DETECTING SHIFTS IN THE PERFORMANCE OF THE SENTIMENT CLASSIFICATION ALGORITHM USING TWITTER FEED RELATED TO A PUBLIC COMPANY.

Ilya Samokhvalov

A Capstone Project Proposal Submitted to the
University of North Carolina Wilmington in Partial Fulfillment
of the Requirements for the Degree of
Master of Science

Department of Computer Science
Department of Information Systems and Operations Management

University of North Carolina Wilmington

2022

Approved by

Advisory Committee

Curry Guinn

Manoj Vanajakumari

_____

_____

_____

Douglas Kline, Chair

Accepted By

_____

Dean, Graduate School

TABLE OF CONTENTS

Abstract

The paper investigates the ability of well-established Statistical Process Control Techniques to efficiently monitor the performance of a Machine Learning classification algorithm and signal a possible deterioration of such performance. A timely and justified reaction to such signals may decrease the overall cost of maintaining the algorithms. The paper highlights the weaknesses of the current popular techniques and proposes a low-cost methodology to monitor the ongoing performance of classification algorithms.

List of Figures

Introduction

In recent decades, we have seen various transformations to every aspect of our lives, beginning with how we buy goods and services, catch a ride, communicate with each other via voice, video, or text, read books, or watch a favorite show. Undoubtedly, such transformations happened due to the incredible development of computers and related communication technologies.

The latest developments in the technology sector brought us various choices of how we get our news.

We can read them on a Facebook page, listen to a podcast of a news anchor, watch some YouTuber who reviews for us what happened lately, or you can get the print version of the Herald Tribune or turn on the cable news channel. With the growing variety of delivery choices, we see a growing number of news outlets and entertainment entities that fight for our attention. Bright and misleading headlines and opinions instead of the facts are just the tip of the iceberg of such fights. News outlets constantly republish the same news and developments on a 24/7 basis, keeping our attention tight. They do continuous coverage online and through other means. If you are not prudent enough, you can be buried under the sheer volume of useful and useless information.

Chankar [1] recommends skimming the headlines and disregarding certain distractions not to get lost in this pile of news. Alonzo and Tegmark [2] present an automated method for measuring media bias. The Machine Learning model they developed divides news outlets into a two-dimensional landscape – left-right bias and establishment bias. But such division contains

preconceived notions and definitions per se. It ignores the essence of journalism – neutral reporting of facts. The dictionary defines the word journalism as *the production and distribution of reports on current events based on facts and supported with proof or evidence* [3].

Broockman and Kalla [16], in a recent study, found evidence that partisan media impacts voting behavior. They claim that some news outlets shape the consumers' perception of events, indoctrinating into their minds what they need to think about an event. In other words, news outlets gradually become "opinion factories," and readers and viewers, on the other side of the receiver, become unwilling "opinion consumers."

Facts are emotionless, but the interpretation of facts can bear various sentiments. Imagine that you follow the coverage of a company you heavily invested in through your 401k account. You read through volumes of emotionless accounting data and see the word "write-down." Your brain instantly registers a bad sign, but is the overall sentiment of the article negative towards the company of your interest? It means you have to analyze the topic further. And what if there are hundreds of articles that potentially can bear some sentiment? Can you investigate them all so you don't miss any change in public perception that may, in turn, drive the stock price? This is where Artificial Intelligence (AI) and Machine Learning (ML) techniques may become indispensable helpers in the initial screening of the ever-growing amount of news. We shall monitor every aspect of the news and news cycle and weigh its sentiment and corresponding change. This way we may quantify and spot the news cycle trends and shifts in news editors' agenda and reporting.

The project attempts to structure the approach of transitioning news and other qualitative data into quantitative data that can be easily measured, monitored, and reported

with well-established statistical tools. The primary step of such an approach would be the development of sentiment analysis and text analytics techniques that can do the initial screening on a large scale in real-time. The proposed model employs a supervised ML algorithm that classifies the sentiment of the public messages posted on Twitter, a microblogging and social networking service on which users post and interact with messages known as “tweets.” The unique way to cross-intertwine messages, topics, interests, and accounts by special characters such as @ and #, made Twitter a powerful instrument in the hands of operators to attract the desired attention of brands and companies of interest and to track the public sentiment towards the areas of such interest.

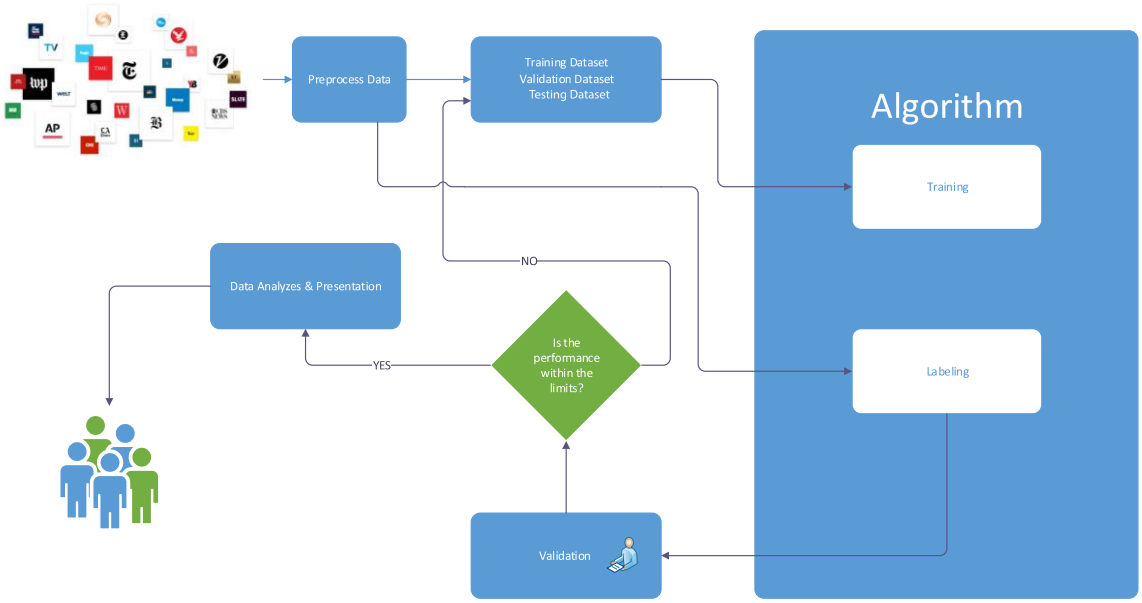

Figure 1. The flow of data within the Model.

The delegation of screening and classification assignments requires tools and methods to continuously monitor the quality of performed tasks. This project demonstrates the value of using Statistical Process Control (SPC) techniques for monitoring ML algorithm’s performance.

Figure 1 demonstrates the flow of data within the proposed model. The initial data represents unstructured textual data that must be preprocessed upon collection. The next stage includes the creation of the training dataset – a labeled dataset that becomes a learning tool for the ML algorithm. Once the algorithm is trained, it is deployed for fieldwork. The data flows through the same preprocessing steps and then is classified by the machine into predetermined classes. The next step involves random sampling from the pool of labeled data. Then human annotators classify the tweets, build the confusion matrix and analyze correct and incorrect classifications with SPC techniques, such as Control Charts for Attributes or Fraction Nonconformities per Unit. If the analysis of correct and incorrect classifications satisfies the requirements of the modeler, the model continues to work as is; if not, the algorithm is a subject for recalibration or retraining. Such validation and analysis are repeated on regular timely basis.

The project produces a unique data set of tweets dedicated to a single topic and collected over several months. The subset consists of 5,001 human-labeled tweets and serves as a training, validation and testing dataset for the ML algorithm. The project's main benefit is the development of techniques that can reduce the ML maintenance costs, and costs of retraining by signaling exactly when the performance has deteriorated, and interference is justified. The proof-of-concept demonstration spans a period of 17 weeks of observations.

Review and Analysis

Machine Learning and Public Sentiment Classification

The Twitter sentiment analysis topic is widely researched in the academic field and among practitioners. The overall difficulty of Twitter sentiment classification problems is aggravated by specifics of Twitter user's manner of speech, the prevalence of sarcasm, and other specifics of sentiment expression. In addition, the messages' length limitation forces the public to invent and use shortcuts such as emojis, emoticons, and other means of graphical and textual sentiment expression. This in turn, allows the development of a lexicon, a collection of symbols that define certain sentiments by public perception. Thereby, practitioners and scientists prefer to use three major approaches to classify tweets' sentiment—lexicon-based, supervised-learning, and a third hybrid approach, which combines the variation of both approaches. During the research of related work, we found that most studies focused on a comparative evaluation of different algorithms and limited themselves by use of common and easily available datasets without the analysis of performance of the algorithms during deployment.

Rout et al. [4] investigated different MA algorithms' relative performance on the classification of tweets acquired from the Twitter public domain. The Multinomial Naive Bayes (MNB), Maximum Entropy and Support Vector Machine algorithms were trained based on the unsupervised approach and lexicon-based approach. The authors reported an accuracy of 0.807 using the unsupervised approach and 0.752 using the lexicon-based approach, while the MNB algorithm achieved an accuracy of 0.67 using the unigram feature, a representation of word sequences where only one word is present.

Ren et al. [5] discussed the topic-enhanced word embedding for Twitter sentiment classification problems. The authors looked at the classification problem from an overall sentiment perspective rather than from the perspective of topic-based classification. The authors tested three algorithms overall – Naive Bayes, Maximum Entropy, and SVM. The accuracy of all three of them ranged from 0.73 to 0.83, with a mean of 0.77 and a standard deviation of 0.0253.

Ryan Ong [7] described his challenge identifying and categorizing offensive language within Twitter. The author experimented with variations of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) layers. The researcher concluded that the ordering of layers is extremely important for optimal model architecture and reported an F1-score, a measure of the overall accuracy of a prediction, within the range from 0.46 to 0.75.

Appel et al. [8] studied the comparative performance of the hybrid approach to sentiment analysis against Naive Bayes and Maximum Entropy algorithms trained on the same datasets—movie reviews, and sentiments Twitter datasets. The hybrid approach combines a Sentiment Lexicon, Semantic Rules, Negation Handling, Ambiguity Management, and Linguistic Variables. The authors reported the accuracy of 0.76 for the hybrid approach against 0.62 for the alternative method. Ghosh and Sanyal [9] investigated the comparative performance of the ML classification algorithms concerning the use of feature selection methods such as Information Gain (IG) Ratio, Chi-square, and Gini-index against unigram and bigram feature sets. The investigators used Recall, Precision, and F1-score measures to compare the performance of the algorithm. They reported the accuracy in the limits of 0.87.

Zainuddin et al. [10] proposed their version of the hybrid approach to the sentiment classification problem. The researchers compared the performance of the Principal Component Analysis (PCA), Latent Semantic Analysis (LSA), and Random Projection (RP) feature selection methods. They used three different datasets for the analysis, such as the Hate Crime Twitter Sentiment dataset, which consists of 1078 tweets in total, Stanford Twitter Sentiment dataset, which consists of 353 tweets, and Sanders Twitter Corpus dataset, that consists of 1091 tweets. They concluded that the hybrid sentiment classification, which combines the Association Rule Mining with a heuristics combination in part-of-speech patterns, provides statistically meaningful results at a *p*-value of 5%. However, the highest reported accuracy was reached by the SVM algorithm and was 0.76. The reported performance of the Extreme Learning Machine of Twitter aspect-based sentiment classification showed quite inconsistent results between average training accuracy of 0.84 and average testing accuracy of 0.58, which indicates that the model was overfitted. Stojanovski et al. [11] went with an overcomplicated approach and used CNN for extracting features for further classification experiments on Twitter data. They used three labeled datasets provided by the SemEval challenge. The accuracy for all the sets for 2013, 2014, and 2015 years did not exceed 0.71, with significantly lower results for the emotion identification classification, with which its accuracy didn't exceed 0.52. Kim et al. [12] provided a straightforward, self-explanatory application of AI algorithms in sentiment analysis of messages on the Internet. As an experiment, they "crawled" comments that were posted in online communities concerning cryptocurrencies, such as Bitcoin, Ripple, and Ethereum. The researchers claimed that they achieved an average accuracy of around 0.74 in determining the sentiment.

Abassi et al. [13] addressed the important topic of sentiment analysis techniques that facilitate the detection of hate-speech postings and other extremist publications on the Internet. The researchers used three hand-labeled datasets to train the model. One dataset collected the movie reviews, which Pang et al. [6], and later in 2017, Appel et al. [8] also used for their hybrid classification approach. The other two datasets were directly connected to the domain of interest, a US supremacist forum and a Middle Eastern extremist group forum, thus combining English and Arabic content. The authors developed the Entropy Weighted Genetic Algorithm for feature selection and used the SVM algorithm for the classification problem. They claimed the overall accuracy of over 0.91 on the bench-mark datasets, but, unfortunately, didn't provide out-of-sample performance metrics.

Practitioners prefer not even to report accuracy metrics. For example, MarketPsych, the sentiment data provider in the financial sector, reports only the final numbers – in-house calculated indices of fear, joy, pessimism, optimism, etc. The basis for indices is derived from thousands of news and social media outlets classified accordingly by lexicon-based ML algorithms [14].

Typical Preprocessing Stage of ML Sentiment Text Analysis

The typical preprocessing stage, also known as preparation and wrangling stage, involves filtering, cleansing, removing or replacing hashtags, numbers, punctuation, contractions and repeating characters, lemmatization, and lowercasing, and replacing emoticons and emojis with appropriate placeholder words.

Hashtags are integral elements of Twitter that convey sentiments and opinions and facilitate message delivery and search. For instance, the hashtag *#disappointed* transmits negative sentiment and is indexed along with all other tweets that have such a hashtag. However, the hashtag symbol (#) does not carry any semantic payload and thus has to be removed, equaling *#disappointed* to the word *disappointed*.

The contractions, such as '*don't,'* and repeating characters, such as '*loveeee it*', are common figures of speech which you can find in the Twitter feed. At this stage, contractions are normalized, and repeated characters were limited to two characters to maintain the emphasis.

Emoticons and emojis are pictorial representations of a facial expression using characters such as numbers, punctuation marks, brackets, parenthesis, etc. These representations are a common form of sentiment expression of a sender, saving time and space in short message exchanges via messaging services, especially on Twitter, where the use of characters used to be limited by 140 characters per message. Emoticons and emojis are replaced with corresponding placeholder words. Numbers and punctuation symbols are removed as well as they do not convey specific sentiment. Slang and abbreviations are usually correctly handled by MA algorithms [18] and are kept intact.

Stemming and lemmatization are common Natural Language Processing techniques that reduce inflectional and derivational forms of words to a common base form. For example, the words ‘likes’ and ‘liked’ bears similar positive sentiments but can be perceived by an algorithm as different. Such words are normalized to a common base form, ‘like.’

Dataset Selection

The simple online search reveals many hand-labeled datasets appropriate to train the ML algorithm and evaluate its current performance on out-of-sample data. This data was unseen by an algorithm during the training process but technically derived from the same dataset. For example, SemEval, a series of international NLP research workshops [13], offers human-annotated datasets that contain tweet IDs and sentiment. The dataset represents a collection of tweets on various topics and domains in a single set. Another popular dataset, the Sanders Twitter Corpus dataset, contains tweet IDs collected in 2008 during the Obama-McCain political debates. The Health Care Reform (HCR) dataset was created in 2010 and contained tweets related to public discussion of the Affordable Care Act and includes the hashtag *#hcr*.

These readily available online datasets and others that are free of charge and do not violate the Twitter Developer Agreement and Policy represent a snapshot of the Twitter feed in time. The datasets lack sufficient duration spanning several weeks and months of observations that are crucial for the goal of the project. Due to their data scarcity, they can be suitable for initial training of the various ML algorithms but not for evaluating the ongoing performance over time.

To make the research feasible in a given time constraint, the investigation was limited by following the public sentiment towards a well-known entity that generates a consistent volume of data each day and belongs to a broad public domain. Apple Inc., an American multinational technology company specializing in consumer electronics, computer software, and online services, was chosen as an entity of interest because it fits the requirements. In addition, the tweet collection was constrained to the English language only.

Confusion Matrix and Algorithm Performance Metrics

A confusion matrix is per Brownlee a widely used "*technique for summarizing the performance of a classification algorithm"* [19]. It is a graphical representation of correct classification and misclassification instances summarized with count and broken down by class. Figure 2 shows the schema of a common two-class confusion matrix, but it can be built for multiple classes problems. The columns divide classes by their actual labels defined by the human-annotator. In contrast, rows divide classes by the labels predicted by an algorithm but within the human-annotated classes. Hence, we can count each classification instance individually and keep records of correct and incorrect instances for further analysis. For example, if an algorithm classified an issue as Class 0, while in reality it belongs to Class 1, such instance will be counted and recorded as False Negative or Type II Error; if the real label is Class 0, but was classified as Class 1, the instance will be counted and recorded as False Positive or Type I Error.

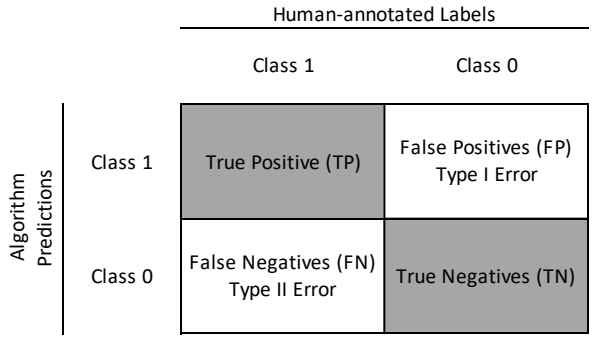

Figure 2. Two-class confusion matrix.

All the counts of instances serve as a basis for further analysis and calculation of various aggregate metrics. The most intuitive and straightforward aggregate metric is called Accuracy, which is a simple ratio of correct predictions, True Positive and True Negative, to the sum of all instances in a classification case. The highest ratio means better classification among comparable algorithms or cases. Unfortunately, Accuracy doesn't take into account the possible high cost of making a classification error and ignores skewness when classes are unevenly distributed.

The F1-score aggregate metric considers False Negative and False Positive counts and is a preferable metric when classes are unevenly distributed [20]. The F1-score measures the accuracy of the algorithm's predictions and is a harmonic mean of the combined ratio of Precision (P) and Recall (R) metrics. Precision (P) is a ratio of True Positives (TP) to the sum of TP and False Positives (FP):

$$Precision = \frac{TP}{TP + FP},$$

and Recall (R) is a ratio of TP to the sum of TP and False Negatives (FN):

$$Recall = \frac{TP}{TP + FN}.$$

The logic behind P and R is straightforward. Precision is the ratio of correct classifications to the total predicted labels, while Recall is the ratio of correct classifications to the total actual labels. Recall is also called a sensitivity. The F1-score states the equilibrium between the Precision and the Recall [20] and is calculated as follows:

$$F1Score = \frac{2 * P * R}{P + R}.$$

These metrics can be calculated for multi-class classification problems. In such a case, P and R are calculated for each class separately and then aggregated into a single metric for ease of comparison. There are several approaches to calculating the aggregate F1-score for multiple-class classifications.

