



Key Factors in Achieving Service Level Agreements (SLA) for Information Technology (IT) Incident Resolution

Ajaya K. Swain¹ · Valeria R. Garza¹

Accepted: 4 February 2022 / Published online: 23 March 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

In this paper, we analyze the impact of various factors on meeting service level agreements (SLAs) for information technology (IT) incident resolution. Using a large IT services incident dataset, we develop and compare multiple models to predict the value of a target Boolean variable indicating whether an incident met its SLA. Logistic regression and neural network models are found to have the best performance in terms of misclassification rates and average squared error. From the best-performing models, we identify a set of key variables that influence the achievement of SLAs. Based on model insights, we provide a thorough discussion of IT process management implications. We suggest several strategies that can be adopted by incident management teams to improve the quality and effectiveness of incident management processes, and recommend avenues for future research.

Keywords IT incident resolution · IT service management · ServiceNow · Cloud-computing · Theory for predicting · Predictive analytics

1 Introduction

If you want the present to be different from the past, study the past. – Baruch Spinoza (1632 – 1667).¹

Quality of Information Technology (IT) services is typically evaluated by the reliability of their uninterrupted operation. Reliable operation is closely linked to the number of IT service incidents, and cost of IT service incidents is high. In 2015, IT downtime cost in terms of employee productivity was \$700 billion for North American companies (Saarelainen, 2016). This cost amounted to 4% of the combined GDP of the United States and Canada as reported by

International Monetary Fund (2015). On average, an outage in a data center cost \$8,000 per minute, and average overall cost per incident is \$630,000 (Ponemon Institute, 2013). The data center outage cost has been on the rise over the years and it increased by 38 percent between 2010 and 2018 (Splunk, 2019).

In 2016, the average outage cost nearly three quarters of a million dollars, an increase of nearly 50% from 2010. Each minute of outage cost nearly \$9,000, which meant the typical IT downtime was approximately 80 min. Although this may appear a short period of time, the cost and stress for the IT department of an outage is overwhelming (Splunk, 2019). In another study of about three hundred firms with downtime incidents, it was found that approximately 15% of the outages in the survey cost over a million dollars. Even in one case, the cost of the outage exceeded 50 million dollars (Uptime Institute, 2018). Information security incidents have been on the rise since 2009 with an average of 117,339 incidents per day, and they have caused millions of dollars in financial losses (Tan et al., 2015). It is important to mention that the numbers reported here are opportunity costs from the outage, rather than direct costs associated with fixing it (Ponemon Institute, 2016).

With expanding online markets, the downside risk from an outage is on the rise. The largest costs associated with

¹ Baruch Spinoza was a Dutch Jewish philosopher, born Benedito de Espinosa, later Benedict de Spinoza. He was one of the foremost exponents of 17th-century Rationalism and one of the early and seminal figures of the Enlightenment. His monumental work, “Ethics”, which was published posthumously in 1677, presents an ethical vision unfolding out of a monistic metaphysics in which God and Nature are identified.

✉ Ajaya K. Swain
aswain@stmarytx.edu

Valeria R. Garza
vgarza33@mail.stmarytx.edu

¹ Greehey School of Business, St. Mary’s University,
San Antonio, Texas 78228, USA

outages are business disruption, productivity, and lost revenue, which combinedly make up over 60% of the total cost of an outage. Costs are highest in high-transaction industries like finance and e-commerce, where downtime means lost money (Dwivedi et al., 2015; Splunk, 2019). Therefore, preventing or eliminating the number of IT service incidents is important at the organizational and national level. Although the financial impact of IT services incidents on society is found to be significant, as illustrated above, few studies have been conducted in this area. There is a need for studies investigating the factors responsible for IT service incidents, so that we can better understand the cause of IT service incidents (March & Scudder, 2019; Saarelainen, 2016). Given the enormous amount of data generated by modern organizations, mining these data might help organizations predict and prevent future IT outages and the costs associated with them.

Large volumes of incidents are challenging to manage even in well-staffed companies. Incident classification is a common challenge (Jäntti & Cater-Steel, 2017) which can lead to inaccurate priority identification and increased ticket reassignment counts. Additionally, mistakes in the creation of incident categorization and prioritization systems, as well as in incident data entry, expose organizations to greater risk of failing to meet SLAs (Astuti et al., 2017). These factors not only prolong incident resolution, but also result in widespread ramifications beyond the information systems; finance, marketing, operations, and supply chain management are among the key areas impacted by service management design and execution (Bardhan et al., 2010). The resulting degradation in employee workflows and customer service leads companies to suffer from reputational and financial repercussions due to customer dissatisfaction (Järveläinen, 2013).

A Service Level Agreement (SLA) is typically an agreement between a customer and a provider to receive a particular service. SLAs contain a variety of parameters such as response time, bandwidth, storage, reliability, deadline, throughput, delay, and cost that must be maintained by a provider. The provider must measure and monitor these parameters during the service to avoid violations that have been agreed in the SLA. Therefore, the SLA plays a role in adapting to dynamic environmental changes and heterogeneous resources (Goo, 2010; Mubeen et al., 2017; Sahal et al., 2016).

In light of the COVID-19 pandemic, business leaders are faced with significant challenges in ensuring business continuity (Biddle, 2020). Effective incident management strategies are thus more critical to business continuity than ever before. Employees and customers around the world have been forced to adopt remote operating practices, so the achievement of incident service level agreements (SLAs) is crucial for companies to sustain quality customer and employee experiences. Adherence to SLAs is also key to

mitigating business risk, fulfilling contractual obligations, and meeting legal requirements, particularly in scenarios where service provision is outsourced all or in part to third party providers (Desai, 2010). While many system events are occurring simultaneously, effective incident management is needed to determine whether the events are truly related to an incident (IBM Cloud Education, 2019), thereby cutting through the noise and avoiding exploration of dead-end troubleshooting routes.

The process of learning from business failure benefits society through application of that knowledge to subsequent businesses to prevent reoccurrence of such failures. Learning typically involves several processes: situation assessment, problem detection, potential solution synthesis, solution implementation, outcome evaluation, and identification of patterns among the problems and failures. Organizational learning results when members of the organization react to changes in the internal or external environment of the organization by detection and correction of failures (Sage & Rouse, 1999). By pursuing success and avoiding failure, firms introduce errors that not just inhibit learning and interpretation processes but also make failure more likely or expensive than necessary. Moreover, it is often easier to point out why a failure has occurred than to explain a success, making failure analysis a powerful mechanism for resolving uncertainty (McGrath, 1999). Arguing that failure is more important than success for organizational learning, Sitkin (1992) suggests that firms that focus entirely on success could suffer from complacency, decreased resilience and eventual failure. By carefully analyzing failures instead of focusing only on successes, scholars and business decision makers can begin to make systematic progress on better analytical models of organizational value creation (McGrath, 1999). There is enough evidence in literature suggesting that organizations can learn from both success and failure (e.g., Chen et al., 2020; KC et al., 2013; Madsen & Desai, 2010). These arguments and evidence provide a motivation for this study.

Over the years, scholars (e.g., Rizk and Elragal 2020) have criticized the dominance of gap-spotting types of research and encouraged forward-thinking and innovative research in IS to bring in theoretical contributions to the IS field via data science. More recently, there appears to be a consensus that research can start with data or data-driven discoveries, rather than with theory. This study responds to the call from researchers (e.g., Agarwal & Dhar, 2014; Shmueli & Koppius, 2011) for more studies that use predictive analytics to investigate research questions. This study attempts to extract predictive and prescriptive knowledge on the phenomenon of IT incident resolution and the ability to predict certain future scenarios that can guide preventive action. The goal here is to derive meaningful insights by leveraging analytical tools that have not previously been

applied to incident SLA analysis. The findings of this study contribute to the incident management body of literature particularly related to ITIL® and service level management, and they can be useful for proactive incident management. The objectives of this study are two-fold. First, we aim to identify a set of key factors influencing SLAs for incident resolution, and then to suggest strategies for prevention or reoccurrence of these incidents.

This is how the paper is organized. In the following section, we present a review of related work in incident resolution along with the data and research methodology. The results with major findings are presented in Sect. 3. In Sect. 4, we discuss the implications of the findings of this study for theory and practice. A concluding summary of the study is presented in the final section with a discussion on limitations and future research options.

2 Literature Review and Methodology

2.1 IT Service Management

Grown out of the need for addressing and managing complex IT architecture issues, IT service management (ITSM) is a strategy that focuses on defining, managing, and delivering customer-focused IT performance as a service (Winniford et al., 2009). To guide ITSM and in response to businesses demanding better and more disciplined IT services, best practice frameworks such as the IT Infrastructure Library (ITIL) have emerged to help IT organizations become more adaptive, flexible, cost-effective, and service-oriented. ITIL focuses on a complex set of processes, functions, and roles within five stages of the service lifecycle: service strategy, service design, service transition, service operations, and continual service improvement. Most service management efforts emphasize the final stage of the service lifecycle (continual service improvement) to better manage continued IT services such as service desk, incident management, and problem management (Pollard et al., 2010).

