DELWP Final Report

13/10/2019

VICTORIA
State
Government

Environment,
Land, Water
and Planning

Gia Hy Truong (25954202), Jack Ho (27794040), Priyum Joshi (28776895)
Monash University: FIT3164 - Data Science Project 2

Word Count: 9800

Table of Contents

Introduction

The development of Species Distribution Models (SDMs) has become an important factor in the management of landscapes and preservation of habitats for endangered species. One of the most important factors in the creation of these models are the observations of the species being sought, but for the species to be observed professionally, will be quite expensive, thus making the sample size a bit smaller than sought out.

Therefore, volunteering data is now considered towards the addition for datasets of species observations and as expected, the number of observations are now rapidly increasing. However, a problem occurs where volunteers may not have appropriate training or equipment to create reliable observations and experts are burdened with the task of verifying these observations before the data can be stored into a repository.

To ease the burdens for experts, predictive models that can quickly interpret and differentiate these volunteering data into categories of reliability and unreliability are being developed. The construction of this predictive model allows two things: an easy access to a list of highly reliable data to add to the repository and an understanding of variables that are beneficial to deciding the reliability of a data point.

The construction of the predictive model requires a few elements to it. One, a sample data set to create training and test data to predict with. Secondly, geolocational data for comparison of the observations probable environment. Thirdly, a learning method in which an algorithm is able to predict and see which elements are reliable or unreliable and the variables that make it reliable. And finally, for the whole construction of this predictive model is able to be built inside a container so that any user is able to run this on any machine and with any operating system.

There is also the requirement of the predictive model being usable by the user, so a user interface (UI) is also needed where it can interact with the results to show more relevant information.

Therefore, the purpose of this report is to show this attempt at accomplishing this predictive model so it can further assist the experts who need to sort out the volunteering data. Although the product did end up with a few limitations, the overall product is able to answer and accomplish the tasks being sought here.

Literature Review

(Updated version of our previous literature review from our project proposal. Reference at the bottom)

Species distribution modelling (SDMs) have become an interest to many environment departments in government. As these models assist the government entities, who use this information, by providing a forecast of how the impact of landscaping can affect a species of plant or animal through extrapolating limited information in order to estimate the distribution of the species over large areas (Newbold, 2010). These models are formed using mathematical relationships of the confirmed species locational predictors (Guisan & Zimmerman, 2000) and these locational environmental predictors include information on climate, topography, soil, and vegetation structure (Zellweger et al., 2016). These two points form the foundation of any SDM.

As the major application of SDM is through conservation planning, it is a highly sought out method for government entities to use. However, despite the growing use of SDMs, there are many limitations associated with creating a model that can predict species distributions accurately. As Guisan and Zimmerman (2000) have stated nearly twenty years ago, “Nature is too complex and heterogeneous to be predicted accurately in every aspect of time and space from a single, although complex model” (Guisan & Zimmerman, 2000, p 150). Therefore, models should aim to optimize accuracy or generality when curating models to fit ecological datasets (Guisan & Zimmerman, 2000). Since species distributions data have changed in the past two decades, the latter is preferred when creating SDMs (Howard, Stephens, Pearce-Higgins, Gregory & Willis, 2014). Optimizing generality for SDMs is the preferred method because of how varied the environmental predictors can affect different plant and animal species. Additionally, optimizing SDMs that suit a variety of different species addresses many of the key limitations of SDMs (Guillera-Arroita et al., 2015).

There is also another limitation towards these SDM, as data gathering today has also included in opportunistic data (Lin, Lin, Lien, Anthony & Petway, 2017). Data quality has become an issue to the results of these SDM’s and their results and these issues stem from either illegitimate data or positional errors from the observations gathered. “These outliers need to be identified, checked and removed to improve data quality and minimize the impact on subsequent analysis” (Liu, White and Newell, 2017). So new algorithms and modelling techniques are being developed to assist experts in their attempt to address data quality issues.

Pseudo-Absence Data

SDMs widely used can be categorised in two groups: methods that only require presence data vs those that require both presence and absence data. (Lobo, Jiménez-Valverde & Hortal, 2010). The best and most accurate species distribution models however require not only presence data, which are readily and abundantly available, but also presence-absence data (Barbet-Massin, Jiguet, Albert & Thuiller, 2012). Presence only data consists of a sample of locations with known presences and a separate group of locations sampled from the population with unknown presences (Ward, Hastie, Barry, Elith, & Leathwick, 2008). True absence data is obviously the opposite of what presence data is, where it is repeatedly observed that a species is not present in the particular location (Huijbers, 2016), but this true absence data is important to have, since we are unable to infer if the lack of observation outside of the presence data areas have no observations possible or because no one has simply observed there. Not only that, true absences are very difficult to obtain especially for mobile species and require higher levels of sampling effort to ensure their reliability (Barbet-Massin, Jiguet, Albert & Thuiller, 2012). So, the use of pseudo-absence data is more generally used in today's SDMs.

However, in the creation of pseudo absence data, there are two things that need consideration before creating pseudo absence data so that it can provide the greatest impact on the model's predictive accuracy (Barbet-Massin, Jiguet, Albert & Thuiller, 2012). In areas where higher specificity is valued over high sensitivity, randomly selected pseudo-absences (so not just areas outside of observed areas) have proven to have done better overall (Barbet-Massin, Jiguet, Albert & Thuiller, 2012) and the amount of pseudo-absences with the same amount of available presence-only data or even more. Now the creation of pseudo-absence data is now able to assist when presence-absence data is unavailable.

Limitations of Species Distribution Modelling

In general, larger ecological datasets – including more environmental predictors - has provided for a more accurate SDM (Howard et al., 2014; Liu, White & Newell, 2018). However, this does not apply to all SDMs (Guo et al., 2015). For example, Guo's (2015) findings for modelling ninety-two fish species have shown that datasets with small range sizes and four environmental predictors were only needed to produce the most accurate model that predicted fish distribution in an aquatic ecosystem. This makes logical sense since species that live in niche environments should be detected easily because of its smaller geographic range and limited environmental tolerance (Hernandez, Graham, Master & Albert, 2006). This factor should therefore be noted when deciding on modelling certain types of species.

Hernandez (2006) also noted that having the wrong type of environmental predictor variables when modelling species can be attributed to a poor predictive model despite having various performance metrics saying otherwise. For instance, if a certain type of highly mobile species could live in multiple environments, having a model predict on differing environments would not be the best choice for the model. Another case could be having habitats for animal species change during summer and winter, which would call for a different model for each season (Jaberg & Guisan, 2001). This is where domain knowledge on the species being analyzed becomes advantageous when developing an accurate SDM. Additionally, improved designs for sampling data for building models could potentially alleviate some of the limitations of SDMs (Araújo & Guisan, 2006).

Another limitation of SDMs is having to optimize the parameter settings of the model. Due to the nature of SDMs, nearly all modelling techniques will eventually have to rely on optimizing parameters of the model (Guisan & Thuiller, 2005). Having the same model predict on the same environmental predictors with different parameterization can yield vastly different results (Wilson, Westphal, Possingham & Elith, 2005). Wilson (2005) has shown how adjusting the threshold of a model can ensure a highly suitable species distribution, whereas not adjusting it can lead to a less reliable model. Furthermore, by re-calibrating our models from time to time, it can adjust to new cases when new input data is given (Wilson et al., 2005). It is therefore vital to consider several implementations of the same model to allow for a more robust model to be achieved.

Finally, with the advent of technology becoming more widespread, the use of smartphone application software has allowed for the public to volunteer to contribute to adding species distribution data (Lin, Lin, Lien, Anthony & Petway, 2017). As a result, the difficulties in implementing a robust SDM may not necessarily come from the model itself but from the data received. Since most volunteers lack formal survey training (Bonney et al., 2014), misidentification and biases are, unfortunately widespread (Lin et al., 2015a; Lin et al., 2015b). SDMs that build upon volunteering data must therefore consider the biases found in this type of data. Models have existed to detect these outliers such that we can remove and improve data quality (Liu et al., 2018; Liu, White & Newell 2010; Rousseeuw & Hubert, 2011; Araújo & Guisan, 2006). Proceeding from this point on, we will coin the term "pseudo-SDM" as SDMs used to specifically detect outliers. Note that SDMs by nature are able to detect outliers themselves; however, we will specifically use pseudo-SDMs to differentiate between the purposes of SDMs and pseudo-SDMs. SDMs are used to identify species distribution, whereas pseudo-SDMs are used to detect species that are outliers.

To conclude on the limitations of SDMs, we must remind ourselves that even if we address the limitations of our models completely, it is still an estimation of the species' potential distribution. SDMs will never give a complete picture of the distribution of species (Hernandez et al., 2006). The potential of pseudo-SDMs – SDMs used to detect outliers – is however, a new and promising concept used to address some of the limitations of SDMs (Liu et al., 2018). By allowing pseudo-SDMs to specifically detect outliers in the distribution of species, we can utilize volunteer data without having to worry about the biases or misclassification of species.

Current Outlier Detection Methods for Species Distribution Modelling

Many outlier detection methods have been long established over the past two decades (Liu et al., 2018). Over the past few years it has been noted that ensemble-based methods (random forests, Biomod, Maxent and GBM) have consistently outperformed regression-based methods (Howard et al., 2014; Liu et al., 2013; Thibaud, Petitpierre, Broennimann, Davison & Guisan, 2014; Bucklin et al., 2014; Miyamoto, Tamura, Sugimura, & Yamada, 2004). Therefore, we will review the recent advances in ensemble learning as well as some other notable techniques for SDMs.

Random forest is an ensemble method for classification, and it is constructed by using multiple decision trees that are combined to create a refined tree. It has been a popular ensemble learner used for the creation of SDMs because of its' ability to identify non-intuitive relationships (Jeffrey, Melanie, Zachary & Samuel, 2011). Since nature is comprised of complex ecological systems, it often behaves in non-intuitive ways. As such, the flexibility of algorithms such as random forests allows them to discover hidden relationships in a traditional setting.

We alluded earlier that having to change parameter settings for the model is crucial for a good performing SDM (Guisan & Thuiller, 2005). However, this can be time-consuming to do separately, especially when we need to adjust different parameters for each species. Maxent is another popular modelling technique that introduces a high performing modelling method that is almost as good as models that have had its parameter settings tuned to the specific species (Phillips & Dudík, 2008). Briefly, Maxent takes a list of present species, often called presence-only data (Wang & Stone, 2018), as well as a set of environmental predictors. It then extracts a sample of background location that is compared against the presence locations (Merow, Smith & Silander, 2013). Between 2006 to 2013, there has been over one-thousand published applications of Maxent (Merow et al., 2013). The ease at which Maxent's default parameter settings can be changed can produce a very well performing SDM. Unfortunately, most ecological datasets contain sampling bias, that is, bias in the identification of species, especially one that is produced by volunteers (Merow et al., 2013). Because of this, the function of Maxent is limited and is said to be best used to help "ask better questions instead of answering them" (Merow et al., 2013, p.1068). For this reason, if there is sampling bias, Maxent could still be used in the initial exploratory analysis of the species and the dataset.

