UE19CS312 - Data Analytics

PES University, CSE Department

UTILE DOLIS

An Alum's Guide to Acing the DA Course Project

Adithi Satish

Akhil Eppa

Vibha Kurpad

Table of Contents

1. Picking a Problem Statement

Hi there! Welcome to ***Utile Dolis***, An Alum's Guide to Acing the DA Course Project! The project component in Data Analytics is one of the most important components of the entire course. It is meant to test your ability to apply the concepts taught on real-world data in order to gain insights, make predictions/recommendations and derive business value.

Throughout the course of the project, you may find yourself doing one or more of these things: applying different preprocessing methods to clean your dataset and perform feature engineering, plotting vibrant graphs you want to stare at all day, trying multiple models and their diagnostic tests to compare performance metrics, humming the theme to your favourite TV show during that excruciatingly long wait while your model trains and finally, extensively documenting the treasure trove of information your data has revealed.

But wait, we've gotten ahead of ourselves. Let's start at the very beginning. You've picked your teammates, and all of you are raring to go. Now all you have to do is pick a problem statement. A piece of cake, you think? Well...not quite. Deciding your problem statement can often be one of the most confusing things to do. We've been through it too, so here's what we recommend:

1. Have a brainstorm session with your teammates. Find out *what* interests each of you and *why* - this can be an interdisciplinary area of application (say economics, edtech or healthcare), or a particular domain in data analytics you want to explore (time series, recommender systems, etc). An intersection of interests is very important because it gives your team the gusto needed to power you all the way to the finish line. It also helps you come up with a storyline for your problem statement and identify areas of application.
2. Browse through Kaggle Datasets as well as Kaggle Competitions. Kaggle Competitions are a great resource because not only do you have a problem statement defined for you, you don't need to spend any longer searching for your dataset (two birds, one stone)!

3. Check out if you can get real world data that interests you. This could be anything from data related to funding for startups to data related to restaurants in Bengaluru. Once you start searching, you would discover a treasure trove of resources.
4. Finally, if none of the above steps have been fruitful (try them again, you guys!), googling "data analytics project ideas" should be your last resort.

Authors' Notes:

Adithi: My team and I realized that we all wanted to explore problem statements related to macroeconomics and global development factors. So, we started looking into the UN Sustainable Development Goals, World Economic Forum, World Bank, etc., while brainstorming to see if we could come up with a potential problem statement. We hit the jackpot when we came across the World Development Indicators - statistics about global development collected by the World Bank, going all the way back to the 60s. A quick call had us settled on our problem statement, *"Analyzing the Influence of Various Factors on Global Economies"*.

Akhil: My team and I first began looking for datasets before finalizing on the problem statement. We had many brainstorming sessions before finalizing on a dataset. We made it a point to suggest a few datasets each before taking a final decision. Finally we decided to go ahead with a Kaggle Contest because not only does it give us a real world dataset but also helps us assess our solution with all the other participants and gives us a good idea where we stand among the crowd. The contest we chose involved a challenging time series dataset contributed by one of the largest Russian software firms.

Vibha: My team and I, similar to Akhil's team approach, first began looking for datasets before finalizing on the problem statement. Personal suggestion, if you're absolutely clueless on what domain/industry you want to do your project in, going through Kaggle and seeing the various datasets available definitely helps. You get a sense of what kind of data is available to play with and the brainstorming of potential problem statements or even creating various hypotheses kicks in. After spending a good 3-4 hours we narrowed down on two finance related datasets. With finance data, the plus point we realised was not only could we do a technical analysis of the data, but it would also give us an opportunity to learn about various market conditions/terms and also the impact if any of major global events on prices(this is the potential hypothesis creation I was talking about); and since our choice was between analysing stock prices and cryptocurrency, we chose to go with the road less taken i.e cryptocurrencies.

2. Choosing the Right Dataset(s)

Ooh, look who's got their problem statement ready! Or, maybe you're still on the fence about it. Don't worry, you do not need to have a problem statement in hand before you search for datasets. Oftentimes, the two steps are interchangeable and depend on what resources you're able to gather and what you're interested in exploring.

