je využita spektrální informace pro vypočtení hlavních komponent spektra, poté jsou hlavní komponenty a koncentrace komponent využita pro vytvoření modelu. Všechny hodnoty koncentrace komponent jsou vypočítávány zároveň. [16]

Pro získání co nejpřesnějších výsledků je potřeba dodržet následující:

a) Změřit alespoň 3 kalibrační standardy a jeden dodatečný pro každou komponentu
b) Standardy jsou směsi, které obsahují všechny komponenty, které očekáváme, že nalezneme v neznámém vzorku
c) Koncentrace komponent ve standardech se mění nezávisle

Metoda PCR je již delší dobu využívána pro statické analýzy. Metoda využívá veškerá naměřená data v celém rozsahu nebo v určeném regionu. [16]

2.5 Partial Least squares PLS

PLS je metoda, do určité míry podobná metodě PCR. Metoda stejně jako PCR zkoumá určitý region nebo regiony, aby zjistila, které oblasti se mění stejně jako funkce koncentrací

komponent. Oproti metodě PCR, která vytvářela model ve dvou krocích, PLS metoda použije informace o spektru a koncentraci ze standardů. [16]

Metoda dosahuje nejvyšší přesnosti, pokud se dodrží pravidla:

a) Změření alespoň 3 kalibračních standardů a jeden dodatečný pro každou komponentu
b) Standardy obsahují všechny komponenty ve směsi, které očekáváme, že se budou vyskytovat v neznámém vzorku
c) Koncentrace komponent vzorku se mění nezávisle
d) Spektrální změna vyskytující se ve standardech representuje změnu, kterou očekáváme v neznámém

Ve své podstatě se metoda PLS liší od PCR v době d) a díky této odlišnosti je metoda přesnější. Metoda je vhodná pro klasifikaci sloučenin, jako jsou například čisticí prostředky ve vodě, nebo také bor a fosfor ve skle. PLS metoda je ale také vhodnější, pro analýzu vlastností vzorků než pro analýzu složení. [16]

2.6 Classical least squares CLS

Metoda nejmenších čtverců vyjadřuje měřenou absorbanci jako součet vzorků absorbance každé komponenty, která byla měřena. Jinými slovy, model předpokládá absorpci komponent rozšířenou mimo jeden vrchol a může tak sledovat mnoho oblastí spektra, aby nalezla závislosti mezi absorbancí a koncentrací vzorků. Tento typ modelu je vhodný pro rozlišování komponent, které vytvářejí překrývající se skupiny ve spektrálních datech. Matematicky je CLS analýza simultánní aplikace rovnice Beerova zákona.[17]

CLS model může být založen na velikosti spektrálních špiček nebo měření plochy. Je nutné přiřadit alespoň jeden vrchol anebo oblast dat ke každé měřené komponentě. Obvykle se pro měření každé komponenty využívá více vrcholů a oblastí. Také je možné přiřadit více komponentám jeden vrchol či oblast. [17]

Pro dosažení co nejpřesnějších výsledků pomocí CLS, se musí dodržet následující pravidla:

a) Je možné najít alespoň jednu analyzovanou oblast pro každou komponentu
b) Metoda zahrnuje alespoň tolik kalibračních standardů, kolik je měřených komponent
c) Koncentrace komponent kolísá nezávisle (kdykoliv je to možné, vyhnout se lineérním závislostem komponent)
d) Jsou malé anebo žádné chemické reakce mezi komponentami
e) Matice vzorků je dobře známá
f) Závislost mezi absorbancí a koncentrací je téměř lineární

CLS metoda je vhodná pro řešení analytických problémů, které jsou příliš komplexní než aby se dali řešit pomocí jednoduchého Beerova zákona. Například je vhodné využít CLS metodu pro následující případy:

a) Vrcholy nebo oblasti komponent se významně překrývají
b) Základní spektrální čáry jsou proměnné

Materiály, které je možné analyzovat pomocí CLS zahrnují plyny a jiné vzorky, jejichž molekuly spolu nereagují nebo jen velice málo. [17][16]

2.7 Beer – Lambertův zákon

Metoda využívá data o koncentraci komponent a naměřená data absorbance ze standardů, pro vytvoření modelu. Model zobrazuje závislost absorbance na koncentraci. Body charakteristiky jsou proložené lineárně a nemají velkou vzdálenost od lineární křivky, kterou jsou proložené.[16]

Nejvyšší přesnosti metody se dosáhne, pokud se dodrží podmínky:

a) Spektrum standardu má pro každou komponentu, která je měřena, jeden dobře rozpoznatelný vrchol
b) Je k dispozici tolik standardů, kolik je měřených komponent
c) Každá komponenta je zahrnuta alespoň ve dvou různých standardech a dvou různých koncentrací
d) V případě že vzorek je směs, je zapotřebí, aby standard obsahoval všechny komponenty, které očekáváme v měřeném vzorku

.

