

Sammendrag

De fleste land benytter i dag regelverk for regulering av støyplage fra vindturbiner som er basert på lydtrykknivå, mens noen land bruker differensielle støygrenser som lydfremvekst (1). I denne sammenhengen har det blitt utført to ulike lyttetester for å undersøke sammenhengen mellom rangert ubehag og ulike indikatorer. Testene besto av lydeksempler av vindturbinstøy og bakgrunnsstøy satt sammen med ulike lydfremvekster og lydtrykknivå. Det ble også beregnet hørestyrkenivå for de enkelte lydeksemlene. Støy fra vindturbin kommer fra opptak gjort av Electricité de France i Frankrike. Bakgrunnsstøykilder benyttet var fuglelyder og lyd fra veitrafikk.

Testresultatene indikerer at opplevd ubehag hos lytteren henger tettere sammen med lydtrykknivå og hørestyrke enn lydfremvekst. Lydfremvekst kan synes å ha mer innflytelse for rangert ubehag for høyere lydtrykknivå enn lavere.

Det ble også funnet at vindturbinstøy oppleves som mindre ubehagelig med fuglelyder som kilde til bakgrunnsstøy sammenlignet med veitrafikk.

Testmetodene som ble brukt var elementvis vurdering og parvis sammenligning og ble gjennomført med henholdsvis 30 og 31 deltagere. Testmetode synes ikke å ha betydning for sammenhengen mellom sansing og kvanitativ vurdering.

Abstract

Today most countries use regulations for noise pollution from wind turbines that is based on sound pressure levels, while some countries use differential noise limits as sound emergence (1). In this context two different listening tests has been implemented to examine the relationship between rated annoyance and different sound indicators. The tests consisted of sound examples made by wind turbine noise and background noise with different sound emergence and sound pressure levels. Loudness was also calculated for each sound example. The recordings of wind turbine noise was done by Electricité in France. The sources of background noise consisted of sound from birds and road traffic.

The test results indicate that perceived annoyance is closer related to sound pressure level and loudness rather than sound emergence. Sound emergence may appear to have more influence on the ranked annoyance for higher sound pressure levels than lower.

It was also found that wind turbine noise is experienced as less unpleasant with bird sounds as source of background noise compared to road traffic.

The methods used in the tests was direct scaling and pairwise comparison and was conducted with 30 and 31 participants respectively. The method does not seem to influence between sensing of quantitative assessment.

Forord

Det er med delte følelser jeg med denne oppgaven avslutter mastergraden i akustikk ved Norges teknisk-naturvitenskapelige universitet. Det har vært 5 og et halvt år med lange arbeidsdager og mye slit, men også stor begeistring og mestringsfølelse. Studiet på Gløshaugen startet som en utfordring til meg selv, om at jeg kunne få det til. Nå tror jeg at jeg kan sette punktum.

Arbeidet jeg har gjort det siste semesteret har gått mye lettere enn forventet, takket være veileder Guillaume Dutilleux. Takk for rask respons og konkrete løsninger, og ikke minst for tålmodigheten og for at du alltid tror at ting ordner seg. Jeg vil i grunn også takke alle professorene på akustikk for at dere alltid har “åpen dør” og viser stort engasjement.

Tim Cato Netland har generøst bistått med veiledning og utlån av teknisk utstyr. Jeg er også evig takknemmelig for at jeg fikk låne brukergrensesnittet Jean Fosset utviklet til sin praksisplassoppgave. Jeg vil også takke alle 61 lyttekandidater som tok seg til å høre på vindturbiner en times tid!

Studieveileder Astrid Hatlen har fulgt meg helt siden jeg ved en tilfeldighet valgte å bytte linje til Elsys og etterhvert akustikk. Hennes standardsvar har vært at “Det årne vi!” og jeg hadde aldri kommet i mål hvis ikke Astrid hadde vært så løsningsorientert.

Takk til studievenninne Live for smittende engasjement og for at du går fram som et godt eksempel på at hardt arbeid lønner seg. Og ikke minst takk for hjelpen med siste innsjutt.

Utdanningen min har det siste året blitt et prosjekt hvor mange har vært engasjert. Tusen millioner takk til mamma, pappa, svigerforeldre, søstre, tanter, Thale, Ida-Marie og Anne-Elisabeth som har bidratt slik at logistikken har gått opp.

Sist men ikke minst takk til Edvard som får meg tidlig opp hver morgen, og takk til Mathias som holder ut med meg selv når jeg får panikk, - og som trygt får meg ned på jorden igjen.

Aurora Elinsdatter Bang 19.01.20

Innhold

Sammendrag	i
Abstract	iii
Forord	v
Innholdsliste	vii
1 Innledning	1
1.1 Bakgrunn	1
1.2 Tidligere arbeid	1
1.3 Problemformulering	2
1.4 Framgangsmåte	2
1.5 Avgrensning og oppbygging av oppgaven	2
2 Teoretisk bakgrunn	5
2.1 Signal-støyforhold og lydfremvekst	5
2.2 Hørestyrke	6
2.3 Maskering	7
2.4 Lyttetester	7
2.4.1 Typer lyttetester	8
2.4.2 Vurdering av ubehag fra støy	8
2.5 Støy fra vindturbiner	8
2.6 Regelverk	9
2.7 Hovedkomponentanalyse	9
3 Metode	11
3.1 Innsamling av lydfiler	11
3.1.1 Vindturbinestøy	11
3.1.2 Bakgrunnsstøy	11
3.2 Metode brukt i testene	12
3.2.1 Test 1: Direkte skalering	12
3.2.2 Test 2: Parvis sammenligning	12
3.3 Bearbeiding av lydfiler	12
3.4 Brukergrensesnitt	12

3.5	Oppsett og gjennomføring av lyttetester	13
3.5.1	Oppsett	14
3.5.2	Testgjennomgang	14
3.5.3	Kalibrering av hodetelefoner	14
4	Resultat	17
4.1	Lydeksempler	17
4.1.1	Vindturbinstøy	17
4.1.2	Bakgrunnsstøy	17
4.2	Kalibrering av hodetelefoner	18
4.3	Kandidater	19
4.3.1	Fordeling av kandidater test 1	19
4.3.2	Fordeling av kandidater test 2	20
4.4	Bakgrunnsstøy og forstyrrelser	21
4.4.1	Måling av bakgrunnsstøy	21
4.4.2	Forstyrrelser under testen	22
4.5	Lyttetester	23
4.5.1	Resultat test 1	23
4.5.2	Resultat test 2	25
5	Diskusjon og vurderinger	29
5.1	Vurdering av gjennomføring av lyttetester	29
5.1.1	Lokale	29
5.1.2	Lydeksempler	29
5.1.3	Utvalg kandidater	30
5.1.4	Utførelse	30
5.2	Vurdering av testresultat	32
5.2.1	Resultat test 1	32
5.2.2	Resultat test 2	32
5.2.3	Sammenligning av resultat fra test 1 og test 2.	33
5.2.4	Sammenligning av testmetode	34
6	Konklusjon	35
A	Utstyrslister	39
A.1	Utstyrliste benyttet til opptak av bakgrunnskilder	39
A.2	Utstyr benyttet på lyttetester	39
A.3	Utstyr benyttet til kalibrering av hodetelefoner	40
B	Liste over lydeksempler	41
B.1	Lydeksempler test 1	41
B.2	Lydeksempler test 2	43
C	Skript for bearbeiding av lydfiler	45
C.1	Hjelpesfunksjon, Avekting	46
C.2	Hjelpesfunksjon, snrskalering	46
C.3	Hjelpesfunksjon, skalering	47

D	Kalibreringsfilter	49
E	Kalibrering av lydtrykknivå	53
E.1	Hjelpefunksjon, signalrms	54
F	Introduksjonstekst til test	55
F.1	Introduksjonstekst test 1	55
F.2	Introduksjonstekst test 2	55
G	Kode brukergrensesnitt	57
G.1	Test 1	57
G.2	Test 2	67
H	Bilder av brukergrensesnitt	79
I	Måling bakgrunnsstøy	83
I.1	Test 1	83
I.2	Test 2	84
J	Resultat lyttetester	85
J.1	Test 1	85
J.2	Test 2	86

Innledning

1.1 Bakgrunn

Allerede i 1891 ble det gjort forsøk på å fremstille elektrisk strøm fra vindkraft (2). 100 år senere kunne vindturbiner benyttes som en suksessfull energikilde og økningen av opprettelse av vindkraftparker har fortsatt i senere tid. Vindturbiner regnes som miljøvennlige i den grad at de ikke forurensar eller har andre utslipp, men har blitt kritisert for å påvirke miljøet på andre måter (2). Kritikken gjelder blant annet tårnene skygger for sol, de kan interferere med radio og TV-signal, påvirke plante og dyreriket rundt, visuell effekt i landskapet samt forårsaking av støy til omgivelsene rundt. Klagene på støy gjelder både hørbar støy, men også uhørbar infralyd.

En av utfordringene med støy fra vindturbiner er at de ofte plasseres i områder hvor det er lite annen menneskeskapt støy, noe som kan gjøre at selv lave lydnivåer kan høres godt. Dessuten er støy fra vindturbiner lik både for dag og natt. Vindturbinestøy har også vist seg å være mer forstyrrende enn støy fra vei, tog og fly (3) og derfor er anbefalingene for støygrense fra vindturbin strengere enn støy fra samferdsel og industri i Norge.

Ubehag som følge av støy kan lede til søvnvansker som igjen kan lede til psykiske nedsettelseser (4). Spørreundersøkelser har vist at irritasjon for vindturbiner øker med økt lydnivå (4). De fleste land benytter grenseverdien relatert til absolutte lydtrykknivå, mens noen land benytter også differensielle støyindikatorer som er relative til bakgrunnsstøy. En av disse indikatorene er lydfremvekst (*eng:sound emergence*). I fra litteraturen er det ikke funnet noe bevis på at lydfremvekst er en bedre egnet parameter enn lydtrykksnivå for å rangere ubehag (1). Det er heller ikke funnet bevis for at lydfremvekst er direkte knyttet til hørbarheten til en spesifikk lyd.

1.2 Tidligere arbeid

Flere studier har studert sammenhengen mellom støyindikatorer og opplevelse av støy gjennom bruk av lyttetester. En test gjort på opplevelse av høresterke og ubehag på vindturbinestøy i kombinasjon med ulike bakgrunnslyder fant at høresterke og ubehag reduseres når vindturbinestøy presenteres samtidig med annen bakgrunnsstøy (5). Resultatet indikerte også at opplevd høresterke og ubehag reduseres hvis bakgrunnsstøyen overstiger lydnivået til vindturbinestøyen. Det ble også funnet at bakgrunnsstøy påvirker oppfatningen av vindturbinestøy på et høyere nivå enn tidligere antatt, og at maskering av vindturbinestøy med hyggelige omgivelseslyder kan brukes for å forbedre lydforhold i områder utsatt for vindturbinestøy. Vurdering av opplevd høresterke

ble mer påvirket av bakgrunnsstøy enn vurdering av ubehag.

En tidligere masteroppgave har studert om ubehag fra vindturbinstøy ved ulike lydfremvekster er nivåavhengig (6). Denne sammenhengen ble bekreftet og man fant at avhengigheten er størst når lydfremveksten er størst, og avtagende når lydfremveksten minker. I tillegg ble også hørbarhet ved de laveste lydfremvekstene og lydnivåene undersøkt og resultatet viste at hørbarhet avhenger mer av lydnivå enn lydfremvekst.

I en lyttetest gjort med industristøy og bakgrunnsstøy ved ulike lydfremvekster ble det funnet at ubehag avhenger både av lydfremvekst, type støykilde og type bakgrunnsstøy, men at det ikke finnes noen signifikant interaksjon mellom de ulike faktorene (7). Det ble registrert at ubehaget kan være ulikt for ulike støykilder med samme lydnivå. Hørbarheten avhenger også av spekteret til støyen. En bredspektret støy kan ha en grense for hørbarhet på 3 dB(A), mens smalbandet støy kan være hørbar selv ved 0 dB(A).

1.3 Problemformulering

Målet for denne studien er å studere hvordan ulike kvantitative støyindikatorer korreleres med opplevelse, vurdert som ubehag, ved eksponering av støy fra vindturbiner i samspill med annen bakgrunnsstøy. Dokumentasjon innhentes ved gjennomføring av lyttetester. Indikatorer av interesse er lydtryknivå, hørestyrke og lydfremvekst. Et sidemål er å vurdere om typen lyttetest kan ha innflytelse på sammenhengen mellom sansning og kvantitativ vurdering. To ulike former for lyttetester benyttes, elementvis vurdering og parvis sammenligning.

1.4 Framgangsmåte

To ulike lyttetester lages med utgangspunkt i 2 lydklipp av vindturbinstøy og 2 lydklipp av bakgrunnsstøy. Lydklipp av vindturbinstøy innhentes fra opptak gjort i Frankrike. Bakgrunnsstøykilder hentes fra opptak gjort av fuglesang i Rotvollbuka, og egne opptak gjøres for støy fra veitrafikkstøy. Vindturbinstøy og bakgrunnsstøy settes sammen til lydeksempler med ulike lydnivå og lydfremvekst med hjelp av programvaren Matlab. Lydfilene filtreres videre for å tilpasses frekvensresponsen til hodetelefonene som benyttes i testene og det gjøres beregninger for å kunne gjennomføre en kalibrert lyttetest.

Brukergrensesnitt for to ulike lyttetester videreutvikles fra et grensesnitt designet av Jean Fosset i programvaren Python (8). Metode som benyttes i test 1 er elementvis vurdering, mens test 2 er parvis sammenligning. Det skal gjøres en gjennomkjøring av testene med en deltager før teststart.

1.5 Avgrensning og oppbygging av oppgaven

Arbeidet presenteres i 6 kapitler. Det første kapitlet danner basen for problemet som skal utforskes. Videre presenteres definisjoner av nødvendige begrep og relevant teori som utførelsen er basert på. Kapittel 3 presenterer metode for behandling av lydfiler, utvikling av brukergrensesnitt, kalibrering av utstyr og gjennomføring av lyttetestene. Resultatene i kapittel 4 er delt inn i 5 deler fordelt som resulterende lydeksempler, kalibrering av hodetelefoner, fordeling av kandidater, bakgrunnsstøy og forstyrrelser og resultatet til lyttetestene. Diskusjonsdelen tar for seg de samme temaene som presentert under resultat, samt en sammenligning av resultatene mellom de

to testene. Avslutningsvis sammenfattes betraktninger og oppdagelser gjort gjennom arbeidet i kapittel 6.

I denne oppgaven er ordet *ubehag* bruk for det engelske ordet *annoyance*. Ubehag innebefatter i denne sammenhengen både irritasjon, plage, misnøye og generelt forstyrrelse forårsaket av en lyd.

Teoretisk bakgrunn

Lyttetester baserer seg på hvordan en lytter opplever ulike lydklipp. I denne sammenhengen skal lydklipp med ulik lydfremvekst og lydnivå framstilles. For å danne lydklipp med ulike lydfremvekster er det tatt utgangspunkt i sammensetting av ulike signal-støyforhold.

Lyder kan oppleves ulikt ut i fra frekvensspekteret og lydstyrken. Mens lydnivå baseres kun på lydstyrken tilpasses hørestyrke også frekvensspekteret til lyden. Denne enheten kan derfor være med på å beskrive mer detaljerte forskjeller mellom lyder. Hvorfor noen lyder oppleves som mer irriterende eller fremtredende enn andre kan også forklares av maskering.

For at lytteren skal kunne skille mellom ulike lydeksempler er det viktig at lyttetesten legger til rette for vurdering på en god måte. Delkapittel 2.4.2 presenterer noen anbefalinger omkring dette. Delkapittel 2.5 trekker videre frem noen karakteristikk med vindturbinstøy og delkapittel 2.6 omhandler eksempler på regelverk som benyttes i dag for å ivareta områdene rundt en vindturbin.

For behandling av resultatene er det benyttet hovedkomponentanalyse, delkapittel 2.7 beskriver hovedprinsippene med denne analysen.

2.1 Signal-støyforhold og lydfremvekst

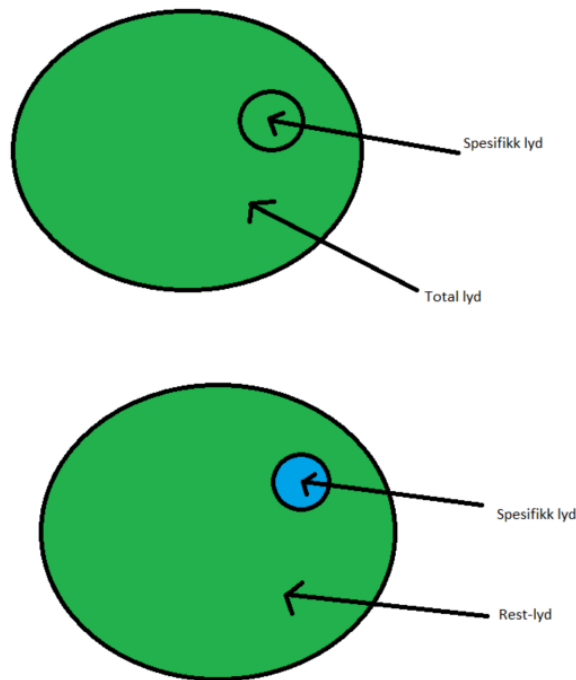
Signal-støyforhold (signal-to-noise ratio), forkortet SNR, beskriver et forhold i desibel mellom et spesifikt signal og støy, som vist i nederste modell i figur 2.1. For et spesifikt akustisk lydtrykk, \tilde{p}_{spes}^2 , og rest-støy \tilde{p}_{rest}^2 , er SNR gitt som

$$SNR = 10\log_{10}\left(\frac{\tilde{p}_{spes}^2}{\tilde{p}_{rest}^2}\right) \quad (2.1)$$

hvor \tilde{p} er kvadratisk gjennomsnitt av trykket.

Lydfremvekst, e , beskriver fremtiden til en spesifikk lyd i et totalt lydbilde vist som øverste modell i figur 2.1. Det totale lydbildet kan bestå av flere spesifikke lyder, men for å beregne lydfremveksten til en bestemt lyd defineres den totale lyden bestående av den spesifikke lyden og rest-lyd. Lydfremveksten er da lydtrykksnivået som er igjen når man trekker i fra lydtrykknivået til rest-lyd fra det totale lydtrykknivået,

$$e = L_{tot} - L_{rest} = 10\log_{10}(\tilde{p}_{tot}^2) - 10\log_{10}(\tilde{p}_{rest}^2) = 10\log_{10}\left(\frac{\tilde{p}_{tot}^2}{\tilde{p}_{rest}^2}\right). \quad (2.2)$$



Figur 2.1: Sammenheng spesifikk lyd og total lyd og spesifikk lyd og rest-lyd.

Siden $\tilde{p}_{tot} = \tilde{p}_{spes} + \tilde{p}_{rest}$ kan lydfremvekst skrives som en funksjon av SNR som

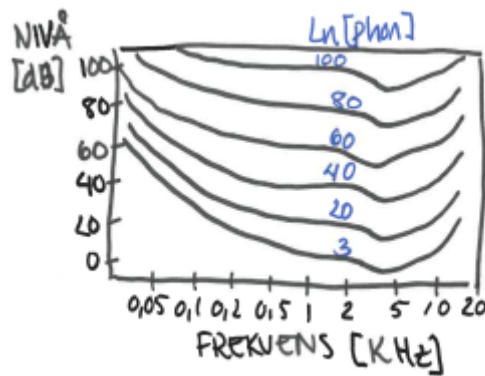
$$e = 10\log_{10}\left(\frac{\tilde{p}_{tot}^2}{\tilde{p}_{rest}^2}\right) = 10\log_{10}\left(\frac{\tilde{p}_{rest}^2 + \tilde{p}_{spes}^2}{\tilde{p}_{rest}^2}\right) = 10\log_{10}\left(1 + 10^{\frac{SNR}{10}}\right). \quad (2.3)$$

2.2 Hørestyrke

Hørestyrke, L_n , (*eng: loudness*) er en enhet som bedømmer hvor sterk en lyd oppleves med utgangspunkt i ørets sensitivitet ved ulike frekvenser. Ved beregning tilpasses lyden en hørestyrkekurve for en rentone og tilegnes en verdi ut i fra avstanden til denne kurven i de ulike frekvensbåndene. Beregningen deles inn i 1/1 eller 1/3 oktavbånd som summeres sammen. Hørestyrkenivå beregnes i enheten phon og tar utgangspunkt i lydtrykknivået til en rentone på 1000 Hz. Det vil si at en tone på 50 dB ved 1000 Hz tilsvarer 50 phon.

Hvor høyt en lyd oppleves avhenger hovedsaklig av lydtrykknivået, men vil også påvirkes av frekvens, spektrum og varighet (9). For korte toner, opp til 100 ms, vil hørestyrken avta med varigheten. Hørestyrken for toner med varighet over 100 ms er omtrent uavhengig av varighet (10).

På grunn av at hørselssystemet er inndelt i kritiske frekvensbånd vil hørestyrken også påvirkes av avstanden mellom tonene i frekvens. Toner som ligger i nærliggende frekvensbånd oppleves som kraftigere enn toner i samme frekvensbånd. Toner som er nære i frekvens vil også kunne maskeres. Dersom avstanden mellom frekvensbåndene blir stor, vil fokuset være på den mest fremtredende tonen og ikke på kombinasjonen av flere toner. Det vil si at hørestyrken tildeles den mest dominerende lyden.



Figur 2.2: Eksempler på hørestyrkekurver i phon. Den nederste linja er terskel for stillhet.

2.3 Maskering

Når man blir utsatt for flere lyder på samme tid kan en spesifikk lyd bli maskert av en mas-kerende lyd som enten inntreffer nære i tid eller frekvens. Maskeringen skjer fordi øret ikke rekker å oppfatte endring av lyd, eller at spesifikk lyd og maskør benytter samme del av et frekvensområde slik at hørselssystemet ikke klarer å oppfatte forskjell.