When using the above methods for creating SDMs, Guisan (2000) has noted that environmental predictor variables can be grouped into two groups: direct and indirect predictors. (Liu, White, Newell & Griffioen, 2013). Direct predictors include temperature, light intensity, and availability of food – factors that directly influence the species living in a specific environment. Indirect predictors include elevation, slope and slope position – factors that do not affect the living conditions of the species directly. Using direct predictor variables for our modelling methods enables us to create a more accurate model overall (Jeffrey et al., 2011). This has been proven by multiple ecologists and researchers. For instance, Zellweger (2016) found that climate and vegetation structure were the best predictors in improving the prediction of butterfly and bird species distribution. Another example is Liu (2013), who initially utilized 26 environmental variables but found that only three direct predictor variables were the most important in obtaining a highly accurate model.

In addition to direct and indirect predictors, the interaction between predictor variables themselves has often been omitted from SDMs (Guisan & Thuiller, 2015). The interaction between different predictor variables can potentially identify significant interactions between environmental predictors. Again, this can increase the performance of any modelling techniques used at the time. Perhaps the reason why researchers have rarely used this method is that it greatly increases the number of parameters in the model. It is possible that it can create a less accurate model if not done correctly.

Conclusion

There are many practical applications of SDMs; ranging from understanding the distribution and evolution of species, to government entities using it for land and conservation planning. The vast amount of different modelling techniques for SDMs is reflective of a highly adaptive ecosystem of researchers aimed at attempting to produce the highest performing SDM. Because of the variety of modelling methods offered in the current literature, we have the ability to cater our datasets to any of the aforementioned modelling techniques. There are many modelling methods not stated in this review, but because of the vast differences in how they operate, the most popular methods were only mentioned. In order to overcome the many limitations of predictive modelling, we must take the time to consider how we model our species distribution data carefully. As a result, careful model selection, parameter optimization, and crucial environmental predictor variables should be carefully chosen when crafting an SDM.

Methodology

The necessary elements to create a predictive model and make it usable so that it could differentiate between reliable and unreliable data points and show relevant information is:

- A data set in which to generate training and test data.
- Geolocational data for generating a data points probable environment
- A learning algorithm in which to show a data element is reliable or unreliable
- The construction to be available in a docker container.
- UI also becoming available.

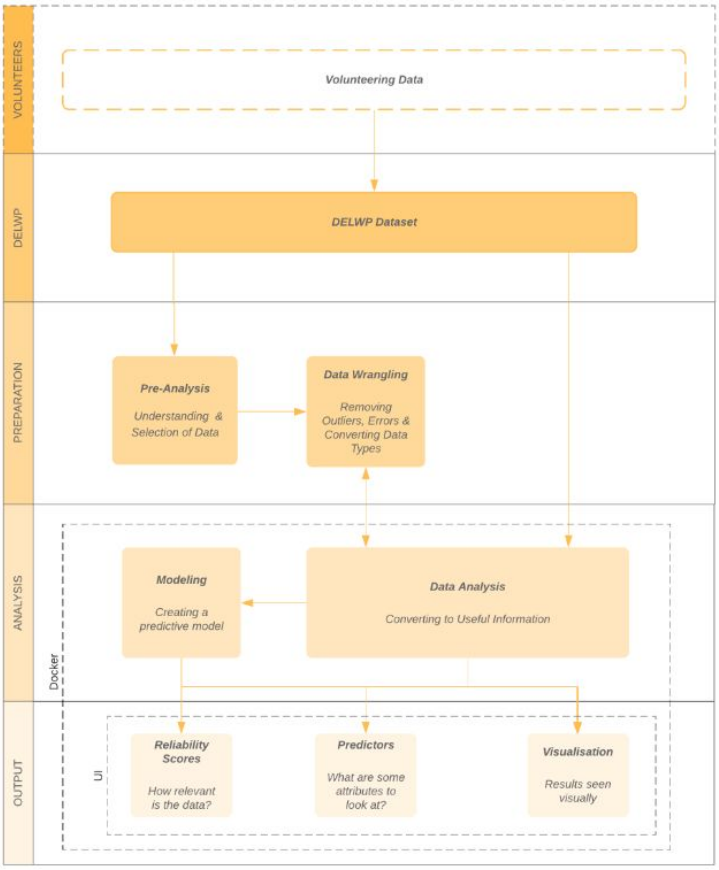

VOLUNTEERS
Volunteering Data
DELWP
DELWP Dataset
PREPARATION
Pre-Analysis
Understanding & Selection of Data
Data Wrangling
Removing Outliers, Errors & Converting Data Types
ANALYSIS
Docker
Modeling
Creating a predictive model
Data Analysis
Converting to Useful Information
OUTPUT
UI
Reliability Scores
How relevant is the data?
Predictors
What are some attributes to look at?
Visualisation
Results seen visually

Image 1. Project Architecture

The following sections will discuss the variety of tools and approaches used to implement the point stated above.

What Has Been Implemented and How

Choice of tools:

The programming language chosen to develop this project was all done through R, whether it is the data cleaning, data analysis, pseudo-absence data creation, visualisations or the UI. With the sole use of R however, it would defer from the project proposal where there was the consideration of using Python as an assistance to develop or clean the project or using Tableau and QGIS to help visualise results. The reason for this decision though, is because R is such a versatile language because of its access to many libraries and packages for the construction of our project. For example, Random Forest package, which is an ensembling learning method for classification, maptool package, manipulates geographic data or RShiny packages,which is an easy way of building interactive web applications from R.

Of course, to fulfill the need of our project being contained in a docker container, the use of Docker will also be needed.

Datasets

To construct the project, datasets have been provided by the client. Datasets provided were twofold; an excel sheet which contains the species observation dataset that were observed and a raster dataset, both of which are maintained and stored in a Google Drive. There are some limitations to the way these datasets are stored and used. One, datasets needed to be downloaded from the drive and this has caused problems through the fact that sometimes, datasets are just far too big to be downloaded and either takes too long or crashes. Good internet or a lot of time is the best solution for this, but this limitation is unavoidable as this dataset has to be readily available for use by others.

Pre - Analysis and Data Wrangling

However, after the construction of our project has gone through its pre - analysis. The datasets received had several limitations. To begin with, the sample data set of observed species for example, had five kinds of species in it. And while some species may have a lot of data points, there are a few that have less than a hundred, making it a little bit difficult to predict anything with so little data. So to supplement the example data sets given by our client, the use of other public data sets were looked at. And in this case, the use of the GBIF repository where it contains the same species to those of the ones given by our client.

The raster dataset given also had limits to usability in terms of geolocational data to assist points in predicting reliability or not. The reason being is for example, some of the raster data range is too small or does not fit with some of the species very well. So while the team was trying to get it to work, the team realised that using another set of raster data was far more useful. In this case, the bioclimatic raster data from worldclim.com works better and all variables cover the species received by the client. The raster dataset being used has been commonly used for other SDMs so it is quite reputable.

Data Analysis

Pseudo Absence Data

The first phase of our data analysis is to generate pseudo absence data, since Random Forest need both presence and absence data for being able to classify correctly. But because absence data is not available, the generation of pseudo absence data is needed.

So to generate pseudo absence data, three methods of generation were looked at, two of which were provided by R libraries; Spatial Eco, SSDM and manual creation. The SpatialEco library generates pseudo-absence samples based on density estimates of known locations. (Evans, n.d). The SSDM library uses a “recommended PA selection strategy that is used depending on the algorithm” (which is based on (Barbet-Massin et al., 2012)) and the third method generates pseudo absence points in areas where the species isn't found.

However, there is a small issue due to time constraints that only one species could be tested. But with the results of this one species, the same methods will be used for the other five species. So out of the three methods, the chosen generation is SSDM. The reason being is because there are locations which are not generalised compared to the other methods and will produce areas correctly. More explanation down in the later sections.

Modelling Techniques

The second phase of data analysis consists of finding the best performing modelling technique that can predict whether a datapoint is reliable or not. For starters, the dataset in which we have observed species are split into two; one training set and one test set. And the training data set is used in each of the models, where it will test with the test data set.

Each individual model is evaluated through by looking at; area under the curve (AUC), sensitivity and specificity. And similar to the above section, there is not enough time to go through all species and test which

The models and combination of models being looked at in this project are:
Artificial Neural Network (ANN), Random Forest (RF), Generalized Additive Mode (GAM), Maximum Entropy (MAXENT), Generalized Boosted Regression Model (GBM), Generalized Linear Model (GLM), RF + ANN, RF + GAM + ANN, RF + MAXENT + GAM + ANN, RF + MAXENT + GBM + GAM + GAM, ALL.

To simplify, the chosen model to use for this project is Random Forest. More discussion on this down below.

Deployment

RShiny is a useful library that integrates into R that allows for user interactivity. The RShiny App shows the reliability scores, in a tab where you can input in a sample file and it will return results based on the dataset given. It also shows visualisations of observed species in an overview and a visualisation of inputted observations. And finally there is a statistic table which shows most relevant predictors in a tab as well.

All of this is available and deployable in a docker container. Our project for docker is split into two. The first, contains all installations and packages while the second contains the app and r files. The reason for this is so that if you ever need to change anything inside the app or r files, you would not need to rerun installations. As installations takes close to an hour or more for package installation.

Project Management

How was the Project Conceived?

The development of the project commenced during semester one where we were able to lay the groundwork for semester two. During this time, we familiarized ourselves with various skills required to succeed in meeting the requirements of DELWP. This included running through different styles of project management as well as completing various case studies, researching primary articles of the project at hand, and learning how we would manage risk in the group. Additionally, during this period we started to become acquainted with our colleagues; learning about our strengths and weaknesses. This had enabled us to start allocating tasks that we would deem reasonable to one another after taking our strengths and personalities into consideration. Finally, the underlying theory and skills achieved during this time did eventually lead into a final presentation where we were able to give a brief overview of our project solution. On top of this, we delivered an extensive project proposal that outlined our project scope and deliverables for the next twelve weeks.

Briefly, we decided that we were to follow a mix of agile and waterfall approach as our project management style. In hindsight, our reasoning for choosing both approaches did seem a bit naïve; and it was only when we started developing for our project that we realized that it would not work out. Some further explanations and thoughts behind this point will be discussed soon. From there, we came up with a development cycle that involves information gathering, design, and a feedback loop including developing, testing and feedback (Image 2).

