**PREDICTING EMPLOYEE ATTRITION USING MACHINE LEARNING**

**Abstract :** The growing interest in machine learning among business leaders and decision makers demands that researchers explore its use within business organisations. One of the major issues facing business leaders within companies is the loss of talented employees. This research studies employee job satisfaction using machine learning models. Using a synthetic data created by IBM Watson, three main experiments were conducted to predict employee job satisfaction. The first experiment involved training the original class-imbalanced dataset with the following machine learning models: support victor machine (SVM) with several kernel functions, random forest and Knearest neighbour (KNN). The second experiment focused on using adaptive synthetic (ADASYN) approach to overcome class imbalance, then retraining on the new dataset using the abovementioned machine learning models. The third experiment involved using manual undersampling of the data to balance between classes. As a result, training an ADASYN balanced dataset with KNN (K = 3) achieved the highest performance, with 0.93 F1-score. Finally, by using feature selection and random forest, F1-score of 0.909 was achieved using 12 features out of a total of 29 features.

**Introduction:**

Employee attrition can be defined as the loss of employees due to any of the following reasons: personal reasons, low job satisfaction, low salary, and a bad business environment.

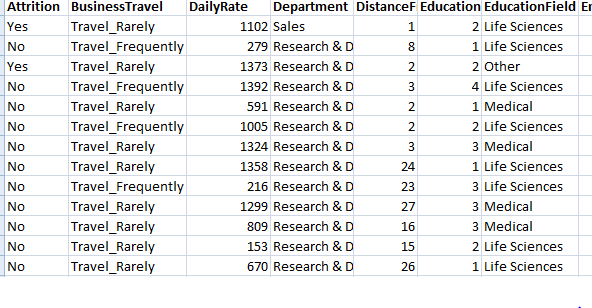
Predicting employees attrition at a company will help management act faster by enhancing their internal policies and strategies. Where talented employees with a risk of leaving can be offered several propositions, such as a salary increase or proper training, to reduce their likelihood of leaving. Using machine learning models can help companies predict employees attrition. Using the historical data kept in human resources (HR) departments, analysts can build and

train a machine learning model that can predict the employees who are leaving the company. Such models are trained to examine the correlation between the features of both active and terminated employees.

**Dataset :**

Dataset used for this paper is IBM HR Analytics Employee Attrition and Performance dataset. It is a fractional dataset created by IBM data scientists which contains the comparable data of employee performance and attrition. The data set contains 35 features like employee age, job role, marital status etc. One of the columns is job satisfaction which is numeric value of rating ranges between 1 to 4. We are going to predict this feature with our proposed algorithms.

The sample of the dataset is shown bellow.



**Algorithms :**

In this paper the model is prepared in two steps. First the imbalanced data is converted to balanced data using SMOTE and Random Sample. Then the data is trained using various Classifier models **like SVM, KNN, Random Forest** , **Naïve Bayes and Artificial Neural** **Network(ANN)**.Then their performance is compared in terms of accuracy, precision and F1 score.

The existing paper uses **ADASYN** for sampling but we are using random sample and **SMOTE** instead. For classification purpose we are using **ANN** and **Naïve bayes** for better comparision.

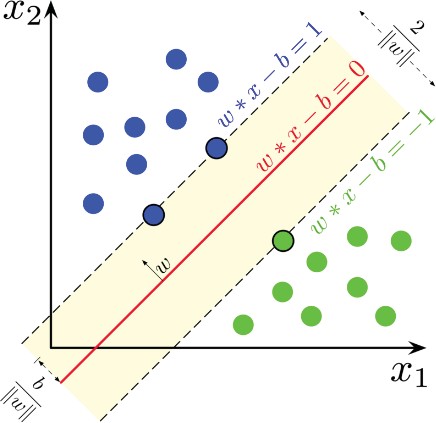
**Modules:**

## Support Vector Machine (SVM) :

Support Vector Machine (SVM) is one of the standard machine learning algorithms used in pattern recognition, spam filtering and anomaly network intrusion detection. SVM can learn the pattern in gives accurate classification by using class labels. The accurate classification is achieved by training machine to classify unknown samples with the training dataset model. SVM has the capability to find the global optimal solution by performing the linear separation finding an optimal hyperplane that separates two classes. The closest data to the hyperplane are support vectors and by getting the features the predicted class is declared.

