Project Plan: Unsupervised Machine Learning - An Application to Seismic Amplitude vs Offset Interpretation

Author: Hugo Benjamin Coussens
Supervisors : Lukas Mosser and Prof Olivier Dubrule

June 28, 2019

1 Rationale

Recent advances in unsupervised machine learning techniques have shown to be very powerful in identifying inherent similarities and features within high dimensional datasets. Initial studies into the application of these algorithms on seismic data have provided promising results. The algorithms have exhibited an ability to learn and represent meaningful geophysical features in the low dimensional space. Unlike conventional techniques the models learn parameters from the full seismic trace, revealing possibly previously un-exposed insights.

The study and utilisation of dimensionality reduction in this domain is still in its infancy. Investigations, software tools and use cases are limited. To further understand the utility and transferability it is vital to have a highly accessible and optimised tool to deliver the analysis.

2 Project Aim

The project aim is to develop a software tool which utilises state of the art unsupervised machine learning techniques to analyse seismic data. The tool will combine pre-processing techniques, unsupervised learning and AVO analysis to deliver previously un-realised insights in seismic data-sets. The primary goal of this analysis is to obtain new information regarding pore-fluid identification, presence and location to the geologist/geophysicists toolkit.

In order to realise a successful tool, it must be useful, usable and sustainable. To be useful in this context means to deliver useful insights drawn from seismic data. The project will involve an element of investigation and optimisation of the different processing and dimensionally reduction methods used. To be usable the tool needs to be able to deliver these insights to a user without prior expertise in computer programming. This means the tool must be able to deliver all identified requirements and features in this context without the need to modify source code. The software will need to come with clear and simple instruction for how a user will install any dependencies and run the tool. Finally to be sustainable the software will be developed with consideration for future additions, modification or generalisation by myself or another. This takes into account the design and working practice of the whole project, from software structure and documentation to recommendations for further work.

3 Relevant Work

3.1 Unsupervised Learning Techniques

Unsupervised learning algorithms and techniques are the subset of machine learning tasked to deal with unlabelled datasets. In absence of the 'correct solutions' or target variables, the task is to draw insight from the the intrinsic and hidden structure in the data. The outcomes from these algorithms are either clustering, data compression, or translation of meaningful features into latent variables.

Data in the real world often possesses high dimensionality, for example; there are many pixels to an image; many discrete values to a signal recording and many words to a magazine article. In order to draw more meaningful representations of this data it is necessary to reduce its dimensionality. Doing so will facilitate visualisation, clustering and compression of the high-dimensional data. Initially dimensionality reduction was confined to linear techniques such as Principal Component Analysis (PCA) or Linear Discriminant Analysis. The obvious downfall of these linear techniques is the inability to project on non-linear manifolds. There have been great advances in the performance of non-linear techniques recently. The Swiss roll data-set is an excellent illustrative example for benefit of nonlinear techniques. The points lie on a two-dimension manifold representing a spiral exuded in three-dimensional space. Linear methods like PCA are unable to represent the full structure of the manifold and effectively squash the layers of the Swiss roll together. In contrast non-linear methods have been shown to find an embedding that maintains structure, in effect un-rolling the Swiss roll into a two-dimensional representation.

3.1.1 Variational Auto-Encoders (VAE)

To understand VAE's, Kingma and Welling (2013), it is helpful to first understand the structure and usage of autoencoders. They are comprised of artificial neural networks (ANN's) which work in two parts: The encoder network reduces inputs into meaningful representations at a low dimensional 'bottleneck', a decoder network aims to reproduce the input back out from the bottleneck representations. The goal is for the network to learn patterns and structures of input data that may appear more clearly in the latent space. The network is usually trained on a simple cross-entropy loss function which penalises the network for inaccurate reconstruction from input to output.

VAE's follow a very similar structure to autoencoders, but with very different capabilities. VAE's act as generative models, whereby learnt distributions of the data can be sampled in order to generate brand new synthetic outputs. A key difference of the VAE is the latent space created by the encoder network is stochastic, is continuous and follows a normal distribution. The objective function of VAE's is known as the evidence lower-bound (ELBO). Comprised of two constituent terms, the first: reconstruction likelihood aims increase accuracy of the 'round trip' between input and latent space. The second: KL-divergence aims to spread the range of latent vectors to match a Gaussian prior.