Maskering i tid kan både skje når en maskør inntreffer kort tid etter spesifikk lyd (for-overmaskering), eller at maskør inntreffer før spesifikk lyd (bakovermaskering) (10). Bakover-maskering er mye mer effektivt enn forovermaskering, og tillater større tidsforskjell mellom spesifikk lyd og maskør.

Både rene og komplekse toner, samt smalbåndet og bredbåndet støy viser alle tendenser til å maskere andre lyder. Lyder som er nære i frekvens maskerer bedre enn ved lengre avstand, samt at toner med høy frekvens maskerer bedre enn lavere frekvenser (9).

2.4 Lyttetester

Formålet med en lyttetest er å komme fram til en objektiv vurdering av auditive inntrykk ved å gjennomføre den samme testen for mange kandidater. Testen gir muligheter for å teste kontrollert variasjon av en stimuli for å avgjøre om resultatene kan lede til konkrete sammenhenger eller om de er tilfeldige variasjoner.

Mange faktorer har innflytelse ved gjennomføringen av en lyttetest og det finnes flere ulike måter å gjennomføre en test på. Lydeksemlene som skal vurderes må være tydelige i sitt budskap og de må være lange nok slik at lytteren rekker å få et reelt inntrykk (11). Måten lytteren skal vurdere lydeksemlene på må være enkel å bruke.

I følge Bech og Zacharow er 20 minutter passe lengde for en lyttetest (11). 30-40 minutter er fortsatt akseptabelt, men dersom tiden strekker seg utover dette bør testen deles inn i mindre deler. Rommet hvor testen gjennomføres bør være innbydende for lytteren.

Testen bør gjennomføres mest mulig likt for alle kandidater. Det innebærer blant annet at kandidatene møter de samme omstendighetene og blir gitt samme informasjon om hvordan testen skal utføres.

2.4.1 Typer lyttetester

Direkte skalering er en av de vanligste måtene å gjennomføre en lyttetest på. Testen gjennomføres ved at lytteren vurderer et og et lydeksempel på en forhåndsbestemt skala. Det er viktig at skaleringen oppleves som meningsfull for lytteren. En utfordring med denne form for lyttetest er å få lytteren til å utnytte hele skalaen (11). Grunnen til dette kan være at lytteren sparer de høyeste verdiene til det kommer et lydeksempel som oppleves mer ekstremt. En annen form for feilkilde er sammentrekningsskjevhet (11). Denne feilen oppstår fordi kandidaten tenderer til å være konservativ, noe som kan gjøre at store forskjeller underestimeres og små forskjeller overestimeres.

Parvis sammenligning er en annen form for lyttetest. Denne benytter indirekte skalering. Lytteren presenteres for to lydeksempler i kort tid etterhverandre og vurderer deretter eksemplene opp mot hverandre. Alle lydeksemlene må sammenlignes og det er derfor en tidkrevende metode. En fordel er at lytteren slipper å knytte inntrykket opp mot et bestemt tall.

2.4.2 Vurdering av ubehag fra støy

ISO 15666 gir spesifikasjoner for sosio-akustiske undersøkelser omkring effekten fra støy deriblant vurdering av ubehag fra støy (eng: *noise annoyance*) (12). Med ubehag fra støy menes både misnøye, plage, irritasjon og forstyrrelse forårsaket av lyden. Standarden skalerer støyplage på en numerisk skala fra 0 til 10, hvor 0 tilsvarer at man ikke opplever noe ubehag i det hele tatt, mens 10 indikerer voldsomt ubehag. For en sosiakustisk studie vil man se tilbake på et lengre tidsrom for å kunne avgjøre hvor plaget man er av en bestemt støykilde.

2.5 Støy fra vindturbiner

En vindturbin består av tårn, maskinhus og vinger og kan ha et lydeffektivå på omkring 105-110 dB. Utstrålt lyd er bredspektret og består både av ikke hørbar infralyd og lavfrekvent og høyfrekvent hørbar lyd. Lyden dannes av både aerodynamisk og mekanisk støy. Lavfrekvent støy bærer langt og støyen kan høres langt fra kilden. Det anbefales derfor å bygge boliger minst 800-1000 meter fra vindturbinen (13). Hvor mye lyd som faktisk stråles ut avhenger av tekniske karakteristikk til vindturbinen samt fysiske forhold som retning og hastighet til vinden, markabsorpsjon, trykk og temperatur (13).

Aerodynamisk støy dannes ved at vingene skjærer gjennom lufta og avhenger av vingespissens hastighet, vingenes form og turbulens. Selve rotorbladet danner en bred hveselyd omkring 1000 Hz, i tillegg til at det også kan genereres impulsive og lavfrekvente lydbølger (2). Når vingene skjærer ned mot bakken oppstår en rytmisk endring som gir en amplitudemodulasjon. Forskjell på trykk, temperatur og vindhastighet mellom vingetuppens øverste og nederste punkt i en rotasjon kan også bidra (13). Lyden karakteriseres gjerne som en *svisjelyd*.

Fra omtrent 3 til 25 m/s produserer vindturbinen kraft, mens for vindhastigheter over dette må den stenges. Ved vindhastighet 8 m/s er vindturbinen på sitt verste støymessig. Ved hastigheter like over dette vil naturlig vindstøy maskere over støyen fra vindturbinen (13).

Maskinstøy dannes fra gir, vifter og generatorer. Rentonestøy kan også oppstå og oppleves ofte som mer forstyrrende enn bredspektret støy. Rentonene kan stamme fra gir, generator og vifter i vindturbinene eller mekaniske lyder ved nedbremsing (13).

2.6 Regelverk

Anbefalte støygrenser skal ivareta opplevelsen ved utsettelse av en støykilde slik at de aller fleste ikke plages av støy. Regelverket for tillatt støygrense for vindturbinstøy er for de fleste land basert på lydtryknivå, men noen land har også inført grenseverdier i form av differensielle støygrenser som lydfremvekst. I en litteraturundersøkelse gjort på utfordringer ved bruk av lydfremvekst som støyindikator ble det funnet at bruk av differensielle støygrenser er dårlig dokumentert og at det har blitt gjort lite forskning på følger av lydfremvekst som støyindikator i et folkehelseperspektiv (1).

I Norge benyttes L_{den} lik 45 dB(A) som anbefalt grenseverdi for bygg til støyfølsomt bruk. L_{den} (den = day-evening-night) er en indikator for gjennomsnittstøy over et helt døgn med 5 dB innskjerping på kveld (kl.19-23.00) og 10 dB for natt (kl.23-07). Grenseverdien er ikke juridisk bindende, men er definert i Retninglinje for behandling av støy i arealplanlegging (T-1442) (14) som en anbefalt støygrense som kommunene kan bruke. I Sverige bruker L_{Aeq} lik 40 dB(A), noe som i praksis blir relativt likt den norske retningslinjen. Svenske retningslinjer anbefaler også en 5 dB innskjerping ved tilfeller av rentonelyd. Det danske regelverket benytter også L_{Aeq} , men har grenseverdier som varierer ut i fra vindhastigheten og har i tillegg en egen grenseverdi for lavfrekvent støy.

Eksempler på andre land i Europa som har benyttet differensielle støygrenser er Frankrike, Italia, Irland og Storbritannia. Støygrensene defineres noe ulikt. Franske regelverk definerer grenseverdier for industri og samfunnsstøy ved lydfremvekst. Regelverket sier at for lydtryknivå høyere enn 45 dB(A) skal ikke støyen overskride bakgrunnsstøyen med mer enn 5 dB(A) på dag og 3 dB(A) på natt (1). Italia har også lenge benyttet grenseverdier tilsvarende lydfremvekst for industristøy deriblant støy fra vindturbiner. Grenseverdiene er satt til 5 dB på dagen og 3 dB på natten, men disse beregnes kun innendørs. Irland anbefaler å bruke en vurdering av lydtryknivåer for å angi maksimalt tillatt bidrag og denne verdien kan igjen avhenge av bakgrunnsstøynivået. For vindturbiner gjelder prinsippet om at støyutslipp skal kontrolleres til absolutte grenser når bakgrunnsstøyen er lav, og i forhold til bakgrunnsstøyen når bakgrunnsstøyen har høyere lydnivå (1).

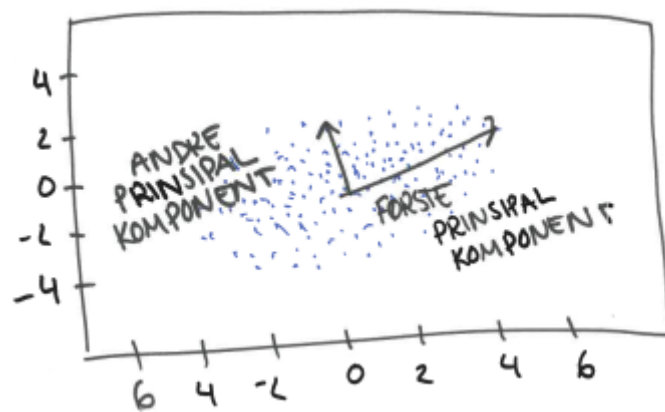
Storbritannia bruker en differensiell støygrense som ligner lydfremvekst, men denne er definert som $L_{tot} - L_{spec}$ hvor sistnevnte er lydnivået til en spesifikk kilden. Dette er definert som rest-støy. Likt for alle eksemplene er at støy skal kontrolleres med en differensiell støyindikator når bakgrunnsstøyen blir høy.

2.7 Hovedkomponentanalyse

Hovedkomponentanalyse, eller *principal component analysis (PCA)*, er en metode som beskriver et datasett i flere dimensjoner ut i fra komponentene som best forklarer variasjonen i datasettet (15). Datasettet konverteres med lineær transformasjon til et sett av ukorrelerte verdier, kalt prinsipalkomponenter, som rangeres etter avtagende varians.

Dersom en n -dimensjonal ellipsoide markeres rundt hele datasettet, vil første prinsipal komponent være den lengste vektoren, og andre prinsipal komponent være den nest lengste som vist i figur 2.3.

Korte akser i ellipsoiden tilsvarer lite variasjon for den bestemte variabelen og derfor kan variasjonen for denne variabelen utelates uten at mye informasjon går tapt. Dette gjør at variabler i mange dimensjoner kan projekteres ned til 1, 2 eller 3 dimensjoner ut i fra prinsipal



Figur 2.3: Første og andre prinsipal komponent markert for et datasett.

komponentene med størst variasjon.

I det nye koordinatsystemet ligger første prinsipalkomponent langs første akse og andre prinsipal komponent langs andre akse. Resten av datapunktene transformeres inn i det nye koordinatsystemet. For en matrise av data må dataene først normaliseres slik at alle variablene får et gjennomsnitt lik 0.

Resultatet kan plottes som en korrelasjonssirkel med første og andre prinsipale komponent langs aksene. Korrelasjonen til de ulike variablene vil da synes som vektorer som peker ut fra origo. Korrelasjonen mellom de ulike variablene blir da synlig ut i fra hvor nærme hverandre vektorene plasserer seg. Vektorer med mindre enn 90 grader i mellom korrelerer. Vektorer som står vinkelrett på hverandre korrelerer ikke, mens vektorer med 180 grader i mellom er negativt korrelerte.

Metode

Lyttetestene baserer seg på valg og behandling lydeksempler, valgt lyttetestmetode, måten lydeksempelene presenteres på og hvordan lytteren kan vurdere eksemplene. I tillegg kan andre fysiske aspekter ha innvirkning på hvordan lyttetesten oppleves. Dette kapitlet beskriver framgangsmåte og gjennomføring av testene.

3.1 Innsamling av lydfiler

Det ble innhentet både lyfiler av vindturbinstøy og ulike kilder for bakgrunnsstøy. Lydfilene ble både gjort med egne opptak og innhentet fra andre kilder.

3.1.1 Vindturbinstøy

Det ble samlet inn to lydfiler av vindturbinstøy gjort av Electricité de France i Frankrike. Opptakene er av en enkelt V90 vindturbin fra Vestas med effekt 3 MW. Vindhastigheten på lydfil 1 ved 80 m høyde ble målt til 9.5 m/s. Opptaket hadde SNR lik 61 dB. Vindhastigheten på lydfil 2 er ukjent. SNR var lik 40 dB.

Opptakene ble utført med mikrofon Brüel & Kjær 2270 i 24 bits på 48 kHz. Mikrofonen ble plassert på en rund plate på gress/jordbakke 125 m fra foten til turbinen. Ingen sekundær vindskjerm ble benyttet. A-vektet ekvivalentnivå ble målt til 50.2 dBA.

3.1.2 Bakgrunnsstøy

Kilder til bakgrunnsstøy ble samlet inn med egne opptak og det ble innhentet opptak gjort i Rotvollbukta. Egne opptak ble utført Bymarka og Ladehammerkaia Trondheim 4 oktober 2019, utstysliste er gjengitt i vedlegg A.1. Det ble gjort opptak av flere ulike kilder blant annet lyden av biler som passerer langs en vei, bekk som renner, fuglelyder, bølger som slår inn mot land, støy fra båt og fabrikk. Opptakene ble gjort med 2-5 meters avstand fra lydkilden og med en varighet fra 30 sekund til 1 minutt. Opptakene fra vei ble gjort omtrent 1 meter fra veien med biler som passerte i begge retninger.

Opptakene i Rotvollbukta ble gjort i skogen 5 m fra Leangenbekken 19.05.18. Det ble benyttet opptaker Sony PCM D100 med integrerte mikrofoner i XY-modus på stativ med vindhette i 16 bits og 44.1 kHz. Opptakene var ikke kalibrert.

3.2 Metode brukt i testene

To ulike testmetoder ble brukt, test 1 benyttet direkte skalering og test 2 benyttet parvis sammenligning. De samme lydfilene ble lagt til grunn, vindturbinstøy V1 og V2, og bakgrunnsstøy B1 og B2. Det ble benyttet ulik lengde på eksemplene, samt at test 2 ikke benyttet alle eksemplene som ble brukt i test 1.

3.2.1 Test 1: Direkte skalering

Direkte skalering ble utført ved at lyttekandidaten hører et og et lydeksempel og rangerer deretter ubehag som følge av lyden på en skala fra 0-10 med en oppløsning på 0,5. Lydeksempelene ble satt til å vare i 30 sekund.

Det ble utviklet en test bestående av 45 ulike lydeksempler som ble avspilt i tilfeldig rekkefølge. Lydeksempelene besto av vindturbinstøy V1 sammen med bakgrunnsstøy B1, vindturbinstøy V2 sammen med bakgrunnsstøy B1 og vindturbinstøy V1 sammen med bakgrunnsstøy B2. Lydklippene ble satt sammen med 5 ulike SNR, (-10, -5, 0, 5, 10), som igjen ble justert til 3 ulike lydtryknivå (30 dBA, 40 dBA og 50 dBA). Full oversikt av lydeksempelene er gjengitt i vedlegg B.1. Den totale lyttetiden ble på 22,5 minutter.

3.2.2 Test 2: Parvis sammenligning

Alle lydeksempler ble sammenlignet én gang i tilfeldig rekkefølge. Hvilket lydeksempel som ble presenteres som nummer 1 og hvilket som presenteres som nummer 2 velges også tilfeldig.

Lengden på hvert lydeksempel ble satt til å vare i 7 sekund og pausen mellom sammenligningen ble satt til å vare i 3 sekund. For å kutte ned antall lydeksempler ble kun vindturbinstøy V1 benyttet. Denne ble satt sammen med bakgrunnsstøy B1 og B2. Dette ble gjort med 3 ulike SNR (-10, 0 og 10) og 3 ulike lydtryknivå (30 dBA, 40 dBA og 50 dBA). Full oversikt over lydeksempelene er gjengitt i vedlegg B.2. Test 2 inneholdt tilsammen 18 lydklipp, noe som gir 153 sammenligninger og en aktiv lyttetid på nesten 43,5 minutter.

3.3 Bearbeiding av lydfiler

Bearbeiding av lydfiler til ønsket SNR og lydtryknivå ble gjort i programvaren Matlab_R2017b. Koden er gjengitt i vedlegg C. Lydfilene klippes til riktig lengde før de A-vektes. Videre justeres lydtryknivået til vindturbinstøyen slik at signal-støyforholdet blir riktig før lydfilene settes sammen til ett lydklipp og skaleres til riktig lydnivå.

Opptakene ble ikke korrigert for luftlydabsorpsjon, men ble filtrert for å veie opp for impulsresponsen til hodetelefonene.

Hørestyrke for lydeksempelene ble beregnet med et program som HEAD acoustics har utviklet i tråd med ISO 532 (16).

3.4 Brukergrensesnitt

Brukergrensesnittet for test 1 og test 2 ble implementert i Python 3.8.0 og er basert på et grensesnitt utviklet av Jean Fosset. Fullstendige skript er gjengitt i vedlegg G.1 og G.2. Bilder av brukergrensesnittene er gjengitt i vedlegg H.

Første og siste del av programmet var likt for begge tester. Første del er utfylling av generell informasjon: subjektnummer, alder, kjønn, yrke og om man har tidligere erfaring med vindturbinestøy. Siste del av programmet er avsluttende spørsmål hvor kandidaten svarer på om det var forstyrrelser underveis i testen, og eventuelt hva denne forstyrrelsen var.

Brukergrensesnittet for hoveddelen er ulike. Begge har en fremdriftslinje som viser hvor lang tid det er igjen på hvert enkelt lydeksempel, samt hvor mange lydeksempler eller sammenligninger som er gjenstående.

Test 1

Hoveddelen på test 1 tar inn 45 lydeksempler og plasserer disse i en tilfeldig rekkefølge. Kandidaten trykker på “play” for å spille av lydeksempelen og bestemmer selv når neste eksempel skal startes. Hvert lydeksempel rangeres etter avspilling på en skala fra 0-10 ved å skyve på en glidebryter. Bryteren kan plasseres med en oppløsning på 0,5.

Brukergrensesnittet lagrer alle rangerte verdier direkte til hvert lydeksempel og sorterer disse til slutt over til en liste som skrives til en excel-fil i csv-format.

Test 2

18 lydeksempler hentes inn i hoveddelen i test 2. Som i test 1 lages det en liste i tilfeldig rekkefølge av lydeksempelene. Deretter dannes en liste som lager alle mulige kombinasjoner mellom to tall fra 1 til 18. Dette brukes for å hente ut lydeksempler fra lista som er i tilfeldig rekkefølge. Listen med kombinasjoner av to tall er bygd opp som 1-2, 1-3, 1-4...1-18 deretter 2-3, 2-4...2-18. Det vil si at for de 17 første sammenligningene blir det samme lydeksempelen sammenlignet 17 ganger, for de neste 16 blir samme eksempel sammenlignet 16 ganger og så videre.

Rekkefølgen på lydeksempelene i en sammenligning gjøres tilfeldig ved at det “kastes en mynt” som gir ut 0 eller 1. Ved “0” beholdes rekkefølgen, ved “1” byttes rekkefølgen om.

Etter at begge lydeksempelene er avspilt velger kandidaten om lydeksempel A eller B gir mest ubehag. Det er også mulig å velge at lydeksempelene stiller likt.

Dersom et lydeksempel velges som mest ubehagelig får dette eksempelet 1 poeng. Dette gjøres ved bruk av Pythons datastruktur *dictionary*. Denne knytter verdi direkte opp til navnet på lydklippet. Antall poeng summeres opp underveis som programmet kjøres.

Lista med poeng for alle lytteeksempler lagres tilslutt i en excel-fil i csv-format.

3.5 Oppsett og gjennomføring av lyttetester

Lyttetestene ble gjennomført i Akustikkhallen på Gløshaugen i Trondheim mellom 1. og 21. november 2019. Ved rekruttering av lyttekandidater ble det påkrevd at kandidatene måtte ha normal hørsel for å kunne delta på testen.

Før lyttetesten startet ble kandidatene instruert til å lese gjennom en skriftlig introduksjon. Disse er gjengitt i vedlegg F.1 og F.2. Etter gjennomlesing var det mulighet for å stille oppfølgingsspørsmål før teststart. Kandidatene ble instruert til å tenke at lydene de skulle høre var lyder som høres hele tiden når man er utenfor egen bolig eller når vinduet står åpent. Det samme ble også presisert i brukergrensesnittet, samt at det også sto en beskrivelse av hvordan lydeksempelene skulle vurderes. Både introduksjonsarket og teksten på grensesnittet var på engelsk.

Kandidatene for lyttetest 1 brukte mellom 30-35 minutter, mens lyttetest 2 tok mellom 50-60 minutter. Kandidatene fikk tilbud om godteri for å holde motivasjonen oppe gjennom testen.

3.5.1 Oppsett

Et tidligere benyttet målerom ble benyttet som lytterom. Rommet har i lengre tid blitt brukt som lager og har derfor preg av dette.



Figur 3.1: Bilde av lytterom. Kandidaten satt ved pulten i midten av rommet.

Rommet er lyd og vibrasjonsisolert, men har ikke ventilasjon og det måtte derfor luftes mellom testene. Det ble oppdaget en tonalitet i lampene i rommet, og lampene ble derfor byttet ut med en midlertidig arbeidslampe.

Dørene er doble og tunge å åpne, og den ytterste av de to dørene ble derfor holdt åpen. Lyttekandidatene ble vist hvordan døra kunne åpnes fra innsiden.

Utstyr benyttet i lyttetesten er gjengitt i vedlegg A.2. Bakgrunnsstøyen i rommet ble kontrollert før testene ble satt i gang. Under testene ble det gjort referansemålinger en gang i sekundet med en lydnivåmåler liggende i rommet.

