Anomaly Detection in Agri Warehouse Construction*

Andrew McCarren Suzanne McCarthy Conor O Sullivan Mark Roantree
Insight Centre for Data Analytics, School of Computing, Dublin City University
Glasnevin Campus, Dublin 9, IRELAND
amccarren, smccarthy, cosullivan, mark {@computing.dcu.ie}

ABSTRACT

As with many sectors, strategists and decision makers in the agricultural sector have a requirement to predict key measures such as product and feed pricing in order to maintain their position and, in some cases, to survive in their industry. Predictive algorithms in the area of *Agri Analytics* have shown to be very difficult due to the wide range of parameters and often unpredictable nature of some of these variables. Improving the predictive capability of Agri planners requires access to up-to-date external information in addition to the analyses provided by their own in-house databases. This motivates the need for an Agri Data Warehouse together with appropriate cleaning and transformation processes. However, with the availability of rich and wide ranging sources of Agri data now available online, there is a strong motivation to process as much current, online information as possible. In this work, we introduce the Agri Data Warehouse built for the DATAS project which not only harvests from a large number of online sources but also adopts an anomaly detection and labelling process to assist transformation and loading into the warehouse.

CCS Concepts

•Information systems → Data warehouses; Data analytics; Data mining; •Computing methodologies → Anomaly detection; •Applied computing → Agriculture;

Keywords

Data Warehouse, Agri, Data Mining, Anomaly Detection

1. INTRODUCTION

In Ireland, the Agricultural (Agri) sector incorporates large-scale multi-national businesses such as Kerry foods [14] and the Kepak Group [15], through to SME's including bespoke food manufacturers and farmer producers. These companies embrace technology and innovation as a part of their daily lives. The world agriculture market for value added products is in excess of $3.2 Trillion [25]. Figure 1 shows the value added to the GDP of the US by the agricultural industry, as compared with other industries.

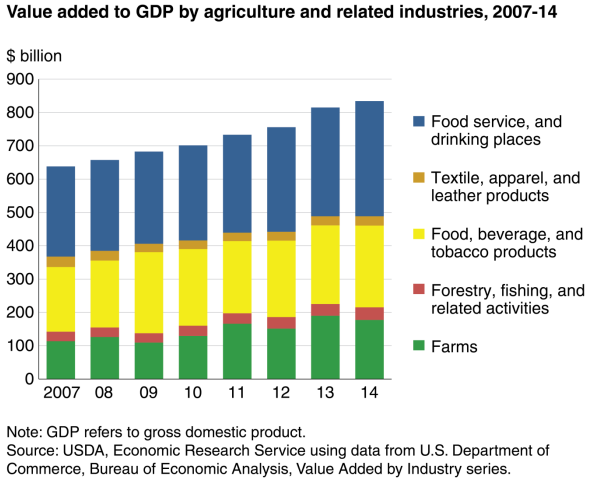

Figure 1: US Agriculture

The challenge is that the sector is currently market-driven rather than taking a leading role in determining its future outcomes. In brief, rather than the business reacting to what is occurring in international markets, it must possess the ability to analytically predict what will occur in their specific sector. The DATAS Project (Deductive Analytics for Tomorrow's Agri Sector) is designed to address these issues. The major goal of the DATAS project is to create data mining algorithms to allow the non-statistician to produce scientifically sound predictions from carefully validated and integrated data sets.

The focus is on obtaining predictive outcomes for an Agri-sector which currently lacks real analytic or predictive powers. Obtaining sound predictions about commodity prices such as pig or beef will offer Agri decision makers a distinct advantage when dealing with customers and suppliers. Additionally, the solution will provide the Agri user or Agri commodity buyers and sellers with a significant level of sophistication not currently seen in current market intelligence software.

*Funded by Enterprise Ireland Grant Ref. CF-2014-4611

ACSW '17, January 31-February 03, 2017, Geelong, Australia

DOI: http://dx.doi.org/10.1145/3014812.3014829

1.1 Problem and Motivation

The success of the DATAS project depends on access to as wide a range of agricultural data as possible, cleaning, transforming and integrating this data, before the development of any data mining functions. Traditionally, data analysts in the Agri sector select from a wide number of commercial or free to use databases which cover a wide number of agricultural topics of interest. The Food and Agricultural Organisation (FAO) of the United Nations [11] provides access to large volumes of information that can be used in many aspects of agricultural planning and prediction exercises. The challenge faced in this research is to obtain quicker access to these databases, and to identify where conflicting information regarding the same events are provided by different organisations. In both cases, this greatly affects the quality of any predictions that are made. A recent study [6] looked at using online prices to create their own database (or index) for analysis and research purposes. This research highlighted some interesting advantages in using online data in this manner. The obvious advantage is the low cost per data element (or observation). There is a cost to harvesting data from websites but this is far cheaper than visiting factories or paying for commercial databases. Further advantages are the speed of access to frequently changing data and the high volumes which make it easier to detect errors in the data.

This motivates our approach to the Agri data problem where we aim to extract data from a wide range of websites and build a data warehouse which supports OnLine Analytical Processing (OLAP) and the generation of datasets for data mining. Our goals are to design a multi-dimensional data model to represent data from heterogeneous online sources but also including company databases and purchased datasets. DATAS will use these sources as indicator variables for the client's efficacy variable. The project aims to incorporate all the relevant data (CSO, climate data, EUROSTAT (European statistical service), Internal Client data (Historical sales, costs, yields etc.), Social Media etc.) into a single large digital repository to meet the requirements of the Agri sector on a sub-sector by sub-sector basis.

