Data Integrity Model for Environmental Sensing

Daniel T. Siegel, Leanne D. Keeley, Poorva P. Shelke, Andreea Cotoranu, Matt Ganis
Seidenberg School of CSIS, Pleasantville, New York

{ds68328p, lkeeley, ps54279n, acotoranu, mganis}@pace.edu

***Abstract*—Since the term "Big Data" began to circulate in the 1990s, a vast array of technologies and techniques have been developed in order to store and process such large and complex data sets. With a greater ability to obtain and compute such data sets, came a loss in the ability to ensure accurate data collection using traditional methods. This study aims to develop a model for data integrity, accurately detecting and compensating for errors and data drift. We will be working with a real-time water monitoring device that measures data in regular intervals. The data is managed in a decentralized manner by a Cloud-based Database Management System. We begin by exploring various drift compensation techniques. Afterwards, several algorithms implemented in similar studies are investigated. Upon choosing the optimal algorithm with any proper modifications, it is then implemented into an environmental sensing web application. In addition, new data visualization techniques are proposed to detect and alert for when data begins to drift.**

***Key Terms*— Data Integrity, Data Drift, Data Cleaning, Machine Learning, Linear Regression**

I. Introduction

Since the Spring of 2017, a real-time water quality monitoring device has been continuously developed by students of Pace University's Seidenberg School of Computer Science and Information Systems and Dyson School of Arts and Sciences. The device dubbed ADA, has been utilized for environmental research, as well as studies pertaining to Internet-of-Things technologies. With several sensors floating a few feet beneath the surface of the Choate Pond, ADA collects several types of environmental data, and sends a set of data to a central computer at Pace University over a Wi-Fi connection. This process occurs in real-time at 15-minute intervals. Initially, this real-time data was presented in a simple chart with limited scalability [1]. By the Fall of 2017, a web-based application was developed to present the data with more data visualization options for the user [2]. Earlier in 2018, a study was conducted to simulate management and analysis of "Big Data" by expanding the data set to include an instance of data taken from a sensor over the course of several years. With this newly implemented mock-data, a team proceeded to incorporate Charts.js into the application to extend the visualization of such a large data set over an extended period of time [5].

With this in mind, there are several challenges to face with this newly incorporated set. While big data has emerged with many opportunities for businesses and organizations to interpret vast amounts of data in real-time and make more informed data-driven decisions than ever before, it has also presented a troubling issue commonly referred to as drift. Data drift is defined as "The unpredictable, unannounced and unending mutation of data characteristics caused by the operation, maintenance and modernization of the systems that produce the data" [20]. Any changes made to the systems on which ADA operates, combined with the jump from traditional data to big data, affect the data.

Several reasons may be listed for how big data contributes to drift, but they can all be summarized by one word: Decentralization [17]. Under traditional data processing methods, data sources were managed by one or more entities within a single organization. The data technologies were usually static, managed from within, and changed through internal IT governance. The data sources had a defined schema that rarely changed, and when they did, those changes were agreed upon by the organization. Finally, under one roof, processing and analysis could be performed in single batches. The shift to the cloud architecture made drastic changes to the ease of traditional data handling. Changes to the schema occur more often and abruptly due to a shift towards less structured data. The data sources are now managed by third parties and are often done so poorly. The data processing is done by a collection of separate software components with changes implemented outside of an organization's IT governance. Processing and analysis have been made more complicated, requiring combinations of batch, event-driven, and streaming operations [20]. Basically, any processing technology that relies on a stable, unchanging environment under centralized control, is set up for failure in a big data-driven world.

Drift often leads to loss of data fidelity and deterioration of data operations, both having an adverse impact on an organization. A common phrase in the field of data science is "Garbage in, Garbage out" [12]. This means corrupted, missing, or misinterpreted data can lead to poor and overlooked insights. Often these bad assumptions, or lack of good ones, drive harmful business decisions or erroneous scientific conclusions in addition to costing the organization time and resources for analysis. Finally, any degradation of an organization's data will harm its reputation.

Currently, deviations from normal data readings are being displayed on ADA's web application. Incorporating a data integrity model is the ideal way to counter drift as well as chance errors. Before designing a model, this study determined the potential type of drift affecting the sensors. From there, various drift compensation methods and algorithms were tested to try to reduce the error rate on the data readings.

II. Literature Review

A. The Jefferson Project

The research surrounding this large-scale environmental research project has been very influential to similar-scale projects such as ADA. A joint effort of IBM Research, Rensselaer Polytechnic Institute, and the FUND for Lake George, this project utilizes a network of 42 sensor platforms, all connected by "Internet of Things" technology [15]. These platforms gather measurements of data relating to water temperature, air temperature, pH, oxygen content, salinity, and dissolved organic matter at regular intervals. This data is then shipped to several supercomputers over a cellular Internet connection.