A modified version of VAE known as β-VAE aims to learn a latent representation in which the parameters of the latent vector exhibit meaningful 'disentangled' features from the input Burgess et al. (2018a)

3.1.2 Manifold Embedding/Neighbour Mapping

An alternative approach to dimensionality reduction comes in the form of neighbour maps or manifold embedding algorithms. These algorithms are particularly useful in the visualisation of high-dimensional data-sets but not always appropriate for low dimensional feature representations. These are often stochastic non-linear processes. Therefore, in contrast to the deterministic techniques such as PCA, these algorithms will return slightly different results with every run.

Significant progress has been made recently in this field with t-SNE by van der Maaten and Geoffrey E. (2008) and UMAP by McInnes et al. (2018).

The t-SNE algorithm finds a low dimensional embedding that minimises the kl-divergance between two distributions, the distribution that measures pairwise similarity in the input data and the distribution which does the same in the low-dimensional embedding.

UMAP brings a similar approach to t-SNE but with a few extra advantages. Both algorithms are efficient in successfully preserving local structure relationships in data. However, UMAP has been shown to better preserve the overall global structure of the data set. UMAP is able to transform data into a representation of any chosen dimension, where t-SNE is computationally limited to 2 or 3. UMAP has also been shown to operate with a superior run-time performance to t-SNE and other similar algorithms.

3.2 Dimensionality Reduction Applied to Seismic Data

The investigations provided by Mosser et al. (2019) are the basis for this project. Unsupervised learning techniques were applied to a seismic data set for visualisation in 2-D. Calibration with AVO analysis revealed the clustering of traces with low fluid-factor values. The dataset used is collected from a well documented oil field located in the Norwegian North Sea - the Glitne field. The benefit of this dataset is the opportunity to correlate findings in the analysis to direct well observations.

The investigation approaches seismic dimensionality reduction in the following workflow:

1. Data Processing:
   - Horizon flattening
   - Data normalisation
2. Calculation of fluid factor for every amplitude sample using standard industry approach
3. Dimensionality reduction applied using UMAP to 2-D
4. Training and running VAE, reducing to latent space of 8-dimensions, represented in 2d by UMAP algorithm
5. Both are visualised with AVO fluid factors overlain as attributes.

Desai (2019) provides an investigation into the robustness and sensitivity of the VAE model used. Geophysically motivated variations were applied to seismic traces and the response made in the model latent space was observed. These varations inlcuded: amplitude, times shifting, Gaussian noise and far offset amplitude increase. Results show the responses of latent variables correlate well with geophysically motivated modifications.

The possible uses/interpretations that can be drawn from the studies of unsupervised learning applied to seismic are as follows:

- VAE's can be used as a tool for the de-noising of seismic data
- VAE's can be used as generative models to create synthetic seismic data
- AVO calibrated manual clustering leads to meaningful spatial plots in map view
- Use of Latent vectors also revleal as meaningful map view plotting attribute
- Dimensionality reduction can be used as an anomaly detection tool for seismic traces

4 Project Execution

In view of creating an overall high quality software tool, the project can be split into two discrete objectives. The primary focus is the development of a complete software tool. The secondary objective is to advance the capabilities and performance of methodolgies encapsulated within the tool. This will involve exploring new ML algorithms, investigations into pre-processing techniques and optimisation of ML model parameter space.

4.1 Development Methodology

4.1.1 Key Requirements

The workflow introduced by Mosser et al. (2019) and Mosser (2018) has provided a successful proof of concept for the scientific utility of this tool. An initial identification of the key software functionalities, predicted challenges and investigations has been compiled in figure 1.

4.1.2 Development Cycles

Two main phases of development are envisioned for the project. After initial software architectural design a first prototype program which satisfies all the key requirements will be developed. Following this will be a period of testing and identification of improvements relating to the software functionality/usability itself, this is distinct from separate investigations on model performance as discussed in 4.2. Examples of the testing includes: profiling and runtime performance analysis; storage optimisation for extremely large data sets; visualisation/plotting improvement and robustness testing on new data and invalid user-inputs. Following this will be the final phase of development to implement the identified improvements and additions. Unit tests, continuous integration, accompanying documentation and a framework for additional features will be developed concurrently with the main software.

4.2 Investigations and Optimisations

The performance of machine learning algorithms rely on two things: the input data and the model hyper-parameters. The project aims to tackle investigations into the optimisation of both.

4.2.1 Performance Benchmark

There is no single performance metric that can capture analysis performance in its entirety. Due to the unsupervised nature of the problem we are not aiming for label prediction but a more subjective interpretation of accurate AVO feature extraction.

