IDEA League

Master of Science in Applied Geophysics

Research Thesis

Cryoseismic Event Analysis on Distributed Strain Recordings Leveraging Statistical Learning Methods

Julius Carl Tilemann Grimm

August 6, 2021

Cryoseismic Event Analysis on Distributed Strain Recordings Leveraging Statistical Learning Methods

Master of Science Thesis

for the degree of Master of Science in Applied Geophysics at
Delft University of Technology
ETH Zürich
RWTH Aachen University
by

Julius Carl Tilemann Grimm

August 6, 2021

Department of Geoscience & Engineering · Delft University of Technology
Department of Earth Sciences · ETH Zürich
Faculty of Georesources and Material Engineering · RWTH Aachen University

Delft University of Technology

Printed in The Netherlands, Switzerland, Germany

IDEA LEAGUE
JOINT MASTER'S IN APPLIED GEOPHYSICS

Delft University of Technology, The Netherlands
ETH Zürich, Switzerland
RWTH Aachen, Germany

Dated: *August 6, 2021*

Supervisors / Committee Members:

Dr. Patrick Paitz

Dr. Eileen Martin

Dr. Pascal Edme

Prof. Dr. Fabian Walter

Prof. Dr. Andreas Fichtner

External Committee Member:

Prof. Dr. Florian Wellmann

Abstract

In the summer of 2020, ETH researchers installed a 9 km long fiber-optic cable on the Rhonegletscher (Switzerland), covering the whole glacier extent from accumulation to ablation zone. The fiber was then interrogated with a DAS system, turning it into a distributed seismic antenna. The DAS system recorded continuously for one month on more than 2000 channels at 1 kHz sampling rate. The large data volume (~18 TB) renders manual event picking and categorization practically unfeasible. For such DAS monitoring experiments, automatic event detection and classification might become indispensable.
Different types of icequakes and meteorological effects are visible on the raw strain-rate recordings. Most cryoseismic events (stick-slip icequakes and surface crevassing) are of short-time duration and small spatial extent. Since most events are only visible on fractions of the array, the data is divided into smaller sub-windows in time and space.
Different array processing techniques are discussed to obtain a low-dimensional feature representation of each sub-window. The output from the array processing is used as input for an unsupervised clustering algorithm assigning a class-membership to each sub-window. This is applied to parts of the Rhonegletscher dataset. Meaningful clusters are returned that correspond to seismic events or noise.
The analysis yields a preliminary overview of signal types contained in the Rhonegletscher dataset and their spatio-temporal distribution. This method can be used to obtain an event catalogue of clustered signals. This could be used as basis for further analyses such as seismic imaging, template matching or supervised machine learning.

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Cryoseismic Event Analysis on Distributed Strain Recordings Leveraging Statistical Learning Methods

Authored by (in block letters):

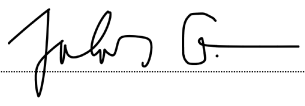
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
Grimm	Julius Carl Tilemann

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**
Zürich, 6.8.2021	

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.

Acknowledgements

First of all I want to thank all the people who have participated in this project. I had the privilege of working together with multiple supervisors and be part of multiple subgroups.
I want to thank Patrick for all the freedom I had with the project and the continuous great support. It has been fun working with you and you always had excellent advice. Same applies to Eileen. My meetings with you and Patrick always were insightful and I always left with many ideas and a good mood. I hope we will finally meet one day on a conference.
Special thanks goes to Pascal, for his support during the thesis project. I learned a lot about signal processing and generally appreciate your direct and precise comments. I am also grateful for Andreas, who created this inspiring work environment. And thank you for helping me prepare for my PhD interview.
I want to give a shout out to Eileen's working group at Virginia Tech. The weekly meetings were always interesting and it has been a joy to listen to students from other fields. The observational seismology group at ETH also was a great source of inspiration. It's really fascinating to see what everybody is working on.
Last but not least, I want to thank all the people with whom I shared this awesome journey these last two years, many of whom have become close friends. Special mentions to my roommate Nicholas, my office mates Tjeerd and Kevin, my fellow Mets fan Daniel, my music compatriot Roos, my former boss Dario and the legendary Eddy. I can't wait to see what the future brings for us!

Swiss Federal Institute of Technology — Julius Carl Tilemann Grimm
August 6, 2021

Table of Contents

List of Figures

Acronyms

AGC Automatic Gain Control

AWS Amazon Web Services

CLI Command Line Interface

CNN Convolutional Neural Network

COTDR Coherent Optical Time-Domain Reflectometry

DAS Distributed Acoustic Sensing

EBS Elastic Block Storage

EC2 Elastic Compute Cloud

HDF Hierarchical Data Format

IU Interrogator Unit

ML Machine Learning

PWS Phase-Weighted Stack

PSD Power Spectral Density

SNR Signal-To-Noise Ratio

STA/LTA Short-Term Average/Long-Term Average

S3 Simple Storage Service

Chapter 1

Introduction

This chapter serves as a general introduction to the thesis topic and motivates the research. The field of cryoseismology is briefly introduced and the basic principles of DAS are explained. An overview of related work is presented and all following chapters are introduced.

1-1 Motivation

Glaciers are of great scientific and social interest as they are important indicators for global climate fluctuations [Haeberli et al., 2007]. Furthermore, they constitute a valuable economical resource in mountainous regions. Glacial meltwater serves as a fresh water supply to local communities, but can also be utilized for generating hydroelectric power. Alpine natural hazards are often linked to glacier dynamics [GLAMOS, 2017]. Increased mass loss of glaciers can destabilize mountain slopes, potentially causing rock or ice avalanches. Where glacial lakes are present, glacier receding may increase the risk of glacial outburst floods [Harrison et al., 2018]. As a consequence, monitoring of glacier changes and dynamics with glaciological and geophysical methods is a topical field of research.
Seismological methods help to detect and characterize dynamic glacier processes such as surface crevasse formation, glacier sliding, ice flow [Podolskiy and Walter, 2016] and glacial erosion [Gimbert et al., 2016]. To accurately monitor these processes, ground motion data with high resolution in time and space is required [Jousset et al., 2018]. Dense arrays are crucial for accurately recording spatial variations of waveforms. These lead to improved estimations of seismic source locations and a higher imaging resolution of subsurface features [Zhan, 2019]. Traditional instruments record ground motion with high fidelity but the data is often spatially aliased. This is because traditional seismic networks often exhibit a low coverage density, largely owed to the high cost of individual sensors and typically large areas of investigation. In addition, installation and maintenance of large seismic networks and arrays are labor-intensive and can be challenging, especially in rough terrain and under harsh climate conditions. These limitations can be overcome by Distributed Acoustic Sensing (DAS), which repurposes a fiber-optic cable as a distributed strain sensor [Walter et al., 2020]. This allows

to measure the strain-rate along the fiber with a high spatial sampling rate (<25 cm possible, [Parker et al., 2014]) and over large apertures. Acoustic and elastic waves propagating along the cable can be recorded by DAS. Compared to conventional geophone chains, installation of DAS systems is straightforward, which makes it an ideal monitoring tool at study sites with difficult access such as glaciers.
In spring 2019, ETH researchers deployed a 1 km long fiber-optic cable on the surface of the Rhonegletscher (Switzerland) and collected DAS data continuously for one week. The DAS system recorded surface and basal events, as well as a rock fall. Since the cable was installed in a triangular configuration, estimating the locations of events with beamforming techniques was possible. Furthermore, the shallow subsurface Rayleigh wave velocity was determined by cross-correlations of ambient noise [Walter et al., 2020]. However, the experiment only covered a small area of the glacier on the ablation zone. Therefore, a follow-up experiment was conducted in the summer of 2020. A 9 km long fiber-optic cable was installed on the Rhonegletscher surface, running from the accumulation zone to the tongue of the glacier. The DAS system recorded continuously for one month. For most of the experiment, the strain-rate was sampled every 4 m along the cable with a temporal sampling frequency of 1 kHz. The resulting dataset has a size of about 18 TB. The huge data volume renders manual picking and characterization of events practically infeasible. To efficiently process data of DAS monitoring experiments, new processing tools are required. Machine Learning (ML), also known as statistical learning, has been successfully used for automated noise clustering on DAS recordings [Martin et al., 2018]. Potentially, this approach cannot only be applied to detect noise but also signals of interest. The main part of this work combines array processing techniques and unsupervised clustering to detect and categorize cryoseismic events on the new Rhonegletscher dataset.

1-2 Cryoseismology

Cryoseismology refers to the study of ice or ice-rich Earth materials, structures and processes through passive or active seismology [Aster and Winberry, 2017]. This includes the field of glacial seismology. Glaciers are elastic structures that are governed by complex dynamic processes. Various glacier processes release energy, part of which propagates as seismic waves through the ice and the underlying bedrock/sediments. In consequence, glacier sites show high seismic activity. The elastic properties of water ice at seismic frequencies are similar to shallow crustal rocks, making tools from solid-Earth seismology fully applicable in glacial settings [Aster and Winberry, 2017]. In recent years, the number of publications related to seismological studies of the cryosphere has experienced rapid growth [Podolskiy and Walter, 2016].
The ground perturbations caused by seismic waves in ice can be recorded by ground motion sensors installed at the ice surface, in boreholes or at more stable sites in the vicinity of the ice [Aster and Winberry, 2017]. Recorded seismic waves carry information about the medium they propagated through, as well as their source mechanism. Seismic experiments provide a window into the internal properties and structure of the ice body, which is otherwise only accessible by expensive drilling [Podolskiy and Walter, 2016]. Moreover, seismological measurements offer a sub-second temporal resolution. This is unrivaled by any other glaciological method [Walter et al., 2020]. For these reasons, arrays of seismic sensors are an ideal tool for monitoring glaciers. Potentially, seismological studies can help to better quantify effects of climate fluctuations and help to predict natural hazards such as ice avalanches and glacial

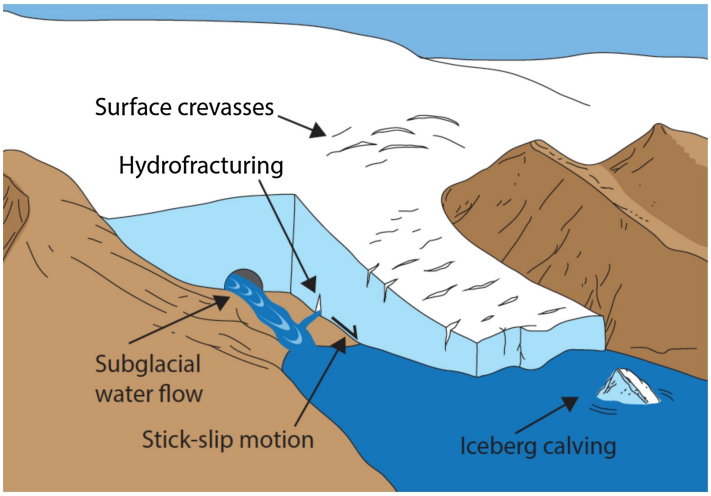


Figure 1-1: Schematic illustration of various sources of cryoseismic events on a glacier. Recordings from mountain glaciers are dominated by surface events. Recent research often focuses on stick-slip motion, as it potentially helps to improve our understanding of glacier sliding. Figure from [Podolskiy and Walter, 2016].

lake outburst floods [Maurer et al., 2020].
Sources of cryoseismic events occur at all parts of the glacier: its surface, its interior and at the glacier-bedrock/sediment interface. One of the most abundant sources of seismicity on glaciers are the formation of crevasses. Deep cracks open at the surface when strain rates of the glacier exceed a depth-dependent fracture criterion of the ice [Podolskiy and Walter, 2016]. In mountain glaciers with high melt or steep slopes, near-surface events make up the vast majority of all short-duration seismic detections [Aster and Winberry, 2017]. Surface icequake waveforms are dominated by simple Rayleigh waves, typically with a frequency content between 10 to 50 Hz [Podolskiy and Walter, 2016].
Glacier sliding plays a crucial role in the evolution of glaciers and land-forming processes. Sliding is strongly influenced by friction at the glacier-bedrock/sediment interface, as well as by subglacial hydraulic conditions. Accordingly, glacier movement varies with the subsurface conditions. Irregularities in the subsurface might oppose glacier movement. These locked asperities, known as sticky spots, may oppose a substantial part of the driving stress and produce basal icequakes with stick-slip source mechanism. Source mechanisms corresponding to the shearing of the base of the ice body along sticky spots can be modeled by double-couple moment tensors [Aster and Winberry, 2017]. Basal icequakes have been observed on glaciers worldwide [Aster and Winberry, 2017]. In reverse, recordings of stick-slip events can provide valuable insights about glacier sliding. Unfortunately, deep icequakes are often difficult to discern midst many surface events. One approach to detect basal icequakes is to analyse the spectral characteristics of recordings. Stick-slip icequakes often display signal with a broad frequency range, commonly containing frequencies above 100 Hz. They usually show P- and S-wave onsets [Podolskiy and Walter, 2016]. Additionally, stick-slip events often occur clustered in time and space and show highly similar waveforms [Gräff and Walter, 2021].
Events originating at the base of the glacier are not necessarily related to a shearing source mechanism. Crevassing not only occurs at the surface but also the base of glaciers. Collapsing cavities are another potential source of seismic energy [Podolskiy and Walter, 2016].

Different glacier processes acting as sources for cryoseismic events are illustrated in Fig. 1-1. Hydraulic processes play an important role in various glacial processes. Temperate glaciers are close to the melting point along the whole ice column and water flow may occur at the surface and base. Storage and flow of water are central to glacial erosion, mass balance and transport of heat. Seismicity due to water movement is mainly related to two physical processes: Turbulent flow and resonance of water-filled cavities [Aster and Winberry, 2017]. [Gimbert et al., 2016] used recordings of seismic noise in combination with water discharge measurements to infer changes in water pressure gradients, subglacial channel sizes and sediment transport.

1-3 Working principles of DAS

DAS is an emerging technology demonstrating great potential in active and passive seismic surveys. By repurposing a telecommunication fiber-optic cable as a sensing element, the strain or strain-rate along the cable can be obtained with high temporal and spatial resolution. Dense measurements over large cable lengths (10s of km and potentially over 100 km; [Zhan, 2019]) are in the process of revolutionizing many areas of seismology. A strong incentive is that dark fibers, existing unused fibers originally deployed for telecommunication, can be connected to a DAS Interrogator Unit (IU) and used as a sensing element without extra modification [Jousset et al., 2018]. Given the amount of dark fiber existing all around the globe, this could greatly help to increase the coverage of the Earth's surface with seismological instruments.

A DAS system is composed of an optoelectronic IU attached to a fiber-optic cable. However, there is no strictly defined architecture for DAS systems and a variety of different measurement principles exist. For a detailed description of existing measurement principles, the reader is referred to [Hartog, 2017]. Here, only the measurement principle known as Coherent Optical Time-Domain Reflectometry (COTDR) is covered. This technology is utilized by the SILIXA iDAS™ v2.4 system, which has been used in both Rhonegletscher experiments. The IU is responsible for sending light into the fiber, receiving the reflecting energy and processing it in real-time. It sends coherent laser pulses into the cable at a fixed pulsing frequency. Impurities in the glass core of the fiber cause variations of the refractive index of light. These variations cause Rayleigh back-scattering of the incoming laser pulse. The energy of the back-scattered light is recorded at the IU, where it is converted from two-way flight time to distance along the fiber using the known speed of light in glass (around 200,000 km/s. This allows to map each time sample of the back-scattered light to its corresponding location in the fiber-optic cable. If a cable segment is compressed or elongated between successive interrogations, this translates into a time or phase shift of the back-scattered signals. This phase shift is proportional to the strain of that cable segment [Parker et al., 2014]. Therefore, by successively sending laser pulses into the fiber, the strain-rate along all segments of the cable can be determined. The measurement principle of COTDR is illustrated in Fig. 1-2.

The physical distance over which the strain-rate is determined is referred to as the gauge length. Longer gauge lengths provide higher Signal-To-Noise Ratio (SNR) ratio of the data but lead to a worse spatial resolution [Parker et al., 2014]. Waves with a wavelength less then half the gauge length cannot be resolved anymore. Typically, a gauge length of 10 m is chosen for seismological experiments [Jousset et al., 2018]. Measurements can be obtained every few meters (or even <1 m) along the cable, depending on the high-speed data processing capabil-

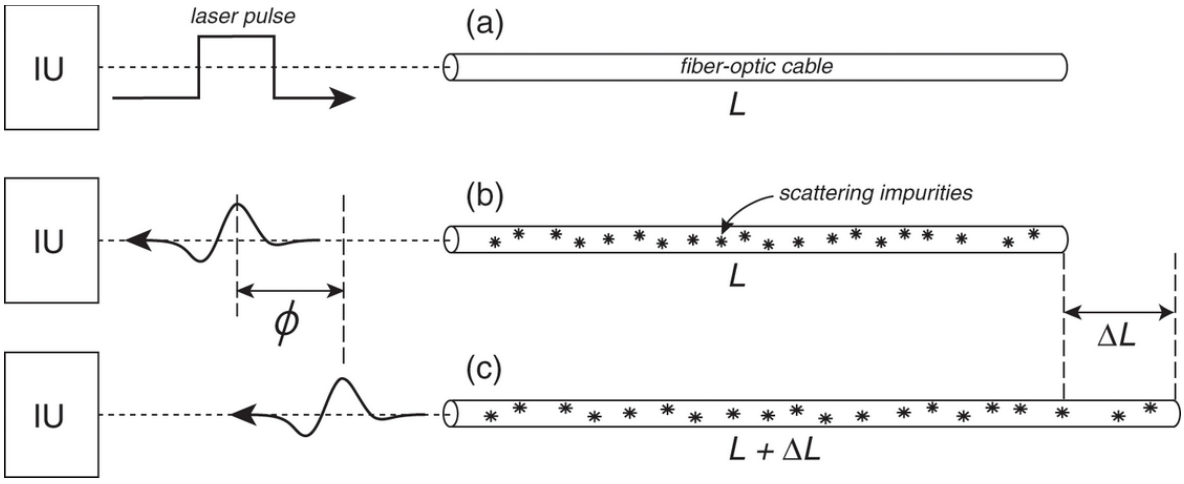


Figure 1-2: Illustration of the working principle of COTDR systems. The IU sends a laser pulse into the fiber, where the light is scattered at impurities in the fiber core. The energy of the back-scattered light is recorded back at the IU. When the fiber is stretched from length L to $L + \Delta L$, the back-scattered light is offset by a time or phase shift ϕ. By interrogating the fiber at a fixed frequency, the strain rate over arbitrary cable sections can be determined. Figure from [Lowrie and Fichtner, 2020], courtesy of Patrick Paitz.

ity of the IU [Parker et al., 2014]. Note that if the channel spacing is less than half the gauge length, neighbouring channels carry some redundancy in their information.

