

# Projekt ISS 2019/20

## 1 Namluvené věty

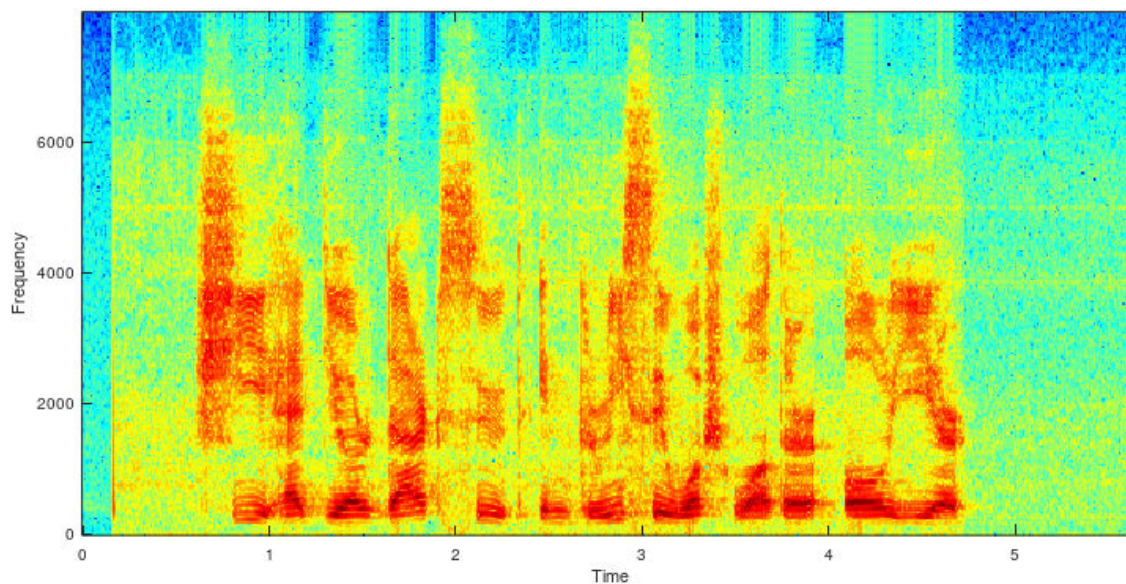
Název souboru	Délka v sekundách	Počet vzorků
sa1.wav	5.69	91000
sa2.wav	5.19	83000
si1071.wav	6.94	111000
si1701.wav	5.44	87000
si2331.wav	6.19	99000
sx261.wav	3.69	59000
sx27.wav	3.56	57000
sx351.wav	5.12	82000
sx441.wav	4.16	66500
sx81.wav	3.69	59000

Přeji si, aby tato data byla použita jen pro vyhodnocení tohoto ISS projektu (možnost (a)).

## 2 Hledaná slova

Název souboru	Délka v sekundách	Počet vzorků
q1.wav	0.91	14491
q2.wav	0.77	12320

## 3 Spektrogram signálu



*sa1.wav*

## 4 Funkce pro výpočet parametrů

Pro výpočet parametrů jsem použil lineární banku filtrů, přičemž jsem zvolil 16 parametrů.

Při maticovém součinu  $F = AP$  by matice  $A$  měla 16 řádků a 256 sloupců. Na prvním řádku by bylo 16 jedniček, zbytek nuly. Na každém dalším řádku by se řada 16 jedniček posunula dál o 16 pozic. Poslední řádek by tedy byly nuly zakončené sérií 16 jedniček.