Multiple class problems require building a multi-dimensional confusion matrix and calculating P and R for each class individually but imply a two-dimensional matrix. Each class of interest is calculated against the rest of the instances and its errors as another class.

Multiple P and R values based on multi-dimensional confusion matrix present challenges for calculating overall accuracy metrics when classes are unevenly distributed and present different interest for the modeler. The most widely used approaches for calculating aggregate metrics are Micro F1-score, Macro F1-score, and Weighted F1-score. The difference between these metrics is in the treatment of the skewness in the class sizes. The poor performance in small classes is not important for Micro and Weighted scores since the number of units belonging to those classes are small compared to the overall dataset size [23]. Hence, the weights of larger classes outweigh the weights of smaller classes even if they represent larger interest for the modeler. The Macro F1-score disregards the size of the classes and treats them as equally important. The Macro F1-score is the default metric that is used by researchers to

assess the performance of chosen algorithms. The detailed example of a three-class confusion matrix is given later in the document.

Statistical Process Control

Statistical Process Control (SPC) is a widely researched field of study as such, but not in the context of the ML classification process per se. Lo [21] employed the Support Vector Classifier to recognize defective messages among customers' online complaints and built the product of such classifications on *p*-charts to reflect unusual service quality changes. Ashton, Evangelopoulos, and Prybutok [22] analyzed customer opinions regarding product or service quality in an unstructured text and employed SPC techniques for the quantitative evaluation of customer acceptance of system process improvement initiatives. The research perceives the algorithms classification process as a separate production activity and treats the product of such classification as a production result justifying the use of SPC techniques per se.

The nature of the confusion matrix of our classification task justifies the use of the Control Chart for Attributes or Control Charts for Fraction Nonconformities, aka *p*-chart. The confusion matrix provides a set of correct predictions and error probabilities to the total class or the whole sample. The fraction nonconforming is a ratio of the number of nonconforming observations to the total number of such observations. The underlying statistical principle for such a chart is based on the binomial distribution.

SPC's objective is to notice instances of shifts in the quality of the production process. As a result, a timely reaction to such cases and corrective action may be possible before many nonconforming units are produced. Control charts as a process-monitoring technique are

widely used in spotting quality shifts, though with some degree of variability. The complete elimination of variability in the process through SPC is not entirely feasible per se, but control charts are an effective tool for reducing such variability. [15]

The control chart is a graphical display (Figure 3) of a characteristic that has been measured or computed from a sample versus the sample number or time. The Center Line represents the average value of the observable characteristic corresponding to in-control state. The Upper Control Limit and Lower Control Limit horizontal lines are chosen so that if the process is in control, all the sample points should fall between them. Hence, if the observations plot is within the control limits, the process is assumed to be in-control, requiring no further interference. [15]

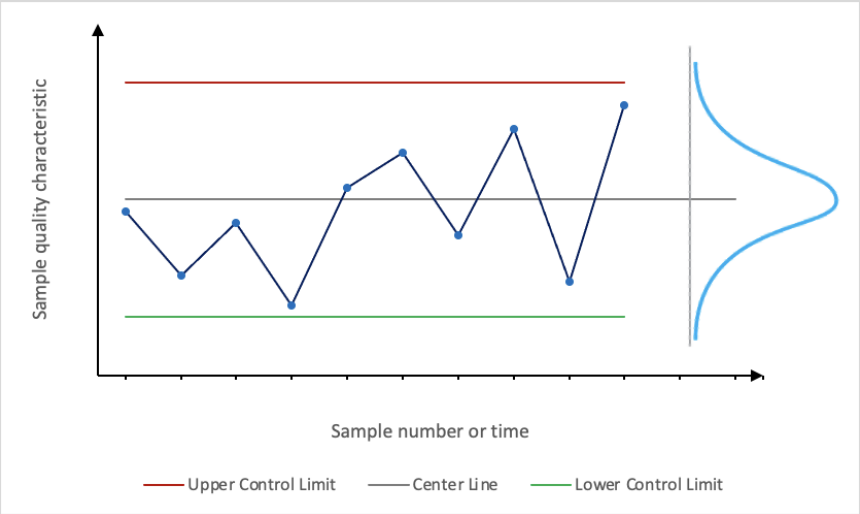

Figure 3. A typical control chart.

The popularity of control charts can be attributed to their ability to prevent unnecessary process adjustments and provide information about process capability and diagnostic information. In addition, control charts are effective in defect prevention and are a proven technique for improving productivity. [15]

Methodology

Building *p*-chart with Confusion Matrix

The binomial distribution defines the mean (μ) is equal to p, and the variance σ^2 is equal to $p*(1-p)$ divided by the sample size n. There are two approaches treating n. The first approach treats n as the sum of all the counts within the class. The second approach treats n as the total number of tweets in the sample.

Approach 1. *p* as a proportion of a class

For the illustration purposes Figure 4 considers one of the first confusion matrices built within the project, the rest of the matrices are presented in Appendix 3.

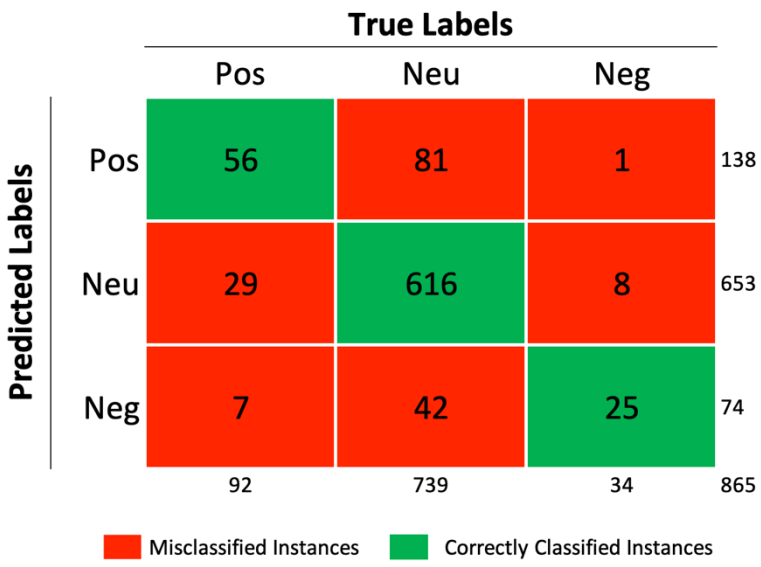

Figure 4. Example of the Confusion Matrix

Green squares represent correct classification instances and red squares represent misclassified instances. The columns designate true labels (classes) marked by human annotator, and horizontal lines represent predicted by ML algorithm labels.

The testing dataset presented in the confusion matrix (Figure 4) contains 865 hand-labeled tweets, with 92 positive, 739 neutral, and 34 negative tweets. Hence, the *p*-value for square 1 is equal to 0.609 (56 over 92), and *(1-p)* is equal to 0.391. In other words, the probability of getting the correct classification of a positive class is 0.609 and the probability of getting a classification error is 0.391 ((29+7)/92) within the positive class.

Approach 2. *p* as a Proportion of All Classes

The second approach is to treat *n* as the total number of tweets in the sample, which is 865 in this case. Thus, *p* for True Positive classification is equal to 0.065 and (1-p) is equal to 0.935. The approach limits the analysis of True Positive, True Neutral and True Negative classifications. These *p*-values, by nature reflect the probability of getting a tweet with positive [positive, neutral or negative] sentiment against other correct and incorrect instances at the point in time. The observation outside the control limits signifies a substantial shift in public perception rather than a shift in the quality of the classification process, given that corresponding errors are constant.

The standard deviation for both approaches is calculated by the following formula:

$$\sigma = \sqrt{\frac{p(1-p)}{n}}.$$

The Upper and Lower Control Limits are calculated as $p \pm L^{*}\sigma$, where *L* is the coefficient, distance of the control limits from the center line. The conventional heuristic of the empirical sciences, states that nearly all values lie within three standard deviations from the mean, given the normal distribution. In terms of probabilities, we can say that 99.7% of all observations lie within three standard deviations from the mean (three-sigma) or we may be

99.7% certain that we observe all the observations within such limits, or there is a 99.7% chance that all observations lie within the limits. Two standard deviations or two-sigma cover 95% of all observations, and one standard deviation or one-sigma covers 68% of all the observations. The given empirical probabilities can serve as a basis for the modeler for finding a trade-off between the sensitivity of the Control Limits to the observed *p*-values and signals it send.

The detailed findings and results are presented and discussed in later sections of the document.

The size of n influences the absolute values of Upper and Lower Control Limits. Hence, the limits for smaller classes in the first approach tend to be more kinked than the corresponding limits in the second approach. For example, the amount of positive tweets in the validation samples, which is *n* in the first approach, ranges from 6 in week seventeen to 65 in week five. In contrast, the sample size and *n* in the second approach are constant from week one to five and six to seventeen, making the limits look like a straight line above and below the central line.

Control Limits and Decision Rules

The major caveat lies in choosing the control limits magnitude. It is critical to understand the trade-off between the overall sensitivity of signals to the risks of getting erroneous signals. The Upper and Lower Limits are estimated with a standard deviation around the mean value, which is the Central Line. The wider limits, the further away from the mean, decrease the risk of a Type I error, which is a risk of a point falling outside the limits, indicating

an out-of-control state, while in reality, it is in-control. At the same time, the wider limits increase the risk of a Type II error, which is a risk of a point falling inside the limits while, in reality, it is out-of-control. The same is true for the opposite case. The lower limits increase the power of a test, decreases Type II error, but increases Type I error. Hence, the higher the coefficient that multiplies the standard deviation, the lower the probability of getting Type I error, but the higher the probability of Type II error [15].

Montgomery [15] distinguishes standard decision rules [action signals] for detecting patterns on a control chart that can signal that the process is out-of-control. They can be summarized as follows:

- one observation lies outside of the three-sigma control limits (three standard deviations away from the mean);
- two out of three consecutive observations lie beyond the two-sigma control limits (two standard deviations away from the mean);
- four out of five consecutive observations lie at a distance of one-sigma or beyond from the Center Line;
- eight consecutive observations lie on one side of the Center Line;
- six consecutive observations steadily increasing or decreasing;
- eight consecutive observations lie more than one-sigma below or above the center line;
- there is an unusual or nonrandom pattern in the data;
- one or more points near a warning or control limit.

Dataset Collection

The project followed the steps of Charlotte Teresa Weber and Shaheen Syed [18]. It was assembled in Python in a few major modules (stages). The first module collected the target tweets containing hashtags @apple, #apple, @aapl, and #aapl in lower and upper case via Twitter API, recorded them to text files, and later fetches them into the MongoDB database running on a local machine. The preprocessing stage (preparation and wrangling stage) involved filtering, cleansing, removing or replacing hashtags, numbers, punctuation, contractions and repeating characters, lemmatization, and lowercasing, and replacing emoticons and emojis with appropriate placeholder words. All the tweets were subject to the same preprocessing techniques, described in the previous sections of the document.

The overall data spanned the continuous period from December 01, 2021, to April 02, 2022, and contained 1,050,210 tweets, with average weekly tweets of around 59,998, ranging from 45,707 to 82,888 tweets per week. For the purpose of the analysis, the data was aggregated into the weekly observations starting with Sunday, December 05, 2021 on 12:00 AM, and concluding with Saturday, April 02, 2022 on 12:00 PM. Hence, the target dataset contained full seventeen weeks of observations, including 1,019,798 tweets (Appendix 3, Figure 56), but excluding the dataset of 5,001 tweets (Appendix 3 Figure 55) that, in turn, were sampled from the period from December 02, 2022 to December 15, 2022.

The 5,001 tweets dataset was hand-labeled by a single human, the author of the project, into three classes bearing distinct sentiment such as positive, negative, and neutral towards the entity of interest, Apple Inc.

The examples of tweets and their corresponding sentiment are given in the following figure (Figure 5):

#	Tweet	Tweet ID	Sentiment
1	Mine is updating now if that helps any	1469089056516628486	Neutral
2	Read the room @ApplePodcasts Prince William is now seen as a misogynistic racist.	1469074927651332099	Neutral
3	Do you use @Apple to listen to your #Podcast ? If so the subscribe now to listen to @NetZeroForNoth1 the home improvement podcast from @TheNHIC	1468996437560250372	Neutral
4	Wtf this IPhone 13 came w the phone & a wire charger. No block. No headphones. What's going on @Apple ?	1469302433683623937	Negative
5	@apple Just like Facebook and Google, Apple is NOT about privacy or meeting or exceeding its customers expectations. Think long and hard before buying your next Apple device	1468689847464312832	Negative
6	@Apple ei, your new OS damage my iphone! Fix it!	1468603711190708241	Negative
7	4,500 mAh. And yet hmmm… #apple iphone battery is still the best.	1469087644181479428	Positive
8	Love you @Apple 💝😘	1468483267854696449	Positive
9	I have tried so hard to get out of being captive to @apple - but DAMN - their products are soo good!	1470215771028549644	Positive

Figure 5. Examples of tweets with corresponding sentiment.

The corresponding sentiment classification were guided by personal perception and principles of the researcher of negative, positive and neutral sentiment towards the entity of interest. Such perception and principles were kept throughout the project and the tweets that were subject for human annotator validation during the monitoring stage were classified by the researcher himself.

Model and Algorithm Selection

The 5,001 tweets dataset was randomly split into three subsets: training set contained 60% of observations, validation set contained 20% of observations, and testing set contained 20% of observations.

Ada Boost Classifier, Support Vector Classifier (SVC), Linear Support Vector Classifier, Logistic Regression, and Decision Tree Classifier algorithms were trained to choose the one with better F1-score, following the work of Charlotte Teresa Weber and Shaheen Syed [18]. The researcher used the scikit-learn library for Python with generic settings to train all five algorithms, as the initial fine-tuning of the algorithm was not a goal of the project. Still, rather mediocre performance would be beneficial for the project as it is more likely to show shifts in performance.

Figure 6 shows the F1-score metrics of all trained algorithms based on the testing set. The Support Vector Classifier demonstrated the highest F1-score and was chosen to conduct the classification task.

Algorithm	F1-score
Linear Support Vector Classifier	0.781638
Logistic Regression	0.780386
Decision Tree Classifier	0.720498
Support Vector Classifier	0.785436
Ada Boost Classifier	0.738704

Figure 6. Algorithms' F1-score

Confusion Matrix and Control Charts

Throughout the document, the researcher refers to confusion matrix cells by designations mapped in Figure 7 for ease of communication and navigating the reader. Cells 1, 5, and 9 represent Correct Classification counts for Positive, Neutral, and Negative classes correspondingly, while cells 2, 3, 4, 6, 7, and 8 represent various classification error counts. Cell 2 contains counts of instances classified (predicted) by the ML algorithm as Positive but were Neutral; cell 3 contains counts of instances classified as Positive but were Negative. Cell 4 contains counts of instances predicted as Neutral but were Positive, and cell 6 contains counts of instances classified as Neutral but were Negative. Finally, cell 7 contains counts of instances predicted as Negative but were Positive, and cell 8 contains counts of instances classified as Negative but were Neutral.

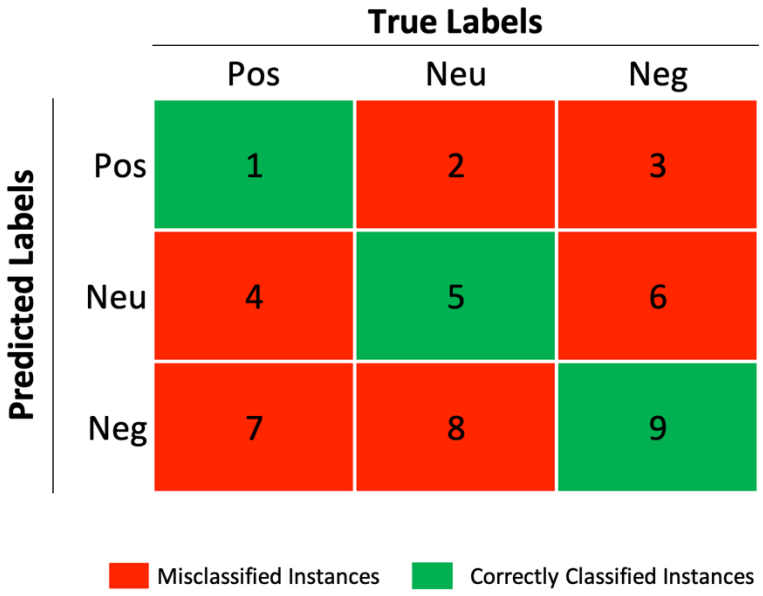

Figure 7. Confusion Matrix "map"

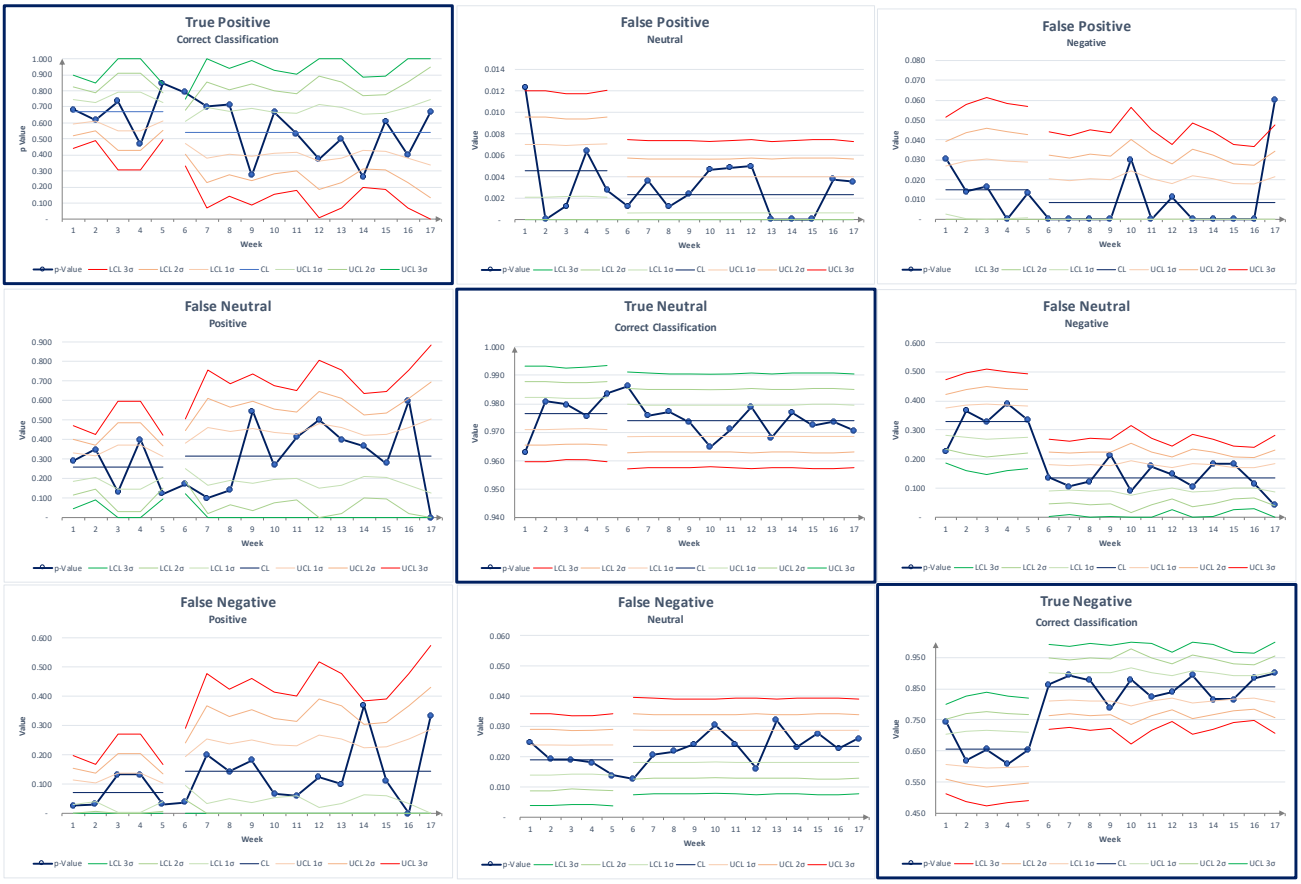

Figure 8. Control Charts for p-values, n = class. Week 17

Figure 8 depicts an example of nine control charts for corresponding *p*-values and various control limits for the first approach for the 17 weeks of observations. Each cell represents a single *p*-chart that, in turn, reflects the change in corresponding portions of classification instances (*p*-value) and the change in control limits on a week-by-week basis. The average value (the Central Line or CL), as well as linked control limits (Upper and Lower Control Limits or UCL and LCL accordingly), may drift with the introduction of a new validation sample and confusion matrix built on such a sample.