The ELBO is an objective parameter that is maximised by the VAE. Increased reconstruction ability through the VAE acts to increase the ELBO. This is a valuable metric as it reveals the quality of the coding to reconstruct an input, however this does not necessarily relate to AVO feature extraction. Therefore, visual interpretation and assessment of the clustering will be used alongside ELBO value as a benchmark of performance.

4.2.2 Pre-processing

Pre-processing techniques or feature engineering can provide great improvements in machine learning algorithms if used effectively. These will involve some meaningful modification to the input data which provides the algorithm with extra information or conditions the information in a favourable way. The investigation will explore the parameter space of existing methods such as horizon flattening and normalisation. It also aims to experiment with new techniques such as creation of new variables based off near and far trace differences.

4.2.3 Model Hyperparameters

All machine learning models have an array of specific hyper-parameters which need tuning for the task specified. An investigation into the optimal hyper-parameters will be undertaken with the scope to experiment with slightly deeper VAE architecture as recommended in Desai (2019). The increased complexity to the model may enable more efficient feature extraction, especially if combined with higher-dimensional input provided by the feature engineering.

4.2.4 Additonal Models/Algorithms

The project aims to implement two new algorithms, previously un-tested on seismic data. These are the β-VAE from Burgess et al. (2018b) and a standard auto-encoder. These algorithms both have certain nuances which may bring valuable insight.The β-VAE is documented to have a disentangling effect of meaningful features in the latent space. Auto-encoders allow for an unconstrained latent space representation, albeit at the expense of continuous latent space variables.

4.3 Current Progress

The first prototype tool has been developed and is documented in 'Prototype_Notebook_1.0.ipynb' located at: https://github.com/msc-acse/acse-9-independent-research-project-coush001. Note: the tool will not be functional without proper installations and setup, however the supporting software and widget GUI can be viewed for reference.

4.3.1 Architecture Design and Back-End

An initial software architecture has been designed and developed which delivers the key software requirements and functionalities, figure 2. The 'back-end' of the software has been developed into a set of API tools, upon which an interface will interact to deliver the analysis. This initial software API is able to run analysis with the flexibility to explore the whole pre-processing and ML model parameter space with only a few lines of code.

4.3.2 Prototype Interface

The prototype interface has been developed with the 'ipywidgets' package built for jupyter notebooks. The package facilitates widget and logic definitions to be developed directly in the jupyter notebook. The graphical user interface interacts with the back-end and provides a user with all the functionality of the tool without need for editing or running of any code.

4.4 Project Management

For a high-level control of project time-management the gantt chart in figure 3 is being utilised. This will ensure time is spread appropriately across the key focus areas and provides a means to track progress. The chart contains expected key milestone completion dates and component sub-task time allocation. Regular project progress reviews will be undertaken to assess the development of the project. If needed, the gantt chart will be updated to reflect any redistribution of time to the key project focuses.

References

Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., and Lerchner, A. (2018a). Understanding disentangling in beta-VAE. (Nips).

Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., and Lerchner, A. (2018b). Understanding disentangling in β-VAE. *arXiv e-prints*, page Burgess2018.

Desai, S. (2019). Application of Deep Learning Techniques for Amplitude Variation with Offset Analysis. *Imperial College MSci Project.*

Kingma, D. P. and Welling, M. (2013). Auto-Encoding Variational Bayes. *arXiv e-prints*, page arXiv:1312.6114.