## 5 Výpočet skóre klíčového slova

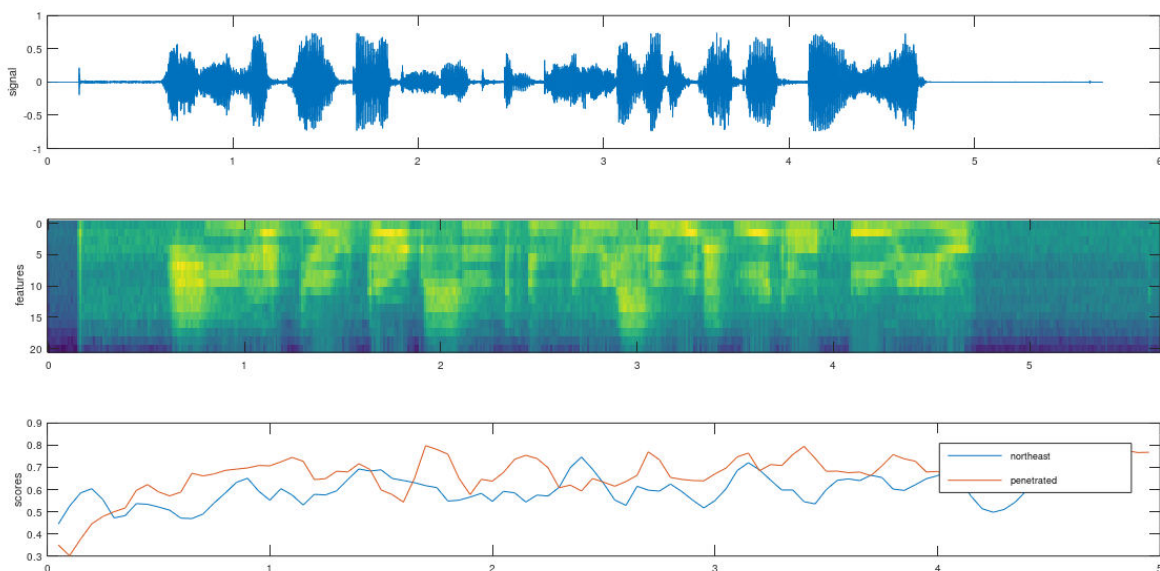
Pro výpočet skóre jsem použil výpočet Pearsonova korelačního koeficientu pro 2 matice sloupcových vektorů. První matice obsahuje parametry query pro všechny rámce, druhá parametry věty pro stejný počet rámců, jako má query. Překrytí těchto matic se po jednom kroku posouvá o pět rámců pro zachování rychlosti výpočtu.

```
score_index = 1; %index matice skóre
%posouvej se po peti ramcich, poslední pozice se plně překryva
for index = 1:5:pocet_ramcu_vety - pocet_ramcu_q1
    score1(score_index) = corr2(Fq1, features(:, index:index+pocet_ramcu_q1-1)); %vypočet koeficientu
    score_index = score_index + 1; %posun indexu
endfor
```