For many advanced seismic techniques, it is of great importance that the seismic wavefield is recorded with high fidelity of amplitude and phase [Parker et al., 2014]. Generally, strain recordings of DAS channels can be compared to linear strainmeters. Both are highly sensitive to waves propagating along the in-line axis and only weakly sensitive to wave propagating broadside to the in-line axis [Zhan, 2019]. The instrument response of DAS still is not very well established and is a field of ongoing research. [Paitz et al., 2020] compared DAS measurements with co-located seismometers and strainmeters for which the instrument response is well-known. After converting converting all instrument recordings to the same physical units, they compared amplitude and phase information for all experiments. The seismic experiments covered a wide frequency range from 3 mHz to 60 Hz. The DAS response is consistent over this interval, meaning that DAS measurements are broadband and theoretically flat. For this thesis project it is noteworthy, that the worst amplitude response (mean amplitude response of 10.42 dB) was observed for an icequake on the Rhonegletscher, with main frequency content between 40 and 60 Hz. A possible explanation for the amplitude discrepancy could be an imperfect co-location of the fiber sensor and reference instrument. Additionally, coupling effects between the glacier and fiber cable could be responsible. In that experiment, the fiber was covered in snow, which was likely damping high frequencies [Walter et al., 2020].

1-4 Context to related work

1-4-1 DAS for microseismic event detection

In the Earth sciences, DAS was first utilized by the energy industry in applications such as vertical seismic profiling [Mestayer et al., 2012]. [Webster et al., 2013] were the first, who utilized DAS for microseismic event detection. They deployed a DAS cable in a wellbore and successfully detected P- and S-waves generated in the subsurface. Installations on the surface have been described as being more challenging [Jousset et al., 2018], but have great potential for many applications such as hazard assessment and seismic imaging [Walter et al., 2020]. Continuing improvement of DAS technology resulted in sufficiently high waveform fidelity needed for earthquake detection and even seismic imaging [Parker et al., 2014]. This has been verified in numerous field experiments (e.g. see [Zhan, 2019] or [Lindsey and Martin, 2021] for excellent overviews).
DAS has also been used for microseismic event detection on ice. [Walter et al., 2020] were able to detect basal and surface icequakes on the Rhongletscher. [Hudson et al., 2020] tested various DAS configurations against co-located geophone recordings for event detection on the Rutford Ice Stream, Antarctica. To investigate coupling effects between the cable and the ground, they tested the different array configurations: a unburied linear cable, a buried linear cable and a buried triangle arrangement. The latter exhibits sensitivity in both horizontal directions, due to the angle between the legs of the triangle. Because of a near-surface firn layer, P-waves from basal icequakes arrived with normal incidence, i.e. completely braodside to the fiber. Therefore, only S-wave arrivals were visible in their data. Every DAS configuration detected less events than the geophone arrays. The triangle configuration detected more than three times as many events as the buried linear configuration. The unburied linear DAS configuration did not detect a single event. The authors mention, that their geophone array had larger extension than the DAS configurations. A cable length comparable to or longer than the ice thickness might provide better results. The 2-D nature of the triangle arrangement seems to be very effective for microseismic event detection, as it partly mitigates the broadside sensitivity inherent to DAS measurements. An unburied cable is subject to increased meteorological noise, which can completely mask events of interest.
[Pretorius et al., 2021] applied a Convolutional Neural Network (CNN) for cryoseismic event detection. They trained the network on a labeled set of englacial events recorded in a vertical borehole on the Stock Glacier, West Greenland. The trained CNN was then applied to passive records in order to identify new events.

1-4-2 Machine learning in seismology

ML methods have been successful in extracting key information from datasets for a wide range of applications [Roscher et al., 2020]. ML is a data-driven approach that learns to perform tasks without being explicitly programmed to do so. This makes ML methods very appealing for automating complex tasks that would otherwise require manual input from experts. ML can also help to discover new patterns and relationships hidden in the data [Bergen et al., 2019]. Especially exploration of large-scale datasets may demand automated data processing.
ML has been applied in the Earth sciences for several decades but only recently gained broader

interest. A major reason for this are recent performance breakthroughs in computing power and the availability of large memory. Additionally, ML algorithms are constantly refined and nowadays widely accessible through easy-to-use toolboxes [Bergen et al., 2019].
ML approaches can be grouped into two primary classes: Supervised learning and unsupervised learning. In supervised learning, the ML algorithm is trained on a set of annotated examples, known as training data. The model learns to make predictions of the likelihood of input data to belong to a certain target. Since the ground-truth data is known to the algorithm (input data and label), the accuracy can be monitored and backpropagation of errors can be used to improve the model. Unsupervised learning methods detect patterns or structure in data without relying on training data, i.e. no labels are known prior. Unsupervised ML is often used for early data exploration and data visualization. The raw data of the Rhonegletscher experiment is unlabeled, and unsupervised learning methods are used for data exploration and visualization. Following, related studies leveraging unsupervised learning methods are discussed.
Clustering methods, such as k-means and self-organizing maps, have been successfully utilized for unsupervised classification of DAS data. [Martin et al., 2018] used wavelet transforms for feature extraction and subsequent clustering of noise sources. [Chamarczuk et al., 2019] used array processing techniques for feature extraction on a large-N array at a mining site. The resulting features were used for clustering and different event types could be determined. [Shi et al., 2021] used array-based coherence of seismometer array above a fault zone to generate input for cluster analysis.
Instead of domain methods (e.g. wavelet transform, beamforming), feature learning could be utilized to reduce the dimensionality of input data. [Valentine and Trampert, 2012] use an autoencoder neural networksto to learn a feature representation of earthquake waveforms. Feature learning can also be combined with clustering, as done by [Holtzman et al., 2018].
Semi-supervised learning methods are on the intersection of supervised and unsupervised learning. Semi-supervised learning can be used with a small set of training data in combination with unlabeled dataset. The training data aides to train a model that find patterns and relationships in the data. One example are pattern-mining approaches, a simple method being template-matching. [Yoon et al., 2015] use a fast similarity-search to efficiently detect weak earthquakes in one week of continuous data. Network-based techniques have also been utilized for event detection by representing seismic arrays in graph structures, where single stations are represented by nodes [Riahi and Gerstoft, 2017]. The edges between the nodes are determined by the array covariance matrix. This helped to identify weak source only visible by a fraction of the large-N array.

1-5 Thesis outline

This work is organized as follows.
The introduction chapter gives a general overview about the problem investigated in this work. The principal motivation for the work is stated. The field of cryoseismology is introduced and the basic working principle of DAS is explained. Context to related work is discussed, in particular the use of DAS for microseismic event detection and application of ML methods in seismology.
Chapter 2 describes the hybrid workflow applied in this work. Different wavefield transformations from the field of array processing are introduced. For the unsupervised classification

part, hierarchical clustering is explained. Clustering is applied to the input features computed with the wavefield transformations.
Chapter 3 gives an introduction to the Rhonegletscher dataset. The chapter starts with an overview of the field campaign conducted by ETH researchers in Summer 2020 at the Rhonegletscher, Switzerland. The study site is described and the basic setup of the DAS experiment is introduced. The different preprocessing steps involved in preparing the data for further analysis are motivated. Additionally, the cloud setup utilized for the processing of the data is described.
The first part of chapter 4 shows the results obtained during the thesis project. First, results from the classical approach of manual picking of events are given. This was used to build a primary icequake catalogue. Furthermore, the limitation of such a manual analysis are discussed. Afterwards, the results obtained by the hybrid workflow are shown. Extracted features of cryoseismic events are displayed and the subsequent cluster analysis is visualized. In the second part of chapter 4, all results are discussed.
The final chapter concludes the findings of the thesis report. Suggestions for further research are stated.

Chapter 2

Methods

This chapter describes all relevant methods applied within the framework of this work. The general processing workflow is a combination of array processing techniques and unsupervised classification. Various 2-D wavefield transformations are described which are used to extract input features for the clustering algorithm. The generic concepts of clustering are outlined.

2-1 Hybrid workflow

This work follows a classification workflow comprised of two main parts: (1) feature extraction by means of array processing and (2) unsupervised classification using the extracted features as input.
Feature extraction can be understood as a type of dimensionality reduction. Typically, the raw data (or search space) contains a large number of data points. As an example, a DAS data sub-window of 200 channels and 15 s length at 1 kHz sampling frequency contains 3 million data points, i.e. it has a dimensionality of 3 million. Feature extraction helps to encode the search space in a lower-dimensional representation. Ideally, the extracted features capture interesting parts of the search space and help to classify it effectively [Khalid et al., 2014]. Finding a good feature representation is arguably the most important and difficult step in the hybrid workflow. The quality of the clustering output primarily depends on the quality and descriptiveness of the input features. Since DAS can be described as a seismic array/antenna [Zhan, 2019], it is plausible to leverage array processing techniques for feature extraction. This approach has been tested previously by [Chamarczuk et al., 2019] for unsupervised classification of signals recorded by a large-N array monitoring an industrial mining site.
Different array processing techniques are candidates for such an analysis. [Chamarczuk et al., 2019] use sum-and-delay beamforming, the InterLoc source location and Power Spectral Density (PSD) for feature extraction. They mention other methods such as the slant-stack, also known as time-frequency, $(\tau - p)$ transform, velocity spectral analysis, the frequeny-wavenumber $(f - k)$ transform, curvelets and kurtosis. However, it is necessary that the output of the feature extraction does not depend on the absolute timing

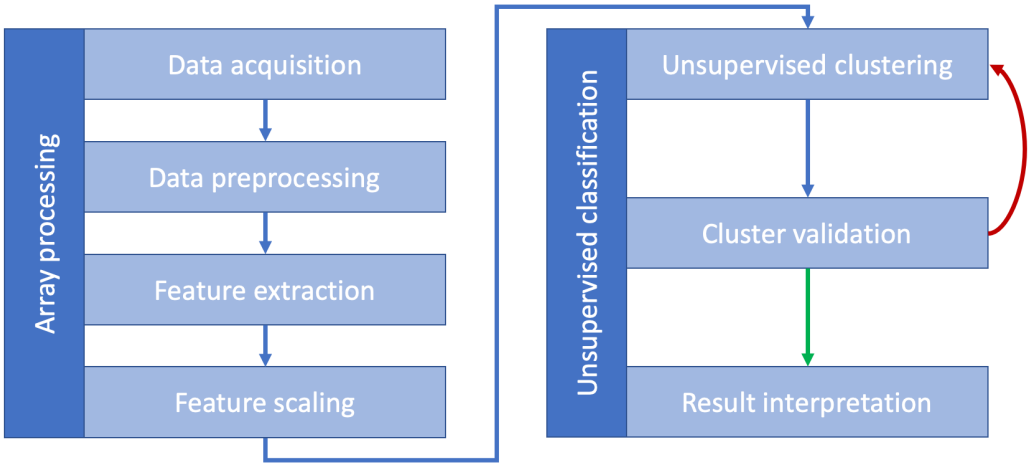


Figure 2-1: Flowchart illustrating the hybrid workflow. Output from the array processing is used as input features for an unsupervised clustering algorithm. In practice, clustering is often an iterative procedure. Adapted from [Chamarczuk et al., 2019].

and position inside a sub-window. This is because the features should not be influenced by translation of events inside a sub-window. It should not matter if a cryoseismic event occurs at the start or end of a time window if it has the same characteristics. Hence, the $\tau - p$ domain is not suited for feature extraction but the frequency-slowness $(f - p)$ domain potentially is.
It is noteworthy, that data preprocessing can also serve as a form of dimensionality reduction. After a low-pass or bandpass filter, the data can often be downsampled, reducing the number of data points. When filters are chosen appropriately, this does not result in a loss of information. On the contrary, the main objective of preprocessing is enhancing the signal of interest.

The complete workflow is shown in Fig. 2-1. After data acquisition, the raw data is preprocessed. Array processing techniques are utilized for feature extraction for smaller sub-windows in space and time. The extracted features are prepared for the unsupervised classification by appropriate scaling, following conventional ML input feature preperation. The output of the array processing is used as input features for the unsupervised clustering. There exist a vast range of different clustering algorithms, based on different assumptions on the underlying data structure [Xu and Tian, 2015]. The right selection of the algorithm and similarity measure is usually an iterative process. After each run, the resulting clusters are evaluated by common clustering metrics as well as manual inspection. Then, the algorithm is adjusted accordingly and re-run until a satisfactory outcome is achieved. Finally, the final result is interpreted on basis of the original problem statement.

2-2 Array processing

Distributed strain recordings provide point-wise measurements of the acoustic and elastic wavefield along the cable. This yields a distributed network of seismic stations, with equal spacing along the cable. [Schweitzer et al., 2012] define that a seismic array mainly differs

from a regular network of seismic stations by the processing techniques applied for analysing the recorded data. A network of seismic stations can therefore be used as an array under certain conditions. For most array processing techniques, a high spatial coherence across the array is a necessary requirement. This leads to some constraints on the geometry and spatial extent of the array geometry, as well as data quality [Schweitzer et al., 2012].
When considering a fiber-optic cable with large extent, most microseismic events are only visible along fractions of the cable. This is due to the small amount of seismic energy released, in combination with attenuation and geometrical spreading of wavefronts. As a consequence, weak cryoseismic events are only coherent over small sections of the DAS array. It therefore seems sensible to divide the cable into smaller sections. This ensures a small aperture and high signal coherence across these shorter segments. Since the cable was predominantly laid out in a straight line, some processing techniques like beamforming are not well suited for analysis. DAS only measures one component (inline strain), which is not enough to constrain the backazimuth and velocity of the arriving wavefront unambiguously. For this, a cross-line or triangular configuration would be required [Zhan, 2019]. For the same reason, source-location techniques are not very suitable for analysis of the Rhonegletscher dataset.
It is possible to view DAS data as a 2-D matrix, with distance along the cable being one dimension and time being the other dimension. This gives a spatio-temporal representation of the seismic wavefield, corrupted by noise [Fernández-Ruiz et al., 2020]. This makes 2-D wavefield transformations a viable tool to analyse the data. In the following, the $f-k$ transformation and the $\tau-p$ transformation are introduced. Additionally, methods based on array-coherency are presented.

2-2-1 Frequency-wavenumber transform

DAS surveys typically exhibit similar spatial sampling compared to conventional geophone arrays deployed in active seismic surveys. The 2-D nature of the DAS data, allows for analysis in the $f-k$ domain by performing a 2-D Fourier transformation of the data. The wavenumber k can be interpreted as the spatial frequency of a plane wave and is given by

$$k = \frac{1}{\lambda_{phase}} = \frac{f}{v_{phase}}, \tag{2-1}$$

where λ_{phase} is wavelength, f is frequency and v_{phase} is the phase velocity of a plane wave. Since DAS is only measuring inline strain, it is not possible to determine the true wavenumber, but instead an apparent wavenumber k_{app} in the cable direction. Hence, it is not possible to obtain the true phase velocity $v_{phase} = f/k$ but only an apparent phase velocity v_{app}. The apparent velocity is a combination of the true phase velocity and the incidence angle θ of an incoming plane wave and is given by the relation

$$v_{app} = \frac{v}{sin(\theta)} = \frac{f}{k_{app}}. \tag{2-2}$$

The $f-k$ transform corresponds to a remapping of the data, with the output being the signal power of the data for a specific frequency bin and wavenumber bin. Note, that we are always dealing with bins because of the discrete nature of the data and corresponding transformations. Equation 2-1 implies that events with the same apparent velocity in the $t-x$ domain are mapped onto a line through the origin in the $f-k$ domain, with the slope

of the line being the inverse of the apparent velocity, also called slowness or ray parameter p. Consequently, events with different apparent velocities can be separated in the $f-k$ domain. This makes it an attractive method to differentiate events based on their wave type (different velocity in ice) or source location. The $f-k$ spectrum of a sub-window can be used directly as an input feature. However, to reduce dimensionality, resampling and smoothing of the $f-k$ spectrum is necessary. A different application of the $f-k$ transform is filtering, as muting in the $f-k$ domain can be used to discard signal that is not in the velocity range of interest. An example would be muting all velocities below 1000 m/s, as we do not expect seismic waves on ice with such a low apparent velocity.
Direct extraction of apparent velocities from an $f-k$ spectrum require integration along slanted lines. This implies interpolating the data in the $f-k$ domain, which can be practically challenging [Dietrich, 1990]. If direct extraction of apparent velocities is desired, it is more sensible to perform the slant-stack transformation, where interpolation is only done in the $t-x$ domain.

