COMP8590 Statistical Learning

Final Project

InterSummer 2021

Topic: Model Selection for Classification of Useless or Orphan Node Package Manager Repositories

Group Members:

Shafkat Waheed - 110060302

Tanya Agarwal - 110060426

Sridhar Gopu – 110059408

Roxy Leonard Palma - 110037154

Submitted To:

Dr. Alioune Ngom

School of Computer Science

INDEX

1. Abstract:

As the popularity of the JavaScript language is constantly increasing, one of the most important challenges today is to assess the quality of JavaScript packages. Developers often employ tools for code linting and for the extraction of static analysis metrics to assess and/or improve their code. In an open source and popular ecosystem like that of JavaScript and the npm registry, there exist many problems that make crawling and processing challenging, especially if one would like to follow a systematic and pristine analysis process. Some of the typical problems encountered with the npm registry are a) The declared GitHub URL of a certain npm package leads to a not found page, b) The declared GitHub URL of a certain npm package redirects to a different (to the one declared) GitHub page (this is a maintenance issue of the package. json file), c) many npm packages contain copied-pasted popular open-source projects with only the package name changed in the package.

Keywords: Npm, AWS, SVM, Deprecation, Classification, Statistical learning, clustering, dimension reduction,

2. Introduction:

If we talk about the language-based package repository growth over the last few years, npm has always come out on top every year. Even the open-source developers from every continent use npm to share and borrow packages, and many organizations use npm package to manage private development as well. Node package manager (NPM) is de facto package manager of NodeJS. Everyday new packages are being uploaded in that repository but navigating and understanding it is a challenging task or new developers. To mitigate this issue, it is imperative e know which package could go deprecated. So, to properly classify the npm package beforehand could reduce development and operation cost.

3. Problem Statement:

(a) Problem Definition:

Since npm is an open-source library repository for developers, any person can write abstract coding and submit it as a library package. But after few years, either due to availability of better standardization of library files by well recognized organizations or innovative approach for that same library has been developed by someone else or the developer just abandons the project in the middle due to assorted reasons, the project dependency for the users gets affected indirectly. In real time cases such as protractor framework, which was a well-known automation framework library from npm has been abandoned 2 years back for no reasons, this abandonment of libraries has been a huge problem in the open-source library managers like npm.

(b) Motivation:

As software developers, we have experienced library packages from one of the popular package managers called Node Package Manager is getting abandoned after several years due to distinct reasons. We wanted to apply and understand the implications of machine learning on deprecated library packages.

(c) Justifications:

Using this approach, we can get to know the lifetime of a library package beforehand and get to choose the reliable package for integrating in our software projects or update with better packages from the market as time goes on.

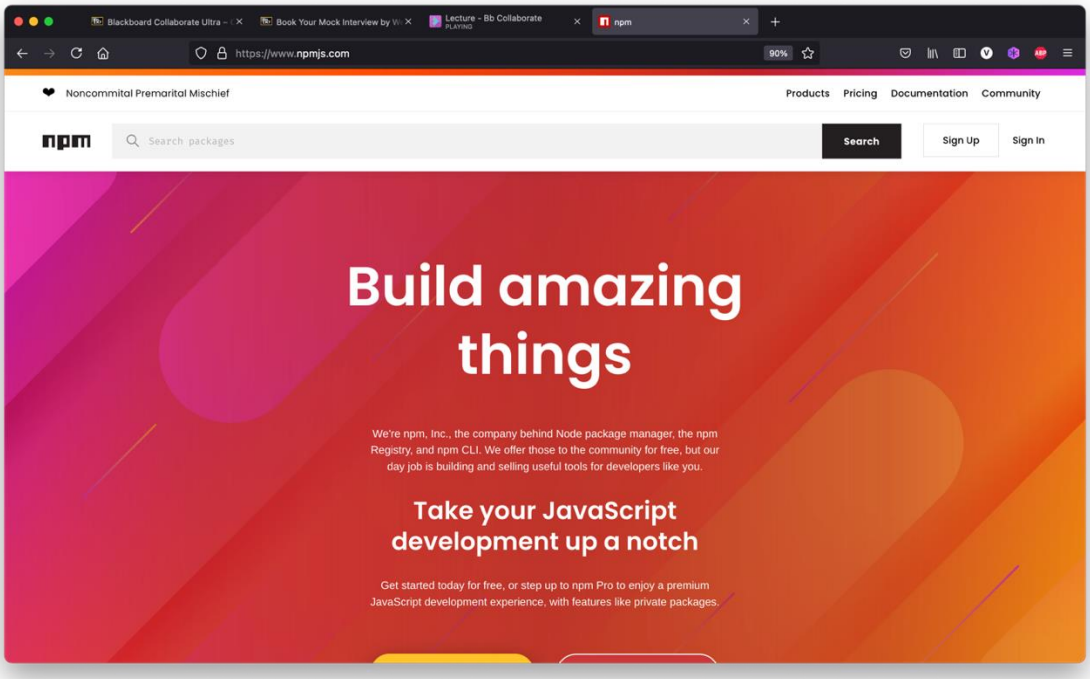
4. Related Works:

In this paper, polynomial, sigmoid and gaussian kernels have been used to classify the data. The sigmoid kernel was quite popular for support vector machines due to its origin from neural networks. Although it has been observed that the kernel matrix may not be semi definite as its properties have not been studied fully[1]. One example shows that the sigmoid kernel matrix is conditionally positive definite in certain parameters and thus are valid kernels there. On the other hand, gaussian kernel is another popular Kernel method used in SVM models for more, it has shown the best performance in terms of accuracy[2]. There is no theoretical statement which states one kernel is better than another or guarantees for one kernel to work better than the other. That is a-priori you never know, nor you can find out which kernel will work better. T-sne is data visualization techniques through which we can visualize data of high dimension [3]. Birch is an efficient clustering algorithm that is suitable for large to medium datasets[4]. We explored datasets using t-sne and birch and used svm which excels at higher dimension data to classify our npm data.

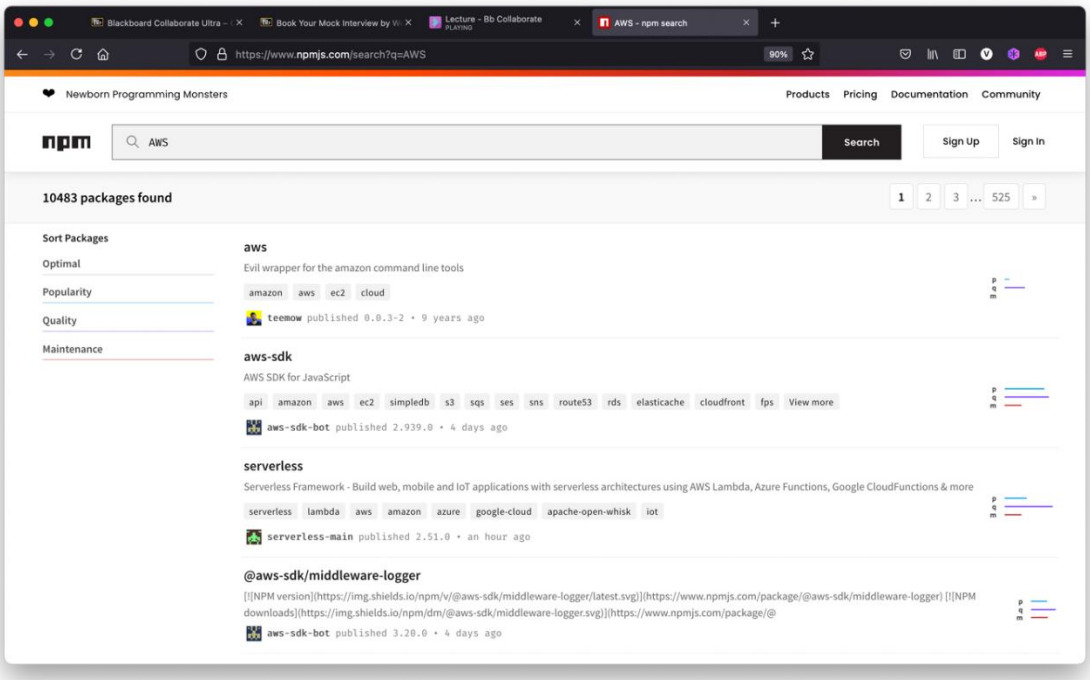
5. Methodology

Materials and Data:

The tool that we have used is: The npm repository can be accessed using the URL (https://www.npmjs.com/), where the developers can enter the search keyword in the search field and gets listed with all the packages related to the search keyword.



The keyword we used for data collection is “AWS”, where Amazon Web Services (AWS) is a cloud service provider for all resources and services needed for application development and deployment in the production. Totally, there is about 10483 packages released in the node package manager.



More information about the tool: The tool is built using one of the popular web automation frameworks from the npm manager called “protractor” to fetch the data from the UI screens using the browsers. This is still under development and the tool will be ported to a new framework in the upcoming days since protractor has been abandoned 2 years back.

The types of data retrieved from the response data for the search keyword are as follows:

"formData__search__|,"formData__search__|__value,objects__package__name,objects__package__scope,objects__package__version,objects__package__description,objects__package__keywords__001,objects__package__keywords__002,objects__package__keywords__003,objects__package__keywords__004,objects__package__date__ts,objects__package__date__rel,objects__package__links__npm,objects__package__links__homepage,objects__package__links__repository,objects__package__links__bugs,objects__package__author__name,objects__package__author__email,objects__package__author__username,objects__package__publisher__name,objects__package__publisher__avatars__small,objects__package__publisher__avatars__medium,objects__package__publisher__avatars__large,objects__package__publisher__created__ts,objects__package__publisher__created__rel,objects__package__publisher__email,objects__package__maintainers__username,objects__package__maintainers__email
,objects__package__keywordsTruncated,objects__flags__unstable,objects__score__final,objects__score__detail__quality,objects__score__detail__popularity,objects__score__detail__maintenance,objects__searchScore,total,time,pagination__perPage,pagination__page,url,user,csrftoken,npmExpansions,isNpme,objects__package__keywords__005,objects__package__keywords__006,objects__package__keywords__007,objects__package__keywords__008,objects__package__keywords__009,objects__package__keywords__010,objects__package__keywords__011,objects__package__keywords__012,objects__package__keywords__013,objects__package__keywords__014,objects__package__author__url

Data Selection:

Most of the data collected had little importance but among the data scraped few predicates had numeric value and categorial value that were meaningful. So, we choose 6

predicates initially to build the system. The predicates we choose were

1) Initial creation time stamp - This column contained timeslot when the npm repository was initially registered

2) Search Score – This column had values that showed which how much the package was searched

3) maintenance -- This column had the values that represented how the package was maintained

4) popularity – This column had Numeric data which showed which package was populare

5) quality – This column showed what was the quality of the package

6) final – A score to average its value and implications in the NPM

7) Response – 0,1 denoting deprecated and non-deprecated package

To generate missing data, we used **interpolation and the concept of k-nearest Neighbour**. The reason being that data which were collected had close relation with its corresponding predicates. Therefore, the datapoints close to each other had some form of similarity.