Tato charakteristika modelu je obvykle nazývána jako pracovní křivka a pro každou komponentu se vytváří zvláštní pracovní křivka.[16]

Pro měření plynů obvykle stačí jeden standard pro srovnávání, naopak pro většinu pevných a kapalných vzorků jsou třeba dva standardy a koncentrace komponent. Pokud navíc očekáváme, že koncentrace komponenty ve vzorku se bude měnit ve velkém rozsahu, je dobré mít pro tento případ další standard pro přesnější měření.[16]

Pro srovnávání měřeného vzorku se standardy je vhodné, aby koncentrace komponent byly pokud možno stejné. Koncentrace komponenty měřeného vzorku by neměla být nižší než koncentrace komponenty ve standardu. Stejně tak, jako vzorek s nejvyšší koncentrací by neměl přesahovat koncentraci ve standardu. [16]

3 Program pro analýzu hlavních komponent v C#

V následující kapitole bude popsán program napsaný pro analýzu hlavních komponent, pro dvourozměrný soubor dat, který jsme získali měřením vzorku metodou FT-IR. Program je napsaný v programovacím jazyku C#, vyvinutým v roce 1972 pro operační systém UNIX.[18]

3.1 Popis souboru vstupních dat

Vstupní data jsou uloženy v souboru CSV, který má na každém řádku uloženu hodnotu osy X a osy Y. Hodnoty jsou oddělené středníkem a jsou ve formátu Ae^x. Hodnoty na řádcích bývají v některých případech oddělovány čárkou, nebo uvozovkami v případě, že v souboru dat budou potřeba čárky například pro desetinné číslo. Nejlepší možností však zůstává užití středníků, protože se v datech příliš často nevyskytují.

Soubory CSV se používají pro zpracování dat tabulkovými editory. Tabulkový editor Excel umí bez potíží otevírat soubory CSV a pracovat s nimi. V základním nastavení rozděluje data pomocí středníků. Je však možnost nastavení dělících znaků uživatelem.

3.2 Popis činnosti programu

Po startu programu je uživatel dotázán na cestu, kde je soubor uložen (cesta může vypadat následovně C:\dir\soubor.csv). Program vyzkouší, zda se soubor na daném umístění nachází. Pokud se soubor na daném umístění nenachází, vypíše se chybová informace na obrazovku a program je ukončen. V případě, že soubor je na daném umístění, program ho otevře pouze pro čtení. To je důležité, protože v takovém režimu program nemůže poškodit zdrojová data. V dalším kroku programu je uživatel vyzván, aby zadal místo pro uložení výsledných dat. Pokud se soubor nepodaří vytvořit (například kvůli omezení práv uživatele), vypíše se chybová informace a program se ukončí. Pokud se soubor pro uložení dat vytvoří, program postoupí k dalšímu kroku. Soubor pro výstupní data je otevřen jen pro zápis.

V dalším kroku program prochází soubor CSV po řádcích, načte jednu řádku a změní desetinné čárky na desetinné tečky. To je proto, že programovací jazyk C# je lokalizován pro anglický jazyk a funkce na zjištění lokalizace, podle místa kde je program spuštěn, nejsou příliš spolehlivé. Následně program načtený řádek rozdělí podle dělících znaků a přičte hodnoty do proměnných. Tyto hodnoty využije pro výpočet průměrů jednotlivých rozměrů. Z toho důvodu program počítá řádky, které načte a využije tuto hodnotu při výpočtu průměru.

Poté se ukazatel v souboru přesune z konce souboru na začátek a vypočítává se kovarianční matice podle vzorce (22). Pro dvourozměrná data teda program vytvoří dvourozměrnou matici se stejnými hodnotami na vedlejší diagonále. Z této matice jsou

vypočteny vlastní čísla (pro dvourozměrnou matici λ_1 a λ_2). Pomocí vlastních čísel program vypočítá vlastní vektory matice.

