C951 — Introduction to Artificial Intelligence

Task 3

Joseph T. Lapp

Western Governors University

June 4, 2022

Note: WGU dictated the structure of this document. The assignment was to create a proposal for a machine learning solution to a hypothetical business need, but I found a real world need to address with a real world audience.

Table of Contents

A1. ORGANIZATIONAL NEED 3

A2. PROJECT CONTEXT AND BACKGROUND 3

A3. REVIEW OF BACKGROUND WORKS 6

A4. SUMMARY OF THE MACHINE LEARNING SOLUTION 11

A5. BENEFITS OF THE MACHINE LEARNING SOLUTION 12

B1. SCOPE OF PROJECT 12

B2. GOALS, OBJECTIVES, AND DELIVERABLES 14

B3. STANDARD METHODOLOGY 15

B4. TIMELINE AND MILESTONES 17

B5. RESOURCES AND COSTS 18

B6. CRITERIA FOR SUCCESSFUL EXECUTION OF PROJECT 18

C1. HYPOTHESIS 19

C2. ANALYTICAL METHOD 19

C3. TOOLS AND ENVIRONMENTS OF SOLUTION 23

C4. MEASURING PERFORMANCE 24

D1. SOURCE OF DATA 25

D2. DATA COLLECTION METHOD 26

D3. QUALITY AND COMPLETENESS OF DATA 26

D4. PRECAUTIONS FOR SENSITIVE DATA 27

E. SOURCES 28

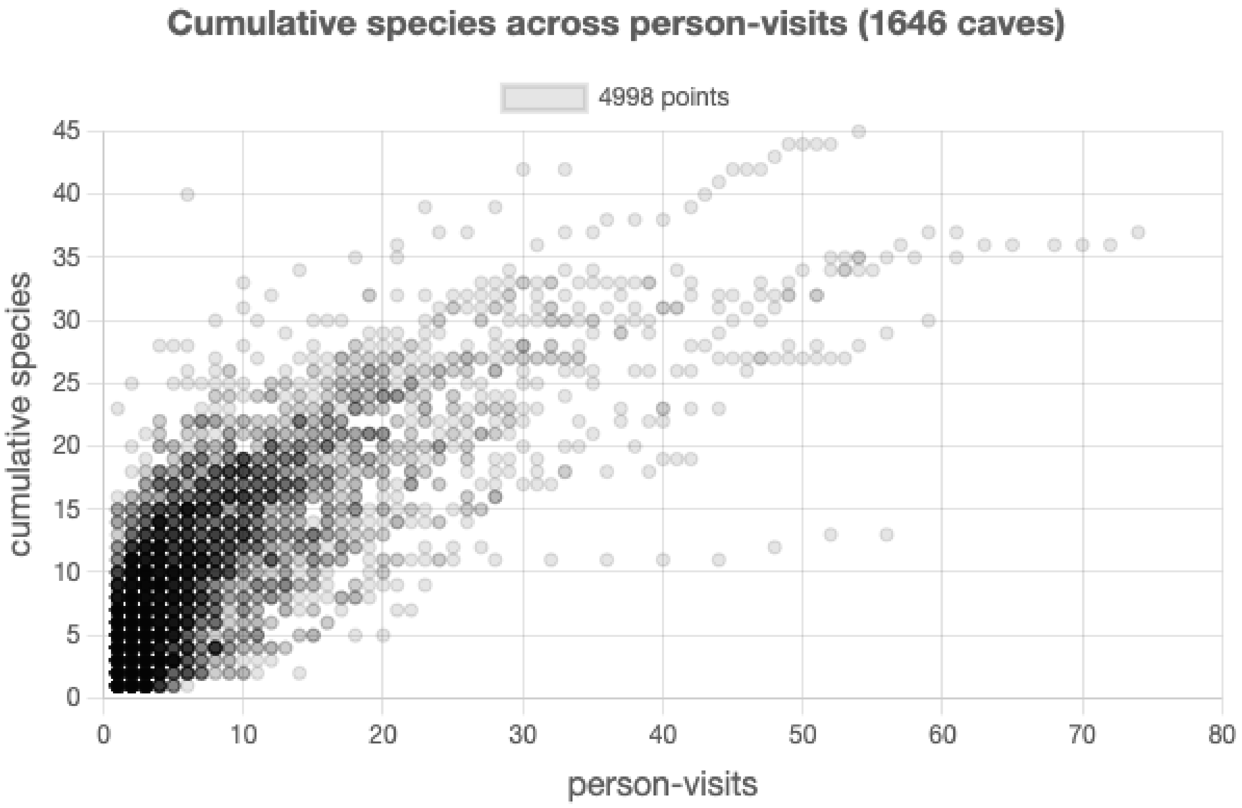
A1. ORGANIZATIONAL NEED

One of the responsibilities of Texas Parks and Wildlife (TPW) is to monitor and protect endangered species in Texas. Some of these species are found in Texas' many caves. Researchers periodically collect specimens from these caves for experts to identify, and once identified, the specimens are preserved and deposited in research collections. With each sampling of a cave, the better known the fauna become and the more effort required to find species not previously found in the cave. Whatever species remain undiscovered may be the rarest and most in need of protection. TPW would like to know how much effort might be necessary to find the next previously-unknown species in any cave, allowing them to decide which caves would most readily benefit from further sampling. The present project proposes a way to estimate the minimum effort necessary to find additional species in Texas caves.