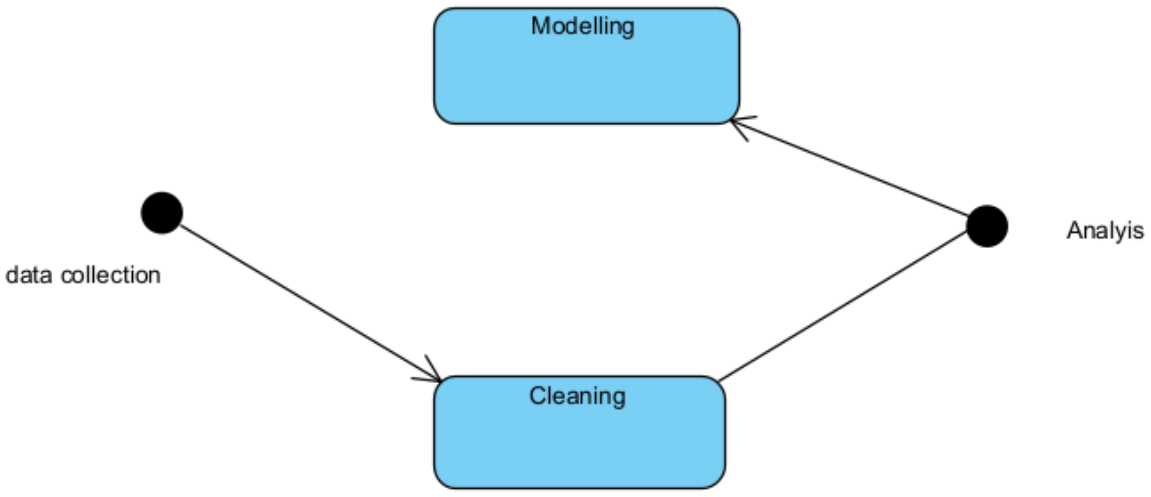
For example, datapoint which had good search score and quality would have good popularity

We had a data of 4908 and we split it into train and test. As the data was small, we did not use validation set. We split the data int ration of 80/20.

Precondition and Assumption

- Our Data was collected from the Node Package Manager(npm).
- We collected data for only Amazon Web Services Packages
- The data contained natural languages, but we only focused on the numeric data.

Followed Method



The methods we followed is a process through which we can find predictive modelling algorithms for data that is not properly cleaned or organized. This method helped us to analyze data that is on the internet full of noises. This method uses an agnostic approach to find patterns in data to decide on a predictive modelling.[5]

In the method we follow and traverse through a process of intuitive visualization to determine what form of predictive model would better suit collected raw data. The process has four steps collection, cleaning, exploratory data analysis and predictive modelling. [6]

6. Computational Experiments:

Implementation (toolkits and software)

The tools used for data collection.