V posledním kroku programu se počítají výsledná data, která program ukládá do souboru, který uživatel zadal na počátku programu. Veškeré výpočty probíhali s desetinou tečkou, a proto je potřeba hodnoty upravit, aby se pro desetinná čísla používaly čárky. Tato úprava proběhne během každého výpočtu, před uložením do souboru. Na konci programu jsou všechny soubory uzavřeny a program se ukončí.

Během chodu programu se vypíšou na obrazovku průměry, kovarianční matice, vlastní čísla, vlastní vektory a počet upravených řádků. Všechny tyto hodnoty jsou také zaznamenány do logu, který se vytvoří ve složce, kde je program umístěn.

3.3 Použité knihovny a funkce

V programu jsou využity knihovny:

Stdio.h – Standardní knihovna pro vstupy a výstupy programu

Stdlib.h – Standardní knihovna pro operace v programu

String.h – Knihovna pro práci s řetězci

Math.h – Knihovna pro matematické operace

Knihovna stdio.h tato knihovna vstupů a výstupů využívá „proudy“ pro práci s fyzickými periferiemi, jako je klávesnice, tiskárny, terminály, rozhraní. Všechny „proudy“ v knihovně mají podobné vlastnosti a mírně se liší podle periferie, se kterou má za úkol komunikovat.

V knihovně stdlib.h jsou základní funkce, jako jsou například alokace paměti, generování náhodných čísel, základní převody a podobné funkce.[18]

String.h je knihovna, která umožňuje práci s řetězci. Ukládání řetězců do polí, vypisování některých částí řetězce, sčítání řetězců, dělení řetězců a podobné.[18]

Pomocí knihovny math.h je možné provádět základní matematické operace jako sinus, cosinus, mocnina na zadaný exponent, odmocnina a další. [18]

Všechny tyto knihovny jsou základní a jsou tedy přístupné pro každý překladač pro programovací jazyk C#. Pro náročnější výpočty je nutné si vytvořit vlastní knihovny nebo stáhnout knihovny z internetu a přečíst přiloženou dokumentaci.[18]

Funkce FILE * fopen(const char * filename, const char * mode); je funkce, která otevře soubor, který je určený parametrem filename a přiřazený k proudu FILE, kterým se mužeme na tento soubor odkazovat. Parametr mode určuje, jakým způsobem se bude pracovat se

souborem (čtení, zápis, čtení i zápis, doplňování). Ukazatel, který určuje proud do souboru, se odstraní funkcí fclose(FILE);.[18]

Funkce printf(); a fprintf(); jsou pro formátovaný výpis. Printf(); se využívá pro výpis na obrazovku a fprintf(); pro výpis do souboru. Ve funkci fprintf(); je třeba ještě zadat proud do, kterého se zapisuje. [18]

Funkcí scanf(); se využívá pro formátovaný vstup do programu. Ve funkci se zadá formát, v jakém mají být vstupní data načtena a adresa, na kterou se zapíše. Scanf(); se používá pro načítání dat zadaných uživatelem z klávesnice.[18]

Funkce if(); else; je funkce, která se na základě podmínky uvedené v závorce funkce if(); rozhoduje, jestli budou provedeny příkazy v bloku if() nebo v bloku else. Pokud je podmínka v závorce rovna 1 (někdy uváděno TRUE, PRAVDA) provede se blok programu v if();.[18]

While() je funkce pro cyklus, který je vykonáván, dokud podmínka v závorce platí. Příkaz testuje podmínku před průchodem cyklem, a proto cyklus nemusí být proveden ani jednou. V případě, že se podmínka stane neplatnou v průběhu bloku cyklu, doběhne program celým blokem až na konec a až poté je podmínka znovu testována.[18]

Cyklus for(;;) je funkce, která je prováděna, dokud jsou podmínky v závorce splněny. Cyklus nemusí proběhnout ani jednou. V cyklu je možné inicializovat proměnou, zadat podmínku jak dlouho se má příkaz provádět a výraz, který funkce provede při každém cyklu. Inicializace proměnné a výraz, který se provede jsou nepovinné. [18]

Funkce pow(); vrací zadanou hodnotu, umocněnou na zadaný exponent. První parametr funkce je číslo, které se má umocnit a druhým parametrem je exponent.[18]