1.2 Contribution

The Billion Prices Project at MIT [6] provides evidence that the use of online data in areas such as international economics provides new levels of information and, as a result, the potential for superior levels of prediction. In their research, online prices were used to build daily prices indices across multiple countries. However, these researchers are importing data into a database, presumably using a relational schema, and are not using data warehousing concepts and technologies. In fact, there is no significant research that we can find where the development of an Agri data warehouse is taking place. There are also no existing research projects that we are aware of that generate an Agri warehouse of this scale or that aim to detect anomalies at this speed. Although anomaly detection in isolation is not new, in this context it has novelty. Our Extract-Transform-Load system loads data and detects anomalies on an almost continuous basis, as opposed to bulk updates which are resource-heavy and time-consuming. Our conformed designed architecture [16] greatly supports the analysis across the variety of data marts that arise in Agri warehouse building. As part of the warehouse construction, we developed an outlier analysis function to provide an automated cleaning process for the online data. Finally, our work has lead to the development of a multi-dimensional canonical Agri data model to which all Agri data sources are mapped.

1.3 Paper Structure

This paper is structured as follows: in §2, we provide an overview of related research; in §3, we describe our design and construction of the Agri data warehouse; in §4, we provide a description of the warehouse cleaning process using anomaly detection; in §5, we provide an evaluation of how the cleaning process worked; and finally in §6, we provide some conclusions.

2. RELATED RESEARCH

The survey presented in [20] motivates the usage of hybrid warehouses where online data (mainly XML at the time the research was undertaken) is integrated into data warehouse systems. The first category of warehouse examined was that using XML at its core. Here, the XML model was used to represent multidimensional data and metadata. This approach is prone to the performance issues and complexities of XML and requires the construction of complex gateway components to company databases. The second category of warehouse adopts the hybrid approach where traditional data types (mostly relational) sit beside the XML formatted data obtained from the web. This approach will also require the development of forward and reverse gateway technologies [4] for queries and warehouse updates involving both sets of data types. The final category of warehouses is a hybrid approach incorporating document centric XML sources. Examples of this type include [22] where the authors build materialised XML queries of shared data not dissimilar to the conformation of data marts in the multidimensional agri model presented here. However, as our Agri data sources do not involve the level of complexity shown in the XML queries in [22], we adopt the approach of conformed dimensions for sharing and interoperability.

The research in [6] is very similar to the work undertaken in our research. However, despite the fact that their work is close to a Big Data problem (hundreds of websites and thousands of web pages), with high volumes of data used to generate the price indices, there is no discussion on the important aspects of warehouse construction. Instead, data extracted from the web *is fitted to a common database schema* after cleaning. Thus, while both projects have similar goals in harvesting online data for price indexing and predictions, our research is underpinned by the construction of a data warehouse and all of the necessary components involved in data extraction, transformation and loading (incorporating updates) into the data warehouse.

What further differentiates our work is the incorporation of anomaly detection in data cleaning and classification. A detailed survey of numerous anomaly detection techniques was undertaken in [8]. They describe various methods ranging from basic histogram analysis to complex MLP. However, the research in [8] does not address the issue of dealing with datasets that are constantly updated and require an immediate response to the researchers question of whether the time point in question is an anomaly. The nature of the online data available to an Agri data warehouse is predominantly of a time series nature. Agricultural prices for periods of time (windows) for example, have tended to fol-

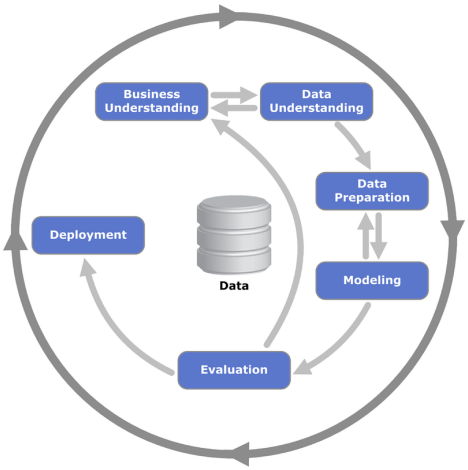

Figure 2: CRISP-DM process model

low a stationary process but on occasion illustrate shifts or events for some time interval before normal cycles return. Typical methods to identifying these shifts, such as wavelet transforms, require the event or shift to have occurred *before* it can be detected. Our anomaly detection algorithms were designed to quickly identify when an event (shift) is occurring.

3. WAREHOUSE DESIGN

This section provides a high level overview of the DATAS architecture, which employs the well-established Extract-Transform-Load (ETL) method [16] to prepare data for the DATAS warehouse.

In both the design of the warehouse and the application of the ETL process, we adhered to the Cross Industry Standard Practice for Data Mining (CRISP-DM) model [7], which provides a guideline for the use of Data Mining to solve business problems. In particular, the Business Understanding stage of the model (see Figure 2) requires an in-depth knowledge of the industry at hand and an understanding of the business requirements in order to design the decision model used to prepare the data for loading to the warehouse. The CRISP-DM model was conceived in 1996 to fill the need for a process in the growing area of data mining that create some standardization for organisations to launch their data mining projects and glean the best results possible. Today, it is still the leading data mining standard.

3.1 Data Extraction

Source data comes into the system in 2 forms: structured data such as company databases or purchased datasets and semi-structured data sources as online data. These are the operational systems of record that capture the transactions of the business. As company databases are highly structured, the transformation process is well understood for most databases and schemas. For semi-structured (online) data, technologies including Selenium [23] and BeautifulSoup [2] are used to build the *wrapper* interfaces to the websites. In general, source systems are not suited to data mining or predictive analytics as they maintain little historical data, whereas the DATAS system requires historical data to provide a high degree of accuracy. These characteristics of original source data have driven our design goals for the architecture described here.

In total, the websites of 27 organisations are crawled at different but regular intervals. As it is necessary to extract multiple web pages, each dataset requires an individual wrapper (web bot). For example, from *DairyAustralia*, we harvest *dairy trade*, *production* and *sales data*. For an understanding of the types of datasets imported into DATAS, we provide a brief overview of a selection of these data sources.

- **Dairy Australia**. Dairy Australia [9] is the national services body of the Australian dairy industry. Their key activities include strategic investment guidance and research and development, funded by a combination of levy, government and leveraged funds. They provide monthly production and sales data for several dairy products in addition to year end trade data which is later transformed to calculate the monthly figures. The reasonably tabular data (see Figure 3) is loaded directly into the data staging area of DATAS.