The upper and lower levels were limited by 1 and 0 accordingly. For example, if the upper level was supposed to be greater than 1, the limit was set to 1, and if the lower level was

supposed to be negative, the limit was set to 0 to maintain the common sense of a probability concept.

Red zones on charts located in the lower parts of cells 1, 5, and 9 designate undesirable areas for *p*-value drifts. The same is true for the red zones located in the upper parts of cells 2, 3, 4, 6, 7, and 8. Red zones represent deterioration, while the opposite represents an improvement. Hence, the drift of *p*-values into red zones may contain signals for the modeler and are the primary interest of the current research.

Initially, the SVC algorithm was trained on a random training set sampled from 5,001 hand-labeled tweets. The testing set represented a subset of 865 tweets and was a basis for calculating a confusion matrix named SVC Performance [Initial] in the Appendix 3, Figure 50.

865 testing set counts were arbitrarily chosen as a count for the validation set to sample each week from the weekly pool of tweets with predicted sentiment labels. The tweet counts by week can be found in Appendix 3, Figure 56.

Following the goal of the project to find whether SPC techniques are able to catch the possible drifts and shifts in the performance of the classification algorithm over time, the SVC was re-trained. One hundred eighty-four misclassified instances of the Positive and Negative class with correct labels were added to the overall pool of hand-labeled tweets, and the training process was repeated. The count of the testing set increased to 902. Validation sample counts for weeks 6 through 17 followed the size of the testing set and increased to 902 as well.

Hence, the *p*-charts reflect two algorithm behavior regimes presented by shifts in the Center Lines and Control Limits in all nine charts.

Results

The visual analysis of the Control Charts provides evidence of out-of-control performance according to Montgomery's common decision rules for detecting patterns [15].

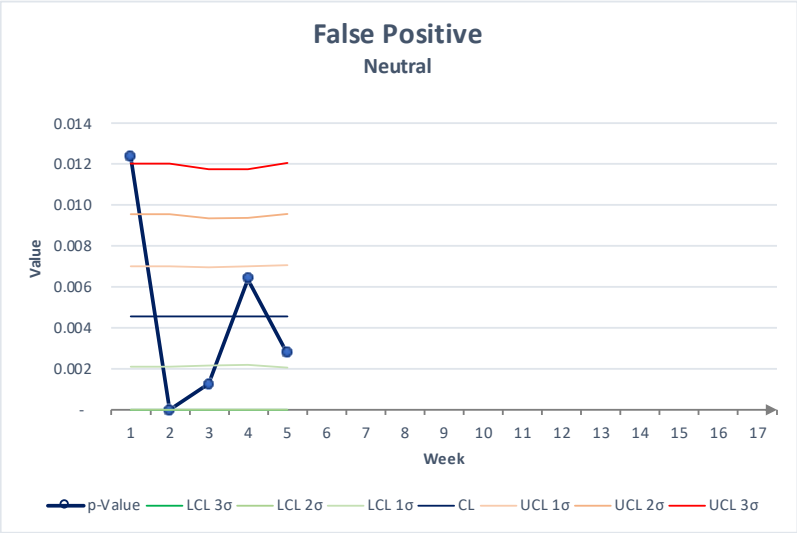

Figure 9. Cell 2 of the Control Charts for p-value for False Positive, n = class. Week 5

Figure 9 presents Cell 2 of the nine-cell Control Chart (Appendix 1, Figure 21) for the approach one for week 5 and signals exceeding three-sigma limit for misclassification errors in week one.

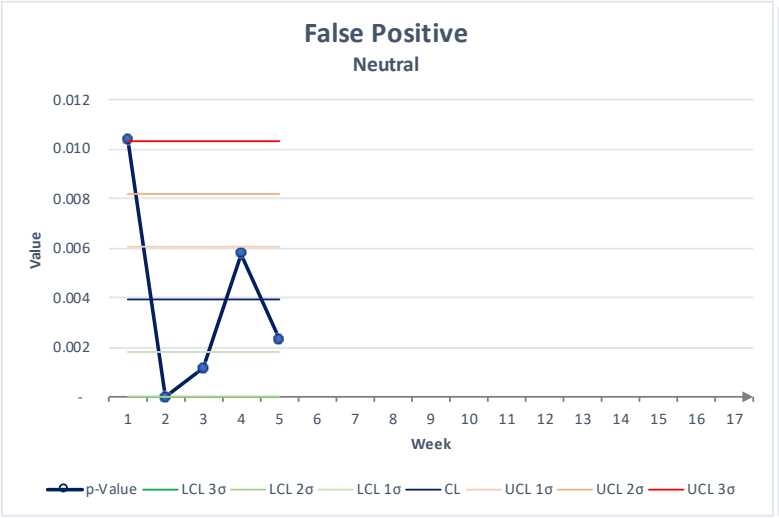

Figure 10. Cell 2 of the Control Chart for p-value for False Positive, n = sample. Week 5

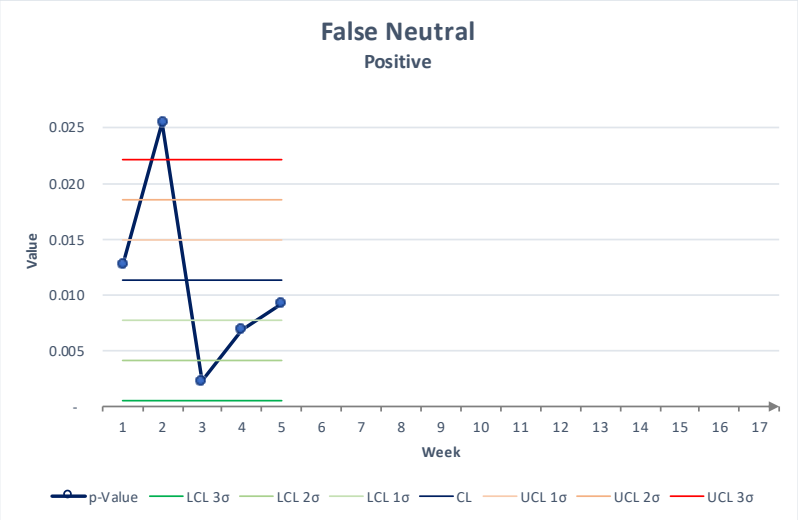

Figure 11. Cell 4 of the Control Chart for p-value for False Neutral, n = sample. Week 5

Figures 10 and 11 present Cell 2 and 4 of the nine-cell Control Chart (Appendix 2, Figure 37) of the second approach for week 5 and signals exceeding three-sigma limit for misclassification errors in weeks one and two.

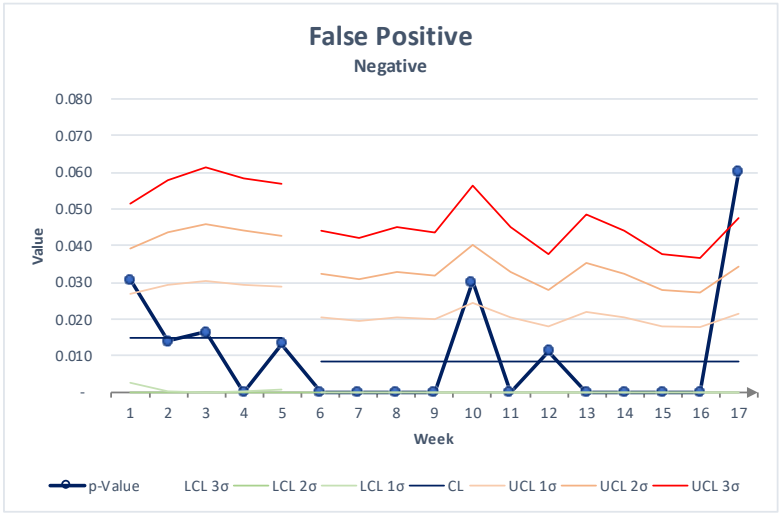

Figure 12. Cell 3 of the Control Chart for p-value for False Positive, n = class. Week 17

Figure 12 presents Cell 3 of the nine-cell Control Chart (Appendix 1, Figure 33) for the approach one for the week 17 and signals exceeding three-sigma limit for misclassification errors.

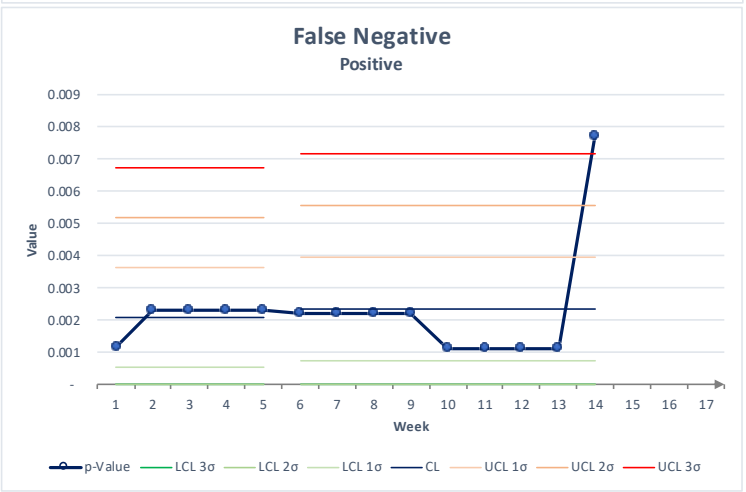

Figure 13. Cell 7 of the Control Chart for p-value for False Negative, n = sample. Week 14

Figure 13 presents Cell 7 of the nine-cell Control Chart (Appendix 2, Figure 46) for the second approach and signals exceeding three-sigma limit for misclassification errors in week 14.

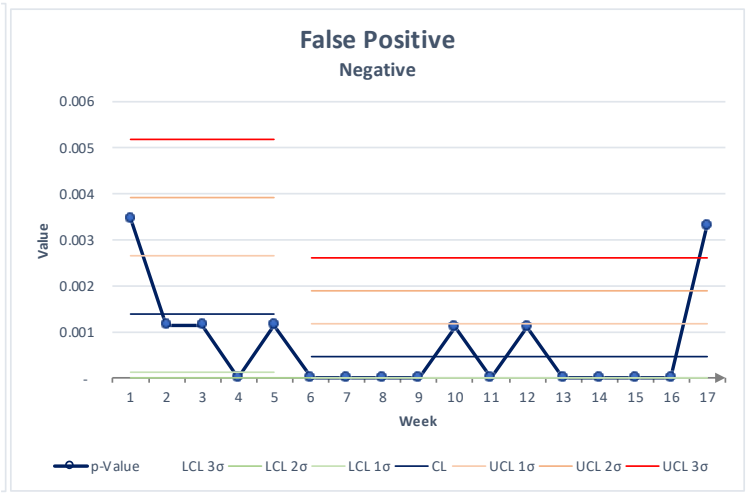

Figure 14. Cell 3 of the Control Chart for p-value for False Positive, n = sample. Week 17

Figure 14 presents Cell 3 of the nine-cell Control Chart (Appendix 2, Figure 49) for the second approach and signals exceeding three-sigma limit for misclassification errors in week 17.

The significant spikes in *p*-values shown in figures 9 to 14 are attributed to the rise in misclassification instances counts in corresponding weeks and types of errors. According to

Montgomery's common decision rules for detecting patterns, the algorithm behavior analysis in other weeks didn't show significant performance deviations in performance [15].

Overall, the second approach to calculating the critical limits where *n* equals the sample size appeared more sensitive in detecting exceeding of the critical limits. There are four clear signals in the second approach against two possible signals in the first approach.

The statistically significant decrease in the algorithm's performance during the seventeenth week was observed with the help of both approaches. The SPC technique signaled the outlier value of misclassification count when the algorithm classified an unusually large number of tweets as positive while, in reality, they were negative.

Discussion

The goal of the project was to find whether SPC techniques are able to catch the possible drifts and shifts in the performance of the classification algorithm in time. The crucial part of the project was to randomly sample tweets from the pool of tweets labeled by the algorithm, hand-label the real sentiment, and compare the classifications and their deviations with the confusion matrix. The procedure was repeated weekly, constantly replenishing the pool of data to analyze. Hence, the central line, upper and lower control limits drifted as its calculation was influenced by the introduction of a new observation every week. Such progress sometimes sent mixed signals. During a week, observations might lie in the territory signifying out-of-control performance, while the next week, the limits drifted, and the observation appeared within limits or vice versa. Such drifts were observed on the Control Charts for False Negative misclassification instances [classified as Negative but were Positive] when the *p*-value neared but didn't exceed the three-sigma control limits during weeks fourteen and fifteen. However, later, it surpassed (Figure 15) the three-sigma control limit retrospectively during week sixteen.

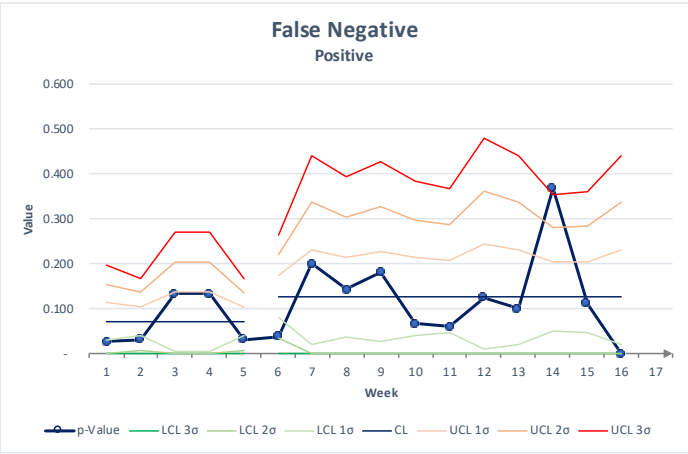

Figure 15. Cell 7 of the Control Chart for p-value for False Negative, n = class. Week 16

Appendix 1 contains the full set of nine-cell charts for approach one, where *n* equals the sum of all the counts within the class. Appendix 2 contains the full set of charts for approach two, that treats *n* as the total number of tweets in the sample. Both sets span the period from week two to week seventeen and reflect one-sigma (LCL 1σ and UCL 1σ), two-sigma (LCL 2σ and UCL 2σ), and three-sigma (LCL 3σ and UCL 3σ) control limits around the mean value depicted on charts together. Appendix 3 contains confusion matrices, linear trend graphs, correlation matrix, examples of misclassified tweets, and dataset information.

The visual analysis of control charts suggests that the algorithm retrained on week six showed slight improvements in the performance. The mean values for misclassification counts became lower while the mean for correct classification counts increased, although not for every indicator.

The more conventional classification algorithm performance metrics did not signal deterioration at all or showed too many signs that resemble noise rather than signals. Micro F1 and Weighted F1 (Figure 16), as well as F1 Neutral Class (individual class metric) are almost flat through all seventeen weeks of observations. Macro F1 (ranged from 0.67 to 0.88), F1 Negative Class (ranged from 0.65 to 0.88), and F1 Positive Class (ranged from 0.36 to 0.87) are too volatile and showed significant decreases in values in weeks four, nine, twelve, and sixteen.

The visual analysis of tweets that were subject to the validation process suggested that significant deterioration of the algorithm's performance aligned more with signals coming from *p*-charts rather than from conventional aggregate metrics such as F1-scores. Figure 54

(Appendix 3) shows the example of tweets that bear Positive sentiment but were classified by the algorithm as Negative during week seventeen.

Figure 16 depicts the evolution of the popular aggregated accuracy metrics in time with linear trends. The trend lines show performance deterioration in individual F1 Positive Class and overall Macro F1 score. Micro F1 and Weighted F1 do not provide valuable information as they are substantially influenced and overweighed by Neutral Class.

