Business Data Analytics project --- quickstart

Formalities

For details, refer to the BDA project Handbook on Moodle.

Text & code: Report & GitHub repo

Text - report as PDF

Your submission will need to consist of uploading **one single PDF** file. This file is a **written report of your work** and should contain a presentation of your work, including all your textual explanations, as well as graphics and tables (generated by your Python code or possibly in a BI system).

Code - GitHub repo

You need to **maintain a GitHub repository containing all code** you used in the project. See BDA Handbook, p. 6:

> Include a link to your GitHub repository in your report. The repository must contain:
>
> - Clean and well-documented Python code.
> - Supporting documentation, such as a README.MD file, that explains the repository's structure, how to reproduce the results, and dependencies, etc.

(If, for any reason, your company uses an alternative version control system (VCS) or platform to git & GitHub, this may be allowed if approved by your supervisors and the university. Please check with them beforehand.)

Using Confidential Data in Your Project

When working with data that cannot be made public due to company secrets or privacy concerns:

- The data **does not need to be shared** at any point, whether on GitHub or any other platform.
- Your **GitHub repository** must contain your code, but it can be private. However, you must share access with the graders. The written PDF report must still be uploaded to Moodle.
- If required by the company, the graders can sign a **Non-Disclosure Agreement** (NDA) to ensure confidentiality.

Crucial formal aspects of the submitted report

- Pay attention to **length requirement**: 10,000 words ± 15%.
  - (More precisely, upwards deviation can be up to +25%.)
- The PDF must start with a **title page and a signed (!) declaration of originality**.
  - See BDA Handbook, *Appendix 2 – Sample Title Page* and *Appendix 3 – Sample Declaration of Originality* (p. 22-23).
- You should also have a include **Executive Summary** (500 words) of your project --- written *after* you complete the project but placed at the very beginning, in the preliminary material.
- ALWAYS **cite other works you use** (not just academic papers; URLs, too, e.g., when using code snippets from stackoverflow; also, importantly cite all LLMs/AI tools you use!).
  - Do "**in-line citation**" where you use the material
  - **+ collect ALL references at the end** of the thesis under a dedicated "References" section!

Citation & referencing

It's not a problem if you build on others' work --- in fact, this is *expected* of you. Do a initial search and review of the literature and work on the topic of your project and use it. It's also good practice to present prior relevant work in a **a review of literature and state-of-the-art (SOTA)** section.

Some considerations about using others' work and referencing:

- Levels of using work other than your own:

    - For **paraphrasing** or when only using ideas, a simple reference is fine.
    - For **short word-for-word citations** from some source, use quotation marks (plus the reference of course).
    - For **long word-for-word citations** from some source, use indentation and/or italics and/or smaller font, etc (plus the reference of course). Warning: *do NOT use more than a paragraph or two word-for-word from other sources!!! Paraphrase!*
- If possible, **be consistent in your referencing style**, and if possible, try to follow the referencing system required by the university (Harvard style).
- It is vital that you reference **any and all material (text, code, solution, idea) that you adopt or adapt from an outside source**. This can be anything from using code snippets from stackoverflow, github, etc. through personal communications from teammates up to using generative AI solutions like ChatGPT or Copilot. It is important to emphasize that there is **nothing wrong** with using any of these sources: in fact, quite the contrary, **it speaks a lot to your efficiency if you can use outside sources to make your workflow better or faster**! (Cautionary note though: *only use material that you understand*, otherwise there is a danger of you making grave mistakes in the adoption thereof.) But you absolutely **HAVE TO be explicit about your sources**: on the one hand, it's basic good academic conduct, and on the other hand, it's often easier to understand, possibly even debug code and solutions if you can follow them back to the source. There are various conventions for citing different sources.
    - For **generative AI solutions** like ChatGPT, if you wanto to go into some best practice citation suggestions you can look at, e.g., https://apastyle.apa.org/blog/how-to-cite-chatgpt --- But the *bare minimum* is to specify in-line something like "this code was adapted from code generated by ChatGPT", and you would include, e.g., https://chat.openai.com/chat etc. under your References section.
    - For **online sources**, again there are good citation practices (e.g., specifying not just the URL, but also the date on which you accessed it), but the *bare minimum* is to cite the URL as your source inline and under your final References section.
    - For **personal communications**, you specify the name of the person and "personal communication" or "p.c.", e.g., "(John Smith, p.c.)". You don't have to list personal communications under the final References, but you do have to specify them in the text.