Incident management defines incidents as unplanned disruptions or quality degradations of service. Therefore, the objective of incident management is not to solve the root problem or systemic cause but to quickly restore normal service to end-users. These users may be final consumers or internal business users, so the ITIL standard may be applied to enhance the quality of various steps in the product or service delivery process (Gil-Gómez et al., 2014). The basic steps in an incident management process as recommended by ITIL are: Incident resolution and recording; Classification and assignment; Investigation and diagnosis; Resolution and recovery; and Incident closure (Muhren et al., 2007). Lucidchart (Lucidchart Content Team, 2019), a platform used by various Fortune 500 companies for collaborative process

design and organization, expands the ITIL recommended steps and describes a typical flow of incident management in eight steps as follows:

1. *Incident identification*—An end user or automated notification reports a service interruption.
2. *Incident logging* – The service desk identifies the report as an incident and logs information including user contact information and issue description.
3. *Incident categorization and prioritization* – The service desk tags the incident under the appropriate categories based on organizational schemas (e.g., hardware—office supplies—printer failure) and assigns priority level, in order to determine how and by whom the issue is addressed.
4. *Initial diagnosis* – The assigned IT support team conducts research to describe the problem, develop a hypothesis, and attempt issue resolution.
5. *Functional and hierarchic escalation* – The initially assigned IT support team, if not successful in the initial resolution attempt, escalates the issue to a specialized technical team or consults with managing authorities to determine whether more resources are required to resolve the incident.
6. *Investigation and diagnoses* – Support technicians continue to troubleshoot the issue and attempt various solutions, based on likely causes, to find the correct diagnosis.
7. *Resolution and recovery* – The support team fixes the issue after identifying the correct diagnosis, in order to restore normal service.
8. *Incident closure* – The service desk confirms with the reporting user that the incident has been resolved and closes the incident ticket.

Service management platforms offer solutions to organize and track user incident reports, which is especially critical for large companies fielding hundreds of IT incidents daily. One standout among service management providers is ServiceNow™, founded by Fred Luddy in 2005. Providing ITIL out of the cloud, it offers access to an ITIL conform service management solution. It is a Platform-as-a-service (PaaS) provider that provides a unified cloud-computing platform for automating and transforming IT processes, including security operations, IT service, asset, business operations and event management. ServiceNow™ offers clients simple, customizable tools that create centralized automation for common IT support tasks, including “tracking incidents, recovering passwords, requesting equipment, setting up new user accounts, troubleshooting and managing IT systems and responses through simply designed service portals” (Chaykowski & Coatney, 2018). Its industry prevalence is marked by its recent No. 1 ranking on the Forbes Most

Innovative Companies list and by its 4,000-strong customer base, including 850 companies on Forbes' Global 2000 list of the world's largest firms (Chaykowski & Coatney, 2018). ServiceNow™'s Incident Management product includes features such as omnichannel capture, predictive intelligence, calculated priority, reports and dashboards, and service level agreements (ServiceNow, 2020). With over 30 common attributes describing each incident, ServiceNow™ provides a wealth of data with which to analyze the effectiveness of a company's incident management processes. This study uses data collected by the ServiceNow™ platform in a large IT services company.

2.2 Data Mining and IT Service Management

Data mining can be defined as "...the analysis of (often large) observational datasets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner..." (Hand et al., 2001). It has been an effective tool in extracting useful information from large multi-dimensional databases to aid decision-making. The data mining methodology uses several analytical tools that use available large data for classification, prediction, clustering, summarization, aggregation, and optimization (Zurada & Lonial, 2011; Motiwala et al., 2019), in a form amenable to use for decision-making purposes. Historically, data mining methodology has been used for fraud detection, customer relationship management, market segmentation analysis, risk/affinity analysis, and healthcare (Swain, 2016; Johnson et al., 2021), but not so much in incident management studies. This study adds a new data mining analysis perspective to the IT incident management body of literature.

Several studies have explored and recommended ways to improve IT Service Management using insights derived from the data generated over the IT services lifecycle (e.g. Kloeckner et al., 2018; Pratiwi & Tanaamah, 2020). Gupta et al. (2018) studied an approach to reduce the number of user input requests in the life cycle of IT tickets (incidents). They proposed a preemptive model to prompt users for information needed to service a ticket, based on the predictions of whether a user input request will be needed to process the ticket, and if so, what specific information is likely to be requested. Kloeckner et al. (2018) used logistic regression, gradient boosting tree, and RDF models to quantify ticket resolution quality on a real-world dataset. They found that the length of the resolution was the best indicator of resolution quality. However, we have not discovered any studies that use decision trees in the context of achieving SLAs in incident resolution.

Regression is one of the most popular techniques for predictive modeling (e.g., Singh et al., 2020). Still, not many studies have used logistic regression models to predict

adherence to incident SLAs. Neural Network (NN) models have found their applications in areas such as natural language processing and information retrieval (Ghosh et al., 2018), but these models have rarely been used by researchers in the context of incident management analysis. Son et al., (2014) described the automation of help desk ticket classification by using a Softmax Regression Neural Network (SNN) text classification algorithm. Lyubinetz et al., (2018) also explored text classification in the automated labeling of bugs and tickets in customer support systems, using recurrent neural networks. Kloeckner et al. (2018) used convolutional neural networks (CNNs) to extract non-functional requirements ("how" services should be delivered) from client requirements documents for IT solutions. The extracted requirements were used to create structured representations for SLAs to assess the compatibility of a proposed solution's capabilities to the solution request. To the best of our knowledge, neural networks have not been used in predicting the achievement of SLAs in incident resolution, as we attempt to do in this study.

Gupta et al. (2018) used Support Vector Machine (SVM) when creating preemptive models to predict characteristics of IT incidents in order to efficiently reduce the need for user input throughout the life cycle of the ticket. Al-Hawari and Barham (2021) also used an SVM-based algorithm to associate help desk tickets with the correct services early in the resolution process, in order to minimize resolution time. Our use of SVM models to predict achievement of incident SLAs and to identify key factors in this adherence adds a new complementary dimension to the existing literature for the application of SVM models in IT incident analysis.

2.3 Data and Variables

The data used in this study are from an event log of an incident management process extracted from the audit system of an instance of the ServiceNow™ platform used by a large IT company. The data are loaded from a relational database underlying a corresponding process-aware information system. The original dataset contains 141,712 records corresponding to each state in the lifecycle of 24,918 unique incidents, so the dataset was limited to the 24,918 records for which the incidents are in the final "Closed" state. The repository notes that missing values (denoted by a "?" symbol) should be considered unknown information, so the content in cells with a "?" symbol was redacted to be empty, in order to facilitate recognition of missing variables within SAS Enterprise Miner. Additionally, company-specific information (names, locations, categories, etc.) was anonymized by the repository for privacy, so many attributes contain values such as "Caller 1", "Location 5", "Symptom 10", "Group 25", and so on.

Table 1 Variable Details

Name	Description	Role	Level
<i>number</i>	Incident identifier	ID	Nominal
<i>made_sla</i>	Boolean attribute that shows whether the incident met the target SLA (1-True, 0-False)	Target	Binary
<i>knowledge</i>	Boolean attribute that shows whether a knowledge base document was used to resolve the incident (1-True, 0-False)	Input	Binary
<i>u_priority_confirmation</i>	Boolean attribute that shows whether the priority field has been double-checked (1-True, 0-False)	Input	Binary
<i>reassignment_count</i>	Number of times the incident has the group or the support analysts changed	Input	Interval
<i>reopen_count</i>	Number of times the incident resolution was rejected by the caller	Input	Interval
<i>sys_mod_count</i>	Number of incident updates until that moment	Input	Interval
<i>assignment_group</i>	Identifier of the support group in charge of the incident	Input	Nominal
<i>category</i>	First-level description of the affected service	Input	Nominal
<i>closed_code</i>	Identifier of the resolution of the incident	Input	Nominal
<i>contact_type</i>	Categorical attribute that shows by what means the incident was reported	Input	Nominal
<i>notify</i>	Categorical attribute that shows what notifications were generated for the incident	Input	Nominal
<i>impact</i>	Description of the impact caused by the incident (1-High; 2-Medium; 3-Low)	Input	Ordinal
<i>priority</i>	Value calculated by the system based on <i>impact</i> and <i>urgency</i> (1-Critical; 2-High; 3-Moderate; 4-Low)	Input	Ordinal
<i>urgency</i>	Description of the urgency informed by the user for the incident resolution (1-High; 2-Medium; 3-Low)	Input	Ordinal
<i>assigned_to</i>	Identifier of the user in charge of the incident	Rejected	Nominal
<i>caller_id</i>	Identifier of the user affected	Rejected	Nominal
<i>location</i>	Identifier of the location of the place affected	Rejected	Nominal
<i>opened_by</i>	Identifier of the user who reported the incident	Rejected	Nominal
<i>resolved_by</i>	Identifier of the user who resolved the incident	Rejected	Nominal
<i>subcategory</i>	Second-level description of the affected service (related to the first-level description, i.e., to category)	Rejected	Nominal
<i>u_symptom</i>	Description of the user perception about service availability	Rejected	Nominal

Table 1 contains a summary of the variables with their role and measurement levels. The incident identifier *number* was selected for the ID role. The attribute *made_sla* is the binary target variable. Thirteen other variables were selected as inputs. Although the ordinal variable *priority* is a function of *impact* and *urgency*, all three variables were included to analyze the individual effect of *priority* on the outcome *made_sla*. The remaining eight nominal variables were rejected due to having too many levels (100 or more) to be useful for analysis. Furthermore, it was not possible to perform a reduction to consolidate the number of levels, due to the anonymized nature of the variables.