2-2-2 Slant-stack transform

The slant-stack transform is another way to decompose a wavefield into its plane-wave components [Yilmaz, 2001]. The slant-stack transform maps data from the $t-x$ domain to the $\tau-p$ domain, where τ is similar to the time-axis and p is slowness. This is achieved by summing amplitudes along slanted paths in the $t-x$ domain. The slowness p is the inverse of the apparent phase velocity and given by

$$p = sin(\theta)/v. \tag{2-3}$$

The slowness corresponds to the slope of a slanted line of angle θ in the $t-x$ domain, with τ being the time-axis intercept of the slanted line at $x=0$. The slant-stack transform is usually implemented in a two-step process. First, a linear moveout correction is applied to each trace. This can be implemented by the coordinate transformation

$$\tau = t - px, \tag{2-4}$$

where p is slowness, t is time, x is offset and τ is the intercept time. Second, the data is summed along the offset-axis by

$$S(p,\tau) = \sum_x P(x, \tau + px). \tag{2-5}$$

Here, $S(p,\tau)$ describes a plane wave with slowness p.
A power spectrum in the $f-p$ domain can be constructed by performing a 1-D Fourier transform along the τ-axis [Yilmaz, 2001]. This results in a frequency-dependent description of the apparent phase-velocity. This is very similar to the $f-k$ domain, but bins corresponding to the same apparent velocities are in a column instead of a slanted line. Therefore, no interpolation is required in a frequency-dependent domain. It is possible to construct features corresponding to a certain velocity and frequency range by summing along parts of a column in the $f-p$ domain.

2-2-3 Covariance matrix analysis

Continuous DAS recordings provide dense spatial sampling of the seismic wavefield. Weak sources cause coherent signal over sections of the cable. In seismology, statistical array processing techniques have been successfully used to identify coherent time-series as a function of time and frequency [Seydoux et al., 2016]. This approach can be used for identifying weak signals embedded in background noise, as is typically the case for local microseismicity. One major advantage is, that no prior knowledge of the study site is required.
The main idea was developed by [Bartlett, 1954] and [Lawley, 1956], who proposed that the number of independent signals can be obtained from the eigendecomposition of the covariance matrix, where larger eigenvalues correspond to the independent signals and smaller eigenvalues correspond to the background noise. Following, the construction and decomposition of the array covariance matrix are described in detail.
The array data vector at a frequency f can be defined as

$$\vec{u}(f) = \langle u_1(f), u_2(f), ..., u_N(f) \rangle, \tag{2-6}$$

where $u_i(f)$ is the complex Fourier spectra of channel i and N is the total number of sensors. The covariance matrix at a frequency f is defined as

$$\mathbf{\Gamma}(f) = E[\vec{u}(f)\vec{u}(f)^\dagger], \tag{2-7}$$

where E represents the expected value and † denotes the conjugate transpose. This means that the outer product $\vec{u}(f)\vec{u}(f)^\dagger$ spans a $N \times N$ matrix [Gerstoft et al., 2012].
The steps for computing the covariance matrix are discussed in [Shi et al., 2021]. The covariance matrix is built from the time average of Fourier cross-spectra matrices calculated over a set of half-overlapping sub-windows. Two window sizes are involved for the construction of the covariance matrix. Shorter time windows, referred to as sub-windows, are used to compute the Fourier cross-spectra matrices. A long time window, referred to as averaging window, is composed of multiple half-overlapping sub-windows. The average of all sub-windows inside an averaging window then gives the covariance matrix for that averaging window. Mathematically this can be stated as

$$\boldsymbol{C}(f) = \frac{1}{M} \sum_{m=1}^{M} \vec{u}_m(f)\vec{u}_m(f)^\dagger, \tag{2-8}$$

where $\vec{u}_m(f)$ is the array data vector of sub-window m and M is the total amount of sub-windows in one averaging window. Note, that by definition the Fourier cross-spectra matrix is Hermitian, meaning that $\Gamma(f)^\dagger = \Gamma(f)$. Since the covariance matrix is an average of many Hermitian matrices, it itself is also Hermitian and always diagonalizable. The length of the averaging window gives the time resolution of the covariance matrix analysis.
[Gerstoft et al., 2012] consider the data vector as a linear model of signal $\vec{s}_k$, coherent propagating noise $\vec{n}_c$ and incoherent noise $\vec{n}_i$. For a number of K discrete sources with complex amplitudes α_k and under the assumption that the number of stations is much larger than the number of discrete sources (i.e., $K << N$), the linear model can be stated as

$$\vec{u}(f) = \vec{n}_c + \vec{n}_i + \sum_{k=1}^{K} \alpha_k \vec{s}_k. \tag{2-9}$$

Based on Eq. 2-9, the covariance matrix can also be denoted as

$$\boldsymbol{C}(f) = \boldsymbol{C}^s(f) + \boldsymbol{C}^p(f) + \boldsymbol{C}^i(f). \tag{2-10}$$

Here, $\boldsymbol{C}^s$ denotes the covariance matrix of the coherent sources, $\boldsymbol{C}^p$ denotes the covariance matrix of the propagating coherent noise and $\boldsymbol{C}^i$ is the covariance matrix of the incoherent noise. [Gerstoft et al., 2012] give a statistical description of the noise covariance matrices and [Seydoux et al., 2016] extended $\boldsymbol{C}^p$ by considering 2-D isotropic surface waves as the dominant propagating coherent noise.
The eigenvalues $\langle \lambda_1(f), \lambda_2(f), ..., \lambda_N(f) \rangle$ can be obtained by solving the linear equation

$$\boldsymbol{C}\vec{v} = \lambda\vec{v} \tag{2-11}$$

at each frequency, with $\vec{v}$ being the corresponding eigenvectors.
Various metrics can be constructed from the resulting eigenvalue distribution. [Seydoux et al., 2016] define the spectral width of the covariance matrix and use it as a simple threshold trigger for earthquake detection at a volcano site. [Shi et al., 2021] use the first eigenvalue, the Shannon entropy of the eigenvalue distribution, the coherency function and the variance of the eigenvalues. They combined three of these metrics and used them as input features for a clustering algorithm. For this study, the coherency function and the first eigenvalue are identified as useful metrics.
The first eigenvalue $\lambda_1(f)$ is simply the largest obtained eigenvalue. The coherency function is defined as the ratio between the dominating wavefield component and the full wavefield [Gersztenkorn and Marfurt, 1999]. This can be expressed as the ratio of the first eigenvalue and the sum of all other eigenvalues, i.e.

$$\sigma_c(f,t) = \frac{\lambda_1(f,t)}{\sum_{i=1}^{N} \lambda_i(f,t)}. \tag{2-12}$$

These two features are obtained for each frequency component at each time step, effectively yielding a time-frequency representation of the wavefield. Since the data is divided into smaller cable sections as well, we also obtain a spatial component.

2-2-4 Event triggering and data stacking

Processing the data channel-by-channel as in conventional seismological studies can quickly become very time intensive for the large number of channels commonly encountered in DAS surveys [Stork et al., 2020]. A possible solution is to stack multiple consecutive channels to reduce the data volume. Stacking can be performed linearly or non-linearly, e.g. phase-weighted, semblance-weighted or Nth-root stacks [Schimmel and Paulssen, 1997]. Stacking can be performed on the raw waveforms but also on derived functions thereof. This can be utilized in a classic signal processing and triggering workflow. Following [Stork et al., 2020], the data is pre-processed and subsequently the Short-Term Average/Long-Term Average (STA/LTA) method is used to compute a characteristic function for each trace. Then, the characteristic functions are stacked over a range of consecutive channels. Event triggering is performed on the output from the stacking. This has two benefits: (1) A large amount of channels can be represented by one single resulting function. (2) Individual channels often show a low SNR,

and locally confined noise with large amplitude variations can easily trigger a false detection on a single channel. Stacking multiple DAS channels can increase the SNR significantly [Hudson et al., 2020].
STA/LTA is by far the most common triggering algorithm in use today [Trnkoczy, 2009]. One main advantage is its simplicity and robustness. The basic algorithm includes to sliding windows: The STA window calculates the amplitude average in a window with short-time duration and the LTA window computes the same metric but given a window with longer time-duration. The STA and LTA for a single channel can be stated as

$$STA(t) = \frac{1}{N_S}\sum_{n=1}^{N_S} y_n, \qquad LTA(t) = \frac{1}{N_L}\sum_{n=1}^{N_L} y_n, \tag{2-13}$$

where N_S is the number of samples in the STA window, N_L is the number of samples in the LTA window and y is the waveform data. Both windows are moved along the complete time series sample-by-sample and the ratio $R(t) = STA(t)/LTA(t)$ is computed at each time sample. The basic idea is that the STA window will be high when it includes a signal onset, while the LTA will be comparably low because it still contains the previous background noise. A peak detection algorithm is then run on the characteristic function $R(t)$. To retain optimal results, the STA/LTA parameters like the triggering threshold and window durations have to be selected carefully. The right selection is often site-dependent [Trnkoczy, 2009]. The characteristic functions can then be stacked over a range of channels. [Stork et al., 2020] test various stacking approaches and ultimately decide for linear stacking, since it is very time efficient and achieves a good performance on their test data set. Linear stacking of the characteristic function is a simple mean, i.e.

$$G_{lin}(t) = \frac{1}{N}\sum_{j=0}^{N} |R_j(t)|, \tag{2-14}$$

where $j = 0, ..., N$ are indexes for all channels considered for the stack.
It is also possible to stack channels in a non-linear way. Here, we consider the Phase-Weighted Stack (PWS) [Schimmel and Paulssen, 1997]. It assigns a weight to each trace based on the coherence of the instantaneous phase. The analytical signal is given by

$$S(t) = y(t) + iH(y(t)) = A(t)e^{i\Phi(t)}, \tag{2-15}$$

where $y(t)$ is the waveform of the channel and $H(y(t))$ is its Hilbert transform. The analytical signal can also be expressed with a time-dependent amplitude $A(t)$ and phase $\Phi(t)$. The coefficients for the PWS can be computed by

$$c(t) = \frac{1}{N}\left|\sum_{j=1}^{N} e^{i\Phi(t)}\right|, \tag{2-16}$$

where N is the number of traces used in the stack. The coefficients range between 0 and 1 as a function of time. For coherent signal, the amplitude of the coefficients are one, while zero amplitude means that the traces summed up destructively [Schimmel and Paulssen, 1997]. In essence, the idea is to mute channels which do not show coherent signal. The final stack is obtained by weighting each trace by its time-dependent coefficient, i.e.

$$G_{PWS}(t) = \frac{1}{N}\sum_{j=1}^{N} y_j(t)\left|\frac{1}{N}\sum_{j=1}^{N} e^{i\Phi(t)}\right|^{\nu}. \tag{2-17}$$

The PWS stack acts as a filter and its sharpness is controlled by the power ν. For $\nu = 0$, the linear stack is obtained. PWS can be used when stacking characteristic functions, but also for the slant-stack in Eq. 2-5.

2-3 Clustering

Clustering, also known as unsupervised classification, is the task of constructing decision boundaries based on unlabeled training data [Xu and Tian, 2015]. The objective is to find natural groupings, called clusters, in multidimensional data, based on measured or perceived similarity between the patterns. Clustering is often considered the most important problem of unsupervised learning [Xu and Tian, 2015]. Clustering is a difficult problem to solve, since clusters can have different sizes and shapes. Moreover, the number of clusters is often dependent on the resolution on which we view the data. There is no universal definition of clustering. A classical functional description was given by [Jain and Dubes, 1988]:

1. Instances in the same cluster must be similar as much as possible.
2. Instances in different clusters must be different as much as possible.
3. Measurements for similarity must be clear and have practical meaning.

Even with these functional definitions of a cluster, it is not easy to come up with an operational implementation. One of the challenges is to select an appropriate measure of similarity to define clusters. A sensible similarity measurement is generally both data (cluster shape) and context dependent. [Xu and Wunsch, 2005] give the following standard steps for clustering analysis:

1. Feature extraction and selection.
2. Designing the cluster algorithm to fit the problem statement.
3. Result evaluation.
4. Result explanation.

When compared to our hybrid workflow (see Fig. 2-1), the first step corresponds to the array processing part while steps 2-4 are the unsupervised classification part. The choice of the feature extraction method determines the density and shapes of patterns in the feature space, on which the clustering algorithm operates. Based on the structure of the data in the feature space, an appropriate clustering algorithm and similarity measurement has to be chosen. Popular similarity measurements are Euclidean (L_2) distance, Manhattan (L_2) distance or cosine similarity. The latter is an interesting choice, as is does not depend on the actual magnitude of features but rather their shape. There exist a great number of different clustering algorithm, based on different notions of what a cluster is. E.g. a cluster can be based on distances between cluster members, point-density inside the clusters or by a description with a statistical distributions.

2-3-1 Hierarchical clustering

Hierarchical clustering is chosen as the basic clustering algorithm, as it can deal with arbitrary shaped data and arbitrary types of attributes. Two kinds of hierarchical clustering exist, which are based on an agglomerative and divisive clustering approach. In practice, only agglomerative hierarchical clustering is widely used [Xu and Tian, 2015]. It works in the following manner: In the initialization stage, each data point gets assigned its own cluster. In the next step, similar clusters are merged together. This procedure is repeated until one single cluster is formed which includes all data points. The complete procedure results in a tree-like structure, called a dendrogram. Here, the final all-enclosing cluster represents the trunk of the tree, while the data points are the leaves. This allows to easily detect the relationship between different clusters [Xu and Tian, 2015].
Various methods, called linking criterion, exist on how to decide which clusters to merge at each iteration. Examples are ward linkage, complete linkage, single linkage and average linkage. The most suitable linkage criterion highly depends on the structure of the underlying data. Complete linkage merges clusters on the maximum distance between elements in each cluster.

2-3-2 Result validation

The cluster algorithm always converges and yields an output no matter the input and chosen parameters. Hence, the results have to be validated in some way and potentially the clustering workflow has to be adjusted. It is inherently difficult to validate unsupervised methods, as they lack ground-truth data for comparison. There exist methods for quantitatively evaluating clustering results. They are based on the classical notion of a cluster (see list 2-3). One widely used metric for evaluating clustering results is the silhouette coefficient [Xu and Tian, 2015]. It gives a measure of how similar a point is to other points in his cluster, compared to points in different clusters. The silhouette value ranges from -1 to +1, with a high value indicating similarity to points in the same clusters.
Additionally, results are visually inspected. Events in the same cluster are inspected for similarity and should be different to events in other clusters. It is also possible to take sub-windows that we perceive as very similar and compute their distance, which should be low. On the other hand, the computed distance should be high if we compare it to a sub-window with a vastly different character.

Chapter 3

Data

This chapter provides a general overview of the Rhonegletscher experiment and the acquired data. The cloud setup used to access and process the data is described. The preprocessing steps applied to enhance the signal of interest are demonstrated on an example time window.

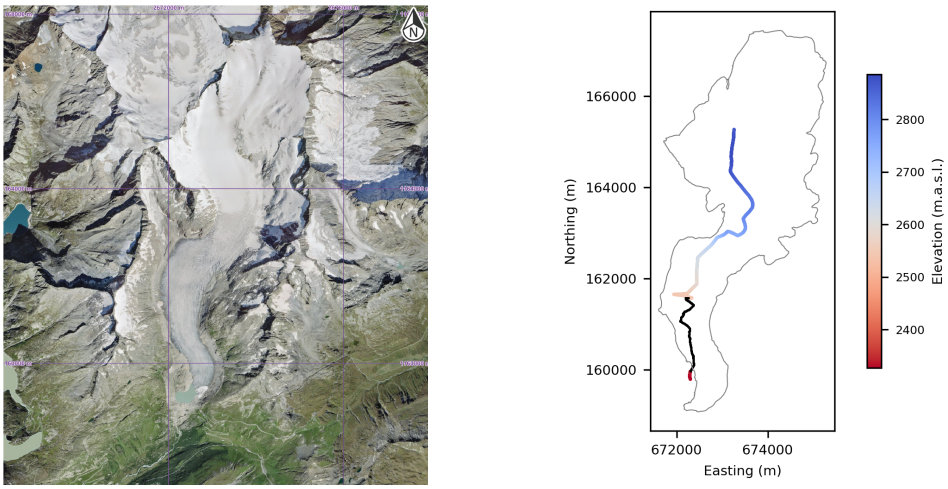
3-1 Rhonegletscher experiment

3-1-1 Study site

The Rhonegletscher is a temperate valley glacier located in the central Swiss Alps. It has a surface area of 14.64 km^2 and a total length of 7.55 km [GLAMOS, 2020]. The glacier flows southward from 3600 to 2200 m a.s.l. with an average slope of 10° [Walter et al., 2020]. Since 1879, the Rhonegletscher has retreated by more than 1500 m [GLAMOS, 2020]. Due to increasing mass loss in the last two decades, a proglacial lake has formed at the glacier tongue in 2005 [Tsutaki et al., 2011]. The ice-column is at melting-pressure [Gräff and Walter, 2021] and during summer, increased surface melt occurs when the sun heats up the glacier surface. In winter, most of the glacier is covered in snow. During the time the DAS experiment was conducted, approximately the upper third of the glacier was still covered in snow, while the lower part was bare ice. An orthophoto of the Rhonegletscher is shown in Fig. 3-1a

3-1-2 Data collection

In July 2020, a SILIXA iDAS™ v2.4 IU was installed and a 9 km long fiber-optic cable was deployed at the Rhonegletscher site. The fiber covered most of the glacier from the ablation zone down to the terminus. The cable layout is shown in Fig. 3-1b. The cable was not actively buried in the ice/snow. However, it has a black coating and thereby warms up faster than the surrounding ice. In the upper part of the glacier, the cable effectively melted itself into the snow and the upper section of the DAS cable can be considered as buried. Below the ice falls, there is no snow cover and surface melting occurs during the day. Therefore, the



(a) (b)

Figure 3-1: (a) Orthophoto of the Rhonegletscher and (b) cable layout of the experiment. The cable covers the whole glacier extent from the accumulation zone to the ablation zone. Source of aerial image in (a): Swiss Federal Office of Topography.

cable is not as well coupled compared to the upper section.
For most of the experiment, the pulsing frequency of the laser was set to 1 kHz, with a pulse width of 50 nm. The gauge length, over which the strain measurement is computed, was 10 m for the entire experiment. The 10 m gauge length is prescribed by the iDAS™ v2.4 system and cannot be adjusted. In the first few days of recording, a channel spacing of 2 m was chosen, but for the majority of the experiment, a 4 m channel spacing was selected. For 4 m channel spacing, a total of 2225 channels were recorded simultaneously. However, not all channels recorded useful signals. The complete cable was spliced together from three separate smaller cables. In between the spliced sections are small cable drums, with fiber coils of various length, of which the data has been disregarded in this thesis. Furthermore, to reduce the tension along the fibre, small, loose loops were occasionally incorporated into the cable layout.
Channel locations were obtained by measuring the location of a few channels with GPS, performing a tap-test and linearly interpolating the remaining channel locations. This leads to some uncertainty in the true channel locations.
The dense spatial sampling results in a huge data volume. One day of DAS recordings has a size of >400 GB. The complete survey acquired 17.6 TB of data. All data was backed up on external hard drives and additionally uploaded to the Amazon Web Services (AWS) cloud.

