This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N° 773330

Deliverable report for

GAIN

Green Aquaculture Intensification in Europe

Grant Agreement Number 773330

Deliverable D1.8

Title: Information Management System: Final release

Due date of deliverable: 31/08/2021

Actual submission date: 29/10/2021

Lead beneficiary: IBM

Authors: O'Donncha, F., Akhriev A.,

WP 1 – Production and Environment

Task 1.4 – Development of a real-time information management system

Dissemination Level:		
PP	Restricted to other programme participants (including the Commission Services)	
PU	**Public**	**Y**
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Document log

Version	Date	Comments	Author(s)
Version 1	01/09/2021	Table of contents	Fearghal O'Donncha
Version 2	23/09/2021	First draft	Fearghal O'Donncha
Version 3	04/10/2021	Revised draft	Fearghal O'Donncha
Version 4	20/10/2021	Revision 1	Roberto Pastres
Version 5	25/10/2021	Revision 2	Joao Ferreira
Version	28/10/2021	Final Document	Fearghal O'Donncha

Recommended Citation

O'Donncha F., Akhriev,. Information Management System: Final release. Deliverable 1.8. GAIN - Green Aquaculture INtensification in Europe. EU Horizon 2020 project grant nº. 773330. 29 pp.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

GLOSSARY OF ACRONYMS

Acronym	Definition
ABM	Aquaculture Biomass Monitor
API	Application Programming Interface
DO	Dissolved Oxygen
FastPCPImputer	fast principal component projection imputer
FCR	Feed Conversion Ratio
GAM	Generalised Additive Model
GUI	Graphical User Interface
HAB	Harmful Algal Bloom
IoT	Internet of Things
LRImputer	Low rank imputer
LSTM	Long short-term memory
ML	Machine Learning
MLP	Multilayer Perceptron
MQTT	Message Queue Telemetry Transport
RF	Random Forest
RNN	Recurrent Neural Networks
XGB	Extreme Gradient Boosting

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Table of Contents

Executive summary 5

1. Introduction 6

2. Structure and functionalities of the GAIN IMS 7

Data Integration 8
Data Pre-processing and Cleansing 9
Machine learning model forecasting 11
Machine learning 11
ML Model deployment, monitoring, and management 13

3. Results and dissemination 16

Trout farm management 16
Data driven insight into fish behaviour 17
Event forecasting for shellfish aquaculture 22

Conclusion 23

REFERENCES: 25

List of Figures 28

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Executive summary

This document describes the final release of the GAIN Information Management System – a cloud platform to help better manage aquaculture farms. The platform integrates sensor data from each pilot partner site into a secure cloud service and complements this with access to additional data streams such as weather, satellite or external model data. A suite of machine learning models were developed to address specific industry pain points and to add value to IoT sensor. The system interfaces with aquaculture-specific nodes developed by LLE and UNIVE. We present the user-experience of the service and the development of models that are provisioned to better inform aquaculture stakeholders.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

1. Introduction

This deliverable describes the final release of the GAIN Information Management System (IMS). It describes the components that comprise the system and how external users (pilot site partners) interact with the service. Further, it considers how the current system serves to improve the management of data collected, generated and pertinent to aquaculture farms. It builds on D1.5 presenting a first release of the platform (Fearghal O'Donncha *et al.*, 2019) and user feedback collected at month 15 and 36 of the project for Milestones 6 (O'Donncha and Gormally, 2019) and 13 (O'Donncha and Akhriev, 2021) respectively.

The GAIN IMS was designed as a distributed system that allowed different components to interact in a flexible and dynamic manner. At the end of GAIN, the IMS consists of three different components:

- A Big Data and AI component that ingests data from disparate sensor data and external sources to a unified cloud platform and processes using Artificial Intelligence (AI). This serves as the core for data ingestion and processing and is interfaced with other components or nodes.
- AquaSense: a component that provides dedicated service and decision support for shellfish aquaculture.
- AQUARADAR: a mechanistic modelling and data assimilation framework for land-based aquaculture systems.

GAIN considered the IMS in terms of two nodes (AquaSense and AQUARADAR) that provided dedicated aquaculture-specific management capabilities, interfaced with a data and AI platform. Extensibility was a core part of the development roadmap and additional nodes can be readily added to address specific user needs and to allow a flexible development ecosystem where many parties can develop (and commercialise) independently.

The document provides an overview of the GAIN Information Management System (IMS) and builds on the user's perspective presented in the MS13 document. We focus on the core data and AI platform to simplify technical presentation. Details on the AquaSense and AQUARADAR nodes are provided in our exploitation deliverables, namely D4.7 ("Website applying sensor data from WP1to model growth and environmental effects for key finfish and shellfish species") and D6.6 ("Industry focused website built on AquaSense engine, with a rich UI/UX"). To provide a more streamlined description of work (and avoid repetition), this deliverable focuses on the technical finalisation and release of the big data and AI platform. D4.7 and D6.6 describes the nodes developed as part of the commercialisation activities.

A key consideration for the GAIN IMS is relating data to operational conditions and farms and translating that information to improved decision making. The GAIN IMS provides a cloud-based data integration, analysis, and forecast service platform for aquaculture. The objectives of the data platform primarily relate to:

1. integration of data from disparate sources into a single unified cloud platform,
2. standardisation of IoT integration across all datatypes, data sources and data formats.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

3. empowerment of standardised data analysis (e.g., anomaly detection, statistical analysis) and bespoke model development (e.g., deep learning modelling, mechanistic models), and
4. dissemination of insight in a rapid, agile process.

This deliverable provides a general overview of the GAIN IMS and specifically how it impacts aquaculture activities; we focus on the core data and AI platform, with the distributed nodes of the system described in D4.7 and 6.6. Building on user feedback presented in MS13 report, this deliverable focuses on 1) the modelling and forecasting aspects and how they impact aquaculture activities and 2) the interaction of different stakeholders with the platform to amplify impact.

2. Structure and functionalities of the GAIN IMS

The GAIN project considers multiple aquaculture types from multiple geographical regions. Consequently, the data collected, and associated pain points of aquaculture farms were varied. We worked with partners to identify industry pain points (for marine finfish, shellfish, and land-based pond and raceway culture) and how an IoT (Internet of Things) and modelling framework could ameliorate those issues. Subsequently, we worked to identify commonality across aquaculture types and geographical locations: these commonalities served to guide development of a core service that considered data integration, curation, and forecasting of pertinent environmental variables such as temperature, dissolved oxygen, and Chlorophyll-a. This environmental forecasting service is in effect a scalable forecasting model, that ingests measurements from sensor (e.g. temperature sensor), parses and contextualises the data (by means of a contextual layer that allows the user to add meta-descriptors to the data to aid the development of models), and trains and stores a machine learning model to generate predictions at the requisite time-scale. The main features of the system are: (1) an efficient pipeline for ingesting IoT time series data in real time; (2) a scalable, hybrid data management service for both time series and contextual data; (3) a versatile semantic model for contextual information which can be easily adapted to different application domains; (4) an abstract framework for interacting with the system in R or Python; (5) deployment services which automatically train and/or score predictive models upon user-defined conditions.

The architecture of the data platform is composed of cloud-based microservices and provides an efficient pipeline for real-time data ingestion, time series, and model data management based on unified and intuitive application programming interfaces (APIs) to interact with the data and models.

Both the data scientist and the end user interact with the system using IBM Watson® Studio (Miller, 2019). This provides a unified user interface where the data scientist, the subject-matter expert, and the end user can collaborate and iterate to visualise and analyse (automated) time series forecasts, and augment with bespoke modelling systems that address specific farm requirements (as an example, in a related paper we describe how the system was used to interrogate the relationships between environmental data and hydroacoustic estimates of distribution of caged salmon (O'Donncha *et al.*, 2021). Figure 1 presents an overview of the different components of the service.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