2.4 Data Exploration

In this study, the SAS Enterprise Miner (SAS EM) 15.1 package was used for data analysis and model building. SAS EM is a Windows based package that allows creation of highly accurate predictive and descriptive models and facilitates comparison of many common data mining techniques. It can create a wide range of predictive models. SAS EM provides a convenient graphical-user-interface (GUI) to create a diagram workspace where a process flow diagram

can be constructed from a tool palette using a drag-and-drop approach.

An initial histogram exploration of three interval variables (*reassignment_count*, *reopen_count*, *sys_mod_count*) reveals interesting relationships with the target *made_sla*. Incidents that had 0 reassignments during their lifecycle had a roughly 2:1 ratio of meeting the SLA. The subgroup with the highest proportion of *made_sla* = TRUE appears to be incidents that had three or fewer updates during their lifecycles. The ratio for meeting the SLA is nearly 6:1 for this subgroup. Conversely, incidents with four or more updates (up to a max of 129) have a proportion of *made_sla* = FALSE which is more than double the proportion of *made_sla* = TRUE. It is worth noting that this subgroup constitutes mostly the incidents that did not meet the SLA.

2.5 Model Selection and Descriptions

As decision tree, logistic regression, neural network, and support vector machine models have not been widely used in analyzing incident resolution management issues, we chose to use all four model types in our study to derive meaningful insights from incident SLA

Table 2 Node Properties

Node	Properties
File Import	Default; variables were assigned roles and levels as described in Table 1
Stat Explore	Default
Data Partition	Train > Data Set Allocations > <i>Training = 50.0, Validation = 50.0</i>
Decision Tree 1	Train > Subtree > <i>Assessment Measure = Decision</i>
Decision Tree 2	Train > Subtree > <i>Assessment Measure = Average Squared Error</i>
Decision Tree 3	Train > Splitting Rule > <i>Maximum Branch = 3</i> Train > Subtree > <i>Assessment Measure = Average Squared Error</i>
Regression 1	Train > Model Selection > <i>Selection Model = Stepwise, Selection Criterion = Validation Error</i>
Regression 2	(Note: The three regression nodes all use the same settings.)
Regression 3	
Impute	Default (Note: There were no missing interval variables in the data. For non-interval variables, the default value for imputation is the mode)
Transform Variables	Train > Variables > <i>reassignment_count = log, reopen_count = log, sys_mod_count = log</i>
Neural Network	Train > Variables > *Selected only top 4 variables from Regression 1 model, in order to achieve convergence Train > Optimization > <i>Maximum Iterations = 500, Preliminary Training: Enable = No</i> Train > Model Selection Criterion = Average Error
Auto Neural	Train > Variables > *Selected only top 4 variables from Regression 1 model Train > Model Options > Train Action = Search, Number of Hidden Units = 1, Tolerance = Low Train > Activation Functions > Direct = No, Normal = No, Sine = No
HP SVM Linear	Optimization Method > Interior Point > Kernel = Linear
HP SVM Poly	Optimization Method > Active Set > Kernel = Polynomial
HP SVM RBF	Optimization Method > Active Set > Kernel = Radial Basis Function
HP SVM Sigmoid	Optimization Method > Active Set > Kernel = Sigmoid
Model Comparison	Default

analysis. Twelve models were created to evaluate model effectiveness based on variations in assessment criteria and data transformation. To have a fair comparison, all models were developed using the same initial dataset, with model-specific settings or transformations applied within the models' properties. A process flow diagram was created in SAS EM to partition the data and run the models. The node properties are summarized in Table 2. Default settings were used unless otherwise noted. The global interactive sampling settings (*Options > Preferences > Interactive Sampling*) were set to *Sample Method = Random* and *Fetch Size = Max*.

Decision Trees The Decision Tree 1 (DT1) model included seven variables (*sys_mod_count*, *knowledge*, *assignment_group*, *priority*, *u_priority_confirmation*, *impact*, and *reassignment_count*) with *sys_mod_count* and *knowledge* having the highest importance. The Decision Tree 2 (DT2) model was based on the assessment criteria of Average Squared Error (ASE) also included seven variables (the same seven as in DT1) that were selected to create the node rules, with *sys_mod_count* and *knowledge* having the highest importance. The Decision Tree 3 (DT3) model was based on the assessment criteria of ASE, with maximum branch settings permitting up to three subsets per splitting rule. Nine variables

were selected to create the node rules (the same seven as in the previous two trees, plus *category* and *urgency*), with *sys_mod_count* and *u_priority_confirmation* having the highest importance.

Logistic Regression The Regression 1 model is a logistic regression created from data that was both transformed and imputed as described in Table 2. Applying the log transformation to *reassignment_count*, *reopen_count*, and *sys_mod_count* greatly reduced the skewness. Eight variables (*LOG_sys_mod_count*, *u_priority_confirmation*, *priority*, *IMP_assignment_group*, *IMP_category*, *knowledge*, *LOG_reassignment_count*, and *M_assignment_group*) were selected by the model, with *LOG_sys_mod_count* and *u_priority_confirmation* having the highest significance. The prefixes are visual identifiers provided by SAS EM to indicate variables that were transformed (*LOG_*), variables that were eligible for imputation (*IMP_*), and dummy variables for events, which had missing values that were, then imputed (*M_*). Table 3 includes a summary of The Regression 1 output.

The Regression 2 model is a logistic regression based on data that was only imputed as described in Table 2. Eight variables (*sys_mod_count*, *u_priority_confirmation*, *priority*, *IMP_assignment_group*, *IMP_category*,

Table 3 Summary of Stepwise Selection for Regression 1

Step/Effect	Statistic		
	Degrees of Freedom (DF)	Chi-Square	Pr > ChiSq
1 <i>LOG_sys_mod_count</i>	1	4748.9949	<0.0001
2 <i>u_priority_confirmation</i>	1	1276.4829	<0.0001
3 <i>IMP_assignment_group</i>	3	633.6992	<0.0001
4 <i>IMP_category</i>	66	663.4149	<0.0001
5 <i>knowledge</i>	46	203.8849	<0.0001
6 <i>LOG_reassignment_count</i>	1	55.7970	<0.0001
7 <i>knowledge</i>	1	43.2769	<0.0001
8 <i>M_assignment_group</i>	1	38.8723	<0.0001

Table 4 Variables Selected by Decision Tree and Regression Models

Variable	Model					
	DT1	DT2	DT3	Regression 1	Regression 2	Regression 3
<i>sys_mod_count</i>	x	x	x	x	x	x
<i>priority</i>	x	x	x	x	x	x
<i>u_priority_confirmation</i>	x	x	x	x	x	x
<i>urgency</i>			x			
<i>impact</i>	x	x	x			
<i>category</i>			x	x	x	
<i>knowledge</i>	x	x	x	x	x	
<i>assignment_group</i>	x	x	x	x	x	
<i>reassignment_count</i>	x	x	x	x	x	

knowledge, *M_assignment_group*, and *reassignment_count*) were selected by the model, with *sys_mod_count* and *u_priority_confirmation* having the highest significance. The Regression 3 model included three variables (*sys_mod_count*, *u_priority_confirmation*, and *priority*) were selected by the model, with *sys_mod_count* having the highest significance.

Neural Networks SAS EM does not have a built-in variable selection method in the neural network tool, so the regression node results facilitate the variable selection process for the models. Due to the high number of levels in categorical variables such as *assignment_group* and *category*, each model had several hundred estimated weights.

Support Vector Machines As with the neural networks, the Linear, Polynomial, Radial Basis Function (RBF), and Sigmoid High Performance (HP) SVM models were connected to The Regression 1 node, so that the data transformation from The Regression 1 path would also be applied to the SVM models.

2.6 Model Comparison

Table 4 contains a comparison of variables selected by the decision tree (DT) and regression models. The accuracy, sensitivity (true-positive), and specificity (true-negative) rates for the neural network and support vector machine (SVM) models are provided in Table 5.