Given the training dataset of n points of the form

Where “yi” are either 1 or -1 each which indicates the class to the point belongs.

The hyper lane can be written as the set of points xi satisfying w.x-b=0

**KNN (K Nearest Neighbours) Classifier :**

K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.

It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as [GMM](https://en.wikipedia.org/wiki/Mixture_model), which assume a Gaussian distribution of the given data).

We are given some prior data (also called training data), which classifies coordinates into groups identified by an attribute.

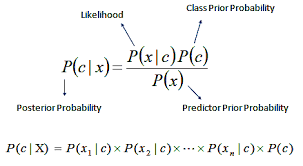
K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970’s as a non-parametric technique.

|  |  |  |
| --- | --- | --- |
| A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor. |  |  |
|  |  |  |

**Naive Bayes classifier :**

Naive Bayes classifiers are a collection of classification algorithms based on **Bayes’ Theorem**. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Bayes’ Theorem finds the probability of an event occurring given the probability of another event that has already occurred. Bayes’ theorem is stated mathematically as the following equation:



**Proposed Methodology:**

**Data Preprocessing :**

The dataset is highly imbalanced and it is converted to balanced data by using upsampling methods like ADASYAN or SMOTE. After that the data is standardized or normalized to avoid over fitting also null values are replaced with zeros. Redundant columns are also removed in this step.

**Data splitting :**

Now 70 % of the data is used for training and 30% is used for testing maintaining the class ratio.

**Training the model :**

As different models are to be used so k-fold cross validation is used for proper model selection.Then we fit our data to the models to evaluate the performance. Also we use different hyper parameters for choosing the best model.

**FRONT END( DJANGO)**

Django is a free and open source web application framework written in Python. A framework is nothing more than a collection of modules that make development easier. They are grouped together, and allow you to create applications or websites from an existing source, instead of from scratch.

This is how websites - even simple ones designed by a single person - can still include advanced functionality like authentication support, management and admin panels, contact forms, comment boxes, file upload support, and more. In other words, if you were creating a website from scratch you would need to develop these components yourself. By using a framework instead, these components are already built, you just need to configure them properly to match your site.

The official project site describes Django as "a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source."

Django offers a big collection of modules which you can use in your own projects. Primarily, frameworks exist to save developers a lot of wasted time and headaches and Django is no different.

You might also be interested in learning that Django was created with front-end developers in mind. "Django’s template language is designed to feel comfortable and easy-to-learn to those used to working with HTML, like designers and front-end developers. But it is also flexible and highly extensible, allowing developers to augment the template language as needed."

If you're going to be working with Python, especially for web applications or web design, you'll want to remember the Django framework. It will certainly come in handy.

CherryPy is another Python-based framework that is great to work with, although it is designed with the absolute minimalist in mind. It's a framework you'll want to explore after you already have some experience working with Python.

**Django helps you write software that is:**

**Complete**

Django follows the "Batteries included" philosophy and provides almost everything developers might want to do "out of the box". Because everything you need is part of the one "product", it all works seamlessly together, follows consistent design principles, and has extensive and up-to-date documentation.

**Versatile**

Django can be (and has been) used to build almost any type of website — from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, XML, etc). The site you are currently reading is based on Django!

Internally, while it provides choices for almost any functionality you might want (e.g. several popular databases, templating engines, etc.), it can also be extended to use other components if needed.

**Secure**

Django helps developers avoid many common security mistakes by providing a framework that has been engineered to "do the right things" to protect the website automatically. For example, Django provides a secure way to manage user accounts and passwords, avoiding common mistakes like putting session information in cookies where it is vulnerable (instead cookies just contain a key, and the actual data is stored in the database) or directly storing passwords rather than a password hash.

A password hash is a fixed-length value created by sending the password through a cryptographic hash function. Django can check if an entered password is correct by running it through the hash function and comparing the output to the stored hash value. However due to the "one-way" nature of the function, even if a stored hash value is compromised it is hard for an attacker to work out the original password.