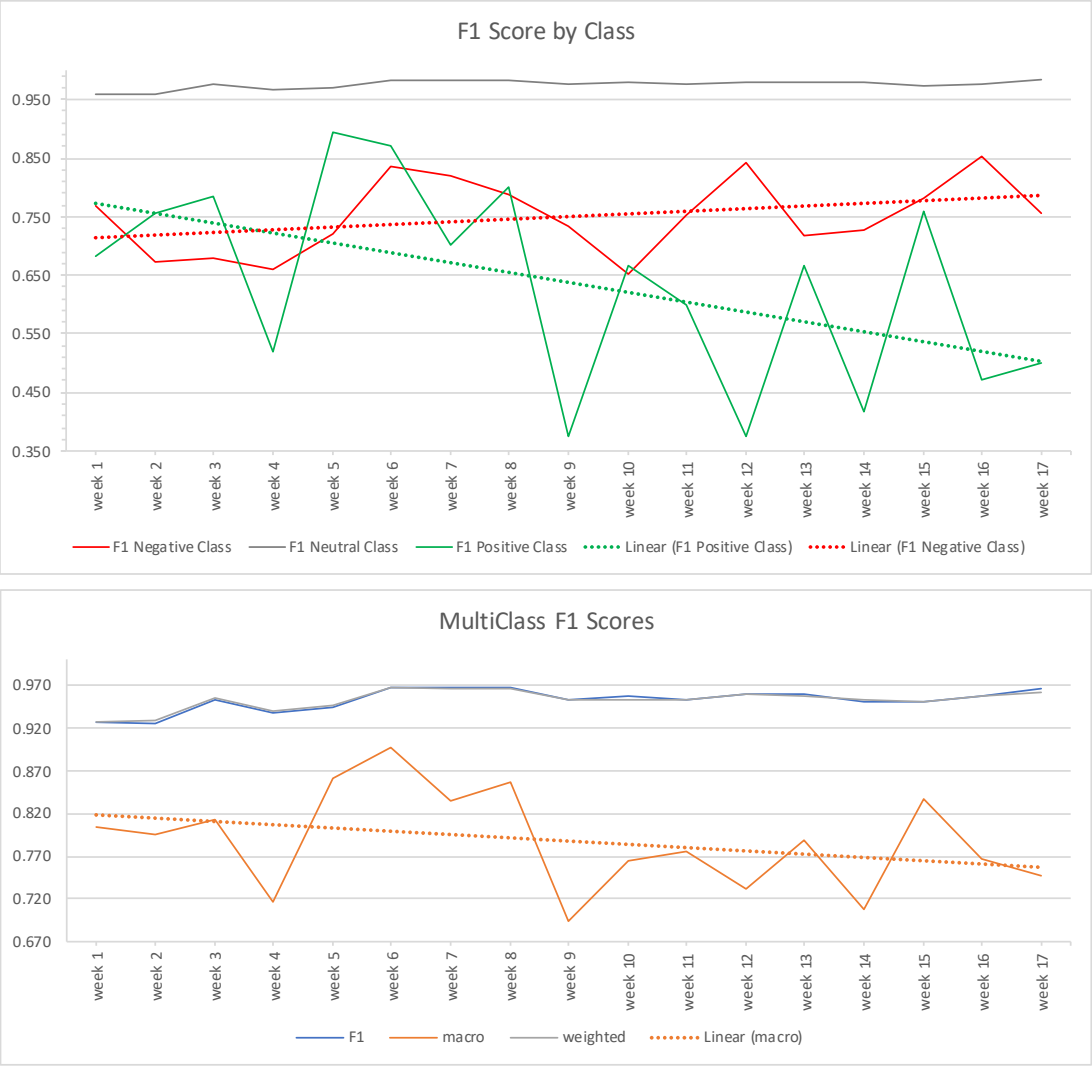

Figure 16. Time-series values of F1 score metrics.

The weakness of F1-score metrics lies in its nature of aggregating the results. As mentioned previously in the document, the calculation process either ignores the uneven distribution of classes, or treats classes evenly.

The Twitter feed generates a significant number of tweets every day, most of which can be classified as neutral as users tend to publish generic informational messages more often. Figure 17 shows predicted sentiment counts of the target dataset and the distribution of tweets with positive, negative, and neutral sentiment towards the company of interest, Apple, Inc.

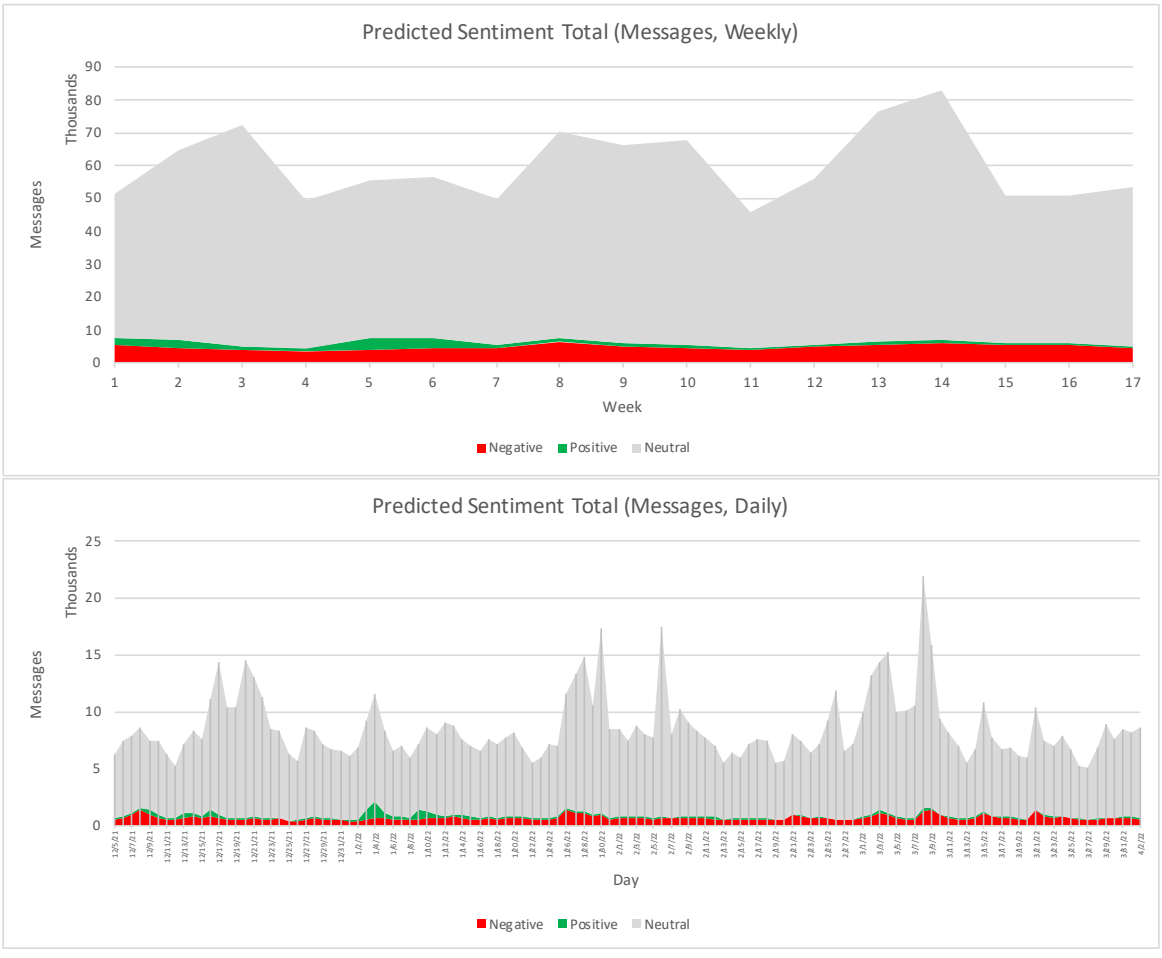

Figure 17. The time-series distribution of the predicted sentiment of the target tweets.

The distribution of tweets is significantly skewed towards the neutral class, which accounts for 90% of total messages. Negative sentiment accounts for 8%, leaving a minuscule 2% for the positive class. Such skewness presents a challenge in detecting signals of performance deterioration and separating signals from noise.

The correlation matrix depicted in Figure 53 (Appendix 3) shows the pairwise correlation coefficients between changes in *p*-values of both approaches, individual class F1-scores, aggregated F1-scores and changes in amount of positive, neutral and negative tweets.

The second approach of treating *n*, shows heightened correlations between the changes in *p*-values and counts of positive, neutral and negative tweets. Also, there are high correlation coefficients between individual class F1-scores and aggregated metrics, where F1 Neutral Class is highly correlated with Micro F1 and Weighted F1, and F1 Positive Class is highly correlated with Macro F1. The change in the number of positive tweets is correlated with F1 Positive Class (0.50) and Macro F1 (0.54) making all such metrics unreliable in terms of quality control. The high correlation coefficient does not imply any causal relationship between variables but such co-movements make the metrics biased.

The further analysis of the pairwise correlation between total counts of tweets by classes and various *p*-values suggests that a mix of *p*-value metrics based on different *n*-values may be a better proxy for monitoring the ongoing performance of the classification algorithm.

Figures 51 and 52 (Appendix 3) show the linear trend lines that run through every *p*-value on the control chart beginning with week 6. Such trends should be treated with care and cannot be a basis for decision-making due to their nature, but rather can substitute the granular analysis of *p*-charts.

The primary stake of the project was not the supervised ML model per se but rather performance monitoring during the deployment and exploitation phase. The project demonstrated the viability of the concept of employing SPC techniques in classification algorithm performance monitoring on the seventeen weeks of observation data.

The signal derived from the control charts can signal the need for recalibration or other fine-tuning technique to improve performance on a timely manner thus decreasing the overall maintenance costs by pointing to when it is really justified and when it is not.

Shifts in negative and positive sentiment towards the entity of interest are of great interest to decision-makers as it shapes such companies' policies, products, and services. Hence, it is essential to separate and disregard temporary noise from the fundamental shifts in public sentiment and from the deterioration in the performance of algorithms that help to track such sentiment per se.

References

[1] B. Shankar, “Reading financial news: The top 10 avoidable distractions,” *CFA Institute Enterprising Investor*, 01-Apr-2021. [Online]. Available: https://blogs.cfainstitute.org/investor/2020/08/04/reading-financial-news-the-top-10-avoidable-distractions/. [Accessed: 08-May-2022].

[2] S. D'Alonzo and M. Tegmark, “Machine-Learning Media Bias,” *arXiv.org*, 31-Aug-2021. [Online]. Available: https://arxiv.org/abs/2109.00024. [Accessed: 08-May-2022].

[3] “Journalism,” *Wikipedia*, 26-Apr-2022. [Online]. Available: https://en.wikipedia.org/wiki/Journalism. [Accessed: 08-May-2022].

[4] J. Rout, K.Choo,A. Dash, S. Bakshi, S. Jena, K. Williams A model for sentiment and emotion analysis of unstructured social media text. *http://dx.doi.org.liblink.uncw.edu/10.1007/s10660-017-9257- 8*. Electronic Commerce Research, 18(1), 181-199. 2018.

[5] Y. Ren, R. Wang, D. Ji A topic-enhanced word embedding for Twitter sentiment classification. *https://doi.org/10.1016/j.ins.2016.06.040*. Information Sciences, 369, 188-198. 2016.

[6] B. Pang, L. Lee, S. Vaithyanathan Thumbs up? sentiment classification using machine learning techniques. *https://doi.org/10.3115/1118693.1118704*. Proceedings of the Conference on Empirical Methods in Natural Language Processing, 79–86. 2002.

[7] R. Ong Offensive Language Analysis using Deep Learning Architecture. *https://arxiv.org/pdf/1903.05280.pdf*. ArXiv. 2019.

[8] O. Appel, F. Chiclana, J. Carter, H. Fujita Successes and challenges in developing a hybrid approach to sentiment analysis. *http://dx.doi.org.liblink.uncw.edu/10.1007/s10489-017-0966-4*. Applied Intelligence, 48(5), 1176-1188. 2018.

[9] M. Ghosh, G. Sanyal Performance assessment of multiple classifiers based on ensemble feature selection scheme for sentiment analysis. *http://dx.doi.org.liblink.uncw.edu/10.1155/2018/8909357*. Applied Computational Intelligence and Soft Computing, 12. 2018.

[10] N. Zainuddin, A. Selamat, R. Ibrahim Hybrid sentiment classification on twitter aspect-based sentiment analysis. *http://dx.doi.org.liblink.uncw.edu/10.1007/s10489-017-1098-6*. Applied Intelligence, 48(5), 1218-1232. 2018.

[11] D. Stojanovski, G. Strezoski, G. Madjarov, I. Dimitrovski, I. Chorbev Deep neural network architecture for sentiment analysis and emotion identification of Twitter messages. *https://doi.org/10.1007/s11042- 018-6168-1*. Multimed Tools Appl77, 32213–32242. 2018.

[12] Y. Kim, J. Kim, W. Kim, J. Im, T. Kim, S. Kang, C. Kim Predicting Fluctuations in Cryptocurrency Transactions Based on User Comments and Replies. *https://doi.org/10.1371/journal.pone.0161197*. PLoS ONE 11(8): e0161197. 2016.

[13] A. Abbasi, H. Chen, A. Salem Sentiment analysis in multiple languages: Feature selection for opinion classification in Web forums. *http://doi.acm.org/10.1145/1361684.1361685* . Inform. Syst. 26, 3. 2008.

[14] "Marketpsych," *MarketPsych*. [Online]. Available: https://www.marketpsych.com/. [Accessed: 08-May-2022].

[15] D. Montgomery, Introduction to Statistical Quality Control. 978-1-118-14681-1. Wiley 2013

[16] D. Broockman, & J. Kalla (2022, April 1). The manifold effects of partisan media on viewers' beliefs and attitudes: A field experiment with Fox News viewers. https://doi.org/10.31219/osf.io/jrw26

[17] "Semeval," *SemEval*. [Online]. Available: https://semeval.github.io/. [Accessed: 08-May-2022].

[18] C. Weber, Syed S. Interdisciplinary optimism? Sentiment analysis of Twitter data. *http://doi.org/10.1098/rsos.190473*. The Royal Society. open sci.6:190473. 2019.

[19] J. Brownlee, "What is a confusion matrix in machine learning," *Machine Learning Mastery*, 14-Aug-2020. [Online]. Available: https://machinelearningmastery.com/confusion-matrix-machine-learning/. [Accessed: 08-May-2022].

[20] K. K. Singh, M. Elhoseny, A. Singh, and A. A. Elngar, in *Machine learning and the internet of medical things in Healthcare*, Amsterdam: Academic Press, 2021, pp. 89–111.

[21] S. Lo, Web service quality control based on text mining using support vector machine, Expert Systems with Applications, Volume 34, Issue 1, 2008, Pages 603-610, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2006.09.026.

[22] T. Ashton, N. Evangelopoulos, & V.R. Prybutok, Quantitative quality control from qualitative data: control charts with latent semantic analysis. *Qual Quant* **49,** 1081–1099 (2015). https://doi.org/10.1007/s11135-014-0036-5

[23] M. Grandini, & E. Bagli, G. Visani, Metrics for Multi-Class Classification: an Overview (2020), https://arxiv.org/abs/2008.05756

Appendix 1

The first set of Control Charts (p-charts), based on *n* equals the sum of all the counts within the class.