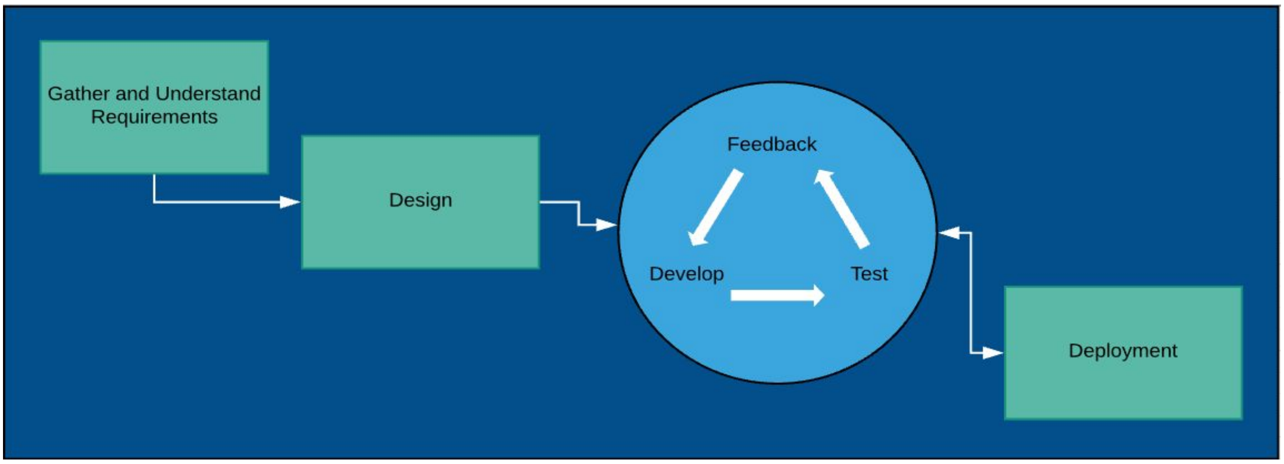


Image 2. Development Cycle

Before our project commenced, our group had also managed to perform some preliminary work around the given dataset. In terms of the development cycle, this stage was aligned with our ‘Gather and Understand Requirements’ phase of the project. This did give our group a perceived advantage as by the time we started off with our project, most of the members did need a quick refresher on what we managed to discover for the species data set. And so it was useful that we had some documentation on what we discovered through this stage of the project.

Finally, with the construction of a risk register that would account for all the setbacks during our time developing our projects, we were fairly confident in commencing our project. We took precautionary measures to alleviate some of the uncertainty that we would potentially face during the commencement of the project. Our risk register give us insights into our go-to action if we were to face difficulties. This would include examples such as if a member did become uncooperative or loss of work/code did occur.

How Was the Project Managed Overall?

Although the outcome of our project deliverables was quite successful, the progression and management of our project overall did have its setbacks. This was more apparent during the first couple of weeks. One member who we decided would assume leadership or 'project manager' of the group did not manage to direct the group in the right direction. As a result, progress on our project during the first few weeks after our project start date fell behind slightly. A change in leadership was required and it was from that point on that we started to head into the right direction. Tasks were delegated to each member each week, and constant updates to our group was made during each meeting (refer to the next section for specifics about our project management style). At times, progress on certain tasks did seem to be stagnant, but it was understandable because it was usually during peak assessment times. Our group did account for this situation in our risk register, and it was our intention that we would catch up during the mid-semester break. Following on from this point in time, we were able to catch up, and meet all the major deadlines for the project.

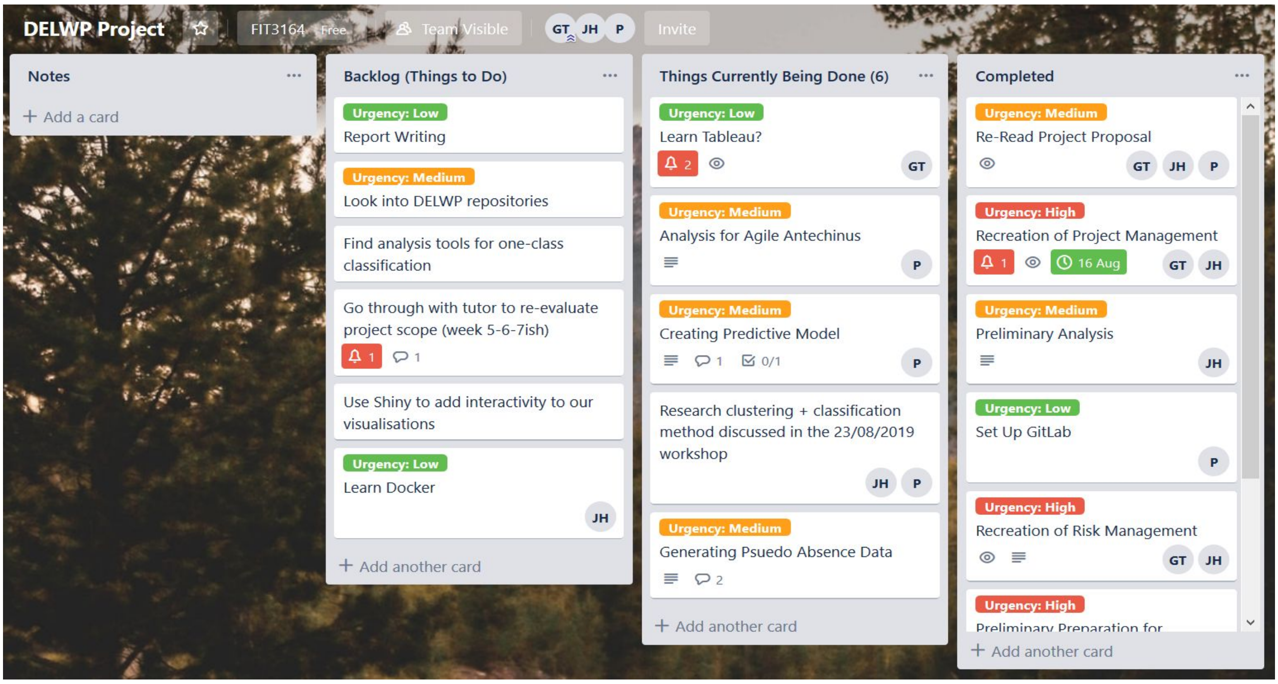
Since the project involved working in a group of three, at times we would have different interpretations and ideas of how we would deliver our project. Subsequently, when a conflict of interest arose from these discussions, we would usually talk through the whole process. From sharing ideas to bringing *in* new ideas; in the end, we would always settle in on an understanding together. This would usually be in the form of a compromise between multiple parties.

So in essence, the management of the entire project did come down to one person. That person would maintain minutes for future reference, promote engagement during weekly meetings, and delegate tasks fairly to one another. However, it also did not rely on just one person managing the project entirely. Engagement of the group also came from everyone who managed to pull through the entire semester and meet their weekly deliverables each week. As a result, we were able to meet our deadlines and deliver a finished product.

Did We Uphold Our Chosen Project Management Style?

As mentioned earlier, our group did have a slow start during the first couple of weeks during our project start date. After a long one month break, most of our group did neglect on our preparation completed during semester one. This included our chosen project management style, our scheduling for the semester, and what we were wanting to accomplish during the first week of the semester. A reminder that we chose to follow a hybrid approach of both agile and waterfall methodology. Because we neglected our chosen project management style, we did not have a structured way of organizing how we were going to manage the group. This became more apparent as we realized that attempting to managing a group of three started to become quite difficult. Considerations such as what each member would do exactly, how we were going to document our discussions, and what our schedule was for the remainder of the semester started to overwhelm our group during the first week. And evidently, during the second week, we were not able to make much progress for the project since we did not uphold our proposal of following an agile and hybrid approach.

After some consolidation, our group decided to dedicate an entire week to researching and coming up with a solid project management style that we would genuinely follow for the remainder of the semester. We opted to stick to one project management style instead of the hybrid waterfall and agile approach that we decided during semester one. We decided to specifically follow the scrum methodology. In a nutshell, the scrum methodology would involve working in a series of sprints. We decided that we would form our sprints to last roughly one week with *some* flexibility. For example, we would make some leniency towards peak assessment weeks. During the meetings we would discuss our current progress as well as what we managed to achieve during our current sprint period. This process is coined the sprint review process. After this discussion, we would then populate our backlog (using Trello) with items that we believed needed to be completed soon. In our case, both scrum and kanban employ a “pull system” where if a certain section of our scheduling system is empty, new card or tasks would get added to the empty column. As a result, this process was exactly what we followed through with each with.


DELWP Project ☆ FIT3164 Free Team Visible GT JH P Invite

Notes

+ Add a card

Backlog (Things to Do)

Urgency: Low
Report Writing

Urgency: Medium
Look into DELWP repositories

Find analysis tools for one-class classification

Go through with tutor to re-evaluate project scope (week 5-6-7ish)
1 1

Use Shiny to add interactivity to our visualisations

Urgency: Low
Learn Docker
JH

+ Add another card

Things Currently Being Done (6)

Urgency: Low
Learn Tableau?
2 GT

Urgency: Medium
Analysis for Agile Antechinus
P

Urgency: Medium
Creating Predictive Model
1 0/1 P

Research clustering + classification method discussed in the 23/08/2019 workshop
JH P

Urgency: Medium
Generating Psuedo Absence Data
2

+ Add another card

Completed

Urgency: Medium
Re-Read Project Proposal
GT JH P

Urgency: High
Recreation of Project Management
1 16 Aug GT JH

Urgency: Medium
Preliminary Analysis
JH

Urgency: Low
Set Up GitLab
P

Urgency: High
Recreation of Risk Management
GT JH

Urgency: High
Preliminary Preparation for

+ Add another card

Image 3. An example of our backlog during week four.

Together, we would select high priority items from the backlog which we would commit for the next sprint.

Selected items are also known as the sprint backlog and we decided that we would limit the sprint backlog to a total of six items (three members multiplied by two = six items max). This means that for a current sprint period, no more than six items were to be completed during one sprint. We decided to implement this feature because we did not want to overwhelm members with too many tasks to meet. This proved successful because the majority of the time, each member was able to meet their deadlines during the end of each sprint period.

In addition to the usual sprint meetings and sprint backlog updates, we hosted a sprint retrospective in order to gauge how our members felt with the group and progress of the project overall. Unfortunately, we were able to only host one sprint retrospective during the semester. Nonetheless, it was an informative discussion where we examined what went well, what went badly, and what could be improved. This feedback enabled us to improve on our setbacks for the remainder of the project. Similar to earlier discussions made, the anonymized feedback included things such as not knowing where to start during the project start date, not having a project management style to follow through with, and having a need to re-explain project objectives as various group members were not on the same page initially.