In 2014, a report by the FUND revealed that rising salinity due to road-salt runoff was a threat to the state of Lake George, and recommended further research. While road-salt runoff from tributaries had long been thought to be a non-issue, the report found that the salt level in the 32-mile lake had tripled since 1980. This was a cause for alarm [11]. Knowing that the application of road-salt during and after storms leads to increases in salt runoff, the researchers reacted. Their response, was the development of sensors with the added function of predicting and adjusting to changing weather and lake conditions. These sensors have been applied to a four-pronged model which includes a weather model, a runoff mode, a salt model, and a circulation model. Each of these are designed to make predictions for how compounds, especially road salt, are affecting the water in the lake, and the precipitation that transports these compounds [10].

With this newfound ability to predict data drift, The Jefferson Project has been able to convince surrounding municipalities to be more tactical regarding road-salt application. The data integrity model in this instance has played a significant role in maintaining the quality of the water, essential to the regional economy.

B. The River and Estuary Observatory Network (REON)

Concerned with the amount of human impact on areas where rivers and estuaries meet the shoreline, The Beacon Institute launched its own collaboration with IBM and Clarkson University. The goal of REON was to create a water monitoring and forecasting network of its own in 2007. Their first monitoring platform, B1, was deployed on the Hudson River. This, along with two additional "B" units, laid the foundation for a second generation of monitoring technology. The web of interconnected sensors was dubbed the Real Time Hydrologic Stations (RTHS). REON's array of sensors gathered massive amounts of data for temperature, pressure, salinity, and turbidity by the minute [19]. For rapid analysis of the high volume of data, REON makes use of IBM's "System S" to stream from the various sources [21].

As the operations expanded to the capital region, New York City, and Cornwall, REON began to offer its own web application providing data visualization on its website. A user may choose one or more variables to observe over the date range of their choice. It is in a similar vein to ADA, although there are multiple rivers or estuaries to choose from on their application. Furthermore, REON's application displays charts of single variables before combining them into a single chart [18].

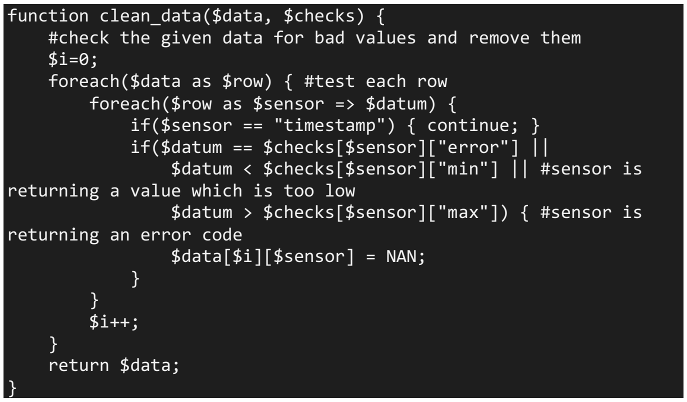
III. Research Requirements

A. Data Integrity

Barbara Martin of The American Water Works Association defines data integrity from a business standpoint stating that the concept "refers to the accuracy and consistency of stored data. Data integrity is imposed…through the use of standard rules and procedures, and is maintained through the use of error checking and validation routines" [13] While this definition comes from businessdictionary.com, Martin claims that it is equally applicable to data analysis of water quality data. When applying this definition to the current study, the two points that resonate are "error checking" and "validation routines". Establishing a baseline for our data integrity model to deal with outliers, or an error mode in software setup as Martin suggests, are potential solutions to explore.

B. Data Cleaning

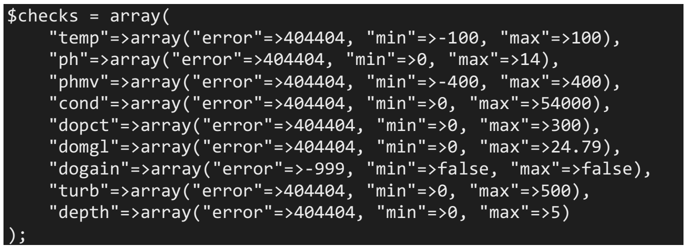
In order to minimize "Garbage in, Garbage out"[12], dirty data needs to be cleaned. Cleaning entails checking for outliers, normalizing raw data, and deciding how to fill in missing values. Data for Project ADA has been received in a CSV file, and outliers have been observed as well as values not consistent with standards. For instance, several cells show a turbidity of 404404, a number far out of the normal range, and represents the HTTP response code for error. We also observe an occasional value of 999.9 for dissolved oxygen levels. Even outliers that represent legitimate data points can deliver undesirable results to data models. While simply deleting records with bad data seems easy enough, if done too often it will reduce the accuracy of our data integrity model. Hence a significant time will be spent on data cleaning. Data cleaning alone takes up to 60% of the time in most data mining processes [12]. As shown by Fig. 1 data cleaning can be accomplished when working with the data using a script that locates error codes and outliers and replaces them with flag values such as "NaN" or "?", indicating that a proper data point is missing.



```
function clean_data($data, $checks) {
    #check the given data for bad values and remove them
    $i=0;
    foreach($data as $row) { #test each row
        foreach($row as $sensor => $datum) {
            if($sensor == "timestamp") { continue; }
            if($datum == $checks[$sensor]["error"] ||
                $datum < $checks[$sensor]["min"] || #sensor is
returning a value which is too low
                $datum > $checks[$sensor]["max"]) { #sensor is
returning an error code
                $data[$i][$sensor] = NAN;
            }
        }
        $i++;
    }
    return $data;
}
```

