

basic Intro to Machine Learning



Outline

(Just overview here; for actual content see Jupyter notebooks)

- ▶ a very loose introduction to Machine Learning (ML)
 - as a problem in regression / optimisation
 - supervised vs. unsupervised
 - sample usage in oceanography
- ▶ example with argo data
 - what is Argo?
 - unsupervised ML example: clustering analysis
 - supervised ML example: neural networks

Some propaganda to start with

ML algorithms are:

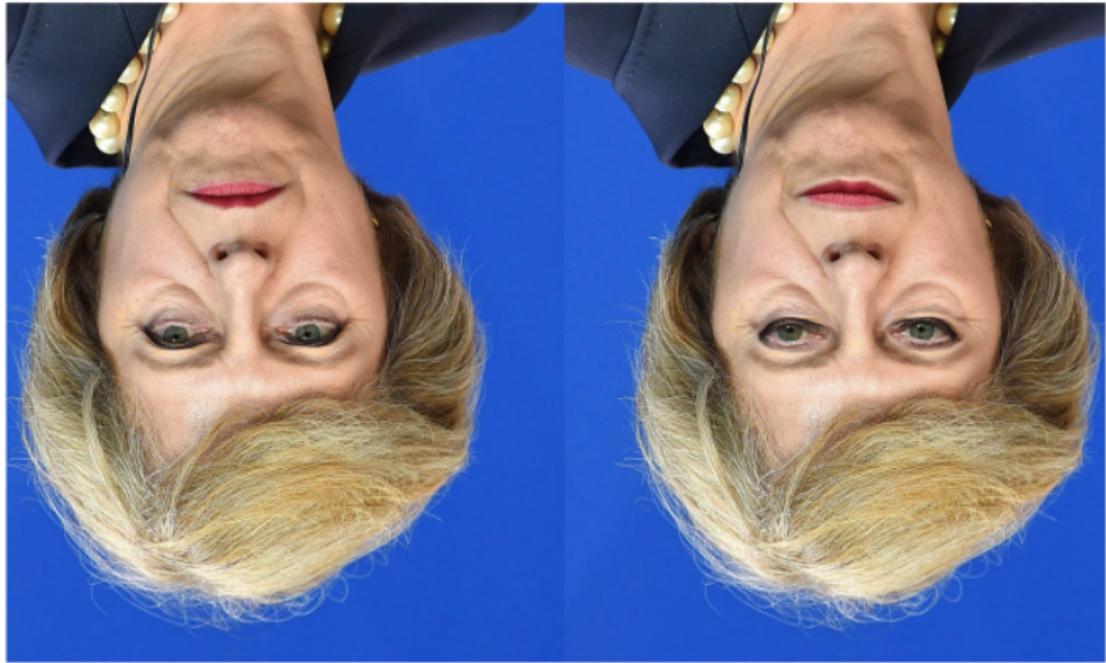
- ▶ algorithms + tools, and that's it
 - very powerful, but context dependent
- ▶ usually **black box**
 - it can work wonderfully / fail spectacularly, but you don't necessarily know why...

!!! prudent to do sanity checks!



Figure: Hermeowus Mora, disciple of Hermaeus Mora the Daedric prince of knowledge and memory

Cursed example: image recognition/generation



Cursed example: image recognition/generation



Machine learning + regression/optimisation

Recall that, in regression, for X the input, Y the output, f the model, where

$$Y = f(X),$$

the aim is to seek f such that we minimise something like

$$J = \sum_i (Y_i - f(X_i))^2$$

→ e.g. linear regression, polynomial fitting

- ▶ ML in a nutshell follows the same principle
 - algorithms are different (e.g. nonlinear, network based, different optimiser, stochastic/probabilistic)

Training, Validation, Testing data

Normally split (X, y) into

- ▶ **training data** ($X_{\text{train}}, Y_{\text{train}}$) (most data should be here)
 - exposed to ML algorithms for training the model
 - used to compute misfits or **loss function**
 - ▶ **validation data** ($X_{\text{val}}, Y_{\text{val}}$)
 - exposed to ML algorithms to tune model **hyperparameters** and/or model selection
 - ▶ **test data** ($X_{\text{test}}, Y_{\text{test}}$)
 - **NOT** exposed to ML algorithm
 - used to test performance of model
- !!! sometimes “validation” and “test” are swapped

Unsupervised vs. supervised

- ▶ unsupervised ML is where data is **unlabelled**, and algorithm picks out features by themselves
→ e.g. PCA (so EOFs), clustering, some examples of neural networks



Figure: Cursed cats/dogs (?) from PCA. Figure adapted from Fig. 10 of Brunton, Brunton, Proctor & Kutz (2013).