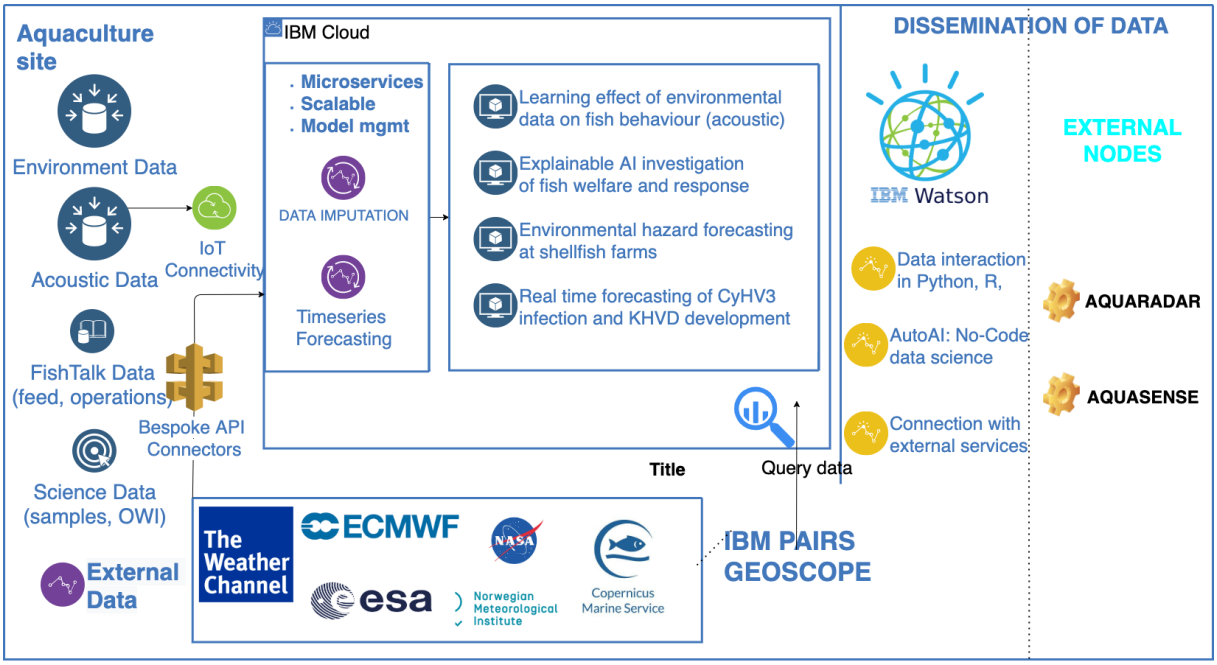

Figure 1: Overview of the system presenting IoT data integration, connectors to external data, microservices and automated model training and scheduling, and user interaction with the system in terms of bespoke (species and farm specific) model development, and dissemination to end user. Interaction with the end user is outlined on the right-hand side and is supported by robust data science and enterprise tools. Python and R packages allow the user to interact using open-source (e.g. Jupyter Notebooks) or commercial tools (e.g. IBM Watson Studio, Google Collab). AutoAI provides a "no-code" approach to allow non-data scientists apply AI to their datasets. External nodes (AQUARADAR, AQUASENSE) can upload and download data to the cloud to enable different levels of connectivity between different services (i.e. one-way connection where data such as sensor or model forecasts are downloaded from the cloud or two-way connection where data is downloaded and model forecasts uploaded and stored in the service).

The fundamental approach can be considered a series of steps:

1. Data integration from the farm to the cloud
2. Data pre-processing and cleansing
3. Machine learning model forecasting
4. Model deployment at the nodes, monitoring, and management.
5. The final step considers the dissemination of data to end user and is described in detail in Section 3

Data Integration

Data integration is described in detail in Deliverable 1.3 (O'Donncha and Purcell, 2019) and summarised briefly here. The Watson IoT platform provides a single point of ingress for all sensor data generated by the GAIN project. It utilizes the MQTT protocol for lightweight data transport and is designed to scale to thousands of devices in parallel with each device providing periodic or intermittent data updates.

The transfer of data to the cloud service was bespoke to each site adapting to the exact sensor configurations deployed.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

- The preferred approach was to push the data using Message Queue Telemetry Transport (MQTT), to the IBM cloud service, which is the standard pattern when handling IoT data.
- For data that were part of proprietary sensor ecosystem and did not facilitate publishing to an IoT service, we developed bespoke API adapters to pull data from vendors' cloud platform and push it into the IBM cloud service.

The approach provided lightweight access to farm data while developing a library of API connectors to widely used aquaculture sensor products (e.g. RealTimeAquaculture, CageEye).

Data Pre-processing and Cleansing

A key objective of our work is to enable forecasting with minimal human interaction. This requires the ability to request data based on a given context, automatically process and cleanse the data, and forward to an appropriate machine learning pipeline for model training and forecasting (or scoring in machine learning parlance). Data preparation is a core component of an applied data scientist's role with an oft-repeated trope that 80% of their time is spent cleaning data.

A core part of data cleansing is handling missing data. This is particularly true in time series applications where the fidelity of the signal depends on having complete coverage over the period. Many machine learning approaches such as recurrent neural networks (RNN) (Connor *et al.*, 1994) or LSTM (Gers *et al.*, 2000) require data at regular intervals to accurately learn the time series patterns (since they learn historical dependencies in the signal). Elementary masking functionality for LSTM is provided in libraries such as Keras but these simply skip the masked timesteps thereby corrupting time series fidelity (i.e. the autoregressive signal of regularly-spaced data).

As part of a more automated machine learning framework, we require the ability to impute missing data and rank the performance of different imputation algorithms. This is particularly true for aquaculture where missing data is an inherent part of marine sensor data. The fundamental structure involves requesting data from the GAIN service for a given **instance**, **entity**, and **signal**. The appropriate data cleansing and imputation approaches are then applied to those datasets and the processed data proceeds to a given machine learning forecasting pipeline. In the IoT contextual information domain, instance, entity, and signal refer to the different hierarchies of data meta descriptors. In our case, instance was a unique identifier for each farm, entity represented a given sensor id or descriptor, and signal denoted a unique variable (e.g., water temperature or dissolved oxygen).

The data imputation approaches considered the different characteristics of data quality issues in aquaculture. The reality of operating in a chaotic environment result in multiple classes of data quality issues: failures in power and connectivity results in data gaps of hours to days; sensor fouling and damage impedes data quality and often demands bias and error corrections; while the multiple temporal and spatial scales of ocean processes often require robust post-processing and noise removal strategies. These issues are amplified by the fact that collecting sufficient data is often difficult and one is rarely able to discount significant portions of data due to data quality issues – one instead desires to repair the data.

While a fundamental and long-studied problem, it is not straightforward to devise a formal protocol and to categorise the "best" imputation system. The difficulty primarily stems from

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

the fact that one requires a representative model of the original timeseries signal to enable reconstruction of a robust imputation – of course this is generally not available in practise. We considered a variety of data imputation approaches to augment model accuracy. These were:

- A Simple Imputer from Scikit-Learn Python package that substitutes a median value instead of a missing one,
- Linear interpolation
- Quadratic interpolation.
- Cubic interpolation and
- Polynomial interpolation of order 5.

We extended these imputation choices with two new imputers:

1. A fast principal component projection imputer (FastPCPImputer) and a
2. Low rank imputer (LRImputer)

Instead of interpolation at missing values, these imputers try to substitute the missing portion of a signal from other, uncorrupted parts of the same sequence. This is done via low-rank matrix approximation. Namely, we put the signal into a square matrix progressively row by row (possibly padding by NaNs at the end of last row). We call this matrix a *data matrix*. The rank of low-rank approximation is chosen as a square root of matrix size in either dimension.

The first imputer (FastPCPImputer) elaborates the idea described in the paper (Rodríguez and Wohlberg, 2013). The fundamental rationale of the approach considers recovering a low-rank matrix (the principal components) from a high-dimensional data matrix despite the presence of sparse errors (Zhou et al., 2010). This has natural applicability for time series data where one may expect repeated patterns at different frequencies such as day (solar radiation), week (traffic volumes) or year (annual or seasonal cycles). We adopt a brute-force grid search approach to select value of regularization parameter λ to achieve best match at uncorrupted entries of the time series.

The second imputer (LRImputer) extends our paper (Akhriev *et al.*, 2020), where low-rank approximation was achieved by decomposition of the data matrix into a product of two low-rank ones $L \cdot R$ (hence the name LR). We use robust loss function with a regularizer that promotes smoothness of the imputation result.

Both imputers are relatively fast. The Python implementation usually takes less than 10 seconds (often 2-3 seconds) on single-core CPU on time series with 10,000+ observations. This approach allows us to standardise data processing for machine learning models (since missing values are handled in an equivalent manner for every sensor), which helps the automation of these processes. Of course, there are many text books written on data cleansing and imputation. Our intention was not to reinvent the wheel but instead provide a pragmatic approach that allows us select from a library of imputation algorithms and choose the one that gives best performance. In our case, best performance was defined as the best performing forecasting model – in effect, we don't need a perfect reconstruction of the signal but such that the training of machine learning model is not hampered by data gaps (or biased by unrealistic imputations).

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Machine learning model forecasting

Machine learning and forecasting is a key component of the GAIN IMS. The user requests a specific dataset (based on farm and sensor context) for a given time period; that data is passed to the data cleansing pipeline above before being forward to the machine learning service.

Given sufficient data, ML models have the potential to successfully detect, quantify, and predict various phenomena in aquaculture. While physics-based modelling involves providing a set of inputs to a model which generates the corresponding outputs based on a non-linear mapping encoded from a set of governing equations, supervised machine learning instead learns the requisite mapping by being shown large number of corresponding inputs and outputs. In ML parlance, the model is trained by being shown a set of inputs (called features) and corresponding outputs (termed labels) from which it learns the prediction task -- in our case, given information on drivers of aquaculture or ocean variability (atmospheric conditions, historical values, etc.) we wish to predict specific variables, such as temperature or DO. With availability of sufficient data (of appropriate quality), the challenge reduces to selecting the appropriate ML model or algorithm and prescribing suitable model settings or *hyperparameters*. A model hyperparameter is a characteristic of a model that is external to the model and whose value cannot be estimated from data. In contrast, a parameter is an internal characteristic of the model, and its value can be estimated from data during training.

Machine learning

Classical works in machine learning and optimisation, introduced the "no free lunch" theorem (Wolpert and Evolutionary, 1997), demonstrating that no single machine learning algorithm can be universally better than any other in all domains – in effect, one must try multiple models and find one that works best for a particular problem.