Fig. 1. Current study's script for error checking

In addition, sanity checks need to be implemented in order to ensure that the values returned by the sensors are valid according to their attribute. For example, pH values below 0 or above 14 are not possible, and water temperatures of over 100 degrees Celsius would represent water vapor. Such values outside of a specified range are best removed prior to analysis [6].



```
$checks = array(
    "temp"=>array("error"=>404404, "min"=>-100, "max"=>100),
    "ph"=>array("error"=>404404, "min"=>0, "max"=>14),
    "phmv"=>array("error"=>404404, "min"=>-400, "max"=>400),
    "cond"=>array("error"=>404404, "min"=>0, "max"=>54000),
    "dopct"=>array("error"=>404404, "min"=>0, "max"=>300),
    "domgl"=>array("error"=>404404, "min"=>0, "max"=>24.79),
    "dogain"=>array("error"=>-999, "min"=>false, "max"=>false),
    "turb"=>array("error"=>404404, "min"=>0, "max"=>500),
    "depth"=>array("error"=>404404, "min"=>0, "max"=>5)
);
```

Fig 2. Values used for data cleaning and sanity checks

C. Drift Type

Data drift has an adverse impact on the reliability of the data, the operations pertaining to it, and ultimately the productivity of the end-users of the data. The transition from traditional data processing to "Big Data" introduced this challenge, with its changes to processing architecture typically based on the assumption of stability. There are three types of data drift: Schematic Drift, Semantic Drift, and Infrastructure Drift.

1) Schematic Drift: Also referred to as structural drift, schematic drift occurs when the data schema changes at the source. Such changes may include the addition, deletion, or reordering of attributes. In addition, changes to the structure and incompatible changes to existing attributes are a significant source of schematic drift. For instance, an identification number attribute may be altered to contain additional digits to support a growing customer database. Without a proper model to support such a transition, any two or more records containing the same set of characters, with regard to the previous form of the ID number, may be conflated [14].

2) Semantic Drift: This type of drift refers specifically to the interpretation of the data. Under this change, consumers may no longer apply their previously understood interpretation. Changes from imperial to metric measurements are classified as semantic drift. In the business world, a common example occurred during the transition from IPv4 to IPv6 protocol. This transition led to misinterpretations of data, leading to false positives originally thought to be revenue spikes [14].

3) Infrastructure Drift: As the transition to Big Data continues to occur, control of data repositories becomes more decentralized. Unlike the traditional single stack model, management of data within the cloud architecture becomes difficult when changes to the underlying software or processing occur. Since each governing body of data source systems has their own standards, any change to the software components can create an incompatibility for existing operations [14].

In the context of this study, there is reason to believe that the type of drift that the team is faced with is Semantic Drift. The reason lies in the interpretation of the ADA data. When looking beyond chance errors, and more into systemic errors relating to values outside an appropriate range, the question becomes "What is the source of these abnormal values being displayed?". Have the sensors, or any part of the infrastructure for that matter, been contaminated, thus corrupting the data? Or are these data values legitimate, and the users are witnessing an entirely different phenomenon, such as the effects of climate change?

Identifying drift in the data can be done by comparing it to similar data collected independently. This can be done a variety of ways. One way is through the deployment of separate sensors to audit the data against. Another is by testing the sensors in a known environment pre- and post-deployment. For example, placing a water temperature sensor in a solution for which the temperature is known in a lab setting [7]. Comparing data against data which has been collected elsewhere, but is comparable, is another method. A final way is by finding baseline numeric relationships between the different environmental metrics collected. If these relationships are accepted as representing normal data, if new data fails to maintain these relationships, the data can be said to have drifted.

D. Drift Compensation Methods

After determining the source of the data drift, the subsequent task is to determine the best method to compensate for this drift. These approaches relate to multivariate techniques for signal processing.

1) Orthogonal Signal Correction: This technique corrects for variance in an array or matrix of sensor data by finding a vector which is orthogonal to (statistically independent of) the observed concentration vector [22]. This allows it to counter drift without altering the existing relationships between the individual data points.

2) Component Correction: The process for Component Correction is as follows; First, find a score vector for a reference metric. Then, calculate the drift component correction using that score vector. Finally, apply the resulting component correction to the non-reference metrics to remove the effects of drift [3].

3) Component Deflation: A method that uses Canonical Correlation Analysis to find linear dependencies between sets of data and uses that to find a regression model that describe the drift that is affecting the data. It then deflates the data by applying the inverse of the regression expression to counter the drift [8].