3-1-3 Data access

All collected data was converted to the Hierarchical Data Format (HDF) and then uploaded to the AWS cloud. The Simple Storage Service (S3) was chosen as an appropriate web service, as it is built for storing large data volumes and provides reasonably fast access. The HDF was chosen, as it is a data format targeted for large datasets and parallel computing. The Rhonegletscher data is stored in files of 30 seconds recording time, each file containing the waveforms of all channels. All files are stored inside an S3 bucket, which is similar to a

file folder on local computers. The bucket is organized in sub-folders corresponding to a complete day of recording. The S3 bucket can be interacted with using the Command Line Interface (CLI) tool provided by AWS. This allows for the copy of large amount of data from the S3 bucket to a local machine or a different cloud web service.

3-2 Handling large-scale datasets

The enormous data volume of the Rhonegletscher dataset limits efficient analysis on personal computers. To effectively handle the amount of data, a cloud-based solution was chosen. Specifically, the Elastic Compute Cloud (EC2) webservice of the AWS ecosystem is used. This is a very natural approach, as the data is already stored on AWS inside the S3 web service.
Many computational operations like pre-processing are performed on a trace-by-trace basis. This can become very time-consuming when computed in a serial fashion. For such operations, computational cost can be improved significantly when using multiple cores and parallel computing.

3-2-1 Cloud computing with AWS

To efficiently perform computations on the data, the EC2 of the AWS ecosystem was utilized. EC2 allows for on-demand computing by creating instances with arbitrary technical specification (e.g. memory size and number of cores) in near-real time. For this project, instances from the C6g[1] group were used, as they are designed for high-performance computing. EC2 instances cannot directly access objects stored in S3 buckets. Instead, instances can be connected to Elastic Block Storage (EBS) volumes, which are similar to external hard drives. In order to solve this problem, EBS volumes of 500 GB size were created and one day of data was copied from the S3 storage to each EBS volume. After installing the necessary software on the C6 instance, it can load the data from the EBS volume, process the data and save the output to the corresponding EBS volume.

Processing of the data was done using self-written Python scripts. The data was read in using the h5[2] library and subsequently converted to NumPy arrays. Signal processing was done using the NumPy [Harris et al., 2020] and SciPy [Virtanen et al., 2020] libraries. Data preperation and clustering was performed with the scikit-learn [Pedregosa et al., 2011] library.

3-2-2 Parallel computing

Many operations on the data are computed individually for each trace. This includes computationally relatively expensive steps like filtering in the time- and frequency domain. Computational cost can be greatly reduced if such operations are not run in series, but parallel on many cores. Since an operation on a single trace is independent on the same operation an a different trace, no communication between different sub-processes is required. This means that it is not necessary to implement complicated locking procedures or a message passing

[1] see `https://aws.amazon.com/ec2/instance-types/c6/` for more information
[2] `https://h5py.org/`

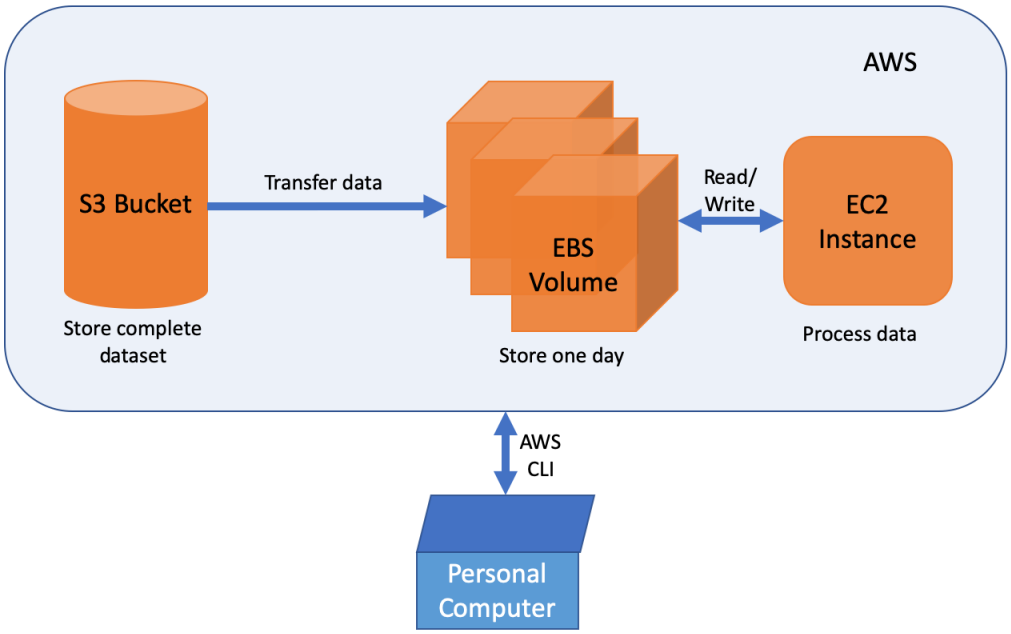


Figure 3-2: Schematic illustration of the cloud setup for processing the Rhonegletscher data.

interface.
A simple tool to parallelize Python scripts is the multiprocessing[3] library. In its most basic form, a function (e.g. containing all preprocessing steps) and an iterable (e.g. all traces) can be passed to a mapping function. The mapping function then loops over the iterables and sends each trace to a sub-process, where the function is applied to the trace. In this way, multiple cores can work in parallel. Afterwards the result is collected and can be used for further analysis.

3-3 Data preprocessing

3-3-1 Removing noisy channels

Since the cable sections and cable drums do not record in-line strain data, these channels have been discarded for the analysis in this thesis. When the cable is not perfectly straight, distance along the cable does not perfectly coincide with absolute distance. Channels corresponding to loops in the cable were identified by their small absolute channel spacing. All channels with less than 0.8 m offset to their neighbouring channel were removed. In total, 215 channels were discarded completely and the remaining 2010 channels were used for all further analyses.
The raw recordings contain a lot of noise, making it impossible to discern any events on the raw data. Due to varying coupling conditions, channels show different recording quality. Some channels appear very quite, while others are very noisy. In order to balance this discrepancy, trace normalization has to be performed. Otherwise, noisy channels with large amplitudes will dominate the data.

[3] https://docs.python.org/3/library/multiprocessing.html

3-3-2 Single channel data preparation

Signals of interest are located within in a certain frequency bandwidth. Surface crevassing has dominating frequencies between 10 to 50 Hz, while basal icequakes can have significant energy up to 100s of Hz. It would be interesting to monitor very low frequencies (e.g. down to 0.1 Hz), but unfortunately that frequency band is dominated by background noise. This noise is likely of meteorological nature, e.g. wind, temperature or surface melt. The snow covering the upper third of the cable, dampens frequencies above 90 Hz [Walter et al., 2020]. To isolate signals of interest, a 4^{th} order Butterworth bandpass filter is applied. After empirically testing different bandwidths on events, 10 Hz was chosen as an appropriate lower cutoff frequency and the high cutoff frequency was set to 90 Hz. After the filtering, there is no energy higher than at least 100 Hz in the data, and it can safely be downsampled in time by a factor of 5. The new samping frequency is 200 Hz, which is more than twice the higher cutoff frequency (Nyquist theorem). This reduces memory cost and speeds up all subsequent computations significantly.

Following [Bensen et al., 2007], spectral whitening was applied to each trace to balance the amplitude spectrum. The motivation is, that dominant frequencies in a sub-window should not dominate frequency-dependent data representations solely due to amplitude, but rather phase. Furthermore, the influence of spikes in the amplitude spectrum is reduced. Spectral whitening was implemented by transforming the data to the frequency-space ($f - x$) domain and running a smoothing window along the frequency axis. A window length of 0.3 Hz was chosen after testing different lengths. Afterwards, the data is transformed back to the time-space ($t - x$) domain. To compensate for different coupling conditions, some sort of normalization has to be applied to each channel. Three different normalization approaches were tested: (1) normalization by maximum amplitude, (2) energy normalization and (3) Automatic Gain Control (AGC) [Dondurur, 2018]. The latter performed the best as it diminishes the effect of large amplitude noise spikes without corrupting signal onsets. An AGC window length of 2000 ms was used to balance the amplitude along individual channels. The idea is to remove absolute amplitude information and calculate coherence solely due to similar phase of coherent signals.

A significant issue during analysis were common modes of the IU. These occur when the IU reverberates, e.g. when performing some task on the DAS system or the generator providing the electricity. Common modes are easy to identify, as they appear as vertical lines with infinite velocity. A simple solution would be to remove the mean along all channels for each time sample. However, this alone does not solve the problem, as an multiple vertical artefacts appear on the higher channels after applying amplitude normalization. This was solved by applying a high-pass filter along the space axis, filtering out extremely low wavenumbers. This filter has to be designed carefully in order to not cutoff useful signal or introduce artifacts. The complete preprocessing workflow is illustrated on an example data chunk containing a small event and an IU common mode in Fig. 3-3. As a last step, it is possible to normalize each trace by its total energy (in the signal processing sense).

After the data is preprocessed, it is cut into smaller sub-windows of 200 channels and 15 s length. Each window has 50% overlap in space with its neighbouring window. In time, a 5 second overlap between neighbouring windows was chosen. In space, a sufficient overlap is required because large events are necessarily cut. The channel overlap ensures, that an event is fully represented in at least one sub-window, if it would otherwise be cut right in the middle. Theoretically, only a small time overlap is necessary, as most events are of sub-second

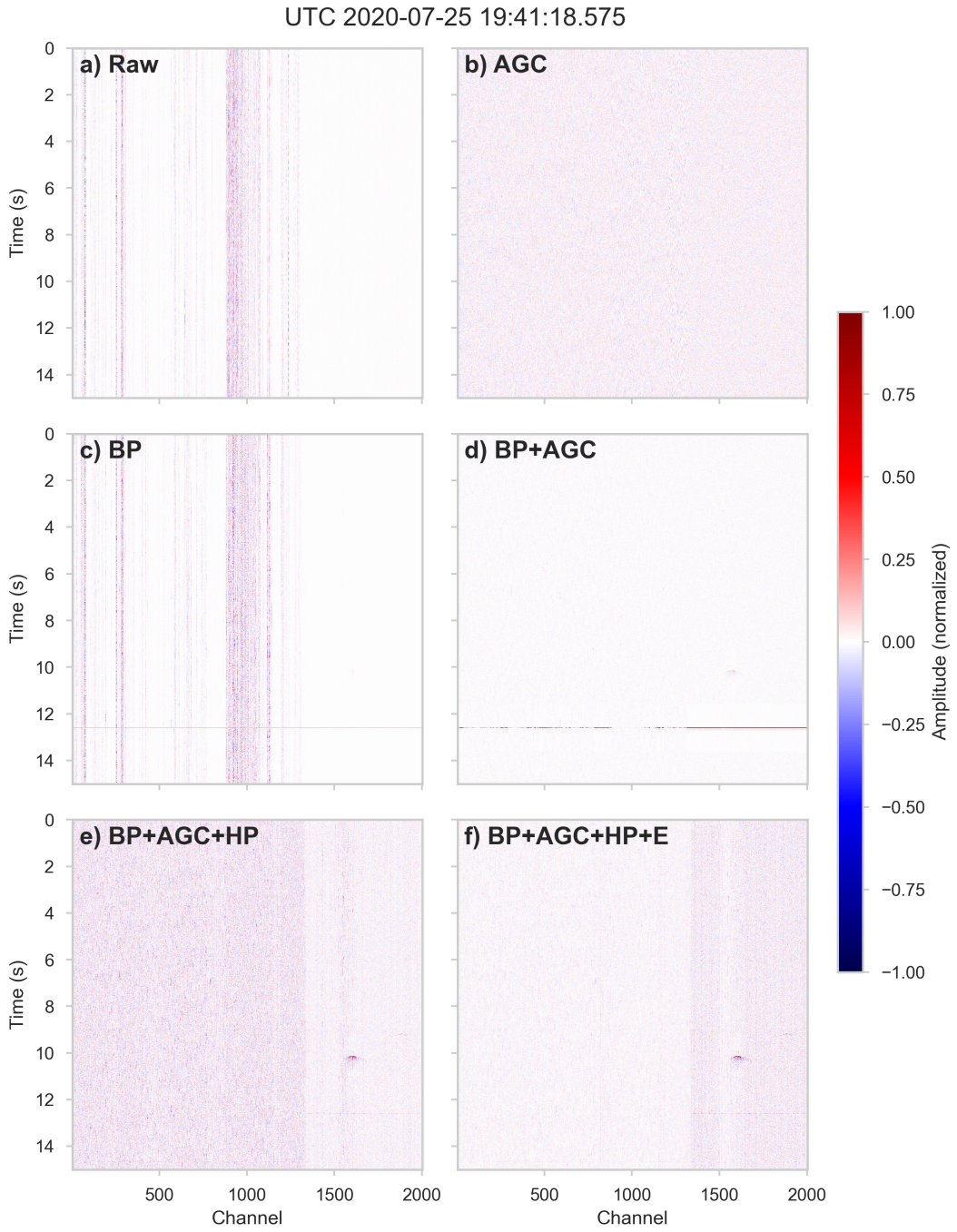


Figure 3-3: Different processing steps applied to the same time panel containing a small event and IU common mode noise. The following steps were applied to each sub-figure: (a) raw; (b) AGC; (c) bandpass filter; (d) bandpass filter + AGC; (e) bandpass filter + AGC + spatial highpass filter; (f) bandpass filter + AGC + spatial highpass filter + energy normalization. Bandpass filters are from 10-90 Hz. In the final image, there are no dominating channels and the common mode has been almost entirely muted.

time duration. The overlap was chosen as slightly longer in case certain filtering operation, like an $f - k$ filter are applied to each sub-window individually. In that case, the window needs to be tapered and a larger overlap would be required in order to not lose information.

Chapter 4

Evaluation

This first part of this chapter presents relevant results obtained by the hybrid workflow. Clustering results for different input features are shown, steps for the cluster validation are demonstrated and an interesting cluster is briefly analyzed. In the second part, all results and their implications are discussed.

4-1 Results

4-1-1 Event detection with conventional signal processing

To get an initial idea of events hidden in the data, manual picking was performed on multiple days of data. First, the data was preprocessed as outlined in section 3-3. Subsequently, a reverse STA/LTA trigger was run on all channels. Cryoseismic events usually are of short-time duration [Podolskiy and Walter, 2016] and an initial value of 0.3 s was chosen for the STA window. The length of the LTA window was set to 10 times the STA window length, i.e. 3 s. An example of a small icequake and the corresponding characteristic functions is shown in Fig. 4-1. The onset of the event is clearly captured by the triggering algorithm. Note that individual channels are quite noisy and STA/LTA ratios above 2 occur regularly, even when no coherent event is present. Due to the length of the STA window, the onset of the event is somewhat smeared out. This allows to stack the characteristic functions without correcting for moveout. The only condition is, that the STA window is longer than the maximum theoretical moveout along the considered channel range. E.g. for a velocity of 1500 m/s and an STA window of 0.3 s length, the maximum offset is 450 m.
An important decision is how many channels to use for stacking the characteristic functions. If we choose a very large aperture, small events will be diluted by all the channels that did not register it. If, on the other hand, we choose a too small aperture, local noise might dominate the stacks. Stacking over 100 channels gave good results when tested on events with multiple sizes. A sliding window with length 100 and step size 50 was shifted along the spatial axis of the STA/LTA matrix. The linear stacks corresponding to Fig. 4-1 are

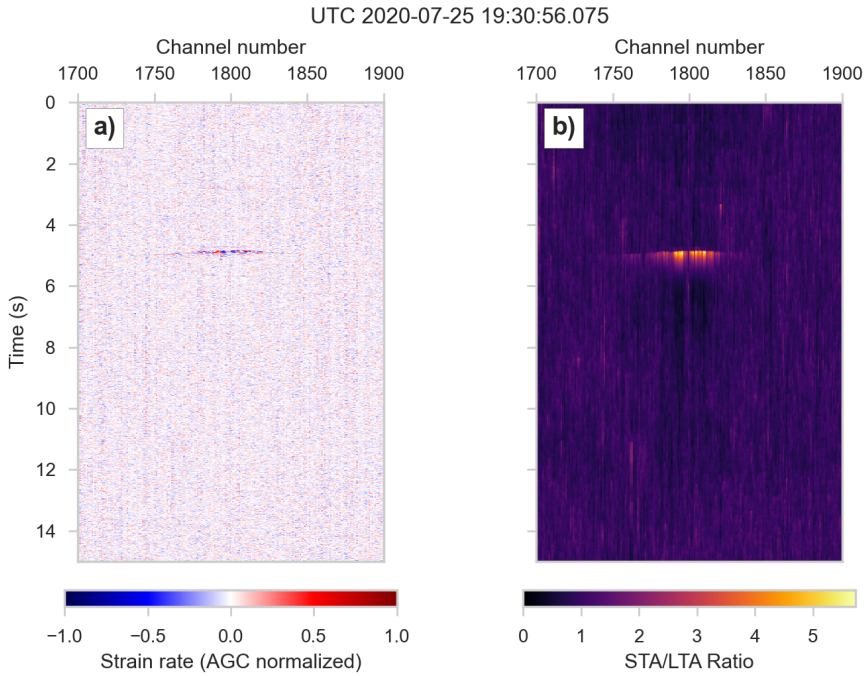


Figure 4-1: Small cryoseismic event and the corresponding characteristic function. The apex of the event is located at channel 1800 and it has a total extent of about 400 m.

