10 Academy Batch 4

Weekly Challenge: Week 2

A/B Hypothesis Testing: Ad campaign performance

Business objective

An advertising company is running an online ad for a client with the intention of increasing brand awareness. The advertiser company earns money by charging the client based on user engagements with the ad it designed and serves via different platforms. To increase its market competitiveness, the advertising company provides a further service that quantifies the increase in brand awareness as a result of the ads it shows to online users. The main objective of this project is to test if the ads that the advertising company runs resulted in a significant lift in brand awareness.

Project Overview

SmartAd is a mobile first advertiser agency. It designs intuitive touch-enabled advertising. It provides brands with an automated advertising experience via machine learning and creative excellence. Their company is based on the principle of voluntary participation which is proven to increase brand engagement and memorability 10 x more than static alternatives.

SmartAd provides an additional service called Brand Impact Optimiser (BIO), a lightweight questionnaire, served with every campaign to determine the impact of the creative, the ad they design, on various upper funnel metrics, including memorability and brand sentiment.

As a Machine learning engineer in SmartAd, one of your tasks is to design a reliable hypothesis testing algorithm for the BIO service and to determine whether a recent advertising campaign resulted in a significant lift in brand awareness.

Why this project?

Hypothesis testing is the cornerstone of evidence based decision making. The A/B testing framework is the most used statistical framework for making gradual but important changes in every aspect of today's business. Please read A Refresher on A/B Testing to get a rich business and historical context.

Data

The BIO data for this project is a "Yes" and "No" response of online users to the following question

> ***Q: Do you know the brand Lux?***
> - O Yes
> - O No

This is a test run and the main objective is to validate the hypothesis algorithm you built. SmartAd ran this campaign from 3-10 July 2020. The users that were presented with the questionnaire above were chosen according to the following rule:

> ***Control***: users who have been shown a dummy ad
> ***Exposed:*** users who have been shown a creative (ad) that was designed by SmartAd for the client.

The data is available for download here.

The data collected for this challenge has the following columns

- **auction_id:** the unique id of the online user who has been presented the BIO. In standard terminologies this is called an impression id. The user may see the BIO questionnaire but choose not to respond. In that case both the **yes** and **no** columns are zero.
- **experiment**: which group the user belongs to - control or exposed.
- **date**: the date in YYYY-MM-DD format
- **hour**: the hour of the day in HH format.
- **device_make**: the name of the type of device the user has e.g. Samsung
- **platform_os:** the id of the OS the user has.
- **browser**: the name of the browser the user uses to see the BIO questionnaire.

- **yes**: 1 if the user chooses the "Yes" radio button for the BIO questionnaire.
- **no**: 1 if the user chooses the "No" radio button for the BIO questionnaire.

Learning Outcomes

Skills:

- Statistical Modelling
- Using core data science python libraries pandas, matplotlib, seaborn, scikit-learn
- ML algorithms Linear regression, Decision Trees, XGBoost
- Model management (building ML catalog contains model feature labels and training model version)
- MLOps with DVC, CML, and MLFlow

Knowledge:

- Data exploration
- Hypothesis testing
- Machine learning
- Hyperparameter tuning
- Model comparison & selection
- Experiment analysis

Communication:

- Reporting on statistically complex issues

Competency Mapping

The tasks you will carry out in this week's challenge will contribute differently to the 17 competencies 10 Academy identified as essential for job preparedness in the field of data science, and Machine Learning engineering. The mapping below shows the change (lift) one can obtain through delivering the highest performance in these tasks.

MC0: Marginal contribution - causes no significant change
MC1: Minor contribution - recognised for routine performance gain
MC2: Measurable contribution - will contribute a value towards portfolio and job readiness metric
MC3: Major contribution - best performance of these types of tasks at least three times within our training leads one to attain job ready level along that competency dimension.