A2. PROJECT CONTEXT AND BACKGROUND

Most of the invertebrates collected from Texas caves are kept in the biospeleology collection at the University of Texas at Austin, which happens to be the largest cave-centric invertebrate collection in the world. Cave researcher James R. Reddell, now retired, has maintained the collection for decades and kept the data on its specimens in a spreadsheet. I was hired to normalize the data, upload it into a standard biological collections database, and create a website that TPW and other researchers could use to study cave fauna and make decisions about cave conservation. After reading the requirements for the present task and for the capstone, I proposed using this data to help researchers understand the benefit of additional sampling of caves, though as an unpaid side project. The stakeholders readily agreed to the project.

The data associates each species with the cave in which it was collected, the date on which it was collected, the number of people participating in the collection expedition, and with other information. With each collecting trip to a cave, the number of species known for the cave either remains the same or increases. Typically in biological sampling, the more participants collecting, the more specimens collected and the more species found. Plotting cumulative species found against person-visits for all caves in Texas therefore seems like a reasonable place to start this investigation. That plot follows.



This plot requires a few clarifications. The darker the point in this plot, the more data that overlaps at that point. The “person-visits” axis indicates the number of visits by individuals to a cave. For example, a single-person expedition to a cave increases the person-visits count by one, while a three-person expedition to a cave increases the person-visits count by three. The cumulative species counts indicate the minimum number of species certain to be in the cave rather than the number of species actually identified for the cave.

This last point requires elaboration and will be important later. Biologists identify organisms by filing each into the taxonomic hierarchy of kingdom, phylum, class, order, family, genus, species, and subspecies. Each of these levels is known as a "taxonomic rank." The deeper the rank, the more taxon-specific expertise required to make the identification, so not all identifications are made all the way to species. In some cases, further progress is not possible because the species is new to science and has not yet been assigned a taxon past a certain rank. And yet, if the data indicates the presence of a specimen in a family, without also specifying genus or species within that family, we still know there is at least one genus and one species in that family present, even if we don't know what these more specific taxa are. Hence, the presence within a cave of a taxon at a particular taxonomic rank contributes one species to the count if it contains no lower-ranked taxa, and it contributes no species to the count otherwise. In the latter case, the lower-ranked taxa will instead contribute to the species count.

The available data is incomplete for several reasons. First, researchers continue to add backlogged data to the list. Second, sometimes specimens are deposited in other collections rather than in the UT Austin collection. Third, the data only reflects trips in which specimens were collected. In particular, James Reddell informs me that sometimes a collecting trip only collects apparently-new species and that sometimes multiple collecting trips are necessary before any specimens at all are found. This third point implies that estimates for effort produced from the data would be a lower bound for the actual effort required. This is fine for our purposes, because researchers are better off with a lower bound estimate than with no estimate at all.

Unfortunately, I have not completed the Texas cave research website enough to make it publicly available yet, so I cannot provide a reference to it at this time.

A3. REVIEW OF BACKGROUND WORKS

The above plot of cumulative species across person-visits makes it clear that species are found at different rates in different caves. It would therefore be inappropriate to establish a single model of effort for all caves. Instead, I'll need to group the caves by similarity, for some measure of similarity, and then model each group separately. One option is to cluster caves by their physical proximity, but this is problematic because cave-specific habitat determines the fauna more than proximity, and because UT Austin does not make the precise locations of caves available for public use. Another option is to cluster caves by the similarity of their fauna. Species are categorical and not numeric, so this requires a clustering algorithm that works on categorical data. Fortunately, the K-mode clustering algorithm does exactly this.

Nath (2021) describes the K-mode clustering algorithm well enough to implement it from scratch. The algorithm consists of defining a dissimilarity measure, selecting suitable initial cluster "modes", and clustering data around the selected modes using the dissimilarity measure. A "mode" is a characterization of a cluster relative to which objects can be compared for similarity. Nath (2021) assumes that the objects to be clustered have a fixed set of attributes each having a categorical value. The mode of a cluster consists of the most common values of these attributes, and the dissimilarity between and object and a mode is the number of attributes of the object that are different from the corresponding attributes of the mode. Nath (2021) also describes selecting the initial modes by dividing the most frequently occurring attribute values among the clusters, assigning to each cluster the object least dissimilar to its associated attribute values, and then designating the mode of each cluster by the attributes of its initial object.

As Nath (2021) further describes, given a dissimilarity measure and the initial modes, the K-mode clustering algorithm proceeds by looping over the following three steps:

1. Iterate over all the objects, assigning each to the cluster whose mode is least dissimilar to it, keeping track of whether any objects changed clusters.
2. After completing all necessary reassignments, re-evaluate each cluster's mode as a function of the attributes of the objects it contains.
3. If any objects changed clusters during this iteration, loop back to step (1). The loop apparently always eventually terminates, given an appropriate dissimilarity measure.

In my case, the objects are caves, and there are no fixed attributes at all, only the taxa known for each cave, which is hierarchical categorical data. Consequently, I need to establish my own dissimilarity measure based on a hierarchy of taxa. For the same reason, I am unable to use the Nath (2021) algorithm for determining the initial modes, but the author does provide a clue for how I might design my own initial modes algorithm: "make the initial modes diverse."