The Model Comparison node was used to generate an aggregate view of the misclassification rate and ASE for all twelve models. The comparison fit statistics summary is provided in Table 6, and an overview of the event classification for all models is provided in Table 7. The SVM models generated the highest validation ASEs of all the models, with a 2.31% difference between the “best” SVM model (RBF) and the “worst” non-SVM model (Regression 3). The SVM (linear)

Table 5 Comparison of Neural Network and SVM Models

Model	Accuracy	Sensitivity	Specificity
Neural Network	0.860262	0.884080	0.818960
AutoNeural	0.854483	0.876234	0.816765
HP SVM Linear	0.860663	0.898001	0.795918
HP SVM Polynomial	0.858817	0.893951	0.797893
HP SVM RBF	0.858817	0.887623	0.808865
HP SVM Sigmoid	0.473473	0.595545	0.261795

model had the lowest misclassification rate of all the models at 13.93%. The Sigmoid model had the worst performance under both ASE and misclassification criteria. The ASEs of the non-SVM models matched very closely, with a range of approximately 1.36% between the minimum and maximum values. Regression 1, Regression 2, and the Neural Network models have the same ASE at 13.96%. Regression 3 generated the highest error rates among the non-SVM models. In the

Table 6 Model Fit Statistics (Misclassification Rate and Average Squared Error)

Model	Validation Misclassification Rate	Model	Validation Average Squared Error (ASE)
HP SVM Linear	0.139337	Neural Network	0.095569
Regression 2	0.139578	Regression 2	0.097498
Regression 1	0.139578	DT3	0.099829
Neural Network	0.139578	Regression 1	0.100174
HP SVM RBF	0.141183	AutoNeural	0.103974
HP SVM Polynomial	0.141183	DT2	0.104043
DT3	0.145517	DT1	0.104190
AutoNeural	0.145517	Regression 3	0.109171
DT2	0.149771	HP SVM RBF	0.132247
DT1	0.150012	HP SVM Linear	0.160068
Regression 3	0.155229	HP SVM Polynomial	0.167199
HP SVM Sigmoid	0.526527	HP SVM Sigmoid	0.253133

Table 7 Event Classification Table for All Models

Model	Count of Events			
	False Negative	True Negative	False Positive	True Positive
DT1	929	3617	940	6973
DT2	929	3620	937	6973
DT3	914	3658	899	6988
Regression 1	629	3252	1305	7273
Regression 2	641	3459	1098	7261
Regression 3	755	3573	984	7147
Neural Network	916	3732	825	6986
AutoNeural	978	3722	835	6924
HP SVM Linear	806	3627	930	7096
HP SVM Poly	838	3636	921	7064
HP SVM RBF	888	3686	871	7014
HP SVM Sigmoid	3196	1193	3364	4706

following section, we discuss the model outputs in more detail in terms of the impacts of key input variables on the target.

3 Results

The Logistic Regression and Neural Network (NN) models appeared to be the most accurate for SLA prediction. These models are in both sets of the four “best” models with the lowest misclassification rate and ASE. While the Neural Network is the best of the three models as it has the same misclassification rate and the best ASE, Regression 1 does mitigate the potential overfitting effects of extreme outliers in *sys_mod_count* (mode = 2; max = 129) and *reassignment_count* (mode = 0; max = 27). Transforming the skewed variables prevents the model from only selecting skewed inputs that appear to dominate additional variables that would help more accurately define the model.

Additionally, though Neural Network has the best ASE, regression model results generally yield more accessible model interpretability than do neural networks. The ASE for Regression 1 is only 0.27% higher than that of Regression 2 and 0.46% higher than that of Neural Network. The minimal difference in ASE, model interpretability, and consistent misclassification rate thus provide the basis for selecting Regression 1 as the best model for predicting the achievement of SLAs.

A summary of the odds ratio estimates for Regression 2 is provided in Table 8. Due to a large number of levels for *IMP_assignment_group* and *IMP_category*, only a partial list of odds ratios for those variables is included. Many of the ratios for *IMP_assignment_group* and *IMP_category* appear to imply that incidents assigned to any other group or category besides Group 9 or Category 9 have a disproportionately higher probability of meeting the SLA than do incidents in Group 9 or Category 9. Contrary to our expectation,

Table 8 Select Odds Ratio Estimates for Regression 1

Effect		Point Estimate
<i>IMP_assignment_group</i>	Group 2 vs Group 9	999.000
	Group 13 vs Group 9	999.000
	Group 25 vs Group 9	999.000
	Group 32 vs Group 9	999.000
	Group 44 vs Group 9	999.000
	Group 56 vs Group 9	999.000
	Group 65 vs Group 9	999.000
<i>IMP_category</i>	Category 4 vs Category 9	999.000
	Category 10 vs Category 9	999.000
	Category 21 vs Category 9	999.000
	Category 50 vs Category 9	766.808
<i>LOG_reassignment_count</i>		1.647
<i>LOG_sys_mod_count</i>		0.022
<i>M_assignment_group</i>	0 vs 1	1.942
<i>knowledge</i>	0 vs 1	2.162
<i>priority</i>	1—Critical vs 4—Low	<0.001
<i>priority</i>	2—High vs 4—Low	<0.001
<i>priority</i>	3—Moderate vs 4—Low	0.029
<i>u_priority_confirmation</i>	0 vs 1	0.082

the *reassignment_count* ratio suggests that the likelihood of meeting the SLA increases by 64.7% each time an incident is reassigned.

The *knowledge* ratio suggests that incidents which were resolved without the use of a knowledge base (KB) article are overall 2.6 times more likely to meet the SLA than those that did use a KB. The *LOG_sys_mod_count* ratio indicates that the likelihood of meeting the SLA decreases by 2.2% each time an incident is updated. The *u_priority_confirmation* ratio implies that the likelihood of meeting the SLA decreases by 8.2% when incident priority is not double-checked. The *priority* ratios indicate that incidents marked with Critical, High, or Moderate priority are less likely to meet the SLA than incidents marked with Low priority.

The odds ratios of Regression models (Tables 9 and 10) provide additional insights. In Table 9, the ratio for *sys_mod_count* indicates that the likelihood of meeting the SLA decreases by 56.5% each time an incident is updated. The *u_priority_confirmation* ratio indicates that the likelihood of meeting the SLA decreases by 11.8% when incident priority is not double-checked. In Table 10, the ratio for *sys_mod_count* indicates that the likelihood of meeting the SLA decreases by 61.2% each time an incident is updated. The *u_priority_confirmation* ratio indicates that the likelihood of meeting the SLA decreases by 11% when incident priority is not double-checked.

Table 9 Select Odds Ratio Estimates for Regression 2

Effect		Point Estimate
<i>IMP_assignment_group</i>	Group 2 vs Group 9	999.000
	Group 13 vs Group 9	999.000
	Group 25 vs Group 9	999.000
	Group 32 vs Group 9	999.000
	Group 44 vs Group 9	999.000
	Group 56 vs Group 9	999.000
	Group 65 vs Group 9	999.000
<i>IMP_category</i>	Category 4 vs Category 9	608.955
	Category 10 vs Category 9	999.000
	Category 21 vs Category 9	999.000
	Category 50 vs Category 9	234.279
<i>M_assignment_group</i>	0 vs 1	2.761
<i>knowledge</i>	0 vs 1	2.830
<i>priority</i>	1—Critical vs 4—Low	<0.001
<i>priority</i>	2—High vs 4—Low	<0.001
<i>priority</i>	3—Moderate vs 4—Low	0.017
<i>reassignment_count</i>		1.356
<i>sys_mod_count</i>		0.565
<i>u_priority_confirmation</i>	0 vs 1	0.118

Table 10 Odds Ratio Estimates for Regression 3

Effect		Point Estimate
<i>priority</i>	1—Critical vs 4—Low	<0.001
<i>priority</i>	2—High vs 4—Low	<0.001
<i>priority</i>	3—Moderate vs 4—Low	0.044
<i>sys_mod_count</i>		0.612
<i>u_priority_confirmation</i>	0 vs 1	0.110

As is evident in Table 6, the difference between the DT3 ASE and that of DT2 is marginal at 0.42%, so the complexity and overfitting of DT3 are not sufficiently justified to select it as a superior decision tree model. DT2 is thus the best decision tree model. The Node Rules from DT2 reveal three key variable profiles presented in Table 11. Events that will certainly not meet the SLA at a 0% achievement rate are described by Profile 1, which is characterized by nineteen *assignment_group* values and *sys_mod_count* values between 7 and 11, inclusive. For the small subset of observations (52 total) assigned to the node, the values of *knowledge* and *priority* appear to indicate that failure to meet the SLA corresponds with not using KB articles on critical or high priority incidents.