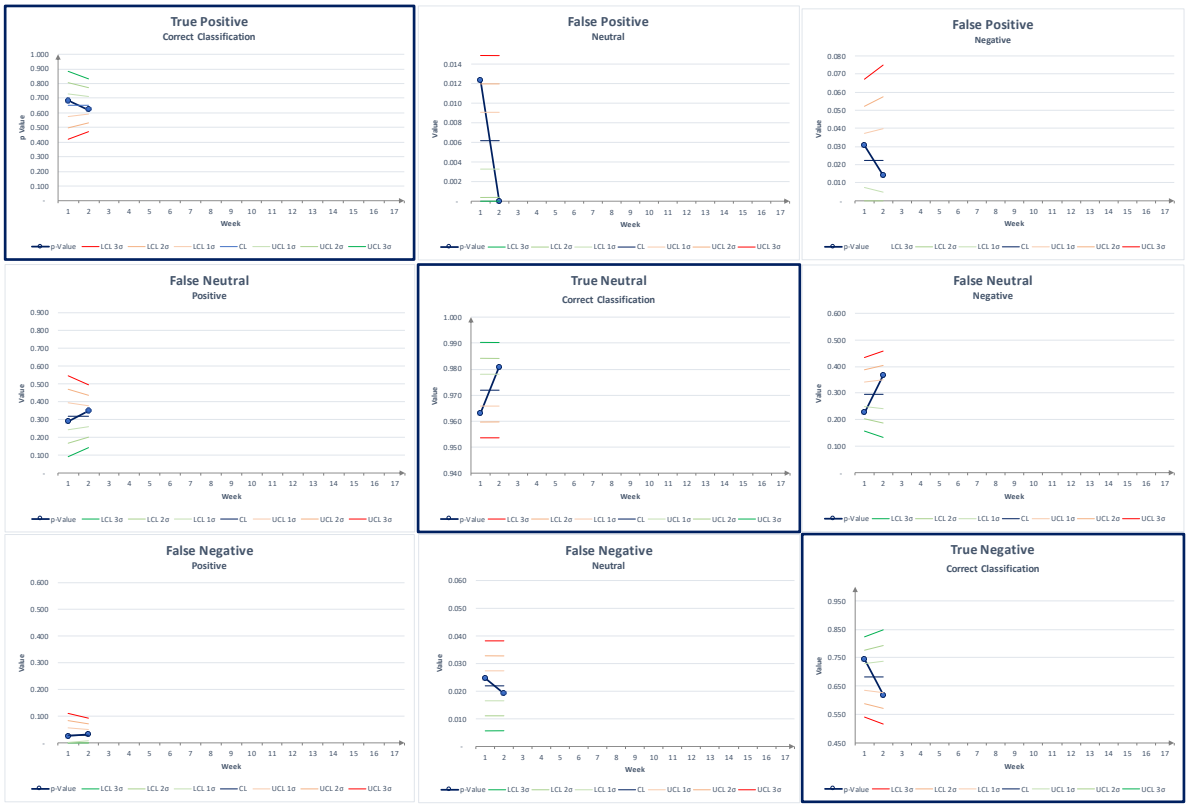

Figure 18. Control Charts for p-values, n = class. Week 2

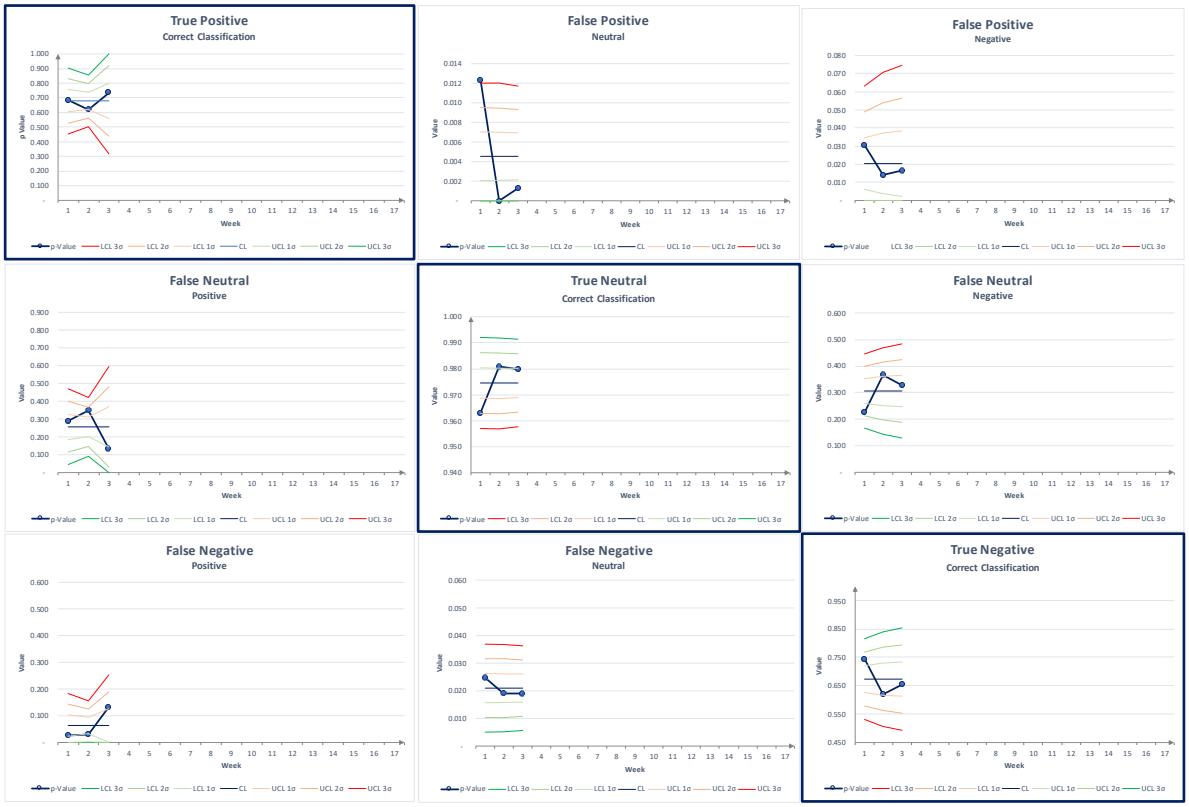

Figure 19. Control Charts for p-values, n = class. Week 3

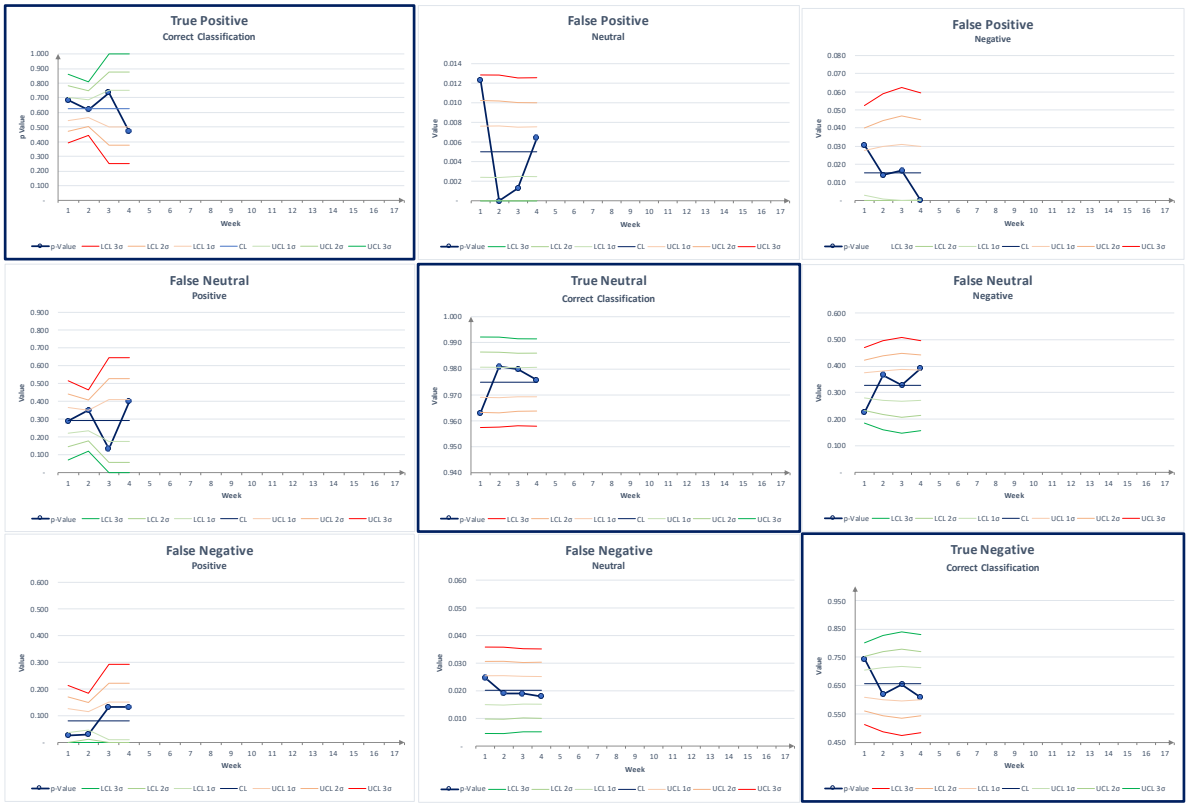

Figure 20. Control Charts for p-values, n = class. Week 4

Figure 21. Control Charts for p-values, n = class. Week 5

Figure 22. Control Charts for p-values, n = class. Week 6

Figure 23. Control Charts for p-values, n = class. Week 7

Figure 24. Control Charts for p-values, n = class. Week 8

Figure 25. Control Charts for p-values, n = class. Week 9

Figure 26. Control Charts for p-values, n = class. Week 10

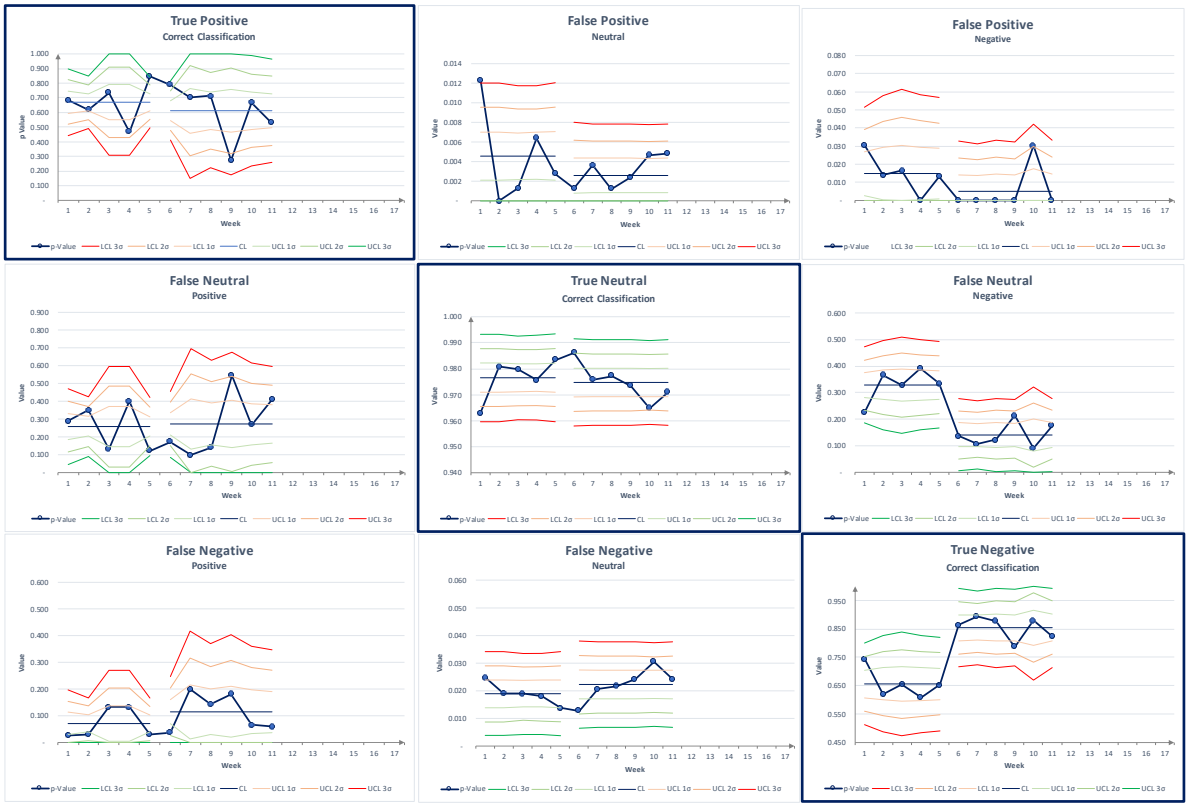

Figure 27. Control Charts for p-values, n = class. Week 11

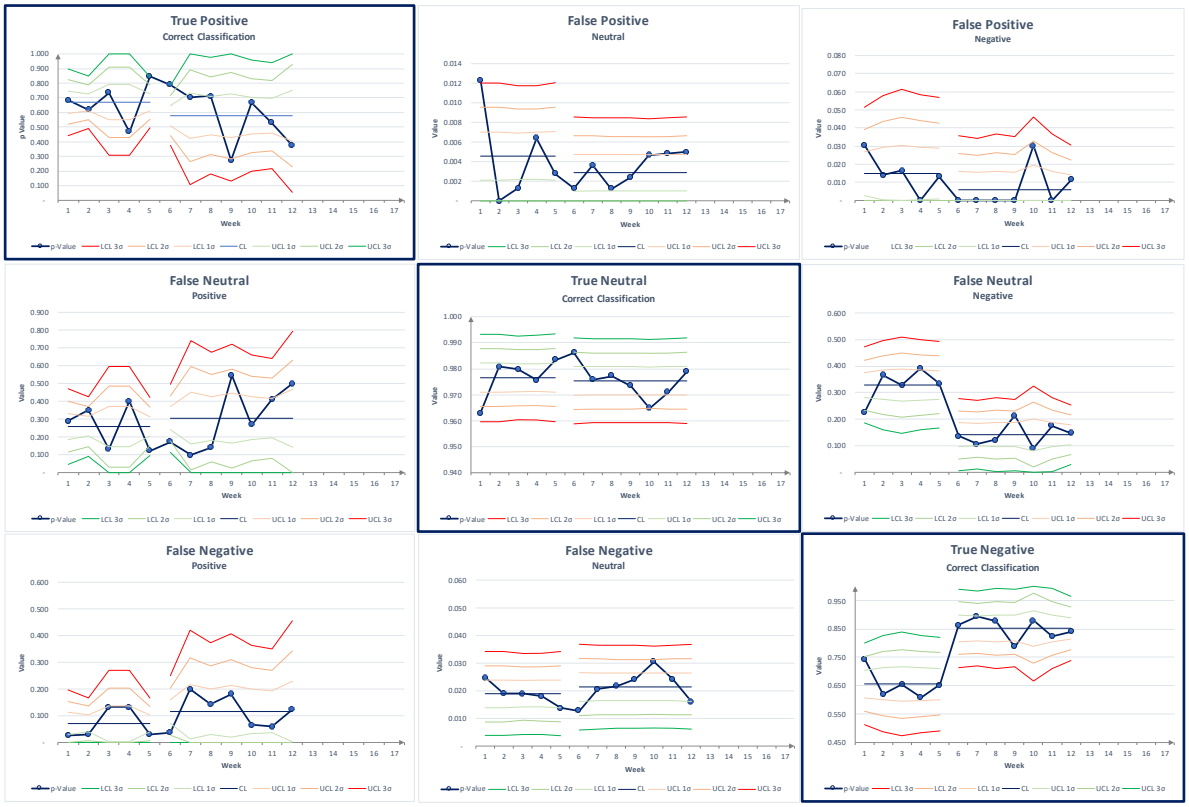

Figure 28. Control Charts for p-values, n = class. Week 12

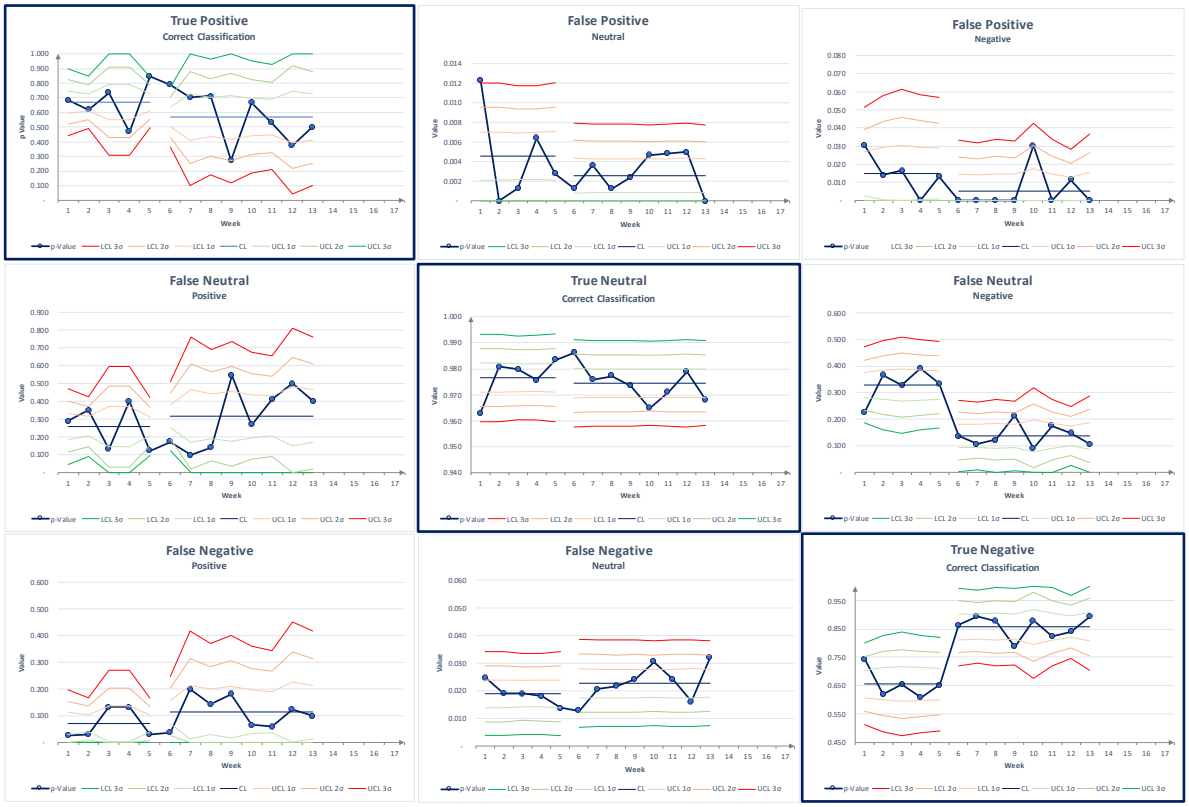

Figure 29. Control Charts for p-values, n = class. Week 13

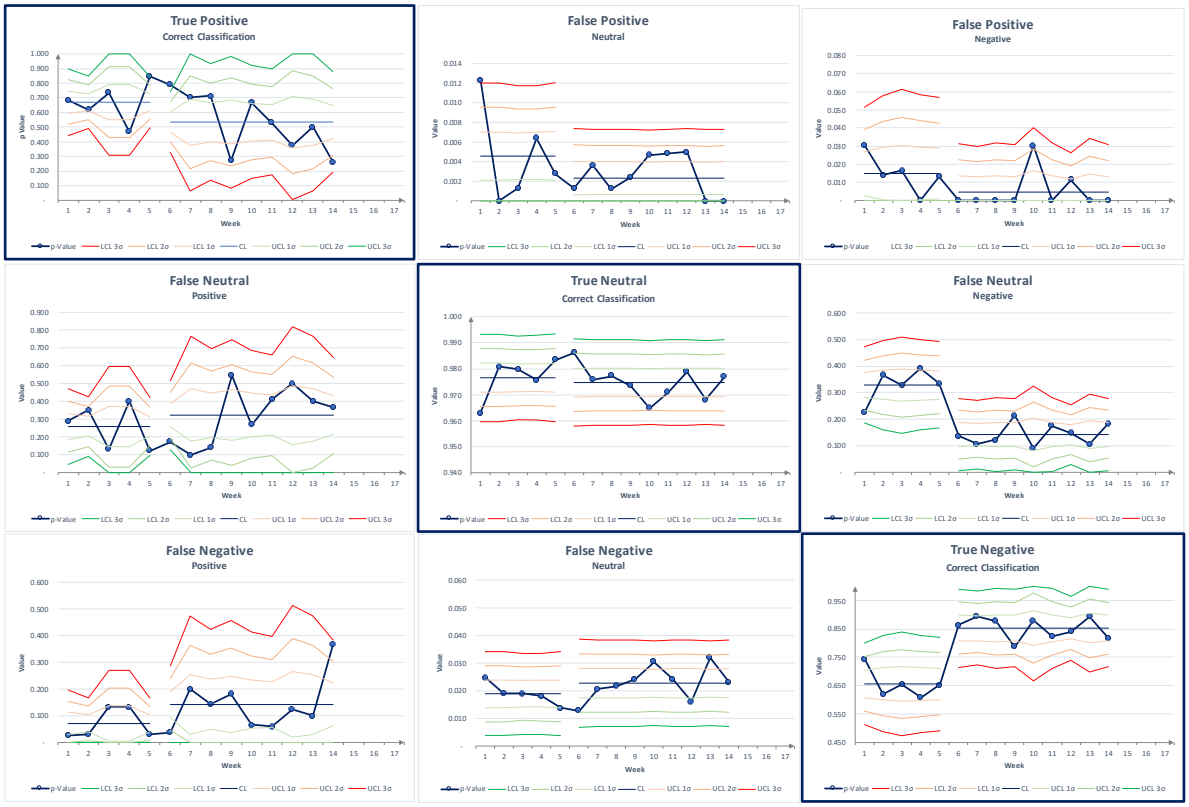