- **Eurostat**. Eurostat [10] is the main database of the European Commission and provides DATAS with *production* and *trade* data. However, as it represents the entire European Union, each dataset requires at least one wrapper. For example, the production data for all dairy products obtained from cow's milk delivered to dairies has its own dataset with proprietary structure. Conversely, trade data is generated by manually created preloaded queries that is auto-updated (with date details) before each run. In other words, 2 separate processors are required: one to generate the updated query, and a second process to extract the trade data.

- **CLAL**. CLAL is an Italian Dairy Economic Consulting firm that analyses the dairy market, and provides a digital dashboard style of data, analysis and news [5]. DATAS uses the CLAL resources to extract market analysis (primarily production, stocks and price data) for dairy products in Europe, US, Oceania and others. Unlike some data source websites, CLAL does not provide files for download. Instead, it is necessary to harvest data directly from HTML (see Figure 4).

- **IMF**. The International Monetary Fund [13] was founded in 1944 to promote global economic stability and cooperation. It defined the unit of account SDR (Special Drawing Rights) and publishes exchange rates for currencies worldwide based on this unit. DATAS extracts a complete set of all of these exchange rates daily.

3.2 Data Staging

The data staging area of the data warehouse is both a storage area and a set of procedures that form the extract-transformation-load (ETL) method [16]. The data staging area houses CSV, HTML, XML and JSON files including exported CSV files from company datasets. In DATAS, MySQL is used for this staging area and at this point, algorithms for cleaning and transformation (see V_1 and V_2 in Figure 5) have been developed and are discussed later in section 4. Data is initially extracted from sources with both the original and CSV files stored, together with the transformation algorithm. This provides a means of auditing every transformation as we attempt to build a gold standard dataset in the warehouse.

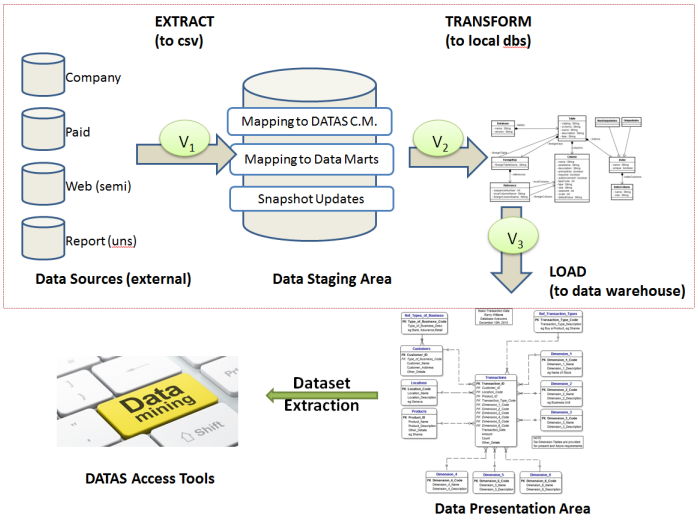

Figure 5: Warehouse Architecture

Table 1: Descriptive Statistics

Correctness	Completeness
No. of Records	Missing Rows (count)
No. of Columns (Cols)	Missing Cols (count)
Total for all Numeric cols	Missing Rows (%)
Average for all Numeric cols	Missing Cols (%)
Std. Dev. for all Numeric cols	List of urls
Skew-ness for all numeric cols	
Kurtosis for all numeric cols	

3.3 Transformation and Validation

Once the data is extracted to the staging area, there are a number of transformations, such as cleaning the data (resolving data conflicts, dealing with missing elements, or conversion into standard formats), combining data from multiple sources and assigning warehouse keys. These transformations are all precursors to loading the data into the data warehouse presentation area. For example, there are three *lookup* tables when an additional stage of transformation is required for before some Dimensions can be updated: *lookup_currency*, *lookup_geo* and *lookup_trade_product*. In this case, *lookup_geo* addresses the issue of having many possible spellings for some countries, including spelling errors. These are mapped to each country's formal name and the correct spelling of the most commonly used name for the country, eg. Irland == Ireland == Republic of Ireland.

Two types of data statistics are recorded for all data imports: descriptive and index statistical data.

Descriptive statistics provide the validation process with a handle on the shape and underlying pattern that exists within each dataset. Typical measurements of variables scraped on a particular site would be the mean, variance, skewness and kurtosis of both the actual value and the *difference* from the previous value. Additionally, we record data on the number of missing values within a particular variable for each dataset and these values are recorded in a descriptive statistical database for each data import.

Index statistics generate an index for both the original web data and the transformed (CSV) data. This records the actual positions of each data point on a particular website. The original web formatted text extracted is then analysed to validate the data transformed into the CSV format. The data in original web formatted text is flattened in order for the positional data to be compared with the CSV dataset. Finally, a frequency word count of the web text is recorded.

Using the data collected in the *descriptive* and *Index* statistical database, we compare each scrape with the **Golden Dataset** statistical database which was manually verified. Initial methods that will be applied to freshly scraped data will range from individual t-tests for continuous variable and contingency tests for frequency data between the Golden Data and the freshly scraped data. Outliers must be highlighted to the data extraction process as there may be erroneous values provided by any website host. These are discussed in detail in section 4

3.4 Data Loading

The schema for the DATAS warehouse was designed in an organic fashion by generating data marts from each online data source. By agreeing a fixed set of Dimensions in advance using company databases, we adopted a conformed approach to data mart design to facilitate future integration across data marts (web sources). Adding or extending dimensions is managed in a tightly controlled process. In effect, the integration of these data marts provided us with our multi-dimensional Agri data model. While this is too large to describe in this paper, the *Fact_Price_Monthly* data mart is illustrated in Figure 6 with 10 dimensions and the structure for 5 data marts is provided in Table 2.

At this point, data is integrated from multiple sources and transformed to the Warehouse Model format, before being loaded into the Data Warehouse. A third validation process, V3, is used to verify that this transformation takes place accurately and without loss of information. As there are many mining and prediction algorithms inside the DATAS service, different datasets with a range of different formats are required by the analysis layer. The Dataset Creation process allows for the formation of the heterogeneous datasets required by each algorithm.