Django enables protection against many vulnerabilities by default, including SQL injection, cross-site scripting, cross-site request forgery and clickjacking (see Website security for more details of such attacks).

**Scalable**

Django uses a component-based “shared-nothing” architecture (each part of the architecture is independent of the others, and can hence be replaced or changed if needed). Having a clear separation between the different parts means that it can scale for increased traffic by adding hardware at any level: caching servers, database servers, or application servers. Some of the busiest sites have successfully scaled Django to meet their demands (e.g. Instagram and Disqus, to name just two).

**Maintainable**

Django code is written using design principles and patterns that encourage the creation of maintainable and reusable code. In particular, it makes use of the Don't Repeat Yourself (DRY) principle so there is no unnecessary duplication, reducing the amount of code. Django also promotes the grouping of related functionality into reusable "applications" and, at a lower level, groups related code into modules (along the lines of the Model View Controller (MVC) pattern).

**Portable**

Django is written in Python, which runs on many platforms. That means that you are not tied to any particular server platform, and can run your applications on many flavours of Linux, Windows, and Mac OS X. Furthermore, Django is well-supported by many web hosting providers, who often provide specific infrastructure and documentation for hosting Django sites.

**Where did it come from?**

Django was initially developed between 2003 and 2005 by a web team who were responsible for creating and maintaining newspaper websites. After creating a number of sites, the team began to factor out and reuse lots of common code and design patterns. This common code evolved into a generic web development framework, which was open-sourced as the "Django" project in July 2005.

Django has continued to grow and improve, from its first milestone release (1.0) in September 2008 through to the recently-released version 2.0 (2017). Each release has added new functionality and bug fixes, ranging from support for new types of databases, template engines, and caching, through to the addition of "generic" view functions and classes (which reduce the amount of code that developers have to write for a number of programming tasks).

Note: Check out the release notes on the Django website to see what has changed in recent versions, and how much work is going into making Django better.

Django is now a thriving, collaborative open source project, with many thousands of users and contributors. While it does still have some features that reflect its origin, Django has evolved into a versatile framework that is capable of developing any type of website.

**How popular is Django?**

There isn't any readily-available and definitive measurement of popularity of server-side frameworks (although sites like Hot Frameworks attempt to assess popularity using mechanisms like counting the number of GitHub projects and StackOverflow questions for each platform). A better question is whether Django is "popular enough" to avoid the problems of unpopular platforms. Is it continuing to evolve? Can you get help if you need it? Is there an opportunity for you to get paid work if you learn Django?

Based on the number of high profile sites that use Django, the number of people contributing to the codebase, and the number of people providing both free and paid for support, then yes, Django is a popular framework!

High-profile sites that use Django include: Disqus, Instagram, Knight Foundation, MacArthur Foundation, Mozilla, National Geographic, Open Knowledge Foundation, Pinterest, and Open Stack (source: Django home page).

**Is Django opinionated?**

Web frameworks often refer to themselves as "opinionated" or "unopinionated".

Opinionated frameworks are those with opinions about the "right way" to handle any particular task. They often support rapid development in a particular domain (solving problems of a particular type) because the right way to do anything is usually well-understood and well-documented. However they can be less flexible at solving problems outside their main domain, and tend to offer fewer choices for what components and approaches they can use.

Unopinionated frameworks, by contrast, have far fewer restrictions on the best way to glue components together to achieve a goal, or even what components should be used. They make it easier for developers to use the most suitable tools to complete a particular task, albeit at the cost that you need to find those components yourself.

Django is "somewhat opinionated", and hence delivers the "best of both worlds". It provides a set of components to handle most web development tasks and one (or two) preferred ways to use them. However, Django's decoupled architecture means that you can usually pick and choose from a number of different options, or add support for completely new ones if desired.

**What does Django code look like?**

In a traditional data-driven website, a web application waits for HTTP requests from the web browser (or other client). When a request is received the application works out what is needed based on the URL and possibly information in POST data or GET data. Depending on what is required it may then read or write information from a database or perform other tasks required to satisfy the request. The application will then return a response to the web browser, often dynamically creating an HTML page for the browser to display by inserting the retrieved data into placeholders in an HTML template.