Profile 2 describes events that will almost certainly not meet the SLA at a 1% achievement rate. Excluding missing values, the 715 events that fall into Profile 2 have

Table 11 Variable profiles from DT2

Profile	Rules	Predictions
Profile 1 (52 observations)	<ul style="list-style-type: none"> • <i>sys_mod_count</i> < 11.5 AND <i>sys_mod_count</i> > = 6.5 or MISSING • <i>priority</i> < = 2—HIGH • <i>knowledge</i> IS ONE OF: 0 or MISSING • <i>assignment_group</i> IS ONE OF: GROUP 56, GROUP 23, GROUP 28, GROUP 24, GROUP 54, GROUP 65, GROUP 70, GROUP 30, GROUP 55, GROUP 27, GROUP 39, GROUP 20, GROUP 6, GROUP 60, GROUP 49, GROUP 69, GROUP 62, GROUP 47, GROUP 51 	<i>made_sla</i> = 1 = 0.00 <i>made_sla</i> = 0 = 1.00
Profile 2 (715 observations)	<i>sys_mod_count</i> > = 17.5 or MISSING <i>impact</i> < = 2—MEDIUM or MISSING <i>assignment_group</i> IS ONE OF: GROUP 56, GROUP 25, GROUP 23, GROUP 28, GROUP 29, GROUP 24, GROUP 54, GROUP 66, GROUP 65, GROUP 70, GROUP 72, GROUP 30, GROUP 55, GROUP 27, GROUP 39, GROUP 73, GROUP 20, GROUP 74, GROUP 33, GROUP 12, GROUP 31, GROUP 76, GROUP 6, GROUP 61, GROUP 68, GROUP 15, GROUP 69, GROUP 3, GROUP 10, GROUP 57, GROUP 58, GROUP 37, GROUP 75, GROUP 59, GROUP 22, GROUP 50, GROUP 26, GROUP 9, GROUP 51 or MISSING	<i>made_sla</i> = 1 = 0.01 <i>made_sla</i> = 0 = 0.99
Profile 3 (4,528 observations)	<i>u_priority_confirmation</i> IS ONE OF: 1 or MISSING <i>sys_mod_count</i> < 4.5 or MISSING <i>priority</i> > = 3—MODERATE or MISSING <i>knowledge</i> IS ONE OF: 0 or MISSING <i>assignment_group</i> IS ONE OF: GROUP 56, GROUP 25, GROUP 23, GROUP 28, GROUP 24, GROUP 66, GROUP 65, GROUP 70, GROUP 55, GROUP 27, GROUP 39, GROUP 73, GROUP 20, GROUP 46, GROUP 5, GROUP 33, GROUP 31, GROUP 6, GROUP 49, GROUP 69, GROUP 64, GROUP 10, GROUP 58, GROUP 22, GROUP 13, GROUP 34 or MISSING	<i>made_sla</i> = 1 = 0.99 <i>made_sla</i> = 0 = 0.01

sys_mod_count values of 18 or greater and *impact* values of Medium or higher, and they are in one of the 39 listed assignment groups. The wide range of assignment groups suggests that incidents with several modifications to the ticket are susceptible to breaching their SLA, regardless of the assignment group. The 4,528 events in Profile 3 will almost certainly meet the SLA at a 99% achievement rate. The node rules for Profile 3 indicate that SLA achievement is associated with incidents that have been modified less than four times, whose priority is confirmed to be moderate or lower, and in which a KB article was not used. The DT2 node rules for Profiles 2 and 3 are thus consistent with the odds ratio observations from the regression models.

4 Discussion

4.1 Implications of Findings

4.1.1 Implications for Theory

This study provides several theoretical contributions. First, scholars (e.g., Rizk and Elragal 2020; Klein & Hirschheim, 2008; Agarwal and Lucas Jr. 2005) criticize the dominance of gap-spotting types of research and highlight the need to develop forward-thinking and innovative research in IS, suggesting to bring in theoretical contributions to the IS field via data science. This study is a small step in that direction. Second, IS studies focus on sociotechnical systems that are difficult to measure and theorize (Müller

et al., 2016). IS studies have mostly relied on self-reported data collected via surveys, experiments, or case studies (Hedman et al., 2013). While these methods of data collection offer many advantages, the processes may be costly, cumbersome, and subject to biases. The data used in this study was user-generated, and it was not initially collected with a specific research purpose in mind hence, it may be less contrived or biased. This potential advantage provides additional authenticity to the study and increases its replicability. Third, Müller et al (2016) suggest that research can start with data or data-driven discoveries, rather than with theory. In this sense, our study advances the call from researchers (e.g., Agarwal & Dhar, 2014; Shmueli & Koppius, 2011; Muller 2016) for more studies that use predictive analytics to investigate research questions. Fourth, data-driven predictive algorithms can help to advance theory (Agarwal & Dhar, 2014; Dhar, 2013) as “...patterns [often] emerge before reasons for them become apparent” (Müller et al., 2016, p. 291). Predictive models can lead to the discovery of new constructs, new relationships, nuances to existing models, and unknown patterns (Shmueli, 2010). Our findings attempt to advance the IS theory for predicting framework by adding more empirical evidence to the existing body of knowledge in the area of predictive analytics. Finally, data in research is increasing in volume, velocity, and variety, calling for new ways of systematic extraction of predictive and prescriptive knowledge. Dhar (2013) argues that the usefulness of the extracted knowledge is vital, as it is actionable for decision-making. In this context, our findings provide

new insights on the phenomenon of IT incident resolution and the ability to predict certain future scenarios that can guide preventive action. Additionally, our use of predictive analytics techniques adds to the methodological diversity within IS studies.

4.2 Implications for Practice

The findings of this study provide a set of specific actionable implications for practitioners. The models identified the following variables in descending order of significance: *sys_mod_count*, *u_priority_confirmation*, *priority*, *assignment_group*, *category*, *knowledge*, *reassignment_count*, and *assignment_group*. The activities associated with these variables should thus be central to creating process improvement strategies to provide efficient service and increase the overall rate of meeting incident resolution SLAs. Based on the analysis reports in the previous section, we present the following summary of practical implications for each variable.

- ***sys_mod_count***. Each time an incident is updated, the probability of meeting the SLA decreases by 2.2%. Though the decrease is small, the risk of breaching the SLA becomes much higher as incident updates accumulate. More updates imply that more time was spent gathering information to resolve the issue, suggesting that there may not be sufficient knowledge resources or procedures to quickly diagnose issues and close incidents.
- ***u_priority_confirmation***. Failure to double check the priority of an incident will decrease the likelihood of meeting the SLA by 8.2%. Double-checking priority can ensure that incidents are assigned the appropriate level of urgency and impact, so that the calculated SLAs are consistent with the true nature of the incidents.
- ***priority***. Higher-priority incidents are consistently less likely to meet SLAs than lower-priority incidents. This insight is counterintuitive, but it may suggest that the SLAs for high-priority incidents need to be revisited. The SLAs may have been based on a generic standard or arbitrary assumption that was not realistic for the company from which the data was gathered; high priority incidents may inherently be more complex and thus require more time to remediate appropriately and effectively.
- ***assignment_group***. Incidents assigned to Group 9 appear to be at an exceedingly higher risk of breaching SLAs than is the case for any other group. This could be an indicator that the support team is short-staffed or lacks the training needed to effectively resolve incidents.
- ***category***. Incidents assigned to Category 9 appear to be at an exceedingly higher risk of breaching SLAs than is the case for any other category. It can be inferred that the software applications associated with this category

have high user traffic that exposes the need to invest in application fixes and hardening.

- ***knowledge***. It was counterintuitive to find that the use of KB articles is detrimental to the probability of meeting the SLA. This could be an indicator of an immature or outdated knowledge base system.
- ***reassignment_count***. The likelihood of meeting the SLA increases by 64.7% each time an incident is reassigned. This statistic is contrary to expectations and needs further investigation. It may be that the increase is only applicable up to a certain threshold of reassignments, after which the likelihood of meeting the SLA will sharply begin to decrease. One possible explanation is that incidents are frequently incorrectly assigned or classified when they are opened, as highlighted by Jäntti and Cater-Steel (2017), so at least one reassignment is needed in order to reach the appropriate support team.

The implications described for each variable provide a baseline from which to identify and implement process improvements for incident resolution. In particular, the highest priority should be given to addressing the effects of incident update count (*sys_mod_count*) and priority confirmation (*u_priority_confirmation*), as these attributes appear to have the greatest influence on SLA achievement outcomes. When analyzing the service management process implications, IT management must include key stakeholders including support team staff, software developers, and software users, so that each party can provide the context necessary to accurately interpret the regression model statistics.

5 Conclusions, Limitations, and Future Research

Data science brings a plethora of opportunities to scientific research, including the IS discipline. By focusing on concepts and following more inductive data-driven approaches, with less emphasis on pre-existing theory, data science could bring additional theoretical and practical contributions to the IS domain. The objective of this study was to identify key factors that influence whether SLAs are achieved in the resolution of IT incidents. By analyzing a dataset of incident event logs for the ServiceNow™ incident management process of an undisclosed IT company, we developed and compared twelve models to predict the outcome of the target Boolean variable *made_sla*. In doing so, our goal was to derive meaningful insights by leveraging analytical tools that have not previously been applied to incident SLA analysis. By extension, we also attempted to establish an initial reference point of the models' performance for incident SLA analysis, which may pave the way for continued research in this area of incident management.

We acknowledge a few limitations of this research. First, the findings pertain to a certain IT service provider and a specific IT area, and thereby may not be easily generalized across an entire industry. It was challenging to find a public, sufficiently large dataset pertaining to incident management, with good candidates for a binary target variable and a variety of input variables. However, the apparent scarcity of publicly available data is a reminder that, even if this research is continued by IT companies internally, it may not be possible or straightforward to validate and compare the models against industry data. As a result, it is also difficult to prescribe best practices for achieving SLAs for IT incident resolution at an industry level.

The interpretation of model results also was not without challenges. Although the models in this study did have the similar variable selection and error rates, output statistics such as odds ratios sometimes indicated relationships that were the opposite of what was expected for those variables (e.g., use of a KB article decreasing the likelihood of meeting the SLA). Additionally, the numerous levels for the selected nominal variables (e.g., *assignment_group* and *category*) created tens of odds ratio entries for those variables, making it difficult to draw accurate, generalized conclusions about how the variables affected *made_sla*.