Project Resources

During the project start date, our group decided that we would start off with setting up our environments for the project. We assigned one member who was most familiar with version control to set up GitLab. This was to ensure that we would maintain version control right away, instead of leaving it to the last minute to configure and maintain. This would establish a high code base quality and enable frequent updates so that all members of the group have access to the latest code.

In conjunction with setting up GitLab, we also assigned one member to start learning Docker. Although not explicitly part of the project objectives, we decided during our project proposal that we would employ the use of Docker containers to ensure access to our application is available across all different operating systems and different versions of the software that we will be utilizing.

From the beginning of the project start date and for the remaining weeks onwards, we also ensured that the version of our R application (application used to create our predictive model) is updated. Additionally, the packages that we will be using are also updated as much as possible.

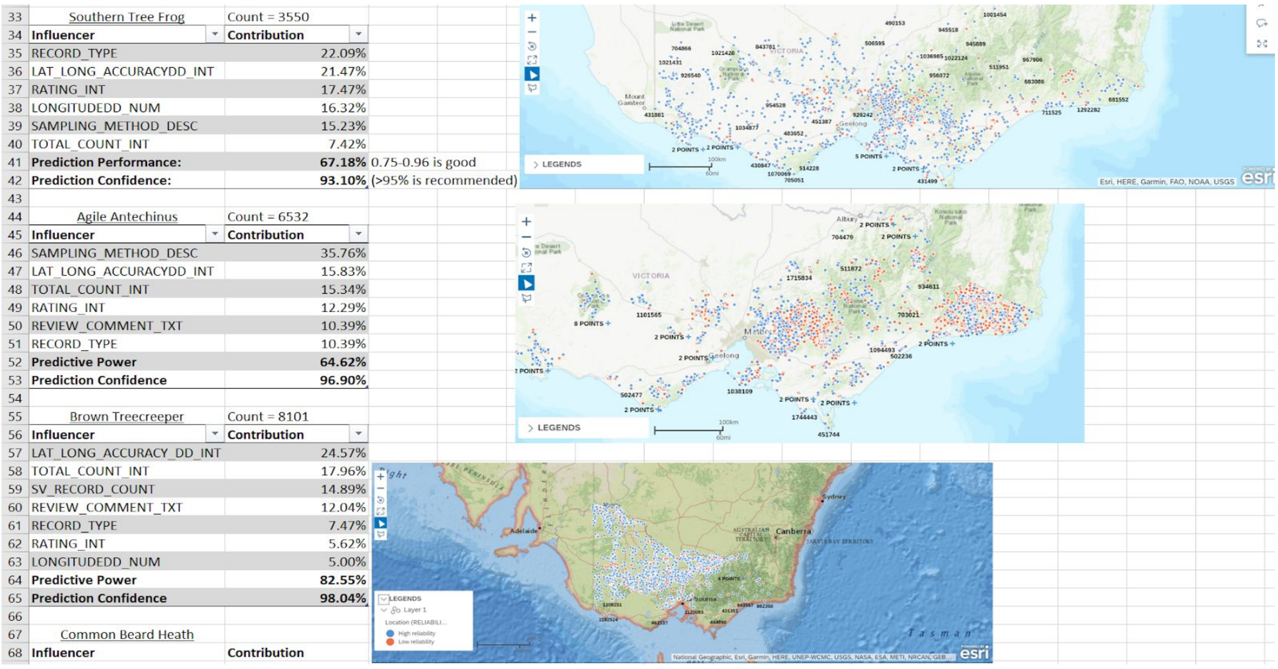
Although our project proposal stated that we were producing a user interface (UI) for our predictive model, we did not specify which application we would use to create our UI. As a result, after a brief discussion during week 2, we decided that we would utilize Tableau to create our UI for our application. A few weeks following on from this point, we realized that creating a dashboard using Tableau for our dataset was not feasible. Integrating between R and Tableau seemed mediocre at best, and decided that we would resort to a different UI entirely. After some discussion, we decided to utilize Shiny; a package within R that allows for building web applications entirely within R. This was favourable as we now only needed to stick to one application through the course of the entire project lifecycle.

Likewise, because we decided to utilize Shiny as our UI for our predictive model, we decided not to use ArcGIS in any shape or form. Similar to our reasoning with not using Tableau, it was an entirely separate piece of software we would have to learn on our own time.

Finally, preparing our environments on our own desktops, we were set with moving on to the planning stage of the project.

Project Planning

Using the Kanban scheduling system, we decided to perform some addition preliminary analysis as most members of the group did not seem to be on the same page. This involved running some fairly simple models on our given dataset. The results from this had showcased the group that little to no attributes within the given dataset had an affect on high or low reliability scores for our species. This breakthrough would then lead us on to realizing that we would require some additional (environmental) data to assist our model with predicting high and low reliability scores for our species. On top of this, the preliminary analysis also involved showing the distribution of all six species (Image 4). Our reason for showing the distribution of these species was so that we would select one species that we could perform our predictive models on. The plan was that we would run one model on one species and then expand the model to run on the remaining five species in the dataset.



33	Southern Tree Frog	Count = 3550	
34	**Influencer**	**Contribution**	
35	RECORD_TYPE	22.09%	
36	LAT_LONG_ACCURACYDD_INT	21.47%	
37	RATING_INT	17.47%	
38	LONGITUDEDD_NUM	16.32%	
39	SAMPLING_METHOD_DESC	15.23%	
40	TOTAL_COUNT_INT	7.42%	
41	**Prediction Performance:**	**67.18%**	0.75-0.96 is good
42	**Prediction Confidence:**	**93.10%**	(>95% is recommended)
43			
44	Agile Antechinus	Count = 6532	
45	**Influencer**	**Contribution**	
46	SAMPLING_METHOD_DESC	35.76%	
47	LAT_LONG_ACCURACYDD_INT	15.83%	
48	TOTAL_COUNT_INT	15.34%	
49	RATING_INT	12.29%	
50	REVIEW_COMMENT_TXT	10.39%	
51	RECORD_TYPE	10.39%	
52	**Predictive Power**	**64.62%**	
53	**Prediction Confidence**	**96.90%**	
54			
55	Brown Treecreeper	Count = 8101	
56	**Influencer**	**Contribution**	
57	LAT_LONG_ACCURACY_DD_INT	24.57%	
58	TOTAL_COUNT_INT	17.96%	
59	SV_RECORD_COUNT	14.89%	
60	REVIEW_COMMENT_TXT	12.04%	
61	RECORD_TYPE	7.47%	
62	RATING_INT	5.62%	
63	LONGITUDEDD_NUM	5.00%	
64	**Predictive Power**	**82.55%**	
65	**Prediction Confidence**	**98.04%**	
66			
67	Common Beard Heath		
68	**Influencer**	**Contribution**	

Image 4. Sample file of our preliminary analysis performed during the beginning of the project. Left: Shows the contribution of various attributes affecting high and low reliability scores. Right: Shows the distribution of each species on the map. Blue values indicate high reliability and red values indicate unknown reliability.

Project Execution

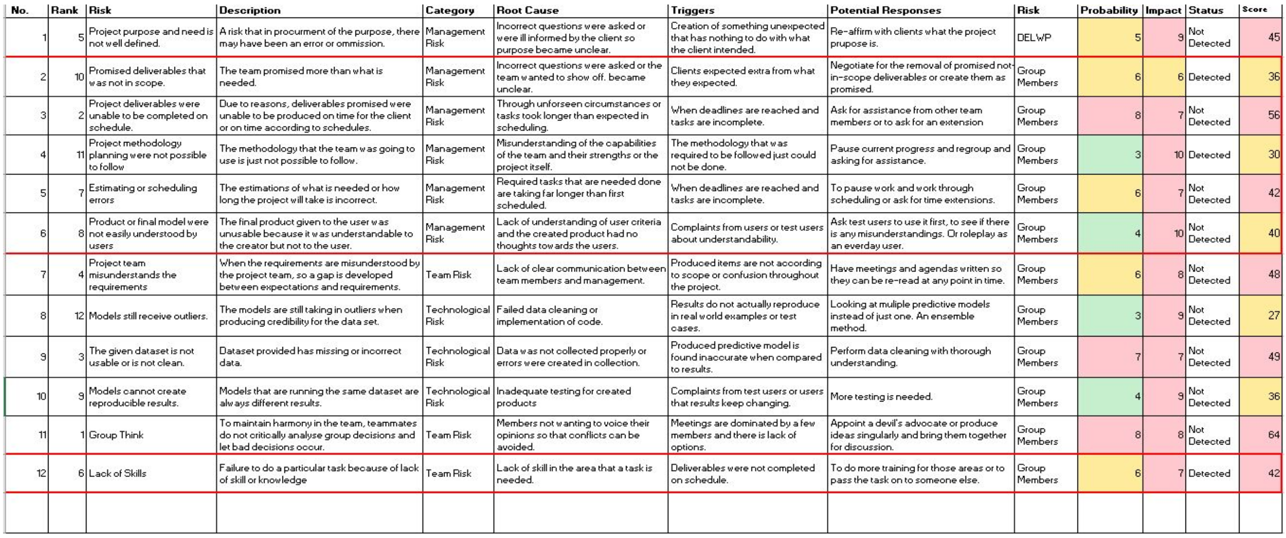
Knowing that we needed environmental predictor variables to support our predictive model, members of the group then separated into finding these environmental variables and researching different predictive models to use. During this time, we decided that we would not utilize the given raster data from DELWP. The reason for this was that it did not seem compatible the R application. Additionally, spending time figuring out how to utilize the given raster data took addition hours from our sprints and so we decided that we would find our own raster data that was compatible with the R application. From this point on, we also tested multiple models including random forests, support vector machines, one-class support vector machines and an ensemble of artificial neural networks, generalized boosted regression models (GBM), generalized linear models

(GLM), generalized additive models (GAM), Maxent and random forests. The list of models after the ensemble methods listed were models performed within the species distribution model package. As a result, this was more of an assortment of algorithms combined together without our input or knowledge of how the models were utilized or combined together. Additional explanations of these various models explored are explained in the methodology section. From this point on, our docker images were created, and a simplistic UI for our model was formed.

For our group the aforementioned tasks were completed using Kanban. Tasks were populated in our backlog and were assigned to different members depending on their strengths and weaknesses. Overall, the group managed to meet their deliverable after each sprint period. Certainly, there were weeks where we could not meet the deadlines we set for ourselves, which is a topic discussed in the next section.