Pomocí funkce fseek() je možné posouvat ukazatel v souboru. Jako první parametr se uvede soubor, se kterým se pracuje, druhý parametr určuje, o kolik se chceme posunout a třetí parametr určuje místo odkud se chceme posunout. [18]

Funkcí strtok() je možné rozdělit řetězec podle zadaných oddělovačů. Je tedy vhodná pro získávání hodnot z CSV souboru. Při prvním zavolání funkce si funkce uloží ukazatel a z tohoto místa začne hledat oddělovače. Vrací pak všechny znaky řetězce, dokud nenarazí na oddělovač a je třeba funkci zavolat znovu s ukazatelem místa, kde funkce skončila.[18]

3.4 Popis bloků programu

Na začátku programu se musí připojit knihovny. To se provede následovně:

#include <stdio.h>

#include <stdlib.h>

```
#include <string.h>
#include <math.h>
```

Kompilátor při svém spuštění odstraní komentáře v programu a zkompiluje program s funkcemi z knihovny, které jsou využity v programu.

Následující blok programu je pro otevírání souborů. V tomto případě otestování jestli jde soubor otevřít a jeho otevření.

```
if((soubor = fopen(s, "r"))==NULL){
  printf("Soubor nebyl nalezen...\n");
  fprintf(log,"Soubor nebyl nalezen...\n");
  system("PAUSE");
  fclose(log);
  exit(0);
 }
```

Pomocí následujícího bloku je provádění nahrazení desetinných čárek za desetinné tečky.

```
while(fscanf(soubor,"%s", hodnota)!= EOF){
  for(i=0;i<=89;i++){
    if(hodnota[i] == ',')hodnota[i] = '.';
  }
```

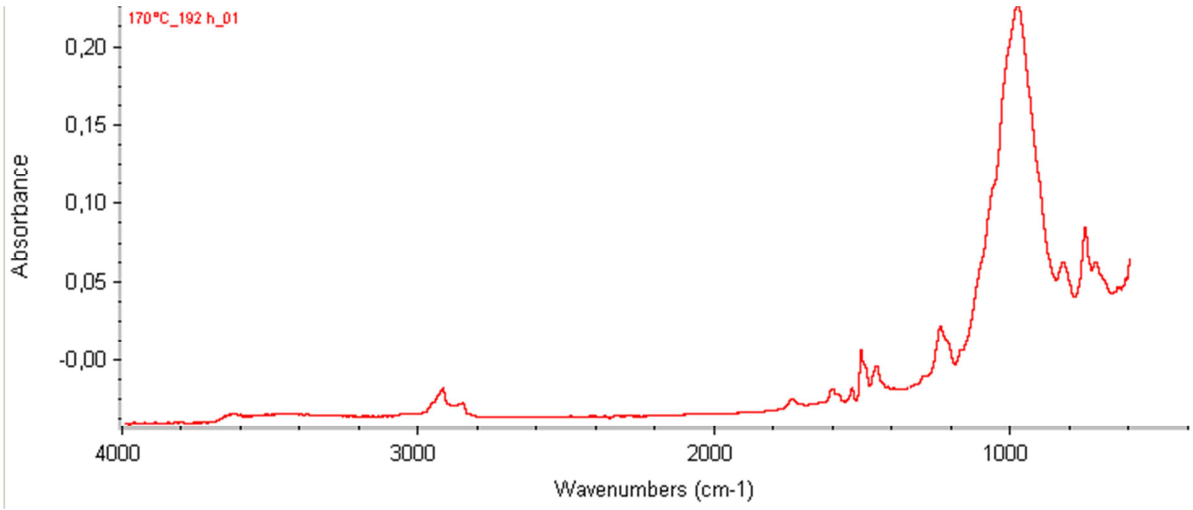
Funkce změní čárky na tečky v celém souboru dat. Pro samotné načítání dat je potom vyučit blok

```
while(fscanf(soubor,"%s", hodnota)!= EOF){
  for(i=0;i<=89;i++){
    if(hodnota[i] == ',')hodnota[i] = '.';
  }
  i = 0;
  token = strtok(hodnota, "e;");
  while(token != NULL){
    a[i] = atof(token);
    i++;
    token = strtok(NULL, "e;");
  }
Operace se vstupními daty
}
```

Bloky s výpočty jsou odděleny komentáři v příloze práce.