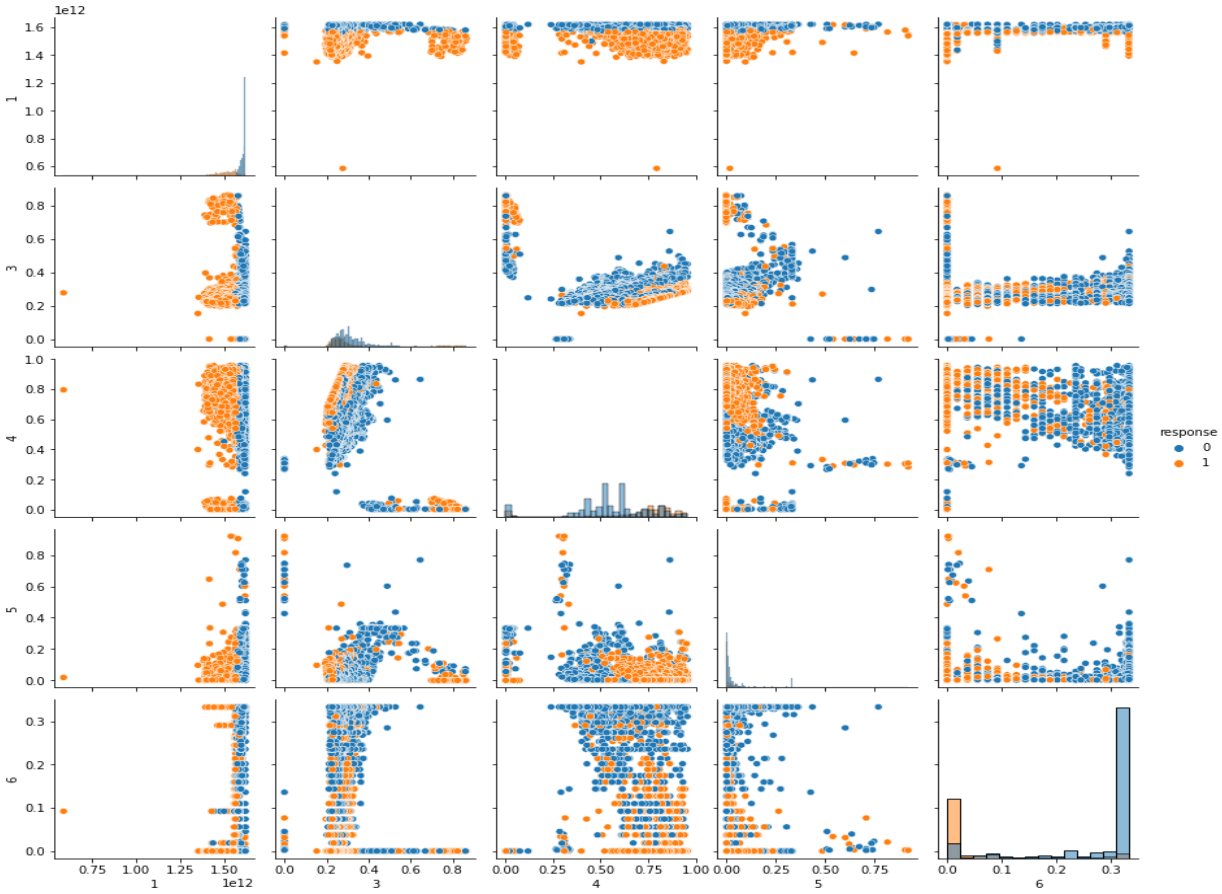
- Open Google browser
- Search for the URL (https://www.npmjs.com/)
- Enter the keyword *AWS* in the search field
- Click Search button
- Open Chrome Devtools
- Navigate to network tab
- Click on each page number from pagination option
- For each click on the previous steps, request and response will be listed in the network tab
- The script will collect response data from each API request and save it to the .csv file
- Once all the pages are navigated by the script, it will close the browser and exit from the execution.
- The dataset required for implementing this model was not available in any public or priced repositories, hence we built a tool on our own to interact with the npm manager to collect and store the required data. https://github.com/vikkysri77/Scraptractor

The tools used for experiments and analysis.

- Sk.learn
- Colaboratory
- T-Sne,SVM,Clustering-birch
- Cloud Computing
- Pandas,numpy

Exploratory Data analysis or Experiments:

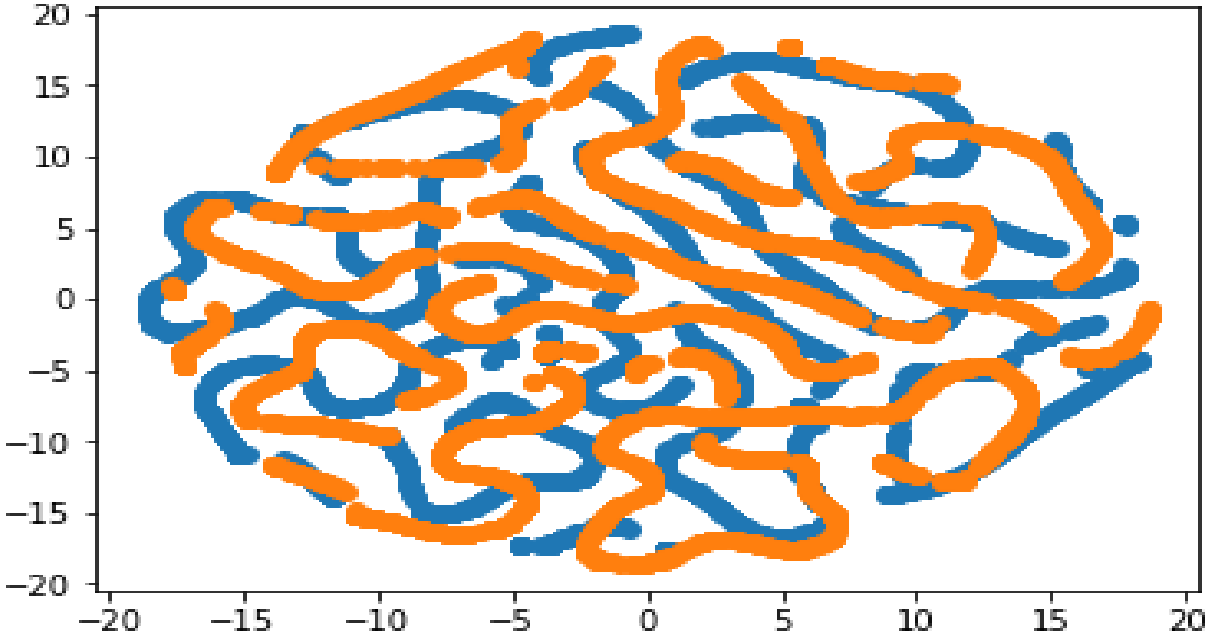
In order to understand the data, we initially started with the pair plot to understand how the predicates interact with each other.



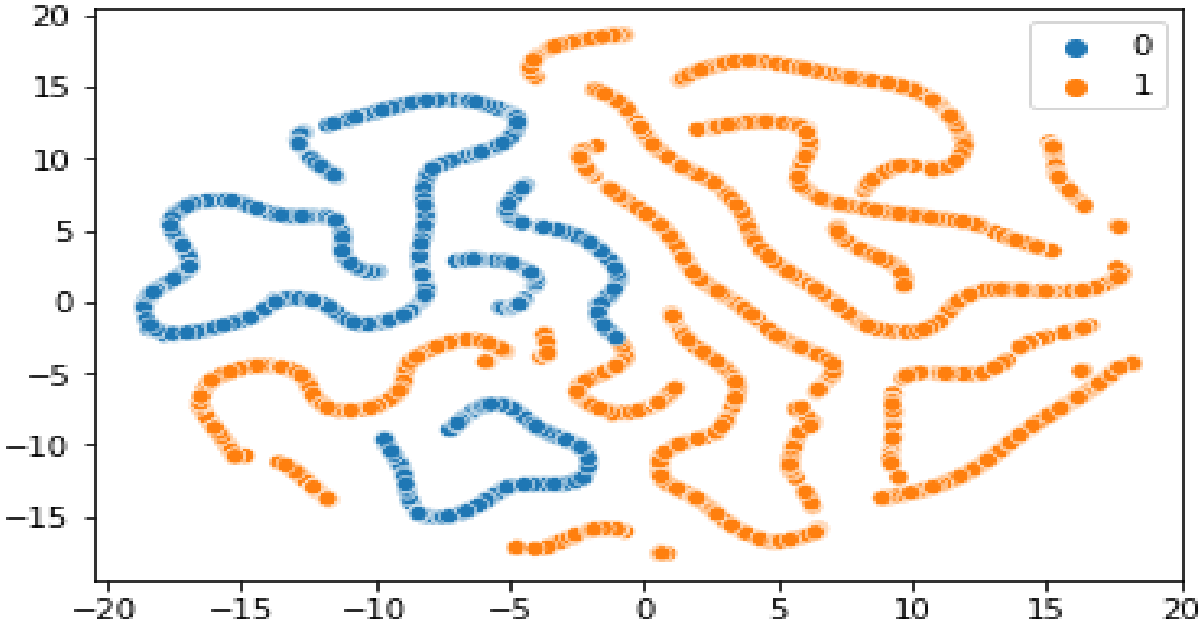
From the above diagram we can get some idea that how the two class are distributed when plotted against two predicates.