The uncertainties inherent to any model algorithm means that the model provides an estimate of the true state. Ensembles generated either by perturbing model inputs or combining different models or algorithms can serve to better capture the true solution state. Considering different model algorithms allows flexibility in terms of final model to select or implementation of ensemble aggregation (O'Donncha *et al.*, 2018; F. O'Donncha *et al.*, 2019) or uncertainty quantification approaches.

Ensemble approaches enhances robustness. Further, our microservices-based platform aligns naturally with multi-model approach, where different algorithms are different choices to the system with the final decision based on which model performs best. In GAIN we considered four different machine learning algorithms to predict ocean variables: Generalised Additive Models (GAM), Random Forest (RF) XGBoost, and Multi-Layer Perceptron (MLP). Extensive details on these algorithms are provided in many statistical and machine learning textbooks (Friedman, Hastie and Tibshirani, 2001; Goodfellow, Bengio and Courville, 2016), and considerations for their application to ocean datasets are described in (Wolff, O'Donncha and Chen, 2020). The models are briefly summarised here.

Generalised Additive Models (GAMs) extend on linear models by relating the outcome to unknown smooth *functions* of the features or inputs. Predicting y from the vector of covariates **x**, at time t is as (Hastie and Tibshirani, 1986):

$$g(y) = \alpha + f_1(x_1) + f_2(x_2) + \ldots + f_i(x_i) + \epsilon$$

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

where each $f_i(\cdot)$ is an unspecified function and $g(\cdot)$ is a link function defining how the response variable relates to the linear predictor of explanatory variables (e.g. binomial, normal, Poisson) (Wijaya, Sinn and Chen, 2015).

Random Forest (RF) is a classification and regression method based on the *aggregation* of many decision trees. Decision trees are a conceptually simple yet powerful prediction tool that breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The resulting intuitive pathway from explanatory variables to outcome serves to provide an easily interpretable model. In RF (Breiman, 2001), each tree is a standard Classification or Regression Tree (CART) that uses what is termed node "impurity" as a splitting criterion and selects the splitting predictor from a randomly selected subset of predictors. Each node in the regression tree corresponds to the average of the response within the subdomains of the features corresponding to that node. The node impurity gives a measure of how badly the observations at a given node fit the model. In regression trees, this is typically measured by the residual sum of squares within that node. Each tree is constructed from a bootstrap sample drawn with replacement from the original data set, and the predictions of all trees are finally aggregated through majority voting (Boulesteix *et al.*, 2012).

While XGBoost shares many characteristics and advantages with RF (namely interpretability, predictive performance, and simplicity), a key difference facilitating performance gain is that decision trees are built *sequentially* rather than *independently*. The tree ensemble model follows a similar framework to RF with prediction of the form (Chen *et al.*, 2016):

$$\hat{y}_i\ \phi(x_i) = \sum_{k=1}^{K} f_k(x_i),\ \ f_k \in F$$

where we consider K trees, $F = f(x) = w_{q(x)}$ represents a set of classification and regression trees (CART), q represents each independent decision-tree structure, and $w_{q(x)}$ is the weight of the leaf which is assigned to the input x.

An MLP model is organised in sequential layers made up of interconnected neurons, each consisting of a *weight* and *bias* term that allows the network to learn highly nonlinear patterns in data (shown schematically in Figure 2). A loss function is defined in terms of the squared error between the observations and the machine-learning prediction (plus a weights regularisation contribution). By minimising the loss function, the supervised machine learning algorithm identifies the mapping between the predictors and the predictands. The machine learning model is trained on a data set to establish the weights parameterising the space of nonlinear functions mapping from **X** to y.

$$\vartheta = \frac{1}{n}\sum_{i=1}^{n}(\hat{\boldsymbol{y}}_i - \boldsymbol{y}_i)^2 + \lambda R(\Theta),$$

Where $\hat{\boldsymbol{y}}_i$ denotes observations or labels, $\boldsymbol{y}_i$ represents our model prediction and $R(\Theta)$, represents our regularisation term (parameterised by λ). The regularization term penalizes complex models by enforcing weight decay, which prevents the magnitude of the weight vector from growing too large because large weights can lead to overfitting.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

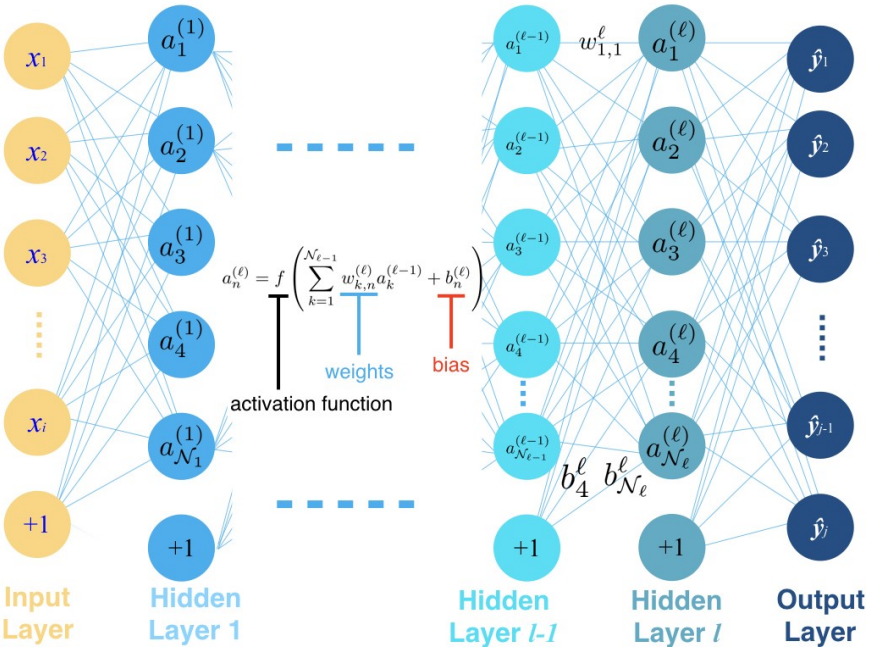

Figure 2: Schematic of an MLP machine learning network.

ML Model deployment, monitoring, and management

The machine learning algorithms described in the previous section were applied to the data collected at each pilot site (Service *et al.*, 2019). The model training, deployment, and monitoring were managed by a flexible model management cloud infrastructure described in detail in (Chen *et al.*, 2018). For each model implementation, pertinent data was requested from the server, preprocessed as described above to remove outliers and impute missing values, converted into appropriate time-aligned matrices for the model implementation, and used to train the machine learning model (or update a previously trained model). The trained model was then stored in a model management database and deployed in forecasting mode against a verification dataset to evaluate performance metrics and goodness-of-fit. Trained models are tagged and stored allowing flexibility to select different model iterations (e.g., roll back to a model that we know gives a certain level of performance while also testing new model implementations). At all stages, a variety of preprocessing and forecasting algorithms are available to the system to enhance forecasting skill and allow us to address many different variables and conditions.

This flexible approach that simplified model training and deployment, allowed us to rapidly test many different model implementations and scenarios. Hyperparameter optimisation (the tuning of model parameters external to the model learning algorithm) was done using a greedy grid search approach that searched over a user-defined range of hyperparameters with an embarrassingly parallel approach. Pywren (Jonas *et al.*, 2017) managed the search, which then returned the parameters that minimised MSE for the training dataset, using a MapReduce operation.

The entire process is guided by relatively simple configuration files. An example is provided in the following for our UNIVE trout farm in Italy. Basic details are provided regarding farm and sensor identifiers. This is coupled with information such as how long to make forecast (here we make forecast for 12, 24, 60, and 240 hour horizons), what imputer properties to adopt

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

(lowrank), and specialised details regarding the algorithm choice itself. Of key importance here is standardisation across all sites, and that the approach lends itself to automation.

```
#-----------------------------------------------
visible_model_name: model6_xgb_2021.07.08_22.25.44

instance: UNIVE # Identifier for the farm site
entity: FARM_SENSOR # Context identifier for sensor
signal: WATER_TEMPERATURE # Context identifier for variable
geography: GIS_POINT , latitude: 46.045 , longitude: 10.759

distribution_name: gain-models # Cloud space to store models
# details on model training and update
train_repeat: None
train_task: train
train_time: 2021-07-08T22:25:44+00:00
# details on model scoring or forecasting
score_repeat: 1_hours
score_task: score
score_time: 2021-07-08T22:25:44+00:00

# parameters passed to the machine learning model
user_parameters:
{
     'algo_params': {'algorithm': 'xgb',
                     'collect_feature_importance': 0,
                     'lag_nsteps': 36,
                     'learning_rate': 0.05,
                     'max_depth': 10,
                     'n_estimators': 500,
                     'num_jobs': 1,
                     'reg_lambda': 0.5,
                     'scale_y_labels': True},
      'deep_learning_env': True,
      'forecast_horizons': [12, 24, 60, 240],
      'frequency': 'H',
      'historical': None,
      'imputer': {'return_lowrank': True,
                  'seasonality': 24,
                  'skip_long_gaps': True,
                  'type': 'pcp2'},
     'min_train_nsamples': 1000,
      'retrain_period_hours': 72,
      'run_locally': False,
      'timezone': 'Europe/Rome',
      'train_ndays': 365,
      'verbose': 1,
      'visible_model_name': 'model6_xgb_2021.07.08_22.25.44'
}
```

Figure 3: Sample configuration file for the GAIN modelling service. The key factors the user needs to consider

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

are the data we provide to the model (given sensor or data context information), imputer properties to handle missing or corrupted data, and selection of algorithm (or use the default choice of random forest).