E. Linear Regression Function

To accurately predict errors and drift, this study explores machine learning algorithms that are commonly used in predictive tasks. The linear regression function is a common knowledge discovery method that contribute to prediction as well as other data mining tasks [12]. This classifier takes numeric inputs and learns a linear regression model. Some implementations find one attribute that provides the lowest squared error, others create a function that incorporates multiple attributes. The resulting function can be used to define the relationship between variables [12] and detect changes in the relationships.

This study has chosen to apply the linear regression function. It is a powerful machine learning algorithm that generates a simple function that can be easily implemented and comprehended, even by those not familiar with machine learning techniques. In many cases, the relationships it describes between the sensors can be validated by a search of the environmental sensing field's literature. Additionally, the limited number of variables in a linear regression equation allows it to be easily rebuilt on a regular basis as more data is incorporated into the model.

IV. Analyses

For this study, air temperature data collected by NOAA's NCDC at the Westchester County Airport and air and water temperature data collected by the HRECOS project at Piermont Pier on the Hudson River was used to audit data collected by ADA [16] [9]. This is done by merging the data on time stamps and plotting the data in a line graph to visualize trends in the data.

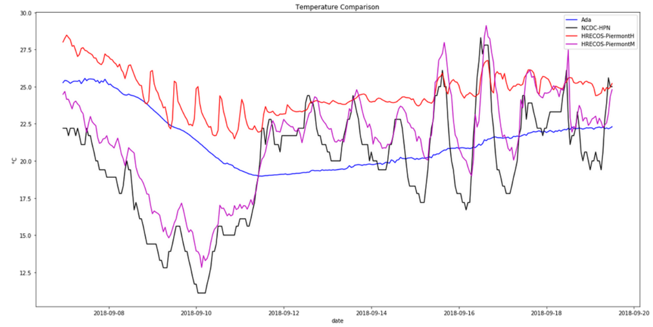


Fig. 3. Comparison of data trends between four sources

An increasing difference between the value of the ADA sensor data and the other temperature data is visualized by plotting the delta value between the ADA data and the other data against time to see if the difference is growing over time, indicating that ADA's data has drifted.

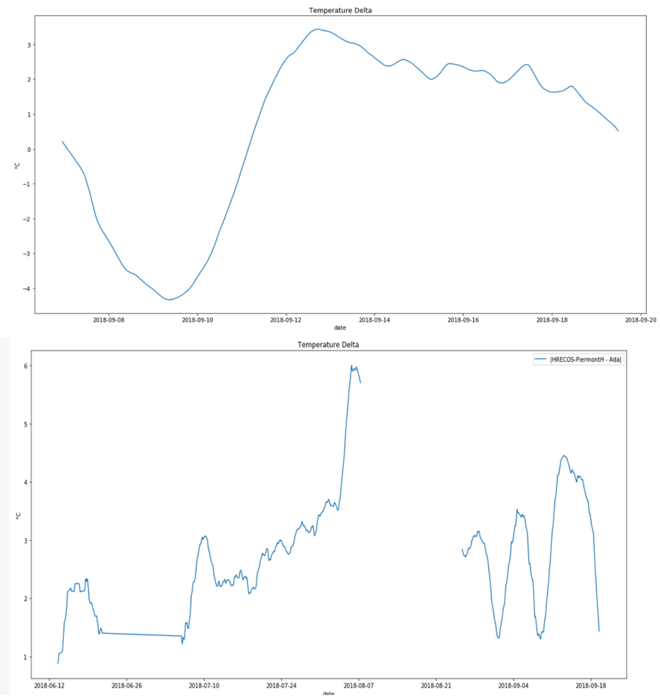
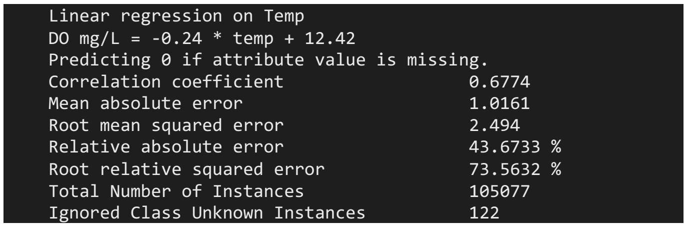


Fig. 4. Temperature delta between ADA and three other data sources

Another method of detecting data drift is by auditing the various sensors against one other. We know that temperature affects dissolved oxygen [4]. If a function can be found that adequately expresses the relationship between these data points, we can use it to tell if one or both of the sensors is drifting. This can be done by using a machine learning algorithm to calculate the best linear regression for the known data. The result of this is a simple mathematical expression of the way the data is usually related.



```
Linear regression on Temp
DO mg/L = -0.24 * temp + 12.42
Predicting 0 if attribute value is missing.
Correlation coefficient                 0.6774
Mean absolute error                     1.0161
Root mean squared error                 2.494
Relative absolute error                 43.6733 %
Root relative squared error             73.5632 %
Total Number of Instances               105077
Ignored Class Unknown Instances         122
```

Fig. 5. Linear Regression for DO mg/L found using WEKA

This can be deployed on a live server as a simple pass / fail check comparing new data with expected data. If new data does not match the expected data to within a certain confidence, we can issue an alert that the data from one of the sensors is drifting. See Fig. 9 for this study's implementation of such a check. It can also be used to visually compare predicted and observed values. See Fig. 10 for an implementation of this idea.