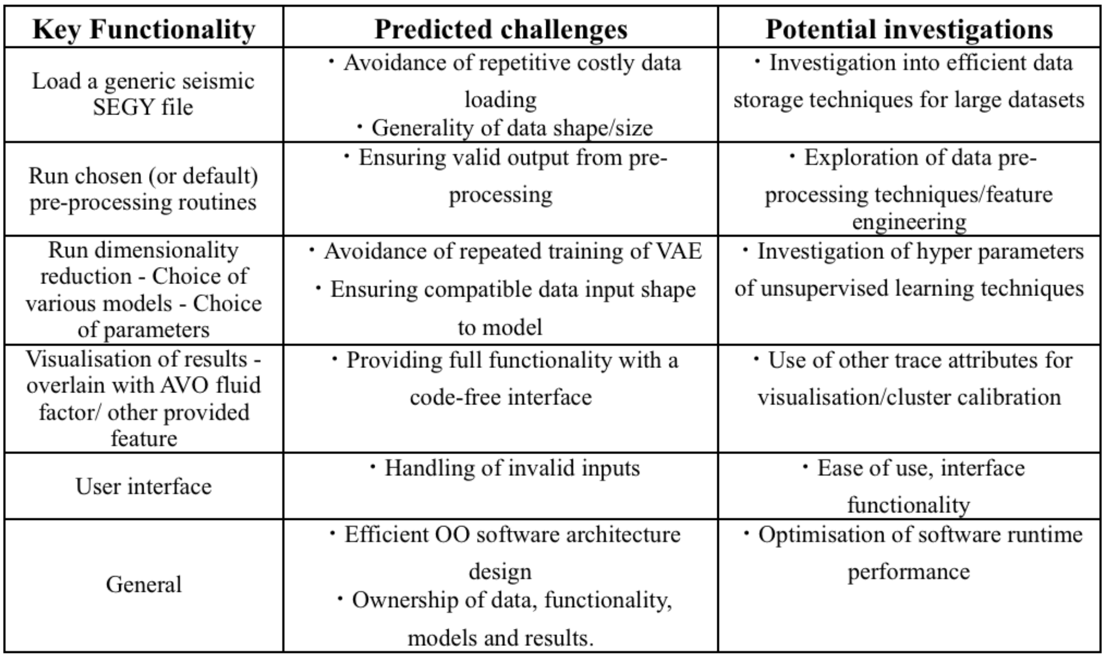
McInnes, L., Healy, J., and Melville, J. (2018). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv e-prints*, page arXiv:1802.03426.

Mosser, L. (2018). https://github.com/LukasMosser/ASAP.

Mosser, L., Monte, A. A., Avseth, P., Draege, A., and Macgregor, L. (2019). Investigating the AVO Signature of a North Sea Oil-Field Using Pseudo-Wells and Unsupervised Deep Learning. In *81st EAGE Conference and Exhibition 2019 - AI/Digitalization for Interpretation - AVO Application.*

van der Maaten, L. and Geoffrey E., H. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 164(2210):10.

Appendix



Key Functionality	Predicted challenges	Potential investigations
Load a generic seismic SEGY file	• Avoidance of repetitive costly data loading • Generality of data shape/size	• Investigation into efficient data storage techniques for large datasets
Run chosen (or default) pre-processing routines	• Ensuring valid output from pre-processing	• Exploration of data pre-processing techniques/feature engineering
Run dimensionality reduction - Choice of various models - Choice of parameters	• Avoidance of repeated training of VAE • Ensuring compatible data input shape to model	• Investigation of hyper parameters of unsupervised learning techniques
Visualisation of results - overlain with AVO fluid factor/ other provided feature	• Providing full functionality with a code-free interface	• Use of other trace attributes for visualisation/cluster calibration
User interface	• Handling of invalid inputs	• Ease of use, interface functionality
General	• Efficient OO software architecture design • Ownership of data, functionality, models and results.	• Optimisation of software runtime performance

Figure 1: Table containing information on the key fuctionality and associated challenges and investigations to be faced.