The above allows for a simple deployment scheme for models. A related topic is management of the different models. What we required was a framework that simplified the development and deployment of machine learning models, as well as information on performance of the different models to guide development. Model monitoring consisted of both a programmatic and graphical approach. The service API allowed to readily access different model ids and versions and interrogate performance against most recent observations. Simple metrics such as Root Mean Square Error were returned to the user to provide summary metrics. A Gain graphical user interface (GUI) was used to help the scientist rapidly compare different models visually. Figure 4 presents a screenshot from this GUI giving an overview of some of the information contained.

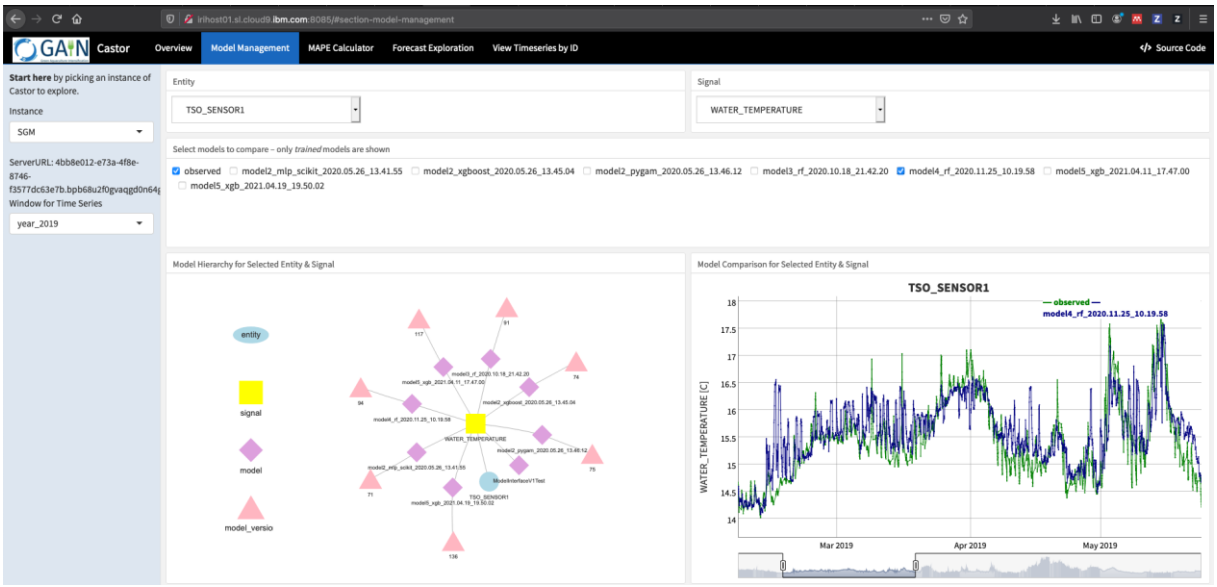

Figure 4: Screenshot from beta GAIN GUI that allowed users to visualise forecast results against observations. The panel on left hand side allows users to select a particular instance (or farm site) and period, while the upper panel provides dropdown list of entity (sensor) and signal (variable). The panel on the bottom left primarily relates to model management and can be considered a tool of the data scientist rather than the end user. This presents information on the different models that are trained (for this example it reports different machine learning models including Generalised Additive Model (GAM), XGBoost, and Random Forest), as well as different versions of those models to allow for iterative improvement and fine tuning of those models. It allows the user to easily select different model algorithms, as well as different implementations and versions of these models. This combination of complexity (many different models with different configurations) and simplicity (an easy comparison and analysis of which models performs well) provided us with a valuable tool for rapid and iterative application of machine learning to aquaculture.

We considered the combination of graphical and programmatic approach provide a unique selling point to allow us to adapt to the complexity of aquaculture across different species, regions, and scales. The GUI above provides a high-level overview of the system to the developer while we developed and extended a Python (and R) package that allows external users to interact with data in similar manner. While the python and R package can be deployed any local system (one simply installs Python/R and the relevant libraries), we used IBM Watson Studio (IBM, 2021) for all analysis. The primary advantage of this approach is it provides a

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

unified environment where the data scientist and domain expert can collaborate to address specific challenges. We discuss these applications in detail in the next section.

3. Results and dissemination

The GAIN project focused on developing a flexible data integration, processing and forecasting system that was specialised to the requirements of aquaculture (many different data streams, noisy uncertain and missing data, multiple forecasting variables and pain points). Of utmost importance here is the ability to interface GAIN forecasting models and assets within existing operations and software assets employed on the farm (e.g., using forecasts of environmental variables for fish growth estimates, integrating analysis of fish welfare with veterinary decision making).

For all deployments, we used IBM Watson Studio since it's a widely used IBM product across many industries. Users create an account on Watson Studio (using the free "Lite" plan) and log-in through that system. We then create a unique space for each different GAIN pilot site. All data and GAIN model forecasts pertinent to that farm are accessible within that space (but users cannot access data from another pilot site) and we extend those with site-specific interrogation and analysis that addresses the complexities of different farms and species.

In this section, we consider different examples of these.

Trout farm management

At the trout raceway farm in Preore, Italy, inlet freshwater that flows through the raceways comes from the Sarca river. If this ensures quite constant levels of Dissolved Oxygen it must be noticed that inlet water presents high dynamics in terms of temperature due to short-term (daily irradiance oscillation) and long-term (seasonal rhythm) influences.

From such a perspective, and in conjunction with our pilot site partners, we identified a need for 10-14 day-ahead forecasts of water temperature to support decision in terms of:

- Fish vaccination: The main pathogen impacting trout during their growth is a bacterium, *Lactococcus garvieae*, which development became more aggressive when daily minimum temperature is higher than 15°C.
- Disease mitigation: Disease mitigation in fingerlings is primarily based on the control of viral pathogens development which are strongly influenced by water temperature.
- Feed quantity: Day to day high variations in water temperature sometimes occurs due to meteorological events and can have high negative impacts on trouts' ability to correctly assimilate supplied feed.
- Fish growth forecasts: GAIN partners have extensive expertise and a diverse portfolio of fish growth models. These models require reliable estimates of water temperature as inputs.

Forecasts of water temperature are available through Watson Studio with an example provided here. Figure 5 presents a snapshot of observed and forecasted values of temperature for a 10-day ahead forecast. The basic structure is (for a given sensor) to interrogate the model database for models attached to this sensor; we then request data for those forecasts and visualise.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

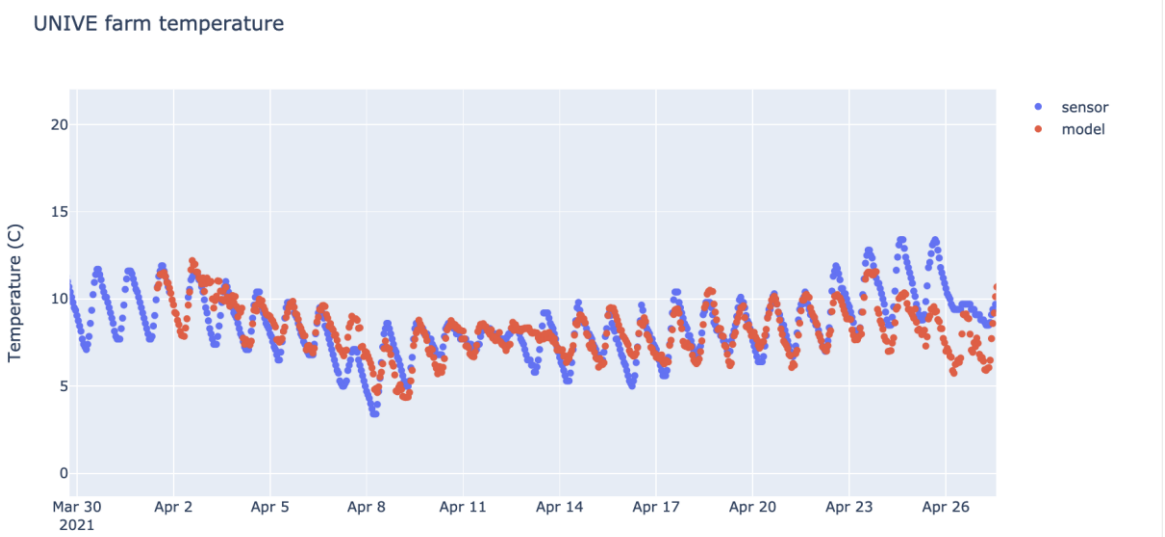

Figure 5: Comparison of observed temperature against modelled values from the UNIVE trout farm. Blue curve denotes sensor observations while red indicate model estimate

One of the key advantages of this approach is extensibility. We can readily collaborate with domain experts, iterate on different models and analysis, and any updates are applied immediately.