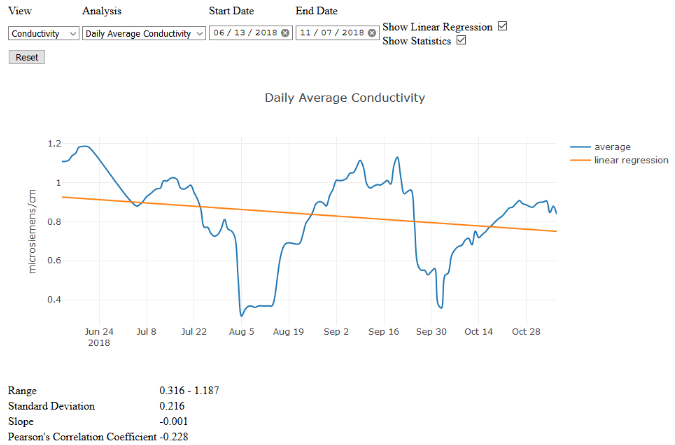
V. Integrity Model & Visualizations

In conjunction with an added function of preserving data integrity, this study's clients at Pace University's Environmental Systems CoLab have suggested implementation of new data visualization tools in order to view problematic data values. These new features expand upon the work of ADA's previous capstone teams, offering new options for viewing the data over time. The static graph has remained an option for viewing data for a specified period of time. An animated graph has been added, displaying the past week of data returned by the sensors graphed together. Finally, an admin dashboard has been included for administrators of the ADA project to assess the overall health of the sensor platform. The information provided by this dashboard can be used in the diagnosis and remediation of sensor issues and drift.

A. Static Graph

A static line graph for the data has been implemented using Plotly for Javascript as well as Chart.js. Plotly allows more flexibility in terms of zoom options, as well as interactivity. Charts.js offers a smoother, easier to read image. Two dropdown menus give the user the ability to choose which sensor data to observe on the right and left axis. The start date and end date inputs have remained, allowing the user to choose a desired time frame to view the data. Additional dropdown menus for each axis allow the user to choose from a variety of analyses. The default option selected is Raw, while other options include Daily Average, and Daily Range for each sensor. Growing Degree Days and Heating Degree Days are analyses that can be selected for temperature data. Checkbox inputs allows the user to view a table of statistics describing the data they are looking at and view a linear regression line for the selected data.

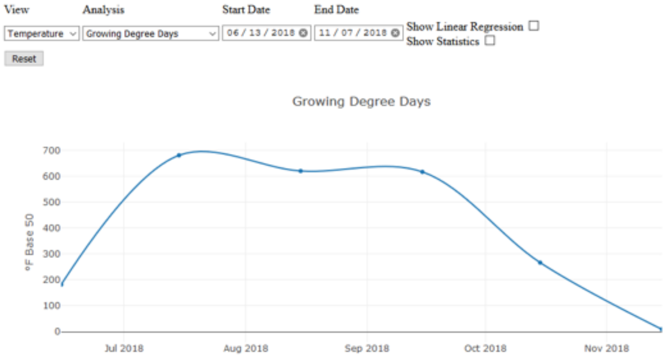
The requested data is pulled from the database in which the sensor data is stored. While the raw data stored is not altered by any data processing, any records requested via user input will undergo a process before being presented on the static graph. Even when the "Raw" analysis is chosen by the user, the data is still run through the cleaning algorithm. The error checking code alluded to in Fig. 1 is applied. Any records consisting of "NaN" values or impossible values are removed. After the selected data has been cleaned, Plotly and Chart.js plot the sensor data on the Y-axis and time on the X-axis. A user may gain a more detailed view of data points by hovering their cursor over the line, a feature provided by Plotly. When the "Daily Temperature Range" analysis is selected, the difference between the highest value and the lowest value recorded for each date is plotted. The Daily Average analysis computes the (Max – Min) / 2 for each day in the selection. For the Growing Degree Days analysis, the cumulative growing degree days (represented by the function *GDD = (Tmax – Tmin)/2)-BaseTemp* for each month within the selected range is plotted [5]. The base is set at 50ºF. Heating Degree Days is calculated similarly with the function *HDD = BaseTemp – (Tmax – Tmin) / 2)* [5]. The base is set at 65ºF. The statistics that can be calculated and displayed at the request of the user include the range, the standard deviation, the slope calculated for a linear regression line fitted to the data, and Pearson's Correlation Coefficient, which measures the linear relationship between the data and timestamp.