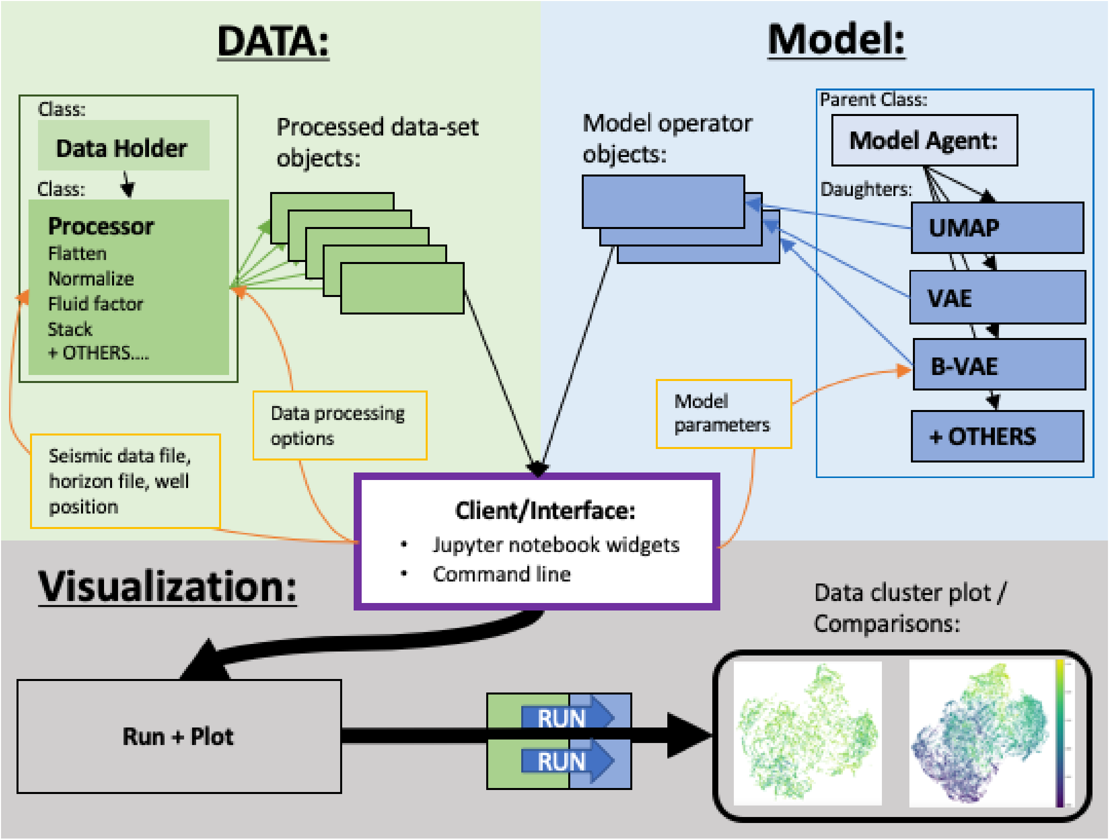


Figure 2: Initial software architecture design

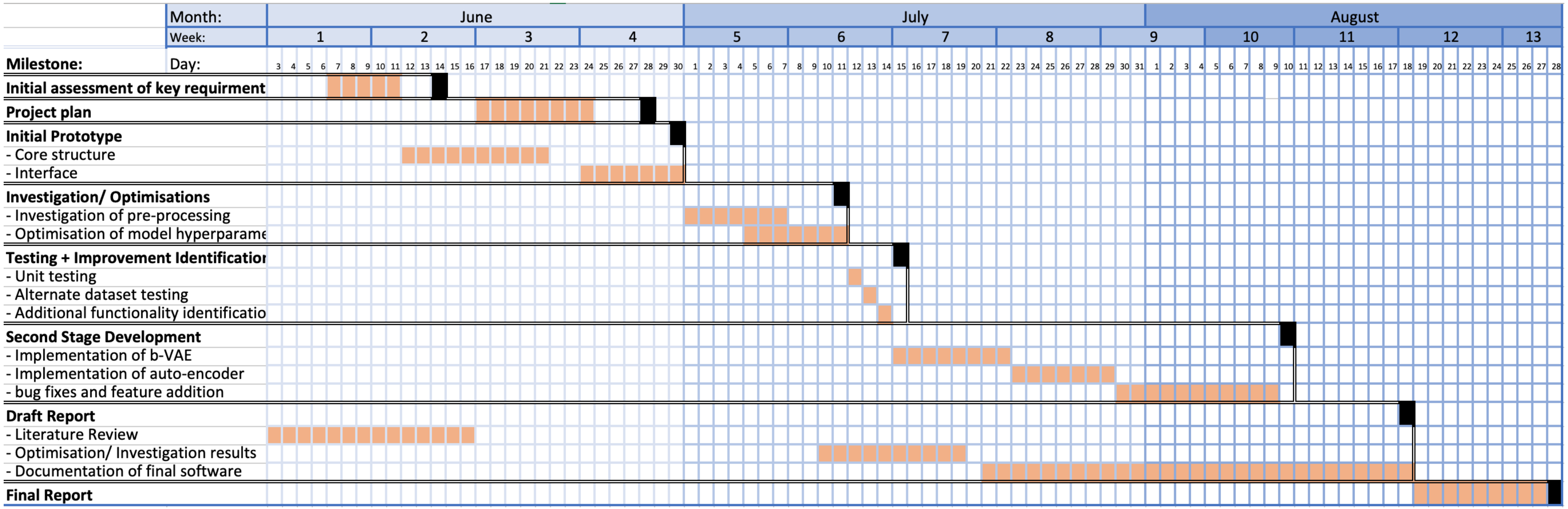


Figure 3: Here's a large drawing of Felix the Cat that wouldn't fit in a portrait page