Data driven insight into fish behaviour

In a recent paper (O'Donncha *et al.*, 2021), we presented a scientific analysis of the environmental drivers of fish behaviour in a cage. Data from hydroacoustic and environmental sensors were interrogated using statistical and machine learning approaches.

Hydroacoustic methods provide a proxy measure for density and distribution of marine animals in form of acoustic backscattering (Foote, 2009). The fundamental principle is based on emitting a signal of known type and power level from a transducer. As it encounters regions of the medium with differing properties, also called heterogeneities, the sound is generally redistributed, or scattered, in all directions. This makes possible detection of the scattered sound with transducer and suitable receiver electronics. Advantages linked to hydroacoustic sampling techniques include, high spatial and temporal resolution, autonomous long-term sampling duration, range (especially during poor visibility when visual-based methods tend to fail), and a non-invasive surveying approach (Scherelis *et al.*, 2020). Given these advantages, hydroacoustics is increasingly used to characterise animal behaviour in the marine environment, and considered a promising system to improve management of aquaculture farms (Juell, Furevik and Bjordal, 1993).

In GAIN, hydroacoustic data were collected by one of two sensors ``CageEye" (Scherelis *et al.*, 2020) or "Aquaculture Biomass Monitor". Broadly speaking, processed hydroacoustic data generates two metrics: volume backscattering strength (S_v), is often considered as a proxy for fish biomass; while target strength (TS) is an acoustic measure of fish length (Simmonds and MacLennan, 2008)si. TS is a measure of the acoustic reflectivity of a fish, which varies depending on the presence of a swim bladder and on the size, behaviour, morphology, and physiology of the fish. These outputs can be used to generate estimates of fish density and biomass (Boswell, Wilson and Wilson, 2007) within a cage.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Figure 6 presents a density plot of daytime and nighttime fish positions for both CAN and NOR (due to lack of nighttime observations, SCO was excluded). To remove the effects of long sunshine hours during June and July in NOR these two months were excluded from the plot. Results demonstrated a clear difference between daytime and night-time behaviour for the CAN site and a similar but much less pronounced difference for the NOR site. In Canada, fish congregated at about 3.6m depth and the spread around this was quite narrow during the day, while at night, fish were distributed more widely across the water column with a mean depth of 2.8m. Similar trends were observed in Norway (although not as pronounced). The mean difference between daytime and night-time positions were 0.52m while fish were also more uniformly spread across the water column at night.

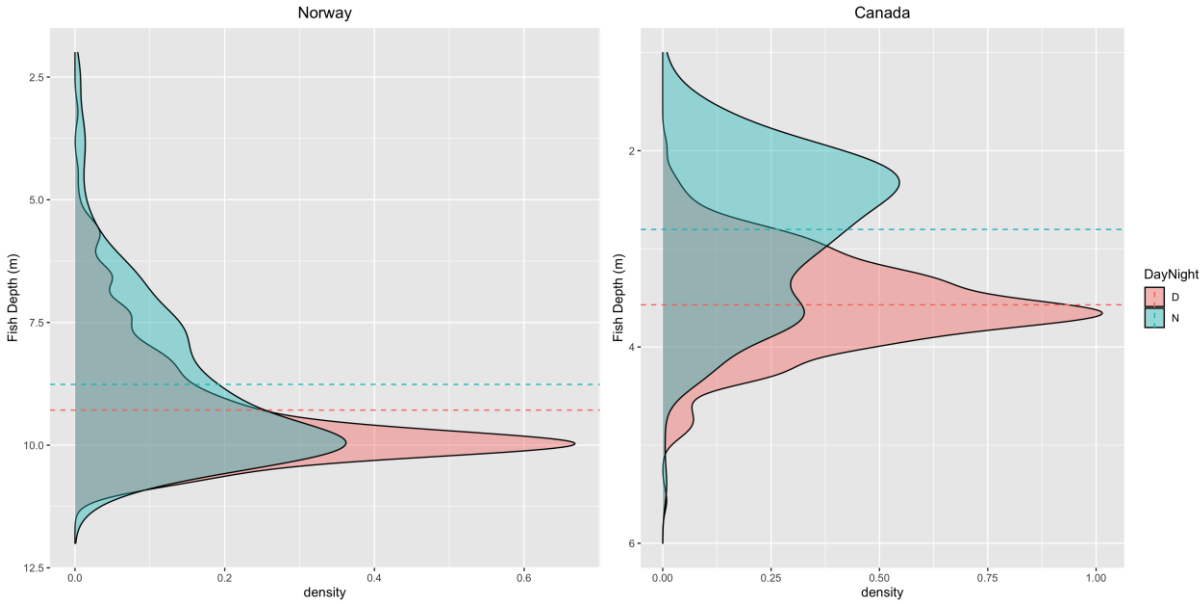

Figure 6: Distribution of fish depth data for the NOR (left) and CAN (right) farm over the duration of the study period. The data is split into daytime and nighttime periods to explore how behaviours vary between those periods. The dashed vertical lines denote the mean for both periods

We interrogated relationships between vertical distribution of fish in a cage (as sampled by the CageEye system or the Aquaculture Biomass Monitor system), and environmental variables at the three GAIN salmon sites (Norway, Scotland, and Canada). Statistical analysis explored the diel patterns, and how data distributions varied over the duration of the study, while IBM AutoAI was used to quantify the effects of environmental variations on the vertical movement of the fish.

Gartner -- the respected research and advisory firm for enterprise -- identified the automation of ML model deployments as one of the ten key technology trends for 2020 (Cearley *et al.*, 2019). Termed AutoML or AutoAI, these approaches aims to help automate the steps and processes involved in the life cycle of creating, deploying, managing, and operating AI models (Dickson, 2020). Gartner highlighted its capabilities for "democratizing AI" by enabling development of low-code ML models that do not require high levels of data science experience to setup and parametrise the models (Cearley *et al.*, 2019). A variety of AutoML or AutoAI products exist with the most prominent being IBM's AutoAI, Google's autoML, and H20.ai's H2O.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

The fundamental idea of AutoAI approaches can be considered as "AI for AI". Using machine learning, it aims to interrogate user data and discover the optimal structures, data transformation, and tunable parameters (or hyperparameters) for machine learning regression and classification. AutoAI approaches are particularly valuable for benchmarking studies since they can be easily replicated by others, and don't require high levels of data science expertise. Many of the tools such as IBM AutoAI offer free plans that are particularly amenable towards scientific and academic studies.

Sensor and pertinent model data (of environmental variables) are extracted from the GAIN cloud service and fed to the AutoAI framework via a “one-click” deployment. AutoAI interrogates various transformations on the data (normalisation, logarithmic scaling, principal component analysis, etc.) and different machine learning algorithms (Random Forest, Gradient Boosting, XGBoost, etc.). The system then returns the optimal model pipeline and user can view via a simple relationship map summarising the different pipelines and predictive skill.

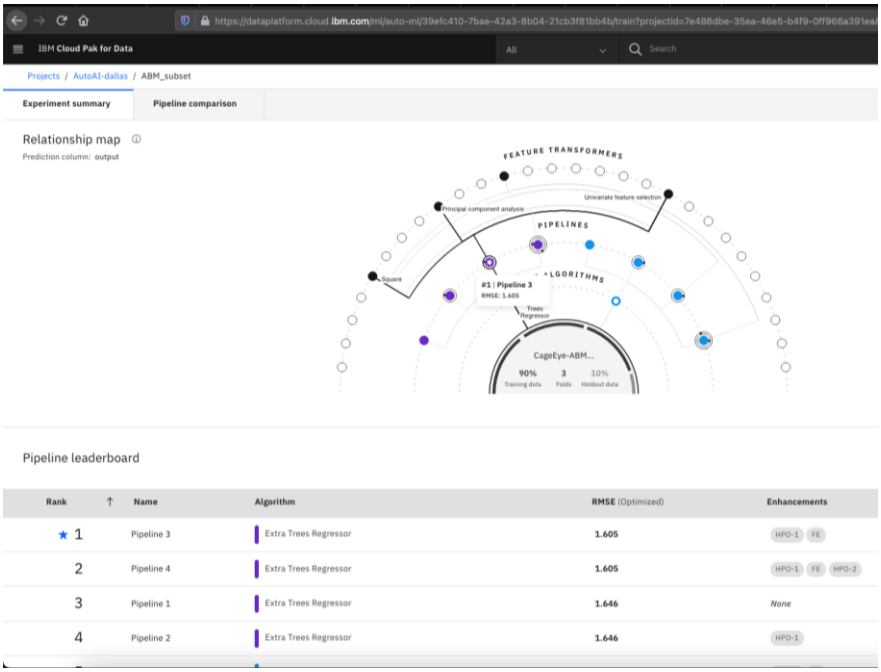

Figure 7: Visualisation of AutoAI results interrogating fish response to environmental conditions at the Norway site. The relationship map illustrates the different “pipelines” (data transformation and machine learning algorithms) that were explored while the Pipeline leaderboard displays the best performing model. Results comparable to what can be achieved by a skilled data scientist can be achieved without need for any code.

For the machine learning interrogation, we provided input features that literature suggests influence salmon behaviour (and were available at the study sites). For our study, these were temperature, DO, current speed, wind speed, and salinity, together with hour-of-day. The resultant model explained 59%, 64% and 61% of variance for the Norway, Scotland, and Canada sites, respectively.