View: Conductivity
Analysis: Daily Average Conductivity
Start Date: 06 / 13 / 2018
End Date: 11 / 07 / 2018
Show Linear Regression ☑
Show Statistics ☑
Reset

Range	0.316 - 1.187
Standard Deviation	0.216
Slope	-0.001
Pearson's Correlation Coefficient	-0.228

Fig. 6. Graph of Daily Average Conductivity showing linear regression and statistics



View: Temperature
Analysis: Growing Degree Days
Start Date: 06 / 13 / 2018
End Date: 11 / 07 / 2018
Show Linear Regression □
Show Statistics □
Reset

Fig. 7. Graph of Growing Degree Days

B. Animated Graph

In addition to a static graph representing data in a specified range, users have the option of viewing an animated graph showcasing a week's worth of data returned by the sensors graphed together. The default view is to look at data for all the available sensors. The user can use checkboxes to select which specific sensors they wish to view.

Preprocessing of a week's worth of data is as follows. The application pulls the latest 672 data records. Recall that a new record is created every 15 minutes. Four records multiplied by twenty-four hours multiplied by seven days yields 672 records. The same data cleaning function used for the static graph is applied to the animated graph before the data is presented. Then, the application applies a linear scale to the datasets returned by each sensor, in order to make the ranges presented by each sensor equal, so as to graph them together. Each sensor returns data with different units, so the data is presented as percentages of the observed range so that they will occupy the same vertical space on the graph. Thus, the relationships between the different lines can be more precisely observed. This is achieved with the function $f(x) = z' + ((y' - z')/(y - z)) * (x - z)$ where x is the variable; y is the maximum observed and y' is a given range, in this case 100; z is the minimum observed and z' is 0.

The data is displayed smoothly across as much of the screen as possible. Each sensor line is of a different color and drawn simultaneously to highlight how the data changes overtime and to visualize the relationship between different types of data. To achieve this, the domain and range of the graph is set before drawing the lines, so the graph is not zooming out and panning left to right as the graph is drawn. The data is then graphed two points at a time for a smooth progression through time. Animation time is minimized and the graph is extended as opposed to being redrawn to make the animation proceed quickly.

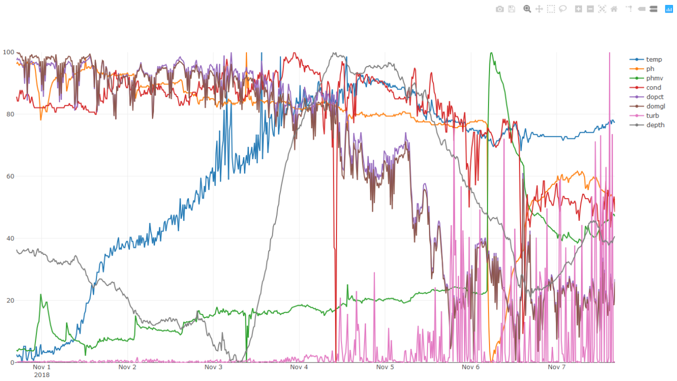
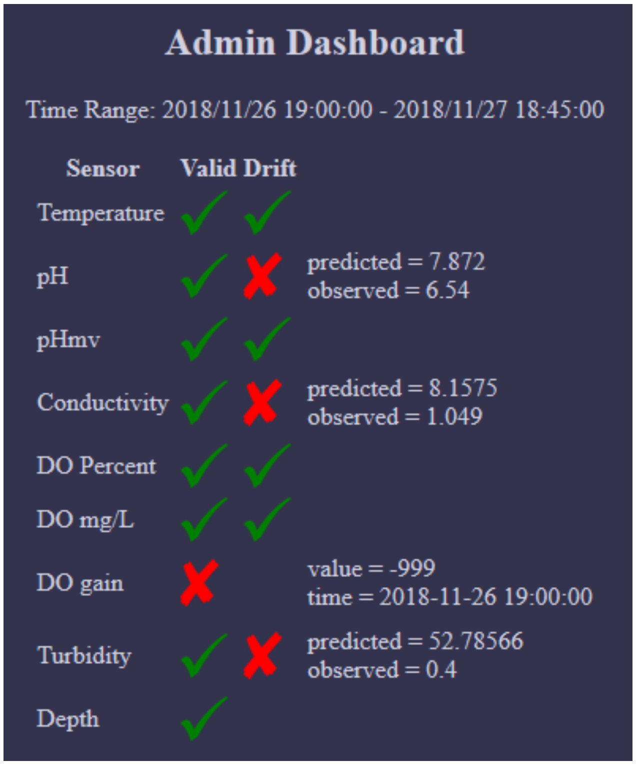


Fig. 8. Graph of all sensors graphed with the same range

C. *Admin Dashboard*

The health of the ADA sensor platform is presented to the administrator in chart format. The past twenty-four hours of data from each sensor are presented. If some problem is detected within this time period, the sensor will be flagged, or marked as bad. In those cases, the first problematic data point is presented with its timestamp. Health is determined by doing a validation check and a drift detection check. For validation, the data is checked against a variation of the data cleaning function (See Fig. 1). Instead of replacing a value with "NaN", the sensor is simply marked as returning problematic data, represented by a red X mark beside the sensor's title and printing the offending data point and its timestamp.