To properly understand the data and plot we used t-sne ad dimensionality reduction techniques for non-linear data and clustering. From the Par-plot we got some idea regarding how the data is distributed we found the data to be non-linear visually we applied t-sne to reduce the dimension then connected clustering from the original data to understand if the data is separable into two classes [8]. As SVM builds supports vectors two separate the data our idea was to visualize that separation to properly pinpoint the svm kernel.

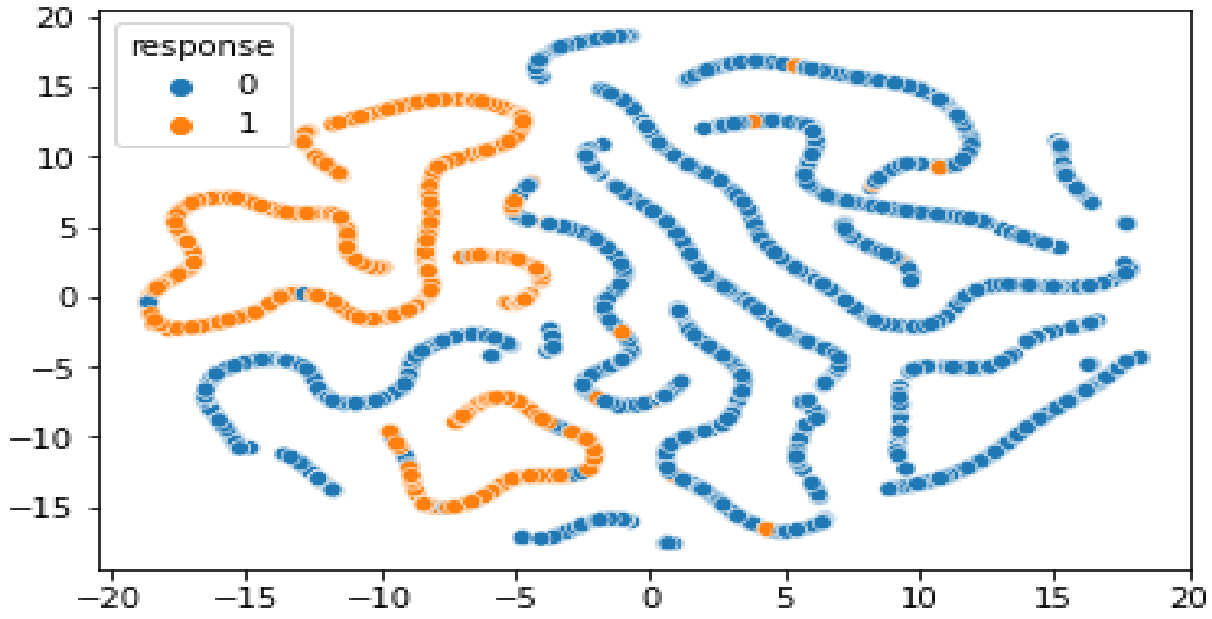
T-SNE – to reduce data into two dimensions



Clustering and the response of the data



Here we can see how clustering has divided the data into two segments. We used birch as a clustering to find if the data is separable



In the above figure we plotted the response of our dataset, and we can clearly see how its related to the clustering and how it's possible to separate the data. We can also see some outliers in the data more visually than ever before.

So, in the case we are unable to extract the response we can easily use clustering to find if the package was deprecated or not.

Predictive Modelling

The Support Vector Machine (SVM) is like a sharp knife – it works on smaller datasets, but on the complex ones, it can be much stronger and powerful in building machine learning models. It is a supervised ML (Machine Learning) algorithm that is often used for regression or classification related problems. The sigmoid kernel was quite popular for support vector machines due to its origin from neural networks. It requires labelled data sets.[7] The vector machine demands to plot each data item as a point. The plotting is done in an n-dimensional space where n is the number of features of a particular data.