Competency	Contribution	Potential contributions from this week

Business Understanding	MC2	Understanding and reasoning the business context. Thinking about suitable analysis that matches the business need. Thinking about clients and their interests.
Data Engineering	MC1	Thinking about how to store data for easy analysis, and what format to use to build responsive dashboards.
Data Understanding	MC1	Understanding the data provided and extract insight. Exploring different techniques, algorithms, statistical distributions, sampling, and visualisation techniques to gain insight.
Dashboard & Visualization	MC1	Building a dashboard to explore data as well as to communicate insight. Advanced use of modules such as plotly, seaborn, matplotlib etc. to build descriptive visualizations. Reading through the modules documentation to expand your skill set.
Mathematics and Statistics	MC3	Thinking about statistical distributions, sampling, bias, overfitting, correlations.
MLOps & Continuous Delivery	MC3	Using Github for code development, thinking about feature store, planning analysis pipeline, using MLOps tools for code, data, model, artifact versioning, setting up docker containers for automated microservice deployment.
Modelling and evaluation	MC3	Comparing multiple ML algorithms; training and validating ML models; choosing appropriate architecture, loss function, and regularisers; hyperparameter tuning; choosing suitable evaluation metrics.
Python programming	MC3	Advanced object oriented python programming. Python package building.
SQL programming	MC1	Building feature stores using SQL or NOSQL databases.

Fluency in the Scientific Method	MC3	Thinking about evidence. Generating hypothesis, testing hypothesis. Thinking about different types of errors.
Ethics	MC2	data privacy, data security, ethical use of data. The 8 principles of responsible machine learning
Statistical & Critical Thinking	MC3	Thinking about the difference between causal vs chance correlation. Giving reasonable recommendations. Thinking about uncertainties.
Software Engineering & Dev Environment	MC2	Reading articles on software project planning. Unit testing.
Impact & Lifelong learning	MC2	Learning new concepts, ideas, and skills fast, and applying them to the problem at hand.
Professional Culture & Communication	MC2	Writing a well formatted presentation with no mistakes, formatted nicely.
Social Intelligence & Mentorship	MC2	Asking for help early, providing help for those who need it, avoiding being stuck.
Career Thinking	MC2	Working within groups in a successful way

Team

Instructors: Yabebal, Abubakar, Mahlet, Kevin, Paulcy

Key Dates

- Discussion on the case - 11:30 UTC time on Monday 19 July 2021. Use #all-week2 to pre-ask questions.
- Interim Solution - 8:00PM UTC time on Wednesday 21 July 2021.
- Final Submission - 8:00PM UTC time on Saturday 24 July 2021

Leaderboard for the week

There are 100 points available for the week.

20 points - community growth and peer support. This includes supporting other learners by answering questions (Rocket chat), asking good questions (Rocket chat), participating (not only attending) daily standups (GMeet) and sharing links and other learning resources with other learners.

25 points - presentation and reporting.

10 points - interim submission. This is measured through:

- Answering Task 1.1 correctly and with enough detail (4)
- Presenting statistically valid interpretation of the result from Task 1.2 (4)
- Presenting clear understanding of the difference between classical, sequential, and machine learning based A/B or significance testing (2)

15 points for the final submission. This is measured through:

- Evidence to publication or submission of report in a blog e.g. medium, linkedin or other similar platforms (4 points)
- Clarity of message, writing structure e.g. logically separated sections, and appropriate usage of graphs (5 points)
- Professionalism/production value (free of spelling errors, use of same font, well produced) (3 points)
- Balance between being 'full of information' and 'easy to understand' (3 points)

55 points - Technical content

10 points - Interim submission

- Github link submission (10)
  - Object oriented (5)
  - Frequent commit (2)
  - Professional plot production code (readable axes labels, title, and legend; good choice of color) (3)

45 points - Final submission

- MLOps setup (20)
  - Screenshot of DVC data versions (5)
  - Screenshot of MLFlow dashboard (5)
  - Working CML git workflow implementation in repo (5)
  - Screenshot of example CML report when git push a code (5)
- Github Link submission (25)
  - Implementing all required ML models (8 points)
  - Screenshot of MLFlow model versions from hyperparameter tuning (8)
  - Object oriented coding (5)
  - Frequent git commits (4)

Badges

Each week, one user will be awarded one of the badges below for the best performance in the category below.

In addition to being the badge holder for that badge, each badge winner will get +20 points to the overall score.

Visualization - quality of visualizations, understandability, skimmability, choice of visualization
Quality of code - reliability, maintainability, efficiency, commenting - in future this will be CICD/CML
Innovative approach to analysis -using latest algorithms, adding in research paper content and other innovative approaches
Writing and presentation - clarity of written outputs, clarity of slides, overall production value
Most supportive in the community - helping others, adding links, tutoring those struggling

The goal of this approach is to support and reward expertise in different parts of the Machine learning engineering toolbox.

Group Work Policy

Everyone has to submit all their work individually.