Drift is determined by auditing the sensors against one another. This is achieved by running the full set of available data at the time when the application is built as a training set from the sensor through WEKA's Simple Linear Regression classifier with one sensor selected as the value to be predicted. To include several iterations of the yearly cycle of seasons, the data was extended with data downloaded from the Piermont Pier observing station maintained by HRECOS [16]. This results in a simple equation that describes the relationship between the sensor being predicted and the sensor that best predicts the observed value. This linear equation (in addition to the value to be predicted in the case of a missing value) is used to build a function which can be implemented by a webserver to predict a value. The root mean squared observed by WEKA in the training dataset is then used to determine if the observed value falls within an acceptable range of the predicted value. If the absolute value of the difference between the predicted and observed values is greater than the root mean squared error, the value is marked as possibly representing data drift.



Admin Dashboard

Time Range: 2018/11/26 19:00:00 - 2018/11/27 18:45:00

Sensor	Valid	Drift	
Temperature	✓	✓	

pH	✓	✗	predicted = 7.872
			observed = 6.54

pHmv	✓	✓	

Conductivity	✓	✗	predicted = 8.1575
			observed = 1.049

DO Percent	✓	✓	
DO mg/L	✓	✓	

DO gain	✗		value = -999
			time = 2018-11-26 19:00:00

Turbidity	✓	✗	predicted = 52.78566
			observed = 0.4

Depth	✓		

Fig. 9. Example of validation check and drift detection on admin console

This should then prompt the administrator to take further action, such as auditing the sensor in question in a controlled environment to double-check its calibration. One example would be the water temperature sensor auditing the dissolved oxygen as a percentage sensor. Such an inverse relationship has been documented [4], as well as confirmed by the WEKA machine learner as the best predictor of DOpct as alluded to in Fig. 3. It is important to note that no reciprocal drift detection checks should be allowed. For instance, if temperature is used to audit the dissolved oxygen sensor, then dissolved oxygen should not be used to audit the temperature sensor. The reason for this is, if these two forms of data drift in parallel, this could go undetected. A drift detection algorithm has been implemented for each sensor with the exceptions of the DO gain sensor, which is not currently functioning; and the Depth sensor, which cannot logically be predicted using any of the available other sensors.

The administrator can click on any of the sensor names to view a graph of the data of the past 24 hours of data for that sensor. The graph includes a line representing the value predicted by the WEKA Simple Linear Regression and a shaded area representing the acceptable range presented by the root mean squared error.

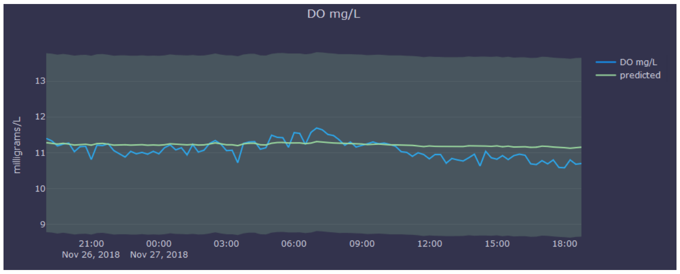


Fig. 10. Example of sensor view showing predicted and observed values

VI. Future Work

The ADA project has reached the point where an attentive administrator can act whenever readings of invalid or drifting data points occur. In the case of data drift, the web application presents a predicted value against an observed value outside of an expected threshold. Additional statistical analysis could be valuable to fine-tune the algorithms used for detecting drift and errors. An important next step, as suggested by Dr. Matthew Ganis, is to provide a countermeasure towards correcting this drift.

The ADA platform is still very young; it has not yet been in the pond for a full year. The sensors are sitting in a small, relatively static, body of water in the middle of an area highly trafficked by both motor vehicle and pedestrian traffic. This study was unable to find another sensor in the tristate area that is in a comparable environment. While the HRECOS data collected in the air and water on the Hudson River [16] and the NOAA air temperature data collected at the Westchester County Airport [9] were instrumental in building this study's models, we do not yet know in what ways the data collected by ADA will differ from these datasets. As additional high-quality data comes from the platform, the models developed by this study should evolve and change to fit this unique project.

VII. Conclusion

The results of the ADA project's current study, prove that techniques related to drift detection and error correction can easily be implemented into "Big Data" infrastructure. The combined use of error and sanity checking scripts, a linear regression function, and sensor auditing, all contribute to a robust data integrity model for a system continuously collecting data over an extended period of time. As such, the key requirements of data cleaning, finding trends, and drift alert set by our clientele, have been met. To demonstrate the value of this model, consider the following: Suppose the daily average temperature of the Choate Pond is observed to be three degrees higher than predicted over the course of a week. Under a system without sensor auditing, administrators would assume the readings to be legitimate and would be tempted to take unnecessary action. With the integrity model developed by this study, an administrator can see if the new temperatures fall within expected ranges or if the data are corrupted. Across various industries, data integrity models such as these may prove to be a valuable safeguard against consumer distrust, as well as a key to more reliable insights of data.