Unsupervised vs. supervised

- ▶ supervised ML is where data is labelled, and algorithm fits model between input and output
 - often want this for prediction purposes
 - e.g. (multi-)linear regression, some examples of neural networks
- ▶ other characterisations (e.g. semi-supervised, reinforcement)

Unsupervised vs. supervised

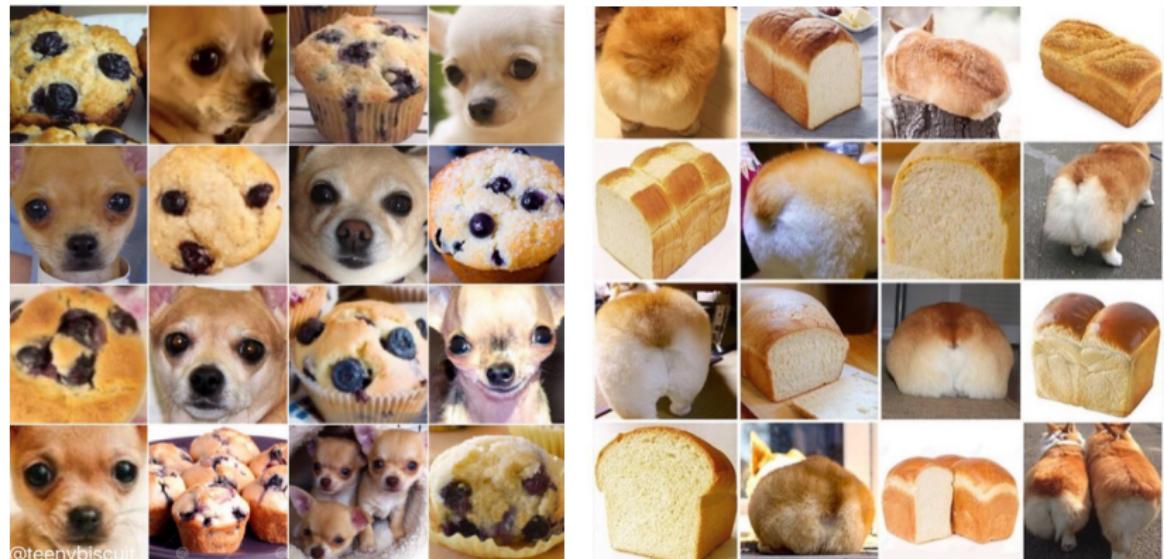


Figure: Various entries from the “animal or things” meme, as found on the internet.

Unsupervised vs. supervised

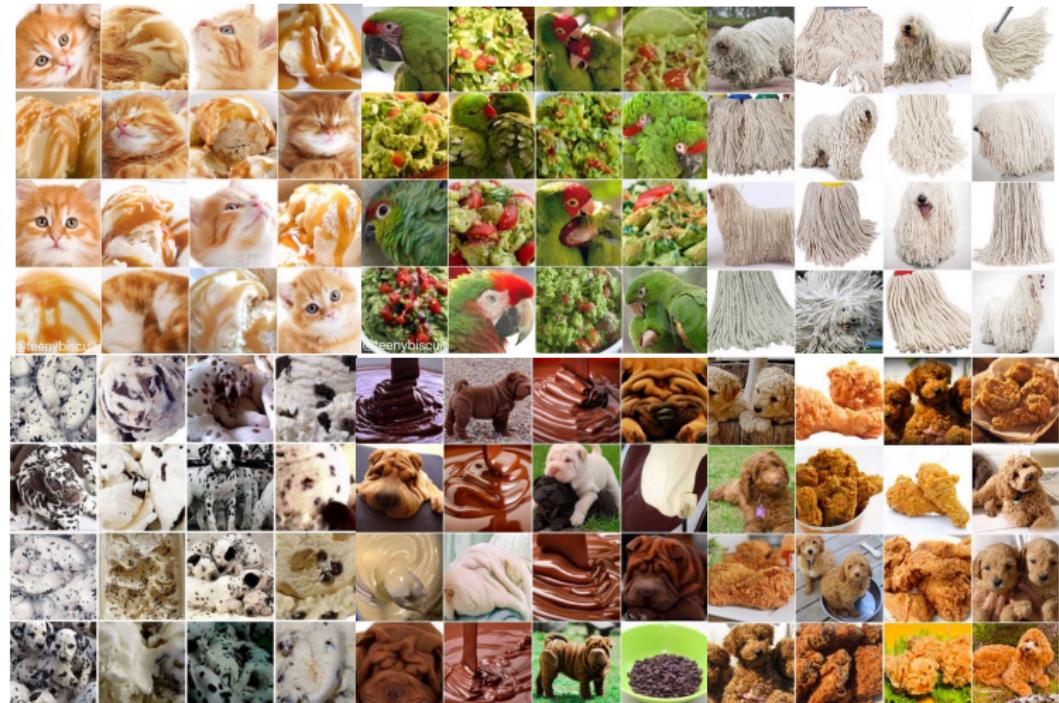


Figure: Various entries from the “animal or things” meme, as found on the internet.

Oceanographic examples

Example: Argo

- ▶ A CTD gets
 - conductivity to get S
 - temperature for T
 - it really measures p to get depth
 - can put other sensors on (e.g. pH, oxygen, etc.)

- ▶ argo system consists of CTDs that floats around the ocean



Figure: An Argo float being thrown off a ship. Image from NOAA.