3.4.1 Warehouse Statistics

At present, there are 24 Dimensions and 26 Fact tables currently in the database. As the design approach was based on conformed dimensions, these are shared by all fact tables. As the structure of the warehouse is too large to describe here, we provide a subset of both facts and dimensions but also show that those dimensions are shared across multiple facts.

In Figure 6, the *Fact_Price_Monthly* data mart is shown to have a single measure *price* and foreign keys to ten dimensions. This facilitates the generation of datasets with a potentially high level of dimensionality and as can be seen from dimensions such as *dim_geo* in Figure 6, within dimensions we can have a degree of granularity which, on one hand, adds to the power of query expression but also to the complexity in managing the performance of these query expressions.

4. OUTLIER DETECTION FOR DATA ENHANCEMENT

Extracting data from web sources leads to many potential hazards with regard to ensuring that valid data is being

Fact	Dimensions	Grain
fact_cold_storage_monthly	dim_date_monthly	Month
	dim_geo	
	dim_product	
	dim_unit	
	dim_source	
	dim_outlier	
fact_consumption_annual	dim_date_annual	Year
	dim_geo	
	dim_product	
	dim_unit	
	dim_source	
	dim_outlier	
fact_demographic_ population	dim_date_annual	Year
	dim_geo	
	dim_gender	
	dim_age_group	
	dim_status	
	dim_population_economics_type	
	dim_unit, dim_source	
	dim_outlier	
fact_economic_forecast	dim_date_annual	Year
	dim_geo	
	dim_product	
	dim_measurement_feature	
	dim_unit	
	dim_source	
	dim_outlier	
fact_price_monthly	dim_date_monthly	Month
	dim_geo	
	dim_product	
	dim_price_type	
	dim_currency_monthly	
	dim_status	
	dim_grade	
	dim_unit	
	dim_source	
	dim_outlier	

Table 2: Selection of Marts in DATAS

retrieved for potential forecasting models. Errors can be induced from a number of sources such as wrong data being entered by the host supplier, changes in units by the host supplier, positioning of the data on the web service and errors induced by the web scraping software. In order to ensure the data collected for the initial scrape from each website is correct, quality control measures were put in place. This section provides further detail on the validation procedures that are used to protect the integrity of the DATAS warehouse. The data extracted from websites consisted of commodity prices over time, along with other variables that can potentially influence the market (e.g. weather, production, trade etc.). At this point, three variations of the anomaly detection algorithm are used to optimise the process. In this section, we present each variation of the algorithm together with a brief description.

Algorithm 1 Baseline test: Univariate Anomaly Detection

function DETECT OUTLIERS(X)
  $X \leftarrow sorted(X)$ ▷ Data is sorted chronologically
  $\mu \leftarrow mean(X)$ ▷ Mean of data is calculated
  $\sigma \leftarrow std\_dev(X)$ ▷ Standard deviation of data is calculated
  $UpperBound \leftarrow \mu + (Z \times \sigma)$
  $LowerBound \leftarrow \mu - (Z \times \sigma)$
  for X_t in X where $t \in 1...T$ **do**
    if $UpperBound < X_t$ OR $LowerBound > X_t$ **then**
      Label X_t as outlier
    else
      Label X_t as standard
    end if
  end for
end function

4.1 Univariate Anomaly Detection

For our initial attempt at anomaly detection, we deliberately implemented a simple algorithm which simply detected and labelled outliers within the data. Algorithm 1 shows the initial univariate method for detecting outliers. This may then be considered a baseline test upon which we set out to improve. The data is extracted from each table and broken up into a subset X e.g. French pig prices. The data subset X is then sorted chronologically. The mean μ and standard deviation σ of X are then calculated. μ and σ are used to calculate the upper and lower bounds along with n where $n \in \mathbb{Z}$ and n is the set of all positive integers. For this initial experiment, n is set to 3. By iterating through all values in X, any values that were above $UpperBound$ or below $LowerBound$ were labelled as an outlier. In section 5, we discuss its usage and detection accuracy.

4.2 Anomalies for Non-stationary Data

As will be discussed in section 5, Algorithm 1 failed to take non-stationary data into account. In order make the data appear stationary, Algorithm 2 calculates the difference of each value *and* its predecessor. The dataset subset X is sorted chronologically as in the previous algorithm. The difference $X_t - X_{t-1}$ is calculated for every value of X and labelled $DiffX$ in the algorithm. μ, σ, $UpperBound$ and $LowerBound$ are then calculated for the $DiffX$ dataset. Again, by iterating through all values in $DiffX$, any values

Algorithm 2 Univariate Anomaly Detection *v2*

function DETECT OUTLIERS(X)
  $X \leftarrow sorted(X)$
  $DiffX \leftarrow diff(X)$ ▷ for every value in X, $X_t - X_{t-1}$ is calculated
  $\mu \leftarrow mean(DiffX)$
  $\sigma \leftarrow std\_dev(DiffX)$
  $UpperBound \leftarrow \mu + (n \times \sigma)$
  $LowerBound \leftarrow \mu - (n \times \sigma)$
  for X_t in X where $t \in 1...T$ **do**
    if $UpperBound < DiffX_t$ OR $LowerBound > DiffX_t$ **then**
      Label X_t as outlier
    else
      Label X_t as standard
    end if
  end for
end function

that were above $UpperBound$ or below $LowerBound$ were labelled as an outlier.