These challenges highlight the importance of thinking creatively and (when possible) consolidating variable levels in order to extract meaningful insights from extensive incident management datasets. The limitation of data anonymization made it impossible to consolidate the variable levels in a meaningful way. However, companies that wish to conduct similar research internally would have full access to identifiable, company-specific information pertaining to the variables. The thoroughness of this study may thus be improved by evaluating the following action items:

- Consolidate the number of levels for nominal variables based on practical boundaries that are known to the company (e.g., *assignment_group* consolidated by departments, *category* consolidated by software application portfolios), in order to facilitate the interpretability of decision tree node rules and regression odds ratios. Level consolidation will also permit more variables (e.g., *assigned_to*, *caller_id*) to be included for model consideration, which may surface new relationships between the inputs and *made_sla*.
- Create reports combining metrics about application incidents with metrics about environmental factors, such as user traffic to applications or network/infrastructure stability, in order to identify possible trends and patterns between incidents and environmental factors.

- Implement real-time incident monitoring for the most significant factors (e.g., *sys_mod_count* and *u_priority_confirmation*), in order to detect incidents at risk of breaching the SLA and proactively prescribe actions to facilitate incident resolution.

The findings of this study contribute to the incident management body of literature particularly related to ITIL® and service level management, and they can be useful for proactive incident management. Incident management teams may consider modifying applicable processes according to the implications and suggestions presented in this study. The authors await studies that further explore linkages and relationships among factors influencing SLAs with a larger sample from more firms across more industries. In the meantime, the authors believe that the integration of company-specific information with the methodologies presented in this research will enable IT analysts and managers to develop more robust models and long-term strategies for effective and efficient IT incident management.

Appendix: Additional Discussion on Predictive Models

Decision Trees

Decision trees are rule induction type models that can segment data by applying a set of rules. Using search heuristics, decision trees find explicit and understandable rules-like relationships among the input and output variables. Search heuristics use recursive-partitioning algorithms to split a large collection of observations into smaller homogeneous group with respect to a particular target variable. The algorithms find the optimum number of splits and determine where to partition the data to maximize the information gain.

Decision trees are built of nodes, branches and leaves that indicate the variables, conditions, and outcomes, respectively. Given a target and a set of explanatory variables, decision algorithms automatically determine which variables are most important, and subsequently sort the observations into the correct output category. The target variable is usually categorical, and the decision tree model calculates the probability of a given record belonging to each of the target categories, or classifies the record by assigning it to the most likely category. The common decision tree algorithms in data mining software are chi-square automatic interaction detector (CHAID), classification and regression tree (CART) and C5. CART uses Gini; C5 uses entropy; and CHAID uses chi-square as the splitting criteria (Yap et al., 2011).

To build a decision tree (Wu et al., 2014), the space of values for the predictors $x = (x_1, \dots, x_n)$ is first partitioned

into a collection of M regions $\omega_1, \dots, \omega_M$ that correspond to a tree's terminal nodes or leaves. The data consist of N observations $(x_i, y_i) = (x_{i1}, \dots, x_{iN}, y_i)$ for $i = 1, \dots, N$. In a node m , suppose there are N_m observations, and $\hat{p}_{ml} = \frac{1}{N_m} \sum_{x_i \in \omega_m} I(y_i = l)$ denote the proportion of class l ($l = 0, 1$) observations in that node for the response y_i . The observations within node m are then classified to the majority class $l_m = \operatorname{argmax}_l \hat{p}_{ml}$.

Decision trees have several advantages over other modeling techniques. These models provide interpretable logic statements in the form of classification (if–then) rules. Each rule represents a path from the root node to each leaf. They can also classify observations with missing data, and classification can be performed for both continuous and categorical variables with fewer complicated computations. However, decision trees are susceptible to noisy data and do not perform as well for nonlinear data and time series data without visible trends and patterns.

Logistic Regression

Logistic regression models can predict a binary outcome variable or multi-class dependent variables using a mix of continuous and discrete predictors. The model can predict the odds of outcome occurrence instead of a point estimate as predicted in a traditional linear regression model. Logistic regression is a widely used statistical modeling technique in which the probability of a dichotomous outcome ($Y = 1$; event happened, or $Y = 0$; otherwise) is related to a set of potential predictor variables. It belongs to the larger class of generalized linear models that link the expected value of the target variable to a linear predictor through a logit link function.

The logistic regression model can be written as:

$$\log \left[\frac{P(Y = 1)}{1 - P(Y = 1)} \right] = \alpha_0 + \sum_{j=1}^n \alpha_j x_j \quad (1)$$

where n denotes the number of predictors. The term on the left is called logit link or log of odds function that transforms the domain $[0, 1]$ into a real line $(-\infty, +\infty)$ through the linear predictor. Assuming that the data consists of N observations, $(x_i, y_i) = (x_{i1}, \dots, x_{iN}, y_i)$ for $i = 1, \dots, N$, the regression parameters α_i are estimated from this modeling dataset.

The logistic regression models developed in SAS Enterprise Miner fit a given model through maximum likelihood estimation, either using Fisher-scoring or the Newton–Raphson optimization algorithm, resulting in the fitted model below (Wu et al., 2014).

$$\log \left[\frac{\hat{p}_i}{1 - \hat{p}_i} \right] = \alpha_0 + \sum_{j=1}^N \alpha_j x_{ij} \quad (2)$$

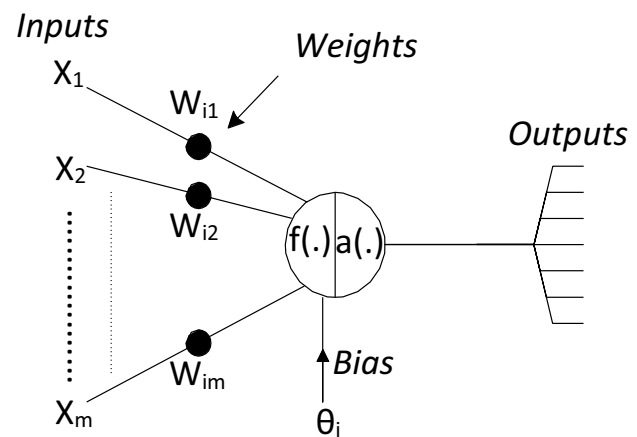


Fig. 1 Artificial Neuron Model (Hanbay et al., 2008)

where $i = 1, 2, \dots, N$ and \hat{p}_i denotes the estimated probability of p_i .

Neural Network

Artificial neural network (NN) models are biologically inspired analytical techniques used for pattern recognition and data classification. By mimicking the neurophysiological and cognitive learning functions of the human brain (Bishop, 1995), NN models can predict new observations (on specific variables) based on other observations (on the same or other variables) from existing data. This is done by combining a large number of simple processing elements called neurons or units into a highly interconnected network, hence the name neural network.

A feedforward network, a class of flexible nonlinear regression, discriminant, and data reduction models, is the simplest and most popular type of neural network. NN models have been widely used for nonlinear mapping, data reduction, pattern recognition, clustering, and classification because of their parallel processing capabilities. They are especially useful for real world prediction problems where mathematical formulae and prior knowledge on the relationship between inputs and outputs are unknown. NN models particularly perform well in applications when the functional form is nonlinear (Sengur et al., 2007).

Figure 1 shows a schematic diagram for an artificial neural network model. The model consists of neurons and connections among those neurons. There are three types of nodes: input nodes, hidden nodes for internal computations, and output nodes that compute the predicted values and compare them with the target variable values. Most connections in a network have an associated numeric value called a weight or parameter estimate. Training a neural network is the process of setting the best weights on the inputs of each of the units. The training methods attempt

to minimize the error function by iteratively adjusting the values of the weights. Most nodes also have one or two associated numeric values called bias and altitude, which are also estimated parameters adjusted by the training methods. Hidden and output nodes use two functions to produce their computed value: a combination function which yields a single value by feeding all the computed values from previous nodes into a given node; and an activation function that then transforms the value produced by the combination function into outputs which involves no weights or other estimated parameters.

As mentioned before, neural network models are trained by experience. When an unknown input is applied to the network, the network can generalize from experiences and produce a new result. The output of the neuron net is given by the following equations:

$$y(t+1) = a\left(\sum_{j=1}^m w_{ij}x_j(t) - \theta_i\right) \quad (3)$$

$$f_i \Delta net_i = \sum_{j=1}^m w_{ij}x_j(t) - \theta_i \quad (4)$$

where $X = (X_1, X_2, \dots, X_m)$ represents the m input applied to the neuron,

w_i represents the weights for input X_i , θ_i is a bias value, and $a(\cdot)$ is the activation function.

Among many available activation functions, nonlinear activation functions such as the S-shaped sigmoid function are by far the most used. The sigmoid is given by the following equation:

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (5)$$

The advantage of NN models lies in their non-dependence on the assumptions about the independence and distribution of residuals or collinearity of input variables. The multilayer perceptron (MLP) is one of the most popular neural networks. MLPs can be used when there is little prior knowledge of the relationship between inputs and targets. However, a large volume of data is required for training the NN models, and the neural network's parameters/connection weights provide little insight into the details of the process. This is an obvious disadvantage because connection weights cannot be easily converted to if-then rules that can be well understood.