Once the caves have been partitioned into clusters by similarity, I plan to model the data in each cluster using linear regression. The data is clearly curved, but it should suffice to apply a transformation to the data and then model the transformed data. For reasons that will be clear later, I've chosen to model the data with the equations $y = Ax^P + B$ and $y = Ax^2 + Bx + C$, where the constants A, B, C, and P need to be determined. In the case of $y = Ax^P + B$, I'll be experimenting with a variety of values of P, and for each, using linear regression to get A and B. This leaves me with a different model for each possible value of P. For each cluster, I'll select the model for which the regression metrics indicate the best fit.

A variety of metrics are available for evaluating the goodness of fit of regression models. I draw from several sources in the following summaries of common metrics:

- **Pearson's Correlation Coefficient (statistic r)** – According to Frost (2019, pp. 7-10), Pearson's correlation coefficient provides a measure of the "strength and direction of the linear relationship between two continuous variables.” Values near zero indicate the absence of a linear relationship, while values near -1 and 1 indicate the presence of a linear relationship. The sign of the value is the sign of the slope of the relationship, though the value is not itself the slope. An absolute value near 0.8 indicates a “fairly strong” relationship, while an absolute value near 0.6 indicates a “moderate” relationship. Absolute values less than 0.6 are not suggestive of a linear relationship.

- **R^2 (Coefficient of Determination)** – Nami (2020) describes R^2 as representing the “strength of the fit” of the regression. Values range from 0 to 1, with values closer to 1 indicating a stronger fit. Frost (2019, p. 121) explains that R^2 “measures the scatter of the data points around the fitted line” and that it provides the “percentage of the dependent variable variation that a linear model explains.” Both emphasize that R^2 does not by itself indicate whether a model is appropriate, because the scatter might not be random. Both also explain that low values of R^2 need not rule out a model, because it’s still possible that the model is the best available for the data; better models can have lower R^2.
- **Adjusted R^2** – Frost (2019, pp. 128-192) emphasizes that this metric compares models of different numbers of independent variables to determine which is best. The present project uses only one independent variable, so the metric presumably does not apply.
- **Mean Absolute Error (MAE)** – MAE is the average of the absolute values of the errors. According to Nami (2020), one might choose to find the model that minimizes MAE in order to minimize the impact of outlier points on the model. In this case, prospective models would be compared based on their MAEs. I’ll need to see the data for the individual clusters before I can decide how I might deal with outlier points.

- **Root Mean Squared Error (RMSE)** – RMSE is the square root of the average squared error. Nami (2020) explains that RMSE better reflects the presence of large errors in the model, compared to MAE. Choosing a model based on minimizing RMSE therefore also minimizes the size of errors. Frost (2019, pp. 31-31) instead discusses the Sum of Squared Errors (SSE), which is RMSE squared times the number of points. It describes SSE as a "measure of variability", a characterization that necessarily also applies to RMSE, and it explains that using SSE for regression finds "the best possible line" (p. 31). Both emphasize the sensitivity of this metric to outliers.
- **Standard Error of Regression (SER)** – Zavarella (2017) describes the standard error of regression, which is also known as the "residual standard error" or just the "standard error." This metric is similar to the RMSE, except that it accounts for the number of degrees of freedom in the model. It can be treated as a standard deviation in that it tells us the percentage of errors that are within multiples of the SER. For example, we know that about 95% of the errors are within twice the SER from the fitted model. Frost (2019, pp. 133-137) emphasizes that the SER provides a better measure of goodness of fit than R^2.
- **p-value** – The p-value is used in tests of a model's statistical significance. Prior to testing a model, one chooses the significance level deemed sufficient to indicate statistical significance. Testing the model produces a p-value for the model as a whole, and if this p-value is less than the chosen significance level, the model is declared statistically significant. Authors describe this as "rejecting the null hypothesis" in favor of the proposed alternative hypothesis, as explained in Lee (2019) and Gupta (2020), for example. The typically chosen significance level is 0.05.

Frost (2019, pp. 190-195) further emphasizes examining plots of residual errors. The errors should appear to be randomly distributed, with no apparent patterns, as otherwise a better model is likely possible. Zavarella (2017) suggests examining a histogram of the absolute values of residuals to make sure they have a normal (bell-shaped) distribution, indicating that errors more commonly occur near the fitted model than far from it, with no unexplained bias. It seems that after checking plots and histograms of residuals, Pearson's correlation coefficient and the SER are the most important metrics for evaluating competing models. When multiple models are similar in these values, R^2 may be the next most important metric to examine. And finally, I'll use the p-value to assess whether a model is statistically significant.

The purpose of modeling the effort to find additional species is to estimate the effort necessary to find one more species in any given cave. If that additional point falls within the range of samples of other caves of the cluster, I may have to interpolate the value. If the cave has a large number of data points compared to other caves of the cluster, I may have to extrapolate the value beyond the sample range. Holmes (2017) explains how to deal with these situations. As Holmes (2017, section 8.1) explains, the model estimates the mean expected value, but the actual value will lie within some range around the predicted value. The goal is to determine the smallest range of values that can be expected to contain the actual value. One must decide the probability with which the value should fall within the range. "Levels less than 90% are considered of little value." The resulting range is called the "confidence interval."