3.5.2 Testgjennomgang

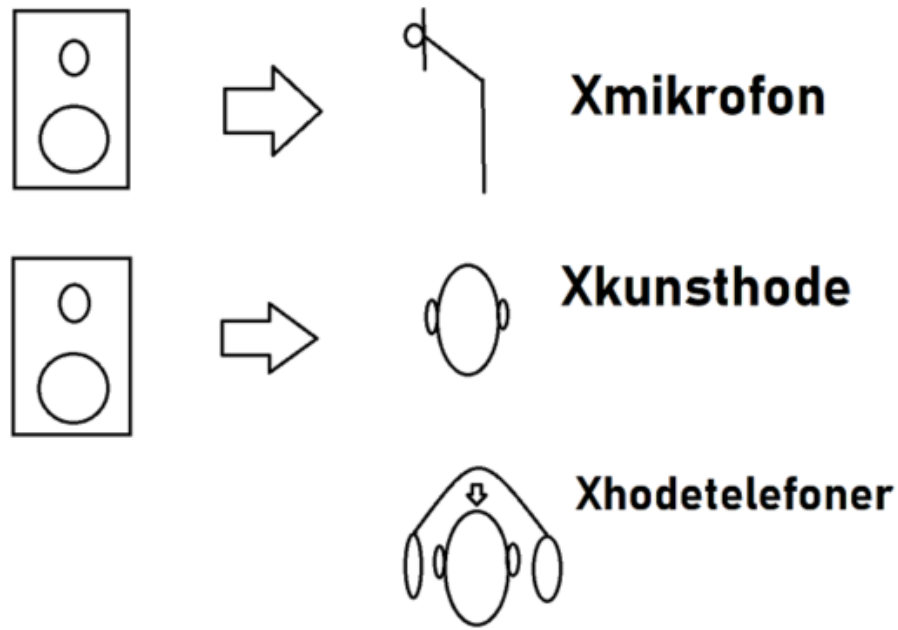
I forkant av lyttetestene ble det kjørt en gjennomkjøring av testen. Testen ble kjørt som om det var en vanlig test, og kandidatene ble bedt om å gi tilbakemelding om det var noe det burde endres på i etterkant. Både lyttetest 1 og 2 fikk tilbakemelding på at det var mange lytteeksempler å gå igjennom, men det ble besluttet å beholde antall eksempler.

3.5.3 Kalibrering av hodetelefoner

Kalibreringen av hodetelefonene ble gjort i to omganger, først kalibrering av frekvensrespons og deretter kalibrering av lydtrykknivå. Databehandlingen ble gjort i EASERA og Matlab_R2017b. Det ble benyttet et kunsthode som ble kalibrert ved å sammenligne utsignalet med utsignalet til en kalibrert kondensatormikrofon. Det ble antatt at kondensatormikrofonen hadde flat frekvensrespons.

Frekvensrespons

For å finne frekvensresponsen ble det gjort opptak av rosa støy fra høyttaler med mikrofon, $X_{mikrofon}(f)$ og kunsthodet, $X_{kunsthodet}(f)$. Opptakene ble utført i et klangrom. Videre ble det



Figur 3.2: Oversikt over opptak. Xmikrofon er fra høytaler til mikrofon, Xkunsthode fra høytaler til kunsthode, Xhodetelefoner fra hodetelefoner til kunsthode.

sendt rosastøy fra høretelefonene direkte inn på kunsthodet, $X_{hodetelefon}(f)$. Frekvensresponen til kunsthodet er da gitt som

$$H_{kunsthode}(f) = \frac{X_{kunsthode}(f)}{X_{mikrofon}(f)} \quad (3.1)$$

og frekvensresponen til hodetelefoner på kunsthodet er

$$H_{hodetelefonkunsthode}(f) = \frac{X_{hodetelefon}(f)}{X_{kunsthode}(f)}. \quad (3.2)$$

For å finne responen til hodetelefonene, og eliminere responen til kunsthodet, må signalet tatt opp på kunsthodet deles på $H(f)_{kunsthode}$ og $H(f)_{hodetelefonkunsthode}$ noe som gir at

$$H_{hodetelefoner}(f) = \frac{X_{kunsthode}(f)}{X_{hodetelefoner}(f)/X_{mikrofon}(f)}. \quad (3.3)$$

For å få ut frekvenskomponenter til responen til signalene ble behandlingsverktøyet EASE-RA benyttet. Det ble hentet ut frekvenskomponenter i 1/3 oktavbånd og 1/6 oktavbånd som ble brukt for å designe et firfilter i matlab. For verdier mellom 100-1000 Hz var det lite variasjon i responen og 1/3 oktavbåndsverdier ble benyttet. For verdier mellom 1000 og 20000 Hz ble det benyttet 1/6 oktavbåndsverdier for å bedre ivareta variasjonene. Verdier under 100 Hz ble ikke vurdert som nødvendige. Koden for design av filteret er gjengitt i vedlegg D.

Lydtrykknivå

Dataprosessering for kalibrering av lydtrykknivå er gjengitt i vedlegg E. Skaleringen skjer i flere omganger. Først beregnes skaleringsfaktor, $s_{mikrofon}$, som gir kalibrert mikrofonverdi, ved å sammenligne lydtrykksnivået av et kalibratoropptak og oppgitt verdi på kalibratoren. Deretter skaleres opptak av rosastøy med mikrofon med denne faktoren. Forholdet mellom rms-verdi til rosastøyopptak med mikrofon og kunsthodet brukes til å finne en ny skaleringsfaktor, $s_{kunsthode}$, som gjør at signal tatt opp med kunsthodet kan skaleres til kalibrert nivå.

Hver gang datamaskinen som ble brukt i testen ble startet på nytt ble det gjort kontroll av lydnivå. Dette ble gjort ved at et av lydeksempelene ble sendt fra hodetelefonene til kunsthodet og skalert med $s_{kunsthode}$.

Kapittel 4

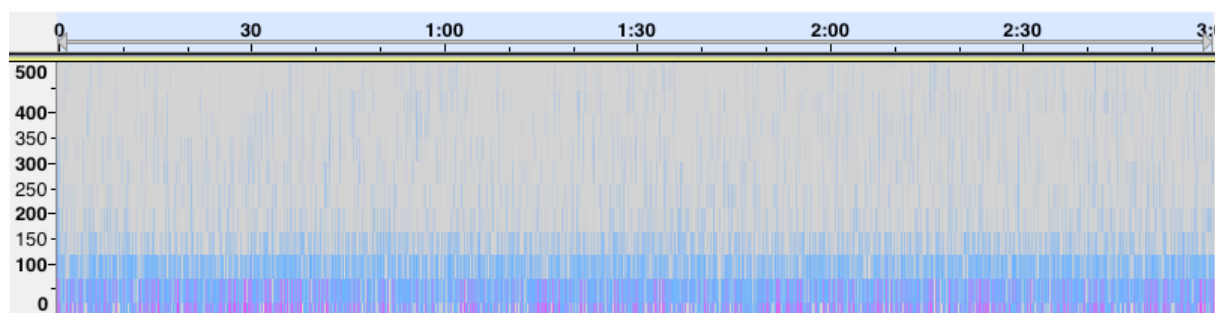
Resultat

Her følger beskrivelse resulterende faktorer som har påvirkning på lyttetesten, samt endelig resultat av lyttetestene. Resulterende faktorer innebærer utvalgte lydfiler, responskurve for kalibrering av hodetelefoner, fordeling av deltagende lyttekandidater og bakgrunnsstøy. Testresultat er fordelt som resultat for test 1 og for test 2.

4.1 Lydeksempler

4.1.1 Vindturbinstøy

Som utgangspunkt for vindturbinstøy ble det besluttet at det skulle benyttes lydklipp med aerodynamisk støy, fritt for mekanisk støy og tonalitet. Det ble valgt ut to klipp, V1 og V2, på 30 sekunder hvor den aerodynamiske støyen var nokså jevn. V1 var fra lydfilen med SNR lik 61 dB, V2 fra lydfil med SNR lik 40 dB.

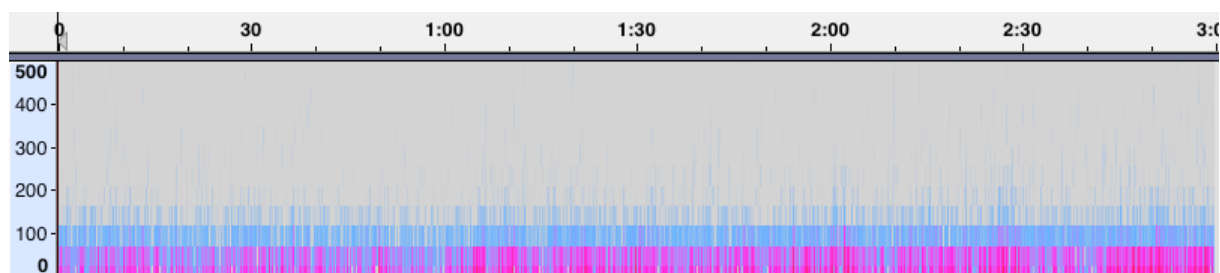


Figur 4.1: Vindturbinstøy V1 gitt som frekvens [Hz] som funksjon av tid. Sterkere farge indikerer høyere lydtrykksnivå.

Figur 4.1 og figur 4.2 viser frekvensspekteret til lydeksempelene fremstilt med programvaren Audacity. V1 inneholder høyere frekvenser mens V2 har høyere lydnivå enn V1 for de laveste frekvensene. V1 oppleves til å ha noe “skarpere” uttrykk mens V2 har et mer “romlende” preg.

4.1.2 Bakgrunnsstøy

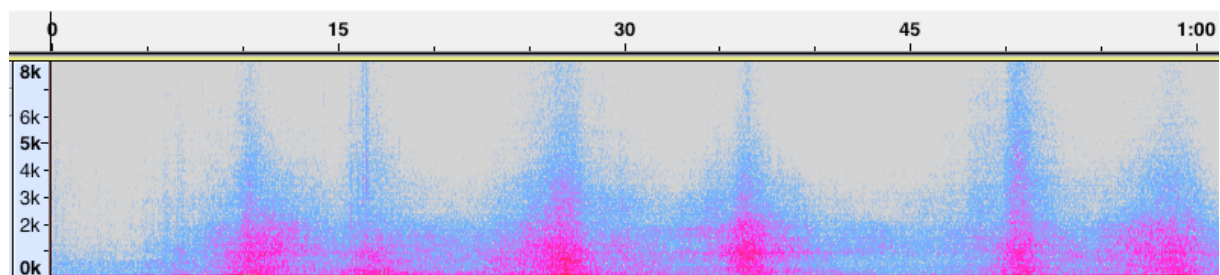
Som bakgrunnsstøy ble det besluttet at en av lydfilene skulle bestå av mekanisk støy, B1, og en av naturlig støy, B2. Som mekanisk støy ble det brukt et opptak av veitrafikkstøy. Biler passerer i



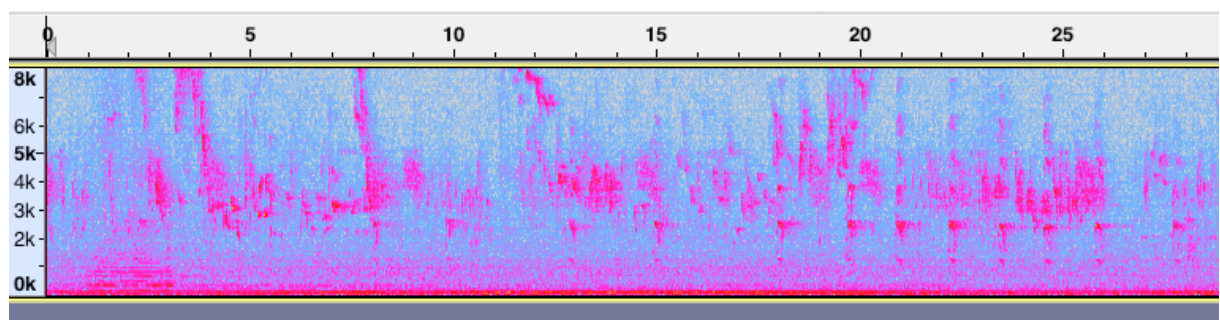
Figur 4.2: Vindturbinestøy V2 gitt som frekvens [Hz] som funksjon av tid. Sterkere farge indikerer høyere lydtrykksnivå.

begge retninger langs en vei i omtrent 50 km/t. Det er noen sekunder mellom at hver bil passerer og intensiteten på lydklippet varierer som vist i spektrogrammet i figur 4.3.

Som naturlig støy ble det brukt opptak av fuglelyder. Ingen av opptakene gjort i Byåsen var egnet da det var for lite fuglelyder, det ble derfor besluttet å bruke opptaket fra Rotvollbukta som består av lyder både av ulike fugler, fuglekvisper samt måkeskrik, og lyden av en bie som flyr forbi. Spektrogram for B2 er vist i figur 4.4.



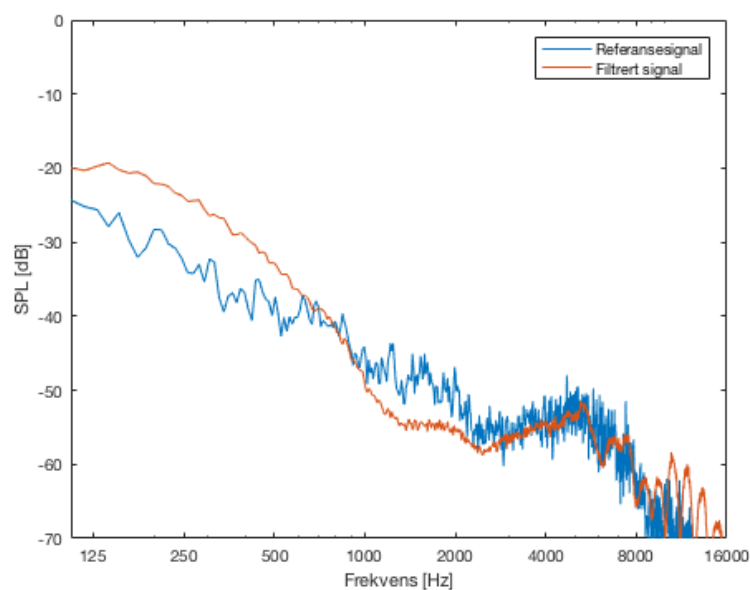
Figur 4.3: Bakgrunnsstøy B1 gitt som frekvens [Hz] som funksjon av tid. Sterkere farge indikerer høyere lydtrykksnivå.



Figur 4.4: Bakgrunnsstøy B2 gitt som frekvens [Hz] som funksjon av tid. Sterkere farge indikerer høyere lydtrykksnivå.

4.2 Kalibrering av hodetelefoner

Egenskapene til responsfilteret kan vurderes ved å sammenligne opptak gjort med kunsthode. Figur 4.5 sammenligner opptak gjort av rosa støy av kunsthode sendt fra høyttaler og rosa støy sendt via hodetelefoner til kunsthode og som deretter er filtrert med filterfunksjon matlab.



Figur 4.5: Sammenligning av referansesignal og filtrert signal sendt fra hodetelefoner.

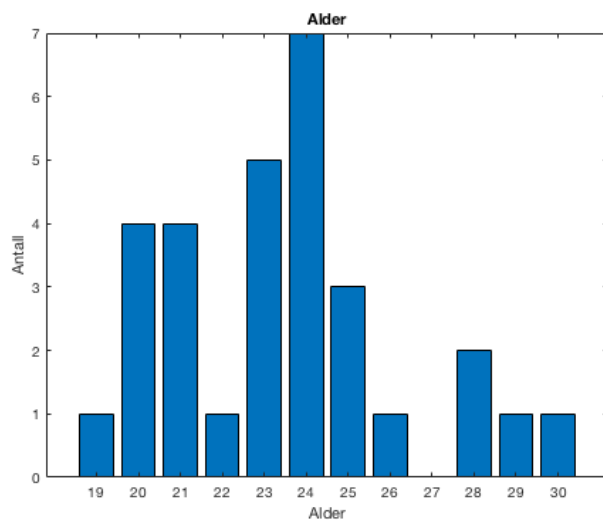
Resultatet viser noen avvik mellom referansesignal og filtrert signal. Opp til 750 Hz ligger filtrert signal omtrent 5 dB over, og omtrent 5 dB fra 1000 til 2000 Hz.

4.3 Kandidater

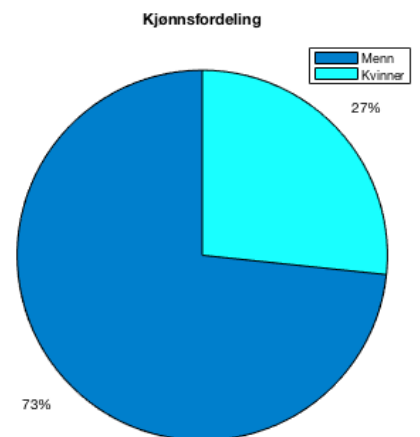
30 kandidater ble rekruttert til test 1, og 31 til test 2. Kandidatene var i hovedsak studenter.

4.3.1 Fordeling av kandidater test 1

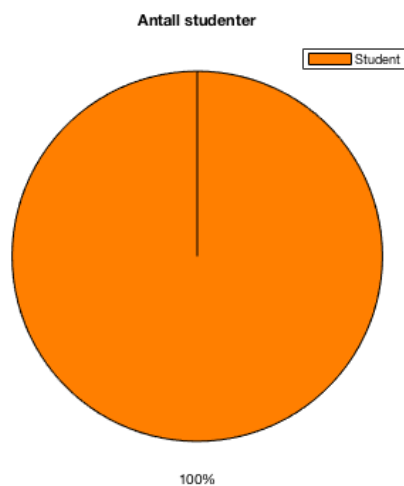
Aldersfordelingen for kandidatene som tok test 1 var fra 19 til 30 år. Andelen menn var større enn kvinner, og alle kandidatene var studenter. 3 av kandidatene oppga at de hadde tidligere erfaring med vindturbinestøy.



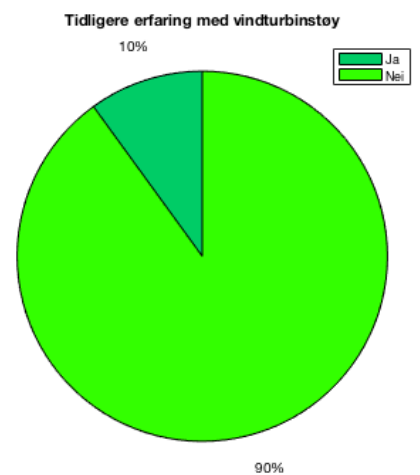
((a)) Aldersfordeling test 1.



((b)) Kjønnfordeling test 1.



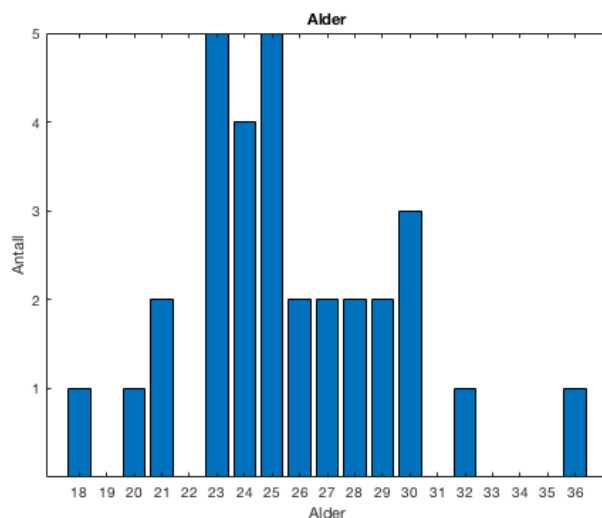
((c)) Andel studenter test 1.



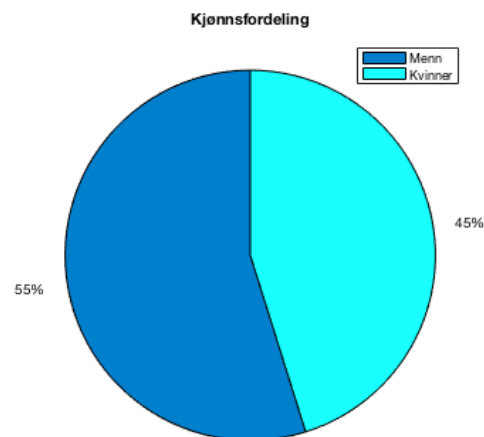
((d)) Andel erfaring med vindturbinestøy test 1.

4.3.2 Fordeling av kandidater test 2

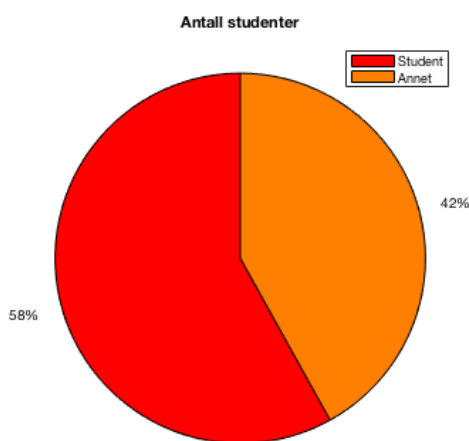
Aldersspennet for kandidatene som tok test 2 var fra 18 til 36 år, og det var nesten like mange kvinner som menn. 42% av kandidatene var ikke studenter og 3 av kandidatene oppga at de hadde hatt tidligere erfaring med vindturbinestøy.



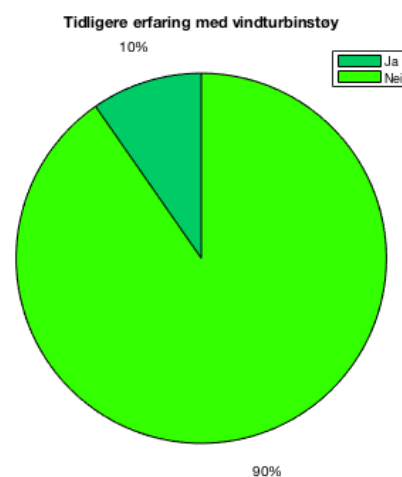
((a)) Aldersfordeling test 2.



((b)) Kjønnsfordeling test 2.



((c)) Andel studenter test 2.



((d)) Andel erfaring med vindturbinestøy test 2.