Example: Argo

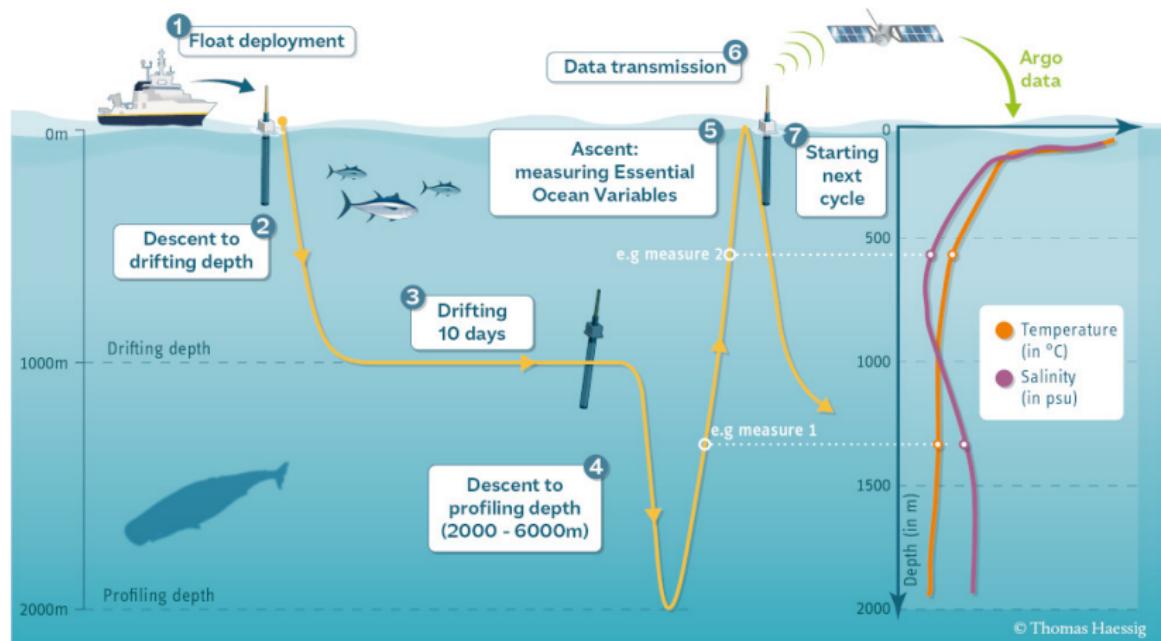


Figure: Argo float cycle schematic. From argo.ucsd.edu

Example: Argo

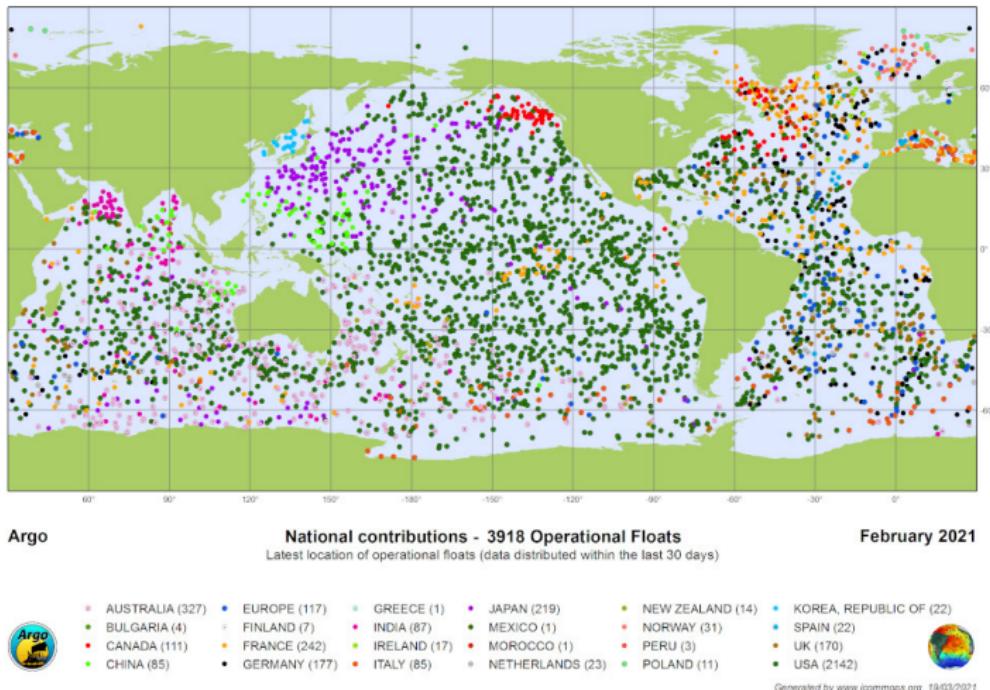


Figure: Argo locations as of Feb 2021. Note the dots are enhanced in size, so coverage is not as dense as it seems.
From argo.ucsd.edu

Example: Argo

» Dimensions:	(DEPTH: 302, N_PROF: 128910)		
▼ Coordinates:			
DEPTH	(DEPTH)	float32 0.0 -5.0 -10.0 ... -1500.0 -1505.0	
LATITUDE	(N_PROF)	float32 dask.array<chunksize=(67010,), meta=n...>	
LONGITUDE	(N_PROF)	float32 dask.array<chunksize=(67010,), meta=n...>	
TIME	(N_PROF)	datetime64[ns] dask.array<chunksize=(64455,), meta=n...>	
▼ Data variables:			
BRV2	(N_PROF, DEPTH)	float32 dask.array<chunksize=(67010, 302), met...>	
DBINDEX	(N_PROF)	float64 dask.array<chunksize=(67010,), meta=n...>	
PSAL	(N_PROF, DEPTH)	float32 dask.array<chunksize=(67010, 302), met...>	
SIG0	(N_PROF, DEPTH)	float32 dask.array<chunksize=(67010, 302), met...>	
TEMP	(N_PROF, DEPTH)	float32 dask.array<chunksize=(67010, 302), met...>	
» Attributes: (12)			

Figure: Argo dataset in `zarr` format opened as a `xarray` object.