References

[1] Adelman, Jordan. Lamaute, Norissa. Reicher, Dan. Van Norden, Dallas. and Ganis, Matt, "Remote Sensing in a Body of Water Using an Adafruit Feather", Seidenberg School of Computer Science & Information Systems, Pace University Pleasantville, NY 10570, May 2017.

[2] Andari, S., Caruso, M., Ganis, M., Robbins, C.B., Whit, C., and Zada A. "Web Application for Environmental Sensing", Seidenberg School of Computer Science & Information Systems, Pace University Pleasantville, NY 10570, Dec 2017.

[3] Artursson, T., Eklov, T., Lundström, I., Mårtensson, P., Sjöström, M. & Holmberg, M. (2000). "Drift correction for gas sensors using multivariate methods", *Journal of Chemometrics, Special Issue: Proceedings of the SSC6* 14(5-6): 711–723.

[4] Beal, Bill. "Examining the Relationship Between Dissolved Oxygen and Water Temperature" *Project Watershed* http://projectwatershed.org/sites/projectwatershed.org/files/Relat_dissolved_oxygen_temperature.pdf Accessed November 2, 2018

[5] Caruso, Mark. Hassan, Joseph, Keeley, Leanne. Nikam, Sheetal. and Zada, AJ. "Web Application for Environmental Sensing: Monitoring and Analyzing Water Temperatures", Seidenberg School of Computer Science & Information Systems, Pace University Pleasantville, NY 10570, Feb 2018.

[6] Fischer, David. Kelly, Vickey. (2018) Personal communication, 18 October.

[7] Gastil-Buhl, Gastil. (2018) Personal communication, 18 October.

[8] Gutierrez-Osuna, Ricardo "Signal Processing Methods for Drift Compensation", *PRISM* 2nd NOSE II Workshop, Department of Computer Science, Texas A&M University College Station, TX 77843, May 2003

[9] "Hourly Data" NOAA National Centers for Environmental Information, Westchester Co Airport. Station WBAN:94745 Hourly Data. https://www.ncdc.noaa.gov/cdo-web/datatools/lcd Accessed October 17, 2018

[10] "In the Lab" The Jefferson Project at Lake George" Department of Biological Sciences, BT2149 Rensselaer Polytechnic Institute Troy, NY 12180 http://jeffersonproject.rpi.edu/lab Accessed Oct 1, 2018

[11] Johnson, Scott K. "Science by Robot: Outfitting the World's "Smartest" Lake", Ars Technica https://arstechnica.com/science/2015/04/science-by-robot-outfitting-the-worlds-smartest-lake/ Apr 18 2015

[12] Larose, Daniel T. "An Introduction to Data Mining" *Wiley-Interscience* John Wiley & Sons Inc. Hoboken, New Jersey 2005

[13] Martin, Barbara. "Tech-Tip – Ensuring Water Quality Data Integrity" *American Water Works Association* https://www.awwa.org/resources-tools/water-and-wastewater-utility-management/partnership-for-safe-water/partnership-resources/partnership-resources-details/articleid/4134/tech-tip-ensuring-water-quality-data-integrity.aspx Apr 12, 2016

[14] Pancha, Girish. "Big Data's Hidden Scourge: Data Drift", CMS Wire. https://www.cmswire.com/big-data/big-datas-hidden-scourge-data-drift/ Apr 8, 2016

[15] Picard, Ken. "The Jefferson Project Turns Lake George Into the World's Smartest Lake", *Seven Days* https://www.sevendaysvt.com/vermont/the-jefferson-project-turns-lake-george-into-the-worlds-smartest-lake/Content?oid=18412829 Jul 25, 2018

[16] "Piermont Pier Hydrologic Station Data" Hudson River Environmental Conditions Observing System. 2018.://www.hrecos.org/ Accessed October 17, 2018

[17] Prabhakar, Arvind. "Continuous Ingest in the Face of Data Drift (Part 1)", Cloudera http://vision.cloudera.com/continuous-ingest-in-the-face-of-data-drift/ Feb 1, 2016

[18] "Real-Time Hydrologic Sensing" *REON* http://rths.us Accessed Oct 1 2018

[19] "River and Estuary Observatory Network" *Beacon Institute for Rivers and Estuaries* https://www.bire.org/river-and-estuary-observatory-network/ Accessed Oct 1, 2018

[20] "Taming Data Drift – The Silent Killer of Data Integrity" *StreamSets Inc.* https://19ttqs47cfw33zkecq3dz58m-wpengine.netdna-ssl.com/wp-content/uploads/2016/07/Taming-Data-Drift-White-Paper.pdf

[21] "Why REON?" *Beacon Institute* https://www.thebeaconinstitute.org/approach/whyreon.php Accessed Oct 1, 2018

[22] Wold, S., Antti, H., Lindgren, F., Öhman, J. "Orthogonal signal correction of near-infrared spectra" *Chemometrics and Intelligent Laboratory Systems*, Volume 44, Issue 1-2, 14 December 1998