4.4 Bakgrunnsstøy og forstyrrelser

En kontroll av rommet ble gjort i forkant av testene og viste et lydtryknivå på 18 dB(A) for en måling gjort på 10 sekunder. Det gir en differanse på mer enn 10 dB opp til laveste lydtryknivå i lyttetesten (30 dB).

4.4.1 Måling av bakgrunnsstøy

Bakgrunnsstøy ble målt for 29 av 30 kandidater som tok test 1 og 29 av 31 kandidater som tok test 2. Middelerdi og 95 % konfidensintervall for alle kandidatene er gjengitt i vedlegg I.1 og I.2. Samlet middelerdi, standardavvik og konfidensintervall for de to testene er gjengitt i tabell 4.1.

Test	μ	σ	95% konfidensintervall
1	22,38	5,48	[22,33, 22,43]
2	23,24	6,51	[23,29, 23,38]

Tabell 4.1: Samlet bakgrunnsstøy for test 1 og test 2. Verdiene er gitt i dB(A).

4.4.2 Forstyrrelser under testen

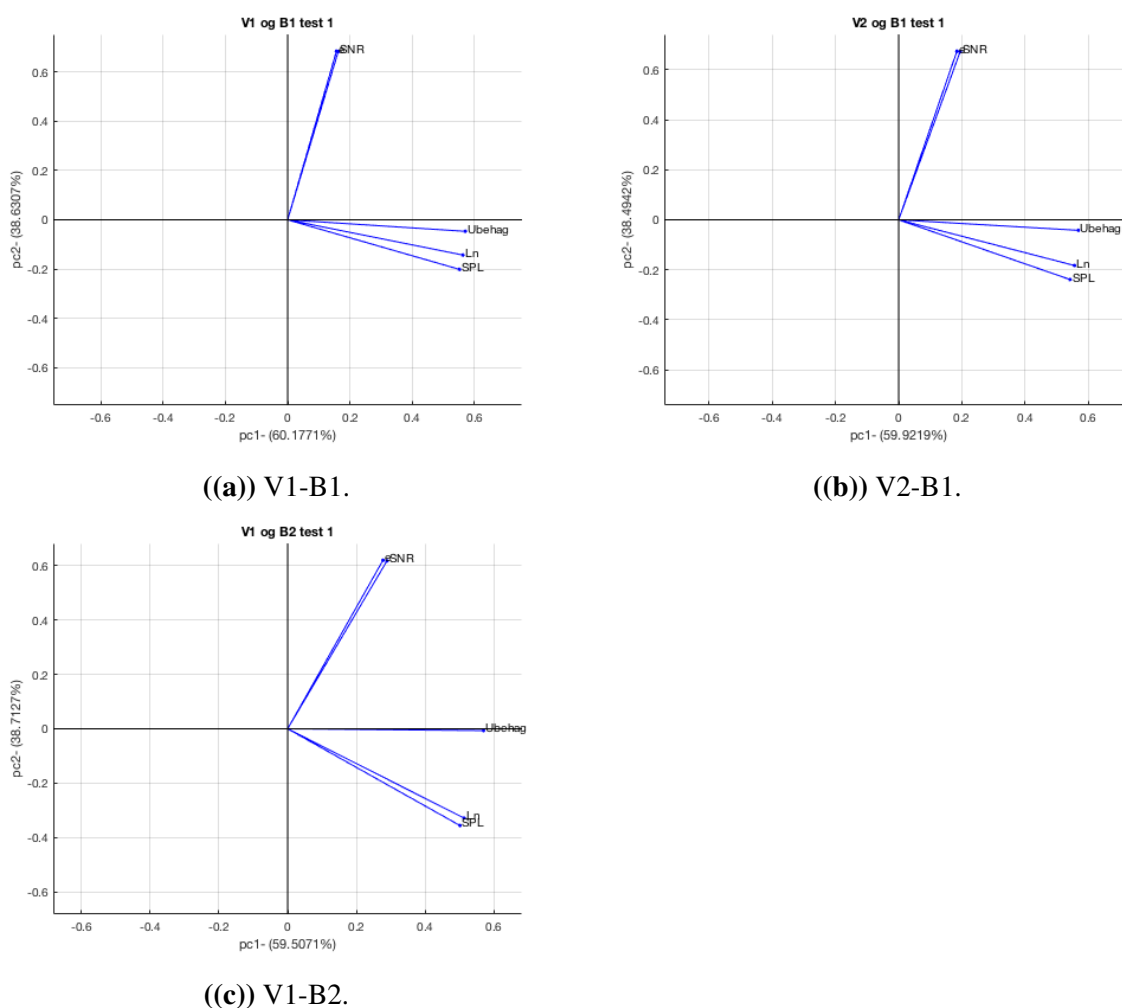
Kun 8 av kandidatene ga tilbakemelding om at de ble forstyrret underveis i testen. 2 av kandidatene mente at de ble forstyrret av omgivelsene, mens 3 andre poengterte at de synes det var vanskelig å konsentrere seg over tid og at det var frustrerende å høre de samme lydklippene mange ganger. De 3 siste nevnte mindre forstyrrelser fra lyder utenfra.

4.5 Lyttetester

Fullstendig tabell med gjennomsnittsverdi, maksimumsverdi, minimumsverdi og median for test 1 og test 2 er gjengitt i vedlegg J.1 og J.2. På grunn av en feil med implementeringen av brukergrensesnittet benyttet i test 2 ble dessverre ikke verdiene for lydeksempel 18 lagret og disse mangler derfor i resultatlista.

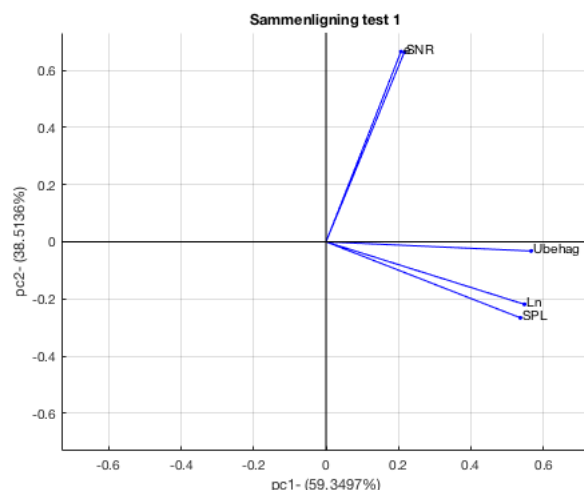
4.5.1 Resultat test 1

Biplot for sammenligning V1 - B1, V2 - B1 og V1 - B2 er vist i figur 4.8.



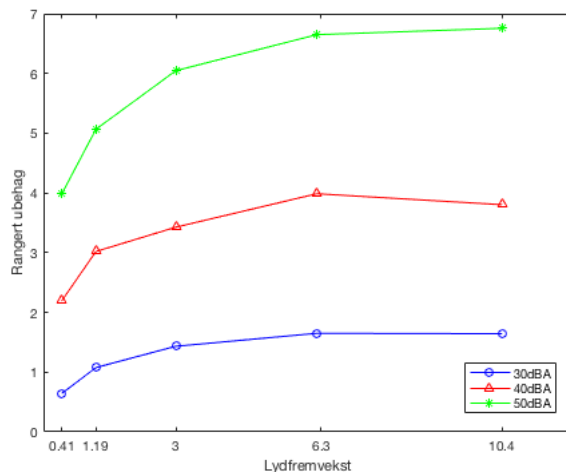
Figur 4.8: PCA sammenligning mellom SNR, lydfremvekst e , lydtrykknivå SPL , hørestyrke Ln og ubehag for ulike sammensetninger av lydfiler. Vektorene for SNR og e ligger over hverandre.

PC1 og PC2 utgjør over 98%. Det samme gjelder for sammenligningen av alle lytteeksempelene vist i figur 4.9.



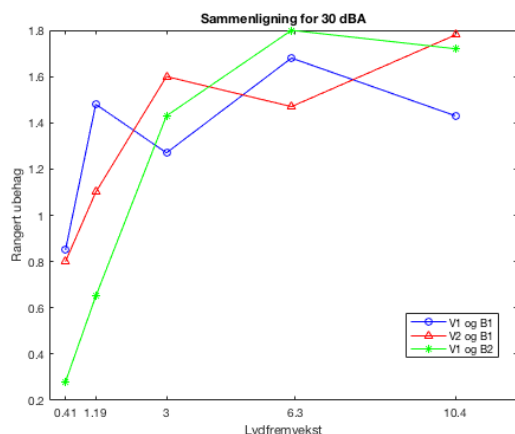
Figur 4.9: PCA sammenligning mellom SNR, lydfremvekst e , lydtrykknivå SPL , hørestyrke Ln og ubehag for alle lydeksempler test 1. Vektorene for SNR og e ligger over hverandre.

Sammenhengen mellom lydtrykknivå, lydfremvekst og rangert ubehag kan studeres nærmere i figur 4.10. Punktene tar utgangspunkt i middelverdien. Det kommer da tydelig fram at rangert ubehag avhenger av lydtrykknivået. Det er også interessant å se at rangert ubehag avhenger mer av lydfremveksten for det høyeste lydtrykksnivået, mens det er lite variasjon for det laveste lydtrykksnivået.

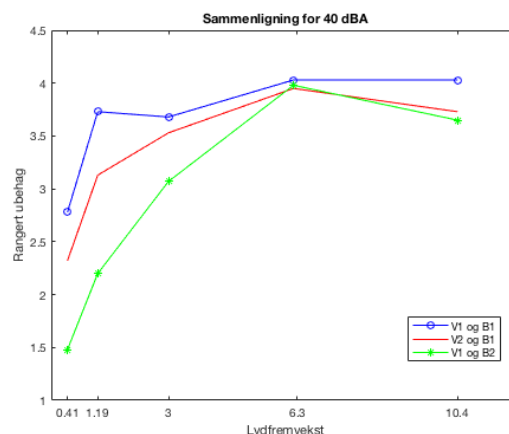


Figur 4.10: Sammenligning lydfremvekst, lydtrykknivå og rangert ubehag.

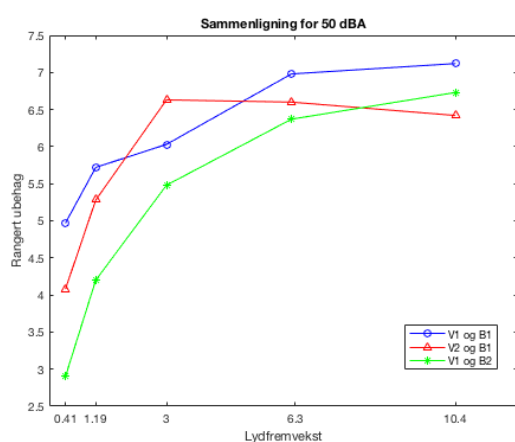
Figur 4.11(a), 4.11(b) og 4.11(c) sammenligner de ulike lydeksempelene for de ulike lydnivåene. For 30 dB(A) er det ganske jevnt mellom de ulike lydeksempelene, mens for 40 dB(A) og 50 dB(A) rangeres lydeksempelene med fuglesang, V1 og B2, som litt mindre ubehagelig enn de to andre.



((a)) 30 dB(A).



((b)) 40 dB(A).



((c)) 50 dB(A).

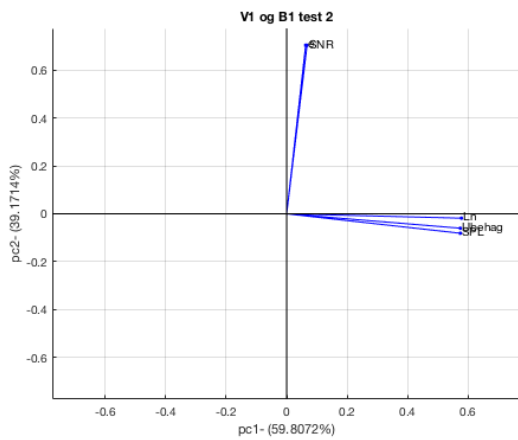
Figur 4.11: Sammenligning av rangert ubehag for ulike lydfremvekster på ulike lydnivå.

4.5.2 Resultat test 2

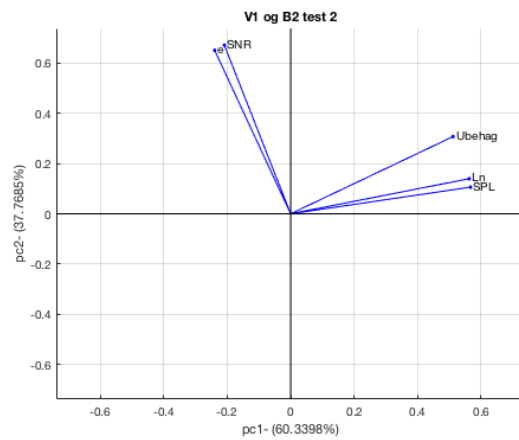
Figur 4.8 viser PCA framstilling for de to lydeksemlene benyttet i test 2. De to første prinsippkomponentene dekker tilsammen over 97% av variasjonen.

PCA-sammenligning av alle lydeksemlene i test to er gitt i figur 4.13 og viser den samme tendensen som i figur 4.12.

Figur 4.14 sammenligner rangert ubehag som funksjon av lydfremvekst for ulike lydtryknivå. Sammenlignet ubehag for de ulike lydeksemlene ved ulike lydtryknivå er gitt i figur 4.11(a), 4.11(b) og 4.11(c). Figur 4.11(c) har bare verdier for de to laveste lydfremvekstene for V1 og B2.

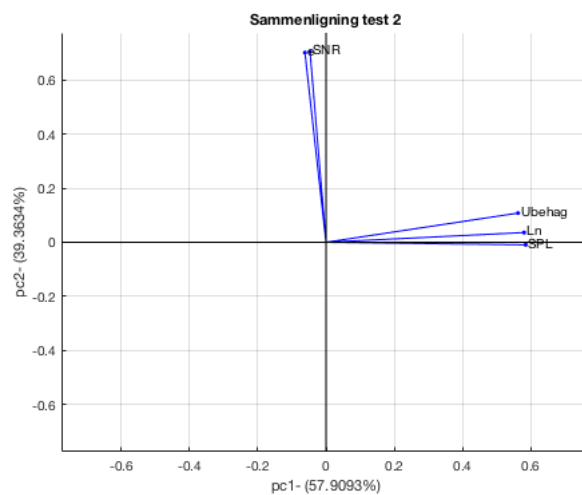


((a)) V1 og B1

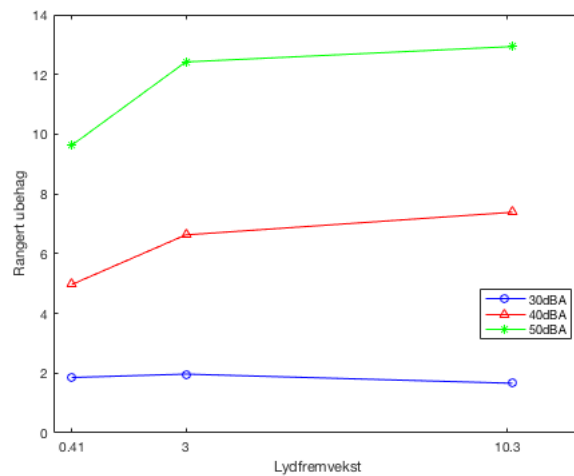


((b)) V1 og B2

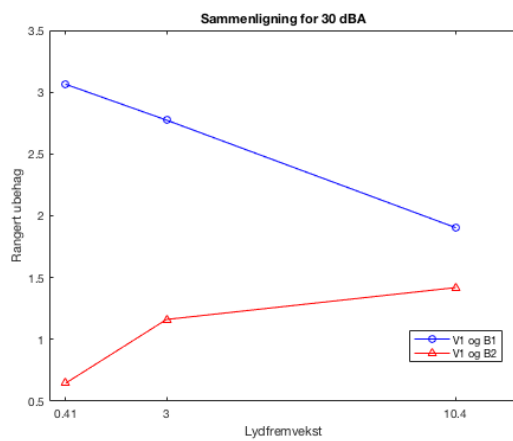
Figur 4.12: PCA sammenligning mellom SNR, lydfremvekst e , lydtrykknivå SPL , hørestyrke Ln og ubehag for ulike sammensetninger av lydfiler. Vektorene for SNR og e ligger over hverandre.



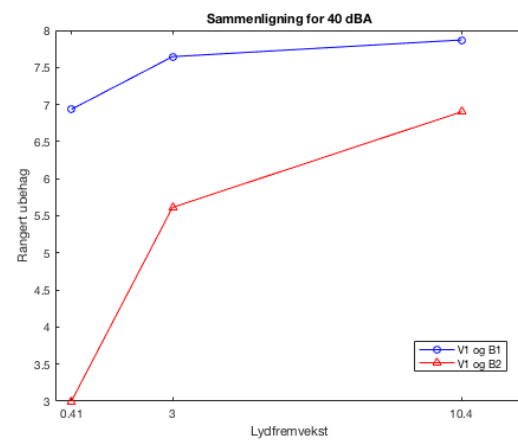
Figur 4.13: PCA sammenligning mellom SNR, lydfremvekst e , lydtrykknivå SPL , hørestyrke Ln og ubehag for alle lydeksempler i test 2.



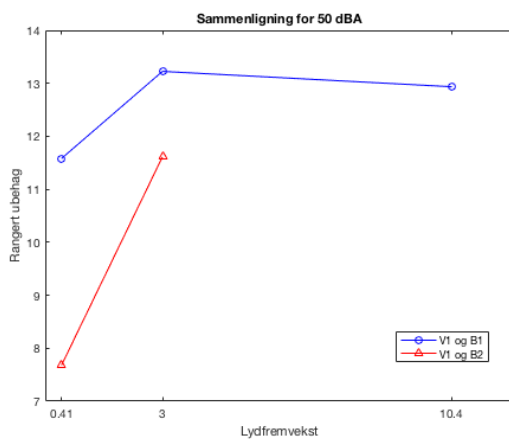
Figur 4.14: Sammenligning ubehag som funksjon av lydfremvekst for ulike lydtryknivå



((a)) 30 dB(A).



((b)) 40 dB(A).



((c)) 50 dB(A).

Figur 4.15: Sammenligning rangert ubehag for ulike lydfremvekster på ulike lydnivå.

Diskusjon og vurderinger

For å kunne vurdere troverdigheten resultatene til lyttetesten er det essensielt og vurdere gjennomføringen av lyttetestene. Støy fra vindturbinestøy har tidligere blitt vurdert både gjennom bruk av lyttetester, men også bruk av spørreundersøkelse til mennesker som bor i nærheten av en vindturbin. Fordelen med å benytte lyttetest er at det er selve støyen som skal vurderes, og hvilke tanker lytteren har omkring bruk av vindturbiner generelt kan forventes å ha mindre påvirkning.

5.1 Vurdering av gjennomføring av lyttetester

Lytte kandidatene møtte de samme omstendighetene når de kom for å ta testen. Informasjonsarket sikret at alle kandidatene ble gitt samme informasjon.

5.1.1 Lokale

Lokalet benyttet i testene var teknisk sett godt egnet da det var lyd- og vibrasjonsisolert, men det var lite innbydende for kandidaten. Det hadde også dårlig ventilasjon og den tunge døra kunne gi en klaustrofobisk følelse. Gode fysiske forhold er en viktig faktor da den kan ha påvirkning på sinnstilstanden til lytteren. Dette kan ha hatt spesielt innflytelse på test 1.

Det var få tilbakemeldinger på at kandidatene ble forstyrret underveis. Gjennomsnittlig bakgrunnsstøy var på 22,38 dBA for test 1 og 23,24 dB(A) for test 2. Med tanke på at de laveste lydeksempelene som ble presentert var på 30 dB(A), og at testen ble utført med hodetelefoner er dette et akseptabelt resultat. På grunn av at test 2 tok lenger tid enn test 1 er det ikke et overraskende at resultatet blir noe høyere, da støyen kan komme av at lytteren flytter på seg og lignende.

5.1.2 Lydeksempler

Lydklippene av vindturbinestøy ga et inntrykk av aerodynamisk støy fra vindturbinene. Klippene representerte ikke mekanisk støy eller tonalitet, noe som kan oppleves som de mest forstyrrende elementene fra en vindturbin. Samtidig er dette faktorer som kommer på grunn av tekniske feil med vindturbinen og er derfor støy som kan fjernes, mens den aerodynamiske støyen er kontinuerlig.

En av utfordringene med aerodynamisk støy fra vindturbiner er at den er ujevn og uforutsigbar. Dette kan virke mer forstyrrende enn jevnt varierende støy. Lydklippene brukt i disse testene er ganske jevne og får kanskje ikke gjenskapt uforutsigbarheten godt nok. Lengden til lydklippene vil også være av betydning for dette.

Lydklippene ble satt til å vare 30 sekunder for test 1, og 7 sekunder for test 2. Lengden ble satt for å kunne gi en opplevelse av at støyen pågår over en lengre periode og er kontinuerlig. Det kan virke som at 30 sekunder er tilstrekkelig for å gi dette inntrykket, mens dette aspektet faller litt bort for test 2 hvor hensikten med testen er at eksemplene skal sammenlignes kort tid etter hverandre.

Tanken bak bakgrunnskildene var at de skulle representere bakgrunnslyd som kan oppleves sammen med vindturbinstøy. Det er verdt å påpeke at bakgrunnskildene kan virke forstyrrende i seg selv. Noen av kandidatene påpekte også at lydene virker mer irriterende når de blir gjentatt mange ganger. Eksempelvis hadde bakgrunnsstøyen med fuglelyder i test 1 måkeskrik som kan virke forstyrrende i lengden.

Opptakene er tatt nærmere støykildene og er ikke korrigert for luftlydsabsorpsjon. I en reell situasjon vil støyen erfares i en avstand på 800-1000 meter, og dette kan på påvirke lyden noe da luftlydsabsorpsjon demper noe av de høyeste frekvensene.