- ▶ argo data to be downloaded given in `zarr` format
 - need `zarr` package, can open data through `xarray`
 - `ungridded` data here
 - see also `argopy` package

Example: Argo

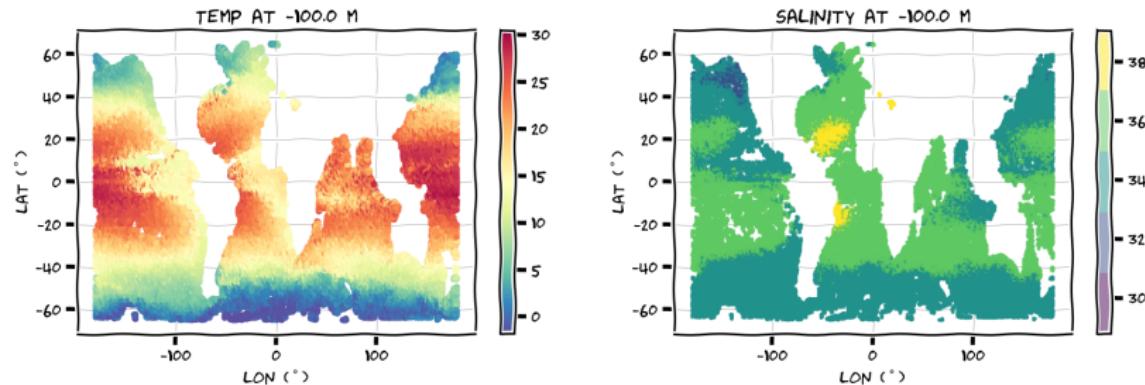


Figure: Argo (in-situ) temperature and (practical) salinity at some fixed depth as a scatter plot coloured by data entry.

- ▶ ungridded data, each point is an entry
→ scatter plot, with dot coloured by data

Clustering

- ▶ we know different watermasses have different properties
 - e.g. NADW is more salty
 - unsupervised learning?

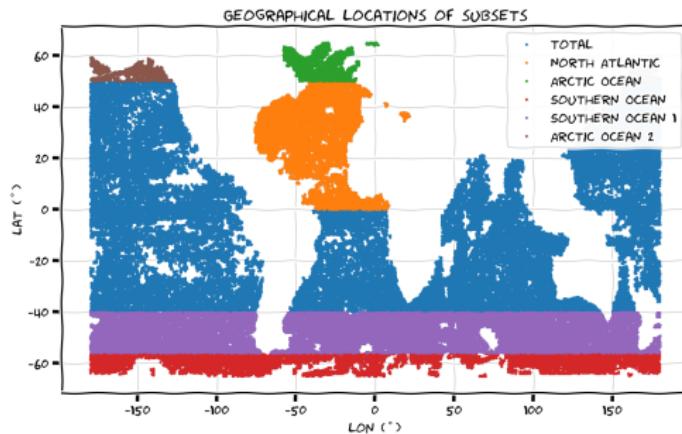


Figure: Artificial clustering.

Clustering

- ▶ we know different watermasses have different properties
 - e.g. NADW is more salty
 - unsupervised learning?

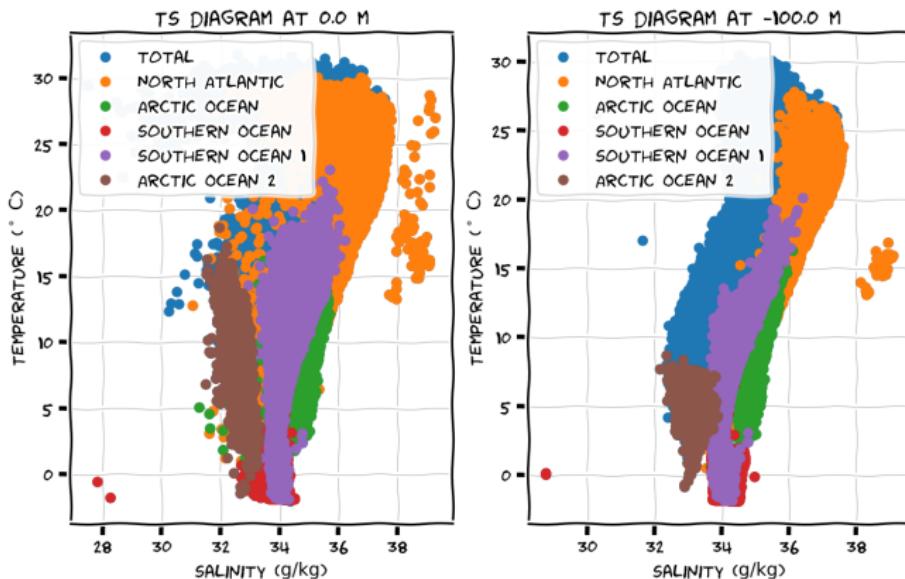


Figure: Artificial clustering as above, but in TS-diagram.