Permitted level of AI use

See BDA Handbook (p. 28.):

> For the text of the Business Data Analytics Project
>
> - Level 2: AI-Assisted Idea-Generation and Structuring
>
> For generating the codes:
>
> - Level 4: AI-supported Task Completion

Submission

- The submission link "Business Data Analytics Project submission" will ONLY be visible to you **once you complete the "Business Data Analytics Project title and final exam"** quiz. As the final exam (see BDA Handbook p. 19, 4.7), you must answer the 3 reflective questions in at least 300 words.
    - Pay attention to the **deadline** and do not leave submission to the last minute!
    - Bear in mind that there is a **minimum word limit** of 300 words to these questions. As long as your answer fails to reach this word limit, your final exam won't be marked as completed, and the BDA project submission link will not appear!
- Submission of project itself under **"Business Data Analytics Project submission"** --- same deadline as the final exam, but submission link will only appear on Moodle on completion of the final exam.
    - **Submit a single PDF, containing your written report**, and in an appendix, a link to the GitHub repository containing your code.

IMPORTANT: Pay attention to the **deadline** (10:00 A.M. on the last day of the teaching period)! There is no possibility of accepting late submissions or submissions outside of the Moodle assignments! In case of a problem, turn to **Requests / Student admin** (see "Contact us" page of IBS), but you need to have very good grounds to deviate from any regular submission procedures. Technical issues, last-minute problems, or forgetting the deadline will not be considered valid reasons for exceptions. Make sure to plan ahead and submit your work well before the deadline to avoid any complications.

Assessment criteria

You have to **reach a pass mark (50%) on ALL criteria** to get an overall pass mark.

Also, "Business Data Analytics Projects **below 8,000 words will not be marked** and receive a zero grade."

> Check out the detailed *marking grid* in *Appendix 5 – Business Data Analytics Project Assessment* (p. 25) of the BDA Handbook on Moodle.

- **Problem Definition and Analytical Approach**. Clarity and relevance of the problem statement; depth and quality of the analytical approach used to explore and define the problem. [15%]
- **Understanding and Integration of Business and Analytics Concepts.** Depth of understanding of business concepts and analytics techniques; successful integration of both domains to address complex challenges, including actionable insights. [25%]
- **Data Collection, Preprocessing, and Exploratory Data Analysis.** Quality of data sources identified; effectiveness and rigor in data cleaning, preprocessing, and exploratory data analysis (EDA). [15%]
- **Implementation and Optimization of Algorithms and Models.** Complexity and appropriateness of coding techniques used; use of Python to implement and optimize suitable algorithms and models to address the problem. [20%]
- **Model Validation and Evaluation.** Effort and success in validating and evaluating models; evidence of accuracy, robustness, and appropriateness for the problem. [15%]
- **Interpretation, Communication, and GitHub Usage.** Clarity in presenting findings and insights to different audiences; effective use of visualizations; effective use of GitHub to share code and documentation (60%+). [10%]

Assessment procedure:

- **2 readers** do a detailed evaluation based on the criteria.
- Checked by an internal and an external **moderator**.
- Approved by **Exam Board** (for exact date, check IBS's academic calendar).
    - → this is the point after which you get the results and the feedback.

Structure of the written report

As far as the **presentation and structure of the written report** of your project is concerned, refer to **page 8 of the BDA Handbook**. The structure suggested there (including word counts) serves as a recommendation, but every project is unique, and can be presented differently. Your project might have different goals or a different focus. (E.g., some projects are more data analysis oriented, while some involve more (data or ML) engineering etc.)

Main parts of the written report

Preliminary material

Must-haves:

- **Title page** (1 page)
- **Declaration** (1 page)
- **Executive summary** (~500words)
- **Table of contents**

Main body

- **Introduction**: Written *before* you begin working on the project. This should present the project, define its goals, outline expectations, and provide context.
- **Core Sections of the Report**: Structured logically into sections such as Literature Review, Methodology, Exploratory Data Analysis (EDA), Modeling, Evaluation, Results, Business Insights, etc.
- **Conclusion**: summary of key findings + limitations and further work.
    - Be honest — your goal is *not* to exaggerate how great the results are. The conclusion should reflect the actual outcomes of your project, even if they are disappointing.

Post-main body

- **References**
    - A complete list of all sources used in the project. Ensure that citations are included in-line within the main body as well.
- **Appendices**
    - Required: A link to your GitHub repository.

Structuring your project work

The following are some typical stages and ML-related considerations for your project when you are actually working on it.

Set up clear goals and expectations

Before starting on the project, set up clear goals and expectations. The project proposal is an excellent start, but you'll want to precisify the subtasks, the input and target features for modelling tasks, etc. E.g., starting from "I want to know the effect of temperature on electricity prices", you should break this down into:

- What EDA can be appriate? --- E.g., correlations, scatter/line plots...
- What ML models will I need?
    - Input: temperature and other independent features. Target: electricity price.
    - Time series (forecasting) or non-sequential data?
    - How to explore effect of temperature? E.g., model-internal approaches (coefficients, feature importances), surrogate models (LIME, SHAP), etc.