Figure 30. Control Charts for p-values, n = class. Week 14

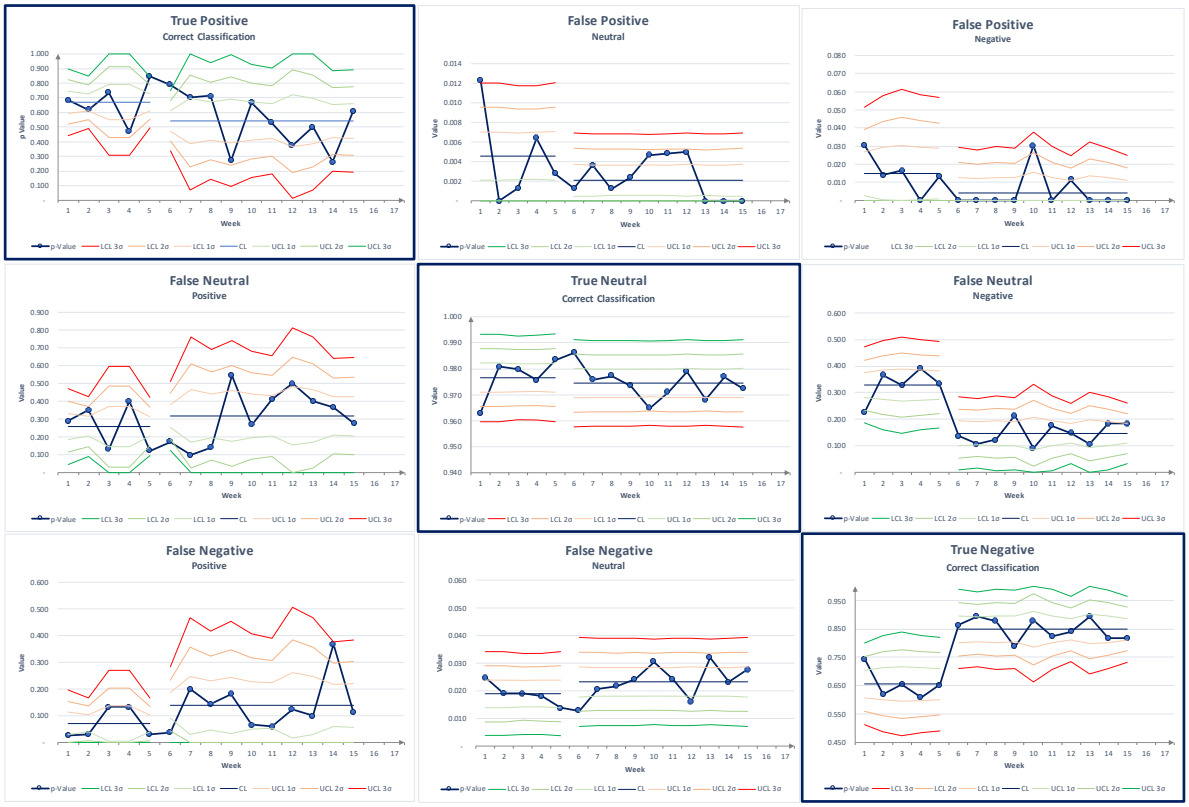

Figure 31. Control Charts for p-values, n = class. Week 15

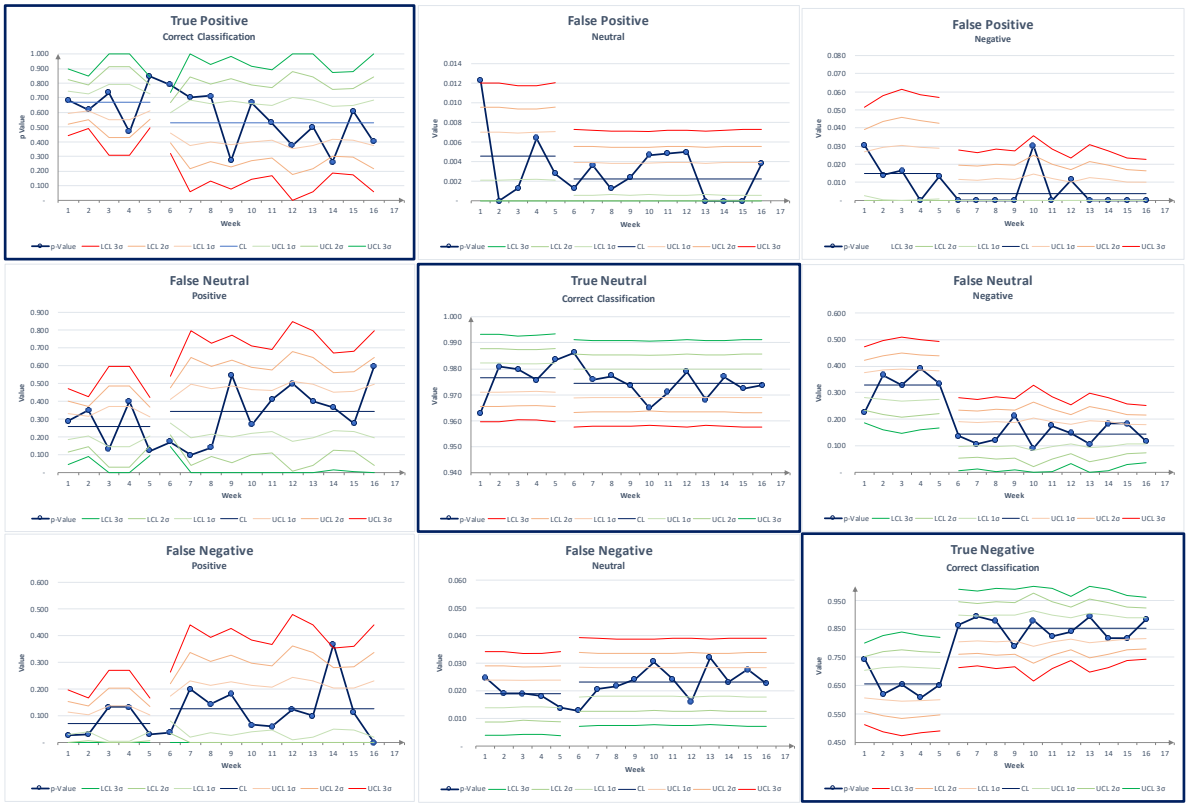

Figure 32. Control Charts for p-values, n = class. Week 16

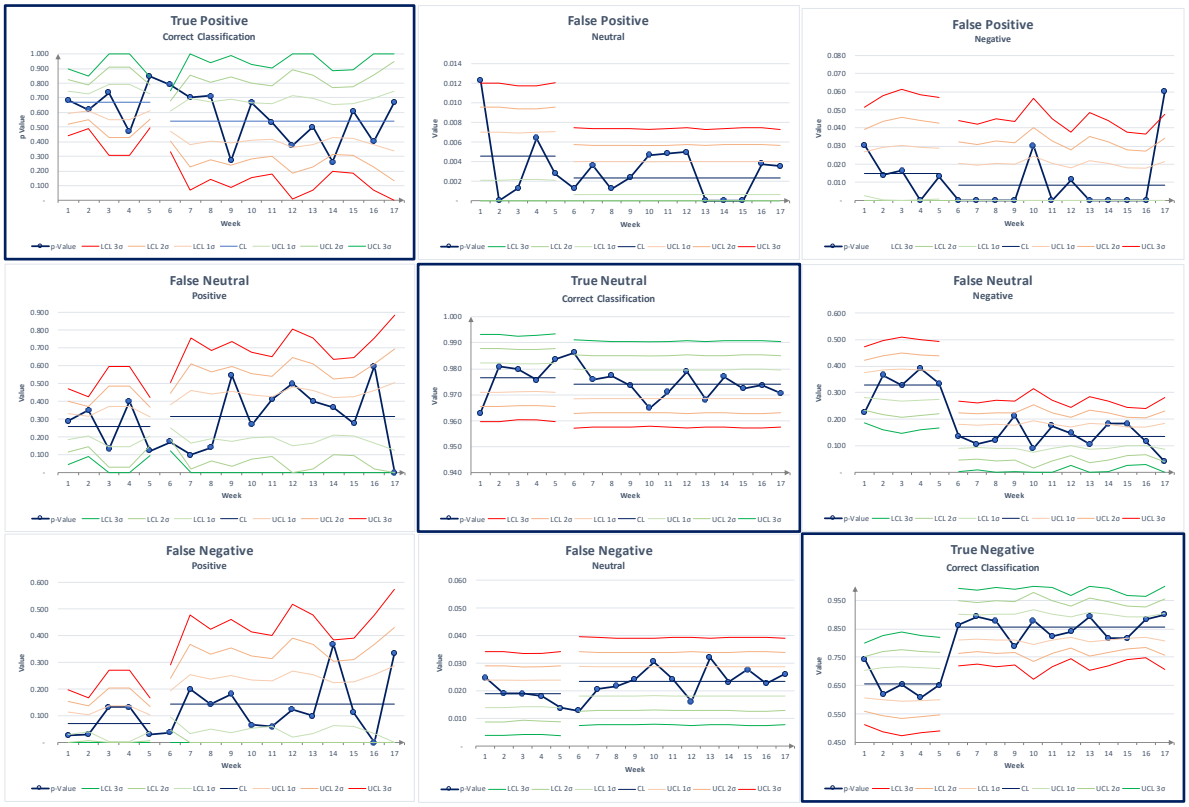

Figure 33. Control Charts for p-values, n = class. Week 17

Appendix 2

The second set of Control Charts (p-charts), are based on *n* equals the sample size. The shaded areas represent cells for *p*-values that, by nature, reflect the probability of getting a tweet with positive [positive, neutral, or negative] sentiment against other correct and incorrect instances at the point in time. The observation outside the control limits signifies a substantial shift in public perception rather than a shift in the quality of the classification process, given that corresponding errors are constant. Thus, the analysis of such cells was limited by the researcher.