Support Vector Machine (SVM)

The term SVM is typically used to describe classification with support vector methods. The foundations of SVM were developed by Vapnik (1995) and are gaining popularity due to many attractive features and promising empirical

performance. The SVM technique uses structural risk minimization, whereas the neural network technique uses empirical risk minimization. This difference equips SVM with a greater ability to generalize, which is the goal in statistical learning (Gunn, 1998).

SVM models are generated through their search for an optimal hyperplane which will maximize the distance from the nearest training data points of any class (Ali et al., 2019). Similar to neural networks, SVM models possess the well-known ability of being universal approximators of any multivariate function to any desired degree of accuracy. Traditional neural network approaches tend to produce models that can overfit the data, resulting in difficulties with generalization. This is a consequence of the optimization algorithms used for parameter selection and the statistical measures used to select the “best” neural network model.

The capability of generalization to new data objects, absence of local minima, flexible non-linear decision boundary, and dependence on very few hyper-parameters are properties that make SVMs suitable for use in various types of classification problems (Ali et al., 2019; Maldonado et al., 2014). Notable fields of research include classification of human biological data, data mining, customer fraud detection, credit scoring, network security monitoring, and image classification (Şen et al., 2020; Yuan et al., 2010; Çomak et al., 2007).

Ali et al. (2019) provide a thorough description of the formulation of an L_2 regularized SVM model, which we present in adapted form as follows: There exists a dataset S with k instances: $S = \{(x_i, y_i) | x_i \in R^P, y_i \in \{-1, 1\}\}_{i=1}^k$ where x_i denotes i^{th} instance and P denotes the dimension of each instance or feature vector. The term y_i denotes the class label, which may be 0 or 1 for an SLA achievement binary classification problem. The function $f(x) = w^T * x + b$, where b is the bias and w is the weight vector, is used by the SVM model to learn hyper-planes. The margin $\frac{2}{\|w\|_2}$ is maximized and the classification error is minimized by the hyperplane of the SVM model. The margin is computed as the sum of the distances to one of the closest positive and one of the closest negative instances. The introduction of the slack variables $\xi_i, i = 1, \dots, k$ and the penalty parameter C force the SVM model to try to balance the minimization of $\|w\|_2^2$ and the minimization of the misclassification errors. The formulation given below presents the corresponding optimization equation and constraints (Ali et al., 2019):

$$\min_{w, b, \xi} \underbrace{\frac{1}{2} \|w\|_2^2 + C}_{\text{Regularizer}} \underbrace{\sum_{i=1}^k \xi_i}_{\text{Error or loss}} \quad s.t. \begin{cases} y_i(w x_i + b) \geq 1 - \xi_i, \\ \xi_i \geq 0, \quad i = 1, \dots, k \end{cases}$$

Declarations

Conflict of Interest The authors have no conflicting interests to declare that are relevant to the content of this article.

References

- Agarwal, R., & Dhar, V. (2014). Editorial - Big data, data science, and analytics: The opportunity and challenge for IS research. *Information Systems Research*, 25(3), 443–448.
- Agarwal, R., & Lucas Jr, H. C. (2005). The information systems identity crisis: Focusing on high-visibility and high-impact research. *MIS Quarterly*, 29(3)381–398.
- Ali, L., Niamat, A., Khan, J. A., Golilarz, N. A., Xingzhong, X., Noor, A., Nour, R., & Bukhari, S. A. C. (2019). An optimized stacked support vector machines based expert system for the effective prediction of heart failure. *IEEE Access*, 7, 54007–54014.
- Al-Hawari, F., & Barham, H. (2021). A machine learning based help desk system for IT service management. *Journal of King Saud University - Computer and Information Sciences*, 33(6), 702–718.
- Astuti, H. M., Muqtadiroh, F. A., Tyas Darmaningrat, E. W., & Putri, C. U. (2017). Risks Assessment of Information Technology Processes Based on COBIT 5 Framework: A Case Study of ITS Service Desk. *Procedia Computer Science*, 124, 569–576.
- Bardhan, I. R., Demirkan, H., Kannan, P. K., Kauffman, R. J., & Sougstad, R. (2010). An Interdisciplinary Perspective on IT Services Management and Service Science. *Journal of Management Information Systems*, 26(4), 13–64.
- Biddle, C. J. (2020). Epidemics and pandemics as high consequence events: Expanding leadership challenges and responsibilities in business continuity during the COVID-19 pandemic and beyond. *Journal of Business Continuity & Emergency Planning*, 14(1), 6.
- Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford University Press.
- Chaykowski, K., & Coatney, M. (2018). From Broke To Billionaire: How Fred Luddy Built The World's Most Innovative Company. Retrieved August 29, 2020, from <https://www.forbes.com/feature/innovative-companies-service-now/>. Accessed 11 March 2020.
- Chen, Q., Li, J. W., Liu, J. G., Han, J. T., Shi, Y., & Guo, X. H. (2020). Borrower learning effects: do prior experiences promote continuous successes in peer-to-peer lending?. *Information Systems Frontiers*, 23(4), 963–986.
- Çomak, E., Arslan, A., & Türkoğlu, İ. (2007). A decision support system based on support vector machines for diagnosis of the heart valve diseases. *Computers in Biology and Medicine*, 37(1), 21–27.
- Desai, J. (2010). *Service Level Agreements: A legal and practical guide*. Ely: IT Governance Publishing.
- Dhar, V. (2013). Data science and prediction. *Communications of the ACM*, 56(12), 64–73.
- Dwivedi, Y. K., Wastell, D., Laumer, S., Henriksen, H. Z., Myers, M. D., Bunker, D., Elbanna, A., Ravishankar, M. N., & Srivastava, S. C. (2015). Research on information systems failures and successes: Status update and future directions. *Information Systems Frontiers*, 17(1), 143–157.
- Ghosh, S., Ghosh, K., Ganguly, D., Chakraborty, T., Jones, G. J., Moens, M. F., & Imran, M. (2018). Exploitation of social media for emergency relief and preparedness: Recent research and trends. *Information Systems Frontiers*, 20(5), 901–907.
- Gil-Gómez, H., Oltra-Badenes, R., & Adarme-Jaimes, W. (2014). Service quality management based on the application of the ITIL standard / Gestión de la calidad de servicios apoyados en la aplicación del estándar ITIL. *DYNA*, 81(186), 51–56.
- Goo, J. (2010). Structure of service level agreements (SLA) in IT outsourcing: The construct and its measurement. *Information Systems Frontiers*, 12, 185–205. <https://doi.org/10.1007/s10796-008-9067-6>
- Gunn, S. (1998). *Support Vector Machines for Classification and Regression* (pp. 5–16, Tech.). Image Speech & Intelligent Systems Group, University of Southampton. Retrieved 2020, from <https://svms.org/tutorials/Gunn1998.pdf>. Accessed 14 Aug 2020.
- Gupta, M., Asadullah, A., Padmanabhuni, S., & Serebrenik, A. (2018). Reducing user input requests to improve IT support ticket resolution process. *Empirical Software Engineering: An International Journal*, 23(3), 1664.
- Hanbay, D., Turkoglu, I., & Demir, Y. (2008). An expert system based on wavelet decomposition and neural network for modeling Chua's circuit. *Expert Systems with Applications*, 34(4), 2278–2283. <https://doi.org/10.1016/j.eswa.2007.03.002>
- Hand, D. J., Mannila, H., & Smyth, P. (2001). *Principles of data mining*. MIT Press.
- Hedman, J., Srinivasan, N., & Lindgren, R. (2013, December). Digital traces of information systems: Sociomateriality made researchable. In proceedings of the 34th international conference on information systems (ICIS), December 15–18. Milan: Italy.
- IBM Cloud Education. (2019). IT Infrastructure Library (ITIL). Retrieved September 21, 2020, from <https://www.ibm.com/cloud/learn/it-infrastructure-library#toc-what-does--u0gdth6p>. Accessed 11 March 2022.
- International Monetary Fund. (2015). World Economic Outlook Database October 2015. Retrieved December 10, 2020, from <https://www.imf.org/en/Publications/WEO/weo-database/2015/October>. Accessed 10 Dec 2020.
- Jäntti, M., & Cater-Steel, A. (2017). Proactive Management of IT Operations to Improve IT Services. *JISTEM - Journal of Information Systems and Technology Management*, 14(2), 191–218. <https://doi.org/10.4301/s1807-17752017000200004>
- Järveläinen, J. (2013). IT incidents and business impacts: Validating a framework for continuity management in information systems. *International Journal of Information Management*, 33(3), 583–590.
- Johnson, M., Albizri, A., & Harfouche, A. (2021). Responsible artificial intelligence in healthcare: Predicting and preventing insurance claim denials for economic and social wellbeing. *Information Systems Frontiers*. <https://doi.org/10.1007/s10796-021-10137-5>
- Kc, D., Staats, B. R., & Gino, F. (2013). Learning from my success and from others' failure: Evidence from minimally invasive cardiac surgery. *Management Science*, 59(11), 2435–2449.
- Kitchin, R. (2014). Big Data, new epistemologies and paradigm shifts. *Big Data & Society*, 1(1), 2053951714528481. <https://doi.org/10.1177/2053951714528481>
- Klein, H. K., & Hirschheim, R. (2008). The structure of the IS discipline reconsidered: Implications and reflections from a community of practice perspective. *Information and Organization*, 18(4), 280–302.
- Kloekner, K., Davis, J., Fuller, N. C., Lanfranchi, G., Pappe, S., Paradkar, A., Schwartz, L., Surendra, M., & Wiesmann, D. (2018). *Transforming the IT Services Lifecycle with AI Technologies*. Springer.
- Lucidchart Content Team. (2019). How to Implement a Successful Incident Management Process. Retrieved August 29, 2020, from <https://www.lucidchart.com/blog/incident-management-process>. Accessed 11 March 2022.
- Lyubins, V., Boiko, T., & Nicholas, D. (2018). Automated labeling of bugs and tickets using attention-based mechanisms in recurrent neural networks. 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), Lviv, 2018, pp. 271–275. <https://doi.org/10.1109/DSMP.2018.8478511>.
- Madsen, P. M., & Desai, V. (2010). Failing to learn? the effects of failure and success on organizational learning in the global orbital launch vehicle industry. *Academy of Management Journal*, 53(3), 451–476.
- Maldonado, S., Pérez, J., Weber, R., & Labbé, M. (2014). Feature selection for support vector machines via mixed integer linear programming. *Information Sciences*, 279, 163–175.