Here are two possible situations that you might find yourselves in, and what we recommend for it:

1. You've decided on a problem statement and need to find/collect data. There is an overwhelmingly large amount of data available on the Internet. From curated datasets on Kaggle (it's got everything from movie reviews to crime statistics, and more!) to domain specific datasets, for example, Global Health Observatory, UNdata, Datasets from the Indian Government, NASA Open Data Portal, etc, data is available online for a host of problem statements. In addition to this, many universities also provide datasets online for free: UCI Machine Learning Repository, Stanford Large Network Dataset Collection, etc. ***Bonus: a lot of public GitHub repositories also have datasets - use the right keywords in the search bar and bingo ;)***

   But…what if you aren't able to find anything that is appropriate for the problem statement you have in mind? Don't panic, but it's time to buck up, because data collection is the way to go! You can collect your data using different methods like surveys, polls, focus groups, and more. You could also scrape data from the internet (both Python and R provide helpful libraries like BeautifulSoup and RVest respectively, that you can use to scrape data). However, a note of caution, when you decide to collect data on your own, be prepared to spend more time on preprocessing (a lot of the datasets you find on Kaggle are precleaned to an extent) and cleaning your data. Also, do not forget to account for any kind of bias (voluntary/non-voluntary) that may pose a problem while modelling.

2. You've found a couple of datasets, but are yet to formulate a problem statement. At this point, you really should have a brainstorming session with your

teammates (in case you haven't already). Look through documentation for each of the datasets that you've narrowed down on and try to find the answers to some of these questions:

a. What are the attributes?
b. How might they be related to each other?
c. What does the data mean?
d. Is there a clearcut target variable?
e. If there are multiple files in your dataset, how do you think they are related?
f. How well can you visualize your data?
g. What kind of preprocessing might be necessary?

Answering those questions would be the first step in identifying potential areas of analysis using said data. Once you have an inkling of what you want to model or predict, it's time to figure out *why* you want to do that. What's the purpose your analysis would serve in terms of business value? What areas of application can it be used in? More often than not, this would require some amount of domain knowledge (there's no need to go into great detail at this point - you can save this for your literature survey!)

We believe that the *what* and *why* are the most important questions that need to be answered before you move on to the next phase - *how* are you going to solve this problem?

Note: We want to reiterate that choosing a problem statement and a dataset can be a simultaneous process; they are listed in this guide in that particular order only to enhance lucidity.

3. Reviewing Literature

Congratulations, you've figured out what you want to do and why it is important, and now you need to find a place to start! Reviewing literature can be very daunting especially if this is your first time doing so. As is the case with data, the amount of research literature out there is intimidating (and yes, that's an understatement). But enough with our theatrics, here are three steps we recommend you follow:

1. Finding the right research papers: Start by searching for papers related to the problem statement you have chosen on websites like Google Scholar, ResearchGate, Springer, Arxiv, IEEE Xplore and more. ***Pro tip: When you search for papers on Google Scholar, make sure you also check the papers that have cited them.***

   If there aren't any helpful results, then widen your search boundary. Search for papers in the domain your problem is set in, as well as the papers that use similar types of datasets (basically, try to explore what kind of modelling has been done on similar datasets and whether it could be incorporated in your problem statement as well).

   Besides this, the references section in the papers you have found, could lead to discovering more meaningful and related resources which you otherwise would not have found. So make sure to go through the references section of a research article for more resources related to your project domain.

2. Reviewing the literature: Now that you've found the papers you want to look at, in more detail, it's time to study them! Don't worry, the first time we read a research paper, it went completely over our heads as well. However, instead of starting at the beginning and plowing your way to the end, we recommend a *Three Pass Approach*:
   a. First Pass: In the first pass, read the title, abstract, introduction, section headings and the conclusions. Oh, and glance through the references as well (this is really helpful as it provides more resources to refer to). That's

it. Just the section headings, not the content. This would ideally take about 10-20 minutes. After this pass, try to answer the 5 Cs:

   i. *Category:* What type of paper is this?
   ii. *Context:* What other papers is it related to?
   iii. *Correctness:* Do the assumptions appear to be valid?
   iv. *Contributions:* What are the paper's main contributions?
   v. *Clarity:* Is the paper well-written?