Holmes (2017, section 8.1) describes how one might compute the confidence interval when neither interpolation nor extrapolation is required. The chosen probability is translated into the number of standard deviations Z_α that encompasses the probability. If we have at least 30 data points for the value of x and we know their standard errors, we can expect the actual data point to lie within a range $y = x \pm Z_{\alpha} (s / \sqrt{n})$, where n is the number of data points. If we have fewer than 30 data points available for x, other more complicated methods are required, suggesting that I might as well treat the value as interpolated.

Holmes (2017, section 13.6) distinguishes between the confidence interval and a "prediction interval." The confidence interval characterizes a mean value of many data points, while a prediction interval characterizes the expected value of a single additional data point. For a given chosen probability, the prediction interval has a greater spread than the confidence interval, making it less desirable to use. This suggests that large clusters of many data points and many caves would rely on the confidence interval, while small clusters of few data points or few caves would rely on the prediction interval. The computations for these two kinds of intervals are apparently complex, so I won't attempt to describe them here, but they can be used to establish intervals for any portion of the linear model, making them useful for providing the ranges of interpolated values. Illustrations of these intervals show them widening at the ends of the available data, indicating that they reduce in precision under extrapolation.

A4. SUMMARY OF THE MACHINE LEARNING SOLUTION

The solution is a website that enables researchers to explore estimates of the amount of effort needed to find additional species in Texas caves. The website will also predict the next species most likely to be found in any cave, enabling researchers to make decisions about which caves to further sample and how much sampling to do.

A researcher first specifies the number of clusters of caves to use. The website will use a K-mode clustering algorithm to partition the caves into this many clusters, clustering them by the similarity of their known fauna. The website will also find a best-fit linear regression model for the data in each cluster, using the effort (visits or person-visits) required to find that number species as the independent variable and the cumulative number of species found as the dependent variable. This choice of variables associates a confidence or prediction interval with the number of additional species expected to be found for a given amount of additional effort.

The researcher then selects a cave to examine. The website presents the data for that cave along with the regression model of its associated cluster. If there is enough data for the selected cave, the website also calculates and presents a regression model using only that cave's data. Both plots show the fitted mean values and the associated confidence or prediction intervals. This gives the researcher a choice of two models to consider for making predictions about the cave. For each model, the website also reports the predicted effort needed to find one more species in that cave, along with the associated confidence or prediction interval.

The website further reports the species that occur across all caves of the selected cave's cluster and the frequencies of these species within the cluster. It compares these species to the species known to the selected cave and reports the species of the cluster that have not yet been found in the cave, listing these missing species in their order of frequency within the cluster. This list suggests to the researcher which species are most likely to be found next in the cave.

A5. BENEFITS OF THE MACHINE LEARNING SOLUTION

The solution allows conservation researchers to gain an understanding of the expected return on effort for further sampling caves in Texas. Specifically, it helps them determine the minimum effort needed to extend the species checklist for any cave, and it helps them predict which species that effort would most likely add to the checklist. Together this information should help them prioritize sampling caves according to available resources.

B1. SCOPE OF PROJECT

The project includes the following deliverables and activities:

- A website hosted on UT Austin’s in-house TACC hosting infrastructure, provided by the university.
- Web pages on the website serving the solution described in this document.
- A daemon for automatically reading the university’s cave data and transforming it into the data needed by the machine learning components.
- Automated tests for the website and daemon.
- Source code for the above website, daemon, and tests, all in a GitHub repository.
- Several rounds of feedback with stakeholders for improving website usability, including subsequently providing those improvements.
- Instructions for using the website embedded within the website.

The following products and activities are **not** included in the project:

- A website hosting platform. The university’s TACC department takes care of this.
- A separate instruction manual.
- Correcting problems with the cave collection data. (This is done in advance.)
- Installing and managing a database for website data. The website will use an existing university database for this purpose.
- Populating the database with cave collection data. Fortunately, the raw data has already been imported into the required database.
- Database backup services. TACC takes care of this.
- Long-term maintenance of the source code.
- Long-term assessment of the success of the project.
- Data query and presentation features not already described within this solution.

The team for creating this project is as follows:

- Myself, the software developer.
- The university’s TACC department, which will host the website and the database.
- The cave collection curator, Alex Wild, who has responsibility for the long-term maintenance of the website.
- The cave collection manager, James Reddell, who is a prospective user of the website and an expert who can provide helpful feedback.
- The Texas Parks and Wildlife invertebrate conservation biologist, Ross Winton, who will be one of the primary users of the website. He can provide user feedback and arrange for feedback from other conservationists and researchers.

The project must be completed by August 15, 2022.

B2. GOALS, OBJECTIVES, AND DELIVERABLES

Goals

- To improve the ability of cave conservation researchers to monitor and conserve cave invertebrate species in Texas caves, so that researchers can better conduct their jobs of conserving invertebrate cave species in Texas.
- To help cave conservation researchers make more effective use of the limited resources they have for sampling caves, because researchers want to be as effective as possible despite the lack of money for conservation work.
- To improve the insight that cave conservation researchers have for the invertebrate fauna found in Texas caves, because insight improves the ability for researchers to fulfill their mission of cave species conservation.

Objectives

- To have at least 10 cave conservationists regularly using the website by the end of the first year of deployment, demonstrating the usefulness of the solution.
- To have increased the number of species known to occur in each of at least 20 caves by the end of the first year of deployment of the solution, demonstrating the effectiveness of the solution.
- To have at least 4 published papers citing the website and reporting insights learned about cave invertebrate fauna by the end of the first year of deployment, demonstrating that the solution improves researcher understanding of caves.