Figure 8 summarises model performance at the Canada site. It illustrates that the model captures data trends quite well reporting correlation score of 0.78. Visually, the model captures observed fish depth quite well considering the highly dynamic nature of the signal. In particular, trends in the data are adequately tracked and the model accurately replicates whether the fish move up or down in the cage in response to the provided model inputs. While this provides information

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

on the predictability of the signal, our objective is to use this information to interrogate key drivers of fish response.

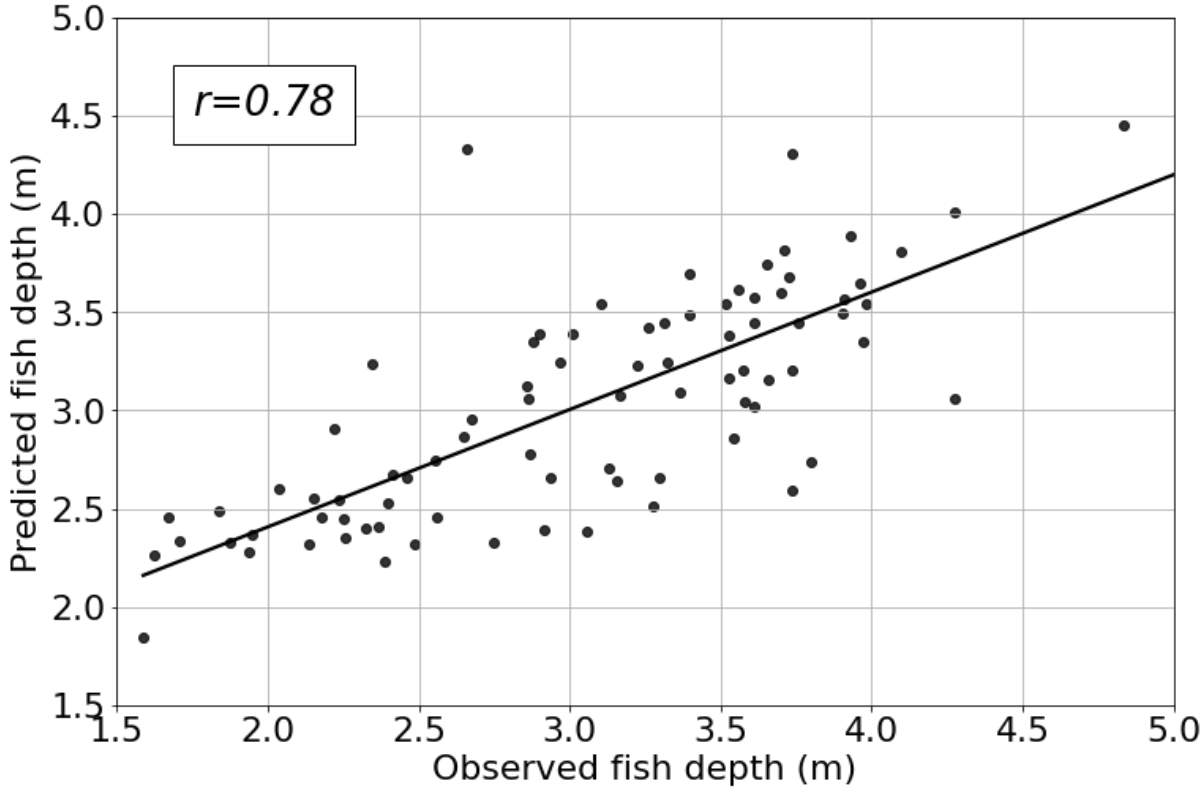

Figure 8: Scatter plot of model predicted fish depth plotted against observed values for the CAN site. Inputs to the model are environmental data time-aligned with the target data, and hour of day to represent temporal variations

We analyse this in detail using *feature importance* analysis and *Accumulated Local Effects* (ALE) interrogation (see our paper (O'Donncha *et al.*, 2021) for details on the ALE analysis which are omitted here for brevity). Feature importance measure computes the contribution or importance of each feature by calculating the increase of the model's prediction error after permuting the feature. A feature is "important" if permuting its values increases the model error, because the model relied on the feature for the prediction. A feature is "unimportant" if permuting its values keeps the model error unchanged, because the model ignored the feature for the prediction (Breiman, 2001).

Figure 9 presents the variable importance computed for the three locations in Norway, Scotland, and Canada. While there were similarities in the drivers that influenced fish position at the three sites, pronounced variations existed based on the different geography and characteristics of each site. As suggested by both feature importance analysis and boxplot visualisation, time-of-day was a primary driver, particularly at the Canadian farm. At all sites, physical oceanographic variables represented an important driver. Physical mixing by current speeds and wind forcing were particularly critical at the Canada site and three of the five most important variables represented physical stresses and mechanical mixing, namely current direction, wind direction, and wind speed, respectively (in order of influence). Wind stress did not represent an important driver of fish depth variance at the Norway site. This is likely due to the increased depth of cage and fish position serving to shelter from local surface dynamics.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Interestingly, salinity was the primary driver of fish position at the Norway site which illustrates both fish sensitivities and local bay characteristics. Analysis of temperature data illustrated a pronounced thermal stratification during the summer months, that breaks down into a well-mixed water column in spring and autumn. Variations of vertical salinity were more complex illustrating relatively low surface salinity values in September, which may be influenced by precipitation or freshwater runoff. Literature indicates that Atlantic salmon are influenced by salinity variations when younger than three months and during spawning periods, while indifferent to salinity at other times (Oppedal *et al.*, 2010). The behavioural influence detected in this study may be a result of salmon expressing preference for lower salinity waters in spring, during the return migration period of salmon towards freshwater. However, analysis indicated that the vertical variation in salinity was relatively small, and additional study is necessary to understand the influence this may have on salmon variations.

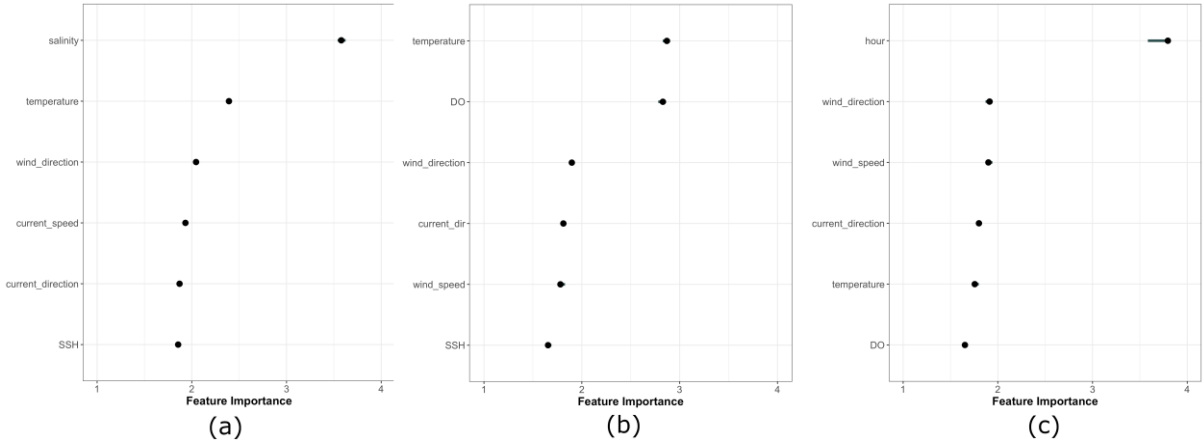

Figure 9: Feature importance reported at the (a) Norway, (b) Scotland, and (c) Canada sites. The y-axis reports the ranked list of features that contributed the most to variation in fish depth measurements, while the x-axis presents relative magnitude of those contributions. Ranking predictors in this manner can quickly help sift through large datasets and understand data trends (Kuhn and Johnson, 2013)

We implemented this bespoke analysis in Watson Studio (summarised here) that allowed us 1) collaborate with domain experts in GIFAS, UoS and DAL to enhance the data science analysis and 2) disseminate results rapidly in a flexible manner.

We also worked to open source the analysis (and the data). Since the data was commercial and sensitive in nature, we were not able to release data from all the farms. Instead, we made public all data from one site: GIFAS – which being a research and operational farm had more flexibility towards open science approaches.

The code and data are available here with complete instructions on how to install and run. We extended this further as part of WP5 activities around skills development. Hence, we also developed introductory data science courses that allows one with no data science experience to learn the basics and implement the analysis on real world aquaculture datasets (ideally using similar structure but applying to their own data). These code lessons are available here with a complete overview provided in our open.edu course here

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Event forecasting for shellfish aquaculture

Adverse water quality events are a major factor for shellfish farms. GAIN involves two shellfish farms with quite different characteristics. While both are subject to toxins from external sources that pose significant challenge for farmers, the sources of those are quite different:

- For SGM, harmful algal bloom events are the primary source of adverse water quality events. These are closely connected to upwelling conditions emanating from offshore drivers.
- Currently HABs are less of a concern at the AFBI site in Northern Ireland although this may change because of climate change. Instead, toxin events at AFBI farms are highly dependent on runoff from agricultural and urban processes. In turn these are driven by precipitation, river levels, and land surface information.