Testen ble utført med hodetelefoner og lydfilene ble filtrert for å motvirke frekvensresponsen til disse. Sammenligningen av referansesignal og filtrert signal i figur 4.5 viser avvik mellom signalene. Avviket opp til 1000 Hz er av spesiell betydning da vindturbinstøyen ligger innenfor dette frekvensområdet og i følge modellen forsterkes da signalet feilaktig. Det er tydelig at filterfunksjonen benyttet ikke er nøyaktig nok. Grunnen til dette kan være at opptakene ikke er utført presist nok. Opptakene kunne med fordel ha vært utført i et ekkofritt rom. Kunsthodet som ble benyttet er heller ikke av nyeste sort. En annen feilkilde kan være at det er benyttet for få eller feilaktige filterkomponenter.

5.1.3 Utvalg kandidater

Problematikken omkring støy fra vindturbiner kan være et betent tema for enkelte og kan påvirke innstillingen man har til å utføre en lyttetest. De fleste svarte at de ikke hadde tidligere erfaring i denne sammenhengen og det var heller ikke satt spesielle krav til om lytteren hadde hatt tidligere erfaring med vindturbiner. Fokuset var å se på om hvordan lydeksempelene ble rangert forskjellig og ikke hvor forstyrrende vindturbinstøy i seg selv oppleves, men det kan ha påvirkning på hvilken del av skalaen som benyttes i test 1. Test 2 vil ikke la seg påvirke på samme måte.

De aller fleste av kandidatene var studenter og aldersspennet var mellom 18 til 36 år. Kjønnfordelingen var overvekt av menn på begge tester, det har ikke blitt sett på om menn og kvinner vurderte lydeksempelene ulikt.

5.1.4 Utførelse

Gjennomføring av lyttetest med brukergrensesnitt gjør at lytteren kan gjennomføre testen uten andre personer i rommet, samt at lytteren selv kan bestemme tempo.

Test 1

Tilbakemeldinger gitt av lyttekandidater etter endt test 1 var blant annet at det opplevdes som irriterende å høre på støy fra bil og fugler i lengden. Noen av kandidatene mente også at lengden på klippene kunne kuttes ned utover testen, fordi man ikke behøver like lang tid for å evaluere klippene. Noen poengterte også at det kan være forskjell på vurderingen de lydeksempelene man hører på i starten og slutten av testen. For eksempel kan klippene som presenteres mot slutten tolkes mer negativt fordi man er sliten. Sammentrekningskjevhet nevnes også som en kilde til feil. Dette kan påvirkes av rekkefølgen lydeksempelene presenteres i.

Ved å se på omfanget til resultatene er det tydelig at kandidatene har brukt skalaen ulikt. Noen har brukt alle verdier fra 0 til 10 mens noen har et omfang på mindre enn 4. Dette kan enten bety kandidaten syns at alle lydklippene befinner seg i en del av skalaen, for eksempel at lytteeksempelene ikke oppleves som spesielt plagsomme, eller det kan bety at man ikke har fått utnyttet hele skalaen fordi man ventet med ekstremverdiene i tilfelle det kommer enda mer ubehagelig eller mindre ubehagelige lydklipp.

Test 2

Det tok nesten 1 time å gjennomføre test 2 og det var tydelig at kandidatene var veldig utslitt etter endt lyttetest. Mange ga uttrykk for at testen opplevdes som alt for langvarig. Testen kunne med fordel vært delt inn i to sekvenser med en pause i mellom.

Noen av kandidatene ønsket at de skulle hatt mulighet til å spille av lydklipp de var usikker på en gang til. Ulempen med en slik løsning er at testen igjen ville vare lenger.

Rekkefølgen på sammenligning av lydklippene var satt i tilfeldig rekkefølge, men slik koden er generert sammenlignes det første lytteeksempel med alle andre 17 lydeksempler før det starter på neste som sammenlignes med resterende 16 og så videre. Dette gjør at det ikke blir full variasjon mellom lydklippene, og det at samme lydklipp spilles av mange ganger etter hverandre kan påvirke resultatet. Derfor kunne brukergrensesnittet med fordel vært videreutviklet slik at lydklippene ble presentert mer blandet.

Rekkefølgen på den enkelte sammenligning var også generert tilfeldig med at det blir “kastet mynt” over hvilket klipp som skal spilles av først og sist. Rekkefølgen kan ha innvirkning på hvilken man opplever som mest plagsom. For å unngå denne effekten kan det vurderes å sammenligne alle eksempler to ganger i ulik rekkefølge. Igjen er dette en mer tidkrevende løsning.

En ulempe med parvis sammenligning er at lydklippene kan rangeres som stadig verre eller bedre enn hverandre. For eksempel kan eksempel 1 rangeres som mer ubehagelig enn eksempel 2, og 2 rangeres som mer ubehagelig enn 3 og 3 kan igjen rangeres som mer ubehagelig enn eksempel 1. Resultatet fra denne testen ga kun totalpoengsum for hvert lydeksempel, og det kunne derfor ikke kontrolleres for denne typen sammenligninger.

Testen krever høy konsentrasjon til lytteren for å ta aktive valg på hver sammenligning. En utfordring med testen er at det var mulig å velge at eksemplene stiller likt, dette kan lett bli et alternativ dersom man er usikker. Det kan også vurderes om lydeksempelene skal bli gitt poeng eller ikke i denne sammenheng. I denne gjennomføringen fikk eksempler som ble vurdert som like ingen poeng.

5.2 Vurdering av testresultat

Diskusjon omkring testresultater er delt inn i tre deler. Først vurderes resultatene for begge testene enkeltvis før resultatene sammenlignes.

5.2.1 Resultat test 1

Framstilling i sirkelkorrelasjonsplot i figur 4.8(a), 4.8(b) og 4.8(c) viser at variablene er godt representert gjennom de to første prinsipalkomponentene. For at det skal være korrelasjon mellom variabler må de ligge i samme kvadrant, og det er større korrelasjon jo nærmere vektorene ligger. Det er små forskjeller mellom framstillingene i figur 4.8(a), 4.8(c) og 4.8(b). Vektoren til *ubehag* ligger nærmere *lydtryknivå* og *hørestyrkenivå* for de to første sammenligningene enn for sammenligningen V1-B2. Den ligger fortsatt såvidt innenfor samme kvadrant, og den ligger nærmere *hørestyrkenivå* og *lydtryknivå* enn *SNR* og *lydfremvekst*.

For sammenligningen av alle lydeksempler i figur 4.9 grupperer *lydfremvekst* og *SNR* seg vinkelrett mot *hørestyrke*, *lydtryknivå* og *ubehag*. Det viser at det er en korrelasjon mellom *ubehag*, *lydtryknivå* og *hørestyrke*, mens *lydfremvekst* og *SNR* ikke er korrelert med *ubehag*. Dette indikerer at opplevd *ubehag* avhenger mer av *lydtryknivået* enn av *lydfremveksten*.

Figur 4.10 viser også at opplevd *ubehag* følger *lydtryknivået* i større grad enn *lydfremveksten*. Figuren viser også en økning av *ubehaget* som funksjon av *lydfremvekst*. Økningen blir større for høyere *lydtryknivå*.

Sammenligning mellom de ulike lydeksempelene viser liten variasjon mellom lydeksempelene ved 30 dB(A), mens det er et tydelig skille for høyere lydnivåer hvor lydeksempelene med fuglesang, B1, rangeres som mindre ubehagelig både for 40 og 50 dB(A). Årsaken til dette kan være at fuglesang oppleves som en mer positiv bakgrunnslyd enn støy fra biltrafikk. Fuglesang opptrer ellers som en dårlig maskør, da den er smalspektret og ligger høyt i frekvensregister og dermed maskerer dårlig i frekvensområdet hvor vindturbinstøyen opptrer. Men likevel kan kanskje det som oppleves som en positiv omgivelseslyd gjøre at lytteren flytter fokus bort fra vindturbinstøyen. Forskjellen mellom vindturbinstøy 1 og 2 er mindre tydelig, men V2 rangeres hovedsakelig som mindre ubehagelig enn V2 for 40 og 50 dB(A).

Ut i fra sammenligning av poengsum gitt i resultattabell i vedlegg J.1 ser man også at det er mindre variasjon mellom avgitt minsteverdi og maksimalverdi for de laveste lydtrykkene enn for de høyeste. For to av lydklippene skiller det 10 poeng. Avstand i rangering av *ubehag* kan komme av individuelle forskjeller men kan også være et resultat av sammentrekningskjevhet.

5.2.2 Resultat test 2

PCA framstilling i figur 4.12 viser at korrelasjonen er større for V1-B2 enn V1-B1 hvor det er noe større avstand mellom vektoren til *ubehag* og *hørestyrke* og *lydnivå*. Figur 4.13 viser tett korrelasjon mellom *ubehag*, *lydtryknivå* og *hørestyrkenivå* og ingen korrelasjon mellom disse og *SNR* og *lydfremvekst*. Vektoren for *hørestyrkenivå* ligger nærmere *ubehag* enn *lydtrykk*, noe som indikerer større korrelasjon for *ubehag* og *hørestyrke*.

Figur 4.14 viser også at lydeksempelene med høyere lydtryknivå blir bedømt som mer ubehagelig enn eksempler med lavere lydtryknivå. For 40 dB(A) og 50 dB(A) er det også et klart skille fra den laveste lydfremveksten, hvor bakgrunnsstøyen er høyere enn vindturbinstøyen, og lydfremvekster hvor vindturbinstøy er like høy eller høyere enn bakgrunnsstøyen. Ubegagat rangeres som bedre for den laveste lydfremveksten.

Sammenligning av V1-B1 og V1-B2 ved ulike lydnivå i figur 4.15 viser en klar tendens til at lydeksempler med fuglelyder oppleves som mindre ubehagelig enn eksempler med bilstøy.

Det er interessant å se at kurven for V1-B1 for 30 dB(A) i figur 4.15(c) er avtagende for høyere lydfremvekster. Det vil si at lydeksempelen hvor bilstøy er mer fremtredende enn vindturbinstøy evalueres som mer ubehagelig enn lydeksempelen hvor vindturbinstøy er mest fremtredende. Den samme tendensen er ikke tilfelle for 40 og 50 dB(A). Dette er en faktor det kunne vært interessant og studert nærmere for flere lydfremvekster og lydtryknivå. Spesielt med tanke på at vindturbinstøy beskrives som mer forstyrrende enn biltrafikkstøy (3).

5.2.3 Sammenligning av resultat fra test 1 og test 2.

Ut i fra testresultatene er det interessant å sammenligne lydtryknivå, hørestyrke og lydfremvekst, samt forskjeller mellom lydklipp med ulik bakgrunnsstøy mellom de to testene.

Sammenligning lydtryknivå, hørestyrkenivå og lydfremvekst

Sirkelkorrelasjonsplottene for test 1 og test 2, figur 4.9 og 4.13 kommer fram til like sammenhenger: det er korrelasjon mellom rangert ubehag, lydtrykk og hørestyrke. Korrelasjon virker å være noe større for test 2 enn for test 1 da vektorene ligger tettere samlet. Både test 1 og test 2 viser større korrelasjon mellom ubehag og lydtryknivå og hørestyrke for lydeksempler med bilstøy enn for lydeksempler med fuglelyder.

Ved sammenligning av rangert ubehag for ulike lydtryknivå som funksjon av lydfremvekst ser man også samme tendens i test 1 og test 2: ubehaget følger hovedsakelig lydtryknivået. Test 1 viser at ubehaget også er økende for økende lydfremvekster og at forskjellen er størst for de høyeste lydtryknivåene. I test 2 er rangert ubehag konstant for 30 dB(A) men økende for 40 og 50 dB(A). Også her er forskjellen størst for 50 dB(A).

Wærås fant i sin masteroppgave at ubehag fra vindturbinstøy ved ulike lydfremvekster er nivåavhengig, og at avhengigheten er størst jo høyere lydfremvekst (6). Det samme resultatet kommer fram i denne studien. Figur 4.10 viser at opplevd ubehag følger lydtryknivået i større grad enn lydfremveksten, men at rangert ubehag har større variasjon for det høyeste lydtryknivået. Ubegaget stiger i takt med lydfremveksten. Dette støtter også de differensielle støyindikatorene benyttet i Fankrike, Italia, Irland og Storbritannia som trer i kraft når bakgrunnsstøyen er over en viss grense.

Ut i fra disse lyttetestene finnes det ikke noe belegg for å kunne si at lydfremvekst er en bedre egnet indikator enn lydtryknivå med tanke på rangert ubehag. Det vil si man ikke kan avslå grensebetingelser ved bruk av absolutte lydtryknivå fullstendig. Men det kan være interessant å studere nærmere hvilken påvirkning lydfremveksten har på ubehag for de høyeste lydtryknivåene.

Det kan virke som at hørestyrkenivået har en tettere korrelasjon til ubehag enn lydtryknivå, men dette må også studeres nærmere. Hørestyrkenivå innebærer også praktiske utfordringer da det innebærer en mer komplisert beregningsmetode.

Sammenligning ulike lydeksempler

Sammenligning av ulike lydeksempler ved ulike lydtryknivå viser at lydeksempler med fuglelyder rangeres som mindre ubehagelig enn eksempler med biltrafikkstøy. Dette gjelder både test

1 og test 1. Tendensen blir sterkere for de høyeste lydtryknivåene. Dette stemmer godt overens med resultatene til studiet gjort av Bolin m.fl. (5) som viser at hyggelige omgivelseslyder kan brukes for å forbedre lydforhold i områder utsatt for vindturbinstøy. Det samme gjelder for studiet gjort av Viollon m.fl. (7) som fant at ubehag avhenger både av typen støykilde og typen bakgrunnsstøy. Dette betyr i praksis at hyggelige omgivelseslyder kan brukes for å forbedre lydforhold i områder med vindturbinstøy.

5.2.4 Sammenligning av testmetode

Rangeringen av ubehag er ulik for de to testene. En utfordring med test 1 er at ulike deler av skalaen kan benyttes ut i fra hvor ubehagelig lytteren anser et lydeksempel i sin helhet. Dette gjør at variasjonen av rangeringen for et lydeksempel kan bli stort, noe man også ser på forholdet mellom rangert maks og minimumsverdi.

Test 2 rangerer alle lydeksempler mot hverandre, men også her er avstanden mellom maks og minimumverdi stor. Det kan avhenge av rekkefølgen til lytteeksemplene, men også hvor aktive valg lytteren tar. Testen blir sårbar for forstyrrelser siden lydeksemplene er korte og avgjørelser må bli tatt ofte. Det kan også vurderes om lytteren rekker å få det samme inntrykket av et lydeksempel når lyttetiden blir kort, eller om det bør argumenteres for å ha lydeksempler som varer lengre siden det rangeres lyd som vanligvis høres over lengre perioder.

I denne sammenhengen viser resultatene til de to testene mye av de samme tendensene og det kan virke som at typen test ikke har innflytelse på den kvantitative vurderingen. For å studere dette nærmere hadde det vært interessant å la de samme testpersoner gjennomføre begge lyttetestene for å sammenligne resultatet til hver enkelt kandidat.

Konklusjon

Gjennomføring av lyttetester med bruk av elementvis vurdering og parvis sammenligning har gitt indikasjoner på at det er en større sammenheng mellom opplevd ubehag og lydtryknivå og hørestyrke enn for ubehag og lydfremvekst.

Resultatene tyder også på at det kan finnes en sammenheng mellom lydfremvekst og ubehag, og at denne blir mer tydelig for høyere lydtryknivå, men dette må undersøkes nærmere. Vindturbinstøy med positiv bakgrunnslyd, i form av fuglelyder, oppleves for lytteren som mindre ubehagelig enn veitrafikkstøy som bakgrunnskilde.

Begge formene for lyttetest kom fram til de samme sammenhengene mellom de ulike indikatorene. Dette gir en indikasjon på at typen lyttetest ikke har påvirkning på sammenhengen sansing og kvantitativ vurdering.

Bibliografi

- [1] Dutilleul, G., Gjestland, T., & Licitra, G. Challenges of the use of sound emergence for setting legal noise limits. *International Journal of Environmental Research and Public Health*, 2019. 16, 4517–.
- [2] Hau, E. *Wind Turbines, Fundamental, Technologies, Application, Economics*,. 3.utg., Springer. 2013.
- [3] Doolan, C. A review of wind turbine noise perception, annoyance and low frequency emission. *Wind Engineering*, 2012. 37(1), 97–104.
- [4] Bakker, R., Pedersen, E., van den Berg, G., Stewart, R., Lok, W., & Bouma, J. Impact of wind turbine sound on annoyance, self-reported sleep disturbance and psychological distress. *Science of the total environment*, 2012. 425, 42–51.
- [5] Bolin, K., Kedhammar, A., & Nilsson, M. E. The influence of background sounds on loudness and annoyance of wind turbine noise. *Acta Acustica united with Acoustica*, 2012. 98, 741–748.
- [6] Wærås, O. L. K. Perception an quantification of wind turbine noise. Masteroppgave, Norwegian University of Science and Technology, 2018.
- [7] Viollon, S., Marquis-Favre, C., Junker, F., & Baumann, C. Environmental assessment of industrial noises annoyance with the criterion sound emergence". In *International Congress of Acoustics*. 2004, Kyoto, Japan.
- [8] Fosset, J. Perception et quantification du bruit d'une éolienne. Praksisplassrapport, Ecole Nationale des Travaux Publics de l'Etat, 2019.
- [9] Wheeler, R. M. *The Science of Sound*. 3.utg., Pearson New International Edition. 2014.
- [10] Fastl, H. & Zwicker, E. *Psychoacoustics, facts and models*. 3.utg., Springer. 2007.
- [11] Bech, S. & Zacharow, N. *Perceptual Audio Evaluation, Theory, Method and Application*. John Wiley & sons. 2006.
- [12] ISO Central Secretary. Acoustics - assessment of noise annoyance by means of social and socio-acoustic surveys. Standard ISO/TS 15666:2003, International Organization for Standardization, Geneva, CH, 2003. URL:

<https://www.standard.no/no/Nettbutikk/produktkatalogen/Produktpresentasjon/?ProductID=12881>, [hentet 20.06.19].

- [13] Miljødirektoratet. 2018. M-128: Veileder til retningslinje for behandling av støy i arealplanlegging. <https://www.miljodirektoratet.no/publikasjoner/2014/februar-2014/veileder-til-retningslinje-for-behandling-av-stoy-i-arealplanlegging/>.
- [14] Retningslinje for behandling av støy i arealplanlegging (t-1442), 2016. <https://www.regjeringen.no/no/dokumenter/retningslinje-for-behandling-av-stoy-i-arealplanlegging/id2526240/>.
- [15] Singh, R. Principal component analysis (pca) with code on mnist dataset. <https://medium.com/analytics-vidhya/principal-component-analysis-pca-with-code-on-mnist-dataset-da7de0d07c22> [Hentet 11.12.19].
- [16] Head acoustics. https://www.head-acoustics.com/fr/press_releases_calculation_of_loudness_of_stationary_and_time_dependent_sounds.htm [Hentet 10.12.19].

Utstyrslister

A.1 Utstyrliste benyttet til opptak av bakgrunnskilder

Utstyr	
Opptaker	Sound Devices 722
Vindskjer	Rycote Cyclone Windshield
8-talls mikrofon	Sennheiser, MKH 30 P48
1/2 " frittfelt mikrofon	Bruel & Kjaer 4190
Forforsterker	Norsonic NOR 1201

Tabell A.1: Utstyr brukt til opptak

A.2 Utstyr benyttet på lyttetester

Utstyr	type
Lydkort	Roland UA - 1610 Studio Capture
Datamaskin	Dell Latitude 5580
Hodetelefoner	Beyerdynamic DT 990
Lydnivåmåler	Norsonic 150

Tabell A.2: Utstyr brukt på lyttetesten.

A.3 Utstyr benyttet til kalibrering av hodetelefoner

Utstyr	type	serienummer
Mikrofon	Bruel & Kjør 4190	2263075
Preamp	Norsonic nor 1201	
Kalibrator	Brüel & Kjør 4231	
Høytaler	egenprodusert NTNU	
Lydkort	Roland UA - 1610 Studio Capture	
Datamaskin	Dell Latitude 5580	02/043
Kunsthode	Neumann KU 81i	
		NTNU utstyrsnummer IN-4003
Høretelefoner	Beyerdynamic DT 990	
Programvare	Audacity	
	Matlab _R 2017b	
	EASERA	

Tabell A.3: Utstyrliste for kalibrering av hodetelefoner

Liste over lydeksempler

B.1 Lydeksempler test 1

Lyd-eks.	Vindturbin-støy	Bakgrunns-støy	SNR	Lydtrykk-nivå [dBa]	Loudness LN [Phon]
1	V1	B1	-10	30	47,0
2	V1	B1	-5	30	47,7
3	V1	B1	0	30	48,6
4	V1	B1	5	30	49,3
5	V1	B1	10	30	49,7
6	V1	B1	-10	40	60,4
7	V1	B1	-5	40	61,0
8	V1	B1	0	40	61,8
9	V1	B1	5	40	62,3
10	V1	B1	10	40	62,6
11	V1	B1	-10	50	71,3
12	V1	B1	-5	50	71,8
13	V1	B1	0	50	72,5
14	V1	B1	5	50	73,0
15	V1	B1	10	50	73,2

Lyd-eks.	Vindturbin-støy	Bakgrunns-støy	SNR	Lydtrykk-nivå [dBa]	Loudness LN [Phon]
16	V2	B1	-10	30	47,1
17	V2	B1	-5	30	47,7
18	V2	B1	0	30	48,7
19	V2	B1	5	30	49,3
20	V2	B1	10	30	49,6
21	V2	B1	-10	40	60,5
22	V2	B1	-5	40	61,1
23	V2	B1	0	40	61,9
24	V2	B1	5	40	62,4
25	V2	B1	10	40	62,7
26	V2	B1	-10	50	71,3
27	V2	B1	-5	50	71,9
28	V2	B1	0	50	72,6
29	V2	B1	5	50	73,1
30	V2	B1	10	50	73,3
31	V1	B2	-10	30	48,8
32	V1	B2	-5	30	49,9
33	V1	B2	0	30	50,7
34	V1	B2	5	30	50,7
35	V1	B2	10	30	50,3
36	V1	B2	-10	40	61,8
37	V1	B2	-5	40	62,7
38	V1	B2	0	40	63,3
39	V1	B2	5	40	63,4
40	V1	B2	10	40	63,1
41	V1	B2	-10	50	72,5
42	V1	B2	-5	50	73,3
43	V1	B2	0	50	73,8
44	V1	B2	5	50	73,9
45	V1	B2	10	50	73,6

Tabell B.1: Lydeksempler test 2. V1: vindturbin med SNR lik 61, V2: vindturbin med SNR lik 40, B1: veitrafikkstøy, B2: fuglelyder.