Deliverables

- A website hosted at the University of Texas at Austin.
- A daemon for automatically reading the university's raw cave data and for transforming it into the forms needed for clustering and regression.
- Automated tests for the website and daemon.
- Source code for the above website, daemon, and tests in the form of a GitHub repository.

B3. STANDARD METHODOLOGY

The project will employ the SEMMA methodology for data mining. The stages of this methodology and their relevance to the project are as follows:

- **Sample** – There are about 29,000 records of specimens from Texas caves, which is not too many to process in their entirety. However, the caves have different faunal compositions, making it inappropriate to regress all records at once. In this first step, the caves are partitioned into clusters of similar fauna, so that each cluster represents a sample of the available data that can be modeled as a unit.

- **Explore** – It is not yet clear which similarity measure to use for clustering, how many clusters are necessary to produce accurate models, or which transformation of the independent variable to use in the model. This step explores combinations of these factors to identify the most accurate approach.
- **Modify** – As another effort is underway to clean the collection data, this project assumes it already has clean data to work with. However, the data is raw, specifying only information about specimens collected on a per-specimen basis. This project analyzes aggregate data on a per-cave basis, so in this step, additional tables are created from the raw data to capture all the species collected on each visit to a cave and then to capture all the species collected across all visits to each cave. This step precedes the exploration step, because the exploration step relies on the data produced.
- **Model** – After processing the data for exploration, and after exploring clustering algorithms, transformations, and regressions, the best approach can be chosen and made available to researchers. This step implements that approach, enabling researchers to model cave sampling for making predictions about effort.
- **Assess** – The solution makes predictions about the effort required to find additional species in caves, so assessment of the solution will take at least a year to perform. Texas Parks and Wildlife and other research organizations will be able to collect data about the benefit of the website. It would be useful for UT Austin to institute a process for acquiring this data in order to make future improvements to the website.

B4. TIMELINE AND MILESTONES

This project started on May 21, 2022 and will end on August 15, 2022.

Start Date	End Date	Task
21-May-2022	21-May-2022	**Completed**. Design and populate a database table whose rows are aggregate data for each visit to each cave.
22-May-2022	22-May-2022	**Completed**. Design and populate a database table whose rows are the aggregate data of all visits to each cave. Write unit test cases.
22-May-2022	23-May-2022	**Completed**. Create a REST API for reading aggregate cave data, and create web pages that use the API to graph the data. Write unit test cases.
23-May-2022	24-May-2022	**Completed**. Prototype the K-mode clustering algorithm for taxonomic attributes, including an algorithm for selecting the caves that seed each initial cluster mode (server-side).
25-May-2022	25-May-2022	**Completed**. Implement a REST API for using the clustering algorithm and plot each cluster on a web page.
4-Jun-2022	5-Jun-2022	Implement power- and quadratic-transformed linear regressions on the cluster plots and present the resulting statistics on the website.
11-Jun-2022	12-Jun-2022	Implement a means for selecting a cave to plot and regress via both quadratic- and power-transformations, plotting the per-cave regression in addition to regression for its cluster.
13-Jun-2022	17-Jun-2022	Present the species most likely to be found next in any cave, given by what it lacks from its containing cluster, showing species in their order of frequency within the cluster.
18-Jun-2022	24-Jun-2022	Explore variations of the clustering algorithm and linear transformations for how well they fit clusters and individual caves. Present plots and UI screenshots to stakeholders and get feedback on their preferred approach.
25-Jun-2022	4-Jul-2022	Revise the solution according to stakeholder feedback. Write test cases for their selection, including the final REST API.
5-Jul-2022	15-Jul-2022	Deploy release #1 of the website for stakeholder use and feedback. This is in advance of having written integration tests, due the need to get feedback from them ASAP.
6-Jul-2022	9-Jul-2022	Write penetration tests for the REST API to make sure sensitive data within the underlying database is protected.
10-Jul-2022	15-Jul-2022	Write browser-based integration tests for the website.
16-Jul-2022	24-Jul-2022	Correct bugs found by stakeholders and implement their suggestions. Write tests for everything new.
25-Jul-2022	5-Aug-2022	Deploy release #2 of the website and get feedback.
25-Jul-2022	5-Aug-2022	Implement the daemon for automatically generating the data on which the algorithms depend and write tests for it.

6-Aug-2022	14-Aug-2022	Correct any remaining issues with the website, making any final change requests and writing tests for them.
15-Aug-2022	N/A	Deploy release #3 of the website (the final release). Also deploy the daemon and deliver the GitHub repo.

B5. RESOURCES AND COSTS

This is a real-world project that is already underway. I’ve listed the likely costs incurred by the university department that employs me and houses the cave collection. I’m doing the project for free because it is not within the scope of the grant that pays my salary, but for completeness, I’ve also listed the estimated value of the resources.

Resource	Likely Cost	Estimated Value
UT Austin in-house website hosting by TACC	$0	$120/year
UT Austin in-house Postgres database hosting	$0	$120/year
UT Austin in-house automatic database backup	$0	$60/year
UT Austin-owned GitHub repo (open source)	$0	$0
One software developer (150 hours at $60/hour)	$0	$9,000
Collection curator oversight (4 hours at $40/hour)	$160	$160
Totals	**$160**	**$9,160 + $300/year**

B6. CRITERIA FOR SUCCESSFUL EXECUTION OF PROJECT

I am employed at UT Austin under a grant that ends in October, so I personally will not be involved in a long-term evaluation of the solution. However, I can establish criteria for evaluating the short-term success of the project, along with criteria that the university might use to evaluate the long-term success of the project.