Algorithm 3 Univariate Anomaly Detection *v3*

function DETECT OUTLIERS(X)
  $X \leftarrow sorted(X)$
  $DiffX \leftarrow diff(X)$ ▷ for every value in X, $X_t - X_{t-1}$ is calculated
  $\mu \leftarrow mean(DiffX)$
  $\sigma \leftarrow std\_dev(DiffX)$
  $UpperBound \leftarrow \mu + (n \times \sigma)$
  $LowerBound \leftarrow \mu - (n \times \sigma)$
  for X_t in X where $t \in 1...T$ **do**
    if $UpperBound < DiffX_t$ OR $LowerBound > DiffX_t$ **then**
      if $UpperBound < DiffX_t$ AND $LowerBound > (X_{t+1} - X_t)$ **then**
        Label X_t as mistake
      else if $LowerBound > DiffX_t$ AND $UpperBound < (X_{t+1} - X_t)$ **then**
        Label X_t as mistake
      else
        Label X_t as event
      end if
    else
      Label X_t as standard
    end if
  end for
end function

4.3 Anomaly Classification

What became clear during our research was the varying nature of different anomalies. Indeed, some of these outliers are likely to be events of interest to data miners. For this reason, we decided to classify the *type* of outliers that occurred in order to identify mistakes and outliers from what appeared to be an indicator of some change to the pattern in the data.

As a result, Algorithm 3 is quite similar to Algorithm 2 except for the classification aspects. Here, the algorithm checks the values of $X_{t+1} - X_t$ also.

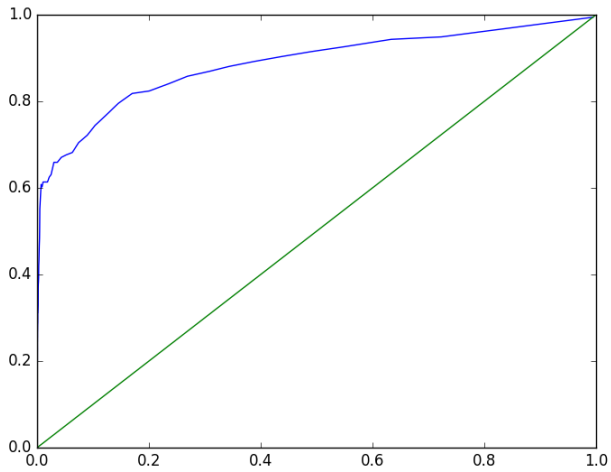

Figure 7: ROC curve for BordBia pig price data

- If $UpperBound$ is less than $DiffX_t$ and $LowerBound$ is greater than X_{t+1}, X_t is classified as an *error* or *outlier*.
- Similarly, if $LowerBound$ is greater than $DiffX_t$ and $UpperBound$ is less than $X_{t+1} - X_t$, X_t is again classified an *error* or *outlier*.
- Otherwise, the remaining anomalies detected are classified as *events*.

5. EVALUATION

In this section, we describe the results of outlier detection in the cleaning and marking up of data. As described in the previous section, our approach is the continued customisation of the outlier detection algorithm based on the results from previous iterations. This differs from our approach in [19] where we employed different detection algorithms in sensor generated datasets. Additionally, with the sensor generated datasets in [19], there is persistent non-stationary in the data as opposed to the Agri data which, when it shifts, it is usually from a stationary process. In order words, shifts can be identified as sudden or abrupt movements rather that a continuous upward or downward movement. This difference is crucial as here, the main issue with online Agri data is our lack of control over how data is generated or published. Our evaluation uses two datasets which offer suitable levels of heterogeneity for the purpose of testing the accuracy of our algorithms.

- Bordbia (www.bordbia.ie). This dataset delivers the price of pig meat in EU countries over time, recorded at weekly intervals. The dataset has 23,499 instances, covering a total of 21 countries and regions.
- Dairy Board (AHDB http://pork.ahdb.org.uk/). This contains the slaughter count of pigs (per head) for specified EU countries, recorded at weekly intervals. The dataset contains 2,830 instances across 6 countries.

5.1 Evaluation Process

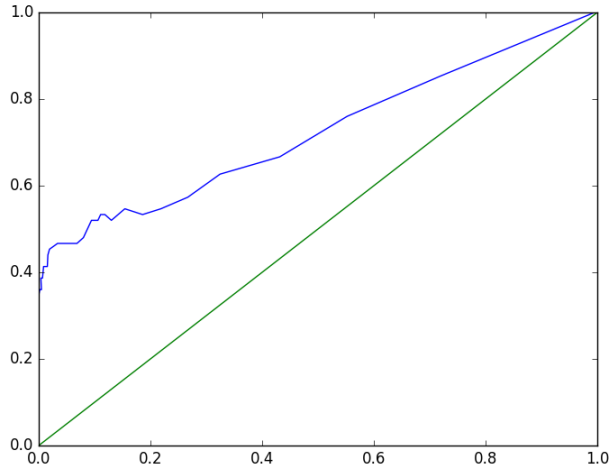

Figure 8: ROC curve for AHDB

Validation of our algorithm was performed by a collaborator in the Agri sector, henceforth called the Independent Manual Outlier Verification (IMOV) process. The outlier detection algorithm was run on both Bordbia and Dairy Board datasets. Results from algorithm 1 were not at expected levels are are not discussed here. The modification which resulted in algorithm 2 are discussed in section 5.2 and algorithm 3 which introduced a classification system is discussed in section 5.3. The evaluation process was carried out as follows:

- The results of the algorithm were extracted to CSV files from the database.
- The IMOV viewed graphical visualisations of these datasets and flagged the datapoints which were deemed anomalous.
- Two Research Assistants transferred the highlighted graph outliers to CSV files, as raw data. These transformed files contained a column for the time variable (date or year and month) and three columns for each country in the dataset: the data value, the status given by the algorithm (standard or anomaly) and a column for outlier flags from the IMOV.
- A number of analyses were conducted:
  1) A Receiver Operating Characteristic (ROC) curve and corresponding Area Under the Curve (AUC) [18].
  2) A Confusion Matrix for each dataset for the proposed discrimination threshold.
  3) Logistic Regression[17] is used to understand the relationship between the IMOV and the algorithm. By treating the IMOV as the *dependent variable* and the algorithm as the *independent variable*, we can determine the appropriate discrimination threshold for the algorithm by taking the threshold that gives the minimum p value for the varying thresholds. The GLM package (generalised linear model) which is part of the R library was used for the logistic regression.