Clustering

- ▶ one example is *k*-means
 - partition data and find means
 - move partitions slightly and compute new means
 - iterate on partitions such that distance to partition means are minimised (finds local minimums)

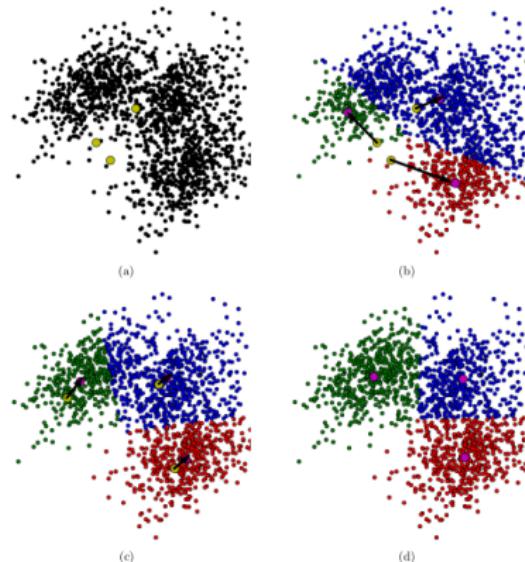
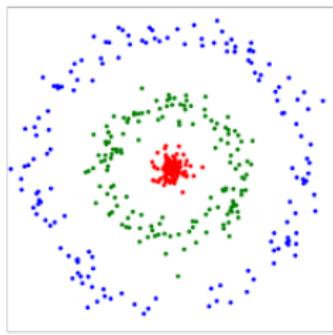


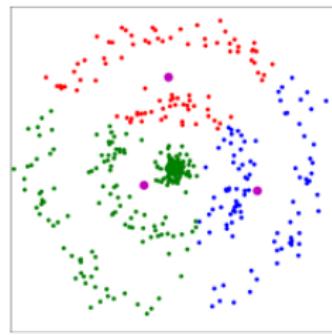
Figure: Demonstration of *k*-means algorithm. Diagram taken from Machine Learning course of Christoph Haase and Varun Kanade at University of Oxford.

Clustering

- ▶ example where k -means can fail



(a) Data with three clusters as concentric circles



(b) Output of k -means algorithm

Figure: Demonstration of failure of k -means algorithm. Diagram taken from Machine Learning course of Christoph Haase and Varun Kanade at University of Oxford.

Clustering

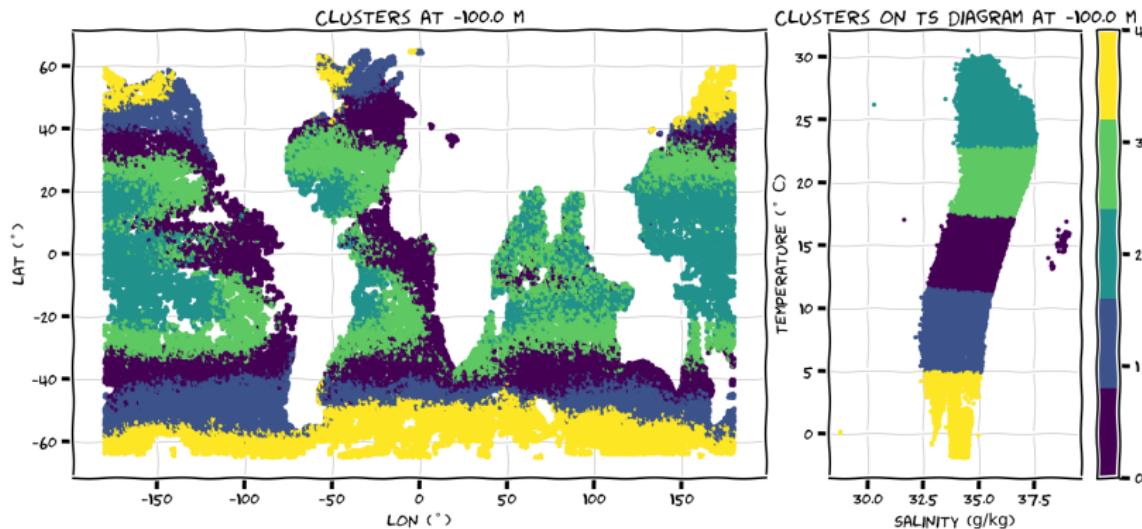


Figure: Clustering from k -means algorithm.

- ▶ k -means in TS -space really
- ▶ some physical rationalisation possible

!!! didn't standardise data here (probably should have)