Exploratory data analysis

Taking stock of independent and target variables

- Description, initial reasoning
- Data types (e.g., datetime)
- Sample/time step/pixel count, data dimensionality etc.
- Descriptive statistics

Initial visualisations

- Data- and variable-appropriate plots
- Domain-specific explorations
- Eg. time series: trend, periodicity etc.
- Explore relationship between variables
- Pairplots
- With target: (Lagged) correlations, autocorrelation

Data preparation

Missing values

- Exploration, reasoning about handling them etc.
- Don't just blindly apply a method to all features --- you might choose alternative solutions for different datasets and different features! (E.g., filling with zero might be appropriate for a few features, filling with mean or median or mode for some features, forward-filling or multivariate NA-imputation for some others, dropping in some cases etc.)

Anomaly detection and solutions for outlier handling

- Detection:
    - Non-ML: Boxplots, z-values, IQR-based etc.
    - Or some anomaly detection
    - But choose method appropriate for data (special care with time series, for example!)
- Handling:
    - E.g., clipping, etc.
    - Motivate both need or lack of need for handling (and consider during evaluation).

Feature engineering

Feature encoding

- e.g., cyclic encoding of temporal features, one-hot-encoding of categorical features etc.

Feature selection, possible derived features, possible additional (e.g., time-related) features

Reason about dropping a variable!

- categorical… -> too many unique values, e.g. 1000 samples, 800 unique values,
    - and no simple way to reduce category count.

- Intuitively not important, e.g. ID tag.
- Too many NA values, e.g., 800 NA out of 1000.
- train models with and without and see which model performs best
- explore the feature importances in different models
- Linear reg.: coefficients (after scaling!)
- RF: feature importances
- L1 regularization
- only use some for now, rest: “future work”

Refer to EDA, e.g., correlations – but be aware of limitation (e.g., linear)!

- High absolute correlation
    - Useful for predicting target?
    - Not useful, because information already present in other variable? -> reducing * multicollinearity, dimensionality reduction…
- Forecasting task -> *Lagged* correlations relevant.

Potential data normalisation

- Standard/MinMax scaling, reasoning about for which model it’s needed etc. (Generally, except for some rare data and cases, linear scaling won't hurt any model, but might be crucial for many models.)
- Good practice to include as part of a pipeline.

Potential dimensionality reduction

- E.g., PCA or LDA for modelling or visualisation, or UMAP for visualisation?

Potential resampling

Domain-specific data prep

- E.g., detrending, deseasonalising, rolling aggregation for smoothing if needed.

Preparation of input and target data for modelling

- Maybe use a pipeline.
- Pay attention to shape of data needed for different models.
- Reason about whether normalisation is vital for a model (e.g., neighbourhood/distance/density sensitive approaches, neural networks).
- Train-valid-test split (or prepare for cross-validation). Shuffle true or false? --> Has to be the same for all models.

Modelling, predictions

Deciding on goodness of fit criteria

- *Task-appropriate* metrics and visualisation to be applied in the evaluation of *all* predictions. --- Decide in advance and apply consistently for all models!
- Don’t forget to *inverse transform* target when evaluating, if any prior transformations done to the target (detrending, scaling etc.).
- Classification metrics vs. regression metrics; description of metrics used.
- Visualisation, inspection of *true and predicted values* to see and reasoning about where issues lie. (Do qualitative evaluation, don't just use a bunch of metrics.)
- *Train vs. out-of-sample* performance comparison for each model. --- Best practice (if enough data): train-validation-test split, or Cross-Validation + test.
    - Pay attention to disabling shuffling for sequential data!

Non-ML baselines

- Non-machine-learning baselines, e.g., average, rolling average, last, etc.
- Description of the baseline and evaluation.

Baseline ML models, e.g., linear/logistic regression

- Description of the relevant model type and its values and limitations.
- (Hyperparameter tuning / motivation of hyperparam choices if applicable.)
- Evaluation.

More advanced ML models, generic and domain-specific

- Description of the relevant model type and its values and limitations.
- Hyperparameter tuning, motivation of hyperparam choices if applicable.
- Evaluation.

Evaluation, reflection

- As mentioned, use multiple evaluation metrics, as well as qualitative (typically visual) evaluation.
- Reasoning about and reflection on the results.
- Compare to your goals and expectations.

Remember: the ML process is circular, so you might want to go back to earlier points (e.g., even EDA and data preparation --- e.g., outliers or missing value handling) on observing some issues in modelling.

> IMPORTANT: do NOT do this repetitive process on the basis of performance on the test set, which should be set aside for an evaluation of your final chosen model/pipeline! Use separate validation data or CV if computationally feasible.