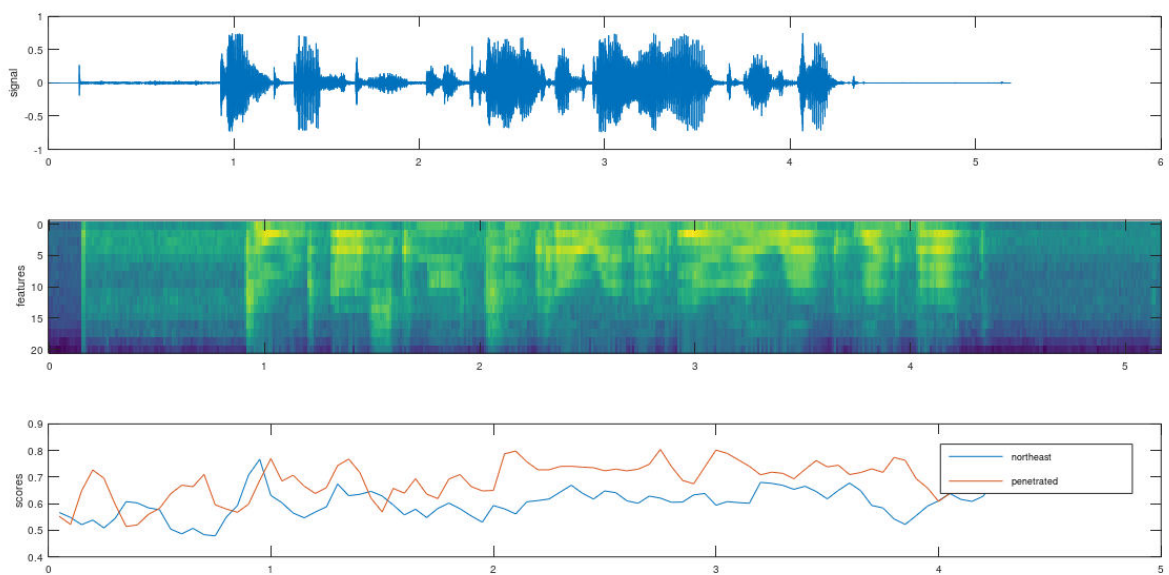
*Kód pro výpočet skóre*

## 6 Skóre jednotlivých vět

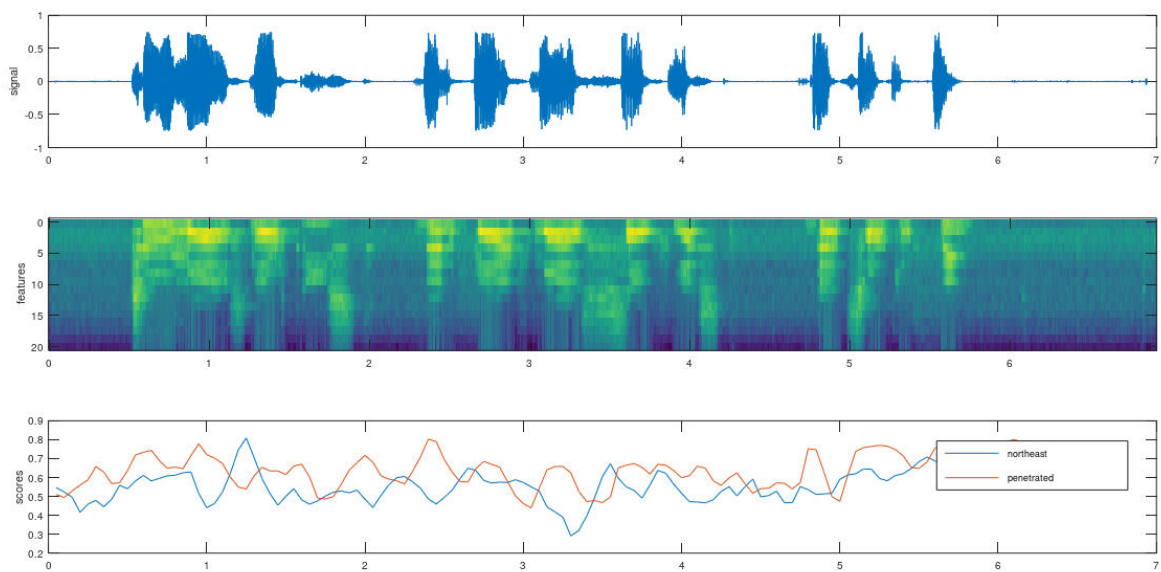
„northeast“ and „penetrated“ vs. sa1



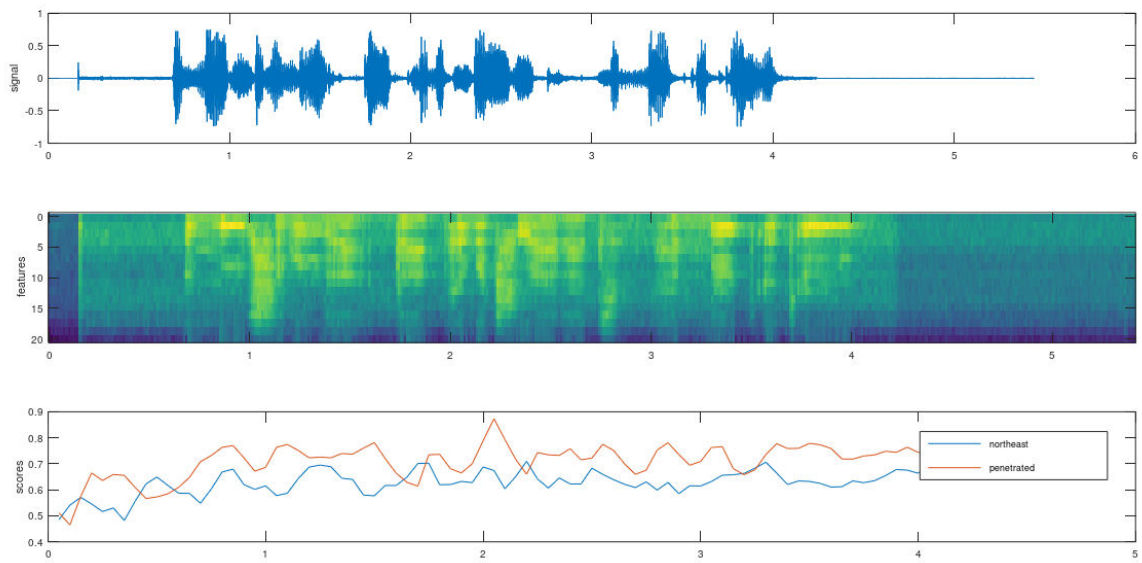
### „northeast“ and „penetrated“ vs. sa2