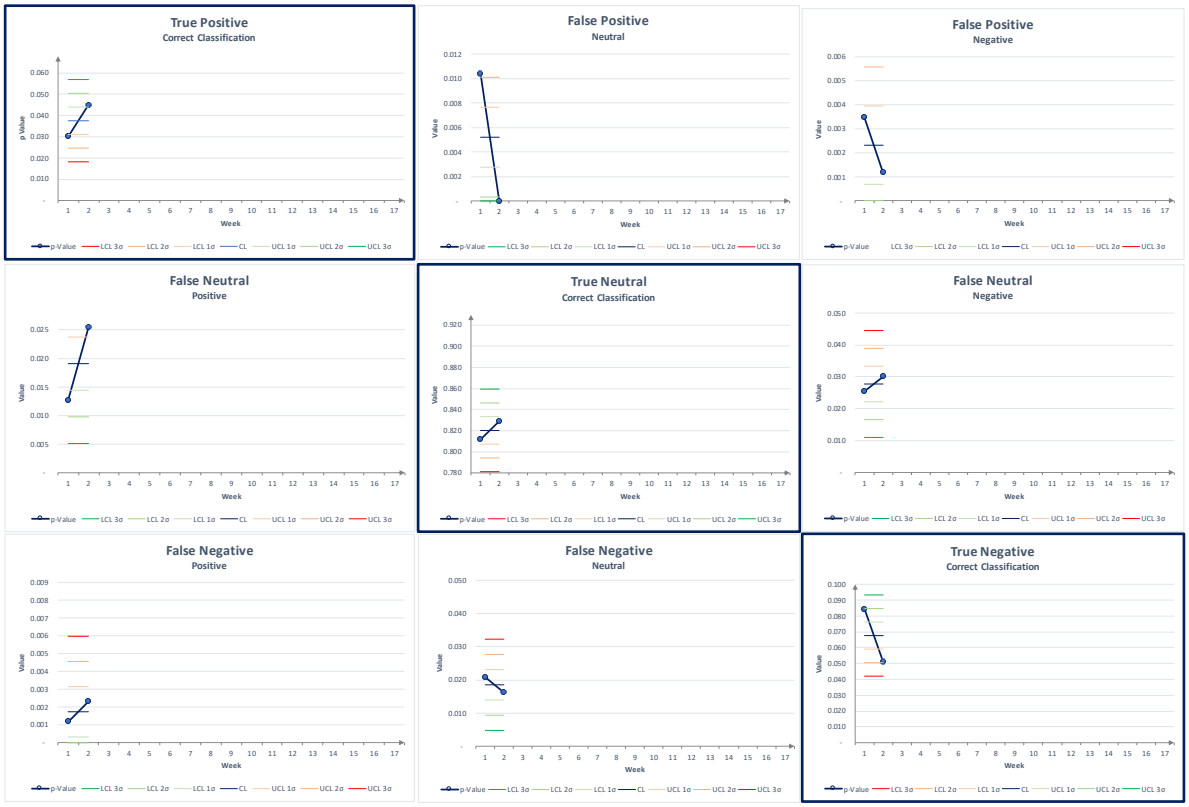

Figure 34. Control Charts for p-values, n = sample. Week 2

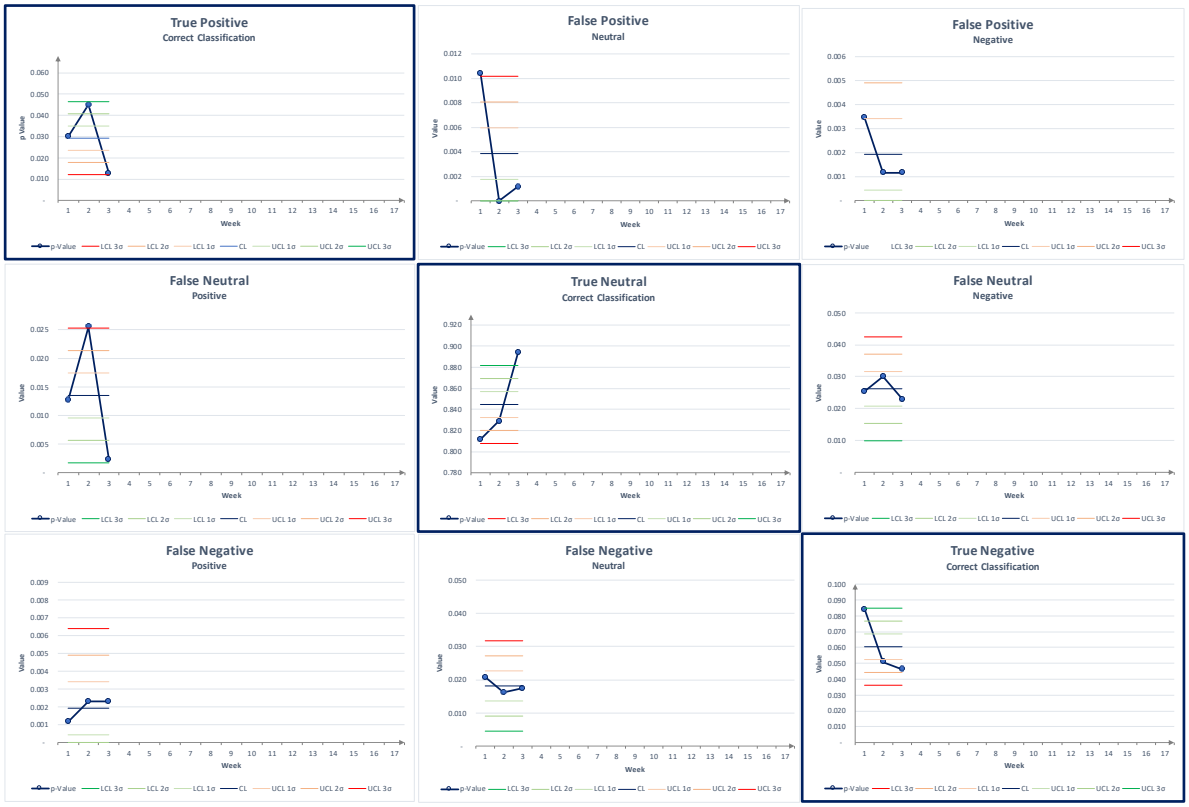

Figure 35. Control Charts for p-values, n = sample. Week 3

Figure 36. Control Charts for p-values, n = sample. Week 4

Figure 37. Control Charts for p-values, n = sample. Week 5

Figure 38. Control Charts for p-values, n = sample. Week 6

Figure 39. Control Charts for p-values, n = sample. Week 7

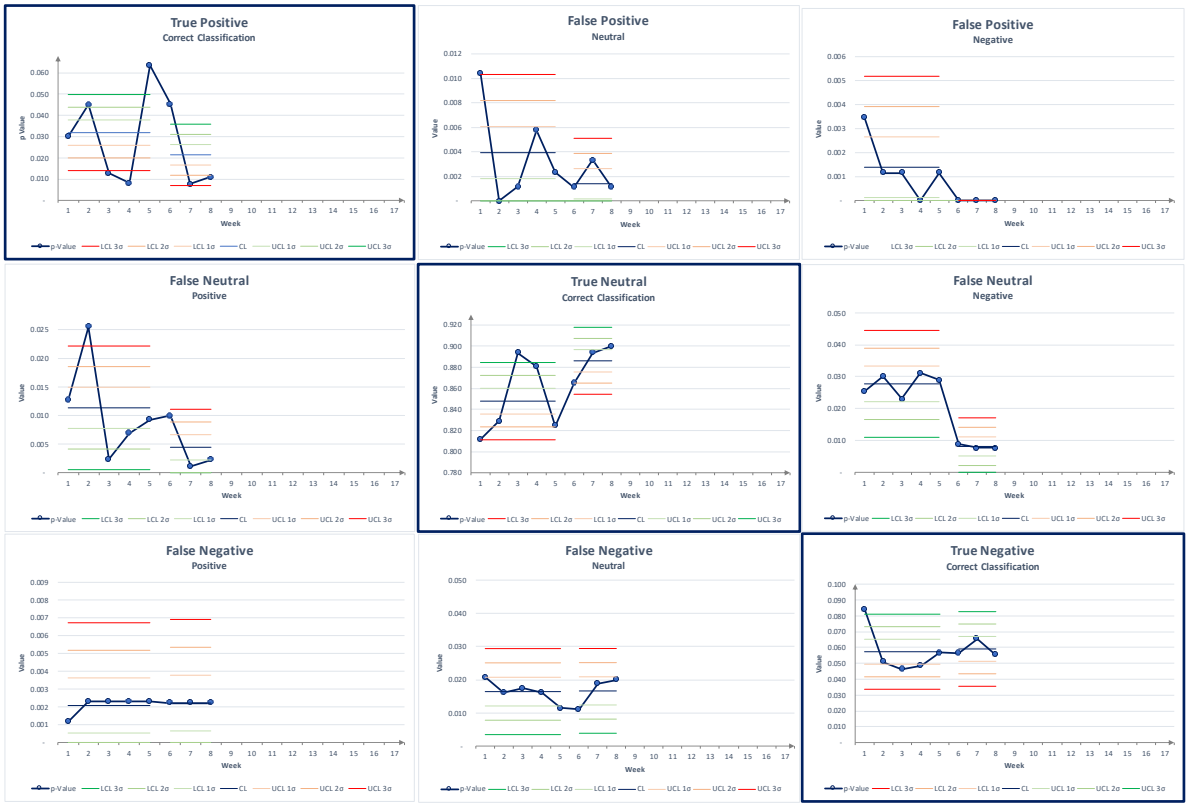

Figure 40. Control Charts for p-values, n = sample. Week 8

Figure 41. Control Charts for p-values, n = sample. Week 9

Figure 42. Control Charts for p-values, n = sample. Week 10

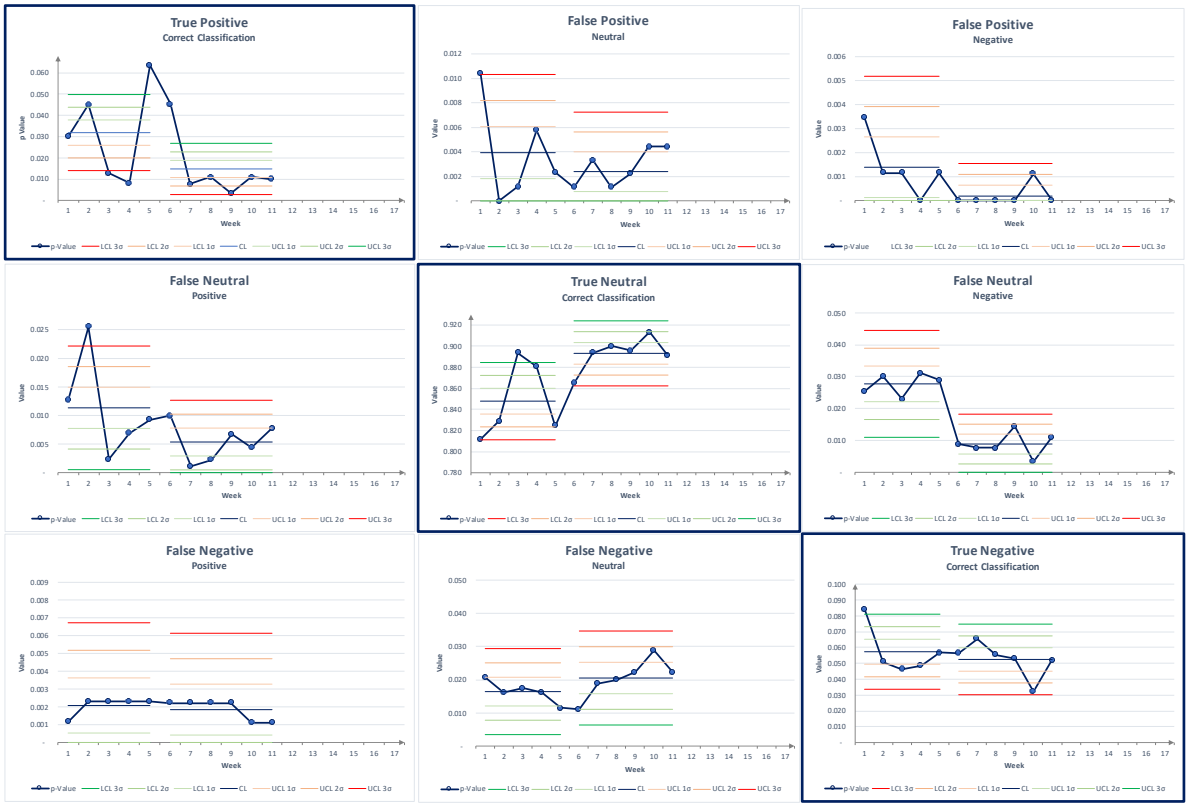

Figure 43. Control Charts for p-values, n = sample. Week 11

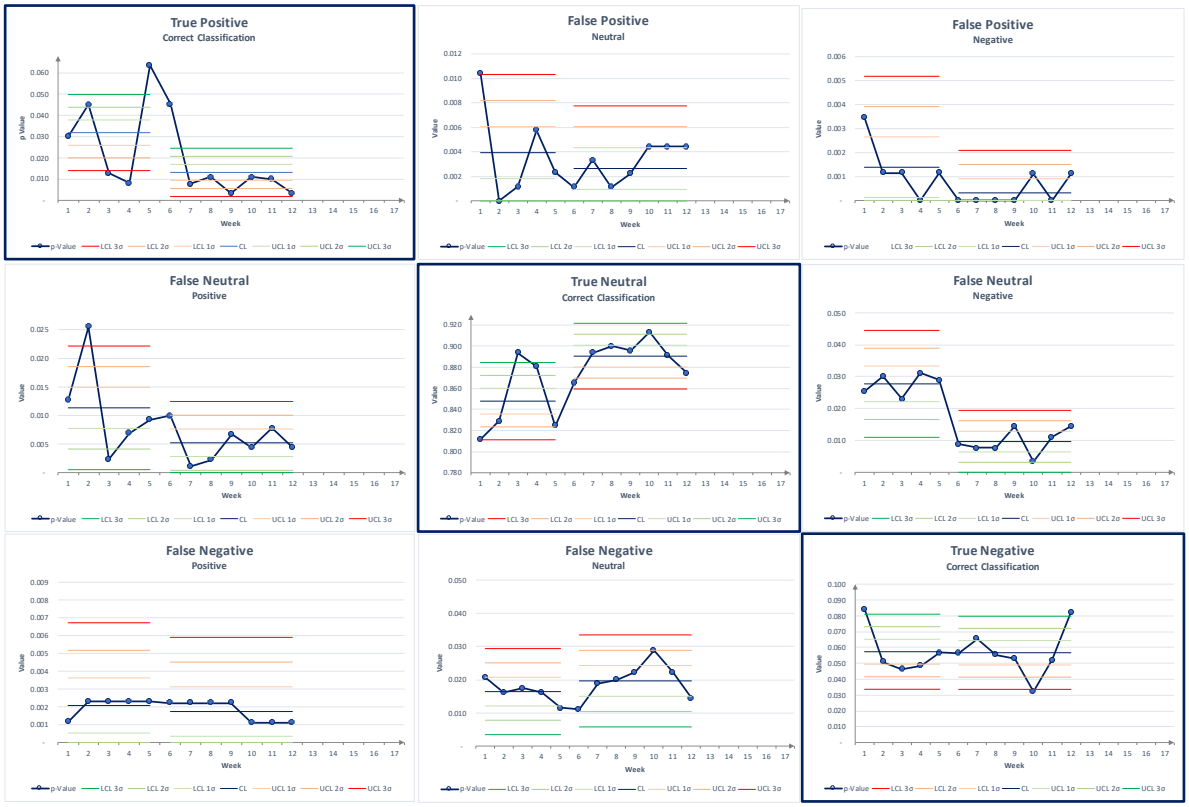

Figure 44. Control Charts for p-values, n = sample. Week 12

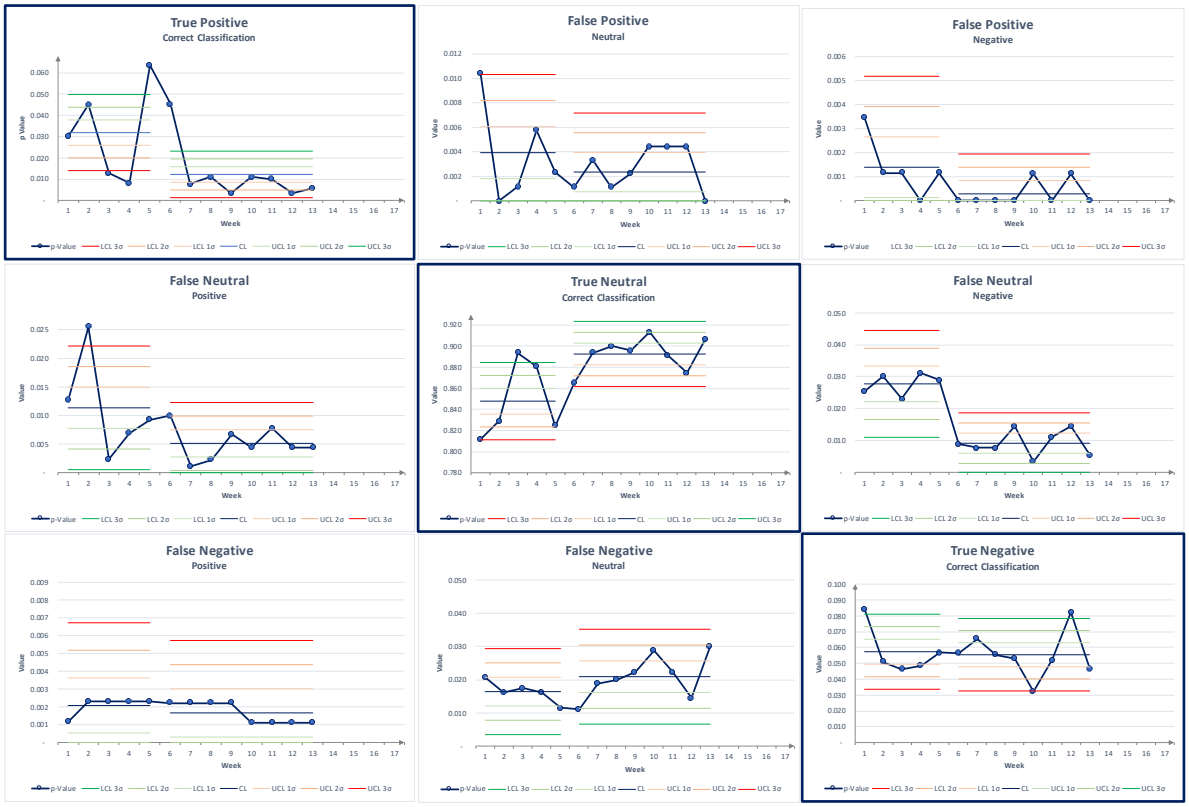

Figure 45. Control Charts for p-values, n = sample. Week 13

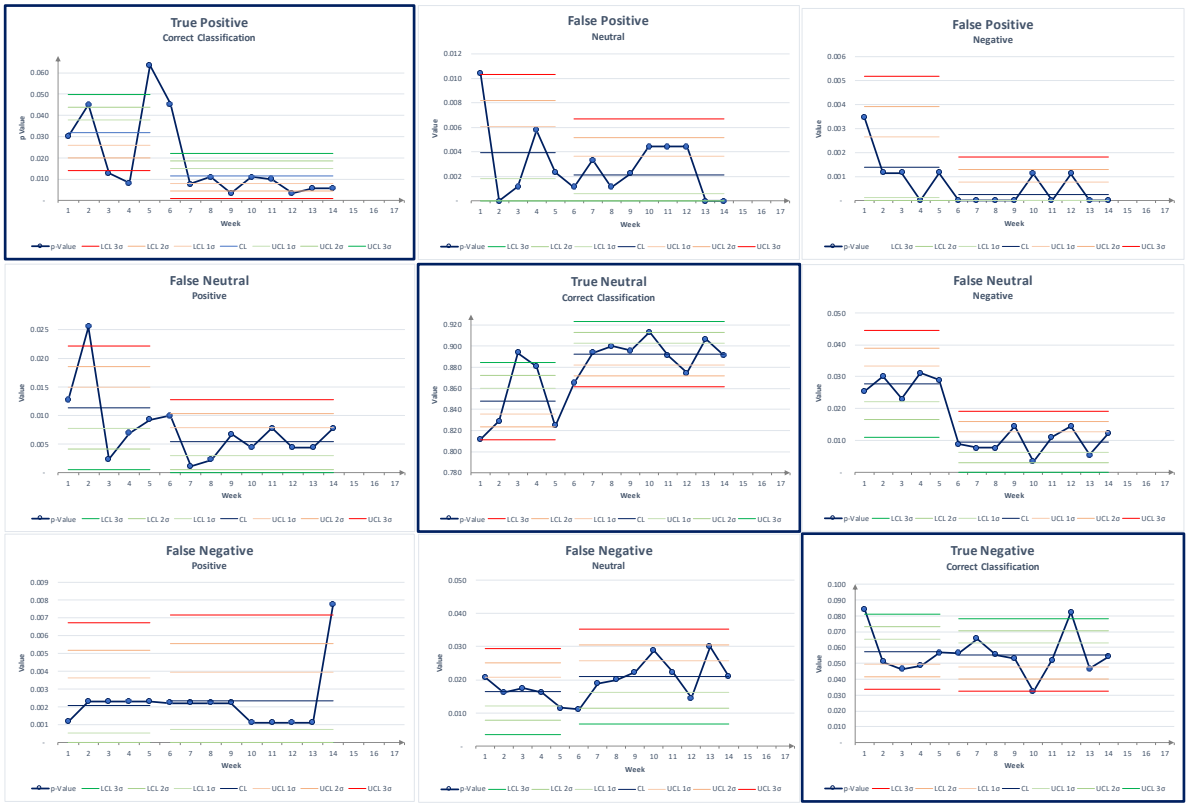

Figure 46. Control Charts for p-values, n = sample. Week 14

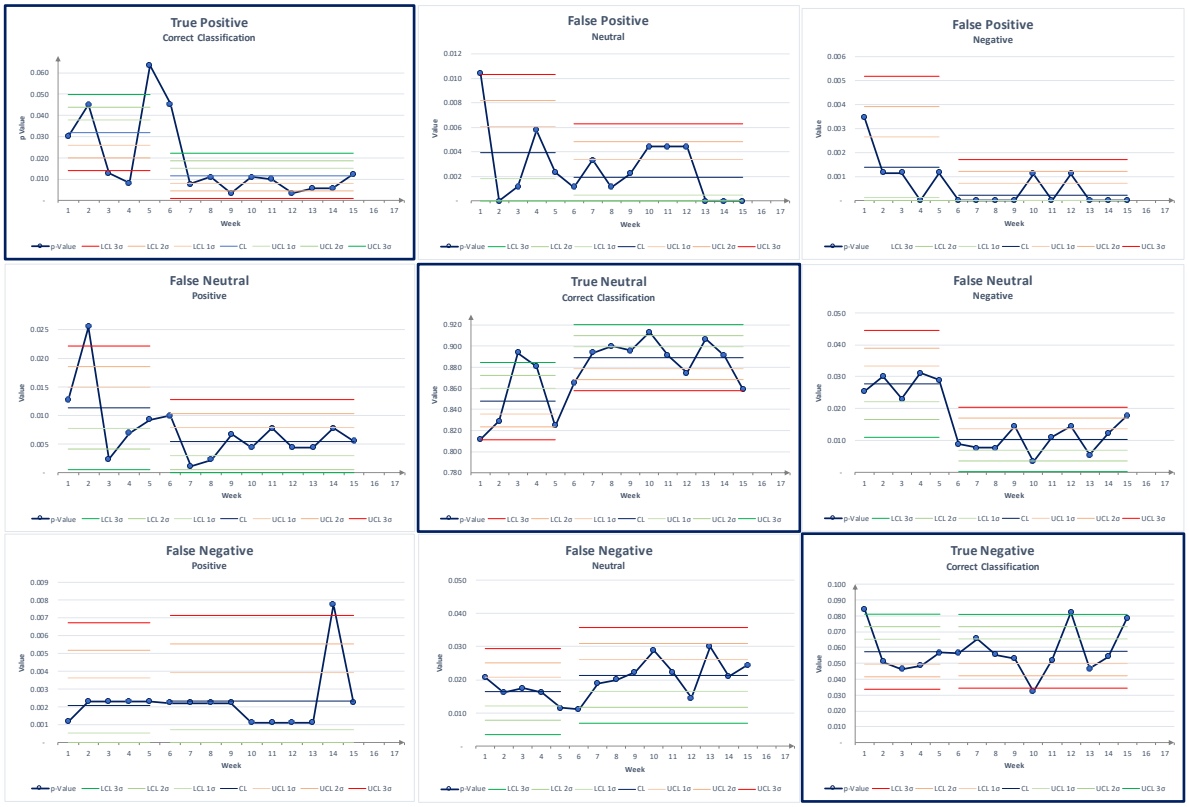

Figure 37. Control Charts for p-values, n = sample. Week 15

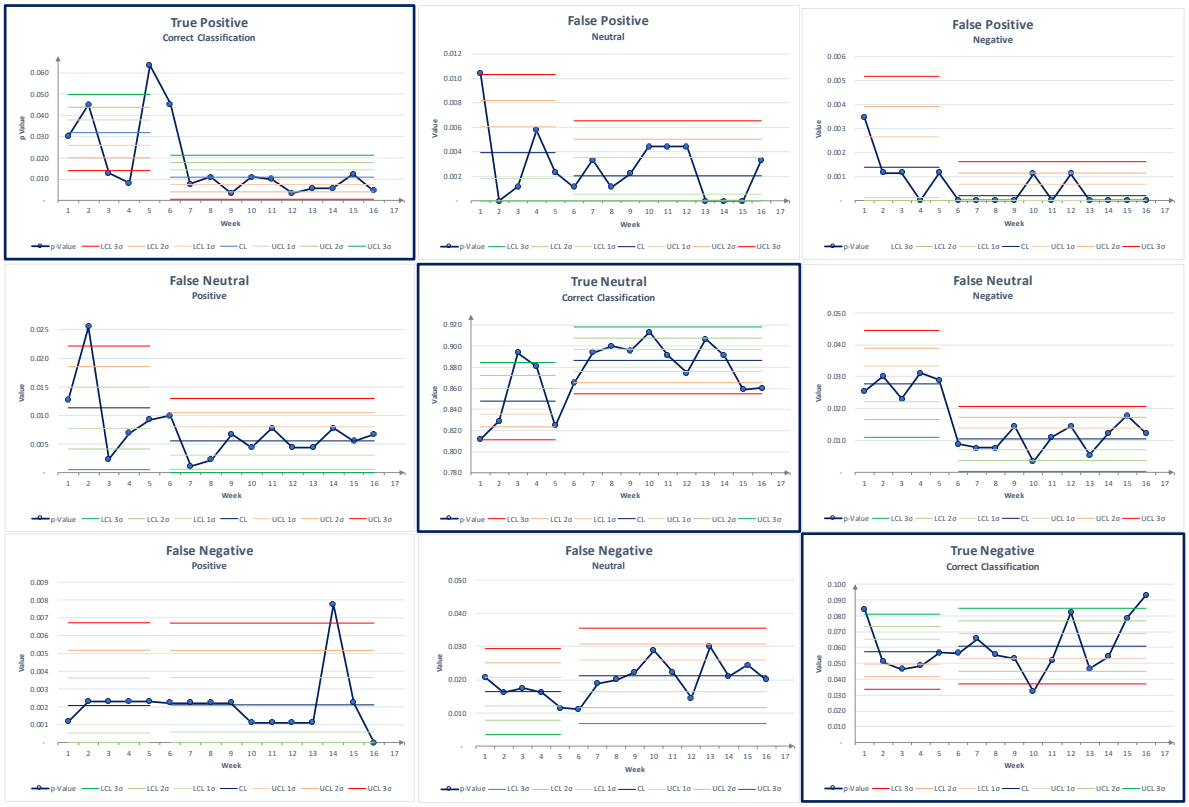

Figure 48. Control Charts for p-values, n = sample. Week 16

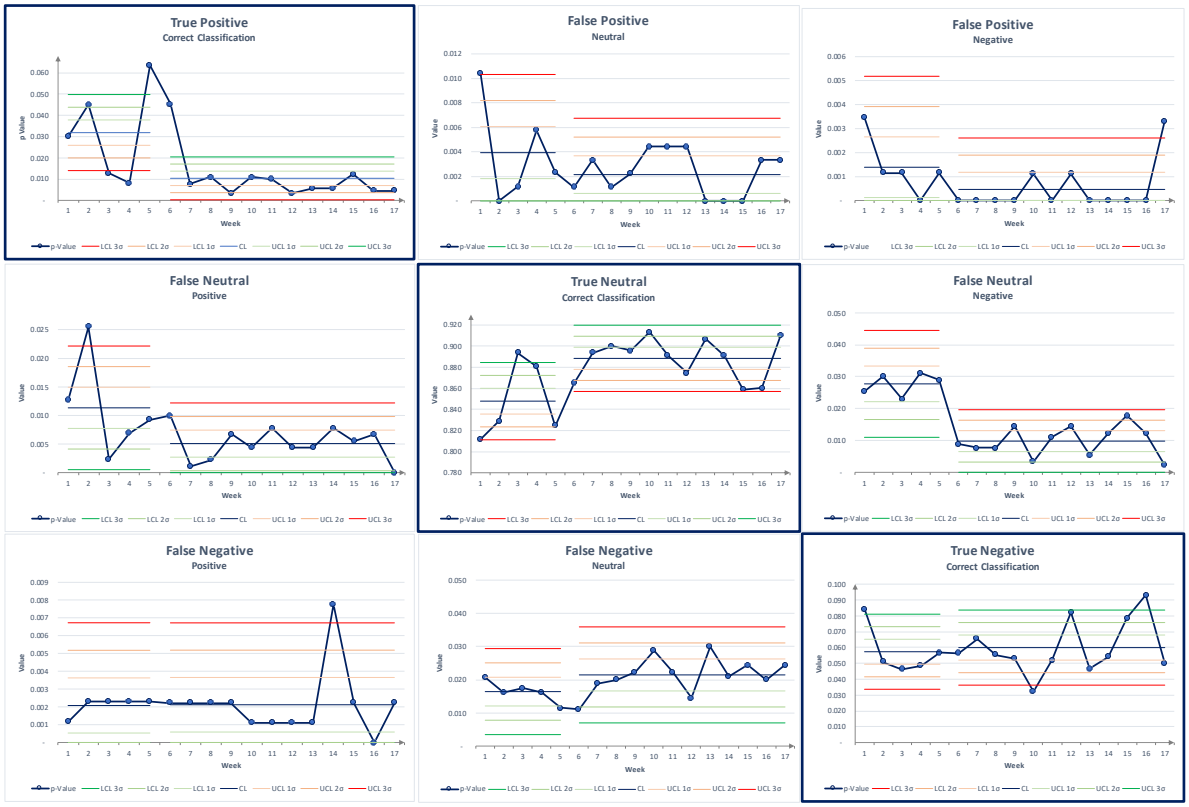

Figure 49. Control Charts for p-values, n = sample. Week 17

Appendix 3

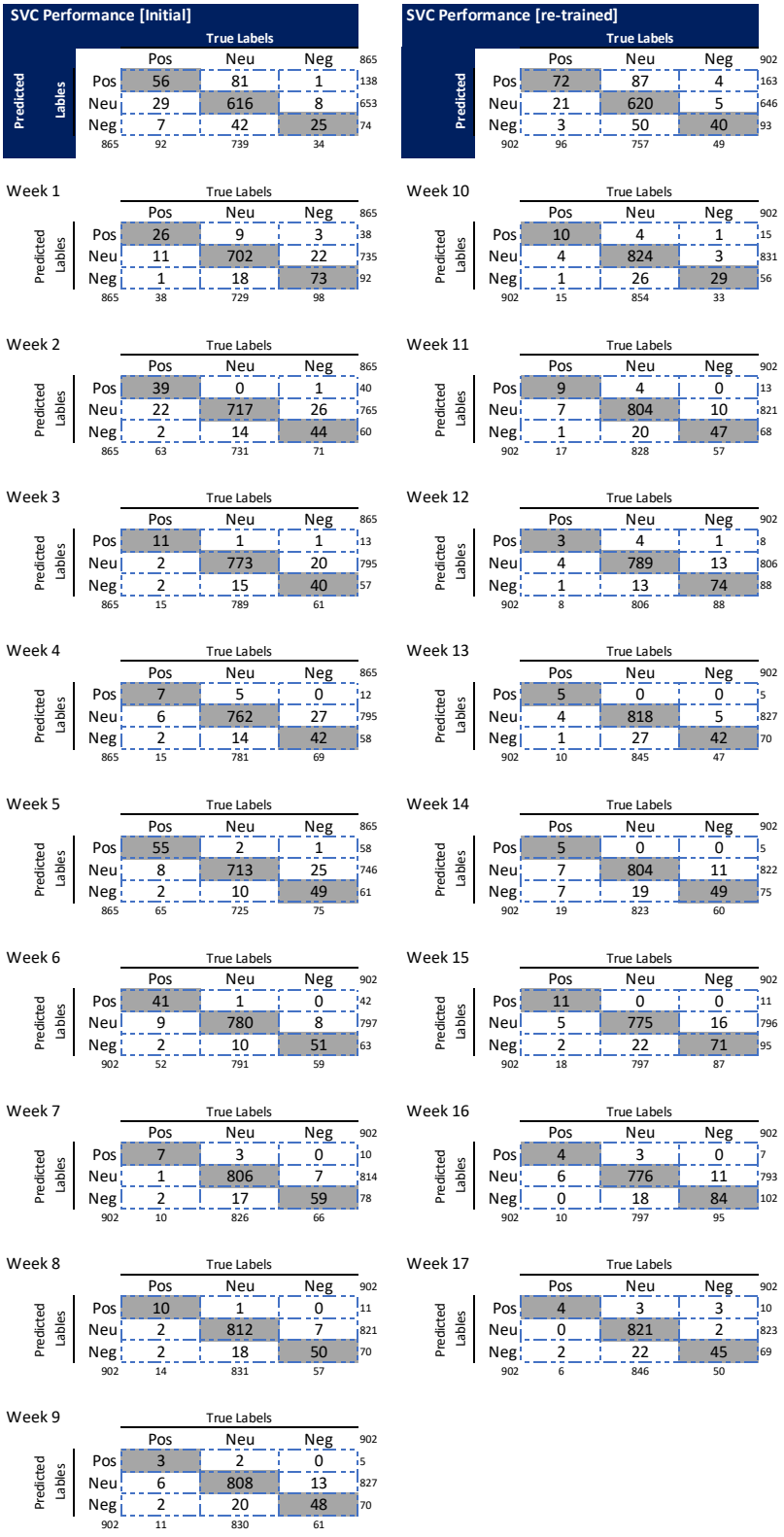

Figure 50. Confusion matrices for all the samples.