Django web applications typically group the code that handles each of these steps into separate files:

**URLs:** While it is possible to process requests from every single URL via a single function, it is much more maintainable to write a separate view function to handle each resource. A URL mapper is used to redirect HTTP requests to the appropriate view based on the request URL. The URL mapper can also match particular patterns of strings or digits that appear in an URL, and pass these to a view function as data.

**View:** A view is a request handler function, which receives HTTP requests and returns HTTP responses. Views access the data needed to satisfy requests via models, and delegate the formatting of the response to templates.

**Models:** Models are Python objects that define the structure of an application's data, and provide mechanisms to manage (add, modify, delete) and query records in the database.

**Templates:** A template is a text file defining the structure or layout of a file (such as an HTML page), with placeholders used to represent actual content. A view can dynamically create an HTML page using an HTML template, populating it with data from a model. A template can be used to define the structure of any type of file; it doesn't have to be HTML!

**BACKEND:**

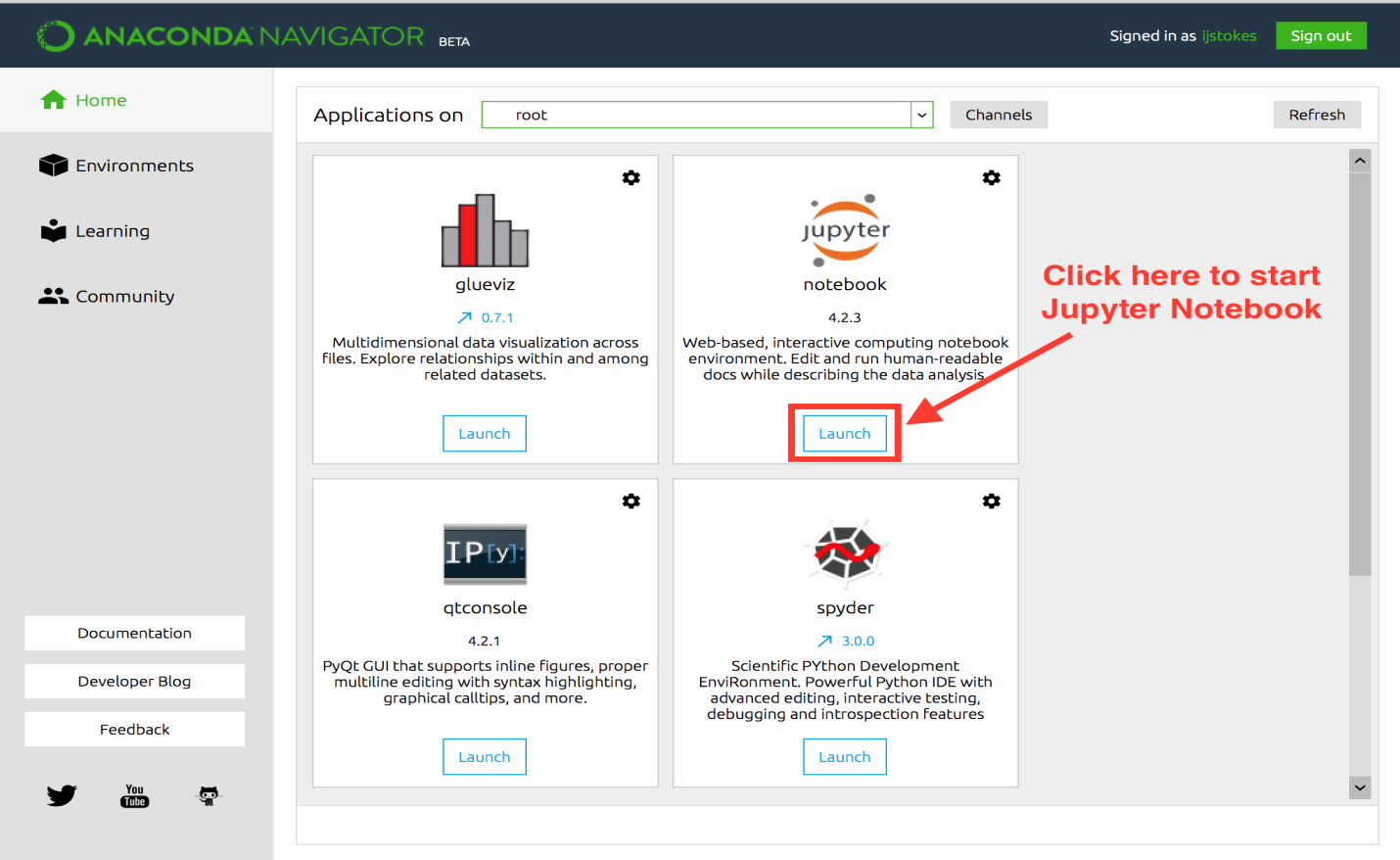
**ANACONDA**

It is a free and open-source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc.), that aims to simplify package management and deployment.

**Anaconda distribution** comes with more than 1,500 packages as well as the [Conda](https://en.wikipedia.org/wiki/Conda_(package_manager)) package and virtual environment manager. It also includes a GUI, **Anaconda Navigator**, as a graphical alternative to the Command Line Interface (CLI).

The big difference between Conda and the pip package manager is in how package dependencies are managed, which is a significant challenge for Python data science and the reason Conda exists. Pip installs all Python package dependencies required, whether or not those conflict with other packages you installed previously. So your working installation of, for example, GoogleTensor flow, can suddenly stop working when you pip install a different package that needs a different version of the Numpy library. More insidiously, everything might still appear to work but now you get different results from your data science, or you are unable to reproduce the same results elsewhere because you didn't pip install in the same order.