The ROC curve is a graphical plot that illustrates performance of a binary classifier system [3] by plotting the *True Positive Rate* against the *False Positive Rate* as its discrimination threshold is varied between upper and lower bounds. In this way, it helps the researcher to determine an appropriate threshold for the algorithm. The corresponding Confusion Matrix and the Logistic Regression will help demonstrate that the algorithm does in fact emulate the IMOV process. The AHDB and BordBia datasets were used to demonstrate the ability of the algorithm to perform as well as the IMOV when an appropriate discrimination threshold is chosen, based on the results of the ROC curve. When choosing a threshold, there is inevitably a certain trade-off between *sensitivity* and *specificity*. The aim is to minimise the need for further manual checking by creating an algorithm that will reliably flag all obvious true anomalies (sensitivity) while ignoring trivial variations and therefore, producing the most accurate set of anomalies.

Table 3: BordBia Confusion Matrix: threshold = 3

		Algorithm		
		Anomaly	Standard	Total
IMOV	Anomaly	107	69	176
	Standard	124	17223	17347
	Total	231	17292	17523

5.2 Results of Basic Anomaly Detection

Figure 7 shows the ROC curve for the Bordbia dataset. The diagonal represents the *line of no discrimination*, or the theoretical results of random guessing, such as flipping a coin. As can be seen, the curve shows a considerable improvement on this line. The area under the curve (AUC) for this dataset is generated by our Python software which builds the graph and here, it is calculated at 88.13%. Figure 8 shows the ROC curve for the AHDB dataset where the AUC was calculated at 71.7%.

Although the ROC and AUC results for these datasets were in the range of *'Very good'* in the case of Bordbia and *'Fair'* for AHDB [1], the problem remained of how to choose a specific threshold to apply to the algorithm. It was decided that the significance of the independent variable (algorithm) *vs* the dependent variable (IMOV), as shown by a Logistic Regression, would be the deciding factor in terms of choosing thresholds T_B (number of standard deviations above the mean) and T_A (number of standard deviations below the mean). The Logistic Regression analysis was run on both datasets multiple times while varying T_B and T_A in each case. The significance of the results of this analysis allowed us to accurately determine the optimum threshold. The results of the Confusion Matrix and Logistic Regression for the optimum threshold can be seen for each dataset in tables 3 and 4.

At a highly significant level ($p < 0.05$), a Wald test[24] with a test-statistic value $Z > 2$ is considered a strong result. We can use the test-statistic and the significance level to determine the optimum number of standard deviations from the mean to use as a threshold for the algorithm. For Bordbia, the Logistic Regression Analysis showed a highly significant ($p < 0.001$) result with a test-statistic of $Z = 30.05$. For AHDB, Logistic Regression Analysis also showed a highly significant ($p < 0.001$) result with a test-statistic $Z = 11.5$

For both datasets, the algorithm performed well at a very

Table 4: AHDB Confusion Matrix: threshold = 1.9

		Algorithm		
		Anomaly	Standard	Total
IMOV	Anomaly	34	41	75
	Standard	20	1018	1038
	Total	54	1059	1113

Table 5: AHDB Contingency table classifications

Count Total %	event	mistake	standard	
event	23 2.01%	0 0.00%	0 0.00%	23 2.01%
mistake	0 0.00%	3 0.26%	0 0.00%	3 0.26%
standard	1 0.09%	48 4.20%	1069 93.44%	1118 97.73%
	24 2.10%	51 4.46%	1069 93.44%	1144

significant ($p < 0.001$) level, indicating that the algorithm reliably predicted the classifications of the IMOV. Similarly, for both datasets, the ROC and AUC showed that the algorithm is highly in agreement with the IMOV for identifying outliers. For the AUC, the higher the statistic is above 50%, the stronger the model being tested. A perfect model would yield an AUC of 100%.

5.3 Results of Classification Process

In order to analyse the results of the outlier detection algorithm with the added feature of the classification of anomalies into *events* and *errors* (algorithm 3), a Contingency Table was produced and a Chi-Square completed. Table 5 shows the agreement between IMOV and the algorithm in counts and percentages of each level of classification for AHDB, with Table 6 showing the results for the Bordbia dataset.

A Chi-Square analysis showed a strong, positive relationship between the IMOV's classifications and those of the algorithm for AHDB and Bord Bia with values of $\chi^2(1, N = 1144) = 1159.76$, $p < 0.001$ and $\chi^2(1, N = 19057) = 10925.3$, $p < 0.001$, respectively.

It must be noted that, due to the high number of empty cells in the above tables (cases of no agreement), a warning was generated that the results of the Chi-Square may be unstable. Despite this, the results show a strong relationship between the classification of outliers by the IMOV and the classification by the algorithm.

Table 6: BordBia Contingency table classifications

Count Total %	event	mistake	standard	
event	86 0.45%	0 0.00%	109 0.57%	195 1.02%
mistake	0 0.00%	21 0.11%	15 0.08%	36 0.19%
standard	0 0.00%	69 0.36%	18757 98.43%	18826 98.79%
	86 0.45%	90 0.47%	18881 99.08%	19057

6. CONCLUSIONS

In industries such as agriculture, the usage of online data offers significant advantages, due mainly to the high volumes and wide varying nature of the data and the fact that many of these streams are continually updated. In this paper, we describe our research into building an Agri data warehouse, which primarily takes its updates from the web pages and file exports from 27 different Agri organisations. These imports go through a controlled Extract-Transform-Load process as they are mapped to our conceptual Agri data model, a multi-dimensional model suited to OLAP queries for extracting datasets across a number of dimensions and at varying levels of granularity. We also presented one element of our automated cleaning process where we employ different forms of anomaly detection to label outlier data during both the Extraction and Transformation processes. Our current research follows two separate streams. On one hand, the high volumes of data now in the warehouse has led to the construction of an efficient lattice to keep the extraction of datasets for mining and prediction as fast as possible, but the high levels of dimensionality for some of our data marts mean that this work is ongoing. However, our algorithms have optimised the process of detecting anomalies during the process of loading the data on a close to continuous basis as opposed to bulk updates after the fact. Separately, we are now focusing on the main goal of the DATAS project which is to provide predictive capabilities for Agri decision makers.