Risk Management

During the project life cycle, we were able to maintain and follow through with our risk register. However, during the first couple of weeks where we did not follow through a project management style, we started realizing that we needed to rely on our risk register. Unfortunately, our risk register did not cover all the situations we were facing at the time. And at the time, we did not have a project management style in place. Therefore, in conjunction with spending an entire week finding a project management style that we eventually followed, we also gave an overhaul to our risk register. This updated risk register would attempt to cover most of the scenarios that we may eventually face during the project life cycle.



No.	Rank	Risk	Description	Category	Root Cause	Triggers	Potential Responses	Risk	Probability	Impact	Status	Score
1	5	Project purpose and need is not well defined.	A risk that in procurment of the purpose, there may have been an error or ommission.	Management Risk	Incorrect questions were asked or were ill informed by the client so purpose became unclear.	Creation of something unexpected that has nothing to do with what the client intended.	Re-affirm with clients what the project prupose is.	DELWP	5	9	Not Detected	45
2	10	Promised deliverables that was not in scope.	The team promised more than what is needed.	Management Risk	Incorrect questions were asked or the team wanted to show off. became unclear.	Clients expected extra from what they expected.	Negotiate for the removal of promised not-in-scope deliverables or create them as promised.	Group Members	6	6	Detected	36
3	2	Project deliverables were unable to be completed on schedule.	Due to reasons, deliverables promised were unable to be produced on time for the client or on time according to schedules.	Management Risk	Through unforseen circumstances or tasks took longer than expected in scheduling.	When deadlines are reached and tasks are incomplete.	Ask for assistance from other team members or to ask for an extension	Group Members	8	7	Not Detected	56
4	11	Project methodology planning were not possible to follow	The methodology that the team was going to use is just not possible to follow.	Management Risk	Misunderstanding of the capabilities of the team and their strengths or the project itself.	The methodology that was required to be followed just could not be done.	Pause current progress and regroup and asking for assistance.	Group Members	3	10	Detected	30
5	7	Estimating or scheduling errors	The estimations of what is needed or how long the project will take is incorrect.	Management Risk	Required tasks that are needed done are taking far longer than first scheduled.	When deadlines are reached and tasks are incomplete.	To pause work and work through scheduling or ask for time extensions.	Group Members	6	7	Not Detected	42
6	8	Product or final model were not easily understood by users	The final product given to the user was unusable because it was understandable to the creator but not to the user.	Management Risk	Lack of understanding of user criteria and the created product had no thoughts towards the users.	Complaints from users or test users about understandability.	Ask test users to use it first, to see if there is any misunderstandings. Or roleplay as an everday user.	Group Members	4	10	Not Detected	40
7	4	Project team misunderstands the requirements	When the requirements are misunderstood by the project team, so a gap is developed between expectations and requirements.	Team Risk	Lack of clear communication between team members and management.	Produced items are not according to scope or confusion throughout the project.	Have meetings and agendas written so they can be re-read at any point in time.	Group Members	6	8	Not Detected	48
8	12	Models still receive outliers.	The models are still taking in outliers when producing credibility for the data set.	Technological Risk	Failed data cleaning or implementation of code.	Results do not actually reproduce in real world examples or test cases.	Looking at muliple predictive models instead of just one. An ensemble method.	Group Members	3	9	Not Detected	27
9	3	The given dataset is not usable or is not clean.	Dataset provided has missing or incorrect data.	Technological Risk	Data was not collected properly or errors were created in collection.	Produced predictive model is found inaccurate when compared to results.	Perform data cleaning with thorough understanding.	Group Members	7	7	Not Detected	49
10	9	Models cannot create reproducible results.	Models that are running the same dataset are always different results.	Technological Risk	Inadequate testing for created products	Complaints from test users or users that results keep changing.	More testing is needed.	Group Members	4	9	Not Detected	36
11	1	Group Think	To maintain harmony in the team, teammates do not critically analyse group decisions and let bad decisions occur.	Team Risk	Members not wanting to voice their opinions so that conflicts can be avoided.	Meetings are dominated by a few members and there is lack of options.	Appoint a devil's advocate or produce ideas singularly and bring them together for discussion.	Group Members	8	8	Not Detected	64
12	6	Lack of Skills	Failure to do a particular task because of lack of skill or knowledge	Team Risk	Lack of skill in the area that a task is needed.	Deliverables were not completed on schedule.	To do more training for those areas or to pass the task on to someone else.	Group Members	6	7	Detected	42

Image 5. Updated risk register

When we re-evaluated our risk register, we identified an additional six risks that we had not considered during the initial construction of our risk register. The updated risk register included situations such as what we had to do if we could not meet our project objectives, if we did not follow through with our chosen project management style, or having a lack of skills to complete certain objectives.

An example of a risk we faced during the project lifecycle was when we had a lack of skill to complete a certain task. During the initial weeks, one member was delegated the task of learning how to run Docker so that we could utilize a Docker container to package our model. When we faced this situation, we resorted to our risk register (probability of said risk being 6/10), and looked for a potential response. Our written response for this situation was to either spend extra time in the area training themselves to complete the task or to pass the task onto someone else in the group. In our case, we allocated some extra time for the person. However, as we judged our group members based on their strengths and weaknesses, we decided to allocate the docker task to a different member. From this point on, the member was successful in learning how to use Docker and integrate it with our entire project. Therefore, by using our risk register, we were able to alleviate all of the risks that we faced during the semester.

For the remaining risks that we also faced during the project lifecycle, we followed through with the potential responses written in the risk register as much as possible. This alleviated some of the stress and unforeseen situations during the project lifecycle.

Limitations Encountered During the Management of Our Project

The initial limitation our group had faced during our project was not upholding to our initial project management style (both agile and waterfall approach). As discussed in the previous sections, we alleviated the situation with an updated risk register, and solidifying on a specific project management style together as a group.

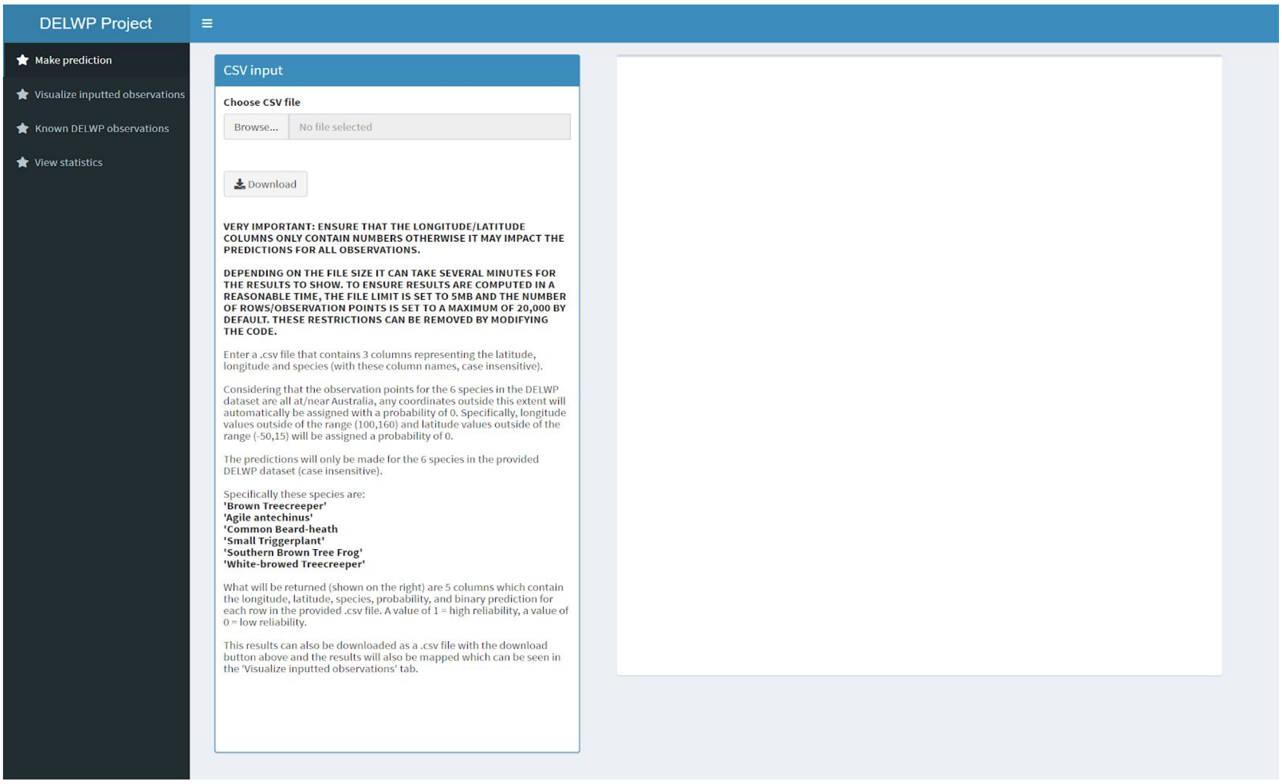
Similar to how our group had managed our risks, there were some limitations encountered during our project. For example, although we did have a momentum going for following our scrum and kanban scheduling system initially, we did seem to struggle to have all members follow through with the methodology. This involved summarizing what we produced during each sprint period, updating the backlog, and allocating new tasks to different members. At times, members did not follow through with this structured way for our discussions during our weekly meetings. Additionally, members did not seem to see the value in minute taking which resulted in some poor quality documentation of our minutes. Constant reminders of our structured way of forming a discussion was required to inform all members of following the scrum and kanban methodology. In hindsight, we were quite successful in following through with the methodology as much as possible. This became more apparent when the complexity of the project started increasing during the final stages of the project. Code became more difficult to read as we started building on our models and UI and Docker started to become difficult to understand by members who did not know how to use Docker. Subsequently, our minutes became very important as a reference for each member. Since all members of the group were also tackling other subjects, it became difficult to remember what some of our discussions were made during some weeks. It was convenient that we were able to use our minutes to gain insights into what we discussed previously.

An additional limitation we faced during the semester was not following through with the scrum and kanban methodology during the later stages of our project. As the weeks neared the project live demo date, the group started to lose focus in following the scrum and kanban methodology. No tasks were being updated in the backlog, nor were tasks being assigned to each member. The potential response in the risk register for this situation was to regroup and re-evaluate ourselves. This was to make our group members aware of their situation and follow through with our project management style. However, an unexpected repercussion from that situation was that each member knew exactly what they had to do each week prior to the live product demo during week ten. Furthermore, most members in the group were frequently updated with the whereabouts of the project. Therefore, it could be said that we did not need to utilize kanban during this time as we knew exactly what we needed to deliver. In this case, we did not follow through with our potential response because the momentum of the group's progress was at an all time high. Therefore, in order to not disrupt this momentum, the *project manager* decided not to follow through with the written potential response in our risk register.