3.5 Testovací spektrum

Program byl testovaný na vzorku, který byl po dobu 192h vystaven teplotě 170°C. Spektrum vstupních dat je na obrázku 6.



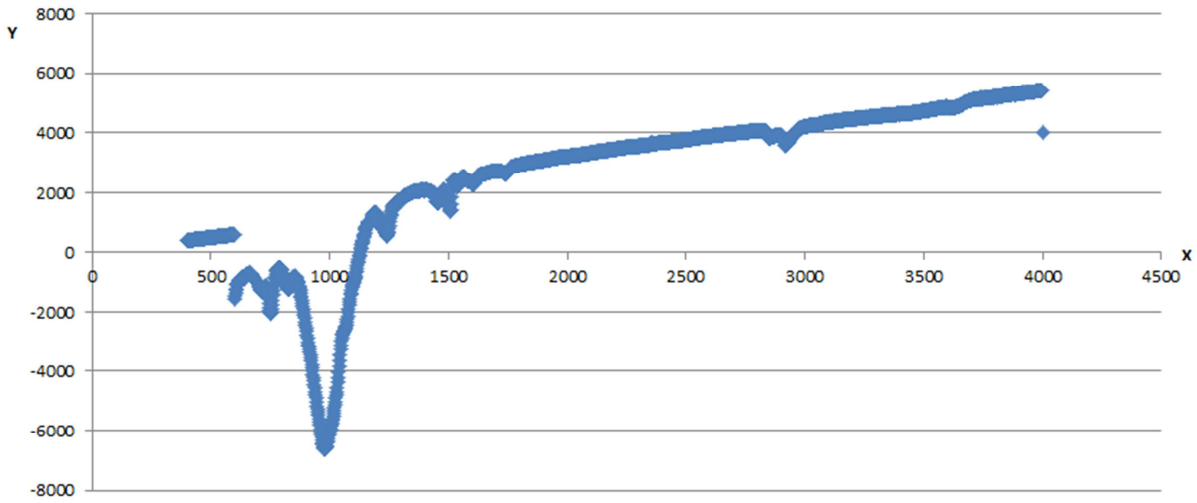
Obr.6: Spektrum signálu měřeného vzorku

Okno programu v průběhu výpočtů je na obrázku 7.

```
Zadejte cestu k souboru:
c:\vstup\vzorek.csv
Zadejte cestu, kam bude vystup ulozen:
c:\vstup\PCA.csv

Prumer X: 2200.489264,
Prumer Y: -0.011983

Kovariancni matice:
  1083270.445386      -32.107433
  -32.107433      0.002540

Vlastni cislo 1: 1083270.446337
Vlastni cislo 2: 0.001589

Vlastni vektor pro vlastni cislo lambda = 1083270.446337
1.000000      -33738.935035

Vlastni vektor pro vlastni cislo lambda = 0.001589
1.000000      0.000030

1869 zaznamu bylo zpracovano.Pokračujte stisknutím libovolné klávesy...
```

Obr.7:Výpis z programu při výpočtech

Výsledný graf po úpravě pomocí PCA je na obrázku 8.



Obr.8.:Graf po úpravě programem

Podle vlastního čísla 2, které je vidět na obrázku 7, je možné tento rozměr dat vypustit, protože mají velice malé vlastní číslo a tedy i velice malou důležitost. Jedná se o lineární data, která nenesou z hlediska PCA téměř žádnou informaci. Došlo by tak ke komprimaci dat na polovinu. Pro jasnější zobrazení je lepší nechat i tento rozměr.

Závěr

Infračervená spektrometrie je v dnešní době velmi často využívána pro nejrůznější aplikace, od zkoumání vzorků v laboratoři, až po zkoumání nečistot v ovzduší. Pro tyto aplikace existují různé měřící systémy a měřící metody. V diplomové práci jsou popsány některé z nich, jak probíhají a jak jsou vhodné pro různé materiály. Transmisní metody měření vzorků se hodí spíše pro plynné a kapalné vzorky, pokud však dokážeme udělat tenkou tabletu, kterou záření projde, je možné použít i materiály pevné. Pro pevné materiály jsou však vhodnější reflexní metody.