As the data is non-linear, we used three non-linear kernel to build our predictive model, the reason for choosing svm is because, after the experiments and exploratory data analysis we can visually represent the distinction in the data. Therefore, it would be possible for svm to find its support vectors. We used three kernels to test our support vector machines.

For evaluating the Predictive models, we used

Confusion Matrix – We are using the confusing matrix to so actually see how many data point was properly classified. As the data set was small, we tried to understand what were the false positive, false negative, true negative and true positive.

Weighted Average - To calculate weighted accuracy, we must take several data and multiply it by a predefined weight. The calculated result emphasizes on the vital pieces of information that is gathered from the dataset. It has some advantages and one of them played an important part on our project. For instance, when the relationship could not be identified by a handful of data it provided flexible evaluation. As the response we got from the data was quite unbalanced it was amazingly effective. [9]

Results

Name	0	1	Weighted average (Accuracy)
	663	7	0.94
Poly	49	263	
	670	0	0.47
Sigmoid	312	0	
	666	4	0.92
Gaussian	77	235	

From, the results its evident that poly and gaussian kernel perform the best compared to sigmoid. One of the reasons it performs so well is because the data was non-linear.

7. Conclusion

So, we looked at how clustering and t-sne can be used as a method of guidance to choose a predictive model for classifying npm package data accurately as possible. t-sne visualizes high-dimensional data by giving each datapoint a location in a two or three-dimensional map.

Future Work

We only applied the algorithm on packages of Amazon Web services. We focused on a subdomain of a rather than the complete domain. To properly understand the idea of the of package deprecation it is necessary to collect data from other services and see its implications.

Open Problem

We applied different form of support vector machines to see its result, but SVM is more efficient for higher dimensions and does not perform well in overlapped classes, in this case the other algorithms could produce even better results. The other algorithms include ensemble methods and multi layered perceptron.

8. References

1] H.-T. Lin and C.-J. Lin, "A Study on Sigmoid Kernels for SVM and the Training of non-PSD Kernels by SMO-type Methods," p. 32.

https://ieeexplore.ieee.org/abstract/document/7319698?casa_token=tbLswhNfV-UAAAAA:HO5cl305IlBwVQiQIH4D96XrVnt45EcoHEv_7Ue8iPX3RXA-2vO_mPupj2XbV3jeRxEUGvCUUX8

[2] M. G. Genton, "Classes of Kernels for Machine Learning: A Statistics Perspective," p. 14.

[3] "t-SNE: The effect of various perplexity values on the shape — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/auto_examples/manifold/plot_t_sne_perplexity.html#sphx-glr-auto-examples-manifold-plot-t-sne-perplexity-py (accessed Jul. 07, 2021).

[4] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: an efficient data clustering method for very large databases," ACM SIGMOD Rec., vol. 25, no. 2, pp. 103–114, Jun. 1996, doi: 10.1145/235968.233324.

[5] "(PDF) Data Product: Analysis, Visualization and Prediction." https://www.researchgate.net/publication/314059796_Data_Product_Analysis_Visualization_and_Prediction?enrichId=rgreq-30897baf6221209263ce3442ab6a4573-XXX&enrichSource=Y292ZXJQYWdlOzMxNDA1OTc5NjtBUzo0NjU4OTU5MDM1MDIzMzZAMTQ4ODA4OTY0NzM3OQ%3D%3D&el=1_x_3&_esc=publicationCoverPdf (accessed Jul. 07, 2021).

[6] https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf

[7] https://www.hindawi.com/journals/mpe/2013/928054/

[8] http://www.cs.toronto.edu/~hinton/absps/tsne.pdf

[9] F. B. A. G. is a forex trading expert who has 20+ years of experience, I. D. R. for A. Trading, risk, money management decisions made at A. L. H. has earned a bachelor's degree in biochemistry, an M. from M.S.U, and is also registered commodity trading advisor L. about our editorial policies A. Ganti, "Weighted Average Definition," Investopedia. https://www.investopedia.com/terms/w/weightedaverage.asp (accessed Jul. 08, 2021).