Outcomes

Results

The outcome of our project delivered a docker container that encapsulates our predictive model and Shiny UI application. Users who use the application will have the ability to run the program on a docker container making it portable and compatible with any operating system. Additionally, users will have the option to run the Shiny application as a web service, making it easily accessible on any desktop without having the need to use Docker. However, this feature comes at a monthly cost if users wish to keep the application running.



VERY IMPORTANT: ENSURE THAT THE LONGITUDE/LATITUDE COLUMNS ONLY CONTAIN NUMBERS OTHERWISE IT MAY IMPACT THE PREDICTIONS FOR ALL OBSERVATIONS.

DEPENDING ON THE FILE SIZE IT CAN TAKE SEVERAL MINUTES FOR THE RESULTS TO SHOW. TO ENSURE RESULTS ARE COMPUTED IN A REASONABLE TIME, THE FILE LIMIT IS SET TO 5MB AND THE NUMBER OF ROWS/OBSERVATION POINTS IS SET TO A MAXIMUM OF 20,000 BY DEFAULT. THESE RESTRICTIONS CAN BE REMOVED BY MODIFYING THE CODE.

Enter a .csv file that contains 3 columns representing the latitude, longitude and species (with these column names, case insensitive).

Considering that the observation points for the 6 species in the DELWP dataset are all at/near Australia, any coordinates outside this extent will automatically be assigned with a probability of 0. Specifically, longitude values outside of the range (100,160) and latitude values outside of the range (-50,15) will be assigned a probability of 0.

The predictions will only be made for the 6 species in the provided DELWP dataset (case insensitive).

Specifically these species are:
'Brown Treecreeper'
'Agile antechinus'
'Common Beard-heath
'Small Triggerplant'
'Southern Brown Tree Frog'
'White-browed Treecreeper'

What will be returned (shown on the right) are 5 columns which contain the longitude, latitude, species, probability, and binary prediction for each row in the provided .csv file. A value of 1 = high reliability, a value of 0 = low reliability.

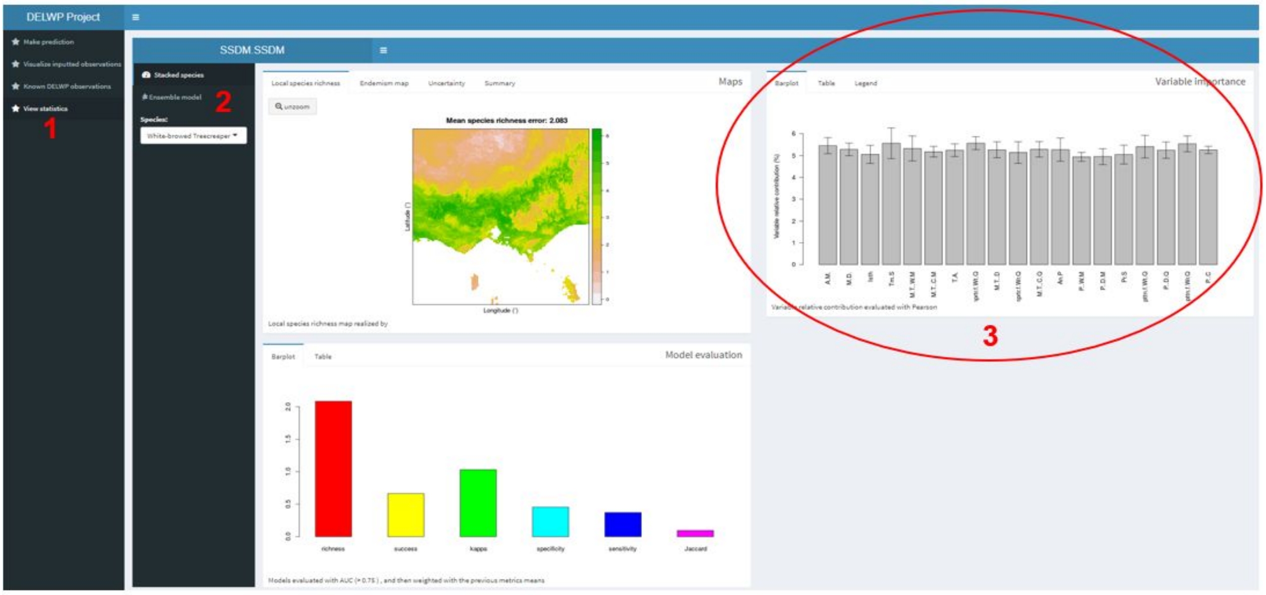
This results can also be downloaded as a .csv file with the download button above and the results will also be mapped which can be seen in the 'Visualize inputted observations' tab.

Image 6. Users are greeted with a homepage when running our predictive model application.

Within the web application, users are greeted with a homepage that allows users to upload .csv files containing species distribution data. Additional help & documentation is provided through various text fields in the application, including what file formats are accepted, how the .csv files should be structured, and what species are accepted in the web application.

How Are the Initial Requirements Met

The minimal viable product of our project was to create a predictive model using R for six species within Victoria. The predictive model model was to sort new observations into high and low reliability categories. The reason behind using R was because DELWP has an inhouse employee that is knowledgeable with the R programming language. As a result, it was ideal that the project were to be completed using this language. Ever since we presented our solution as a product demo, we have been able to produce said model using a random forest model. This model utilizes nineteen different environmental variables as predictors for our model. An additional requirement from DELWP was to also identify predictors (variables) that have the most impact on successful categorisation. Although not necessarily a strict requirement from DELWP, we have also encapsulated our predictive model as a web application using the Shiny package within R. For our project, we have showcased this information by going through the "View Statistics" tab, and "Ensemble Model". This process is also highlighted in image 4.



Local species richness map realized by

Models evaluated with AUC (> 0.75) , and then weighted with the previous metrics means

Variable relative contribution evaluated with Pearson

Image 7. Steps on how to view predictors that have the most impact on successful categorisation.

An example of an environmental predictor variable that seemed to have contributed to the reliability of the species data includes the 'Temperature - Seasonality' variable. This does seem to correspond to some level of rationality as species do tend to situate themselves at ideal seasonal temperatures in Victoria.

Additional requirements such as running the application on a Docker container has also been met. Users will have the ability to make the application portable across different systems without the annoyance of having to maintain different versions of the application and packages utilized in this web application. Furthermore, users will have the ability to run the shiny application as a web service - a further requirement declared by DELWP.

Justification of Key Decisions Made

During the development of our project, our group had decided early on that we would put our focus on producing a reliable predictive model first - and then focus on the additional requirements such using Docker and producing a UI second. As a result, the user interface of our application may look somewhat clunky. It was convenient that the Shiny application also had a package that related to species distribution modelling. From that point on, we decided to use this package to assist us with developing the user interface of our predictive model.

Since we decided to focus on the predictive model itself, various models were explored. These models included artificial neural networks, random forests, support vector machines, one-class support vector machines and xgboost. Reasons for selecting random forests have been stated in the testing report. However, to explain briefly, especially for models such as support vector machines and xgboosting (not explained extensively in the test report), we had decided to stick to the random forest model.

The reason for not selecting support vector machines in our case, was because the results were remarkably similar to the results shown by our random forest models. This holds true for one-class support vector machines - a model that only relied on one of the two binary values to predict on. Because of the results, we had opted to use random forests solely through the results given by other researchers; which we had explored through our literature review.

Moreover, we decided to not use xgboost - an alternative random forest model that relied on the boosting algorithm. A consequence of using a model that utilizes a boosting algorithm, is the fact the models have a tendency to overfit. For our dataset with the supported environmental predictor values, we have seen the xgboost model constantly overfit in comparison to the random forest model. Again, through these results, we had resorted to the random forest model instead.

Additional key decisions that were made for our project, was also the homepage of our application. After the review of our product demo, we decided that it would be best that we would greet users with the ability to upload .csv files containing the species. Previously, we had displayed a distribution map of six species. This became counter intuitive as the map seemed to have served no purpose. Therefore, by greeting users the ability to upload their data, it should enhance the usability of the web application by a large margin.

Discussion

As it stands, the current results of our project showcases a random forest model that predicts on nineteen different environmental predictors. Furthermore, since we have managed to show the level of contribution each environmental predictor variable has on the output of the model, we have the ability to analyze our results in detail.

We will discuss our analysis on one species as it was the majority of our focus when curating our predictive model. We will then discuss the wider implications of the model for the selected species for the remaining five species. The species that our project had focused on for the duration of the development cycle was Agile Antechinus.

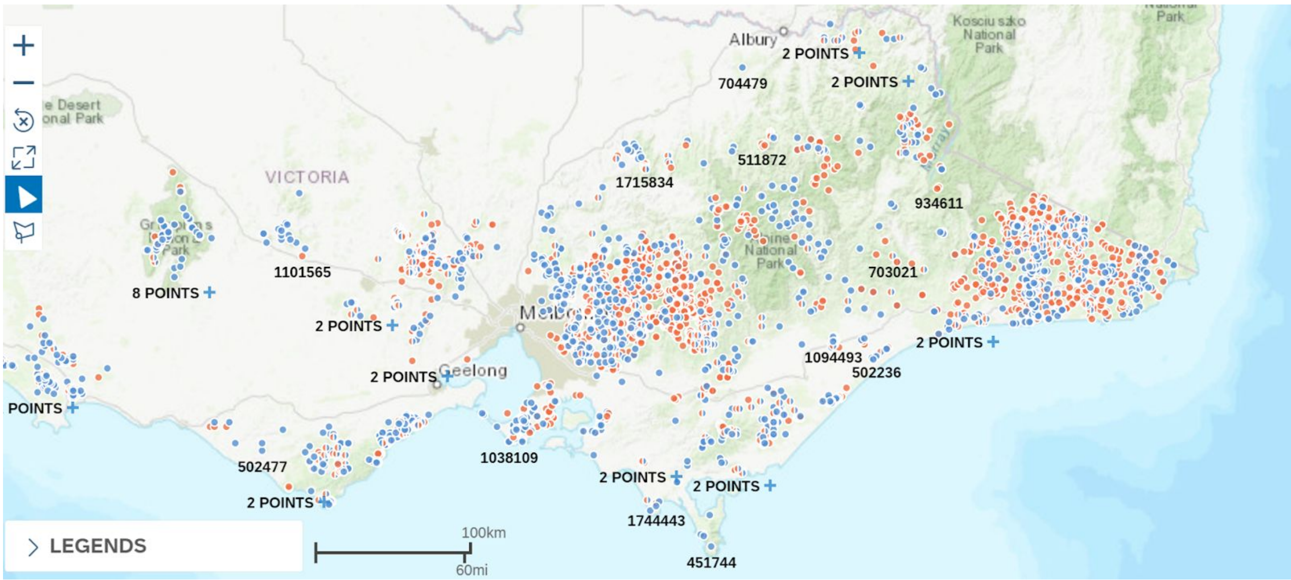


Image 8. Distribution of the Agile Antechinus species in Victoria. Blue values indicate high reliability and red values indicate unknown reliability.