Pro analýzu naměřených dat je pak velké množství statistických nástrojů. Ne všechny se ale hodí pro analýzu infračervených spekter. Nejpoužívanější statistické nástroje pro analýzu infračervených spekter jsou analýza hlavních komponent PCA, CLS, PLS. Tyto statistické nástroje využívá většina softwarů, pro práci s infračervenými spektry.

PCA analýza je velmi často využívána pro zjišťování hlavních komponent souboru, a případné vypuštění některých dimenzí souboru dat, v případě, že v souboru nemají velkou váhu. Z toho vyplývá i možnost využití PCA pro komprimaci dat, při přenosu signálu, či ukládání na paměťové úložiště. PCA se využívá i ve forenzním zkoumání, například při rozpoznávání obličejů.

Program pro PCA napsaný v C# má přesnost výpočtů srovnatelnou s přesností, kterou je možné dosáhnout pomocí Excel. Je však důležité v programovacím jazyku C# dodržovat správné datové typy. Občasné potíže můžou nastat, pokud se dělí dva celočíselné typy int a výsledek se ukládá například do proměnné typu double. Přesto, že double je typ pracující s desetinnými čísly, výsledná hodnota bude stejná jako by se ukládalo do celočíselné proměnné int. Proto se musí celočíselné typy přetypovat na typ double.

Použitá literatura

[1] Mentlík, V.; Pihera, J.; Polanský, R.; Prosr, P.;Trnka, P. Diagnostika elektrických zařízení. 1.vyd. Praha : BEN - technická literatura, 2008.439 s. ISBN 978-80-7300-232-9.

[2] DERRICK, Michele R.; STULÍK, Dušan; LANDRY, James M. *Infrared Spectroscopy in Conservation Science*. Los Angeles: The Getty Conservation Institute, 1999. 235 s. ISBN 0- 89236-469-6 [3] G. N. Petrov.: *Elektrické stroje 2*; Academia Praha 1982

[3] SINICA, Alla. Spektrometrie ve viditelné oblasti spektra. [online]. [cit. 2014-04-20]. Dostupné z: http://www.vscht.cz/anl/lach1/5_Foto.pdf

[4] NOVÝ, Petr. Infračervené spektrometrie. [online]. [cit. 2014-04-20]. Dostupné z: userweb.pedf.cuni.cz/wp/kch/files/2010/10/ICSpektroskopie.pdf

[5] SMITH, Brian C. *Fundamentals of Fourier transforminfraredspectroscopy*. 2nd ed. BocaRaton, FL: CRC Press. ISBN 978-142-0069-303

[7] GRIFFITHS, Peter R a James A DE HASETH. *Fourier transform infrared spectrometry*. 2nd ed. Hoboken: John Wiley, 2007, 529 s. ISBN 978-0-471-19404-0

[8] *Michelsonův interferometer* [online]. 2009-11-02 [cit. 2014-04-23]. Dostupné z: https://moodle.fp.tul.cz/nano/pluginfile.php/1580/mod_resource/content/1/Michelson%C5%AFv%20interferometr.pdf

[9] *Vznik speciální teorie relativity* [online]. 2010-08-12 [cit. 2014-04-24]. Dostupné z: http://www.gymhol.cz/projekt/fyzika/16_vznik_str/16_vznik_str.htm

[10] KANIA, Patrik. Infračervená spektrometrie. [online]. [cit. 2014-04-25]. Dostupné z: http://www.vscht.cz/anl/lach1/7_IC.pdf

[11] I. SMITH, Lindsay. A tutorial to Principal Component Analysis. [online]. 2002 [cit. 2014-05-01]. Dostupné z: http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf

[12] MELOUN, Milan a Jiří MILITKÝ. *Kompendium statistického zpracování dat*. ACADEMIA. ISBN 80-200-1008-4.

[13] MELOUN, Milan a FREISLEBEN. *Klasifikace podzemních vod diskriminační metodou* [online]. [cit. 2014-05-3]. Dostupné z: http://meloun.upce.cz/docs/publication/226.pdf

[14] Meloun M., Militký J., Hill M.: Počítačová analýza vícerozměrných dat v příkladech. Academia, Praha 2005

[15] Meloun M., Militký J.: Statistická analýza experimentálních dat. Academia, Praha 2004

[16] *TQ Analyst User's Guide*. [cit. 2014-04-29].