Conda analyzes your current environment, everything you have installed, any version limitations you specify (e.g. you only want tensor flow>= 2.0) and figures out how to install compatible dependencies. Or it will tell you that what you want can't be done. Pip, by contrast, will just install the thing you wanted and any dependencies, even if that break other things. Open source packages can be individually installed from the Anaconda repository, Anaconda Cloud (anaconda.org), or your own private repository or mirror, using the **conda install** command. Anaconda Inc compiles and builds all the packages in the Anaconda repository itself, and provides binaries for Windows 32/64 bit, Linux 64 bit and MacOS 64-bit. You can also install anything on PyPI into a Conda environment using pip, and Conda knows what it has installed and what pip has installed.Custom packages can be made using the **conda build** command, and can be shared with others by uploading them to Anaconda Cloud, [PyPI](https://en.wikipedia.org/wiki/Python_Package_Index) or other repositories.The default installation of Anaconda2 includes Python 2.7 and Anaconda3 includes Python 3.7. However, you can create new environments that include any version of Python packaged with conda.

****

**Anaconda Navigator** is a desktop [Graphical User Interface (GUI)](https://en.wikipedia.org/wiki/Graphical_user_interface) included in Anaconda distribution that allows users to launch applications and manage conda packages, environments and channels without using [command-line commands](https://en.wikipedia.org/wiki/Command-line_interface). Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages and update them. It is available for [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux).

The following applications are available by default in Navigator

* [Jupyter Notebook](https://en.wikipedia.org/wiki/Project_Jupyter#Jupyter_Notebook)
* [QtConsole](https://qtconsole.readthedocs.io/en/latest/)
* [Spyder](https://en.wikipedia.org/wiki/Spyder_(software))
* [Glueviz](http://glueviz.org/)
* [Orange](https://en.wikipedia.org/wiki/Orange_(software))
* [Rstudio](https://en.wikipedia.org/wiki/Rstudio)
* [Visual Studio Code](https://en.wikipedia.org/wiki/Visual_Studio_Code)

**VISUAL STUDIO**

Microsoft **.NET** is a set of Microsoft software technologies for rapidly building and integrating XML Web services, Microsoft Windows-based applications, and Web solutions. The .NET Framework is a language-neutral platform for writing programs that can easily and securely interoperate. There’s no language barrier with .NET: there are numerous languages available to the developer including Managed C++, C#, Visual Basic and Java Script. The .NET framework provides the foundation for components to interact seamlessly, whether locally or remotely on different platforms. It standardizes common data types and communications protocols so that components created in different languages can easily interoperate.