This week, however, you are expected to complete the project with your assigned group. In the table below, your name is assigned to one of the groups we formed. You are expected to collaborate very closely with your team and finish all tasks.

Within your group, you can share concepts, references, codes, figures, and have similar answers to the questions. You **MUST** write both the interim and final reports yourself, submit Github links from your Github account, and take screenshots from your computer. All group members may have similar code in their git repository, but you should have at least run the code in your local machine and make frequent commits. Remember to make git branches as necessary, do pull requests, and other good software development practises.

We expect all group members to contribute equally. We leave the assignment of roles within groups to the group members.

Group	Group Members
1	BEZAWITALEM YIMER Haftom Tekleweyni Natnael Sisay Khairat Ayinde
2	Ethani Caphace Desmond Onam Yosef Alemneh Muhammed Kutashi Robert Ssebudandi
3	Maelaf Estiphanos Binyam Sisay Daniel Zalalem Azaria Tamrat Michael Darko Ahwireng
4	smegnsh demelash Amon Kimutai Eliphaz Niyodusenga Solo Ache same michael
5	Elias Andualem Bethelhem Sisay Natnael Teshome sibitenda harriet Eyerusalem Gebreegzabiher
6	Chimdessa Tesfaye Hordofa Eyosias Desta Luel Hagos Abel Mitik jakindah Oluch
7	christian ZANOU Fumbani Banda Michael Tekle Dorothy Cheruiyot Toyin Hawau Olamide
8	Steshy Kibika Germain Rukundo Behigu Gizachew Blaise Papa Deborah Haile
9	Bereket Kibru UWASE Rachel Mizan Abaynew FAITH ASHIONO Nebiyu Samuel

10	Abreham Gessesse Sabaagizew Woldeamanuel Vincent Wayuga David Mukuzi Kate Njoki
11	Mubarak Sani Milky Bekele Yosef Engdawork Elizabeth Nanjala
12	Kora Marzouck Lafia Jocelyne Mukamisha Boris Papineau Hirwa Euel Fantaye Zelalem Getahun

Late Policy

Our goal is to prepare successful trainees for the work and submitting late, when given enough notice, shouldn't be necessary.

For interim submissions, those submitted 1-6 hours late will receive a maximum of 50% of the total possible grade. Those submitted >6 hours late may receive feedback, but will not receive a grade.

For final submissions, those submitted 1-24 hours late, will receive a maximum of 50% of the total possible grade. Those submitted >24 hours late may receive feedback, but will not receive a grade.

When calculating the leaderboard score:

- From week 4 onwards, your lowest week's score will not be considered.
- From week 8 onwards, your two lowest weeks' scores will not be considered.

Instruction:

Objectives:

The analysis objective of this project are divided into 4 sub-objectives that overall guides the workflow

- Setting up A/B testing framework
- Setting up repeatable ML framework
- Performing A/B testing with classical, sequential and Machine learning methods using MLOps best practices
- Extracting statistically valid insights in relation to the business objective

Here is the summary of tasks you will perform.

- Read this document carefully and make sure you have understood the business and data analysis objectives.
- Obtain the data from here
- Read the main reference paper and blog entries. We highly recommend you get a good understanding of the subtleties involved in the A/B testing framework. In particular why is it important to not perform the classical A/B testing analysis while the experiment is running? Study the recommended Kaggle kernels to get a better understanding.
- Understand the data. Make visualisation and ensure you understand how the data is collected and what each feature is.
- Attempt all tasks defined below.
- Upload your jupyter notebook to your Github public repository.
- For the interim submission, a PDF report and link to your GitHub repository is required & for final submission both link to your GitHub repository and your PDF report are requested .
- If you have any questions or confusions regarding what you are expected to do in this project or how to submit, please contact the team

Task 1: A/B testing framework

Task 1.1: Understanding A/B testing framework

Please prepare a document (max two pages) with brief answers to the following questions.

- Which online users belong to the control and exposed groups?