Clustering

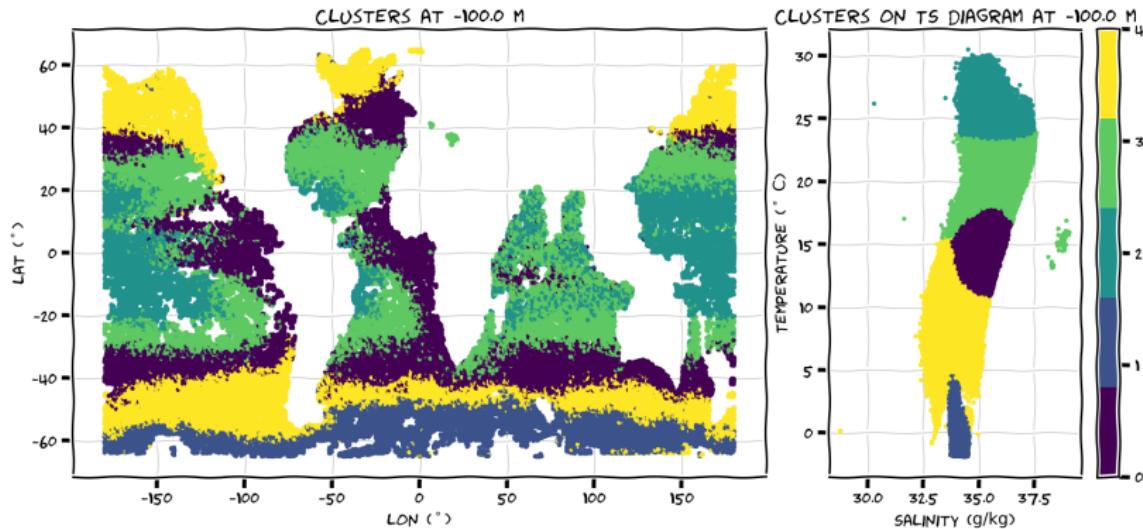


Figure: Clustering from Gaussian Mixture Model.

- ▶ similar to above but with some subtle differences
 - ▶ GMM used in oceanography before (e.g. Jones *et al.*, 2019, in Southern Ocean)
- !!! didn't standardise data here (probably should have)

Neural Network

- ▶ suppose we want to predict salinity from temperature, i.e. prediction/reconstruction
 - more for demonstration really...
 - split data first (`sklearn.train_test_split`)

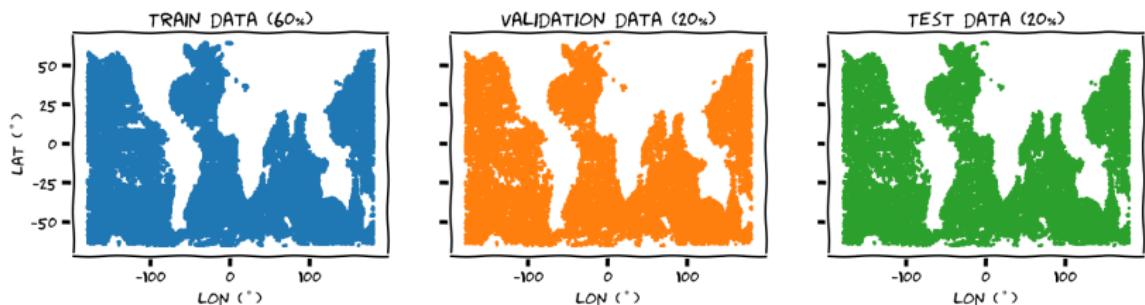


Figure: Splitting of argo data into training:validation:test as 60:20:20.

Neural Network

- ▶ linear regression?
 - standardise data
 - train with training data (could in principle use everything)

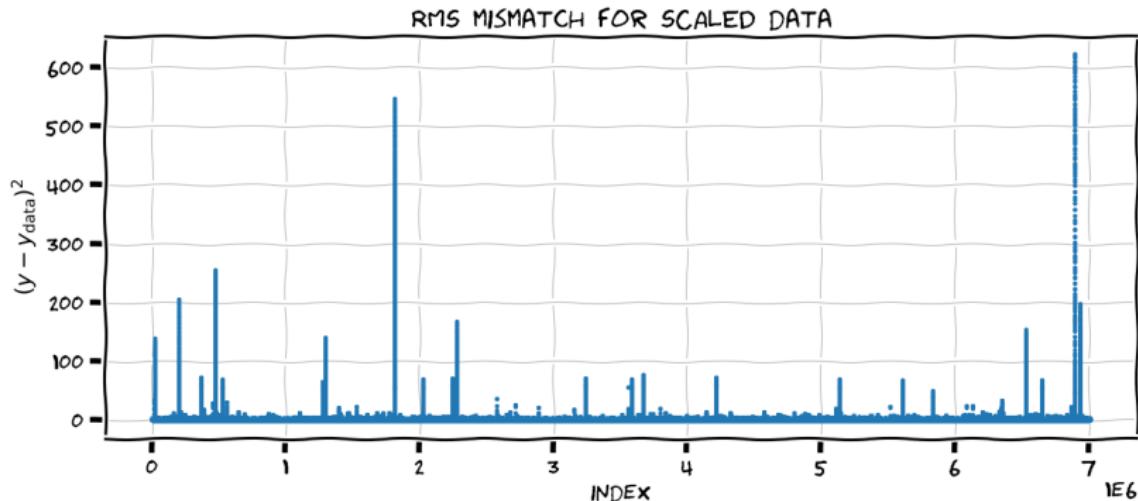
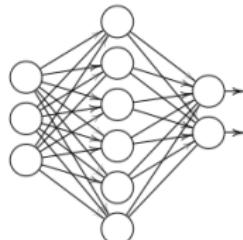
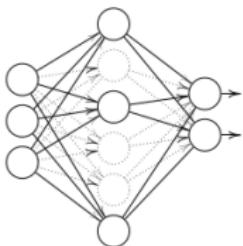


Figure: Root-mean-square loss against index for scaled data, so anything larger than 1 is pretty bad.