The short-term success criteria follow:

Actual Short-Term Objective	Success Criteria
Complete	No remaining to-do items for things to add to the code or double-check in the code or in execution
Reliable	All tests passing and no known bugs

Stakeholders satisfied	No change requests pending for changes stakeholders deem urgent or necessary

Long-term project success criteria that the university might employ:

Proposed Long-Term Objective	**Success Criteria**
Useful	At least 10 cave conservationists regularly using the website by the end of the first year of deployment
Effective	Having increased the number of species known to occur in each of at least 20 caves by the end of the first year of deployment
Provides insight	At least 4 published papers citing the website and reporting insights learned about cave invertebrate fauna by the end of the first year of deployment

C1. HYPOTHESIS

This project is founded on the hypothesis that the effort required to find new species in caves can be modelled via linear regression with appropriate transformations. Specifically, after clustering the caves of Texas according to the similarity of their fauna, the number of visits or person-visits required to discover additional species within a cluster should conform to either the power expression $y = Ax^P + B$ or the quadratic expression $y = Ax^2 + Bx + C$, where x is the cumulative number of species discovered and y is the number of visits or person-visits. If I can find a model whose p-value is less than a significance level of 0.05, the model will have been shown statistically significant, and the hypothesis will have been confirmed.

C2. ANALYTICAL METHOD

Section A3, Review of Background Works, describes the algorithms used in this solution, cites sources, and explains the utility of each aspect of each algorithm to the present project. However, that section left out a few significant details, which I’ll cover here. These are the details of the solutions that the sources left unspecified and that I had to design myself.

The first step of the solution is to cluster caves according to the similarity of their fauna using the unsupervised K-mode clustering algorithm. This clustering algorithm requires establishing a dissimilarity measure, selecting an algorithm for choosing the initial mode of each cluster, and iteratively reassigning caves to clusters according to the dissimilarity measure until an iteration produces no change in assignments. I have already described this algorithm in detail, but the dissimilarity measure and the initial modes selection require further definition.

Unfortunately, I was unable to find a dissimilarity measure in the literature that seemed applicable to the present scenario. Most of the literature assumes that the objects to be clustered each have the same set of attributes but different values for those attributes, such as in Nath (2021). In the present scenario, each object – each cave – is characterized by a tree of values. The values in these trees are the taxa of the organisms found in the caves. As previously explained, these taxa form a hierarchy of ranks. Different caves may or may not have taxa common to their respective trees. When a low-ranked taxon occurs in a tree, so do all the higher ranks of that taxon. It's possible for two trees to contain a common high-ranked taxon but only one tree to contain taxa below it. These trees are similar to that common rank and dissimilar below that rank, except that it's possible that the cave indicating only the high-ranking taxon included the taxon because it contains organisms of the lower-ranked taxon given in the other tree. This happens often when the work of identifying a specimen is incomplete.

To proceed under these circumstances, I had to implement the K-mode clustering algorithm and simultaneously experiment with dissimilarity measures and algorithms for selecting the initial modes. Recall that a "mode" is a characterization of a cluster that can be used for comparing individual objects (caves) with the cluster to decide if the object (cave) belongs in the cluster. The dissimilarity measure therefore compares a cave to a mode. I defined the mode of a cluster as the set of all taxa found in any cave of the cluster. After maybe half a dozen experiments, I found the following dissimilarity measure effective:

1. Count the number of the taxa common to both the cave and the mode. This takes advantage of the fact that when a low-ranking taxon occurs in a tree, so do all its higher ranks. The higher ranks get counted only once as multiple lower ranks under each high rank continue to increase the count.
2. Subtract from this sum the number of taxa found in the cave but not in the mode.

The lower this number, the more dissimilar the cave is from the mode. This measure worked well with the following algorithm for selecting the initial modes:

1. Sort the caves by the number of taxa found in the cave, highest number first.
2. Designate the first cave of this sort as representative of the first cluster.
3. Search all caves not yet representing a cluster for the cave with the most taxa not in the cumulative taxa of all caves already representing a cluster.
4. Return to step (3) until the desired number of clusters have representative caves.
5. Designate the initial mode of each cluster as the set of taxa in the cave that represents the cluster according to this algorithm.

This solution was inspired by the Nath (2021) suggestion to “make the initial modes diverse.” Sorting the caves by their number of taxa and searching caves in this order for each next representative cave also allowed for short-circuiting the search when it was no longer mathematically possible to find a subsequent cave with more differences.

Amazingly, as Nath (2021) predicted, no matter how many clusters I chose to generate, the K-mode algorithm always found a partitioning of the 1,646 caves and came to a halt. Even so, despite having found a dissimilarity measure and an algorithm for the initial modes, I do not yet know whether the resulting clusters are ideal for regression modelling. For this reason, I plan to experiment more with dissimilarity measures and initial mode algorithms.