- How are the users targeted?
- Could we use the counts of yes and no answers to make a judgement on which experiment is performing better? For example if #yes > #no for the exposed group than the control group, could we declare that the ad had a significant impact Why or why not?
- What is the statistical process that generates the data? Which kind of statistical model will you use if you were to simulate the data?
- Assessment of the statistical significance of an A/B test is dependent on what kind of probability distribution the experimental data follows. Given your answer above, which statistical tests (z-test, t-test, etc.) are appropriate to use for this project?
- In classical (frequentist) A/B testing, we use p-values to measure the significance of the experimental feature (being exposed to an ad in our case) over the null hypothesis (the hypothesis that there is no difference in brand awareness between the exposed and control groups in the current case). How are p-values computed? What information do p-values provide? What are the type-I and type-II errors you may have in the analysis? Can you comment on which error types p-values are related?
- How does the classical A/B testing (using z-test, f-test, etc.) framework work?
- How does sequential A/B testing work?
- What are some of the advantages of sequential A/B testing?
- How is A/B testing done using machine learning? What is the core idea behind this approach? In other words, what part of the machine learning analysis provides the insight regarding the high or no significance of the experimental feature?
- What are the pros and cons of using Machine learning to perform A/B testing?
- In max three statements, make a problem formulation for machine learning and specify the target variable

Task 1.2 : Classic and sequential A/B testing analysis

- Perform data exploration to count unique values of categorical variables, make histogram, relational, and other necessary plots to help understand the data. For each of the plots you produce, write a description of what the plot shows in markdown cells.
- Perform hypothesis testing: apply the classical p-value based algorithm and the sequential A/B testing algorithm for which a starter code is provided..
- Are the number of data points in the experiment enough to make a reasonable judgement or should the company run a longer experiment? Remember that running the experiment longer may be costly for many reasons, so you should always optimize the number of samples to make a statistically sound decision.
- What does your A/B testing analysis tell you? Is brand awareness increased for the exposed group?

Task 2: A/B testing with Machine Learning

Task 2.1: MLOps planning and set up

- Following the Data Versioning and Reproducible ML with DVC and MLflow guideline do the following
    - Create a git repository named "abtest-mlops". Add your previous codes and follow a similar structure as what is shown in the video above.
    - Set up DVC in your github repository
    - Set up MLFlow
    - By following this reference, add a CML CI/CD workflow in your repository.

Task 2.2: ML modelling with MLOps

- Split data by browser and platform_os, and version each split as a new version of the data in dvc.
- For each version of the data do the following
    - Split the data into 70% training, 20% validation, and 10% test sets.
    - Based on the reading material provided, apply machine learning to the training data. Train a machine learning model using 5-fold cross validation using the following 3 different algorithms:
        - Logistic Regression
        - Decision Trees
        - XGBoost
    - Define the appropriate loss function for the model using the validation data.
    - Compute feature importance - what's driving the model? Which parameters are important predictors for the different ML models? What contributes to the goal of gaining more "Yes" results?
    - Which data features are relevant to predicting the target variable?
    - For each of the ML algorithms above, find the best model by tuning their hyperparameters and each time adding the tried models in MLFlow.
    - Prepare a Dockerfile for your project so that your model can be deployed in a docker container.

Task 3 : Interpretation & Reporting

- Explain what the difference is between using A/B testing to test a hypothesis vs using Machine learning to learn the viability of the same effect?
- Explain the purpose of training using k-fold cross validation instead of using the whole data to train the ML models?
- What information do you gain using the Machine Learning approach that you couldn't obtain using A/B testing?

- Prepare a presentation (20 slides max) to present your analysis to your company. This should include:
    - Objective of the study
    - Methods
    - Data
    - Results using both methods
    - Comparison of the two methods
    - Overall results
    - Recommendation and outcomes
    - Limitations of the analysis
    - References.

Interim Submission

- Share a report that addresses the points from task 1 (answer all questions in task 1.1.). Maximum of 3 pages - PDF format please. Prepare this in a format that you could share as a learning exercise with 3rd-year students at your university.

Feedback

You may not receive detailed comments on your interim submission, but will receive a grade.

Final Submission

- Link to your code in GitHub
- ScreenShots showing
    - Multiple data versions in your DVC store
    - Multiple model versions in your MLFlow dashboard
- A blog post entry (which you can submit for example to Medium publishing) in the form of a PDF report. The main thesis of your report should be about applying machine learning for A/B hypothesis testing. You should cover at the minimum the following topics
    - Basics of A/B testing and its use cases.
    - Limitations and challenges of classical A/B testing.
    - Sequential A/B testing pros and cons.
    - A/B testing formulation in Machine Learning context
    - Data review and ML A/B testing result.
    - The advantage of using MLFlow and DVC in ML experimentation.

Feedback

You will receive comments/feedback in addition to a grade.