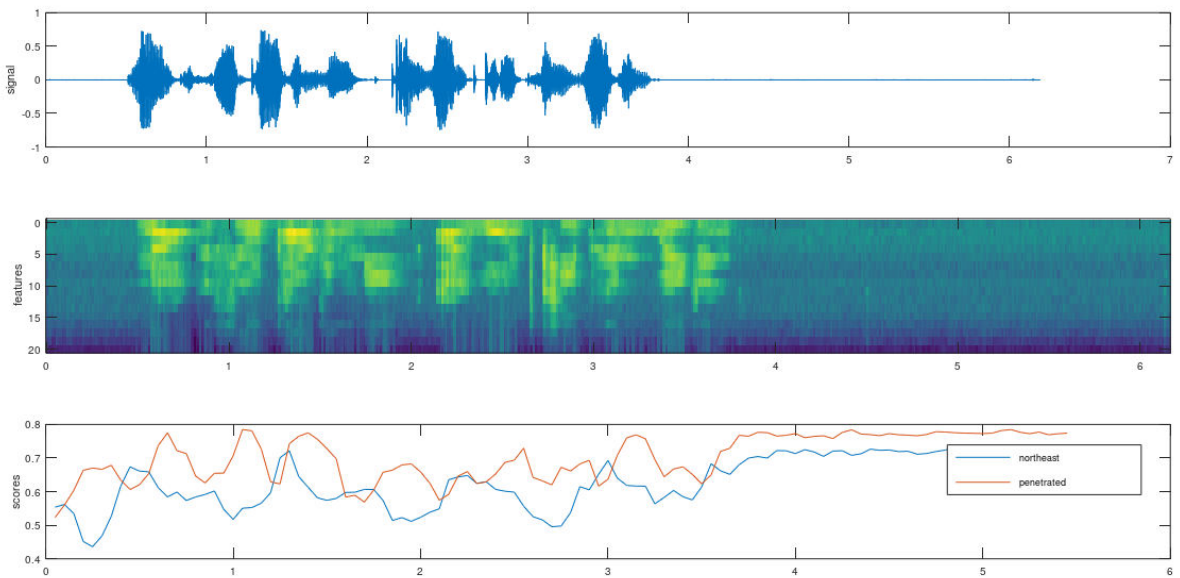
### „northeast“ and „penetrated“ vs. si1071