Although this custom K-mode algorithm successfully partitions caves by hierarchical characteristics, as required, it has a few downsides. First, because it is a variation of the K-means clustering algorithm that replaces the dissimilarity measure (Nath, 2021), and because K-means clustering is NP-hard (Meena et al., 2012), K-mode clustering is also NP-hard, so we do not know that it always completes (Nath, 2021). Second, because I established my own dissimilarity measure, I do not have a sense of whether it is the optimal dissimilarity measure. Third, the clusters that the algorithm yields are dependent on the choice of the initial modes (Nath 2021), and I don't know if my algorithm for choosing the initial modes is optimal.

After partitioning the caves into clusters, the solution performs a linear regression on each cluster to model effort required to find more species in caves. The plot presented earlier, "Cumulative species across person-visits," shows the number of person-visits on the x-axis and the cumulative number of species on the y-axis. The plot is clearly curved, and it's reasonable to expect the plots of clusters to also be curved. To perform a linear regression on this data, one or both of the variables must be transformed. I experimented with various logarithmic transformations and found none of them helpful in linearizing the data. A power transformation of the form $y = Ax^P + B$ did appear to linearize the data, where x is the effort in person-visits, y is the number of species, and P is somewhere near 0.5. The data has a parabolic appearance, so it is also worth exploring transformations of the form $y = Ax^2 + Bx + C$. The solution will therefore perform regressions on both equations, looking for a model with the best residual plots, histogram of residuals, correlation coefficient, SER, R^2, and p-value.

Given that the model is a linear fit of transformed data, it may not be meaningful to plot the dependent variable against the transformed independent variable. Researchers would rather see a plot of total effort required to total species expected. I will therefore need to transform the model back into these meaningful units. If I have a model that expresses y as f(x), then I have a y for each x. Assuming f(x) has an inverse on the relevant range of y, I can take the known model point (f(x), y) and transform it into (x, f^{-1}(y)) for purposes of plotting. The confidence/prediction interval would also need to be transformed. As Hanlon and Larget (2011, slide 12) indicates, for any given interval (a, b), with a < f(x) < b, it is reasonable to convert the endpoints to meaningful units by expressing it as the interval (f^{-1}(a), f^{-1}(b)), allowing me to plot the interval as well.

There are some drawbacks to modelling the data with linear regression. First, because researchers may not have reported visits to caves for which they collected no species or found no new species, the data itself has a bias for increasing species counts at less than actual effort. This suggests that the model may not uniformly apply to all caves or that it may not uniformly apply to all visits in any given cave. Second, researchers are interested in using the model for prediction, which necessarily entails extrapolating to the next visit to a cave. As Holmes (2017, section 13.6) illustrates, prediction intervals widen significantly beyond the sampled range, decreasing the confidence researchers can have in estimates of effort. Third, because the regression necessarily operates on transformed variables, the model itself will be hard to interpret and may not provide insight into the relationship between effort and species found.

C3. TOOLS AND ENVIRONMENTS OF SOLUTION

The solution will be written in TypeScript, HTML, CSS, Svelte, and PostgresQL. Svelte is a modern reactive web user interface framework analogous to REACT. The website will be hosted by UT Austin’s TACC department, which has provided an Ubuntu image and an installation of the PostgresQL database. The backend will run on Node.js, which is a platform for running server-side JavaScript, into which TypeScript transpiles.

The solution will employ the following open-source JavaScript libraries:

- Express.js for hosting REST APIs and serving the JavaScript client
- Jstat for performing linear regression and producing statistics
- Chart.js for plotting graphs
- Bootstrap for standardizing the web user interface
- Jest for running unit tests
- Playwright for running browser-based integration tests
- Benchmark.js for measuring algorithm performance

In addition, the solution will piggyback on the university's existing initiative to clean the cave data and make it available via a database, reducing the total effort involved.

C4. MEASURING PERFORMANCE

The resulting website is expected to have at most a few dozen users, as there are not many cave conservation researchers. Therefore, there is no expectation to have to meet high bandwidth or throughput requirements. On the other hand, the clustering algorithm necessarily runs server-side, and its resource usage could potentially cause problems. Client HTTP requests should not time out, and executing the clustering algorithm on the server should not bog the server down so much that it can't handle additional HTTP requests. The linear regression itself will be performed client-side in the browser, using standard software, so it will be the user's responsibility to use a client that can process the data in a timely manner.

Consequently, it is only necessary to measure the performance of the clustering algorithm. The benchmark.js library provides a reliable way to measure algorithm duration. Browsers will typically timeout after waiting a few minutes for a response, but users will likely assume something is wrong if the response takes more than a few seconds. I will run the benchmark on the production server to see how long the clustering algorithm takes, and if necessary, look for ways to parallelize it and reduce the response time.

The algorithm will potentially consume a lot of memory, so it’s also important to ensure that it stays within the heap limit. Node.js provides command line switches for outputting memory usage, as well as switches for establishing the heap size. I’ll run the algorithm with these switches to see whether usage is within a reasonable tolerance. I will need to assume that several people might run the algorithm simultaneously, possibly with different numbers of clusters, so the heap size should be a multiple of the amount the algorithm requires. It will also be necessary to establish and include the baseline heap usage when no algorithm is running. Given that only a few people will ever be running the algorithm at once, and given that all runs of the algorithm base themselves on the same underlying data, the database is not expected to be a performance bottleneck, particularly due to database-side caching.