7. REFERENCES

[1] AUC. 2016. http://gim.unmc.edu/dxtests/roc3.htm.

[2] Beautiful Soup Documentation. 2016. https://www.crummy.com/software/BeautifulSoup/bs4/doc/.

[3] Cox D.R. 1958. The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society.* Series B (Methodological), 20,2, 215-242.

[4] Stonebraker M. and Brodie M. 1995. *Migrating Legacy Systems: Gateways, Interfaces and the Incremental Approach.* Morgan Kaufmann.

[5] CLAL. 2016. http://www.clal.it/en/.

[6] Cavallo A. and Rigobon R. 2016. The Billion Prices Project: Using Online Prices for Measurement and Research. *Journal of Economic Perspectives.* 30, 2, 151-178.

[7] Shearer C. 2000. The CRISP-DM Model:The New Blueprint for Data Mining. *Journal of Data Warehousing.* 5,4,13-22.

[8] Chandola V., Banerjee A. and Kumar V. 2009. Anomaly Detection: A Survey. *ACM Computing Surveys.* 41,3.

[9] Dairy Australia Publications. 2016. http://www.dairyaustralia.com.au/Industry-information/About-Dairy-Australia/Publications-2.aspx.

[10] Eurostat: Your key to European statistics. 2016. http://ec.europa.eu/eurostat/about/overview.

[11] FAO. 2016. http://www.fao.org/statistics/databases/en/.

[12] Golub et al. 2013. Global climate policy impacts on livestock, land use, livelihoods, and food security. *Proceedings of the National Academy of Sciences of the United States of America.* 110, 52, 20894-20899.

[13] International Monetary Fund. 2016. http://www.imf.org/external/index.htm.

[14] Kerry Group. 2016. http://www.kerrygroup.com/our-company/history/.

[15] Kepak. 2016. http://www.kepak.com/about-us/.

[16] Kimball R. and Ross M. 2002. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling (2nd ed.)*. Wiley.

[17] James G., Witten D., Hastie T. and Tibshirani R. 2013. *An Introduction to Statistical Learning*. Springer.

[18] Swets J. 1996. *Signal Detection Theory and Roc Analysis in Psychology and Diagnostics: Collected Papers*. Lawrence Erlbaum Associates.

[19] Donoghue J., Roantree M., Cullen B., Moyna N., O Sullivan C., and McCarren A. 2015. Anomaly and Event Detection for Unsupervised Athlete Performance Data. *Proceedings of LWZ 2015, CEUR Workshop Proceedings*. 1458, 205-217.

[20] Perez J., Berlanga R., Aramburu J. and Pedersen T. 2008. Integrating Data Warehouses with Web Data: A Survey. *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*. 20,7.

[21] Perry B., Graceb D. and Sonesc K. 2013. Current drivers and future directions of global livestock disease dynamics. *Proceedings of the National Academy of Sciences of the United States of America*. 110,52, 20871-20877.

[22] Roantree M. and Liu J. 2014. A heuristic approach to selecting views for materialization. *Software - Practice and Experience*. Vol. 44:10, 1157-1179.

[23] SeleniumHQ Browser Automation. 2016. http://www.seleniumhq.org/docs/.

[24] Harrell, Frank E., Jr. 2001. *Regression modeling strategies*. 9,3. New York: Springer-Verlag.

[25] United States Department of Agriculture. 2016. http://www.ers.usda.gov/data-products/chart-gallery/detail.aspx?chartId=40037.

Manufactured dairy products production report 15/16 by Product (tonnes)

Dairy Australia

		Butter			Butteroil			SMP			BMP			Cheese*			WMP			Whey Powder		
		14/15	15/16	Var%	14/15	15/16	Var%	14/15	15/16	Var%	14/15	15/16	Var%	14/15	15/16	Var%	14/15	15/16	Var%	14/15	15/16	Var%
July		5,389	5,130	-4.8%	946	715	-24.4%	16,157	19,671	21.7%	795	653	-17.9%	17,487	19,042	8.9%	7,996	8,877	11.0%	2,684	2,605	-2.9%
	YTD	5,389	5,130	-4.8%	946	715	-24.4%	16,157	19,671	21.7%	795	653	-17.9%	17,487	19,042	8.9%	7,996	8,877	11.0%	2,684	2,605	-2.9%
August		6,136	6,593	7.5%	828	1,086	31.3%	18,548	22,771	22.8%	780	998	27.9%	23,989	26,195	9.2%	5,331	4,131	-22.5%	4,039	3,638	-9.9%
	YTD	11,525	11,724	1.7%	1,774	1,801	1.6%	34,704	42,441	22.3%	1,575	1,651	4.8%	41,476	45,237	9.1%	13,327	13,008	-2.4%	6,722	6,243	-7.1%
September		9,122	8,438	-7.5%	1,775	2,199	23.8%	29,039	32,124	10.6%	1,302	1,079	-17.1%	30,068	30,663	2.0%	8,795	8,275	-5.9%	5,025	4,162	-17.2%
	YTD	20,647	20,162	-2.4%	3,549	4,000	12.7%	63,743	74,565	17.0%	2,877	2,730	-5.1%	71,544	75,899	6.1%	22,122	21,283	-3.8%	11,748	10,405	-11.4%
October		10,143	10,294	1.5%	1,263	2,420	91.6%	30,437	34,677	13.9%	1,319	1,436	8.9%	34,534	32,262	-6.6%	14,510	12,682	-12.6%	5,718	4,429	-22.6%
	YTD	30,790	30,456	-1.1%	4,812	6,420	33.4%	94,180	109,242	16.0%	4,196	4,166	-0.7%	106,079	108,161	2.0%	36,631	33,965	-7.3%	17,466	14,834	-15.1%
November		10,094	8,986	-11.0%	1,293	1,715	32.6%	28,569	32,047	12.2%	1,369	1,145	-16.4%	33,800	33,063	-2.2%	12,368	8,308	-32.8%	5,817	4,543	-21.9%
	YTD	40,885	39,442	-3.5%	6,105	8,135	33.2%	122,749	141,289	15.1%	5,565	5,311	-4.6%	139,878	141,224	1.0%	48,999	42,272	-13.7%	23,283	19,377	-16.8%
December		7,737	8,343	7.8%	1,370	1,823	33.1%	23,324	28,021	20.1%	1,146	1,218	6.3%	32,534	30,514	-6.2%	12,262	7,057	-42.4%	5,443	4,418	-18.8%
	YTD	48,622	47,785	-1.7%	7,475	9,958	33.2%	146,073	169,310	15.9%	6,711	6,529	-2.7%	172,412	171,739	-0.4%	61,262	49,330	-19.5%	28,726	23,794	-17.2%
January		8,160	6,427	-21.2%	1,066	1,063	-0.3%	23,641	20,535	-13.1%	1,059	833	-21.3%	28,897	28,993	0.3%	4,279	5,573	30.2%	5,875	4,093	-30.3%
	YTD	56,781	54,211	-4.5%	8,541	11,020	29.0%	169,714	189,845	11.9%	7,770	7,362	-5.3%	201,309	200,731	-0.3%	65,541	54,902	-16.2%	34,601	27,887	-19.4%
February		5,127	4,762	-7.1%	1,274	882	-30.8%	14,297	13,600	-4.9%	760	615	-19.1%	23,810	26,135	9.8%	4,820	3,775	-21.7%	4,348	3,453	-20.6%
	YTD	61,909	58,974	-4.7%	9,815	11,902	21.3%	184,011	203,445	10.6%	8,530	7,977	-6.5%	225,119	226,866	0.8%	70,361	58,677	-16.6%	38,948	31,340	-19.5%
March		5,341	5,349	0.1%	919	837	-8.9%	12,950	14,011	8.2%	725	649	-10.5%	25,845	22,991	-11.0%	5,288	2,261	-57.2%	3,253	2,746	-15.6%