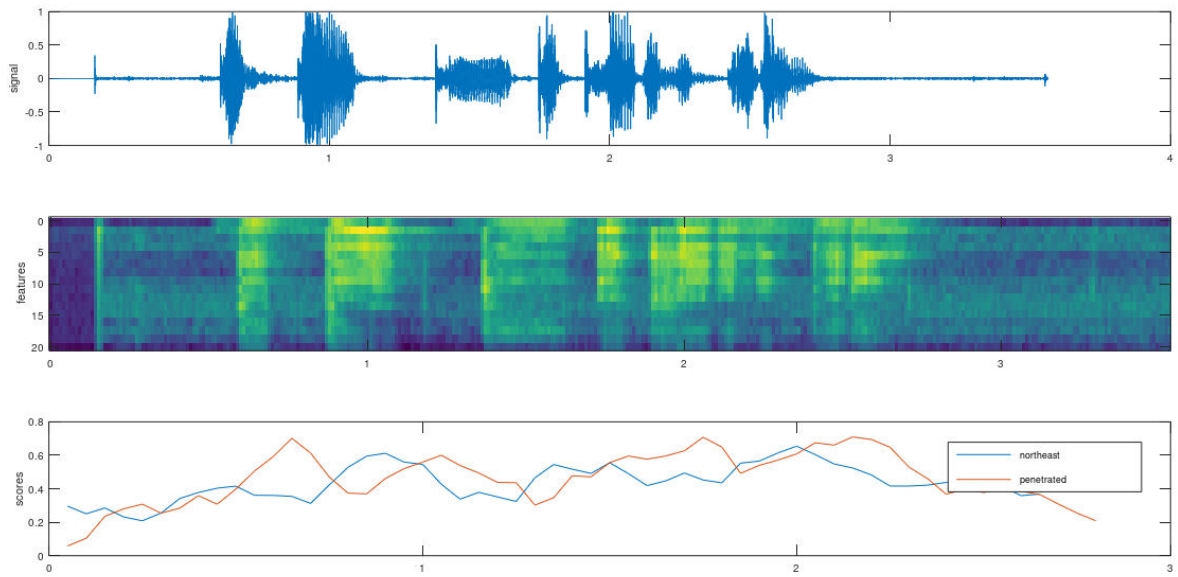
### „northeast“ and „penetrated“ vs. si1701



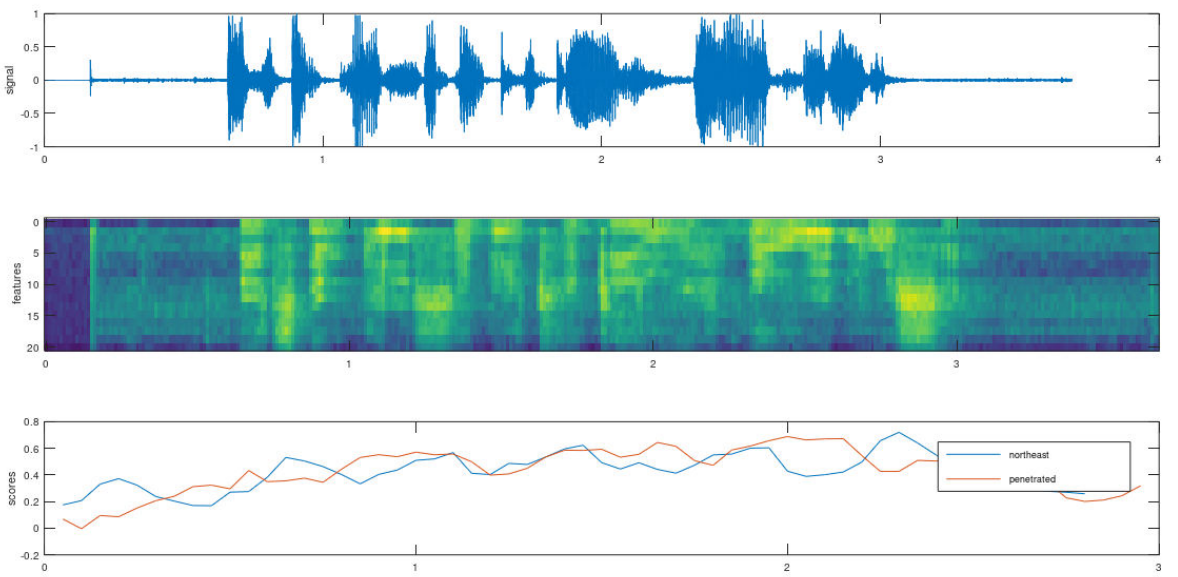
### „northeast“ and „penetrated“ vs. si2331