Neural Network



(a) Neural Network



(b) Neural Network with Dropout

Figure: Schematic of neural network. Diagram adapted from Machine Learning course of Christoph Haase and Varun Kanade at University of Oxford.

- ▶ a **neural network** has
 - **nodes** containing features or transformation rules
 - **links** linking the nodes
 - **weights** tagged with links specifying weighting or transition probability going from one node to another
- ▶ simple case would be adjusting weights to minimise the mismatch / loss function
 - could in principle adjust features in nodes etc.
 - **drop off** procedure as a stabiliser

Neural Network

- ▶ train model with training and validation data
→ data has been standardised here
- ▶ model trained over 30 epochs
→ think 30 complete passes/iterations

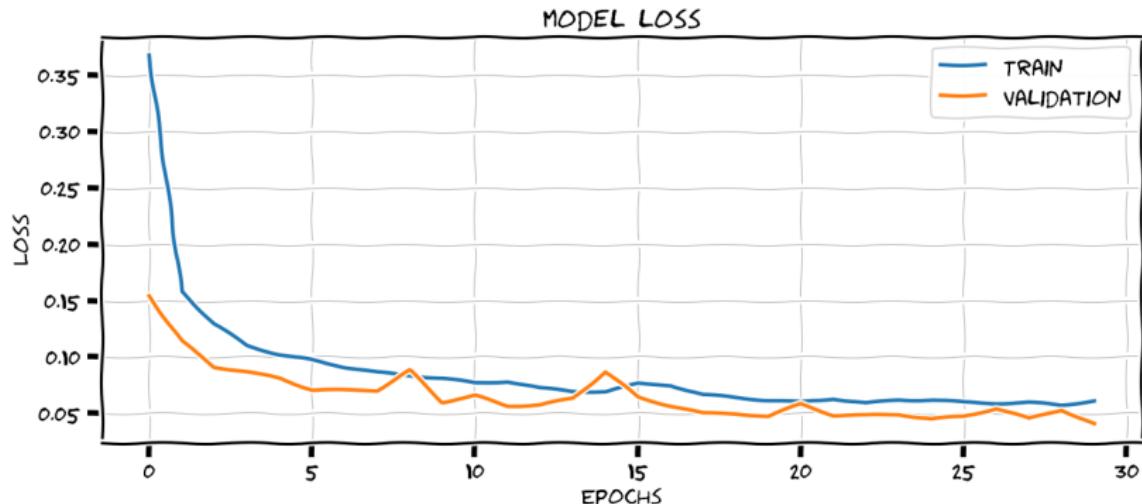


Figure: RMS loss against epoch. Note the RMS loss is not zero (and we don't expect it to be).

Neural Network

- model takes an input depth varying *in-situ* temperature and returns a salinity profile
→ three random realisations below, with scaling inverted

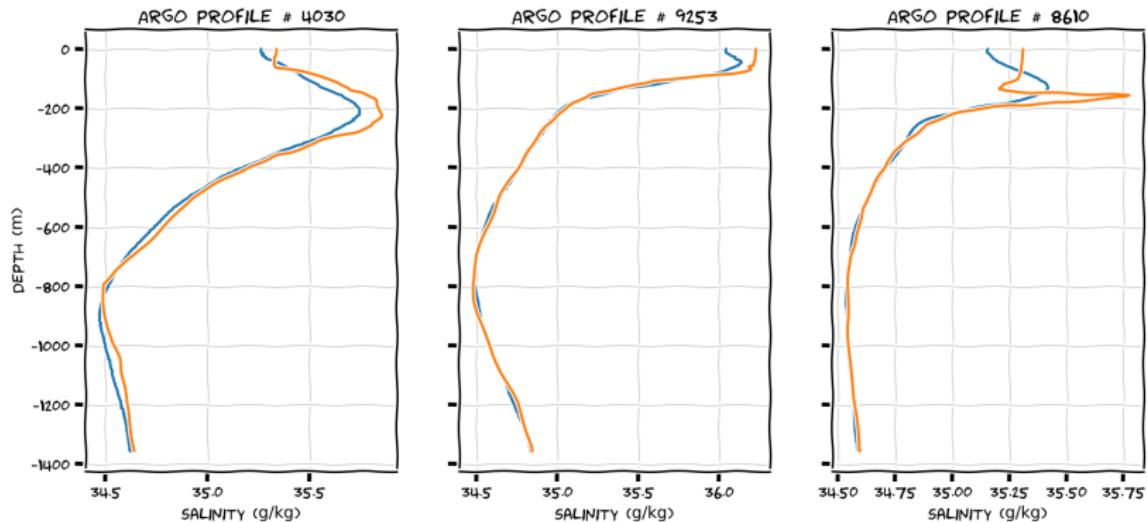


Figure: Three examples of using the trained neural network.