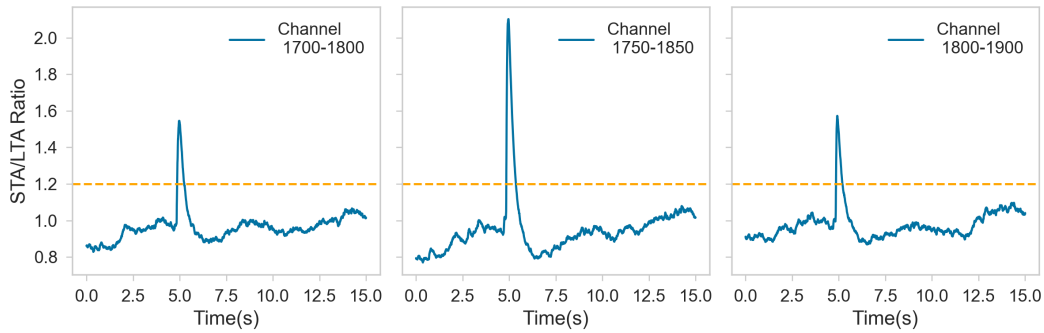


Figure 4-2: Linearly stacked characteristic functions for Fig. 4-1. Each stack comprises 100 channels and there is 50% channel overlap between consecutive subplots.

shown in Fig. 4-2. The highest STA/LTA ratio is obtained for the stack of channels 1750 to 1850. This is logical since that exactly corresponds to the spatial extent of the event. However, the other two windows also detect the event, even though it does not cover the whole channel ranges of those windows. Instead of linear stacking, PWS was also tested. In general, it gave similar results for stacking characteristic functions with high SNR. Therefore, the simpler and faster linear stack was used for triggering. The outlined algorithm was run on a single day of data at a time, in total for 7 days (July 15 - July 23). If the STA/LTA ratio exceeded a certain threshold, an event was triggered. The start and end times, as well as the corresponding channel range were saved in a catalogue. It is desirable, that triggered

entries on neighbouring channels should be merged together, if they correspond to the same event. Therefore, the resulting catalogue was post-processed. If channel ranges and event time of entries overlapped, they were merged together in a single entry. In this way, we would obtain just a single entry for the event in Fig. 4-2, even though it was triggered three times separately with our windowing method. With a triggering threshold of 1.5, the maximum amount of events triggered for a single day was 1436 (July 15). The least events triggered for a single day was 485 (July 17). Most events were triggered on the upper cable sections above channel 1100. Below, the data quality is significantly more noisy and the few triggered events displayed poor SNRs.

4-1-2 Visualization of feature extraction methods

The processing techniques described in chapter 2 were applied to triggered events in order to test their capabilities of discriminating between different events and noise. Fig. 4-3 shows a cryoseismic event transformed to the $f-k$ and $f-p$ domain, respectively. The cryoseismic event is shown in the appendix in Fig. A-1. Clear moveouts are visible in the $f-k$ domain, with energy travelling from North to South (defined as positive velocity) being more pronounced. The $f-p$ domain displays the frequency-dependence of the positive slownesses, that were scanned by the slant-stack. Two separated regions of high slant-stack energy are visible,

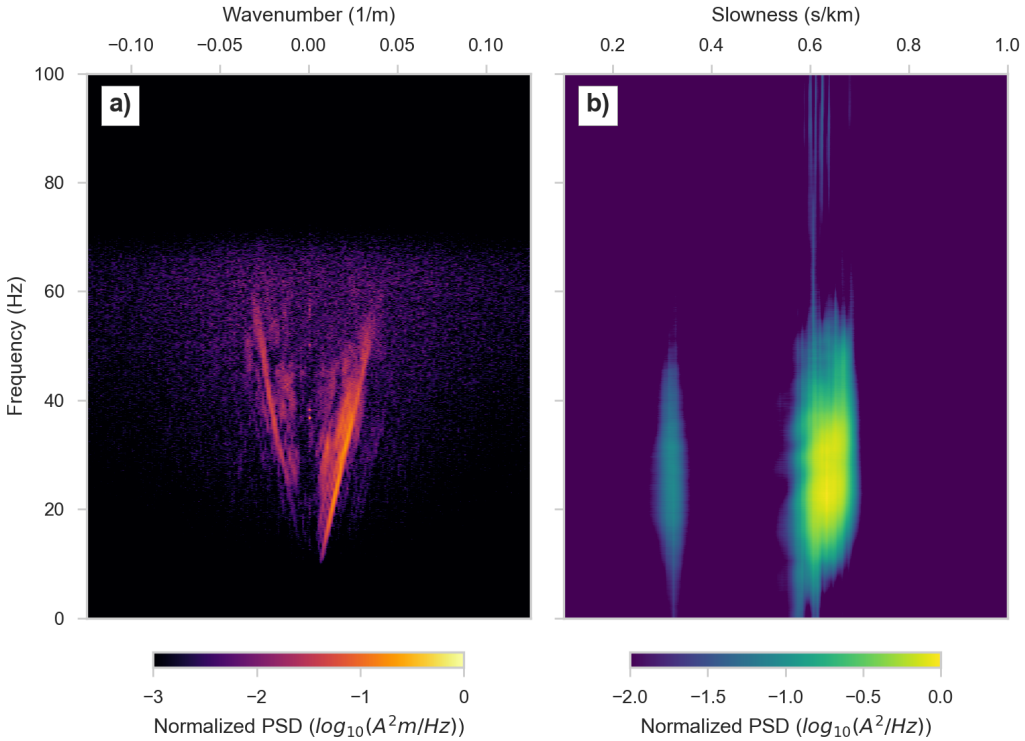


Figure 4-3: Frequency-wavenumber and frequency-slowness spectrum for the cryoseismic event shown in Fig. A-1. The right plot shows energy travelling from North to South. Low values have been clipped for visualization purposes.

likely corresponding to different seismic phases. The more pronounced phase has a apparent velocity of 1600 m/s, while the less pronounced phase is significantly faster with an apparent velocity of 3150 m/s. It is more difficult two discern the two separated velocities in the $f-k$ domain, as the image appears more noisy.
Both the $f-k$ and $f-p$ transform return outputs with a high dimension. Thus, they are not directly suited as input features for clustering. In order to achieve a low-dimensional representation of velocity information, the energy in the columns of the $f-p$ domain was summed up for certain frequency ranges. Summing the energies between 10 and 50 Hz results in the left plot of Fig. 4-4. The output is scaled in such a way, that it sums up to 1. This is to

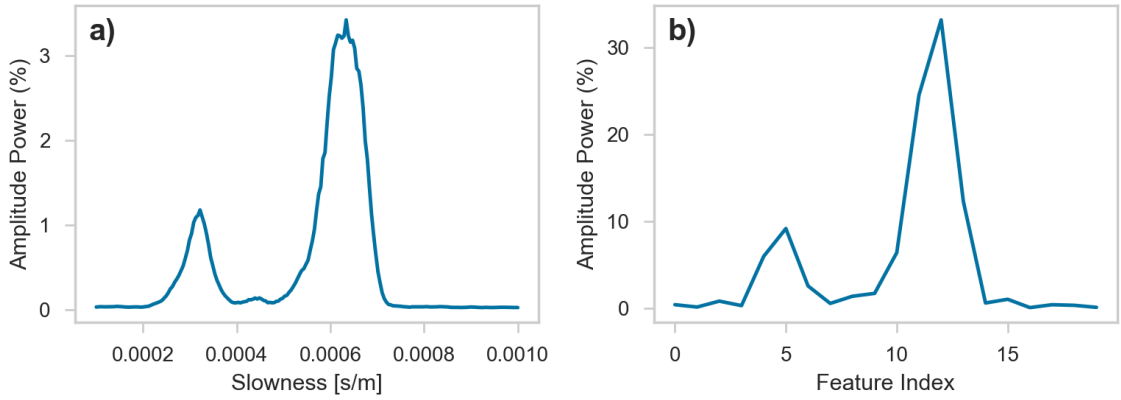


Figure 4-4: Extracted velocity features of the cryoseismic event shown in Fig. A-1. (a) Sum off the energies along the frequency-axis of the $f-p$ domain in Fig. 4-3 between 10 and 50 Hz. (b) Same as (a), but downsampled to 20 values, which can be used as input features.

ensure, that the total energy of the event does not overly influence the features bu rather the relative proportions of velocities present in the sub-window. To further reduce dimensionality, the velocity series is downsampled to 20 values, as shown Fig. 4-4 b). The general shape of the velocity distribution is preserved. Exactly the same process can be followed to obtain features representing the negative slownesses. Using both positive and negative slownesses instead of absolute slowness helps to discriminate between events travelling in opposite directions. A window containing only noise is expected to have a level feature distribution without any pronounced peaks.
For the coherence-based methods, the averaging window is set to the same length as the sub-windows, i.e. 15 s. The short window used to compute the Cross-Spectra matrices was set to 0.6 s. For each sub-window, a frequency-dependent measurement of the array coherence is computed. The short window was set to 0.6 s. Only every 4^{th} channel was used for constructing the covariance matrix, in order to avoid channels with an overlapping gauge length. Accordingly, 50 stations were used for every sub-window to construct the covariance matrix. Fig. 4-5 shows the covariance matrix at 30 Hz for the cryoseismic event in Fig. A-1. The matrix displays high covariance coefficients off the main diagonal. The main diagonal gives the auto-correlation for the corresponding channel. The matrix was then decomposed into its eigenvalues, which were ordered by their magnitude. Fig. 4-5 b) displays the 20 largest eigenvalues of the covariance matrix ordered from largest two smallest. Clearly, the eigenvalue distribution is dominated by the first eigenvalue. For comparison, the eigenvalue distribution

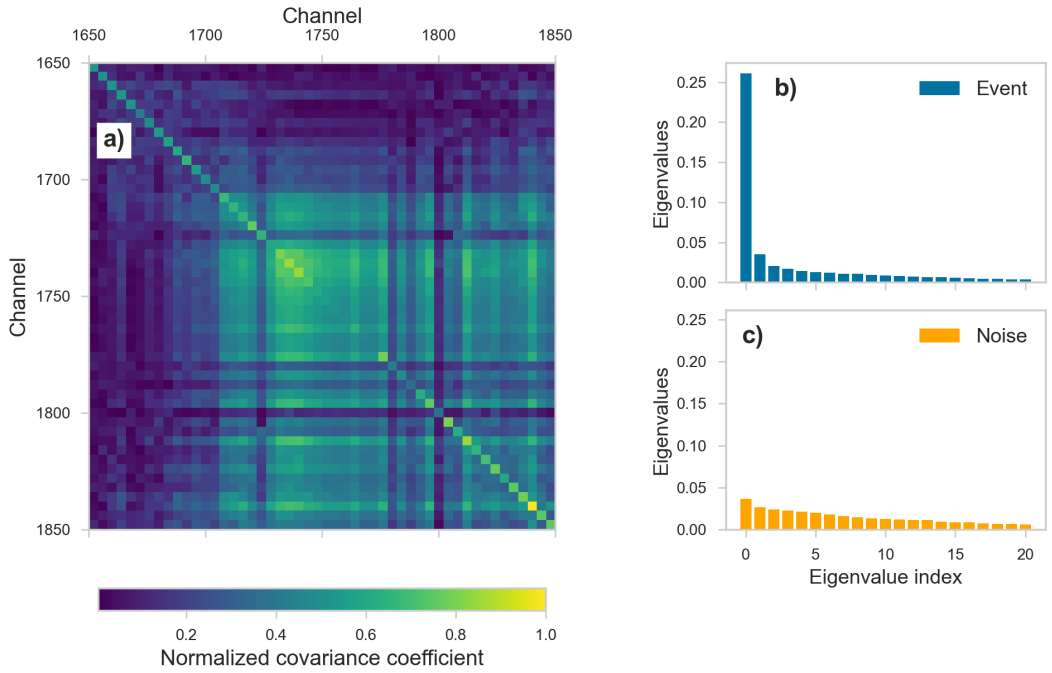


Figure 4-5: (a) Covariance at 30 Hz for the coherent event in Fig. A-1. The covariance matrix was constructed by using every 4th channel. (b) The first 20 ordered eigenvalues for the covariance matrix in (a). The first eigenvalue clearly dominates. For comparison,(b) shows the first 20 ordered eigenvalues for the same channels, but 40 s later, where there is only noise contained in the sub-window.

of a sub-window containing only noise is shown in Fig. 4-5 c). The eigenvalue distribution is quite even, and does not exhibit abnormally high eigenvalues. The sub-window for the noise stems from the exact same cable segment as the cryoseismic event, but 40 s after where no coherent signal was present in the sub-window.

4-1-3 Clustering of wavefield features

The extracted features are used as input for a clustering analysis. The clustering algorithm assigns a label to each sub-window. The exact number of labels that the algorithm assigns, i.e. the number of clusters, has to be set in advance by the analyst. Since each sub-window contains both space and time information, the clustering result can be organized in a matrix, which is from now on referred to as the label matrix. This helps to easily visualize the clustering result. For a given entry in the matrix, the row number provides the spatial location and the column number gives the time stamp. The class membership of the entry can be indicated by a color. Fig. 4-6 shows a 100 min long excerpt of a label matrix on July 25. The clustering for this matrix was obtained based on the eigenvalues and coherency between 10 and 80 Hz of each sub-window. Hierarchical clustering with complete linkage criterion was used and Euclidean distance chosen as the similarity measurement. The number of clusters was set to 6, as will be justified in section 4-1-4. Most entries in the matrix belong to cluster 0. The color for this cluster was set as white to facilitate visualization. Upon visual inspection, it appeared that sub-windows corresponding to cluster 0 do predominantly not

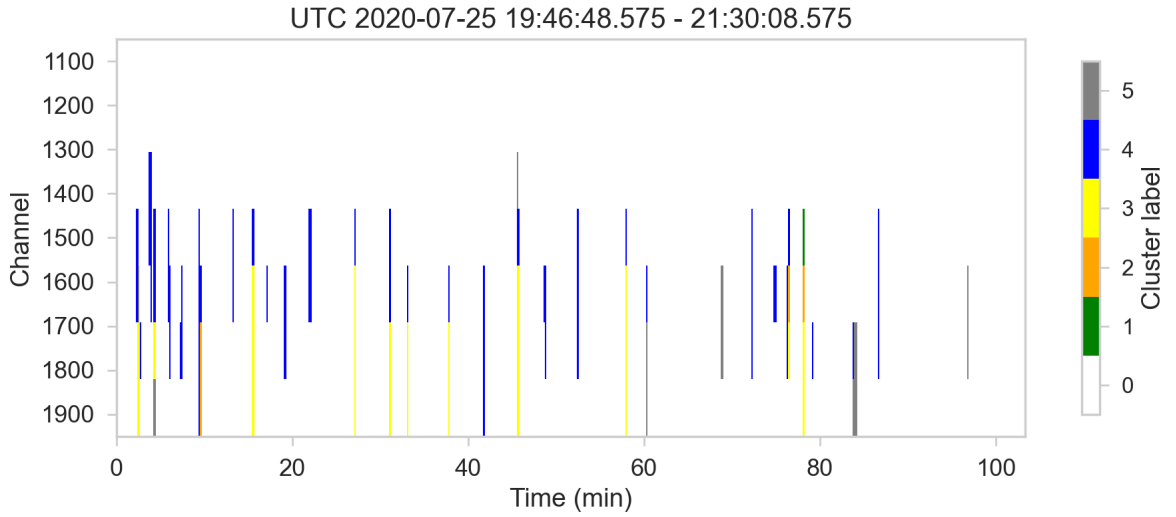


Figure 4-6: Clustering results using coherence-based features and hierarchical clustering for 100 min of data on July 25. The color of a matrix entry indicates the cluster membership of each sub-window.

contain any coherent signal. All other clusters contain sub-windows which display events. Note that the label matrix only shows channels from 1100 upwards. On the lower segment, no events were registered for the displayed time frame. Effectively, all sub-windows belonging to channels lower than channel 1100 were assigned to cluster 0. This was also confirmed by the manual triggering. Therefore, lower channels were discarded for further analysis to speed up processing. Fig. 4-7 also shows a label matrix computed with coherence-based features as input but displays 30 min of data on July 15. Again, hierarchical clustering was utilized with