D1. SOURCE OF DATA

This project uses the data that invertebrate cave researcher James Reddell has been collecting and maintaining for the past several decades at the University of Texas at Austin. He has kept the data in a spreadsheet, but I have been working with him over the past year to clean up the data and prepare it for uploading to a database and a website. I have access to various forms of the data and permission to use it in this project. The data provides raw information about every specimen preserved in the collection, including data collected, collector names, location information, and taxonomic determinations.

D2. DATA COLLECTION METHOD

James Reddell sends me an updated spreadsheet every few weeks. I run my software on this data to generate problem reports. The software partly autocorrects the data for these problems, and James corrects what the software can't autocorrect. The data will likely take the rest of the year to completely clean up, so it will be necessary to discard the problematic portions of the data until then. Due to the hand-recorded nature of the data, some problems can never be corrected sufficiently for use with this project, as explained in the next section. However, the data comes directly from the person who has spent decades tracking most of the invertebrate specimens collected in Texas caves, so the information is expansive and definitive.

D3. QUALITY AND COMPLETENESS OF DATA

Most of the available data is complete and usable for this project. However, information for some caves and some specimens is incomplete. For example, some specimens are only marked for the year collected and not the day of the year. Some specimens have no collector names listed. And some specimens have not yet been taxonomically identified. This project depends on ordering collecting trips by date, knowing the number of people who collected (for determining person-visits), and comparing the species found across visits. About a thousand records are missing this crucial information, and I won't be able to use them for the project.

A few hundred specimens were collected via pitfall traps left in caves over several days or weeks. This project has not identified a way to plot or estimate effort for these collection activities, so this data will also be excluded from the project.

After filtering for the usable data, the data must be aggregated into the information that the algorithms require. The raw data merely lists information about each collected specimen, with the specimens collected on any one visit distributed across many rows. The first step is therefore to accumulate the data associated with each trip to each cave, storing this data in a "visits" table of the database. This table includes columns such as location, date visited, last names of the collectors, and a list of all taxa found during the visit.

The data in the "visits" table must then be aggregated into information about each cave. The second step is therefore to examine all visits for each cave, as stored in the "visits" table, and store the accumulated information in an "effort" table containing one row for each cave. The "effort" table includes columns such as location, first and last dates collected, total number of visits, total number of person-visits, and a list of all taxa ever found in the cave. The clustering algorithm can work directly from the data in this table.

It is possible that some caves within a cluster will have very different plots from other caves of the cluster. It may be useful to identify these caves, remove them from the regression, re-generate the model for the cluster, and report the removed caves to the user. However, given that there are at least 1,646 caves represented in the data, this process must be automated. Perhaps the simplest way to identify these outliers is to examine the model's error for the last visit of each cave and remove caves for which this error exceeds three standard deviations. The last visit captures the cumulative species found and the total effort expended, making it representative of the cave as a whole and therefore useful for indicating outliers.

D4. PRECAUTIONS FOR SENSITIVE DATA

The data contains the exact locations of caves, but the University of Texas does not publish these exact locations. This is to protect the caves from degradation due to visitors, as well as to protect the species they contain. No other cave data is sensitive, not even the presence of rare species. One of the transformations I have already written for the data prepares it for upload to the GBIF biological database, and the data in this database is entirely public, so this transformation sanitizes cave locations. More specifically, it only provides the coordinates to two digits of precision, and it removes location descriptions. To prevent leaking cave locations, I will use the file output by this transformation as input for the present project.

E. SOURCES

Frost, J. (2019). *Regression Analysis: An Intuitive Guide for Using and Interpreting Linear Models*. Statistics by Jim Publishing.

Gupta, A. (2020, November 2). Key Statistical Concepts in Data Science: A compilation of some must-know statistical concepts. *Towards Data Science*. Retrieved May 29, 2022 from https://towardsdatascience.com/key-statistical-concepts-in-data-science-265cf5126fba

Hanlon, B., & Larget, B. (2011, November 10). *Assumptions and Transformations* [PDF of slides]. University of Wisconsin—Madison. https://pages.stat.wisc.edu/~st571-1/11-assumptions-2.pdf

Holmes, A., Illowsky, B., Dean, S., & Hadley, K. (2017, November 29). *Introductory Business Statistics*. OpenStax. https://openstax.org/details/books/introductory-business-statistics

Lee, A. (2019, July 13). P-values Explained By Data Scientist. *Towards Data science*. Retrieved May 29, 2022 https://towardsdatascience.com/p-values-explained-by-data-scientist-f40a746cfc8

Meena, M., Nimbhorkar, P., & Varadarajan, K. (2012). The Planar k-means Problem is NP-hard. *Theoretical Computer Science*, 442, 13-21. https://www.sciencedirect.com/science/article/pii/S0304397510003269

Nami, Y. (2020, October 2). How to choose the best Linear Regression model — A comprehensive guide for beginners. *Towards Data Science*. Retrieved May 28, 2022 from https://towardsdatascience.com/how-to-choose-the-best-linear-regression-model-a-comprehensive-guide-for-beginners-754480768467

Nath, J. (2021, April 4). Clustering Technique for Categorical Data in Ptyhon. *Medium.com*. Retrieved May 28, 2022 from https://joydipnath.medium.com/clustering-technique-for-categorical-data-in-python-8eb0f581b6f9

Zavarella, L. (2017, September 13). How to Better Evaluate the Goodness-of-Fit of Regressions. *Microsoft Azure*. https://medium.com/microsoftazure/how-to-better-evaluate-the-goodness-of-fit-of-regressions-990dbf1c0091