   This should help you decide whether to continue reading the paper or not.

b. Second Pass: This involves reading the paper in greater detail. However, you can ignore the formulae and proofs for now. This step is intended to help you grasp the concept of the paper. Make sure you look at the figures, tables and other illustrations in the paper thoroughly. Ideally, the second pass takes about an hour and by the end, you should be able to summarize the main thrust of the paper, along with evidence to support it.
c. Third Pass: This pass requires great attention to detail. Put yourself in the authors' shoes, and try to retrace their steps. You should be challenging every assumption and try to *virtually recreate* the paper by analysing how you would present a particular idea the paper describes. This pass also involves reviewing the areas of future work, and can take upto 4-5 hours to complete.

3. Summarizing: Yes, we know the second step sounds very taxing, which is why we recommend summarizing as and when you read the paper, instead of having to go through it all over again while writing your literature review reports. While you read the paper, be sure to note down the main ideas, assumptions, methodology and results. This forms the first part of your paper summary.

   Following that, try to analyze how this paper is relevant to your problem statement - it could be the usage of a similar dataset, models that can be used in your domain, etc. The final section of your summary should ideally involve a critique of the paper, i.e, missing assumptions, what could be done better, potential issues with the methodology, amongst others.

> Try to limit your summary to a couple of paragraphs at most. Remember, it is about the main ideas and assumptions of the paper and how it is relevant to your project, not the nitty-gritty proofs and details.

At this point, we need to stress on the fact that your literature review is not restricted to the papers you read. Do ensure you have made a note of all the other resources you’ve referred to - these can be articles on Medium (Towards Data Science, Heartbeat, anyone?), Analytics Vidhya, etc, or even Kaggle community notebooks. Any resource that helps streamline your solution for the problem statement counts towards your literature survey and should be cited in your reports.

Reference: Three Pass Approach: http://ccr.sigcomm.org/online/files/p83-keshavA.pdf

Authors’ Notes:

Adithi: When it came to my team, we didn’t know a lot about the domain so we first started off by reading domain specific articles and papers. Once that was out of the way, we started searching for literature about our dataset and problem statement on Google Scholar and ResearchGate. We went a step further and also looked up research papers for any models used in papers we found while searching for problem statement specific literature. As and when each of us read a paper, we made sure to summarize them so we wouldn’t have to spend time on this later on. We used Google Sheets to keep track of the papers, with the title, authors, year of publication, link and summary as column headers - this way, all of our literature review was restricted to just one document!

Akhil: My team went about dividing the task of literature survey among ourselves. We spent some time searching for relevant articles and had discussions before narrowing the number of research articles to a few most relevant papers. Each of us took up one or two research articles each and conducted an in depth study of those articles. In the end, each of us came up with summaries for the literature we had surveyed and organized them in separate documents. Each of us reviewed the other papers and summaries to ensure all was good. This way we were effectively able to complete the literature survey through effective collaboration.

Vibha: My team realised that before we started research on how analysis of cryptocurrencies had been done by others, we needed an in-depth knowledge of how cryptocurrencies work and what exactly blockchain is. Our research wasn’t restricted to articles, we found a bunch of youtube videos as well explaining how blockchain is used for cryptocurrencies. Once we were confident about the subject matter we would be analysing, we started searching for literature on ResearchGate and Google Scholar. We shortlisted around 6-7 papers solely based on the abstract and introductions. Each of us took the responsibility of summarising 2 papers each. We created a single Google Doc which had the name of the paper, link to the paper followed by the summary so we could all cross check each other's work and all our summaries were in a single document for easy access!

4. Programming Tips

If you thought we'd be discussing models or libraries you can use in this section, boy, are you mistaken. That's *your* job, guys! What this section will help with instead, is how you can effectively use tools like Google Colab and Kaggle Notebooks in order to collaborate while coding.

What you should've done by now is created a GitHub repository for your project (pro tip: triple check to make sure it's private) and added your teammates as collaborators. I mean, come on, doesn't it get tiresome to send snippets of code, or even entire files over email threads? That's exactly where Git, an open-source version control software, can help. In case you don't have any idea what GitHub is, here's a great playlist on YouTube to start off with: Git and GitHub for Poets.

If your project deals with submitting to a Kaggle contest, it would be easier to use Kaggle Notebooks while writing code and eases the collaboration process. Kaggle notebooks also support versioning, so you could use that to your advantage to conduct various experiments. Kaggle notebooks also have GPU Support. [***TIP: Your upcoming assignment will involve working with Kaggle Notebooks and this would be a good practice when it comes to the assignment.***]

If you're primarily working with Python and want to collaborate effectively while writing code, then you could use Google Colab. Google Colab works seamlessly when dealing with a Python Development Environment and makes the process of real time collaboration easier. Google Colab also has GPU Support in case you need to perform any computationally intensive task. Here's an article to get you started off with: Getting Started with Google Colab. Using R on Google Colab is not as straightforward as working with Python on Google Colab. Here's an article that shows you how you can use R on Google Colab: How to Use R in Google Colab.