“.NET” is also the collective name given to various software components built upon the .NET platform. These will be both products (Visual Studio.NET and Windows.NET Server, for instance) and services (like Passport, .NET My Services, and so on).

Microsoft **VISUAL STUDIO** is an Integrated Development Environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps

**LITERATURE SURVEY :**

# 1. TITLE: HISTOPATHOLOGICAL IMAGE ANALYSIS: A REVIEW

# AUTHORS: S. Kaur and R. Vijay

# ****Published in:****Imperial Journal of Interdisciplinary Research, vol. 2, no. 8, 2016.

# DESCRIPTION:

The Employee turnover has always been a crucial matter of concern for organizations. In today’s era of globalization there are ample opportunities for talented people in this world,    therefore, employees are inclined to move from one organization to another. Due to this Corporates are facing the problem of attrition in the world of economic revival. A large degree of employee turnover is highly deleterious to both the organization as well as the employees. How to reduce employees attrition is a decisive challenge for the HR managers. Lucrative incentives and motivational theories have become useless and are considered as old practices of the human resource management. This article presents a holistic view of attrition and retention of employees in this competitive scenario with reference to Retail Industry. Along with other factors, Job Satisfaction has been considered as the major source of attrition and retention. The relevant literature review has been done for compiling this research paper in order to find out the various factors responsible for the attrition of employees in the retail sector. The research is based on the relevant literature review and also from the secondary data available on the internet.

# 2. TITLE: THE EFFECTS OF PAY LEVEL ON ORGANIZATION-BASED SELF-ESTEEM AND PERFORMANCE: A FIELD STUDY

# AUTHORS : D. G. Gardner, L. V. Dyne and J. L. Pierce

# ****Published in:****Journal of Occupational and Organizational Psychology, vol. 77, no. 3, pp. 307-322, 2004.

# DESCRIPTION:

# Most compensation managers implicitly assume (or perhaps hope) that high pay levels will maintain and enhance future performance. To date, this assumption has been largely untested. Given the importance of pay level and the large expense that pay represents to most organizations, understanding how and why pay level influences the behaviour of employees in organizations is an important question. The purpose of this study is to examine the motivational effects of pay level on employee performance. To examine these issues, we collected field study data from a variety of sources, at three different times, and assessed the effects of employee pay level on subsequent self-esteem and performance. Specifically, we hypothesized that the effects of pay level on performance would be mediated by pay level effects on organization-based self-esteem. We base this hypothesis on the premise that level of pay within an organization communicates a sense of how much the organization values an employee and thus affects employee organization-based self-esteem which, in turn, enhances job performance. After controlling for organization tenure, and previous pay change, results supported a mediated model that suggests that pay level affects employee self-esteem, which in turn, affects employee performance.