[17] M. HAALAND, David. *Partial Least-squares for spectral analyses* [online]. 1988 [cit. 2014-05-04]. Dostupné z: http://www.imedea.uib-csic.es/master/cambioglobal/Modulo_V_cod101615/Theory/lit_support/analchem.pdf

[18] KERNIGHAN, Brian W a Dennis M RITCHIE. *The C programming language*. 2nd ed. Englewood Cliffs, N.J.: Prentice Hall, c1988, xii, 272 s. ISBN 01-311-0362-8.

PŘÍLOHA 1: ZDROJOVÝ KÓD PROGRAMU PRO PCA

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>

int main(int argc, char *argv[])
{
  FILE *soubor, *vystup, *log;
  char s[100], v[100];
  char hodnota[90], *token;
  int i = 0, pom = 0;
  float  pom_mat[4];
  double a[4], prumerX = 0, prumerY = 0, cov_mat[2][2] = {0,0,0,0};
  double polynom[3]= {0,0,0}, vl_1=0, vl_2=0;
  double soustava[2][3], vl_vektor1[2], vl_vektor2[2], pro_vypis[4];

  log = fopen("log.txt","w");
  printf("Zadejte cestu k souboru:\n");
  fprintf(log,"Zadejte cestu k souboru:\n");
  scanf("%s", s);
  fprintf(log,"%s\n", s);

  if((soubor = fopen(s, "r"))==NULL){
   printf("Soubor nebyl nalezen...\n");
   fprintf(log,"Soubor nebyl nalezen...\n");
   system("PAUSE");
   fclose(log);
   exit(0);
  }

  printf("Zadejte cestu kam, bude vystup ulozen:\n");
  fprintf(log,"Zadejte cestu, kam bude vystup ulozen:\n");
```

```
  scanf("%s", v);
  fprintf(log,"%s\n", v);

  if((vystup = fopen(v , "w"))==NULL){
    printf("Soubor se nepodarilo vytvorit...");
    fprintf(log,"Soubor se nepodarilo vytvorit...");
    system("PAUSE");
    fclose(soubor);
    exit(0);
  }
  //výpočet  průměru
  while(fscanf(soubor,"%s", hodnota)!= EOF){
    for(i=0;i<=89;i++){
      if(hodnota[i] == ',')hodnota[i] = '.';
    }
    i = 0;
    token = strtok(hodnota, "e;");
    while(token != NULL){
      a[i] = atof(token);
      i++;
      token = strtok(NULL, "e;");
    }
    a[0] *= pow(10 , (int)a[1]);
    a[2] *= pow(10 , (int)a[3]);
    pom++;
    prumerX += a[0];
    prumerY += a[2];
  }
  prumerX /= pom;
  prumerY /= pom;
  printf("\nPrumer X: %lf,\nPrumer Y: %lf", prumerX, prumerY);
  fprintf(log,"\nPrumer X: %lf,\nPrumer Y: %lf", prumerX, prumerY);

fseek(soubor, 0, SEEK_SET);
```

```
        //výpočet kovarianční matice
        pom = 0;
        while(fscanf(soubor,"%s", hodnota)!= EOF){
          for(i=0;i<=89;i++){
            if(hodnota[i] == ',')hodnota[i] = '.';
          }
          i = 0;
          token = strtok(hodnota, "e;");
          while(token != NULL){
            a[i] = atof(token);
            i++;
            token = strtok(NULL, "e;");
          }
          a[0] *= pow(10 , (int)a[1]);
          a[2] *= pow(10 , (int)a[3]);
          pom++;
          cov_mat[0][1] += ((a[0] - prumerX) * (a[2] - prumerY));
          cov_mat[0][0] += ((a[0] - prumerX) * (a[0] - prumerX));
          cov_mat[1][1] += ((a[2] - prumerY) * (a[2] - prumerY));
        }

        cov_mat[0][1] /= (pom - 1);
        cov_mat[1][0] = cov_mat[0][1];
        cov_mat[0][0] /= (pom - 1);
        cov_mat[1][1] /= (pom - 1);

        printf("\n\nKovariancni matice:");
        printf("\n %lf %lf\n %lf %lf\n", cov_mat[0][0], cov_mat[0][1], cov_mat[1][0],
cov_mat[1][1]);
        fprintf(log,"\n\nKovariancni matice:");
        fprintf(log,"\n %lf %lf\n %lf %lf\n", cov_mat[0][0], cov_mat[0][1],
cov_mat[1][0], cov_mat[1][1]);
      //výpočet vlastních čísel matice
        polynom[0] = 1;
```