It is important to document the code, so that you do not put up a blank face when you are going through it the next time. Besides this, it eases the evaluation process for the evaluator, who cannot spend a lot of time on a single submission. Make sure to use markdown in your notebooks and Github repos to make the documentation more presentable and easy to understand for whoever is going to be going through your project. Here's a basic guide to Markdown Syntax: Basic Syntax | Markdown Guide.

Whenever you are stuck somewhere, it is very likely that someone else would have faced the same issue as you earlier and the best way to find solutions to these is to refer to Stackoverflow. More often than not, you would find the solutions to your problems on Stackoverflow. Furthermore, do not hesitate posting your doubts on PESU Forums for clarification by your peers, TAs or the professor. It does not mean that you put up your code for others to use, as this will be a clear case of plagiarism but rather suggest tips and tricks to overcome the problem.

5. Writing a Literature Review Report

Before you know it, it's time for your first milestone - submitting the literature review report. By now, you've probably read a minimum of 6-7 papers about your dataset, potential solutions or the domain, along with a bunch of articles and tutorials (if you haven't done this yet - gear up for some all-nighters!). Now you're probably wondering, "*what exactly goes into the literature review report?*"

A quick look at the Expected Deliverables document will give you an idea about what is expected - your problem statement, review of literature, a potential solution that you hypothesize based on what you've read so far, and how your work is unique. The report needs to be submitted in the IEEE conference format. While you can use any word processor for this, we highly recommend using LaTeX (you can use it via Overleaf, an online LaTeX editor). It saves you the trouble of having to spend ages trying to align paragraphs with illustrations and tables on Word/Google Docs. Another approach could be to first get the content of your report ready by collaborating via Google Docs and once your content is finalized, you can transfer into using an IEEE template on LaTeX. This will help you collaborate easily while also getting the final alignments for your report right.

Now that you've figured out what tools you will use to write your reports, it's time to get started! But wait, there's also the dreaded page limit. How ever will you fit everything you've learnt about your problem statement in just 4-5 pages? Here's where you need to get selective. Make sure your introduction covers all the necessary aspects, but isn't too long. Answer the *what* and the *why* in this section, along with the context, but keep it brief.

Additionally, you don't need to provide detailed summaries of *every* paper you've read or resource used. Pick **3-4 papers** that you think are most relevant to your problem statement and summarize those under your "Review of Literature" section. As for the other resources, try to weave them into the other sections of your paper (a particular Medium article helped you how to approach your problem? Cite it under your

"Proposed Solution" section!). **Make sure you cite every resource you have mentioned in the paper and list them as "References".**

If you followed our steps during the Literature Review stage, you should already have summaries of these papers available. Proofread them, and then proofread them again (one can never be too thorough). Each summary should ideally consist of the synopsis of the paper (with assumptions, methodology and results), your critique of the paper, and how it is relevant to your problem statement. Make sure this doesn't go beyond a couple of paragraphs (4-5 atmost) per paper.

Moving on to the final part of the report, the proposed solution. This report is **not** your final report, so you need not have figured out all the kinks to your solution yet. All you need to write under this section is what you propose as a solution to your problem statement. State all the assumptions you've made while designing your solution (it's completely alright if these assumptions are naïve at the moment), what models you propose to use and how your proposed solution is different from prior work. In this section, you can cite work that has previously been done, for example, Kaggle notebooks that use the same data, articles that summarize/provide tutorials for a particular model you wish to use and contrast them with your approach.

Pro tip: If you have your exploratory data analysis ready, add them in your reports along with any insights gained, as and when needed, especially while describing your proposed solution approach. You could also have a separate section for EDA (bearing in mind the page limit, of course).

Any documentation might seem like a mundane task, but it is very important, not just for the evaluators but for you as well - especially to keep track of what you've accomplished. Our final tip? Don't wait for the last minute to write your literature review reports. Think of it as a report you are going to submit to a conference, ensuring that your report is of high standards. Finish your draft beforehand so that you have ample time to proofread multiple times and edit it accordingly. The faster you get this out of the way, the faster you can get started on the actual modelling and analysis!