- March, S. T., & Scudder, G. D. (2019). Predictive maintenance: Strategic use of IT in manufacturing organizations. *Information Systems Frontiers*, 21(2), 327–341.
- McGrath, R. G. (1999). Falling forward: Real options reasoning and entrepreneurial failure. *Academy of Management Review*, 24(1), 13–30.
- Motiwalla, L., Deokar, A. V., Sarnikar, S., & Dimoka, A. (2019). Leveraging data analytics for behavioral research. *Information Systems Frontiers*, 21(4), 735–742.
- Mubeen, S., Asadollah, S. A., Papadopoulos, A. V., Ashjaei, M., Pei-Breivold, H., & Behnam, M. (2017). Management of service level agreements for cloud services in IoT: A systematic mapping study. *IEEE Access*, 6, 30184–30207.
- Muhren, W. J., Van Den Eede, G., & de Walle, B. V. (2007). *Organizational learning for the incident management process: Lessons from high reliability organizations*. In proceedings of the european conference on information systems, 65
- Müller, O., Junglas, I., Vom Brocke, J., & Debortoli, S. (2016). Utilizing big data analytics for information systems research: Challenges, promises and guidelines. *European Journal of Information Systems*, 25(4), 289–302.
- Pollard, C. E., Gupta, D., & Satzinger, J. W. (2010). Teaching systems development: A compelling case for integrating the SDLC with the ITSM lifecycle. *Information Systems Management*, 27(2), 113–122.
- Ponemon Institute. (2013). 2013 Cost of Data Center Outages (Rep.). Retrieved December 10, 2020, from Ponemon Institute website: <https://www.ponemon.org/news-updates/blog/security/2013-cost-of-data-center-outages.html>. Accessed 10 Dec 2020.
- Ponemon Institute. (2016). 2015 Cost of Data Center Outages (Rep.). Retrieved December 10, 2020, from Vertiv website: <https://www.vertiv.com/en-us/about/news-and-insights/articles/white-papers/2016-ponemon-institute-report-on-cost-of-data-center-outages/>
- Pratiwi, M. A., & Tanaamah, A. R. (2020). Analysis of IT Service Management in the Salatigaku Application Using the Framework Information Technology Infrastructure Library (ITIL) V3. *Journal of Information Systems and Informatics*, 2(1), 33–43.
- Saarelainen, K. (2016). *How and why things happen: anatomy of IT service incidents*. University of Eastern Finland.
- Sage, A. P., & Rouse, W. B. (1999). Information systems frontiers in knowledge management. *Information Systems Frontiers*, 1(3), 205–219.
- Sahal, R., Khafagy, M. H., & Omara, F. A. (2016). A survey on SLA management for cloud computing and cloud-hosted big data analytic applications. *International Journal of Database Theory and Application*, 9(4), 107–118.
- Şen, D., Dönmez, C. Ç., & Yıldırım, U. M. (2020). A hybrid bi-level metaheuristic for credit scoring. *Information Systems Frontiers*, 22(5), 1009–1019.
- Sengur, A., Turkoglu, I., & Ince, M. C. (2007). Wavelet packet neural networks for texture classification. *Expert Systems with Applications*, 32(2), 527–533. <https://doi.org/10.1016/j.eswa.2005.12.013>
- ServiceNow. (2020). Incident Management - ITSM. Retrieved August 29, 2020, from <https://www.servicenow.com/products/incident-management.html>. Accessed 29 Aug 2022.
- Shmueli, G. (2010). To explain or to predict? *Statistical Science*, 25(3), 289–310.
- Shmueli, G., & Koppius, O. R. (2011). Predictive analytics in information systems research. *MIS quarterly*, 553–572
- Singh, J. P., Kumar, A., Rana, N. P., & Dwivedi, Y. K. (2020). Attention-based LSTM network for rumor veracity estimation of tweets. *Information Systems Frontiers*. <https://doi.org/10.1007/s10796-020-10040-5>
- Sitkin, S. B. (1992). Learning through failure: The strategy of small losses. *Research in Organizational Behavior*, 14, 231–266.
- Son, G., Hazlewood, V., & Peterson, G. (2014). On automating XSEDE user ticket classification. In Proceedings of the 2014 ACM Annual Conference on Extreme Science and Engineering Discovery Environment, pp. 41–48.
- Splunk. (2019). Quantifying the Staggering Cost of IT Outages. Retrieved December 8, 2020, from <https://priceonomics.com/quantifying-the-staggering-cost-of-it-outages/>
- Swain, A. K. (2016). Mining big data to support decision making in healthcare. *Journal of Information Technology Case and Application Research*, 18(3), 141–154.
- Swain, A. K., & Cao, Q. (2014). Impact of online firm generated content (FGC) on supply chain performance: An empirical analysis. In Proceedings of the 47th Hawaii International conference on system sciences (HICSS), Institute of Electrical and Electronics Engineers (pp. 561–573). IEEE.
- Swain, A. K., & Cao, R. Q. (2019). Using sentiment analysis to improve supply chain intelligence. *Information Systems Frontiers*, 21(2), 469–484.
- Tan, K. H., Wong, W. P., & Chung, L. (2015). Information and knowledge leakage in supply chain. *Information Systems Frontiers*, 18(3), 621–638.
- Uptime Institute (2018). Retrieved January 4, 2021, from Uptime Institute website: <https://uptimeinstitute.com/data-center-outages-are-common-costly-and-preventable>. Accessed 4 Jan 2021.
- Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer.
- Winniford, M., Conger, S., & Erickson-Harris, L. (2009). Confusion in the Ranks: IT Service Management Practice and Terminology. *Information Systems Management*, 26(2), 153–163.
- Wu, D. D., Olson, D. L., & Luo, C. (2014). A decision support approach for accounts receivable risk management. *Systems, Man, and Cybernetics: Systems, IEEE Transactions on*, 44(12), 1624–1632.
- Yap, B. W., Ong, S. H., & Husain, N. H. M. (2011). Using data mining to improve assessment of credit worthiness via credit scoring models. *Expert Systems with Applications*, 38(10), 13274–13283.
- Yuan, R., Li, Z., Guan, X., & Xu, L. (2010). An SVM-based machine learning method for accurate internet traffic classification. *Information Systems Frontiers*, 12(2), 149–156.
- Zurada, J., & Lonial, S. (2011). Comparison of the performance of several data mining methods for bad debt recovery in the healthcare industry. *Journal of Applied Business Research*, 21(2), 37–54.
- Rizk, A., & Elragal, A. (2020). Data science: developing theoretical contributions in information systems via text analytics. *Journal of Big Data*, 7(1), 1–26.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ajaya K. Swain Dr. Ajaya Kumar Swain is an Associate Professor of Quantitative Management and the Director of Assurance of Learning in the Greehey School of Business at St. Mary's University. He received his MS in Industrial Management Systems Engineering from the University of Nebraska-Lincoln, and an MBA in Business Statistics and PhD in Operations Management from Texas Tech University. His research interests include predictive and social media analytics, operations and supply chain performance, leadership, and corporate sustainability. His publications have appeared in *European Journal of Operational Research*, *Operations Research Perspectives*, *Information Systems Frontiers*, *Journal of Manufacturing Processes*, *IEEE Computer Society Journal*, and *Journal of Information Technology Case and Application Research* among others. Dr. Swain was a recipient of the St. Mary's University Distinguished Faculty Award and the Greehey School Business Outstanding Research Award.

Valeria R. Garza Valeria R. Garza is a practicing software engineer in the finance and insurance industry. Her career experience includes the design, development, and production support of reactive Java REST APIs. She received both her bachelor's degree in Industrial Engineering and MBA from St. Mary's University.