„northeast“ and „penetrated“ vs. sx27

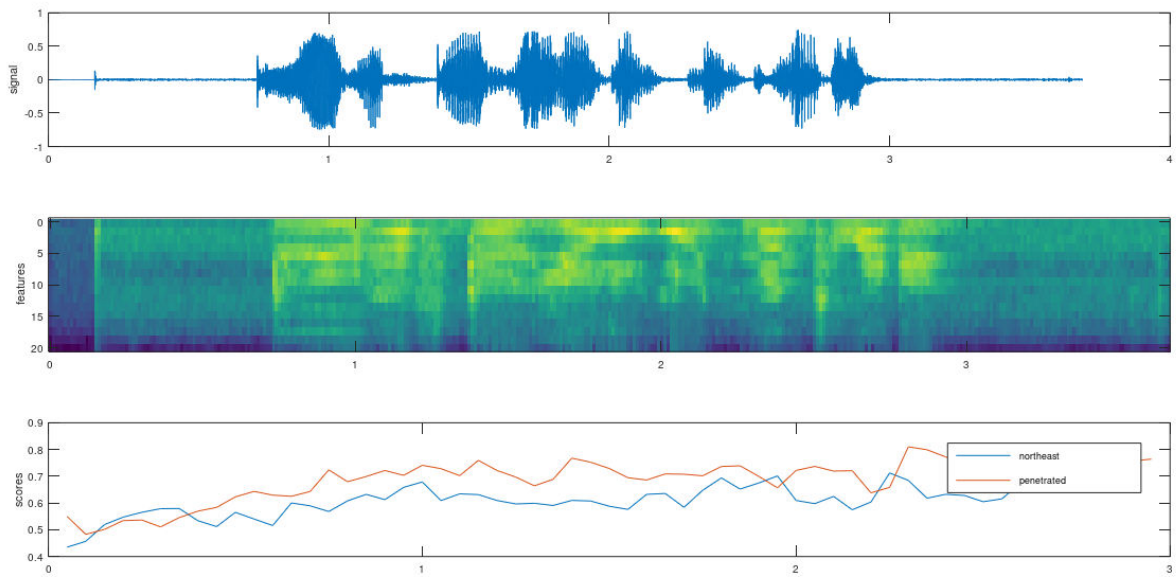


„northeast“ and „penetrated“ vs. sx81

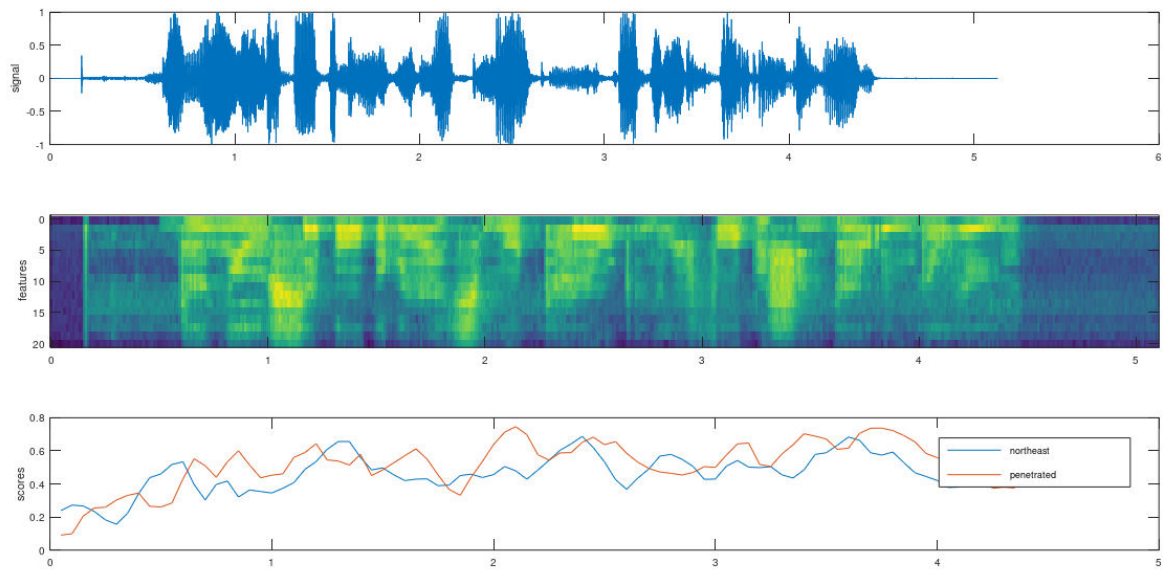




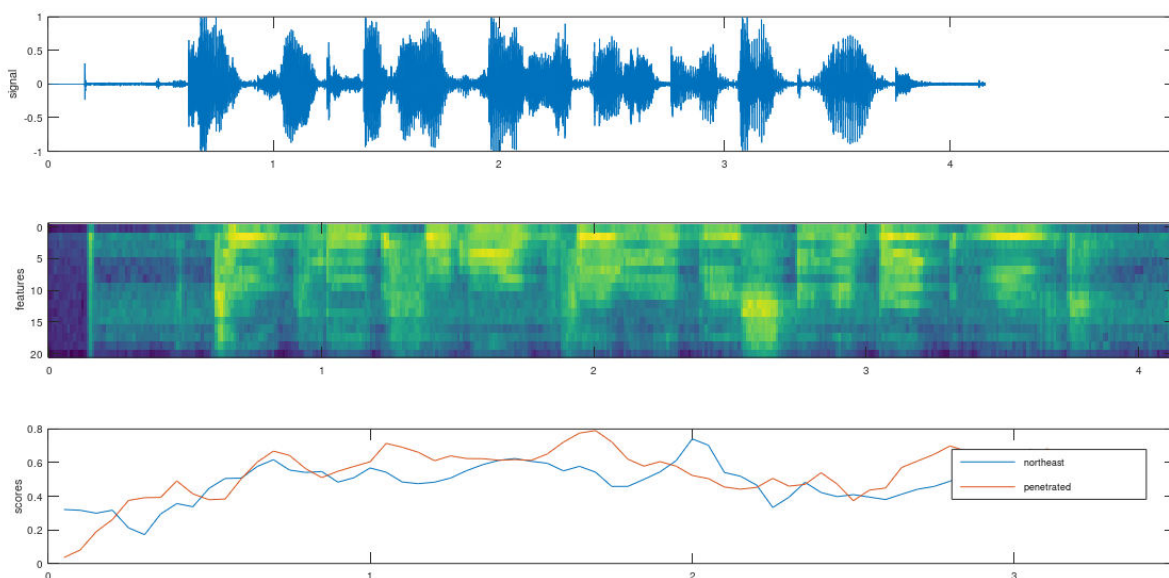
### „northeast“ and „penetrated“ vs. sx261



### „northeast“ and „penetrated“ vs. sx351



## „northeast“ and „penetrated“ vs. sx441



## 7 Určení rozhodovacího prahu

Rozhodovací práh určím tak, aby v každé větě prahem prošlo jen to místo, kde by se moje hledané slovo mělo nacházet. Moje Každé moje hledané slovo se nachází jen na jednom místě, stačí mi se tedy orientovat jen podle těchto dvou míst, najít v nich lokální maximum a nechat určitou rezervu. Pro q1.wav to bude tedy 0.805 a pro q2.wav 0.873.

## 8 Výsledky hledání

Název souboru	Výskyt q1	Výskyt q2	Rozmezí [vzorky]
sa1.wav	ne	ne	-
sa2.wav	ne	ne	-
si1071.wav	ano	ne	19 360-33 600
si1701.wav	ne	ano	32 160-44 160
si2331.wav	ne	ne	-
sx261.wav	ne	ne	-
sx27.wav	ne	ne	-
sx351.wav	ne	ne	-
sx441.wav	ne	ne	-
sx81.wav	ne	ne	-

## 9 Závěr

Výsledky jsou pro mě celkem uspokojivé. Po celkem přesné aplikaci úrovně rozhodovacího prahu prošly opravdu jen ty místa v signálech, kde se hledaná slova nachází, avšak na hodně místech nahrávky bylo skóre dost blízko ke prahu, i když na daném místě se slovo nenacházelo. Úspěšný výsledek připisuji hlavně tomu, že jsem se při nahrávání zvuku snažil svoje queries nahrát tak, aby zněly co nejbližší ke slovům ve větách. Také je dost možné, že jsem měl jen štěstí. Každopádně se experiment zdařil na výbornou.