Authors' Notes:

Adithi: Here's the link to my team's literature review report, for your reference! My team and I didn't have to spend time going through papers all over again because we already had our summaries in place. We picked the 4 papers we found to be most relevant to our problem statement and added their summaries. In addition to this, we also added a Kaggle notebook which helped us streamline our proposed solution under References. Although our EDA involved multiple graphs, we only added the ones that were most pertinent to our problem statement, making sure that we hit all the points necessary without crossing the page limit.

Akhil: Here's the link to my team's literature review report, for your reference. Each of us wrote summaries for a few papers and then we picked the three most relevant papers of the lot. We then combined all the summaries of all relevant literature into the report. We explained the various assumptions and the main takeaways from each paper. Besides this, we also included as to how we went about with our exploratory data analysis. Various relevant graphs were included along with the takeaways from each graph and what it tells us about the data at hand.

Vibha: Here's the link to my team's literature review report, for your reference! Since my team had all our paper summaries in a single doc, we made use of it to fill in our initial literature report. We summarised around 7 papers but ended up including around 5-6 of them in the report. We included the methodologies, various assumptions and the key takeaways from each paper. Apart from this, we included an extensive report on the cryptocurrency industry, the what and how of blockchain and how it comes into play with respect to cryptocurrency.

6. Summarizing Insights

Visualizing the data you have at hand and presenting it in the right format, will go a long way in giving the right impression to the reader and shows that you have really dug deep to get to the results that you have obtained. Use your graphs, from the EDA and post-modelling stage to weave a story around your project. However, this should not be confused with putting up graphs in your report just for the sake of it or for reaching the minimum page limit (***Tip: if you do have extra graphs, you can add a few in the Appendix with appropriate explanations***). Make sure that each graph is relevant to the conclusion you are building up to. Also make sure that the data is represented clearly in the graph and one does not have to look at it while scratching their heads to actually make out what you are trying to convey.

Moving on, try to limit the complexity of the graphs to a certain threshold so that the reader is not flooded with a plethora of information in one figure. Although your intention might be to present all information in one place as concisely as possible, this strategy could backfire and end up looking clumsy, which you certainly do not want. So it is important to decide how much information is not too much in a single graph and how much can be interpreted easily when one is reading it. And yes, everyone loves a colorful and vibrant graph, but make sure you use color palettes in a way that isn't painful to the eye!

Another important point to keep in mind is to avoid redundant graphs. For example, if you have already represented some data through a bar chart, there is no necessity to add another pie chart to represent the same data. If multiple representations are viable, you need to decide which is the best to use so that it is easy to comprehend. Do not add redundant graphs to fill up space in your report. Before putting up any graph, think and rethink whether it is absolutely necessary or is it made obvious by some other representation already included in the report. So, choose your visualizations wisely to create a well rounded report.

Well, we don't know what else we can add here about plotting good graphs that hasn't already been covered in your Statistics for Data Science course and Unit 1 in Data Analytics, so let's move on!

So, you've tried multiple models and nothing seems to be working in your favour. All performance metrics are suboptimal and predictions are haywire. The worst, isn't it? It's okay, it happens to all of us. But stop fretting, because all's not lost just yet! If something isn't working, figure out why. Pinpoint areas where your models might be going wrong. Maybe one of your assumptions was flawed? Maybe the dataset was unreliable? It's alright. You know what didn't work, so note that down and think of possible fixes. If any of these can be implemented, try them!

You do not need to meet any kind of threshold with respect to performance metrics to call your project a success. If the best accuracy you're getting is 50%, scrutinize, try to find out what the issue is along with the respective fixes, and hash out all the details in your report. **This being said, we strongly recommend that you try all possible solutions to improve performance metrics before raising the white flag.** Try multiple models, check your preprocessing methodology and the dataset as well. If none of this works, well, now you know what not to do, and those are valuable insights too!

7. Writing the Final Report

So you've run multiple models on your dataset, obtained performance metrics, made predictions and gained insights. Time to get documenting! The final report serves as the culmination of all the work you've done over the semester. The report should explain the end to end structure of your project. Ideally the report should be 4 to 5 pages in length in the 2 column IEEE conference format. The report consists of the context and introduction to your problem statement, literature survey, assumptions and scope, proposed solution and methodology followed, experimental results, insights, conclusions and contributions. A quick look at the Expected Deliverables document will give you an idea as to how long each section is expected to be.