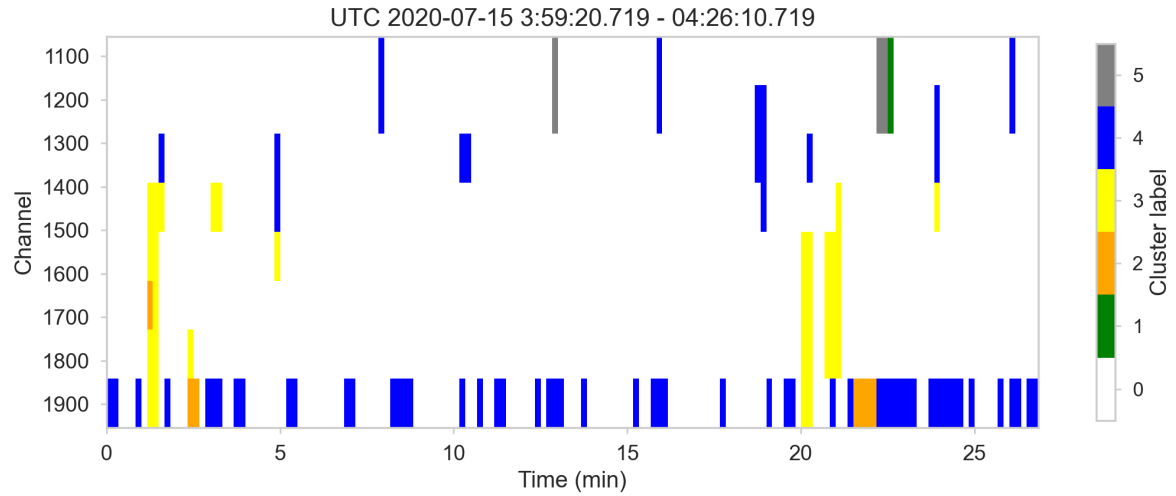


Figure 4-7: Cluster results for 30 minutes of data on July 15 using coherence-based features. The color of a matrix entry indicates the cluster membership of the corresponding sub-window.

complete linkage and Euclidean distance as similarity measure. Icequakes covering most of

the cable extent belong to cluster 3. Sub-windows belonging to cluster 4 mainly occur on the uppermost segment of the cable. Most sub-windows not belonging to cluster 0 are located on the very end of the cable.
Fig. 4-8 shows clustering results for the same time frame as Fig. 4-7, but with velocity features as input instead of coherence-base features. The general pattern is quite similar but there are

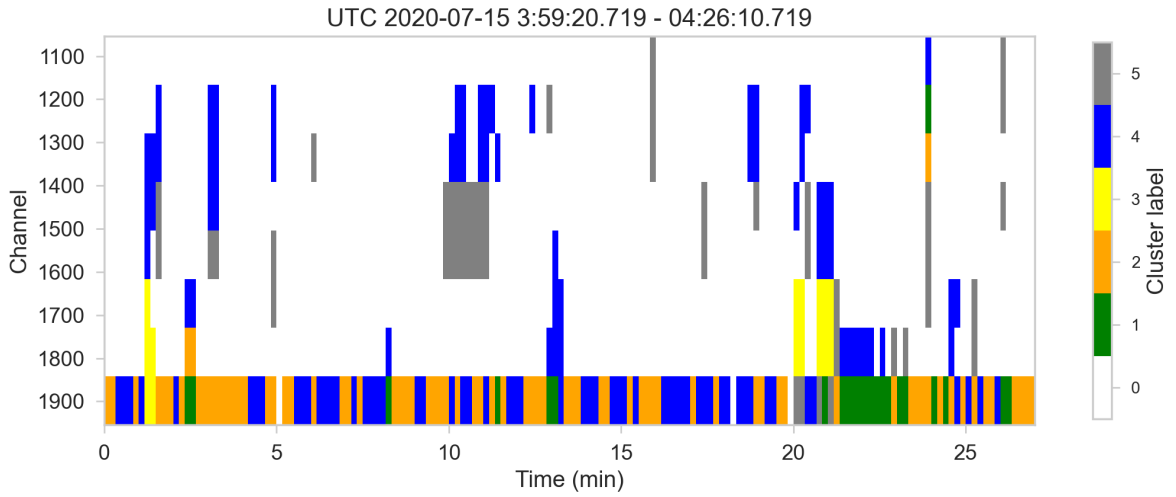


Figure 4-8: Cluster results for 30 min of data using features derived from spectral velocity analysis. The color of a matrix entry indicates the cluster membership of the corresponding sub-window.

distinct differences. Almost all sub-windows located on the very end of the cable are assigned to a cluster other than cluster 0. Some of the larger events (vertical stripes) are similar for both matrices. However, they are sub-divided between two clusters in Fig. 4-8 (blue and yellow), while they do only belong to a single cluster in Fig. 4-7.
Fig. 4-9 shows representative events for each cluster in Fig. 4-8. Cluster 0 contains sub-windows with no events and corresponds to background noise. Cluster 1 and cluster 2 both belong to the excessive amounts of events detected on the uppermost channels. Cluster 1 exhibits larger moveouts and contains multiple repeating waveforms. In both cases, there sometime appear moveouts with significantly higher velocities but they are weaker and attenuated quickly (see first arrival for the representative event of Cluster 1). Events in cluster 3 are predominantly of large extent. Note that they are sometimes shared with cluster 4 (see 4-8). As mentioned before, this does not happen for the coherence-based clustering. Cluster 4 appears all over the array. It shows highly varying waveforms but they predominantly travel down the fiber (from North to South).

4-1-4 Cluster quality control

The number of cluster to search for is a crucial parameter for the clustering analysis. This is often done empirically and by trial and error. There exist a few quantitative ways to determine an appropriate selection of the number of clusters in the data. Fig. 4-10 demonstrates the so-called elbow test applied to the label matrix of the velocity features shown in Fig. 4-8. Hierarchical clustering was run multiple times with a different number of clusters as target.

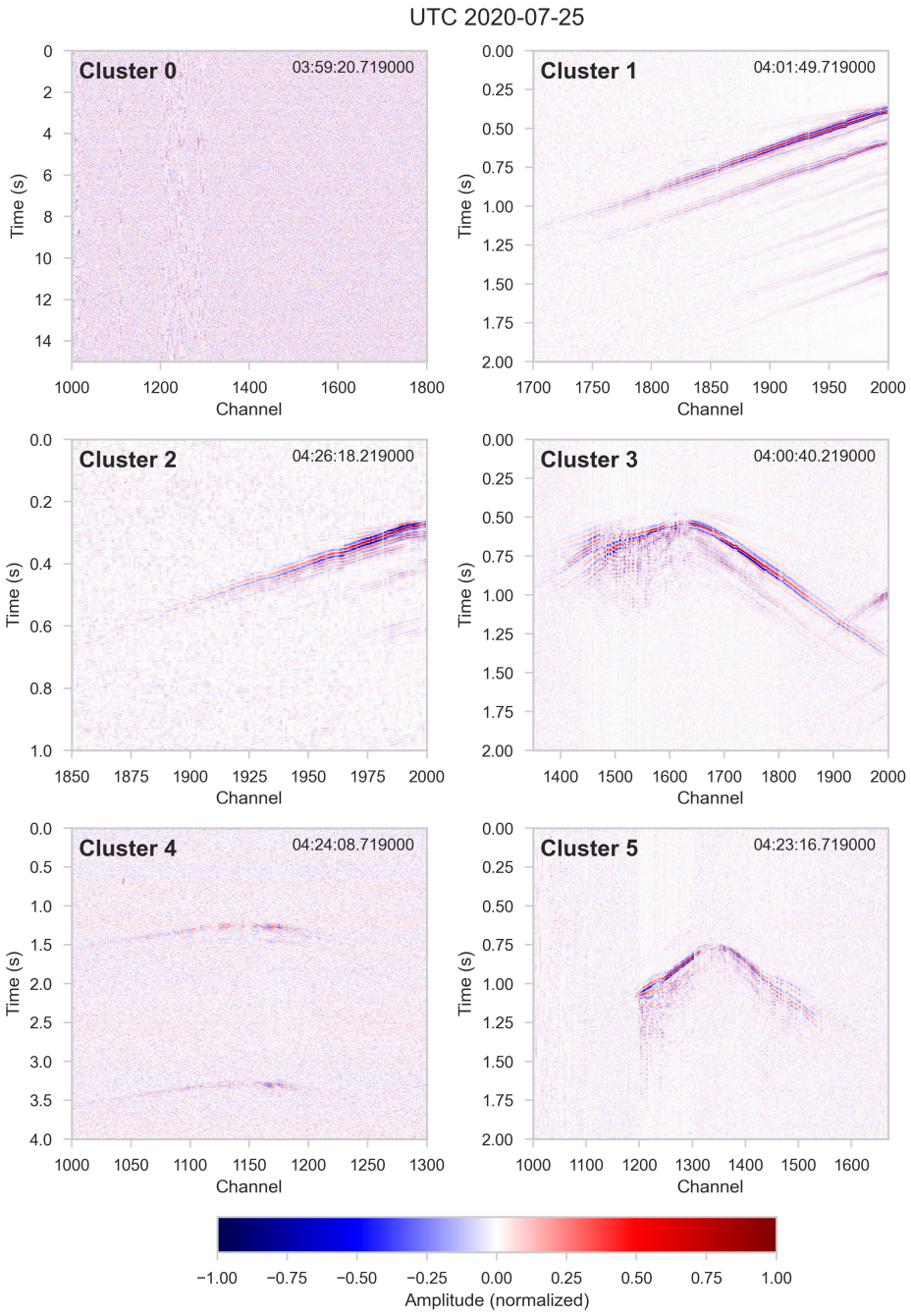


Figure 4-9: Representative events for the clustering result in Fig. 4-8

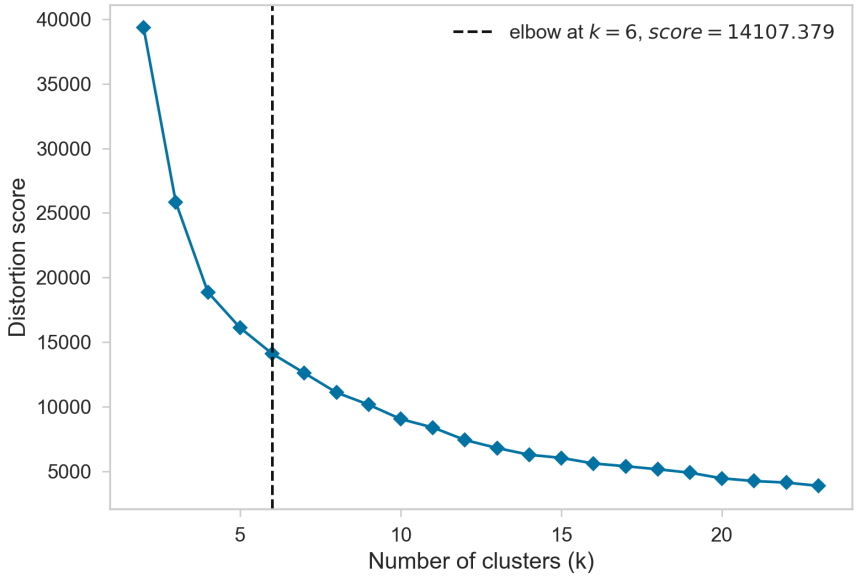


Figure 4-10: Application of the elbow criterion to determine the appropriate number of clusters in the data. For each number of clusters k, clustering is performed. The optimal number of clusters reduces the distortion score while not introducing an excessive amount of clusters.

I.e. the algorithm is told to cluster the data into to groups, then into three and so forth. Alternatively, the dendrogram could be computed once and cut at different distances to achieve the same outcome. The distortion score is computed each time after the clustering algorithm is performed. For each sample, the Euclidean distance to its representative cluster center is calculated. The sum of the distances of all samples gives the distortion score. Consequently, it gives a measure how compact clusters are. The elbow test is performed in order to find a trade-off between number of clusters and their compactness. Based on the distortion score, an optimal number of clusters was around 6.
To validate the cluster assignments, the silhouette score was computed for each sample. Fig. 4-11 visualizes the silhouette scores for the same velocity features, for which the elbow test was performed. The average silhouette score of all samples is 0.63, as indicated by the red dotted line. The biggest cluster by far is cluster 0, which was already recognizable in Fig. 4-8. Cluster 4 and 5 contain a few samples with a negative silhouette coefficient, which means that these samples are not well separated from neighbouring clusters. Most samples show a positive silhouette coefficient value.
For quality control, it can help to visualize high-dimensional features by projecting them onto two or three dimension. Subsequently, they can be plotted in order to gain some understanding of the data structure. Projection onto lower dimension can be done with linear and non-linear methods. A linear method which is often chosen is the principal component analysis, where data is projected to its main principle components, where variance is maximized [Shi et al., 2021]. Fig. 4-12 displays the projection onto two principle axes for the features shown in Fig. 4-4. The plot has to be taken with great care, as the first two principal components only account for 50.2% of the total variance in the feature space. Thus, points located close-by in the 2-D space could be completely separated in higher dimensions. Consequently,

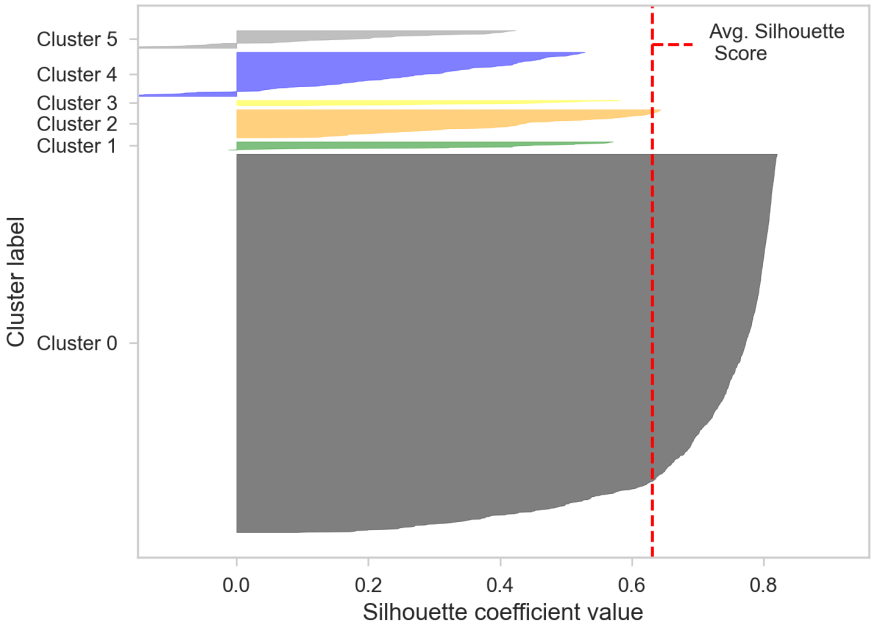


Figure 4-11: Silhouette coefficient computed for each sample. The samples are grouped with their cluster and displayed descending order. The average silhouette score of all samples is indicated by the red dotted line. Cluster 0 contains the majority of samples.

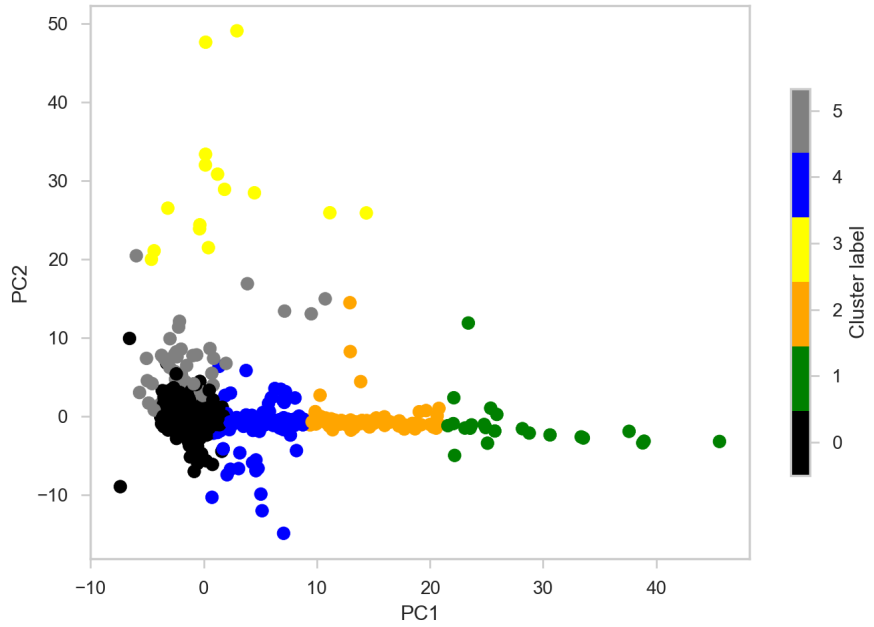


Figure 4-12: Projection of velocity features onto two dimension using principle component analysis. The original feature space had 60 dimensions. The cluster assignment (based on the full feature dimension) is indicated by the color of samples. The first two principal components account for 50.2 % of the total variance in the feature space.

a projection of the input features to two or three dimension is not well suited for analysis purpose in this study. However, even when projected onto only two axes, distinct groupings of points are already recognizable. This seems to be well captured by the clustering results, which were obtained for the high-dimensional input. There is only partly overlap between clusters 0, 4 and 5. This is also displayed in the visualization of the silhouette score, where clusters 4 and 5 contain a few values with negative silhouette score (see Fig. 4-11).