# 3. TITLE: AN EXPLORATORY STUDY OF US LODGING PROPERTIES' ORGANIZATIONAL PRACTICES ON EMPLOYEE TURNOVER AND RETENTION

# AUTHORS : E. Moncarz, J. Zhao and C. Kay

# ****Published in:****International Journal of Contemporary Hospitality Management, vol. 21, no. 4, pp. 437-458, 2009.

# DESCRIPTION:

# The purpose of this paper is to investigate US lodging properties' organizational employee-retention initiatives and practices, and to examine the impact of those initiatives on employee turnover and retention. Design/methodology/approach--Using the Directory of Hotel & Lodging Companies, a convenient sample group of 24 management companies are selected. A self-administered mail survey instrument is developed to measure and test organizational initiatives and practices on employee turnover and retention. Using SPSS 16.0, two statistical tests are employed to test study hypotheses. Correlation analysis is used to identify the relationships between predictor and response variables. Likewise, regression analysis is used to examine the relationships between predictor and response variables hypothesizing that the effectiveness of practicing the human resource management organizational initiatives on management and non-management retention and turnover will differ. Findings--The findings reveal that Corporate Culture, Hiring and Promotions and Training practices influence non-management employee retention. At the same time, Hiring and Promotion practices impact management retention, as well. Moreover, Organizational Mission, Goals and Direction, and Employee. Recognition, Rewards and Compensation were found to positively reduce non-management employee turnover. Research limitations/implications--Owing to the study methodology and the relatively low response rate, generalization of the study findings is limited. Future replication studies are recommended. Practical implications--The findings will equip lodging organizations and industry professionals with the contemporary tools to proactively reduce employee turnover and for maintaining employee retention. This should have a positive impact on workforce productivity. Originality/value--This study makes a major contribution to the relative influence of the practice of eight study-defined organizational initiatives on turnover in lodging businesses. [ABSTRACT FROM AUTHOR] Copyright of International Journal of Contemporary Hospitality Management is the property of Emerald Group Publishing Limited and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use. This abstract may be abridged. No warranty is given about the accuracy of the copy. Users should refer to the original published version of the material for the full abstract.

# 3. TITLE: USING DATA MINING TECHNIQUES TO BUILD A CLASSIFICATION MODEL FOR PREDICTING EMPLOYEES PERFORMANCE

# AUTHORS : A. Al-Radaideh and E. A. Nagi

# ****Published in:** I**nternational Journal of Advanced Computer Science and Applications, vol. 3, no. 2, p. 144–151 , 2012.

# DESCRIPTION:

# Human capital is of a high concern for companies' management where their most interest is in hiring the highly qualified personnel which are expected to perform highly as well. Recently, there has been a growing interest in the data mining area, where the objective is the discovery of knowledge that is correct and of high benefit for users. In this paper, data mining techniques were utilized to build a classification model to predict the performance of employees. To build the classification model the CRISP-DM data mining methodology was adopted. Decision tree was the main data mining tool used to build the classification model, where several classification rules were generated. To validate the generated model, several experiments were conducted using real data collected from several companies. The model is intended to be used for predicting new applicants' performance.

# 4. TITLE: Employee churn prediction

# AUTHORS : G. K. P. V. Vijaya Saradhi

# ****Published in:** Expert Systems with Applications,, vol. 38, no. 3, pp. 1999-2006, 2011**.

# DESCRIPTION:

# Employee churn prediction which is closely related to customer churn prediction is a major issue of the companies. Despite the importance of the issue, there is few attention in the literature about. In this study, we applied well-known classification methods including, Decision Tree, Logistic Regression, SVM, KNN, Random Forest, and Naive Bayes methods on the HR data. Then, we analyze the results by calculating the accuracy, precision, recall, and F-measure values of the results. Moreover, we implement a feature selection method on the data and analyze the results with previous ones. The results will lead companies to predict their employees' churn status and consequently help them to reduce their human resource costs.

# 5. TITLE: ANALYZING EMPLOYEE ATTRITION USING DECISION TREE ALGORITHMS

# AUTHORS : D. A. B. A. Alao

# ****Published in :** Computing, Information Systems, Development Informatics and Allied Research Journal, no. 4, 2013**

# DESCRIPTION:

# Employee turnover is a serious concern in knowledge based organizations. When employees leave an organization, theycarry with them invaluable tacit knowledge which is often the source of competitive advantage for the business. In order foran organization to continually have a higher competitive advantage over its competition, it should make it a duty to minimizeemployee attrition. This study identifies employee related attributes that contribute to the prediction of employees’ attritionin organizations. Three hundred and nine (309) complete records of employees of one of the Higher Institutions in Nigeriawho worked in and left the institution between 1978 and 2006 were used for the study. The demographic and job relatedrecords of the employee were the main data which were used to classify the employee into some predefined attrition classes.Waikato Environment for Knowledge Analysis (WEKA) and See5 for Windows were used to generate decision tree modelsand rule-sets. The results of the decision tree models and rule-sets generated were then used for developing a a predictivemodel that was used to predict new cases of employee attrition. A framework for a software tool that can implement therules generated in this study was also proposed.Keywords: Employee Attrition, Decision Tree Analysis, Data Mining.