References

Key Papers and Blogs

- Key concepts:
  - Standard error - Wikipedia Make sure you understand the difference between the ***standard deviation of the population***, the ***standard deviation of the sample,*** the ***standard deviation of the mean*** (which is the standard error), and ***the estimator of the standard deviation of the mean*** (which is the most often calculated quantity, and is also often colloquially called the *standard error*)
  - Likelihood-ratio test - Wikipedia
- Classical A/B testing
  - http://sl8r000.github.io/ab_testing_statistics/
  - http://www.qubit.com/wp-content/uploads/2017/12/qubit-research-ab-test-results-are-illusory.pdf
  - https://projector-video-pdf-converter.datacamp.com/6165/chapter3.pdf
- Sequential testing
  - https://www.austinrochford.com/posts/2014-01-01-intro-to-sequential-testing.html
  - https://www.jstor.org/stable/2346379?seq=1
  - https://blog.rankdynamics.com/2015/10/27/the-proof-is-in-the-pudding/
- Machine Learning based A/B testing
  - A/B Testing with Machine Learning - A Step-by-Step Tutorial
- Python package
  - https://github.com/shansfolder/AB-Test-Early-Stopping
  - https://github.com/Testispuncher/Sequential-Probability-Ratio-Test
- Sequential testing R package
  - https://github.com/mdcramer/SPRT

Must Read

- Statistical Significance in A/B Testing – a Complete Guide
- A/B test with Python
- A Refresher on A/B Testing
- A/B Testing: Analysis of Credit Card Marketing Campaign | by Kailash Hari | Analytics Vidhya
- A/B Testing Statistics: An Easy-to-Understand Guide
- Sequential A/B Testing: Workflow and Advantages over Classic Experiments

Examples

- (Bio)statistics in R: Part #3
- Unit 3 - Hypothesis Testing

- Learning About User Retention - Meta Kaggle
- https://dvc.org/doc/user-guide
- https://www.mlflow.org/docs/latest/tutorials-and-examples/tutorial.html

Annex

To obtain full marks, you may consider addressing the following high level elements.

Reports & Slides

- Big Picture
    - The objective of the work is clearly stated
    - The stated objective is correct (the reporter has understood the task)
- Details
    - Consistent Voice in a section (1st person or 3rd person)
    - The work is clearly motivated (e.g. main challenge this work directly or indirectly addresses))
    - Data source clearly described
    - Data structure clearly outlined (e.g. date ranges stated, condition at which data collected, etc.)
    - Evaluation metric clearly outlined
    - Method clearly outlined with clear description of
        - Pre-processing
        - Model training, validation, and testing
        - Model deployment
    - Challenges encountered and addressed stated
    - Valid insights drawn
    - Valid conclusions drawn
- Style
    - Uniform across pages and slides
    - Pleasing density (low better), font, color and format. (for slides this guideline)

Community

- Supporting other learners by answering questions
- Asking good questions
- Participating (not only attending) daily standups
- Sharing links and other resources with other learners

Notebook (code) Structure

The main python authority for code style guide is the PEP 8 guideline. All coding styles are guided accordingly. We also use PEP 20 - The Zen Of Python - to make all other coding based judgments. Some important highlights are as follows.

Markdown: in code comments & Readme

- Have section headings in jupyter notebooks, and function comments in functions.
- Have a basic explanation of what the code does below the section
- Have a good explanation of the approach taken for each section of code
- All sections and subsections have headers and have a reasonable explanation

Variable Naming

- Consistent. Function names should be lowercase, with words separated by underscores as necessary to improve readability.
- Variable names follow the same convention as function names.
- Standard (is followed by comment OR descriptive OR creative OR reasonable)

Python Functions

- Follow object oriented programming
- Put general utility functions in a separate module
- Reuse code as much as possible - make functions more general when possible

Figures

Our guideline here is the infamous Google Material Design guideline which is the global definition for UI/UX and Data Visualisations.

Axes

- Appropriate font size (readable)
- Readable titles
- Has units
- Has legend (explains multiple linestyle, colors, markers are used)
- Has caption (explains what the plot is about)

Type of Figure

- Appropriate for the data
- Innovative

Github

- Have CI/CD config files e.g. Github workflow, CML, Travis
- Have a readme that explains what the repository is about and how to use it
- Use multiple branches to manage multiple development streams

- Make code installable and or have Dockerfile with appropriate requirements.txt to allow smooth deployment.
- Have unit tests
- Do frequent commits