Neural Network

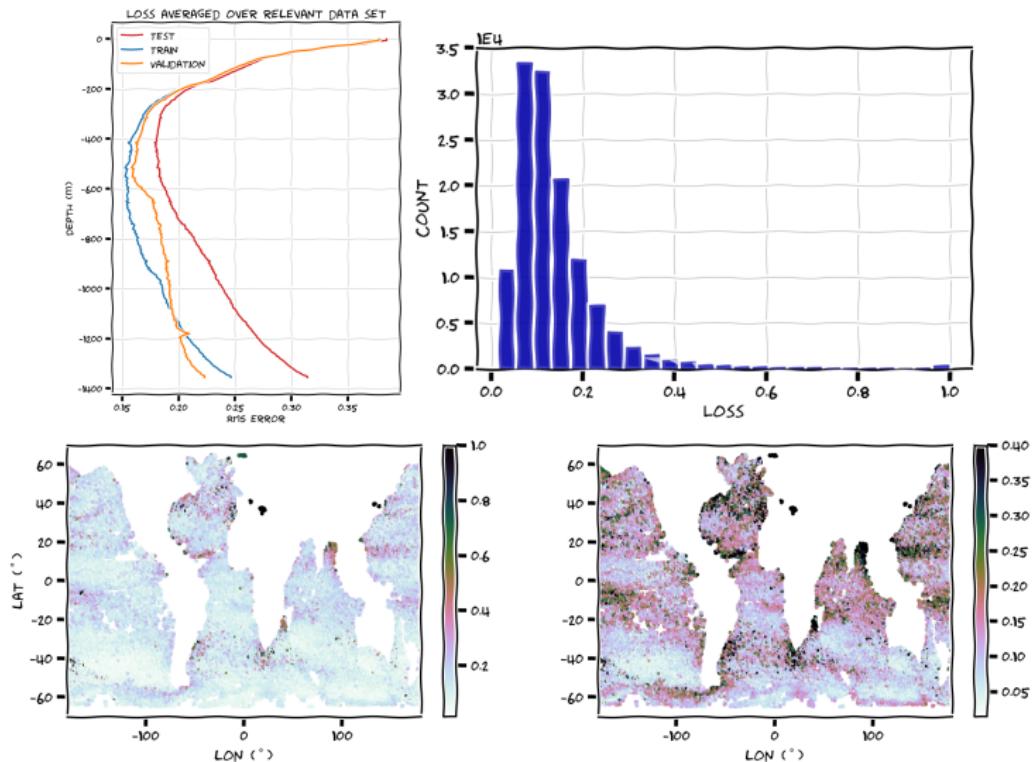


Figure: Some summary plots of the loss.

Neural Network

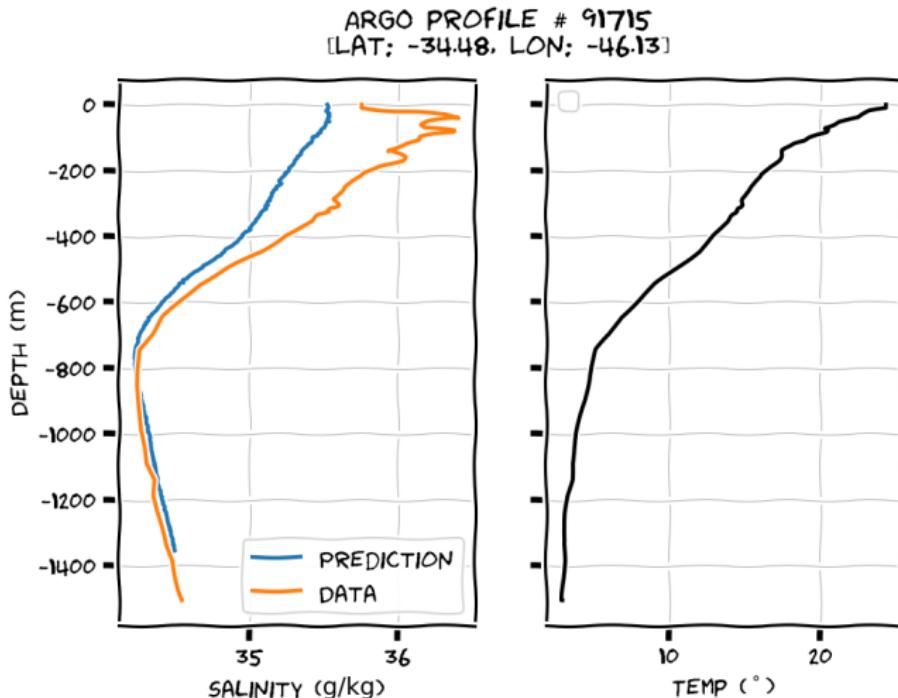


Figure: Example of a case with particularly high error.

Jupyter notebook

bonus Jupyter notebook (with thanks to Fei Er) to get some code practise

- ▶ different ways of reading the argo data
- ▶ different algorithms to try
- ▶ different questions to ask
- ▶ different features to add
→ those based on topology?
- ▶ ...

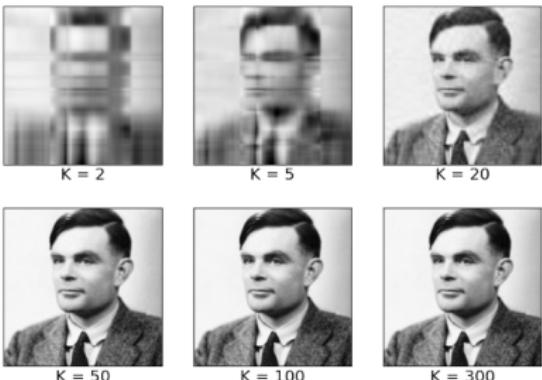


Figure: Image reconstruction: Neural network with data from PCA?