B.2 Lydeksempler test 2

Lyd-eksempel	Vindturbin-støy	Bakgrunns-støy	SNR nivå [dBA]	Lydtrykk- [Phon]	Loudness LN
1	V1	B1	-10	30	47,3
2	V1	B1	0	30	48,8
3	V1	B1	10	30	49,8
4	V1	B1	-10	40	60,6
5	V1	B1	0	40	61,9
6	V1	B1	10	40	62,7
7	V1	B1	-10	50	71,4
8	V1	B1	0	50	72,6
9	V1	B1	10	50	73,3
10	V1	B2	-10	30	49,3
11	V1	B2	0	30	50,5
12	V1	B2	10	30	50,2
13	V1	B2	-10	40	62,1
14	V1	B2	0	40	63,2
15	V1	B2	10	40	63
16	V1	B2	-10	50	72,7
17	V1	B2	0	50	73,7
18	V1	B2	10	50	73,5

Tabell B.2: Lydeksempler test 2. V1: vindturbin med SNR lik 61, B1: veitrafikkstøy, B2: fuglelyder.

Skript for bearbeiding av lydfiler

```
%Dette skriptet bearbeider lydfilene til riktig SNR og lydtrykniva
%Henter ut samplingsfrekvensene
[~,FS]=audioread('Master/V1.WAV');

%Signallengde test 1, 30 sekunder
signalstartB = 1/FS*FSB;
signalendB   = (1/FS + 30)*FSB;

%Signallengde test 2, 7 sekund
% signalstartV = 1*FS;
% signalendV   = (7)*FS;

samplesW=[signalstartV ,signalendV ];
%Onsket SNR og lydtrykniva
Laeq_target=50;
SNRdB=10;

%Henter ut lydfilene til onsket lengde
[Wnoise,FS] = audioread('Master/V1.WAV',samplesW);
[Bnoise,FSB] =audioread('Master/sound_B2new.WAV',samplesW);

%A-veker signalene
Wnoise_A = Avekting(Wnoise);
Bnoise_A=Avekting(Bnoise);

%%Skalarer signalene til riktig SNR

Wnoise_scal=snrskalering(Wnoise_A,Bnoise_A,SNRdB);

%Setter de nye signalene sammen til en lydfil
signaltot = Wnoise_scal + Bnoise_A;
```

```
%Finner skaleringsfaktor for a gi riktig dB verdi
s = skalering(signalrms(signaltot), Laeq_target);
signaltot_s=signaltot*s;
```

```
%Lagrer filen
audiowrite('t2_18.WAV', signaltot_s, FS);
```

C.1 Hjelpesfunksjon, Avekting

```
%A-veker et signal med en filterfunksjon
function [signal_a] = Avekting(signal);
```

```
b = [0.169994948147430, 0.280415310498794, -1.120574766348363,...
      0.131562559965936, 0.974153561246036, -0.282740857326553,...
      -0.152810756202003];
a = [1.000000000000000000, -2.12979364760736134, 0.42996125885751674,...
      1.62132698199721426, -0.96669962900852902, 0.00121015844426781,...
      0.04400300696788968];
```

```
signal_a = filter(b, a, signal);
```

```
end
```

C.2 Hjelpesfunksjon, snrskalering

```
%Skalerer signal ps slik at SNR blir lik SNR_T
```

```
function pws = snrskalering4(ps,pb, SNR_T)
```

```
SNR_0=10*log10(rms(ps)^2/rms(pb)^2);
```

```
K = 10^((SNR_T-SNR_0)/10);
```

```
pws = ps*sqrt(K);
```

```
end
```


C.3 Hjelpesfunksjon, skalering

```
%finner skaleringsfaktoren for at lydtrykknivaet  
%skal bli lik Laeq_target  
function s = skalering(signal, Laeq_target)  
P0=2e-5;  
  
s=10^(Laeq_target/20)*P0/signal;  
  
end
```


Kalibreringsfilter

```
%Setter samplingsfrekven , orden og bandbredde
Fs = 48e3; %samplingsfrekvens
N = 2; %orden
Bw = 500/(Fs/2); %Bandbredde

%Verdier tatt fra 1/3 oktavb,100Hz– 1000Hz
G = [4.7 1 1.3 1.7 0.2 0.7 0.5 -0.6 1.2 0.4 -3.0];
Wo= [100 125 160 200 250 315 400 500 630 800 1000]./(Fs/2);

%Verdier tatt fra 1/6 oktavb , fra 1000Hz – 200000Hz
G2 = [-4.4 -3.5 -4.1 -2.3 -1.2 -1.3 -2.7 -2.4 -1.1 0.1 1.1 -0.2 -2.7...
      -3.5 -3.2 0.9 -3.6 -3.9 -8.3 -14.3 -15.9 -17.2 -16.3...
      -9.6 -5.9 -1.3];
Wo2=[1120 1250 1400 1600 1800 2000 2240 2500 2800 3150 3550 4000.0...
     4500 5000 5600 6300 7100 8000 9000 10000 11200 12500 14000 ...
     16000 18000 20000]./(Fs/2);

%Designer filter parametre
[B1, A1] = designParamEQ(N, G(1), Wo(1), Bw);
BQ1 = dsp.BiquadFilter('SOSMatrix',[B1.',[1,A1.']]');
[B2, A2] = designParamEQ(N, G(2), Wo(2), Bw);
BQ2 = dsp.BiquadFilter('SOSMatrix',[B2.',[1,A2.']]');
[B3, A3] = designParamEQ(N, G(3), Wo(3), Bw);
BQ3 = dsp.BiquadFilter('SOSMatrix',[B3.',[1,A3.']]');
[B4, A4] = designParamEQ(N, G(4), Wo(4), Bw);
BQ4 = dsp.BiquadFilter('SOSMatrix',[B4.',[1,A4.']]');
[B5, A5] = designParamEQ(N, G(5), Wo(5), Bw);
BQ5 = dsp.BiquadFilter('SOSMatrix',[B5.',[1,A5.']]');
[B, A] = designParamEQ(N, G(6), Wo(6), Bw);
BQ6 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');
[B, A] = designParamEQ(N, G(7), Wo(7), Bw);
BQ7 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');
[B, A] = designParamEQ(N, G(8), Wo(8), Bw);
```

```
BQ8 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G(9), Wo(9), Bw);
BQ9 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G(10), Wo(10), Bw);
BQ10 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
%-----%
[B, A] = designParamEQ(N, G(11), Wo(11), Bw);
BQ11 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(1), Wo2(1), Bw);
BQ12 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(2), Wo2(2), Bw);
BQ13 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(3), Wo2(3), Bw);
BQ14 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(4), Wo2(4), Bw);
BQ15 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(5), Wo2(5), Bw);
BQ16 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(6), Wo2(6), Bw);
BQ17 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(7), Wo2(7), Bw);
BQ18 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(8), Wo2(8), Bw);
BQ19 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(9), Wo2(9), Bw);
BQ20 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(10), Wo2(10), Bw);
%-----%
BQ21 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(11), Wo2(11), Bw);
BQ22 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(12), Wo2(12), Bw);
BQ23 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(13), Wo2(13), Bw);
BQ24 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(14), Wo2(14), Bw);
BQ25 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(15), Wo2(15), Bw);
BQ26 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(16), Wo2(16), Bw);
BQ27 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(17), Wo2(17), Bw);
BQ28 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(18), Wo2(18), Bw);
BQ29 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
[B, A] = designParamEQ(N, G2(19), Wo2(19), Bw);
BQ30 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']] );
```

```

[B, A] = designParamEQ(N, G2(20), Wo2(20), Bw);
%////////////////////////////////////////
BQ31 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');
[B, A] = designParamEQ(N, G2(21), Wo2(21), Bw);
BQ32 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');
[B, A] = designParamEQ(N, G2(22), Wo2(22), Bw);
BQ33 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');
[B, A] = designParamEQ(N, G2(23), Wo2(23), Bw);
BQ34 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');
[B, A] = designParamEQ(N, G2(24), Wo2(24), Bw);
BQ35 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');
[B, A] = designParamEQ(N, G2(25), Wo2(25), Bw);
BQ36 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');
[B, A] = designParamEQ(N, G2(26), Wo2(26), Bw);
BQ37 = dsp.BiquadFilter('SOSMatrix',[B.',[1,A.']]');

FC1 = dsp.FilterCascade(BQ1, BQ2, BQ3, BQ4, BQ5, BQ6, BQ7, BQ8,...
BQ9, BQ10);
FC2 = dsp.FilterCascade(BQ11, BQ12, BQ13, BQ14, BQ15, BQ16, BQ17,...
BQ18, BQ19, BQ20);
FC3 = dsp.FilterCascade(BQ21, BQ22, BQ23, BQ24, BQ25, BQ26, BQ27,...
BQ28, BQ29, BQ30);
FC4 = dsp.FilterCascade(BQ31, BQ32, BQ33, BQ34, BQ35, BQ36, BQ37);

hfvt = fvtool(FC1,'Fs',Fs,'Color','white','FrequencyScale','Log');
legend(hfvt,'Cascade of 2nd order filters');

hfvt = fvtool(FC2,'Fs',Fs,'Color','white','FrequencyScale','Log');
legend(hfvt,'Cascade of 2nd order filters');

hfvt = fvtool(FC3,'Fs',Fs,'Color','white','FrequencyScale','Log');
legend(hfvt,'Cascade of 2nd order filters');

hfvt = fvtool(FC4,'Fs',Fs,'Color','white','FrequencyScale','Log');
legend(hfvt,'Cascade of 2nd order filters');

%Henter ut signalet og sender gjennom filteret.
%Filteret ma deles i 4 kaskade
%filtere siden matlab bare tar 8 filterverdier i samme filter.
noise = audioread('Kalibrering/headset.WAV');
nY=FC1(noise);
nY2=FC2(nY);
nY3=FC3(nY2);
nY4=FC4(nY3);

%Lager ny, filtrert lydfil

```

```
audiowrite('kalibreringstest.WAV',nY4,Fs);
```

Kalibrering av lydtrykknivå

```
%Konstanter
p0=20e-6; %Pa, referansetrykk.

%Henter ut samplingsfrekvens og bestemmer valgt lengde pa signalet
[kalibrator,FS]=audioread('kalibratoropptak.WAV');
signalstart = 1*FS;
signalend   = (6)*FS;
samples=[signalstart,signalend];
t=0:1/FS:6/2;

%Henter og klipper signalene sa de blir 5 sekunder.
[kalibrator,~]=audioread('kalibratoropptak.WAV',samples);
[mikrofon,~]=audioread('referanse_mic.WAV',samples);
[kunsthode,~]=audioread('kunsthod_ikkekal.WAV',samples);
[hodetelefon_kunsthode,FS]=audioread('headset.WAV',samples);

%Finner lydtrykksnivaet til opptaket av kalibratoren.
%Det skal vaere 94 dB
s = skalering(signalrms(kalibrator), 94);

%Finner lydtrykkniva for pink noise fra mikrofonen
mikrofon_ref = mikrofon*s;
%Skalerer til kalibrert niva
rms_ref=signalrms(mikrofon_ref);
%Beregner ny rms verdi
Lpref = 20*log10(rms_ref/p0);

%Lydtrykknivaet til kunsthodet ma vaere
%likt lydtrykknivaet til mikrofonen.

s_k = skalering(signalrms(kunsthode),Lpref);

kunsthode_s=kunsthode*s_k;
```

```
Lp_kunsthode_kal = 20*log10(signalrms(kunsthode_s)/p0);
```

E.1 Hjelpesfunksjon, signalrms

```
function signal_rms = signalrms(signal)

signal_rms = rms(sqrt(signal.^2));
end
```


Introduksjonstekst til test

F.1 Introduksjonstekst test 1

Listening test about Wind Turbine noise

In this test, you will listen to sound clips that contain wind turbine sounds in different contexts.

The test will last for approximately 45 minutes and your task is to listen to and evaluate different sound samples.

The sound levels used in the test are quite low, which means there is no risk for hearing damage.

Each example lasts for 30 seconds. After the example is finished you will evaluate your annoyance level on a scale from 0 to 10. 0 indicates that you do not find the sound annoying at all, while 10 indicates that you are experiencing the sound as extremely annoying. Confirm your value by clicking "confirm".

During the test, it is important that you imagine that you live nearby a wind farm and that the sounds you hear are something that you would hear every day for a long period of time. For example such sounds can be heard if you open your bedroom window or if you are outside your house (gardening, sitting on the balcony, or going for a walk).

You can choose when to start the next sound example by clicking on the "play" button. It is OK to take short breaks during the test.

If you are unsure if everything works as it should, or if you need to terminate the test for other reasons, you can walk out of the room the same way as you walked in.

Because of confidentiality you will be given a subject number.

F.2 Introduksjonstekst test 2

Listening test about Wind Turbine noise

In this test, you will listen to sound clips that contain wind turbine sounds in different contexts.

In the test you will listen and compare sound samples. You will listen to two sound samples, A and B, which both lasts for 7 seconds. You will then rate which one of them is the most annoying. If you find sound sample A most annoying you press "A", if sound sample "B" is most annoying you press "B". If you evaluate the annoyance at the same level you press "The same". Wait until both of the sounds are finished before you press the button.

The sound levels used in the test are quite low, which means there is no risk for hearing damage.

During the test, it is important that you imagine that you live nearby a wind farm and that the sounds you hear are something that you would hear every day for a long period of time. For example such sounds can be heard if you open your bedroom window or if you are outside your house (gardening, sitting on the balcony, or going for a walk).

You can choose when to start the next sound example by clicking on the "play" button. It is OK to take short breaks during the test.

If you are unsure if everything works as it should, or if you need to terminate the test for other reasons, you can walk out of the room the same way as you walked in.

Because of confidentiality you will be given a subject number.

Kode brukergrensesnitt

G.1 Test 1

Listing G.1: Insert code directly in your document

```

###Brukergrensesnitt###
#Denne koden er basert pa et skript utviklet av Jean Fosset.

###Modul Import
import os
import time
import random as rd
import csv as csv
import tkinter as tk
import tkinter.ttk
import tkinter.messagebox
import pygame as py

###Definisjon av stier
path_global = 'C:\\Users\\malepc_2\\Desktop\\Aurora'

path_image = path_global + '\\Images'
path_CSV = path_global + '\\Files'
path_sound = path_global + '\\Sons'

os.chdir(path_global)
os.chdir(path_global)

### Initialisering av lydklipp
py.mixer.init() #for pygame

###Global funksjon for valg av lydklipp
def choice_sound(which_list, number_of_sound):
    """Genererer en liste med klipp og blander den"""
    list_sound = []
    path = path_sound + '\\Sound_' + which_list
    os.chdir(path)

    #Alle lydene har navn som sound1, sound2
    for j in range(1, number_of_sound + 1):

```

```

        nom_fichier_son='Sound_'+str(j)+'.wav' #open file .wav
        list_sound.append([ nom_fichier_son ,py.mixer.Sound(nom_fichier_son)])
        #append the new sound in the list

    rd.shuffle(list_sound) #Blander listen i tilfeldig rekkefølge
    return list_sound

###Definisjon av globale
#Definijoner vinduet
screen_x = 3840
screen_y = 2160

#Variabler for a lagre informasjon
infos_user = [] #informasjon om brukeren
mark_unpleasantness = [] #Liste med resultat
yn_cyh = [] #Svar pa om sporsmal om lytteren ble forstyrret under testen
infos_test = [] #Evt. forklaring pa hva forstyrrelsen var.

#Til bruk i selve testfasen
index_number_of_test = 0 #Holder telling lydeksempel
number_of_test = 45 #Totalt antall lydeksempler
test = True
max_progressbar = number_of_test + 0.0001

#Global
waiting_before_answer = 2
#Til testfase
sound_test = choice_sound('test',number_of_test) #Henter inn lista med lydklipp
sound_time=30 #Lengde pa lydklipp i sekund
max_sound_time = sound_time + 0.0001 #Definerer progresjonen til "lydbaren"

###Hovedprogram – grafisk brukergrensesnitt###
##Danner hovedvinduet
root = tk.Tk()
root.attributes('-fullscreen',1)

##Bilde
#Konfigurasjon til rad og kolonne
n_ligne=3
n_column=3
for i in range(1,n_ligne+1):
    root.rowconfigure(i, weight=1)
for j in range(1,n_column+1):
    root.columnconfigure(j, weight=1)
    if j == n_column:
        root.columnconfigure(j, weight=1)

#ogo NTNU
canvas_logo = tk.Canvas(root, width=688, height=176)
#apner bilde og setter innO
os.chdir(path_image)
logo_ntnu = tk.PhotoImage(file='NTNU_logo.png')

```

```

canvas_logo.create_image(688/2, 176/2, image = logo_ntnu)
#Dannelse
canvas_logo.grid(row=1, column=1, sticky='nw', padx=20, pady=20)

canvas_useless = tk.Canvas(root, width=688, height=176)
canvas_useless.grid(row=3, column=3)

###Fase 1
##Funksjoner gjennom a trykke pa knapper
def save_phase_1():
    """Lagrer brukerinformasjon inn i lista "infos_user" og lukker det første vinduet"""
    #Lagrer informasjon
    infos_user.append(field_subject.get())
    infos_user.append(field_age.get())
    infos_user.append(gender.get())
    infos_user.append(who_you_are.get())
    infos_user.append(prev_exp.get())

    #Lukker vinduene
    window_phase_1.grid_remove()

    #forbereder nytt vindu
    window_phase_2.grid(row=2, column=2)

    #oppdaterer
    root.update()

##Grensesnitt
#Danner hovedvindu og dimensjoner
dim_x_1 = 0.26 * screen_x #1000 px en 3840
dim_y_1 = 0.46 * screen_y #1000 px en 2160
window_phase_1 = tk.LabelFrame(root, text='About_you', width=dim_x_1, height=dim_y_1,
labelanchor='n', bd=10)
window_phase_1.grid_propagate(0)

#Konfigurasjon rad/kolonne
n_ligne=8
n_column=3
for i in range(1, n_ligne+1):
    window_phase_1.rowconfigure(i, weight=1)
for j in range(1, n_column+1):
    window_phase_1.columnconfigure(j, weight=2)
    if j == n_column:
        window_phase_1.columnconfigure(j, weight=1)

#Eksplisitt test
label_explication = tk.Label(window_phase_1, text="Welcome_to_the_test!
Please_enter_necessary_personal_information_before_starting_the_listening_test.",
fg='black')
label_explication.grid(column=1, row=1, columnspan=2, rowspan=2, padx=5, pady=5)

#Ramme for subjektnummer
frame_subject_number = tk.LabelFrame(window_phase_1, text="Subject_number",
padx=20, pady=20)
frame_subject_number.grid(column=1, row=3, columnspan=1, rowspan=1, padx=5, pady=5)

```

```

#sone for subjektnummer
field_subject = tk.Entry(frame_subject_number, width = 4)
field_subject.grid()

#Ramme for alder
frame_age = tk.LabelFrame(window_phase_1, text="Age", padx=20, pady=20)
frame_age.grid(column=2, row=3, columnspan=1, rowspan=1, padx=5, pady=5)
#sone for alder
field_age = tk.Entry(frame_age, width=3)
field_age.grid()

#ramme for kjonn
frame_gender = tk.LabelFrame(window_phase_1, text="Gender", padx=20, pady=20)
frame_gender.grid(column=1, row=4, columnspan=1, rowspan=1, padx=5, pady=5)
#knapp for a velge kjonn
gender = tk.StringVar()
gender.set('M') # 'Male' satt som normalverdi
male = tk.Radiobutton(frame_gender, text="Male", variable=gender, value='M')
female = tk.Radiobutton(frame_gender, text="Female", variable=gender, value='F')
others = tk.Radiobutton(frame_gender, text="Others", variable=gender, value='O')
male.grid(sticky='w')
female.grid(sticky='w')
others.grid(sticky='w')

#Ramme for a velge posisjon
frame_wya = tk.LabelFrame(window_phase_1, text="I_am_a", padx=20, pady=20)
frame_wya.grid(column=2, row=4, columnspan=1, rowspan=1, padx=5, pady=5,)
#Knapp for velge posisjon
who_you_are = tk.StringVar()
who_you_are.set('S') #student satt som normalverdi
student = tk.Radiobutton(frame_wya, text="Student", variable=who_you_are, value='S')
ntnu_employee = tk.Radiobutton(frame_wya, text="NTNU_employee",
variable=who_you_are, value='NE')
others_wya = tk.Radiobutton(frame_wya, text="Others", variable=who_you_are, value='O')
student.grid(sticky='w')
ntnu_employee.grid(sticky='w')
others_wya.grid(sticky='w')

#Tekst om tidligere erfaring
frame_previous = tk.LabelFrame(window_phase_1,
text="Do_you_have_previous_experience_with_wind_turbine_noise?", padx=20, pady=20)
frame_previous.grid(column=1, row=7, columnspan=2, rowspan=1, padx=5, pady=5, sticky='')
#Tidligere erfaring
prev_exp=tk.StringVar()
prev_exp.set('Y') # "yes" satt som normalverdi
yes_11 = tk.Radiobutton(frame_previous, text="Yes", variable=prev_exp, value='Y')
no_11 = tk.Radiobutton(frame_previous, text="No", variable=prev_exp, value='N')
yes_11.grid()
no_11.grid()

#Neste knapp
next_button_1 = tk.Button(window_phase_1, text="Next", command=save_phase_1)
next_button_1.grid(column=3, row=8, columnspan=1, rowspan=1, padx=5, pady=5, sticky='se')

#Viser