```
        polynom[1] = ((-1)*(cov_mat[0][0]))+((-1)*(cov_mat[1][1]));
        polynom[2] = (cov_mat[0][0]*cov_mat[1][1])-(cov_mat[1][0]*cov_mat[0][1]);

        vl_1 = (((-1)*polynom[1])+sqrt(pow((polynom[1]),2) - 4 * polynom[0] *
polynom[2]))/2;                    //výpočet kvadratické rce
        vl_2 = (((-1)*polynom[1])-sqrt(pow((polynom[1]),2) - 4 * polynom[0] *
polynom[2]))/2;

        printf("\nVlastni cislo 1: %lf", vl_1);
        printf("\nVlastni cislo 2: %lf", vl_2);
        fprintf(log,"\nVlastni cislo 1: %lf", vl_1);
        fprintf(log,"\nVlastni cislo 2: %lf", vl_2);
      //výpočet vlastních vektorů matice
        soustava[0][0]=cov_mat[0][0]-vl_1;
        soustava[0][1]=cov_mat[0][1];
        soustava[0][2]=0;
        soustava[1][0]=cov_mat[1][0];
        soustava[1][1]=cov_mat[1][1] - vl_1;
        soustava[1][2]=0;

        soustava[0][2]=soustava[0][1]*(-1);
        vl_vektor1[0] = 1;
        vl_vektor1[1] = (1/soustava[0][0])*soustava[0][2];
        printf("\n\nVlastni vektor pro vlastni cislo lambda = %lf\n%lf      %lf", vl_1,
vl_vektor1[0], vl_vektor1[1]);
        fprintf(log,"\n\nVlastni vektor pro vlastni cislo lambda = %lf\n%lf      %lf", vl_1,
vl_vektor1[0], vl_vektor1[1]);

        soustava[0][0]=cov_mat[0][0] - vl_2;
        soustava[0][1]=cov_mat[0][1];
        soustava[0][2]=0;
        soustava[1][0]=cov_mat[1][0];
        soustava[1][1]=cov_mat[1][1] - vl_2;
        soustava[1][2]=0;
```

```
        soustava[0][2]=soustava[0][1]*(-1);
        vl_vektor2[0] = 1;
        vl_vektor2[1] = (1/soustava[0][0])*soustava[0][2];
        printf("\n\nVlastni vektor pro vlastni cislo lambda = %lf\n%lf %lf", vl_2, vl_vektor2[0], vl_vektor2[1]);
        fprintf(log,"\n\nVlastni vektor pro vlastni cislo lambda = %lf\n%lf %lf", vl_2, vl_vektor2[0], vl_vektor2[1]);

      fseek(soubor, 0, SEEK_SET);
       //výpočet dat PCA a ukládání do souboru
        pom = 0;
        while(fscanf(soubor,"%s", hodnota)!= EOF){
          for(i=0;i<=89;i++){
            if(hodnota[i] == ',')hodnota[i] = '.';
          }
          i = 0;
          token = strtok(hodnota, "e;");
          while(token != NULL){
            a[i] = atof(token);
            i++;
            token = strtok(NULL, "e;");
          }
          a[0] *= pow(10 , (int)a[1]);
          a[2] *= pow(10 , (int)a[3]);
          pom++;

            pro_vypis[0]=(int)(vl_vektor1[0]*a[0])+(vl_vektor1[1]*a[2]);
            pro_vypis[1]= (((vl_vektor1[0]*a[0])+(vl_vektor1[1]*a[2]))-pro_vypis[0])*100000;
            pro_vypis[2]= (int)(vl_vektor2[0]*a[0])+(vl_vektor2[1]*a[2]);
            pro_vypis[3]= (((vl_vektor2[0]*a[0])+(vl_vektor2[1]*a[2]))-pro_vypis[2])*100000;

            fprintf(vystup,"%d,%d;%d,%d\n",(int)pro_vypis[0],(int)pro_vypis[1], (int)pro_vypis[2], (int)pro_vypis[3]);
```

```
    }
  printf("\n\n%d zaznamu bylo zpracovano.", pom);
  fprintf(log,"\n\n%d zaznamu bylo zpracovano.", pom);

    system("PAUSE");

    fclose(soubor);
    fclose(vystup);
    fclose(log);
}
```