Figure 3: Dairy Australia Data

IRELAND						
x 1.000 ton	2010	2011	2012	2013	2014	2015
Raw milk area						
Number of cows ('000 head)	1.071	1.117	1.141	1.163	1.226	
Milk production	5.350	5.556	5.399	5.600		
± % from previous year		*+3,9%*	*-2,8%*	*+3,7%*		
Deliveries to dairies (Mil Lt)	5.327	5.537	5.380	5.582	5.816	6.591
% Delivery on production	*99,6%*	*99,6%*	*99,6%*	*99,7%*		
Farm-gate price (Euro per 100 Kg)	31	34	32	38	37	
Dairy Production						
Liquid Milk	507	509	502	494	488	
Cream	10	10	11	11	12	
Butter	138	146	145	152	166	200
Cheese	172	180	186	183	188	207
± % from previous year		*+4,5%*	*+3,2%*	*-1,5%*	*+3,1%*	*+9,8%*
WMP (whole milk powder)	0,0	0,0	0,0	0,0	0,0	0,0
SMP (skimmed milk powder)	58	64	45	40	61	121

Figure 4: CLAL Dairy Market Analyses

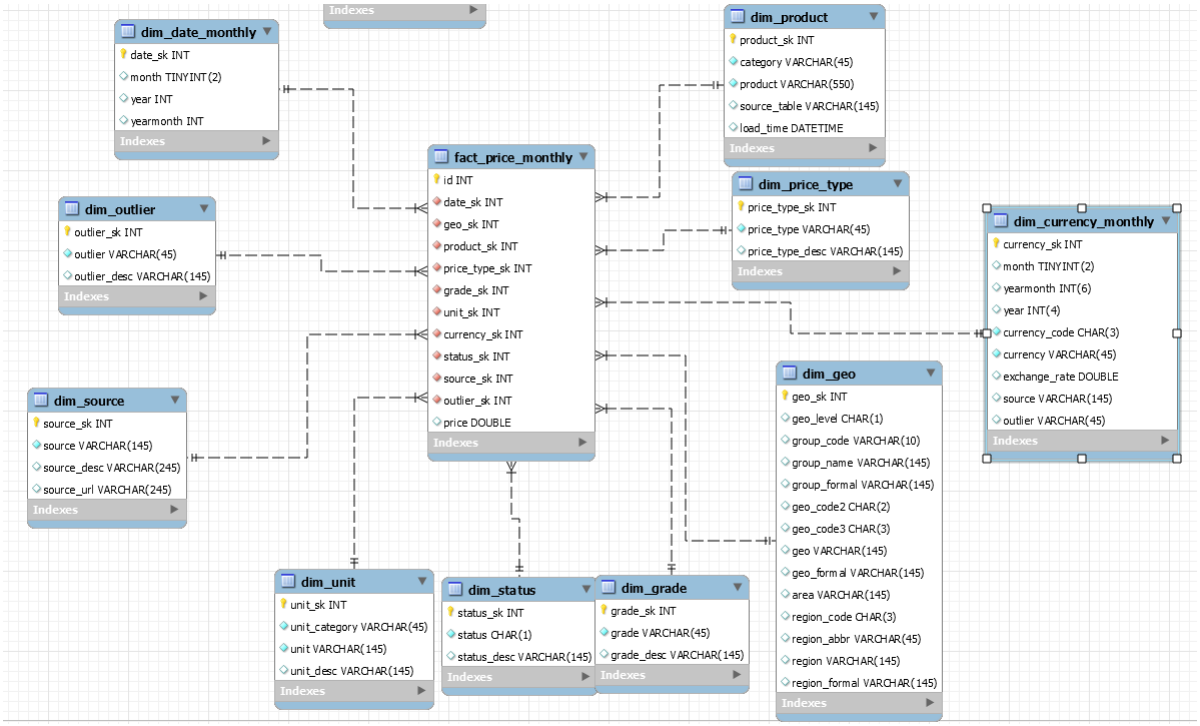

Figure 6: Fact_Price_Monthly Data Mart