```

```

window_phase_1.grid(row=2, column=2)

### Testfase
## Funksjoner via knapp
def debut_test():
    """ Starter lyttetesten """
    #aktiverer knapp
    bouton_playsound_2['state'] = 'normal'
    bouton_start_2['state'] = 'disable'

def play_sound_2():
    """ Spiller av lyd, aktiverer og deaktiverer knapper """
    global test

    #Deaktive lydknapp
    bouton_playsound_2['state'] = 'disable'

    if test: #Sjekker om det er testfase
        global index_number_of_test
        #Avspiller lyd
        son = sound_test[index_number_of_test][1]
        son.play()

    #Progresjons "sound bar"
    #Brukes for at sound bar skal flyttes framover
    for j in range(0, sound_time):
        time.sleep(1) #venter 1 sekund
        progressbar_sound_2.step(1) #flyttes framover
        window_phase_2.update_idletasks() #oppdaterer vinduet

    value_progresssound_2.set(0) #restarter verdier

    #venter for bruker avgir svar
    time.sleep(waiting_before_answer)

    #aktivering av andre knapper
    scale_2['state'] = 'normal'
    bouton_val_scale['state'] = 'normal'

def validation_2():
    """ Bekrefter og lagrer valgt verdi """
    global text_progression_progressbar
    global test

    #resetter alle knapper til normaltilstand
    bouton_playsound_2['state'] = 'normal'
    scale_2['state'] = 'disable'
    bouton_val_scale['state'] = 'disable'

    #Progresjonsbar
    progressbar_2.step(1)

    if test: #Sjekker at det er test
        global index_number_of_test

```

```

#Lagrer verdien
mark_unpleasantness.append([sound_test[index_number_of_test][0],
unpleasantness.get()])
unpleasantness.set(0) #resetter verdien til 0

index_number_of_test+=1

#tekst progresjonsbar
text_progression_progressbar.set(str(index_number_of_test)+'/'
+str(number_of_test))
print('This is the', index_number_of_test, 'sound sample.')

window_phase_2.update_idletasks()

if index_number_of_test==number_of_test: #Nar testen er ferdig
    #resetter knappene
    scale_2['state'] = 'disable'
    bouton_val_scale['state'] = 'disable'
    bouton_playsound_2['state'] = 'disable'

    #Avslutter testen
    test=False

    #Informasjonstekst om at testen er ferdig
    tk.messagebox.showinfo(title = 'Test_end',
    message = 'The listening test is completed.
    Please press next to continue to some finishing questions.')

    bouton_next_2['state']='normal'

#Resetter skaleringen
unpleasantness.set(0)

def fin_phase_2():
    """Avslutter fase testfase (fase 2)"""
    #Lukker vindu
    window_phase_2.grid_remove()

    #Nytt vindu
    window_phase_3.grid(row=2, column=2)

    #Oppdaterer
    root.update()

##Grensesnitt
#Danner hovedvindu og geometri
dim_x_2 = 0.417 * screen_x #1600 px en 3840
dim_y_2 = 0.65 * screen_y #1400 px en 2160
window_phase_2 = tk.LabelFrame(root, text='Listening_test', width=dim_x_2,
height=dim_y_2, bd=10, labelanchor='n')
window_phase_2.grid_propagate(0)

#Konfigurasjon av rad og kolonne

```



```

n_ligne=8
n_column=2
for i in range(1,n_ligne+1):
    window_phase_2.rowconfigure(i, weight=1)
for j in range(1,n_column+1):
    window_phase_2.columnconfigure(j, weight=2)
    if j == n_column:
        window_phase_2.columnconfigure(j, weight=1)

#Forklaringstekst og initialiserer startknapp og progresjonsbar.
text_explication_2 = tk.StringVar()
text_button_start_2 = tk.StringVar()
text_progression_progressbar = tk.StringVar()
if test:
    text_explication_2.set('Imagine that you live nearby a wind farm and \n
    that sometimes when you are outside \n or have a window open you hear this
    sound.\n How disturbing would you experience the sound to be on a scale from
    0-10?\n 0 indicate "not at all" while 10 indicate "extremely disturbing".\n')
    text_button_start_2.set('Start')
    text_progression_progressbar.set('0'+ '/' +str(number_of_test))

#Tekstforklaring
label_indication = tk.Label(window_phase_2, textvariable=text_explication_2, fg='black')
label_indication.grid(column=1, row=1, columnspan=1, rowspan=1, padx=5, pady=5)

#Startknapp
bouton_start_2 = tk.Button(window_phase_2, textvariable=text_button_start_2,
command=debut_test)
bouton_start_2.grid(column=1, row=2, columnspan=1, rowspan=1, padx=5, pady=5)

#Ramme inn
frame_procedure_2 = tk.LabelFrame(window_phase_2, padx=20, pady=20, relief=tk.GROOVE)
frame_procedure_2.grid(column=1, row=3, columnspan=1, rowspan=1, padx=5, pady=5)

#Knapp for a spille av lyd
#ramme
frame_played_2 = tk.LabelFrame(frame_procedure_2, text="1. Play to listen",
padx=20, pady=20)
frame_played_2.grid(column=1, row=3, columnspan=1, rowspan=1, padx=5, pady=5)
#knapp
bouton_playsound_2 = tk.Button(frame_played_2, text="Play", command=play_sound_2,
state='disabled')
bouton_playsound_2.grid()

#Progresjonsbar, lyd
#ramme
frame_progressbar_sound_2 = tk.LabelFrame(frame_procedure_2,
text="Progress Sound Bar", padx=20, pady=20)
frame_progressbar_sound_2.grid(column=2, row=3, columnspan=1, rowspan=1,
padx=5, pady=5)
#knapp
value_progresssound_2 = tk.StringVar()
progressbar_sound_2 = tk.ttk.Progressbar(frame_progressbar_sound_2,
orient='horizontal',

```

```

length=500, mode='determinate', maximum=max_sound_time, variable=value_progresssound_2)
progressbar_sound_2.grid()

#Skaleringsknapp
#ramme
frame_scale_2 = tk.LabelFrame(frame_procedure_2, text="2. Mark your unpleasantness",
padx=20, pady=20)
frame_scale_2.grid(column=1, row=4, columnspan=1, rowspan=1, padx=5, pady=5)
#knapp
unpleasantness = tk.DoubleVar()
unpleasantness.set(0)
scale_2 = tk.Scale(frame_scale_2, orient='horizontal', from_=0, to=10, resolution=0.5,
tickinterval=2, length=350, label='Unpleasantness', variable=unpleasantness,
state='disable')
scale_2.grid()

#Knapp for a bekrefte skaleringsverdi
#ramme
frame_confirm_2 = tk.LabelFrame(frame_procedure_2, text="3. Confirm the value",
padx=20, pady=20)
frame_confirm_2.grid(column=1, row=5, columnspan=1, rowspan=1, padx=5, pady=5)
#knapp
bouton_val_scale=tk.Button(frame_confirm_2, text="Confirm", command=validation_2,
state='disable')
bouton_val_scale.grid()

#Progresjonsbar
#ramme
frame_progressbar_2 = tk.LabelFrame(window_phase_2, text="Progressbar",
padx=20, pady=20)
frame_progressbar_2.grid(column=1, row=7, columnspan=1, rowspan=1, padx=5, pady=5)
#knapp
value_progress = tk.StringVar()
progressbar_2 = tk.ttk.Progressbar(frame_progressbar_2, orient='horizontal', length=500,
mode='determinate', maximum=max_progressbar, variable=value_progress)
progressbar_2.grid()

#Tekst progresjonsbar
label_progressbar_2 = tk.Label(frame_progressbar_2,
textvariable=text_progression_progressbar, fg='red')
label_progressbar_2.grid(column=1, row=7, columnspan=1, rowspan=1, padx=5, pady=5)

#knapp for a avslutte underveis i testen
button_2 = tk.Button(window_phase_2, text="Quick_quit", command=fin_phase_2)
button_2.grid(column=1, row=8, columnspan=1, rowspan=1, padx=5, pady=5, sticky='sw')

#Knapp for avslutte testfase
bouton_next_2 = tk.Button(window_phase_2, text="Next", command=fin_phase_2,
state='disable')
bouton_next_2.grid(column=1, row=8, columnspan=1, rowspan=1,
padx=5, pady=5, sticky='se')

###Phase 3, avslutning###
def save_phase_3():

```

```

"""Lagrer all informasjon om testen"""
#Lagrer informasjon
infos_test.append(disturbance.get())
infos_test.append(champ_disturbance_yes.get(1.0, tk.END))

#Lukker alle vinduer
window_phase_3.grid_forget()
root.destroy()

##Grensesnitt avslutningsfase (fase 3)
#Dannelse av hovedvindu og geometri
dim_x_4 = 0.417 * screen_x #1600 px en 3840
dim_y_4 = 0.65 * screen_y #1400 px en 2160
window_phase_3 = tk.LabelFrame(root, text='Conclusion', width=dim_x_4,
height=dim_y_4, bd=10, labelanchor='n')
window_phase_3.grid_propagate(0)

#Konfigurasjon av rad og kolonne
n_ligne=5
n_column=2
for i in range(1,n_ligne+1):
    window_phase_3.rowconfigure(i, weight=1)
for j in range(1,n_column+1):
    window_phase_3.columnconfigure(j, weight=2)
    if j==n_column:
        window_phase_3.columnconfigure(j, weight=1)

#Ramme for forstyrrelse ("disturbance")
frame_disturbance = tk.LabelFrame(window_phase_3, padx=50, pady=50)
frame_disturbance.grid(column=1, row=2, columnspan=1, rowspan=1, padx=5, pady=5)
#Knapp for forstyrrelse
#ramme
frame_button_disturbance = tk.LabelFrame(frame_disturbance,
text="Were any disturbances when you took the test?", padx=20, pady=20)
frame_button_disturbance.grid()
#knapp
disturbance = tk.StringVar()
disturbance.set('N')
yes_31 = tk.Radiobutton(frame_button_disturbance, text="Yes",
variable=disturbance, value='Y')
no_31 = tk.Radiobutton(frame_button_disturbance, text="No",
variable=disturbance, value='N')
yes_31.grid()
no_31.grid()
#ramme hvis ja
frame_entry_disturbance = tk.LabelFrame(frame_disturbance,
text="If yes, what was disturbing you?", padx=20, pady=20)
frame_entry_disturbance.grid()
#innhold hvis ja
disturbance_yes = tk.StringVar()
champ_disturbance_yes = tk.Text(frame_entry_disturbance, height=4, width=50,
wrap='word')
champ_disturbance_yes.insert(1.0, 'None')

```

```

champ_disturbance_yes.grid()

#Avslutningsknapp
bouton_next_3 = tk.Button(window_phase_3, text="Finish", command=save_phase_3)
bouton_next_3.grid(column=2,row=4, columnspan=1, rowspan=1,
padx=5, pady=5, sticky='se')

###hovedloop
root.mainloop()

###Skrive ut verdi
print(infos_user, '\n', mark_unpleasantness, '\n', yn_cyh, '\n', infos_test)

###Lagre verdier i CSV
##Funksjon
def reorganisation(infos_user, mark_unpleasantness, yn_cyh, infos_test):
    """Reorganiserer listen som skal lagres i CSV"""
    #Fjerner\n
    for i in range(0, len(infos_test)):
        infos_test[i] = infos_test[i].replace('\n', '')
    #Sorterer lista
    mark_unpleasantness.sort()
    yn_cyh.sort()
    #Lager en ny liste som skal lagres
    mark_unpleasantness_csv = []
    yn_cyh_csv = []
    for i in range(0, len(mark_unpleasantness)):
        mark_unpleasantness_csv.append(mark_unpleasantness[i][1])
    for j in range(0, len(yn_cyh)):
        yn_cyh_csv.append(yn_cyh[j][1])
    return [infos_user + mark_unpleasantness_csv + yn_cyh_csv + infos_test]

###Velge om verdiene skal lagres eller ikke
save_CSV = True

###Navn pa csv
name_of_CSV='Resultat_test.csv'

###Lagrer
if save_CSV:
    #Reorganiser liste
    list_final = reorganisation(infos_user, mark_unpleasantness, yn_cyh, infos_test)
    #apner
    os.chdir(path_CSV)
    f = open(name_of_CSV, 'a')
    wr = csv.writer(f, delimiter = ';', lineterminator = '\n')
    print('File_open')
    #Skriver inn i fila
    wr.writerows(list_final)
    #Lukker filen
    print('File_closed')
    f.close()

##END##

```

G.2 Test 2

Listing G.2: Insert code directly in your document

```
###Brukergrensesnitt###

#Dette skriptet er basert pa et skript utviklet av Jean Fosset.

###Modul Import
import os
import time
import random as rd
import csv as csv
import tkinter as tk
import tkinter.ttk
import tkinter.messagebox
import pygame as py
import numpy as np
from itertools import combinations

###Definerer stier
path_global = 'C:\\Users\\malepc_2\\Desktop\\Aurora'

path_image = path_global + '\\Images'
path_CSV = path_global + '\\Files'
path_sound = path_global + '\\Sons'

os.chdir(path_global)
os.chdir(path_global)

### Initialiserer lydklippene som skal brukes
py.mixer.init() #for pygame

###Definisjon av globale funksjoner
def choice_sound(number_of_sound):
    """Lager en liste med lydklipp og blander den i tilfeldig rekkefølge"""
    list_sound = []
    path = path_sound + '\\Sounds_test2'
    os.chdir(path)
    #Lydklippene har navn som : "Sound_1", "Sound_2"...
    for j in range(1, number_of_sound + 1):
        nom_fichier_son='sound_'+str(j)+'.wav' #open file .wav
        list_sound.append([nom_fichier_son,py.mixer.Sound(nom_fichier_son)])
    #Legger til det nye lydklippet i lista

    rd.shuffle(list_sound) #Blander lista
    return list_sound

#Lager en liste som setter sammen to og to tall (1-2, 1-3, 1-4, etc.
randomlist = []
def getlist(number_of_sounds):
    l = range(0,(number_of_sounds))
    r=2
    comb=combinations(l,r)
```

```

    randomlist=list(comb)
    return randomlist

#Funksjon som tilfeldig gir ut 0 eller 1,
#brukes for a avgjore om rekkefølgen skal byttes om eller ikke

def flipcoin():
    coin = (0,1)
    flip=rd.choice(coin)
    return flip

###Definition of globale variabler
#Definerer vinduet
screen_x = 3840
screen_y = 2160

#For a lagre informasjon
infos_user = [] #users informations
infos_test = [] #test using informations
AB_cyh = [] #list with the result of the test phase 2
list_points_sound = {'sound_1.wav':0, 'sound_2.wav':0, 'sound_3.wav':0, 'sound_4.wav':0,
'sound_5.wav':0, 'sound_6.wav':0, 'sound_7.wav':0, 'sound_8.wav':0, 'sound_9.wav':0,
'sound_10.wav':0, 'sound_11.wav':0, 'sound_12.wav':0, 'sound_13.wav':0, 'sound_14.wav':0,
'sound_15.wav':0, 'sound_16.wav':0, 'sound_17.wav':0, 'sound_18.wav':0 }

#For testfasen
index_number_of_test_2 = 0 #number of test play in phase 2
number_of_sounds = 18 #number of sound to be compared
number_of_test_2 = 153 #number of comparisons in phase 2
test_2 = True
max_progressbar_2 = number_of_test_2 + 0.0001
is_switched= False

#Global
waiting_before_answer = 1
#Testfase
sound_test = choice_sound(number_of_sounds) #list of sound using for test in phase
randomlist = getlist(number_of_sounds) #copy the values from random list

print(list(randomlist))
print(list(sound_test))

#Liste av lydeksempler
sound_time_2=7 #Time of sound in seconds
max_sound_time_2 = sound_time_2 + 0.0001
#Definerer maks progresjon for lyd bar

###Hovedprogram – grafisk brukergrensesnitt###
##Lager hovedvindu
root = tk.Tk()
root.attributes('-fullscreen',1)
# root_geo = str(screen_x) + 'x' + str(screen_y) + '+0+0'
# root.geometry(root_geo)

```

```

##Bilde
#Konfigurerer rad og kolonne
n_ligne=3
n_column=3
for i in range(1,n_ligne+1):
    root.rowconfigure(i, weight=1)
for j in range(1,n_column+1):
    root.columnconfigure(j, weight=1)
    if j == n_column:
        root.columnconfigure(j, weight=1)

#Canvas logo NTNU
canvas_logo = tk.Canvas(root, width=688, height=176)
#Apner bilde og setter inn i canvas
os.chdir(path_image)
logo_ntnu = tk.PhotoImage(file='NTNU_logo.png')
canvas_logo.create_image(688/2, 176/2, image = logo_ntnu)
canvas_logo.grid(row=1, column=1, sticky='nw', padx=20, pady=20)

canvas_useless = tk.Canvas(root, width=688, height=176)
canvas_useless.grid(row=3, column=3)

###Fase1
##Funksjoner ved bruk av knapp
def save_phase_1():
    """Lagrer bruker informasjon i lisen "infos_user"
    og lukker f[U+FFFD]vindu"""
    #saving information
    infos_user.append(field_subject.get())
    infos_user.append(field_age.get())
    infos_user.append(gender.get())
    infos_user.append(who_you_are.get())
    infos_user.append(prev_exp.get())

    #close the windows
    window_phase_1.grid_remove()

    #new challenger approaching
    window_phase_2.grid(row=2, column=2)

    #updating
    root.update()

##Grensesnitt
#Lager hovedvindu og dimensjon
dim_x_1 = 0.26 * screen_x #1000 px en 3840
dim_y_1 = 0.46 * screen_y #1000 px en 2160
window_phase_1 = tk.LabelFrame(root, text='About_you',
width=dim_x_1, height=dim_y_1, labelanchor='n', bd=10)
window_phase_1.grid_propagate(0)

#Konfigurerer rad/kolonne
n_ligne=8

```

```

n_column=3
for i in range(1,n_ligne+1):
    window_phase_1.rowconfigure(i, weight=1)
for j in range(1,n_column+1):
    window_phase_1.columnconfigure(j, weight=2)
    if j == n_column:
        window_phase_1.columnconfigure(j, weight=1)

#Tekst forklaring
label_explication = tk.Label(window_phase_1, text="Welcome to the test!
Please enter necessary personal information before starting the listening test.",
fg='black')
label_explication.grid(column=1, row=1, columnspan=2, rowspan=2, padx=5, pady=5)

#Merke for subjektnummer
frame_subject_number = tk.LabelFrame(window_phase_1, text="Subject number",
padx=20, pady=20)
frame_subject_number.grid(column=1, row=3, columnspan=1, rowspan=1, padx=5, pady=5)
#Zone subject
field_subject = tk.Entry(frame_subject_number, width = 4)
field_subject.grid()

#Merke for alder
frame_age = tk.LabelFrame(window_phase_1, text="Age", padx=20, pady=20)
frame_age.grid(column=2, row=3, columnspan=1, rowspan=1, padx=5, pady=5)
#Sone for alder
field_age = tk.Entry(frame_age, width=3)
field_age.grid()

frame_gender = tk.LabelFrame(window_phase_1, text="Gender", padx=20, pady=20)
frame_gender.grid(column=1, row=4, columnspan=1, rowspan=1, padx=5, pady=5)
#Knapp valg av kjonn
gender = tk.StringVar()
gender.set('M') #With this, Male is checked by default
male = tk.Radiobutton(frame_gender, text="Male", variable=gender, value='M')
female = tk.Radiobutton(frame_gender, text="Female", variable=gender, value='F')
others = tk.Radiobutton(frame_gender, text="Others", variable=gender, value='O')
male.grid(sticky='w')
female.grid(sticky='w')
others.grid(sticky='w')

#Merke for hvem man er
frame_wya = tk.LabelFrame(window_phase_1, text="I am a", padx=20, pady=20)
frame_wya.grid(column=2, row=4, columnspan=1, rowspan=1, padx=5, pady=5)
#Knapp for hvem man er
who_you_are = tk.StringVar()
who_you_are.set('S') #with this, Student is checked by default
student = tk.Radiobutton(frame_wya, text="Student", variable=who_you_are, value='S')
ntnu_employee = tk.Radiobutton(frame_wya, text="NTNU employee", variable=who_you_are,
value='NE')
others_wya = tk.Radiobutton(frame_wya, text="Others", variable=who_you_are, value='O')
student.grid(sticky='w')
ntnu_employee.grid(sticky='w')
others_wya.grid(sticky='w')

```



```

#Tekst: tidligere erfaring
frame_previous = tk.LabelFrame(window_phase_1, text="Do_you_have_previous_experience
with_wind_turbine_noise?", padx=20, pady=20)
frame_previous.grid(column=1,row=7,columnspan=2,rowspan=1, padx=5, pady=5, sticky='')
#Tidligere erfaring
prev_exp=tk.StringVar()
prev_exp.set('Y') #with this, yes is checked by default
yes_11 = tk.Radiobutton(frame_previous, text="Yes", variable=prev_exp, value='Y')
no_11 = tk.Radiobutton(frame_previous, text="No", variable=prev_exp, value='N')
yes_11.grid()
no_11.grid()

#Neste knapp
next_button_1 = tk.Button(window_phase_1, text="Next", command=save_phase_1)
next_button_1.grid(column=3, row=8, columnspan=1, rowspan=1,
padx=5, pady=5,sticky='se')

#Utstilling
window_phase_1.grid(row=2, column=2)

###Testfase###
##Funksjoner ved bruk av knapp
def debut_test_2():
    """Start phase 2"""
    bouton_start_2['state'] = 'disable'
    bouton_playsound_2['state'] = 'normal'

def play_sound_2():
    """Play the sound in phase 2"""

#deaktivere knapp
bouton_playsound_2['state'] = 'disable'

if test_2 : #Bekrefter at det er testfase
    #Spill av lyd
    global is_switched
    flip = flipcoin()
    if flip == 0: #Switch order
        sonA = sound_test[randomlist[index_number_of_test_2][1]][1]
        sonB = sound_test[randomlist[index_number_of_test_2][0]][1]
        is_switched = 1
        print('order_switched')
        print( randomlist[index_number_of_test_2][1],
        randomlist[index_number_of_test_2][0])
        print(sound_test[randomlist[index_number_of_test_2][1]][0],
        sound_test[randomlist[index_number_of_test_2][0]][0])
    else: #Beholde rekkefølgen
        sonA = sound_test[randomlist[index_number_of_test_2][0]][1]
        sonB = sound_test[randomlist[index_number_of_test_2][1]][1]
        is_switched = 0
        print('order_the_same')
        print( randomlist[index_number_of_test_2][0],
        randomlist[index_number_of_test_2][1])