While forecasts of environmental variables are valuable for shellfish farm decision making, we are especially interested in how we can use this data to inform decision making. Working with SGM and AFBI pilot partners, we identified the forecasting of harmful algal blooms a key concern.

Obviously, the forecasting of algal blooms is extremely complex involving highly non-linear physical, chemical, and biological processes. While research on forecasting of HABs is extensive, deploying in operations is challenging. We took a more pragmatic approach that aims to identify conditions that lead to closure of shellfish sites due to measured toxins in shellfish. This data is collected by operators and historical data is provided by the Portuguese Institute for Sea and Atmosphere (IPMA) going back to 2014 (available here).

We developed a preliminary forecasting model to provide early warning of closure due to toxins in shellfish. The training label data was historical information on closures above while a variety of environmental data were provided as features or inputs to the models. These data included sensor data collected at both sites (combined with time series forecasting models), open-ocean model data from E.U. Copernicus Marine Service, and wind data from IBM Weather Operations Center. Machine learning forecasting of environmental variables were used to "fill in the gaps" when there was missing data in the sensor time series (and naturally it's a key step to allow this to deployment as forecasting tool). We use this information to develop a classification model to forecast the likelihood that the farm will be closed due to toxins (based on historical closures).

A preliminary demonstration is available here reporting accuracy of 87% on the test data.

A key consideration for these complex machine learning deployments is interpretability. We wish to also know why model makes a particular forecast. Figure 10 presents the *feature importance* for the above model. The model predicts the likelihood of shellfish site closure due to presence of toxins while Figure 10 ranks the environmental drivers that influence this event occurring. The below results suggest that the main environmental variables that influence closures are water temperature, salinity, net primary productivity, water levels, and Chlorophyll-a.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

```
shap_explainer = shap.TreeExplainer(xgb)
shap_values = shap_explainer.shap_values(X_train_scale)
shap.summary_plot(shap_values, X_train_scale, feature_names=X_test.columns)
```

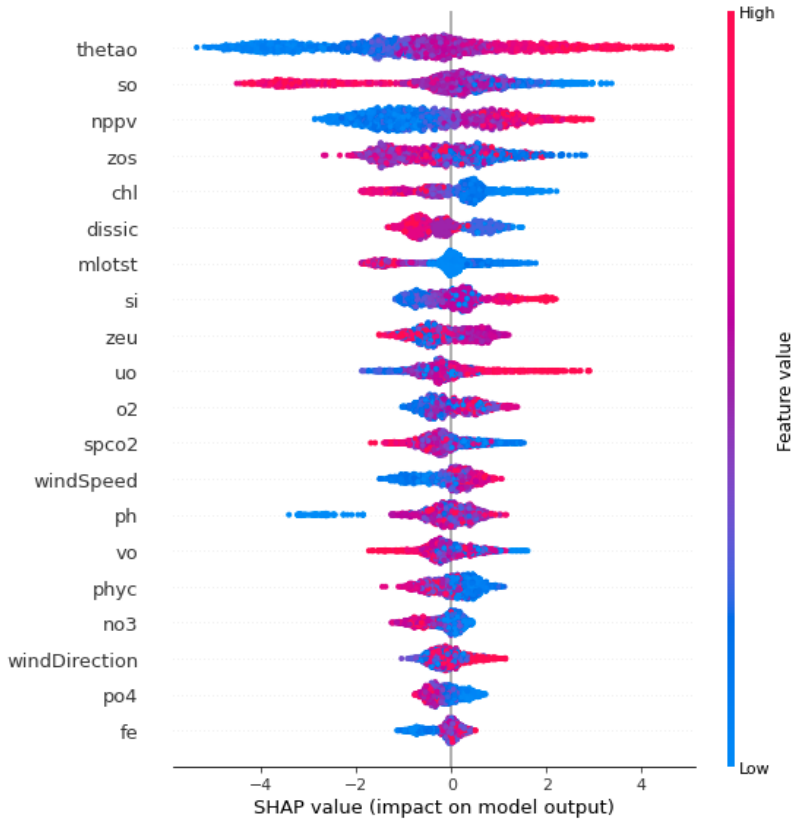

Figure 10: Feature importance for the occurrence of toxins at the SGM shellfish site in Portugal

Conclusion

This document provides an overview of the GAIN cloud service for aquaculture. The document summarises key functionality supported by the service and how those address farmers' needs. Further, user interaction with the service is described which presents a valuable roadmap to dissemination of results to aquaculture stakeholders. In particular, the disparate needs of aquaculture are described and different strategies to address these needs presented.

The management system developed as part of GAIN make a number of contributions towards extracting actionable insight using IoT and operational aquaculture data:

- Interoperability poses a significant challenge as sensors currently cover a wide range of types, suppliers, and levels of sophistication. This extends from legacy sensors storing data in on-board data loggers, to modern sensor stacks reporting in proprietary format, to dedicated cloud platforms. GAIN is committed to an open-standards approach based on standard IoT protocol, that enables rapid extraction of data using common programming environments (Python, R, MATLAB, etc.).
- The fragmented nature of the sensor industry can be significantly enhanced by a unified cloud service to enable access to all datasets and empower users with data

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

sovereignty. Currently, much of the data being collected on farms are in proprietary formats (e.g., HAC format is a common format for hydroacoustic data for which data readers are not readily available), which means users have limited access to data. Indeed, it took quite a lot of effort on the part of GAIN partners to gain access to the data collected by the CageEye and ABM sensor as "vendor lock-in" meant that data was only accessible (visually) through proprietary software. The GAIN service provides users with complete sovereignty over all data being collected on farms in a standardised, interoperable format.

- Interoperability with different services is a key necessity. GAIN presents a roadmap in that direction with different nodes (AQUARADAR, AquaSense) addressing different parts of the aquaculture industry. Precisions aquaculture is dependent on connecting disparate data and innovations to better inform all parts of the aquaculture value chain
- The large variety of aquaculture types and different challenges facing different geographies makes a "one-size-fits-all" solution challenging. Instead, it requires a solution that is modular and extendable, while being interoperable with different aquaculture services.
- At its core, precision aquaculture is dependent on leveraging IoT technologies to move beyond data towards insight. By integrating data from heterogeneous, disparate sources into a unified cloud platform, it allows the extraction of insight from data.

Deriving from these key objectives, this document (and indeed the core GAIN cloud service) focuses on four fundamental components:

1. Data integration from the farm to the cloud
2. Data pre-processing and cleansing to convert to format amenable towards machine learning interrogation
3. Machine learning model development, deployment, monitoring, and management,
4. Results dissemination to end users and interoperability with existing aquaculture and enterprise management software

These contributions can enable farms better manage their operations, while also enhancing data sovereignty, interoperability and standardisation. At its core is modularity and extensibility which allows different components of the GAIN stack be applied at farms and that can readily be extended with bespoke analysis. The GAIN roadmap for industry impact extends on this deliverable in D4.7 and D6.6.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

REFERENCES:

Akhriev, A. *et al.* (2020) 'Pursuit of low-rank models of time-varying matrices robust to sparse and measurement noise', in *AAAI Conference on Artificial Intelligence*, pp. 3171–3178. Available at: https://ojs.aaai.org/index.php/AAAI/article/view/5714 (Accessed: 30 September 2021).

Boswell, K. M., Wilson, M. P. and Wilson, C. A. (2007) 'Hydroacoustics as a tool for assessing fish biomass and size distribution associated with discrete shallow water estuarine habitats in Louisiana', *Estuaries and Coasts*, 30(4), pp. 607–617. doi: 10.1007/BF02841958.

Boulesteix, A. L. *et al.* (2012) 'Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics', *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(6), pp. 493–507. doi: 10.1002/WIDM.1072.

Breiman, L. (2001) 'Random Forests', *Machine Learning*, 45(1), pp. 5–32. doi: 10.1023/A:1010933404324.

Cearley, D. *et al.* (2019) *Gartner top 10 strategic technology trends for 2020-Smarter with Gartner*.

Chen, B. *et al.* (2018) 'Castor: Contextual IoT Time Series Data and Model Management at Scale', in *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, pp. 1487–1492. doi: 10.1109/ICDMW.2018.00213.

Chen, T. *et al.* (2016) 'Xgboost: A scalable tree boosting system', in *22nd {{ACM}} sigkdd international conference on knowledge discovery and data mining*. Association for Computing Machinery, pp. 785–794. doi: 10.1145/2939672.2939785.

Connor, J. *et al.* (1994) 'Recurrent neural networks and robust time series prediction', *ieeexplore.ieee.org*, 5(2). Available at: https://ieeexplore.ieee.org/abstract/document/279188/ (Accessed: 30 September 2021).

Dickson, B. (2020) *8 biggest AI trends of 2020, according to experts*, *The Next Web*. Available at: https://thenextweb.com/news/8-biggest-ai-trends-of-2020-according-to-experts (Accessed: 28 September 2021).

Foote, K. G. (2009) 'Acoustic Methods: Brief Review and Prospects for Advancing Fisheries Research', in *The Future of Fisheries Science in North America*. Springer Netherlands, pp. 313–343. doi: 10.1007/978-1-4020-9210-7_18.