The introduction and context remains the same as your literature review report, unless there have been modifications to your problem statement since then. You could further summarize the content of your literature survey report to include it in the section covering previous work and its limitations. Ideally about two paragraphs for each literature surveyed would be sufficient. For each paper, you can explain how it is relevant to the work you are doing, what you can improve upon and finally what are the limitations of the solution proposed in the paper you reviewed.

While explaining your proposed solution, make sure to include the steps you undertook while preprocessing the data, how you went about building your models and you also need to talk about the performance of your models by explaining the evaluation steps and the results of the evaluation. You can use diagrams to enhance the understanding for the readers.

Moving on to the experimental results, in this section you need to explain all the insights you have obtained and dive deeper into the performance of the models. To be more specific, you need to report under what conditions do you see the model performing well and where does it not produce optimal results. This shows that you have thoroughly understood the work you have performed and the models you have built.

The next section should include whatever insights you've gained from the dataset and problem statement, and how these compare with your assumptions made before designing your solution. Anything you've hypothesized to explain why you've obtained unexpected results can also be included in this section. Just because your performance metrics weren't great does not mean your project failed!

The penultimate part of your report would include your conclusions and future scope of your project. Summarize whatever you've learnt from this project and identify areas of improvement. Also be sure to note down real-world applications of your problem statement.

The final part of your report would include the acknowledgments, references and the appendix. **Be very sure to cite all resources you've referenced in your report under References.** If you're using LaTeX, you can use BibTex to cite your papers and articles. Most websites like ResearchGate and Google Scholar also provide citations for papers in both text and BibTex format, so you'll just need to copy those to your reports. Put any additional information you want to convey inside your Appendix (we recommend putting your individual contributions as the first subsection in your Appendix so it doesn't interfere with the report's content). Have extra graphs you didn't know where to add? To the Appendix it is! But do add appropriate explanations for every graph you add - it's very confusing to look at a series of graphs and have no information about why we're looking at it.

Now that you've figured out what sections to have and what content goes in them, let's get started! You can either use the Word/Docs or the LaTeX IEEE conference templates. Once again, we recommend using the LaTeX template with Overleaf (you can also use it locally on your personal systems). LaTeX is a document processing software that is extensively used in academia for publication of scientific documents.

Overleaf is an online LaTeX editor that allows two people to collaborate on a LaTeX project. Due to this restriction on the number of people who can collaborate, we advocate writing down your content in separate documents according to the sections you plan on having in the report. These can then be copied to your LaTeX project with suitable modifications made (for example, with equations and text formatting). To get started, go through the Overleaf Documentation. However, there's no better way to learn than to actually work on your project. In that regard, here are some tips and tricks to navigating LaTeX.

Your final report is an extremely important document as it describes your entire project in detail, so make sure you keep enough time at the end to draft this document. We can guarantee that it will take multiple rounds of editing to perfect it. You're almost done, so don't lose steam just yet!

Authors' Notes:

Adithi: Here's the link to my team's final report! As you will soon find out if you do choose to glance through it, our performance metrics were well, how do I put this lightly, not great. My teammates and I tried every possible fix we could think of, and although our accuracies weren't increasing by a lot, our anxiety definitely was! We then decided to cut our losses, and use the model that gave us the best performance metrics overall to make predictions (some of which were also haywire). Everytime we saw our model predict something unusual, we made sure to theorize why it might've happened. So even though our accuracy might not have been the best, it did give us a lot to scrutinize and pick apart to see what might've gone wrong, and that's exactly what we illustrated in our report.

Akhil: Here's the link to my team's final report! We summarized the literature review further and included the exploratory data analysis as well. We report as to how we divided the dataset into training, validation and testing sets. We tested out various models and reported the performance metrics for each of the models. We also discuss under what situations each of the models performs well and where the performance is not satisfactory. Finally, we conclude as to which model is most suitable for the task at hand and provide supporting evidence. Keep in mind that you may not always arrive at the best performing model in one try. You will need multiple iterations and experimentations to find the best model.