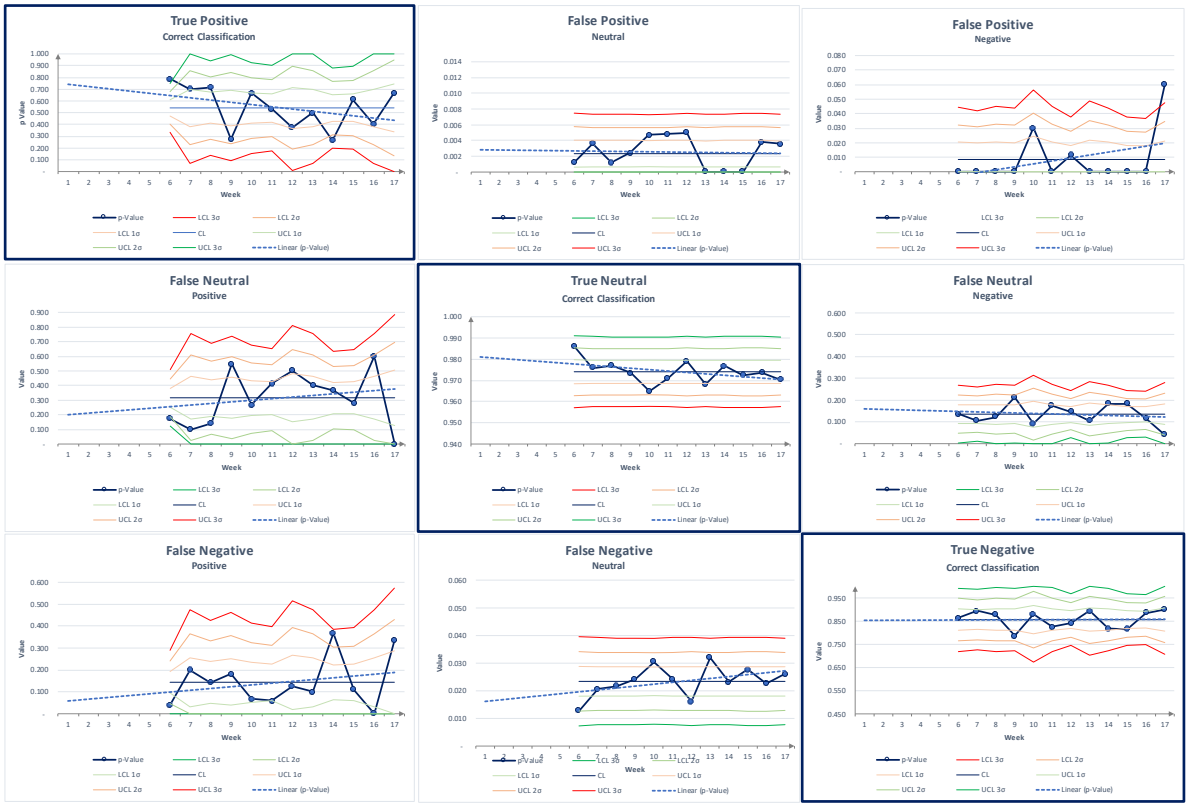

Figure 51. Control Charts for p-values, n = class. Linear trends

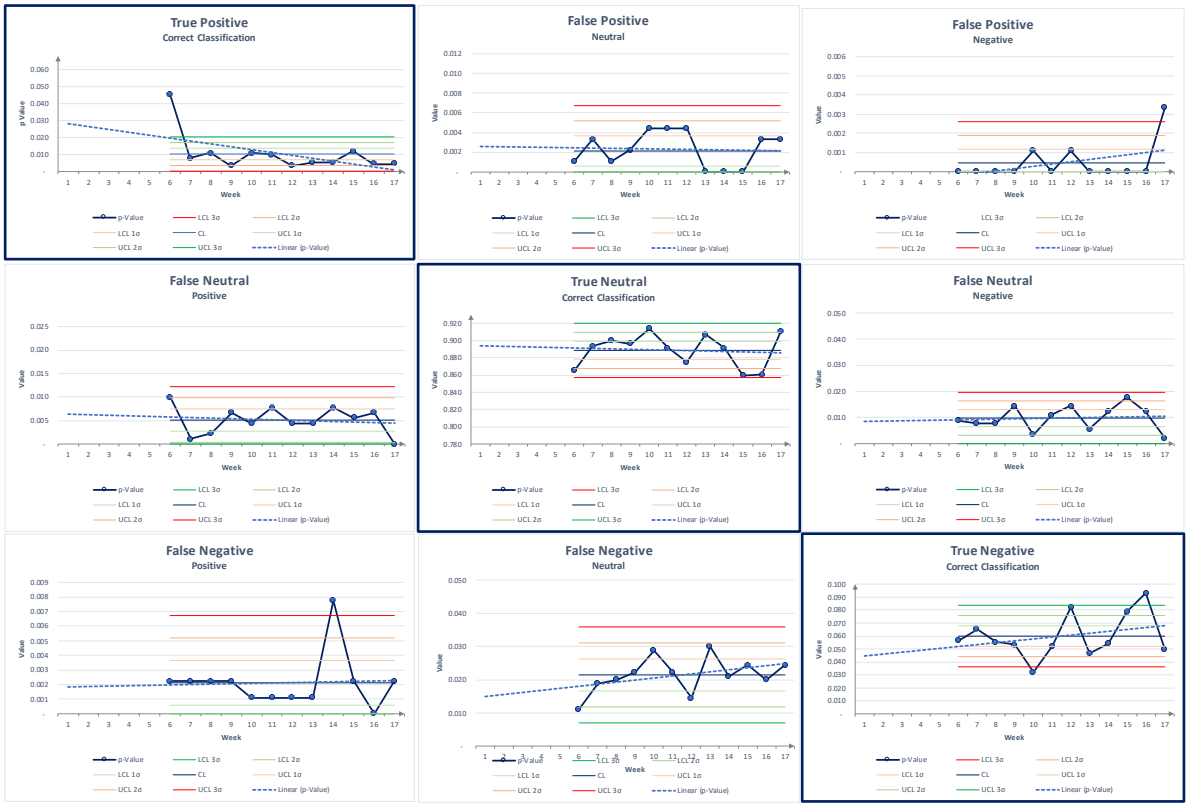

Figure 52. Control Charts for p-values, n = sample. Linear trends.

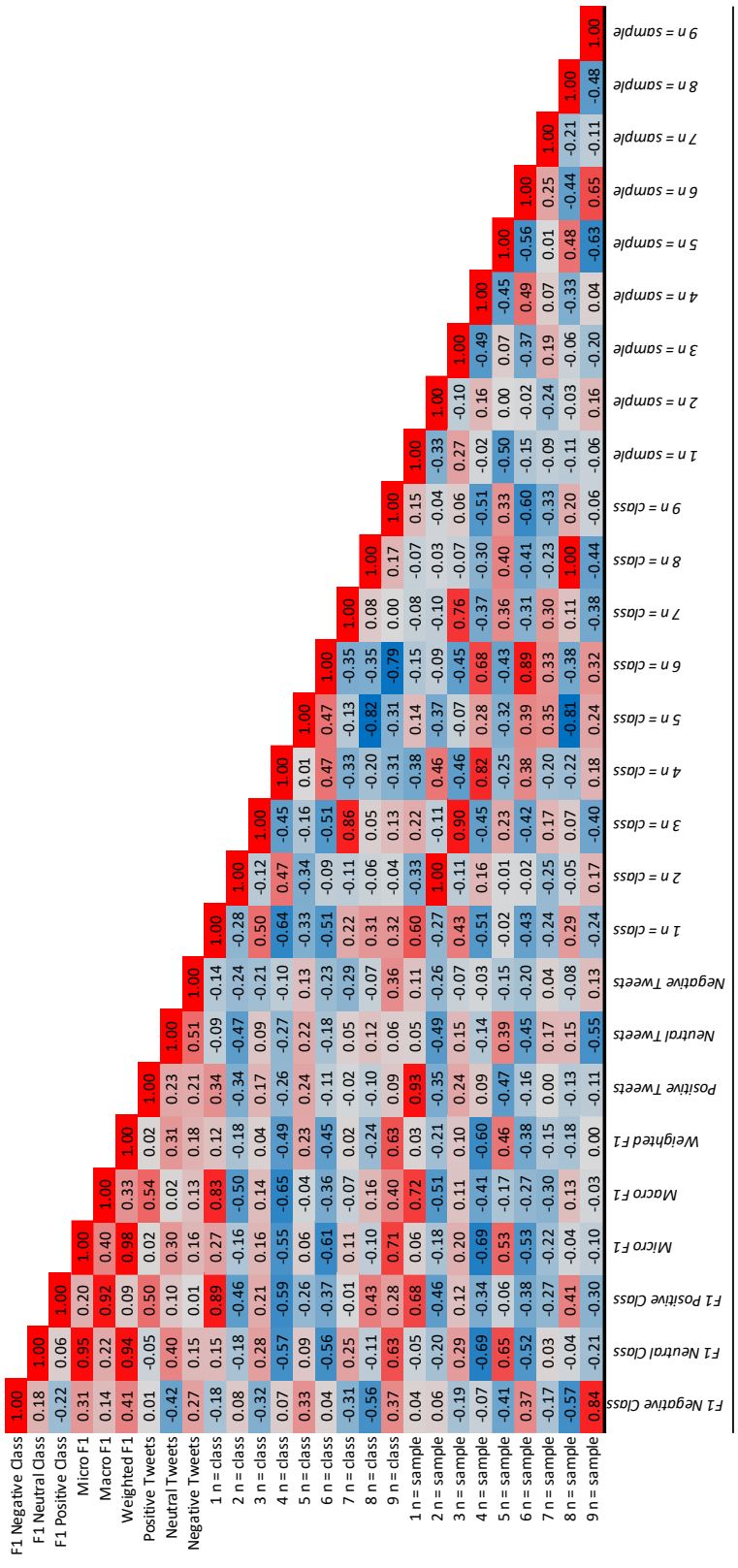

Figure 53. Correlation Matrix

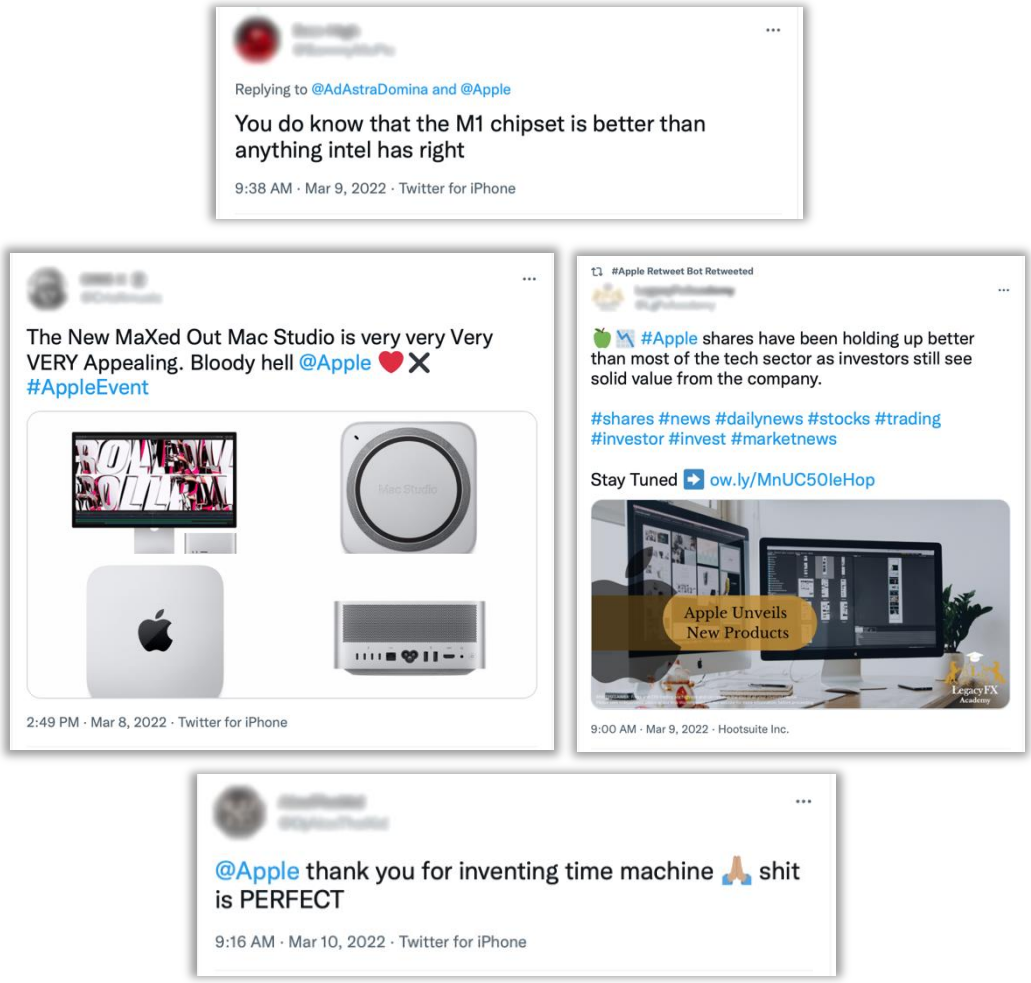

Figure 54. An example of positive tweets that were classified as negative on week 17

Training dataset

#		Count	Fraction
1	Positive sentiment	747	14.94%
2	Negative sentiment	451	9.02%
3	Neutral sentiment	3,763	75.24%
4	Suspended tweets (unavailable at time of observation)	40	0.80%
	Total tweets:	5,001	

Figure 55. Training dataset

The dataset

Week #	Week starts [12:00 AM]	Week ends [12:00 PM]	Count
1	December 5, 2021	December 11, 2021	51,432
2	December 12, 2021	December 18, 2021	64,450
3	December 19, 2021	December 25, 2021	72,246
4	December 26, 2021	January 1, 2022	49,216
5	January 2, 2022	January 8, 2022	55,555
6	January 9, 2022	January 15, 2022	56,302
7	January 16, 2022	January 22, 2022	49,742
8	January 23, 2022	January 29, 2022	70,470
9	January 30, 2022	February 5, 2022	66,290
10	February 6, 2022	February 12, 2022	67,941
11	February 13, 2022	February 19, 2022	45,707
12	February 20, 2022	February 26, 2022	56,036
13	February 27, 2022	March 5, 2022	76,465
14	March 6, 2022	March 12, 2022	82,888
15	March 13, 2022	March 19, 2022	50,650
16	March 20, 2022	March 26, 2022	50,806
17	March 27, 2022	April 2, 2022	53,602
Total tweets:			1,019,798

Figure 56. The dataset