4-2 Discussion

4-2-1 Influence of coupling conditions

The Rhonegletscher dataset contains a plethora of cryoseismic events. Already the manual analysis gave an idea about the wide range of signals present in the data. One major difficulty with the dataset is the modest data quality. The cable was not actively trenched during installation, mainly because of the difficult terrain. Consequently, meteorological effects like wind and surface melt have partly corrupted the data quality. This is especially true for the lower cable segment (below channel 1100) which was lying on bare ice. Overall, the data is much quieter at night and for the upper cable segment (also visible in Fig. 3-3). During the day, sunshine heats up the glacier surface, causing surface melt. During night, the temperature is much lower and commonly freezing conditions prevail. The upper cable segments exhibit a better signal quality, as they were covered by snow. Due to the black coating of the fiber, the cable melted itself into the snow. The effect of the coupling conditions is dramatic, as almost all events are detected on the upper cable segment. The cable lying on top of the bare ice did register only a small fraction of all events present in the Rhonegletscher dataset. This does not at all mean that no events are occurring on the lower part on the glacier but probably the noise level overshadows most signal of interest. Previous studies by [Walter et al., 2020] and [Gräff and Walter, 2021] investigated areas on the lower extent of the glacier and recorded plenty of events. The effect of the coupling conditions agrees well with a recent paper from [Hudson et al., 2020], where DAS recordings were acquired on the Ross Ice Shelf in Antarctica. They tested different array configurations to investigate coupling conditions and array geometries. They recorded data for 6h in total. Their linear unburied fiber did not register a single icequake in that time period, while 139 were detected with the buried linear cable. It appears that good coupling of the fiber and protection from meteorological influences play a crucial role for microseismic studies on icy terrain.

4-2-2 Limitations of conventional triggering

Compared to other seismological sensors, DAS data typically does not display good SNRs on individual channels. In the very beginning of the project, triggering was performed on single channels [Grimm, 2021]. This resulted in an event catalogue, where the vast majority of entries were false detections. False detections on single channels are mainly due to the abundance of high-amplitude localized noise with sharp onsets. This phenomenon is visible in Fig. 4-1, where high STA/LTA ratios are not limited to the icequake but also occur when

there is no coherent signal over the array segment. A prime advantage of DAS data is the dense spatial coverage. Coherence across the DAS array can be exploited in various ways. For manual triggering, this was achieved by stacking of characteristic functions. The idea is the following: A high STA/LTA ratio must be present over a range of channels, in order to trigger an event. This is somewhat similar to voting schemes in conventional seismometer arrays. Stacking of characteristic functions successfully suppresses the false triggers caused by localized noise. Often, this high-amplitude noise with sharp onsets can be traced over two or three channels. This is because neighbouring channels carry some redundancy due to their overlapping gauge length.
A very important choice to make is choosing the number of channels over which the stacks are computed. If the cable extent is too long, small events will be diluted by all the other channels that did not record the event. E.g. stacking characteristic functions over 1000 channels will not work well for detecting events which are only visible on 100 consecutive channels. On the other hand, selecting only a small number of channels for stacking will amplify the effect of high-amplitude noise with sharp onsets. Stacking over segments of 100 m channels achieves a good trade-off and provided satisfactory results. In the beginning of the manual analysis, all triggering parameters were set quite conservatively. The goal was to find only events with high SNRs and reduce the number of false triggers. Already with an onset threshold of 1.5, over a thousand of events were detected for some days. The conventional workflow would be to fine-tune the parameter combinations (STA window length, LTA window length, trigger onset threshold, trigger offset threshold, number of channels for stacking) of the STA/LTA algorithm. Typically, this is done empirically by testing different combinations of parameters, trying to maximise the number of detected events while keeping the proportion of false detections at a minimum. However, with a dataset of this volume, this turns out to be extremely cumbersome. The complete Rhonegletscher dataset is stored on the AWS cloud and only small chunks of it can be stored on the hard drive of an ordinary personal computer. It is difficult to access a certain file on demand, as it must be downloaded first, which introduces lag time. A possibility would be to save all waveforms corresponding to triggered events and store them in a separate location. These waveforms could then be downloaded to a local machine for further processing. This would result in in an enormous number of smaller files which must be read in and processed one by one. For such large data volumes as obtained by DAS monitoring experiments, this quickly becomes impractical and highly time-consuming. It might be fully valid to use interesting data chunks found by a triggering algorithm as the input dataset for an automated analysis. The automated processing would then be performed on the pre-selected data and not the complete dataset. This was done by [Gräff and Walter, 2021], who conventionally triggered icequakes recorded by seismometers and afterwards applied clustering to group the detected events based on their waveform similarity. For the aforementioned reasons, automated processing of DAS monitoring experiments is highly desired and possibly indispensable.

4-2-3 Comparing different feature representations

The objective of the automated processing is not to only to detect events, but automatically sort them into groups, where members of the same group are similar in some way. [Bormann, 1998] mentions that ambient seismic noise recordings can be described by the following characteristics: energy, direction, velocity and frequency content. Hence, it seems

plausible to extract features which describe these characteristics in some way, as was done by [Chamarczuk et al., 2019] for a large-N array. Many array processing techniques directly compute related attributes. [Chamarczuk et al., 2019] used sum-and-delay beamforming, as well as a source localization algorithm to compute features for 10 s long time windows. Both techniques are not available for the Rhonegletscher data, as the one-dimensional nature of the linear DAS array permits unbiased estimations of backazimuths. In this work, mainly frequency-dependent methods are discussed. The motivation is that different sources for cryoseismic events can commonly be discriminated by their frequency content (e.g. surface crevassing and stick-slip icequakes). Besides frequency, velocity of events is highly relevant, as it might help to differentiate between different seismic phases. [Gräff and Walter, 2021] use a P-wave velocity of 3750 m/s and an S-wave velocity of 1875 m/s in ice for estimating source locations of basal icequakes on the Rhonegletscher. [Walter et al., 2020] extract a shallow surface Rayleigh wave velocity of 1700 m/s on the ablation zone of the Rhonegletscher. Frequently, apparent velocities around 1650 m/s are seen for events on the data investigated in this work (e.g. see Fig A-3). Seismic velocities for DAS measurements on the accumulation zone could possibly be lower than previous estimations on the ablation zone, as the fibre was coupled to snow and not as deeply buried as in the experiment by [Walter et al., 2020]. Note, that it is not possible to determine the true phase velocity with a linear DAS array. Instead, only an apparent velocity, also depending on the incidence angle of the incoming wavefront, can be estimated. Apparent velocity therefore is a combination of true phase velocity and source direction, with an infinite amount of possible combinations resulting in the same apparent velocity.

A frequency-velocity description of a sub-window can be obtained via different methods. Here, the $f-k$ and the $\tau-p$ transform were tested. The output from both transforms still lies in a high-dimensional space. A simple way to extract features from the $f-k$ domain is to smooth the image and average over frequency and wavenumber bins. Effectively, this would result in a very course version of the $f-k$ image. This was tested but the clustering results were not convincing. The clustering based on $f-k$ features generally did not discriminate well between different events and mainly clustered noise. Fig. 4-3 demonstrates, that phases with different velocity are much better separated in the $f-p$ domain. The general velocity information of the event is well preserved in the low-dimensional feature representation (see Fig. 4-4 b). Consequently, the clustering results based on the $f-p$ domain provided encouraging results. One fundamental question is, how applicable both methods are given the array geometry. The whole DAS cable does not trace a perfectly straight line, which is a requirement for both $f-k$ and $\tau-p$ transform. However, it turns out that most of the upper half of the array can be well approximated by straight cable section (see Fig. 3-1b). Only one small semi-circle cannot be well approximated by linear sub-sections. Nonetheless, even when that approximation is violated, the transform can still be mathematically computed even if the physical interpretation does not fully hold. In that case similar events from the same source location should still map onto a similar feature representation.

PSD was also tested as a descriptive feature but not shown in the result section. For a sub-window, the PSD was computed on a channel-by-channel basis and then linearly stacked. Unfortunately, the clustering result did not discriminate cryoseismic events from noise. The general pattern showed, that sub-windows belonging to the same channel ranges were grouped together. This likely tells, that the feature representation of PSDs stacked over a segment of channels is highly related to the coupling condition or noise exposure of that channel segment. In essence, time-windows from the same spatial location are group together, no matter if they

contain an event or not.
Finally, frequency-dependent measures of array coherence were used for clustering. Before this report, this was only done by [Shi et al., 2021]. In contrast to this work stands, that they use extremely long time windows (in the order of days) for computing array coherence. They stated as a question for further research, if coherence-based features could also be used for clustering windows of short-time duration. In this work, the theoretical limits of short-time duration for covariance matrix analysis are tested. The practical computation of the covariance matrix requires averaging over many Cross-Spectra matrices.The calculations of the Cross-Spectra matrices themselves are based on a short time sliding window, for which the slowest wave should be able to traverse the complete array segment. E.g. a plane wave with an apparent velocity of 1400 m/s would travel 800 m (200 channels) along the fiber in 0.57 s. Therefore, a time duration of 0.6 s was set for the short-term window. Only every 4^{th} channel is selected for computing the covariance matrix. This is to avoid redundant signal due to the large gauge length. The redundant signal would cause artificially high cross-correlations between directly neighbouring channels. For a sub-window with an extent of 200 channels, only 50 channels are used for computing the covariance matrix. The number of snapshot (Cross-Spectra matrices computed for the 0.6 s long time windows) for estimating the covariance matrix should be at least equal to the number of channels. If set exactly equal and with 50 % overlap between the 0.6 s long short time windows, this gives an averaging window of 15 s. Hence, this is the absolute minimum for the time resolution. The only way to further decrease this limit would be to use less channels or smaller spatial extents for sub-windows. A channel range of 100 channels with 33 of those used for computing covariance matrices was tested but did not deliver better results.
For the Rhonegletscher dataset, features with high values for coherency primarily correspond to coherent cryoseismic events. Fig. 4-7 gives a good representation of the clustering results on coherence-based features. Many events are detected in the rather short time-frame. Smaller events, as well as larger events are detected and grouped in separate clusters. Exactly the same time frame is shown in Fig. 4-8 but clustered on velocity-derived features only. This allows for a direct comparison between the two approaches is. On a brief glance, the overall pattern seems to be similar. This shows that the general workflow is sound and gives similar output for input features derived with different methodologies. However, there are also differences between the two figures. The clustering based on velocity assigns more sub-windows to clusters other than cluster 0. Especially the uppermost cable section is almost constantly assigned to a cluster containing events. After manual quality control, this seems to be more plausible. Fig. A-4 displays a 180 s long spectrogram, giving an idea about the vast amount of events occurring on that channel range in the early morning of July 15, 2020. One drawback of the clustering based on velocity seems to be that many events propagating over larger channel ranges are cut into separate clusters (e.g. compare cluster 3 for both Fig. 4-8 and Fig. 4-7. This is probably because the sign of the velocity changes at the apex of an event, i.e. energy travelling North-South will end up ind a different cluster than energy travelling South-North. Inherently, this effect does not appear for coherence-based clustering. It turns out that cluster 4 in Fig. 4-8 mainly contains events or parts of events that propagate down-glacier. On the other hand cluster 3 contains events propagating up-glacier.
The nature of the events on the uppermost part of the glacier is still unclear at this point. It appears very localized in time on the morning of July 15. Fig. A-2 and Fig. A-3 show an ice-quake belonging to cluster 2 in the $t-x$ and $f-p$ domain respectively. The event propagating at an apparent velocity of around 1650 m/s and has strong frequency contents above 60 Hz.

In the $t-x$ domain, a slightly faster phase is visible for small offsets. Possibly, this could correspond to a P- and S-wave onset. The velocity of the slow phase does not seem dispersive, at least between 10 to 60 Hz. In general, the velocities seems somewhat slow. But as mentioned before, this could be due to the snow cover, where waves should be significantly slower than on ice. A completely different source could be a thunderstorm. [Zhu and Stensrud, 2019] recorded thunders on a DAS array and the spectrogram looked quite similar to the one shown in Fig. A-4. However, this option seems unlikely due to the prevalence of events and the duration during which they occur. More detailed analysis of these events is outside the scope of this work and subject to further studies.

A practically very challenging part but crucial to the results is the cluster analysis itself. It is inherently difficult to validate clustering results, as usually ground-truth data is lacking. Hence, only vague quantifications of cluster quality can be computed, such as the silhouette coefficient score. Still, the quality control shown in section 4-1-4 gives some confidence in the obtained clustering results. Generally speaking, it is equally important to do a manual quality control to see if the clustering results are sensible and provide expected results.

Fig. 4-9 shows representative events for the clustering of Fig. 4-8. Obviously this does not display all the types of events present in the Rhonegletscher dataset. It is a proof of concept that we can apply unsupervised classification to the dataset and retrieve meaningful clusters containing cryoseismic events.

Chapter 5

Conclusion and Outlook

This chapter summarizes the main findings of the thesis project. Suggestions for further research are given.

In the summer 2020, ETH researchers installed a DAS system on the Rhonegletscher (Switzerland). The 9 km long fiber-optic cable covered the whole extent of the glacier from the accumulation down to the ablation zone. This is one of the first times, that a complete glacier was monitored at once by a DAS system. DAS data was acquired for one month continuously at 2000 channels with a 1 kHz sampling rate. The resulting dataset has a total size of (~18 TB). The present work is one of the first, that actively investigated the Rhonegletscher dataset.
DAS monitoring experiments result in large data volumes, rendering manual event picking and classification practically unfeasible. Thus, new automated processing tools are required. Unsupervised clustering has been successfully applied previously to classify noise in a DAS dataset [Martin et al., 2018].
This work investigated the possibility of utilizing unsupervised clustering methods for event detection and classification on the Rhonegletscher dataset. The raw data displays meteorological effects as well as different types of cryoseismic events such as surface crevassing and basal icequakes. Most cryoseismic events are only visible on fractions of the array and therefore the data was divided into smaller sub-windows in space and time. A hybrid workflow was developed that combines array processing techniques with unsupervised clustering. The array processing output serves as the input for the clustering analysis. Different processing techniques were discussed to obtain a low-dimensional feature representation of the raw data. Frequency-velocity and coherence-based feature representations were selected as the most promising.
The developed method was then tested on the Rhonegletscher dataset. Clustering was performed both on velocity-derived and coherence-based input features. The similarities and differences between the results of the two approaches are compared and discussed. In general, both feature extraction methods lead to sensible results, showing the stability of the general workflow. In total, 6 clusters were determined as the optimal number of clusters for the data processed as part of this work. Of those clusters, 5 correspond to cryoseismic events and the other one shows background noise only. Cluster results were validated by standard cluster

metrics such as the distortion score and the silhouette score. Both metrics validated the clustering results. Additionally, manual quality control was performed by checking the obtained clusters for plausibility.
This work demonstrates, that unsupervised clustering can be applied to DAS data in order to detect and classify signals of interest. However, these results are only preliminary and need further validation. As a starting point, the obtained clusters need to better further investigated and understood. As a logical next step, larger portions of the Rhonegletscher data should be processed with the developed method in order to proof its viability. Additionally, other feature extraction methods and combination of those could be tested instead of the velocity and coherence-based feature representations. Examples for other feature extraction methods could be statistical methods such as change point detection and kurtosis.

Bibliography

[Aster and Winberry, 2017] Aster, R. C. and Winberry, J. P. (2017). Glacial seismology. *Reports on Progress in Physics*, 80(12):126801.

[Bartlett, 1954] Bartlett, M. S. (1954). A note on the multiplying factors for various χ 2 approximations. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 296–298.

[Bensen et al., 2007] Bensen, G., Ritzwoller, M., Barmin, M., Levshin, A. L., Lin, F., Moschetti, M., Shapiro, N., and Yang, Y. (2007). Processing seismic ambient noise data to obtain reliable broad-band surface wave dispersion measurements. *Geophysical Journal International*, 169(3):1239–1260.

[Bergen et al., 2019] Bergen, K. J., Johnson, P. A., de Hoop, M. V., and Beroza, G. C. (2019). Machine learning for data-driven discovery in solid earth geoscience. *Science*, 363(6433).

[Bormann, 1998] Bormann, P. (1998). Conversion and comparability of data presentations on seismic background noise. *Journal of Seismology*, 2(1):37–45.

[Chamarczuk et al., 2019] Chamarczuk, M., Nishitsuji, Y., Malinowski, M., and Draganov, D. (2019). Unsupervised learning used in automatic detection and classification of ambient-noise recordings from a large-n array. *Seismological Research Letters*, 91(1):370–389.

[Dietrich, 1990] Dietrich, M. (1990). An algorithm for the plane-wave decomposition of point-source seismograms. *GEOPHYSICS*, 55(10):1380–1385.

[Dondurur, 2018] Dondurur, D. (2018). Chapter 5 - preprocessing. In Dondurur, D., editor, *Acquisition and Processing of Marine Seismic Data*, pages 241–312. Elsevier.

[Fernández-Ruiz et al., 2020] Fernández-Ruiz, M. R., Soto, M. A., Williams, E. F., Martin-Lopez, S., Zhan, Z., Gonzalez-Herraez, M., and Martins, H. F. (2020). Distributed acoustic sensing for seismic activity monitoring. *APL Photonics*, 5(3):030901.

[Gerstoft et al., 2012] Gerstoft, P., Menon, R., Hodgkiss, W. S., and Mecklenbräuker, C. F. (2012). Eigenvalues of the sample covariance matrix for a towed array. *The Journal of the Acoustical Society of America*, 132(4):2388–2396.

[Gersztenkorn and Marfurt, 1999] Gersztenkorn, A. and Marfurt, K. J. (1999). Eigenstructure-based coherence computations as an aid to 3-d structural and stratigraphic mapping. *Geophysics*, 64(5):1468–1479.

[Gimbert et al., 2016] Gimbert, F., Tsai, V. C., Amundson, J. M., Bartholomaus, T. C., and Walter, J. I. (2016). Subseasonal changes observed in subglacial channel pressure, size, and sediment transport. *Geophysical Research Letters*, 43(8):3786–3794.