The reason behind choosing this specific species was mainly through its' distribution within Victoria. Using image 8, we can see that there are visible clustering of the Agile Antechinus species. Upon further research of the species' habitats, we can see that it supports the claim of the species living in mostly wet or moist forest areas within Victoria. Therefore, in order to produce a robust predictive model, we should have a model that outputs high reliability scores for data points in Victoria that contain wet or moist forest areas.

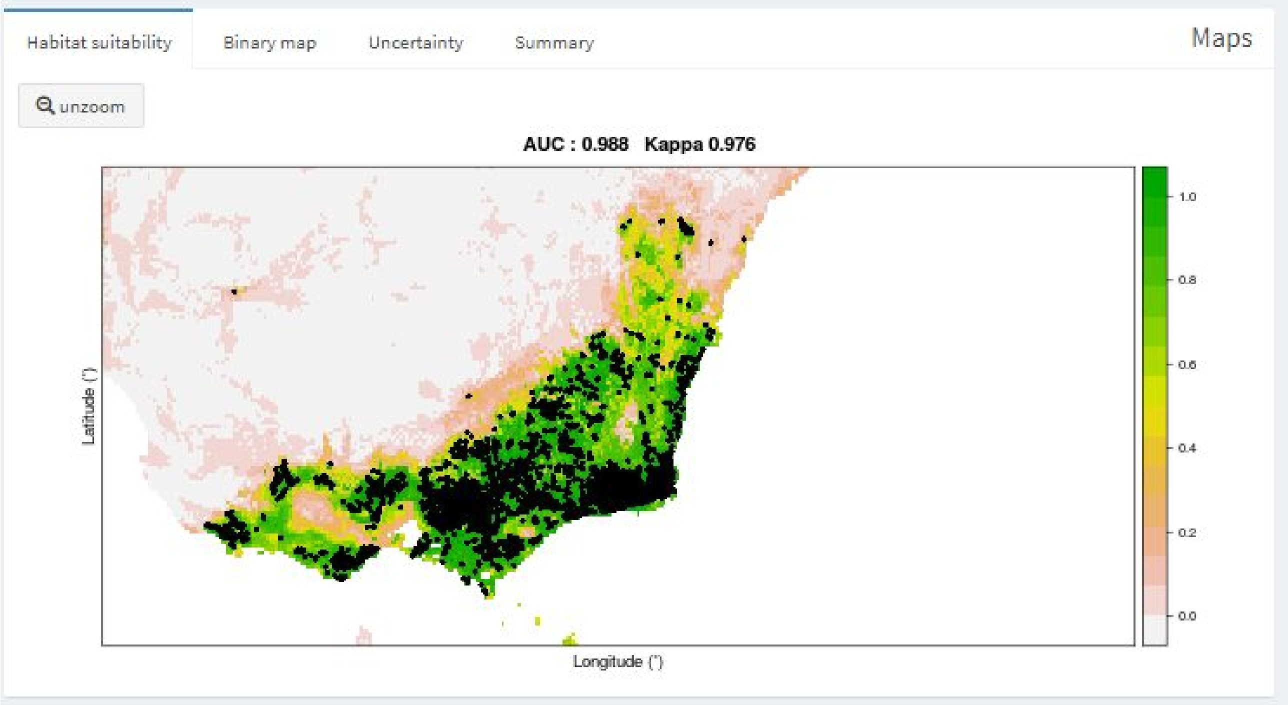


Image 9. Our random forest model output for high and low reliability scores for the Agile Antechinus species.

Using the binary map in image 9, the model has produced a significantly accurate clustering of the Agile Antechinus species. It is correct to say that essentially all of our high reliability species have been clustered into the high reliability binary map (green values). This is further supported by two of our accuracy measures used: the area under the curve (AUC) and kappa. A value of 0.988 AUC and 0.976 kappa indicates a highly accurate random forest model for the species analyzed.

Unfortunately, it could it said that the random forest model is still overfitting the species. With an AUC score of 0.988, it is unlikely that our model is able to accurately predict new observations that contain clusters outside the range of the binary map (Image 9: green values). Furthermore, despite performing a 5-fold cross validation, which is essentially resampling our training and testing data five times, the accuracy measures still remain the same.

Fortunately, for data observations that are contained within the model, it is likely to predict accurately despite the model seemingly overfitting the model. This is further tested using our *unknown* reliability data points for Agile Antechinus. Testing the model on the data points with these unknown reliability scores yielded approximately 100 different Agile Antechinus species data points with low reliability scores (Image not shown). However, we will not know the true level of accuracy of our model until we have confirmed that the unknown reliability data points have been separated into their actual high or low reliability counterparts.

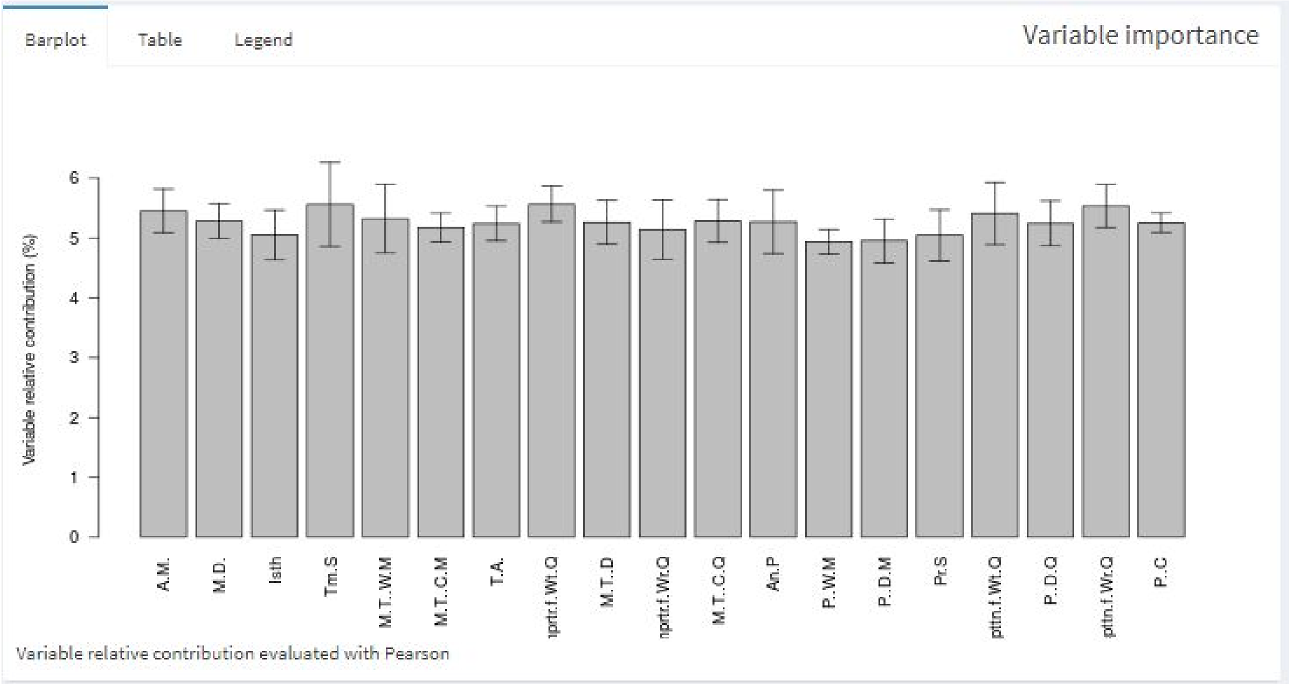


Image 10. Level of significance for our nineteen environmental predictor variables for the Agile Antechinus species.

Following on from the model, the Shiny application has the ability to output the variable importance for all nineteen environmental variables used in our predictive model (Image 10). From the image, we can see each level of significance ranging from 4.8% to 6% contribution to our predictive model.

Using the results from this image, we can see that all nineteen environmental predictor variables have seemed to have given equal levels of importance for each predictor. This seems to be quite inaccurate as for the Agile Antechinus, predictor variables such as precipitation (last 6 bar charts in Image 10) should have a high level of significance because of the species having a tendency towards living in habitats with moist or wet forest areas. This problem could be attributed to the wrong environmental predictor variables utilized for our dataset. Since most of the environmental variables relate to temperature and precipitation, it may be quite difficult to infer species distribution simply using these two predictors. Therefore, we can infer from this bar plot that the random forest model may not be as accurate as we would have liked.

Overall, from the results given for the Agile Antechinus, we can also expand the model output to the five remaining species. All five species have yielded similar AUC and Kappa accuracy scores. Furthermore, the variable importance bar plots (Image 10) for all five remaining species yielded similar results with the level of importance for all environmental predictors ranging from 5-6% contribution to the model each. This implies that our model may still have a tendency to overfit. The impact of the overall results of our model does seem to imply that the accuracy of our model may be quite questionable. Despite our rigorous testing of various training and testing data and the utilization of our unknown reliability data points, it is still difficult to discern the overall fit of our random forest model. With that being said, one concluding point can be made clear from the overall results (Image 9 and Image 10): that our random forest model has a high probability of overfitting.

Limitations of Project Outcomes

Throughout the entire project, many setbacks were faced which caused us to set limitations for our project outcomes. We will attempt to address most of the limitations for our project deliverables.

Following on from the discussion made previously, our model does seem like it is overfitting our dataset. Therefore, the accuracy of our chosen predictive model does seem to be quite questionable.

Although we have successfully created a front-end web application, the user interface can be seen as quite clunky. Unfortunately, user experience was not kept at a high priority because of the packages we utilized to create our Shiny application. Since we utilized the SDM package within the Shiny package, we utilized this feature to create our entire UI. Therefore, any adjustments that we would want to make in terms of the UI, would not be possible. This is further amplified with some of the fields in our web application being empty; it is a part of the UI that we cannot make changes to unfortunately.

Additionally, when users are uploading new species data into our web application, the file sizes are limited to five megabytes, with an additional limitation of 20,000 species data points. The reason for artificially creating this limitation is because data points and file sizes over this limit would create a significant slowdown in the computation of our random forest model. Furthermore, there is a significant slowdown in the visualization of our data when large amounts of data are uploaded. However, users can bypass this artificial limitation on the upload size by editing the R source code to increase the upload sizes.

Another limitation that we faced is through the visualization of our data when the data is uploaded. Unfortunately, users will not have the ability to visualize the distribution of their data in map form. Users will have to directly download the updated model with their associated reliability scores and visualize the data points elsewhere. Due to time constraints, it was a feature we were unable to complete before the project end date.