Friedman, J., Hastie, T. and Tibshirani, R. (2001) *The elements of statistical learning*. Available at: http://statweb.stanford.edu/~tibs/book/preface.ps (Accessed: 28 January 2020).

Gers, F. *et al.* (2000) 'Learning to forget: Continual prediction with LSTM', *ieeexplore.ieee.org*, 12(10), pp. 2451–2471. doi: 10.1162/089976600300015015.

Goodfellow, I., Bengio, Y. and Courville, A. (2016) *Deep learning*. MIT press. Available at: https://books.google.co.uk/books?hl=en&lr=&id=omivDQAAQBAJ&oi=fnd&pg=PR5&dq=goodfellow+deep+learning&ots=MMR2imlINV&sig=ygji347eXNZRvEcWQsbWlcNuNS8 (Accessed: 9 April 2019).

Hastie, T. J. and Tibshirani, R. (1986) 'Generalized Additive Models', *Statistical Science*, 1(3), pp. 297–318. doi: 10.1201/9780203738535-7.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

IBM (2021) *Watson Studio | IBM*. Available at: https://www.ibm.com/cloud/watson-studio (Accessed: 27 April 2021).

Jonas, E. *et al.* (2017) 'Occupy the cloud: Distributed computing for the 99%', in *2017 Symposium on Cloud Computing*. Association for Computing Machinery, Inc, pp. 445–451. doi: 10.1145/3127479.3128601.

Juell, J., Furevik, D. and Bjordal, A. (1993) 'Demand feeding in salmon farming by hydroacoustic food detection', *Aquacultural Engineering*, 12(3), pp. 155–167. doi: https://doi.org/10.1016/0144-8609(93)90008-Y.

Kuhn, M. and Johnson, K. (2013) *Applied predictive modeling*. Springer. New York. Available at: https://link.springer.com/content/pdf/10.1007/978-1-4614-6849-3.pdf (Accessed: 4 October 2021).

Miller, J. (2019) *Hands-On Machine Learning with IBM Watson: Leverage IBM Watson to implement machine learning techniques and algorithms using Python*. Available at: https://books.google.com/books?hl=en&lr=&id=p86PDwAAQBAJ&oi=fnd&pg=PP1&dq=ibm+watson+studio&ots=70iiDEbcrk&sig=6_qPxcUOBC6eUuplzcx9uDTBceY (Accessed: 23 September 2021).

O'Donncha, F. *et al.* (2018) 'An integrated framework that combines machine learning and numerical models to improve wave-condition forecasts', *Journal of Marine Systems*, 186, pp. 29–36. doi: 10.1016/J.JMARSYS.2018.05.006.

O'Donncha, F. *et al.* (2019) 'Ensemble model aggregation using a computationally lightweight machine-learning model to forecast ocean waves', *Journal of Marine Systems*, 199. doi: 10.1016/j.jmarsys.2019.103206.

O'Donncha, Fearghal *et al.* (2019) *Information Management System: First release. Deliverable 1.5. GAIN - Green Aquaculture INtensification in Europe*.

O'Donncha, F. *et al.* (2021) 'Data Driven Insight Into Fish Behaviour and Their Use for Precision Aquaculture', *Frontiers in Animal Science*, 2. doi: 10.3389/FANIM.2021.695054/PDF.

O'Donncha, F. and Akhriev, A. (2021) *Test of the final release of the aquaculture information management system. Milestone 13*.

O'Donncha, F. and Gormally, R. (2019) *Test of the first release of the aquaculture information management system. Milestone 6*.

O'Donncha, F. and Purcell, M. (2019) *Methodologies for big data mining in aquaculture. Deliverable 1.3. GAIN - Green Aquaculture INtensification in Europe.*

Oppedal, F. *et al.* (2010) 'Environmental drivers of Atlantic salmon behaviour in sea-cages: a review', *Elsevier*. doi: 10.1016/j.aquaculture.2010.11.020.

Rodríguez, P. and Wohlberg, B. (2013) 'Fast principal component pursuit via alternating minimization', in *2013 IEEE International Conference on Image Processing, ICIP 2013 - Proceedings*, pp. 69–73. doi: 10.1109/ICIP.2013.6738015.

Scherelis, C. *et al.* (2020) 'Investigating biophysical linkages at tidal energy candidate sites; A case study for combining environmental assessment and resource characterisation', *Renewable Energy*, 159, pp. 399–413. doi: 10.1016/j.renene.2020.05.109.

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Service, M. *et al.* (2019) *Report on Instrumentation of GAIN pilot sites. Deliverable 1.1. GAIN - Green Aquaculture INtensification in Europe*.

Simmonds, J. and MacLennan, D. (2008) *Fisheries acoustics: theory and practice*, *Fish and Fisheries*. Edited by John Wiley & Sons. Wiley. doi: 10.1111/j.1467-2979.2006.00220.x.

Wijaya, T., Sinn, M. and Chen, B. (2015) 'Forecasting uncertainty in electricity demand', in *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*. AAAI. Available at: https://www.aaai.org/ocs/index.php/WS/AAAIW15/paper/viewPaper/10104 (Accessed: 1 October 2021).

Wolff, S., O'Donncha, F. and Chen, B. (2020) 'Statistical and machine learning ensemble modelling to forecast sea surface temperature', *Journal of Marine Systems*, 208(103347). Available at: http://arxiv.org/abs/1909.08573 (Accessed: 28 April 2020).

Wolpert, D. and Evolutionary, W. M. (1997) 'No free lunch theorems for optimization', *IEEE transactions on evolutionary computation*, 1(1), pp. 67–82. Available at: https://ieeexplore.ieee.org/abstract/document/585893/ (Accessed: 1 October 2021).

Zhou, Z. *et al.* (2010) 'Stable principal component pursuit', in *IEEE international symposium on information theory*. IEEE. Available at: https://ieeexplore.ieee.org/abstract/document/5513535/ (Accessed: 30 September 2021).

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

List of Figures

Figure 1: Overview of the system presenting IoT data integration, connectors to external data, microservices and automated model training and scheduling, and user interaction with the system in terms of bespoke (species and farm specific) model development, and dissemination to end user. Interaction with the end user is outlined on the right-hand side and is supported by robust data science and enterprise tools. Python and R packages allow the user to interact using open-source (e.g. Jupyter Notebooks) or commercial tools (e.g. IBM Watson Studio, Google Collab). AutoAI provides a "no-code" approach to allow non-data scientists apply AI to their datasets. External nodes (AQUARADAR, AQUASENSE) can upload and download data to the cloud to enable different levels of connectivity between different services (i.e. one-way connection where data such as sensor or model forecasts are downloaded from the cloud or two-way connection where data is downloaded and model forecasts uploaded and stored in the service). 8

Figure 2: Schematic of an MLP machine learning network..... 13

Figure 3: Sample configuration file for the GAIN modelling service. The key factors the user needs to consider are the data we provide to the model (given sensor or data context information), imputer properties to handle missing or corrupted data, and selection of algorithm (or use the default choice of random forest). 14

Figure 4: Screenshot from beta GAIN GUI that allowed users to visualise forecast results against observations. The panel on left hand side allows users to select a particular instance (or farm site) and period, while the upper panel provides dropdown list of entity (sensor) and signal (variable). The panel on the bottom left primarily relates to model management and can be considered a tool of the data scientist rather than the end user. This presents information on the different models that are trained (for this example it reports different machine learning models including Generalised Additive Model (GAM), XGBoost, and Random Forest), as well as different versions of those models to allow for iterative improvement and fine tuning of those models. It allows the user to easily select different model algorithms, as well as different implementations and versions of these models. This combination of complexity (many different models with different configurations) and simplicity (an easy comparison and analysis of which models performs well) provided us with a valuable tool for rapid and iterative application of machine learning to aquaculture. 15

Figure 5: Comparison of observed temperature against modelled values from the UNIVE trout farm. Blue curve denotes sensor observations while red indicate model estimate 17

Figure 6: Distribution of fish depth data for the NOR (left) and CAN (right) farm over the duration of the study period. The data is split into daytime and nighttime periods to explore how behaviours vary between those periods. The dashed vertical lines denote the mean for both periods 18

Figure 7: Visualisation of AutoAI results interrogating fish response to environmental conditions at the Norway site. The relationship map illustrates the different "pipelines" (data transformation and machine learning algorithms) that were explored while the Pipeline leaderboard displays the best performing model. Results comparable to what can be achieved by a skilled data scientist can be achieved without need for any code. 19

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330

Figure 8: Scatter plot of model predicted fish depth plotted against observed values for the CAN site. Inputs to the model are environmental data time-aligned with the target data, and hour of day to represent temporal variations 20

Figure 9: Feature importance reported at the (a) Norway, (b) Scotland, and (c) Canada sites. The y-axis reports the ranked list of features that contributed the most to variation in fish depth measurements, while the x-axis presents relative magnitude of those contributions. Ranking predictors in this manner can quickly help sift through large datasets and understand data trends (Kuhn and Johnson, 2013) 21

Figure 10: Feature importance for the occurrence of toxins at the SGM shellfish site in Portugal 23

The project has received funding from the European Union's Horizon 2020 Framework Research and Innovation Programme under GA n. 773330