```

```

        print(sound_test[randomlist[index_number_of_test_2][0]][0],
              sound_test[randomlist[index_number_of_test_2][1]][0])
    #Spill av lydene
    sonA.play()
    #Progresjon lydbar
    for j in range(0, sound_time_2):
        time.sleep(1)
        progressbar_sound_1.step(1)
        window_phase_2.update_idletasks()

    time.sleep(3)

    sonB.play()
    #Progresjon lydbar
    for j in range(0, sound_time_2):
        time.sleep(1)
        progressbar_sound_2.step(1)
        window_phase_2.update_idletasks()

    #Reset for progresjon av lyden
    value_progresssound_1.set(0)
    value_progresssound_2.set(0)

    #Ventetid for svar
    time.sleep(waiting_before_answer)

    #Aktivere knapp for a svare
    buttonA['state'] = 'normal'
    buttonB['state'] = 'normal'
    buttonSame['state'] = 'normal'
    bouton_yn_choice['state'] = 'normal'

def validation_2():
    """Bekreftede og lagre verdien til A, B eller 0"""
    global text_progression_progressbar_2

    #resetter alle knapper til normaltilstand
    bouton_playsound_2['state'] = 'normal'
    buttonA['state'] = 'disable'
    buttonB['state'] = 'disable'
    buttonSame['state'] = 'disable'
    bouton_yn_choice['state'] = 'disable'

    #Progresjons bar
    progressbar_2.step(1)

    if test_2: #Sjekker at det er testfase
        global index_number_of_test_2
        global is_switched
        value = cyh.get()

        #Lagre verdien
        if value == 'A':
            if is_switched:
                print(is_switched)

```

```

        list_points_sound[ sound_test[ randomlist[ index_number_of_test_2 ][1]][0]]
        +=1

    else:
        list_points_sound[ sound_test[ randomlist[ index_number_of_test_2 ][0]][0]]
        +=1

        print(is_switched)
    elif value == 'B':
        if is_switched:
            list_points_sound[ sound_test[ randomlist[ index_number_of_test_2 ][0]][0]]
            +=1

        else:
            list_points_sound[ sound_test[ randomlist[ index_number_of_test_2 ][1]][0]]
            +=1

cyh.set(0) #reset

#tekst progresjonsbar
index_number_of_test_2 += 1
print('This is the ', index_number_of_test_2 , ' example.')
text_progression_progressbar_2.set(str(index_number_of_test_2) + '/' +
str(number_of_test_2))
window_phase_2.update_idletasks()

if index_number_of_test_2 == number_of_test_2:
#Dersom det er det sistelydeksempel, deaktiveres alle knapper og
#neste knappen blir aktiv
    bouton_playsound_2['state'] = 'disable'
    buttonA['state'] = 'disable'
    buttonB['state'] = 'disable'
    buttonSame['state'] = 'disable'
    bouton_yn_choice['state'] = 'disable'

#info
    tk.messagebox.showinfo(title = 'Test_end', message = 'The test was finished.
    Please click on next to continue to some finishing questions.')

    bouton_next_2['state'] = 'normal'

def fin_phase_2():
    """Avslutter testfase"""
    #Lukker vinduet
    window_phase_2.grid_remove()

    #Nytt vindu
    window_phase_3.grid(row=2, column=2)

    #Oppdaterer
    root.update()

##Grensesnitt

```

```

#Lager hovedvindu og geometri
dim_x_2 = 0.417 * screen_x #1600 px en 3840
dim_y_2 = 0.555 * screen_y #1200 px en 2160
window_phase_2 = tk.LabelFrame(root, text='Compare the sounds',
width = dim_x_2, height = dim_y_2, bd=10, labelanchor='n')
window_phase_2.grid_propagate(0)

#Konfigurerer rad og kolonne
n_ligne=8
n_column=2
for i in range(1,n_ligne+1):
    window_phase_2.rowconfigure(i, weight=1)
for j in range(1,n_column+1):
    window_phase_2.columnconfigure(j, weight=2)
    if j==n_column:
        window_phase_2.columnconfigure(j, weight=1)

#Definierer variabeltekst for
#forklaring og startknapp og progresjonsbar
text_explication_2 = tk.StringVar()
text_button_start_2 = tk.StringVar()
text_progression_progressbar_2 = tk.StringVar()

if test_2:
    text_explication_2.set('Listen to the sounds. Which of the two sounds A and B
    is most annoying? \n\n Press start to begin the test \n\n TEST \n')
    text_button_start_2.set('Start')
    text_progression_progressbar_2.set('0' + '/' + str(number_of_test_2))

#Forklaringstekst
label_instruc = tk.Label(window_phase_2, textvariable=text_explication_2, fg='black')
label_instruc.grid(column=1, row=1, columnspan=1, rowspan=1, padx=5, pady=5)

#Start knapp
bouton_start_2 = tk.Button(window_phase_2, textvariable=text_button_start_2,
command=debut_test_2)
bouton_start_2.grid(column=2, row=1, columnspan=1, rowspan=1, padx=5, pady=5)

#Rammer inn all prosedyre
frame_procedure_2 = tk.LabelFrame(window_phase_2, padx=20, pady=20, relief=tk.GROOVE)
frame_procedure_2.grid(column=1, row=3, columnspan=1, rowspan=1, padx=5, pady=5)

#Lydknapp
#Ramme
frame_played_2 = tk.LabelFrame(frame_procedure_2, text="1. Press play to listen",
padx=20, pady=20)
frame_played_2.grid(column=1, row=3, columnspan=1, rowspan=1, padx=5, pady=5)
#knapp
bouton_playsound_2 = tk.Button(frame_played_2, text="Play", command=play_sound_2,
state='disabled')
bouton_playsound_2.grid()

#Progressjonsbar for lyd
#ramme
frame_progressbar_sound_1 = tk.LabelFrame(frame_procedure_2,
text="Progress Sound A", padx=20, pady=20)

```

```

frame_progressbar_sound_1.grid(column=2, row=3, columnspan=1, rowspan=1,
padx=5, pady=5)

frame_progressbar_sound_2 = tk.LabelFrame(frame_procedure_2, text="Progress_Sound_B",
padx=20, pady=20)
frame_progressbar_sound_2.grid(column=2, row=5, columnspan=1, rowspan=1,
padx=5, pady=5)

#knapp
value_progresssound_1 = tk.StringVar()
progressbar_sound_1 = tk.ttk.Progressbar(frame_progressbar_sound_1,
orient='horizontal',
length=500, mode='determinate', maximum=max_sound_time_2,
variable=value_progresssound_1)
progressbar_sound_1.grid()

value_progresssound_2 = tk.StringVar()
progressbar_sound_2 = tk.ttk.Progressbar(frame_progressbar_sound_2,
orient='horizontal',
length=500, mode='determinate', maximum=max_sound_time_2,
variable=value_progresssound_2)
progressbar_sound_2.grid()

#Valgknapp
#ramme
frame_choice_2 = tk.LabelFrame(frame_procedure_2, text="2. Which of the sound samples
is most annoying?", padx=20, pady=20)
frame_choice_2.grid(column=1, row=4, columnspan=1, rowspan=1,
padx=5, pady=5)
#knapp
cyh = tk.StringVar()

buttonA = tk.Radiobutton(frame_choice_2, text="A", variable=cyh, value='A',
state='disable')
buttonB = tk.Radiobutton(frame_choice_2, text="B", variable=cyh, value='B',
state='disable')
buttonSame = tk.Radiobutton(frame_choice_2, text="The same", variable=cyh, value = 0,
state='disable')
buttonA.grid()
buttonB.grid()
buttonSame.grid()

#knapp for a bekrefte
#ramme
frame_confirm_2 = tk.LabelFrame(frame_procedure_2, text="3. Confirm", padx=20, pady=20)
frame_confirm_2.grid(column=1, row=5, columnspan=1, rowspan=1, padx=5, pady=5)
#knapp
bouton_yn_choice = tk.Button(frame_confirm_2, text="Confirm", command=validation_2,
state='disable')
bouton_yn_choice.grid()

#Progresjonsbar for testen
#ramme
frame_progressbar_2 = tk.LabelFrame(window_phase_2, text="Progress_bar_test",
padx=20, pady=20)
frame_progressbar_2.grid(column=1, row=6, columnspan=1, rowspan=1, padx=5, pady=5)

```

```

#knapp
value_progress_2 = tk.StringVar()
progressbar_2 = tk.ttk.Progressbar(frame_progressbar_2, orient='horizontal',
length=500,
mode='determinate', maximum=max_progressbar_2, variable=value_progress_2)
progressbar_2.grid()

#Tekst progresjonsbar
label_progressbar_2 = tk.Label(frame_progressbar_2,
textvariable=text_progression_progressbar_2, fg='red')
label_progressbar_2.grid()

#TEST KNAPP
button_2 = tk.Button(window_phase_2, text="Quick_quit", command=fin_phase_2)
button_2.grid(column=1, row=8, columnspan=1, rowspan=1, padx=5, pady=5, sticky='sw')

#Neste knapp
bouton_next_2 = tk.Button(window_phase_2, text="Next", command=fin_phase_2,
state='disable')
bouton_next_2.grid(column=2, row=8, columnspan=1, rowspan=1, padx=5, pady=5, sticky='se')

###Fase 3, avsluttende fase###
##funksjoner ved bruk av knapp
def save_phase_3():
    """Save all information about the test"""
    #saving information
    infos_test.append(disturbance.get())
    infos_test.append(champ_disturbance_yes.get(1.0, tk.END))

    #close the windows and all windgets
    window_phase_3.grid_forget()
    root.destroy()

##Grensesnitt fase 3
#Lager hovedvindu og geometri
dim_x_4 = 0.417 * screen_x #1600 px en 3840
dim_y_4 = 0.65 * screen_y #1400 px en 2160
window_phase_3 = tk.LabelFrame(root, text='Finishing_questions', width=dim_x_4,
height=dim_y_4, bd=10, labelanchor='n')
window_phase_3.grid_propagate(0)

#Konfigurerer rad og kolonne
n_ligne=5
n_column=2
for i in range(1,n_ligne+1):
    window_phase_3.rowconfigure(i, weight=1)
for j in range(1,n_column+1):
    window_phase_3.columnconfigure(j, weight=2)
    if j==n_column:
        window_phase_3.columnconfigure(j, weight=1)

#Ramme for forstyrrelser og "yes"
frame_disturbance = tk.LabelFrame(window_phase_3, padx=50, pady=50)

```

```

frame_disturbance.grid(column=1, row=2, columnspan=1, rowspan=1, padx=5, pady=5)
#Knapp for forstyrrelse
#ramme
frame_button_disturbance = tk.LabelFrame(frame_disturbance,
text="Was there any disturbances when you took the test?", padx=20, pady=20)
frame_button_disturbance.grid()

#knapp
disturbance = tk.StringVar()
disturbance.set('N')
yes_31 = tk.Radiobutton(frame_button_disturbance, text="Yes",
variable=disturbance, value='Y')
no_31 = tk.Radiobutton(frame_button_disturbance, text="No",
variable=disturbance, value='N')
yes_31.grid()
no_31.grid()
#ramme hvis yes
frame_entry_disturbance = tk.LabelFrame(frame_disturbance,
text="If yes, what was disturbing you?", padx=20, pady=20)
frame_entry_disturbance.grid()
#Zone hvis forstyrrelse
disturbance_yes = tk.StringVar()
champ_disturbance_yes = tk.Text(frame_entry_disturbance, height=4, width=50,
wrap='word')
champ_disturbance_yes.insert(1.0, 'None')
champ_disturbance_yes.grid()

#Avslutning knapp
bouton_next_3 = tk.Button(window_phase_3, text="Finish", command=save_phase_3)
bouton_next_3.grid(column=2, row=4, columnspan=1, rowspan=1, padx=5, pady=5,
sticky='se')

###Loop of root
root.mainloop()

###Skriver ut verdi
print(infos_user, '\n', AB_cyh, '\n', infos_test)

###Lagrer verdier i CSV
##Funksjon
def reorganisation(infos_user, _list_points_sound, infos_test):
    """Reorganise list to save in CSV"""
    #Fjerner \n fra teksten
    for i in range(0, len(infos_test)):
        infos_test[i] = infos_test[i].replace('\n', '')
    #Sorterer liste
    #Lager en ny liste
    AB_cyh_csv1 = []
    for j in range(1, 18):
        AB_cyh_csv1.append(list_points_sound['sound_' + str(j) + '.wav'])
    return [infos_user + AB_cyh_csv1 + infos_test]

###Velg verdier skal lagres eller ikke
save_CSV = True

```

```

### tittel pa CSV
name_of_CSV='Resultat_test2.csv'

###Lagre
if save_CSV:
    #Reorganise liste
    list_final = reorganisation(infos_user , list_points_sound , infos_test)
    #Apne
    os.chdir(path_CSV)
    f = open(name_of_CSV, 'a')
    wr = csv.writer(f, delimiter = ';', lineterminator = '\n')
    print('File_open')
    #Skrive inn i CSV fila
    wr.writerow(list_final)
    print('File_closed')
    f.close()

##END##

```

Tillegg **H**

Bilder av brukergrensesnitt

About you

Welcome to the test! Please enter necessary personal information before starting the listening test.

Subject number

Age

Gender

☒ Male

☐ Female

☐ Others

I am a

☒ Student

☐ NTNU employee

☐ Others

Do you have previous experience with wind turbine noise ?

☒ Yes

☐ No

Next

Figur H.1: Bilde av brukergrensesnitt første del.

Listening test

Imagine that you live nearby a wind farm and
that sometimes when you are outside
or have a window open you hear this sound.
How disturbing would you experience the sound to be on a scale from 0 - 10 ?
0 indicate "not at all" while 10 indicate "extremely disturbing".

Start

1. Play to listen

Play

Progress Sound Bar

2. Mark your unpleasantness

Unpleasantness

0.0

0.0

10.0

3. Confirm the value

Confirm

Progressbar

0/45

Figur H.2: Bilde av brukergrensesnitt hoveddel test 1.

Compare the sounds

Listen to the sounds. Which of the two sounds A and B is most annoying?

Press start to begin the test

TEST

Start

1. Press play to listen

Play

Progress Sound A

2. Which of the sound samples is most annoying ?

☐ A

☐ B

☐ The same

3. Confirm

Confirm

Progress Sound B

Progress bar test

0/153

Figur H.3: Bilde av brukergrensesnitt hoveddel test 2.

Finishing questions

Was there any disturbances when you took the test ?

☐ Yes

☒ No

If yes, what was disturbing you ?

None

Finish

Figur H.4: Bilde av brukergrensesnitt avsluttende spørsmål.

Måling bakgrunnsstøy

I.1 Test 1

Kandidatnummer	μ , [dBA]	σ	95% konfidensintervall
1	22,02	5,45	[21,70, 22,52]
2	21,26	5,15	[21,03, 21,50]
3	22,98	6,98	[22,63, 23,32]
4	21,85	5,74	[21,59, 22,10]
5	23,97	7,68	[23,62, 24,33]
6	20,36	4,80	[20,15, 20,58]
7	22,18	5,08	[21,93, 22,42]
8	22,42	6,30	[22,12, 22,73]
9	20,64	5,97	[20,34, 20,93]
10	23,26	6,62	[23,01, 23,50]
11	23,57	6,69	[23,26, 23,89]
12	22,58	5,61	[22,28, 22,89]
13	21,70	3,39	[21,54, 21,85]
14	21,97	2,92	[21,82, 22,12]
15	22,29	4,15	[22,09, 22,49]
16	21,18	3,40	[21,02, 21,34]
17	22,11	5,32	[21,87, 22,35]
18	20,61	3,27	[20,46, 20,76]
19	22,97	4,83	[22,76, 23,18]
20	21,55	3,39	[21,39, 21,71]
21	23,48	5,29	[23,24, 23,72]
22	22,83	4,47	[22,63, 23,04]
23	22,79	4,22	[22,59, 22,98]
24	22,52	4,75	[22,30, 22,74]
25	23,64	6,77	[23,33, 23,94]
26	23,37	6,04	[23,10, 23,64]
27	22,11	4,50	[21,92, 22,31]
28	23,84	6,34	[23,54, 23,14]
29	22,27	5,45	[22,03, 22,52]
30	-	-	-
Samlet	22,38	5,48	[22,33, 22,43]

Tabell I.1: Måling av bakgrunnsstøy test 1

I.2 Test 2

Kandidatnummer	μ , [dBA]	σ	95% konfidensintervall
1	23,07	6,80	[22,84, 23,30]
2	22,37	4,70	[22,21, 22,54]
3	21,23	3,92	[21,08,21,36]
4	23,91	7,24	[23,64, 24,17]
5	23,44	5,67	[23,24, 24,64]
6	26,21	8,50	[25,89, 26,52]
7	22,68	5,89	[22,48, 22,87]
8	23,38	6,24	[23,16, 23,60]
9	22,13	6,13	[21,92, 22, 34]
10	24,17	6,33	[23,94, 24,39]
11	23,46	7,62	[23,20, 23,70]
12	24,36	6,48	[24,16, 24, 57]
13	23,34	5,76	[23,15, 23,54]
14	21,21	4,11	[21,15, 21,47]
15	23,24	5,44	[23,06, 23,43]
16	21,26	5,89	[21,07, 21,46]
17	24,16	7,52	[23,91, 24,40]
18	23,88	8,75	[23,61, 24,16]
19	24,63	5,89	[24,43, 24,83]
20	22,85	6,70	[22,62, 23,08]
21	24,99	8,29	[24,71, 25,28]
22	25,39	5,27	[25,20, 25,69]
23	22,12	5,05	[21,95, 22,29]
24	22,68	5,21	[20,85, 21,21]
25	23,29	5,92	[23,09, 23,49]
26	20,49	4,68	[20,34, 20,65]
27	25,18	7,12	[24,93, 25,42]
28	24,23	6,98	[23,98, 24,47]
29	21,03	5,21	[20,85, 21,21]
30	-	-	-
Samlet	23,24	6,51	[23,20, 23,28]

Tabell I.2: Måling av bakgrunnsstøy test 1

Resultat lyttetester

J.1 Test 1

Lyd- eksempel	Gjennitt	Min	Maks	Differanse min-maks	Median
1	0,85	0	3,5	3,5	0,50
2	1,48	0	5,5	5,5	1,25
3	1,27	0	4,5	4,5	1,0
4	1,68	0	5	5	1,50
5	1,43	0	6,5	6,5	1,0
6	2,78	0	7,5	7,5	2,75
7	3,73	0	8,5	8,5	4,0
8	3,68	0	9	9	3,5
9	4,03	0,5	9,5	9	4,0
10	4,03	1	9,5	8,5	4,0
11	4,97	1	9,5	8,5	5,0
12	5,72	1	10	9	5,25
13	6,03	1	10	9	6,5
14	6,98	1	10	9	7,0
15	7,12	2,5	10	7,5	8,0

Lyd- eksempel	Gjennitt	Min	Maks	Differanse min-maks	Median
16	0,80	0	3,5	3,5	0,5
17	1,10	0	4	4	0,5
18	1,60	0	4,5	4,5	1,5
19	1,47	0	4	4	1,0
20	1,78	0	6	6	1,0
21	2,32	0	5,5	5,5	1,75
22	3,13	0	6,5	6,5	3,0
23	3,53	0	8,5	8,5	3,0
24	3,95	0	9,5	9,5	3,5
25	3,73	0	9,5	9,5	3,75
26	4,08	0,5	10	9,5	4,25
27	5,28	0	10	10	6,25
28	6,63	1,5	10	8,5	7,0
29	6,60	1,5	10	8,5	7,0
30	6,42	0,5	10	9,5	6,0
31	0,28	0	2	2	0,0
32	0,65	0	3	3	0,5
33	1,43	0	7	7	0,75
34	1,80	0	7	7	1,5
35	1,72	0	6,5	6,5	1,5
36	1,48	0	6	6	1,0
37	2,20	0	6,5	6,5	1,75
38	3,07	0	8,5	8,5	2,0
39	3,98	0	10	10	4,0
40	3,65	0	8	8	4,0
41	2,90	0	7,5	7,5	3,0
42	4,20	1	9	8	4,0
43	5,48	0	9	9	6,0
44	6,37	1	10	9	6,75
45	6,73	2	10	8	7,0

Tabell J.1: Resultat test 1

J.2 Test 2

Lyd- eksempel	Gjennom- snitt	Min	Maks	Differanse min-maks	Median
1	3,06	0	7	7	3
2	2,77	0	7	7	3
3	1,90	0	7	7	1
4	6,94	0	11	11	7
5	7,65	0	12	12	8
6	7,87	0	11	11	8
7	11,58	0	16	16	12
8	13,23	0	16	16	14
9	12,94	0	16	16	14
10	0,65	0	5	5	0
11	1,16	0	6	6	1
12	1,42	0	4	4	1
13	3,00	0	7	7	3
14	5,61	0	11	11	6
15	6,90	3	10	7	7
16	7,68	0	12	12	8
17	11,61	1	17	16	12
18	-	-	-	-	-

Tabell J.2: Resultat test 2