Finally, our last limitation is that the model is unable to update itself. What this means is that future data points that are made to predict using our model does not update our random forest model. The reason behind not updating our model is because there is a high computational cost when updating the model. Updating the model would take roughly thirty minutes to an hour and a half to compute. We have attempted to alleviate this limitation by having the library contain a function that enables us to update the models manually using the R source code. Albeit, no documentation is provided to address this specific limitation due to time limitations.

Additional Improvements and Possible Future Works

Within our project, there are many additional improvements that could be made if there were no time constraints.

The first and most important improvement would be to perform additional research on our predictive model and additionally, attempt to fine tune our model to not overfit. For example, we could further fine tune our parameter settings for the model or start exploring different variations of pseudo-absence data that is used to support our model. We could also potentially explore different environmental predictor variables to further support our model. This may alleviate some of the concerns regarding the results of our model.

The second improvement would be to attempt to build the UI of our application from the ground-up to address the clunky UI. The distribution map of our species can be quite sluggish at times and the design of some of the statistical information could be further improved to display relevant information only. The reason we did not complete this in the beginning was because of the high learning curve required to build a shiny application. On top of having to produce a robust predictive model and teach ourselves Docker, we would have to also learn the ins and outs of producing an exceptional Shiny application. Unfortunately, we did not have the time to complete this specific task, and therefore resorted to using a pre-built Shiny UI application that met most of the criteria that we gave to the UI.

Finally, we are unable to adjust parameter settings for our predictive model. It has been noted that it was an optional and additional requirement from DELWP to the ability to adjust different parameters for our random forest model. Therefore, spending some time adding this feature to our product could increase the reliability of our random forest model. Likewise, because we are unable to add additional environmental (raster) data to support our predictive model, it could be an additional improvement to our product if this feature were to be added. This would enable us the ability to easily explore different environmental predictor variables without the restriction on predicting on our pre-selected nineteen environmental data variables that we used to support our random forest model.

Conclusion

The overall outcome of our project has been a successful process. As a group of three, we have been able to follow through the Scrumban methodology from start to end. The result of following through our chosen methodology was that we were able to meet most of our weekly sprints and deliver outcomes that contributed to the production of our final product. This included tasks such as learning Docker and producing a docker container, to learning how to make use of the Shiny package within R so that we could produce a user interface to show our predictive model results. And despite some setbacks faced during the first couple of weeks of our project start date, we were able to meet all initial and additional requirements set by DELWP. There are certainly limitations within our product delivered, however, given the constraints and conditions that we set ourselves in, it resulted in the most optimal outcome that we could possibly achieve as a group of three. Furthermore, additional requirements have been documented so that any individual can potentially improve on this product.

References

Araújo, M., & Guisan, A. (2006). Five (or so) challenges for species distribution modelling. *Journal Of Biogeography*, *33*(10), 1677-1688. doi: 10.1111/j.1365-2699.2006.01584.x

Arthur Rylan Institute for Environmental Research. (2019). Habitat distribution models (HDMs). Retrieved from https://www.ari.vic.gov.au/research/modelling/habitat-distribution-models-hdms

Barbet-Massin, M., Jiguet, F., Albert, C., & Thuiller, W. (2012). Selecting pseudo-absences For species distribution models: how, where and how many?. Methods In Ecology And Evolution, 3(2), 327-338. doi: 10.1111/j.2041-210x.2011.00172.x

Bonney, R., Shirk, J., Phillips, T., Wiggins, A., Ballard, H., Miller-Rushing, A., & Parrish, J. (2014). Next Steps for Citizen Science. *Science*, *343*(6178), 1436-1437. doi: 10.1126/science.1251554

Bucklin, D., Basille, M., Benscoter, A., Brandt, L., Mazzotti, F., & Romañach, S. et al. (2014). Comparing species distribution models c onstructed with different subsets of environmental predictors. *Diversity And Distributions*, *21*(1), 23-35. doi: 10.1111/ddi.12247 (Bucklin et al., 2014)

De'ath, G. (2007). Boosted Trees for Ecological Modeling and Prediction. *Ecology*, *88*(1), 243-251. doi: 10.1890/0012-9658(2007)88[243:btfema]2.0.co;2

Guillera-Arroita, G., Lahoz-Monfort, J., Elith, J., Gordon, A., Kujala, H., & Lentini, P. et al. (2015). Is my species distribution model fit for purpose? Matching data and models to applications. *Global Ecology And Biogeography*, *24*(3), 276-292. doi: 10.1111/geb.12268

Guisan, A., & Thuiller, W. (2005), Predicting species distribution: offering more than simple habitat models*. Ecology Letters*, *8*, 993-1009. doi:10.1111/j.1461-0248.2005.00792.x

Guisan, A., & Zimmermann, N. (2000). Predictive habitat distribution models in ecology. *Ecological Modelling*, *135*(2-3), 147-186. doi: 10.1016/s0304-3800(00)00354-9

Guo, C., Lek, S., Ye, S., Li, W., Liu, J., & Li, Z. (2015). Uncertainty in ensemble modelling of large-scale species distribution: Effects from species characteristics and model techniques. *Ecological Modelling*, *306*, 67-75. doi: 10.1016/j.ecolmodel.2014.08.002

Hernandez, P., Graham, C., Master, L., & Albert, D. (2006). The effect of sample size and species characteristics on performance of different species distribution modeling methods. *Ecography*, *29*(5), 773-785. doi: 10.1111/j.0906-7590.2006.04700.x

Ho, J., Truong, G., Joshi. P., (2019). DELWP Project Proposal. FIT3163 Monash University. Pg 4-8. https://docs.google.com/document/d/1I4gB3jxIPfEMc0DNshgBQVgB7KHzFSSYbRpknVoREbc/edit#heading=h.j93z8w2dijzk

Howard, C., Stephens, P., Pearce-Higgins, J., Gregory, R., & Willis, S. (2014). Improving species distribution models: the value of data on abundance. *Methods In Ecology And Evolution*, *5*(6), 506-513. doi: 10.1111/2041-210x.12184

Huijbers, C. (2016, May 10). Absence data. Retrieved from https://support.bccvl.org.au/support/solutions/articles/6000127043-absence-data.

Jaberg, C., & Guisan, A. (2001). Modelling the distribution of bats in relation to landscape structure in a temperate mountain environment. *Journal Of Applied Ecology*, *38*(6), 1169-1181. doi: 10.1046/j.0021-8901.2001.00668.x

Jeffrey, S.E., Melanie, A.M., Zachary, A.H. & Samuel, A.C. (2011). Modelling Species Distribution and Change Using Random Forest. *Predictive Species and habitat Modeling in Landscape Ecology: Concepts and Applications,* 139-159. doi: 10.1007/978-1-44-19-7390-0_8

Lin, Y., Deng, D., Lin, W., Lemmens, R., Crossman, N., Henle, K., & Schmeller, D. (2015). Uncertainty analysis of crowd-sourced and professionally collected field data used in species distribution models of Taiwanese moths. *Biological Conservation*, *181*, 102-110. doi: 10.1016/j.biocon.2014.11.012

Lin, Y., Lin, W., Lien, W., Anthony, J., & Petway, J. (2017). Identifying Reliable Opportunistic Data for Species Distribution Modeling: A Benchmark Data Optimization Approach. *Environments*, *4*(4), 81. doi: 10.3390/environments4040081

Liu, C., White, A., & Newell, G. (2018). Detecting outliers in species distribution data. *Journal of Biogeography*, *45*, 164-176. doi:10.1111/jbi.13122

Liu, C., White, M., Newell, G., & Griffioen, P. (2013). Species distribution modelling for conservation planning in Victoria, Australia. *Ecological Modelling*, *249*, 68-74. doi: 10.1016/j.ecolmodel.2012.07.003

Lobo, J., Jiménez-Valverde, A., & Hortal, J. (2010). The uncertain nature of absences and their importance in species distribution modelling. Ecography, 33(1), 103-114. doi: 10.1111/j.1600-0587.2009.06039.x

Merow, C., Smith, M., & Silander, J. (2013). A practical guide to MaxEnt for modeling species' distributions: what it does, and why inputs and settings matter. *Ecography*, *36*(10), 1058-1069. doi: 10.1111/j.1600-0587.2013.07872.x

Miyamoto, A., Tamura, N., Sugimura, K., & Yamada F. (2004). Predicting Habitat Distribution of the Alien Formosan Squirrel Using Logistic Regression Model. *Global Environmental Research, 8(1),* 13-21. Retrieved from http://www.airies.or.jp/attach.php/6a6f75726e616c5f30382d31656e67/save/0/0/08_1-02.pdf

Newbold, T. (2010). Applications and limitations of museum data for conservation and ecology, with particular attention to species distribution models. *Progress In Physical Geography: Earth And Environment*, *34*(1), 3-22. doi: 10.1177/0309133309355630

Phillips, S., & Dudík, M. (2008). Modeling of species distributions with Maxent: new extensions and a comprehensive evaluation. *Ecography*, *0*(0), 080328142746259-???. doi: 10.1111/j.0906-7590.2007.5203.x

Rousseeuw, P., & Hubert, M. (2011). Robust statistics for outlier detection. *Wiley Interdisciplinary Reviews: Data Mining And Knowledge Discovery*, *1*(1), 73-79. doi: 10.1002/widm.2

Thibaud, E., Petitpierre, B., Broennimann, O., Davison, A., & Guisan, A. (2014). Measuring the relative effect of factors affecting species distribution model predictions. *Methods In Ecology And Evolution*, *5*(9), 947-955. doi: 10.1111/2041-210x.12203 (Thibaud, Petitpierre, Broennimann, Davison & Guisan, 2014) (Thibaud et al., 2014)

Wang, Y., & Stone, L. (2018). Understanding the connections between species distribution models for presence-background data. *Theoretical Ecology*, *12*(1), 73-88. doi: 10.1007/s12080-018-0389-9

Ward, G., Hastie, T., Barry, S., Elith, J., & Leathwick, J. R. (2008). Presence-Only Data and the EM Algorithm. *Biometrics*, *65*(2), 554–563. doi: 10.1111/j.1541-0420.2008.01116.x

Wilson, K., Westphal, M., Possingham, H., & Elith, J. (2005). Sensitivity of conservation planning to different approaches to using predicted species distribution data. *Biological Conservation*, *122*(1), 99-112. doi: 10.1016/j.biocon.2004.07.004

Zellweger, F., Baltensweiler, A., Ginzler, C., Roth, T., Braunisch, V., Bugmann, H., & Bollmann, K. (2016). Environmental predictors of species richness in forest landscapes: abiotic factors versus vegetation structure. *Journal Of Biogeography, 43(6),* 1080-1090. doi: 10.1111/jbi.12696