[GLAMOS, 2017] GLAMOS (2017). The swiss glaciers 2013/14 and 2014/15. *Glaciological Report No. 135/136 of the Cryospheric Commission (EKK) of the Swiss Academy of Sciences (SCNAT).*

[GLAMOS, 2020] GLAMOS (2020). The swiss glaciers 2017/18 and 2018/19. *Glaciological Report No. 139/140 of the Cryospheric Commission (EKK) of the Swiss Academy of Sciences (SCNAT).*

[Gräff and Walter, 2021] Gräff, D. and Walter, F. (2021). Changing friction at the base of an alpine glacier. *Scientific reports*, 11(1):1–10.

[Grimm, 2021] Grimm, J. (2021). Cryoseismic event analysis on distributed strain recordings. unpublished.

[Haeberli et al., 2007] Haeberli, W., Hoelzle, M., Paul, F., and Zemp, M. (2007). Integrated monitoring of mountain glaciers as key indicators of global climate change: the european alps. *Annals of Glaciology*, 46:150–160.

[Harris et al., 2020] Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825):357–362.

[Harrison et al., 2018] Harrison, S., Kargel, J. S., Huggel, C., Reynolds, J., Shugar, D. H., Betts, R. A., Emmer, A., Glasser, N., Haritashya, U. K., Klimeš, J., Reinhardt, L., Schaub, Y., Wiltshire, A., Regmi, D., and Vilímek, V. (2018). Climate change and the global pattern of moraine-dammed glacial lake outburst floods. *The Cryosphere*, 12(4):1195–1209.

[Hartog, 2017] Hartog, A. H. (2017). *An introduction to distributed optical fibre sensors.* CRC press.

[Holtzman et al., 2018] Holtzman, B. K., Paté, A., Paisley, J., Waldhauser, F., and Repetto, D. (2018). Machine learning reveals cyclic changes in seismic source spectra in geysers geothermal field. *Science Advances*, 4(5).

[Hudson et al., 2020] Hudson, T. S., Baird, A. F., John-Michael, K., Kufner, S.-K., Brisbourne, A. M., Smith, A. M., Butcher, A., Chalari, A., and Clarke, A. (2020). Distributed acoustic sensing (das) for natural microseismicity studies: A case study from antarctica. *Earth and Space Science Open Archive ESSOAr.*

[Jain and Dubes, 1988] Jain, A. K. and Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc.

[Jousset et al., 2018] Jousset, P., Reinsch, T., Ryberg, T., Blanck, H., Clarke, A., Aghayev, R., Hersir, G. P., Henninges, J., Weber, M., and Krawczyk, C. M. (2018). Dynamic strain determination using fibre-optic cables allows imaging of seismological and structural features. *Nature Communications*, 9(1):2509.

[Khalid et al., 2014] Khalid, S., Khalil, T., and Nasreen, S. (2014). A survey of feature selection and feature extraction techniques in machine learning. In *2014 science and information conference*, pages 372–378. IEEE.

[Lawley, 1956] Lawley, D. (1956). Tests of significance for the latent roots of covariance and correlation matrices. *biometrika*, 43(1/2):128–136.

[Lindsey and Martin, 2021] Lindsey, N. J. and Martin, E. R. (2021). Fiber-optic seismology. *Annual Review of Earth and Planetary Sciences*, 49(1):309–336.

[Lowrie and Fichtner, 2020] Lowrie, W. and Fichtner, A. (2020). *Fundamentals of Geophysics*. Cambridge University Press, 3 edition.

[Martin et al., 2018] Martin, E. R., Huot, F., Ma, Y., Cieplicki, R., Cole, S., Karrenbach, M., and Biondi, B. L. (2018). A seismic shift in scalable acquisition demands new processing: Fiber-optic seismic signal retrieval in urban areas with unsupervised learning for coherent noise removal. *IEEE Signal Processing Magazine*, 35(2):31–40.

[Maurer et al., 2020] Maurer, J. M., Schaefer, J. M., Russell, J. B., Rupper, S., Wangdi, N., Putnam, A. E., and Young, N. (2020). Seismic observations, numerical modeling, and geomorphic analysis of a glacier lake outburst flood in the himalayas. *Science Advances*, 6(38).

[Mestayer et al., 2012] Mestayer, J., Cox, B., Wills, P., Kiyashchenko, D., Lopez, J., Costello, M., Bourne, S., Ugueto, G., Lupton, R., Solano, G., Hill, D., and Lewis, A. (2012). *Field trials of distributed acoustic sensing for geophysical monitoring*, pages 4253–4257. Society of Exploration Geophysicists.

[Paitz et al., 2020] Paitz, P., Edme, P., Gräff, D., Walter, F., Doetsch, J., Chalari, A., Schmelzbach, C., and Fichtner, A. (2020). Empirical investigations of the instrument response for distributed acoustic sensing (das) across 17 octaves. *Bulletin of the Seismological Society of America*.

[Parker et al., 2014] Parker, T., Shatalin, S., and Farhadiroushan, M. (2014). Distributed acoustic sensing - a new tool for seismic applications. *First Break*, 32.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[Podolskiy and Walter, 2016] Podolskiy, E. A. and Walter, F. (2016). Cryoseismology. *Reviews of Geophysics*, 54(4):708–758.

[Pretorius et al., 2021] Pretorius, A., Smith, E., Booth, A., Christofferson, P., Nowacki, A., de Ridder, S., Schoonman, C., Clarke, A., Hubbard, B., Chudley, T., et al. (2021). Application of machine learning methods to identify englacial seismicity in a distributed acoustic sensing dataset from store glacier, west greenland. In *EGU General Assembly Conference Abstracts*, pages EGU21–7448.

[Riahi and Gerstoft, 2017] Riahi, N. and Gerstoft, P. (2017). Using graph clustering to locate sources within a dense sensor array. *Signal Process.*, 132(C):110–120.

[Roscher et al., 2020] Roscher, R., Bohn, B., Duarte, M. F., and Garcke, J. (2020). Explainable machine learning for scientific insights and discoveries. *IEEE Access*, 8:42200–42216.

[Schimmel and Paulssen, 1997] Schimmel, M. and Paulssen, H. (1997). Noise reduction and detection of weak, coherent signals through phase-weighted stacks. *Geophysical Journal International*, 130(2):497–505.

[Schweitzer et al., 2012] Schweitzer, J., Fyen, J., Mykkeltveit, S., Gibbons, S. J., Pirli, M., Kühn, D., and Kværna, T. (2012). Seismic arrays. In *New manual of seismological observatory practice 2 (NMSOP-2)*, chapter 9, pages 1–80. Deutsches GeoForschungsZentrum GFZ.

[Seydoux et al., 2016] Seydoux, L., Shapiro, N. M., de Rosny, J., Brenguier, F., and Landès, M. (2016). Detecting seismic activity with a covariance matrix analysis of data recorded on seismic arrays. *Geophysical Journal International*, 204(3):1430–1442.

[Shi et al., 2021] Shi, P., Seydoux, L., and Poli, P. (2021). Unsupervised learning of seismic wavefield features: clustering continuous array seismic data during the 2009 l'aquila earthquake. *Journal of Geophysical Research: Solid Earth*, 126(1):e2020JB020506.

[Stork et al., 2020] Stork, A. L., Baird, A. F., Horne, S. A., Naldrett, G., Lapins, S., Kendall, J.-M., Wookey, J., Verdon, J. P., Clarke, A., and Williams, A. (2020). Application of machine learning to microseismic event detection in distributed acoustic sensing data. *Geophysics*, 85(5):KS149–KS160.

[Trnkoczy, 2009] Trnkoczy, A. (2009). Understanding and parameter setting of sta/lta trigger algorithm. In *New Manual of Seismological Observatory Practice (NMSOP)*, pages 1–20. Deutsches GeoForschungsZentrum GFZ.

[Tsutaki et al., 2011] Tsutaki, S., Nishimura, D., Yoshizawa, T., and Sugiyama, S. (2011). Changes in glacier dynamics under the influence of proglacial lake formation in rhonegletscher, switzerland. *Annals of Glaciology*, 52(58):31–36.

[Valentine and Trampert, 2012] Valentine, A. P. and Trampert, J. (2012). Data space reduction, quality assessment and searching of seismograms: autoencoder networks for waveform data. *Geophysical Journal International*, 189(2):1183–1202.

[Virtanen et al., 2020] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald,

A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

[Walter et al., 2020] Walter, F., Gräff, D., Lindner, F., Paitz, P., Köpfli, M., Chmiel, M., and Fichtner, A. (2020). Distributed acoustic sensing of microseismic sources and wave propagation in glaciated terrain. *Nature Communications*, 11(1):2436.

[Webster et al., 2013] Webster, P., Wall, J., Perkins, C., and Molenaar, M. (2013). *Microseismic detection using distributed acoustic sensing*, pages 2459–2463. Society of Exploration Geophysicists.

[Xu and Tian, 2015] Xu, D. and Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2(2):165–193.

[Xu and Wunsch, 2005] Xu, R. and Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678.

[Yilmaz, 2001] Yilmaz, Ö. (2001). *Seismic data analysis: Processing, inversion, and interpretation of seismic data*. Society of exploration geophysicists.

[Yoon et al., 2015] Yoon, C. E., O'Reilly, O., Bergen, K. J., and Beroza, G. C. (2015). Earthquake detection through computationally efficient similarity search. *Science Advances*, 1(11).

[Zhan, 2019] Zhan, Z. (2019). Distributed Acoustic Sensing Turns Fiber-Optic Cables into Sensitive Seismic Antennas. *Seismological Research Letters*, 91(1):1–15.

[Zhu and Stensrud, 2019] Zhu, T. and Stensrud, D. (2019). Characterizing thunder-induced ground motions using fiber-optic distributed acoustic sensing array. *Journal of Geophysical Research: Atmospheres*, 124.

Appendix A

Supplementary Figures

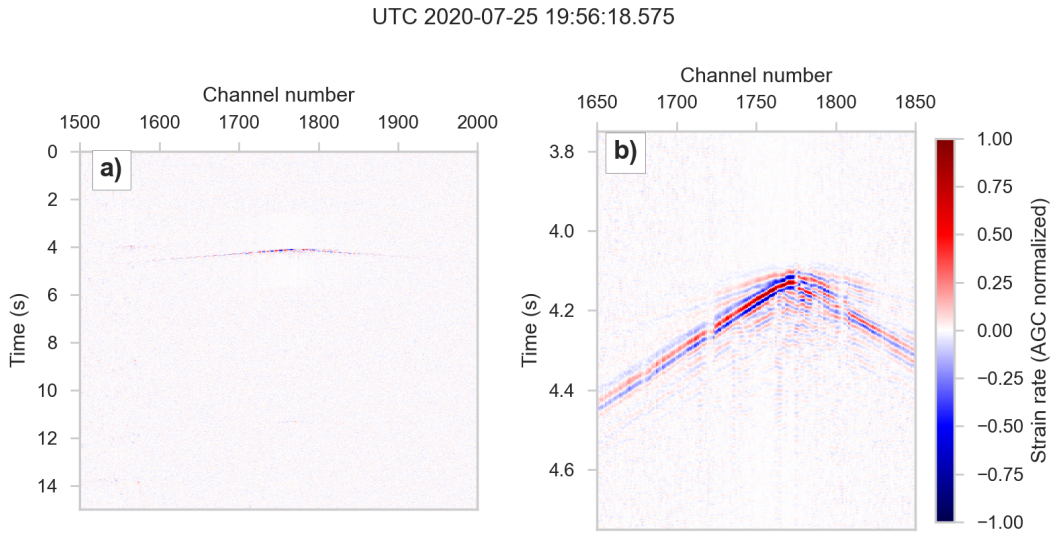


Figure A-1: Icequake for which all feature representations in chapter 4 were computed. For the feature extraction, a window ranging from channel 1650 to 1850 was used, with 15 s length. a) Whole time extent. b) Zoom onto the event onset.

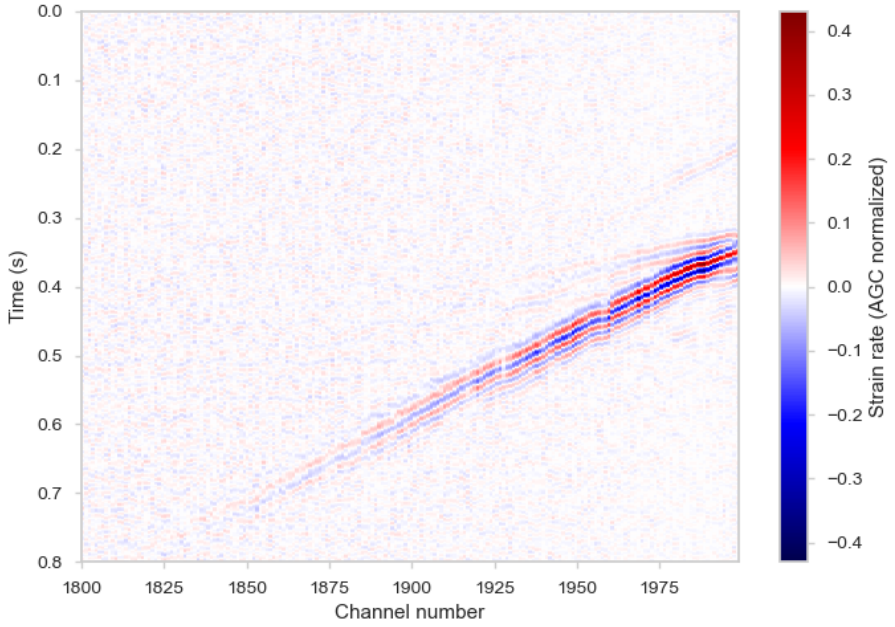


Figure A-2: Cryoseismic event belonging to cluster 1. UTC 2020-07-15T03:59:20.719

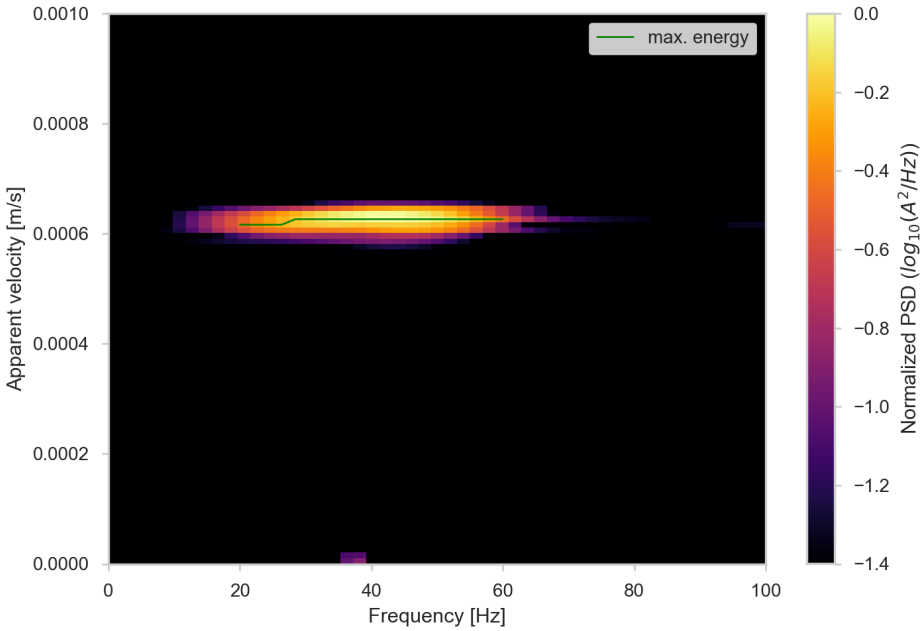


Figure A-3: Representation of the event in Fig. A-2 in the $f-p$ domain. The wave seems to be non-dispersive, at least in the frequency range between 10 to 60 Hz.

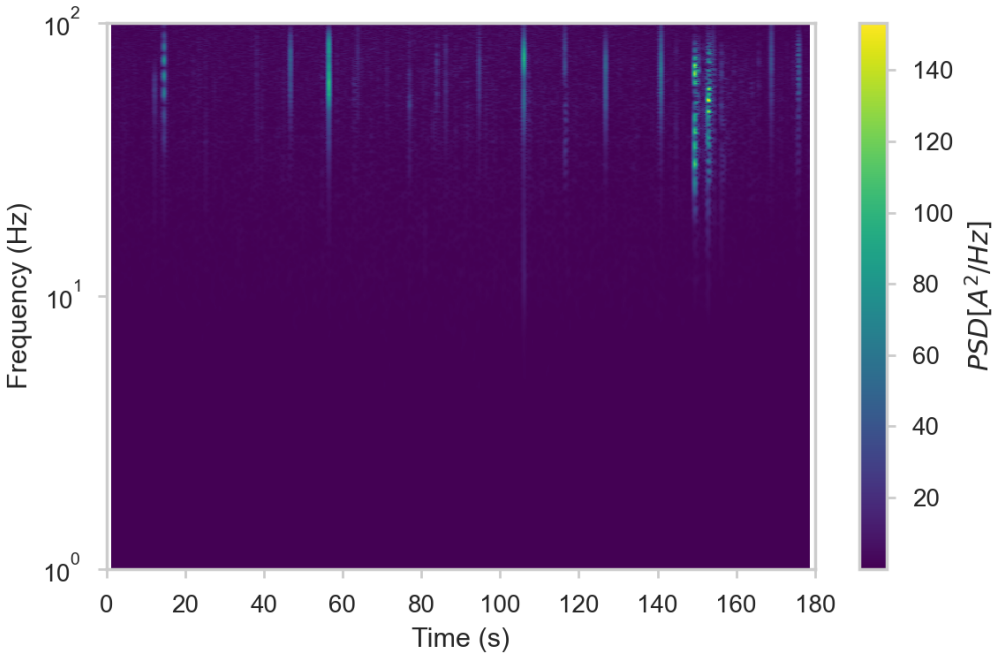


Figure A-4: 180 s long spectrogram for channel 1990. UTC 2020-07-15T03:58:50.719