Vibha: Here's the link to my team's final report! We refined our literature review and included the data cleaning and exploratory data analysis done as well. We described the various forecasting techniques used and reported the performance and metrics of each technique. Similar to Adithi's teams situation, our initial metrics weren't reaching the bar (meep :/) but it gave us an opportunity to explore why that was the case and why certain forecasting techniques performed better than others. We also made sure to include all the graphs plotted and how we arrived at certain model parameters. A key takeaway from all our experiences is that, don't be disheartened if your initial model doesn't give you the results you expected. Try to reason out why that was the case and how other models or techniques might be a better fit, given your data!

8. Making the Video

This is the final step towards the "Ultimate Finished Product" that you are expected to deliver (you're almost at the finish line!). The video you're going to present has to include information about all the tasks you had done during the entire course of the project. The first and most critical rule to keep in mind is to make sure that the video does not exceed the six minutes mark. You need not start with this hard set rule at the beginning itself, as later one you can edit the video by trimming the unnecessary bits to make the length to a maximum of six minutes. However keeping this rule in mind while preparing the video will help you later on to trim the unnecessary bits. For instance, when you are recording you can tell yourself which parts are absolutely necessary and which parts can be done with omission in case the time limit is exceeded.

These are the official guidelines of what has to be included in the video:

[1 minute]	What problem have you selected and what data set are you using to solve the problem?
[1 minute]	Why is what you have done important/ useful?
[1 minute]	What is the approach you have taken?
[1 minute]	How did you evaluate your solution/ the algorithm you implemented?
[1 minute]	Anything interesting that you inferred about the data or learnt through the process? What is the role of each team member?
[1 minute]	Buffer time

We recommend that every member of the team is present in the video (it *is* a team project after all). There could be different strategies to go about showcasing yourselves in the video. For example, each of you could record separate clips and stitch them together or all could join a common meeting and record while speaking in turns.

Now you need to decide how to split the content amongst yourselves. We recommend that every member of your team speak about what their individual contribution is and how it contributes to the big picture. In an ideal situation, where every team member contributes equally, you can divide the time in the same way as well. Make sure you spend ample time on what methodology you followed and the insights you gained.

Writing a script for your video is extremely beneficial. It stops you from veering off track and also functions as a checklist of everything you need to cover. You can either use your final report as your script, or draft a new one:

1. Using your final report: You do not need to cover the entire report in your videos. Figure out what sections are the most important and use only those in your video. For example, although the literature review is a very important part of your report, it need not be covered in your video, so that you have more time to talk about your work.
2. Drafting a new script: Writing a full-fledged script takes time, so make sure you take that into account if you want to go ahead with this alternative. However, it's not compulsory to have a complete script drafted before you can shoot the video - even noting down key points you need to cover and some keywords will be of great help!

Tip: When you're recording the video, try not to read directly from the script. Everyone's reading voice is different from their speaking voice, so make sure you sound conversational, not robotic!

Before you start recording, there is one last thing to decide - to make or not to make a PPT? If you feel that the visualization plays a huge part in your final results, then using a PPT while making the final video would make complete sense. Once again, there is no hard and fast rule as to whether a PPT is compulsory but you can use it to aid while making the video.

You need to make sure that the video has to be a maximum of six minutes in length. This may require some editing after recording. One useful strategy would be to record separate clips for each topic that has to be included and stitch them together to form the final result. This will also help in work division while recording the video and in case of any mistakes, only the respective portions can be rerecorded and added back into the final video. This way, smaller portions can be easily stitched together instead of recording the whole six minute video in one go. You need not go overboard while editing, just make sure that all the required content is covered in your video.

If you've made it till here, then there's just one last thing left to do. Make sure all the required deliverables are in place - your final report, the video, GitHub repository (make sure it is private but add the evaluators as collaborators), the plagiarism check for your report, links to the (subset of) dataset used, and any other links that need to be submitted. As we've repeated multiple times throughout this guide, check all of them once, and then check them again. Once you're good to go, hit that submit button on Forms and breathe that sigh of relief! Time to do a victory lap, because you've finally finished the project!

Well, this brings us to the end of the road. If you've stuck with us so far, we hope all our tidbits of information and tips helped you ace the DA course project, and you had fun doing it! Before we let you go, here's our final tip: ***start looking at conferences where you can possibly publish this as a research paper - that would be a great addition to your resumé!***

Signing off,
Adithi, Akhil and Vibha

****