Introducing the Project

We at 365 Data Science have carefully designed a **real-world project** that can serve as a great addition to your professional portfolio.

Your task is to analyze a database, which shows student engagement with the 365 platform, and identify key areas of improvement. Keeping in mind the metrics that 365's CEO might find useful, build a **single-page dashboard** or **machine learning model**. Share your findings with the community, uncover valuable insights, and gain meaningful experience working on a real-life database.

Project Specifications

Case Description

365 Data Science is an online learning platform specializing in data science courses. Students learn by watching video content, then evaluate their knowledge by taking quizzes, practice exams, course exams, and career track exams. The exams can be attempted more than once.

Project Database and Auxiliary Files

The period under analysis in the database is **January 1, 2022 – October 20, 2022**.

- You can download and work with the data in one of two ways: a MySQL database of 11 tables or a collection of 11 .csv files.
- You are not obligated to use all tables included in the database—only those you find relevant for your analysis.

Project Structure

The project consists of two parts: **dashboard design** and **building an ML model**. You can participate in the challenge with either part or submit both.

There are two main ways to undertake the project:

- **Unguided:**
  You only need to read the brief description of each task and analyze the metrics you find most relevant for depicting students' engagement.
- **Guided:**
  You can use optional descriptions with detailed instructions regarding each task. Please note that this option offers *ideas*, not solutions. There is no right or wrong way to complete the challenge.
-

1. Dashboard Structure

Instructions

The main objective is to build a single-page dashboard with key metrics, insights, and visualizations on student engagement.

Start with defining the questions you would like to give answers to. Determine what visualization best fits a given query and how the information could be filtered. Then, create a rough sketch of the dashboard.

Optional Instructions

You should define more than one question you'd like to tackle with your dashboard. Compose a list of everything you'd like to cover first, and then structure your design accordingly. Here are some examples of questions you may want to answer:

1. Which courses are the most watched by students? How are they rated?
2. How would you define engagement (examples could be onboarding, minutes watched on the platform, exams/quizzes taken, etc.)?
3. What key performance indicators (KPIs) are relevant to the problem?

4. How many students register each month? What fraction of these students are also onboarded?
5. Do students watch more content with time? Is this seasonally dependent? Does it depend on marketing campaigns, promo periods, etc.?
6. How do the students engage with the platform based on user type (free or paid), subscription type (monthly, quarterly, or annual), and country?
7. Which are the countries with the most students registered? Does this number scale proportionally with the number of minutes watched per country?

Please note that you're not limited to these questions only. Feel free to add any other interesting analyses from which your dashboard could benefit.

In the downloadable resources, you will find a **dashboard skeleton** that we have created as an example solution to the project. You could use it either as an inspiration or follow it closely.

The skeleton consists of the following elements.

1. Charts

1. A **funnel** showing the total number of users from a given country:
    a. Display the first 5 countries with the largest number of users.
    b. Depict each country as a horizontal bar.
    c. The bar length depends on the number of students from that country.
    d. Stack all 5 horizontal bars and sort the chart in descending order.
2. A **funnel** showing the minutes watched on the platform by users from a given country:
    a. Display the first 5 countries with the largest number of users.
    b. Depict each country as a horizontal bar.
    c. The length of the bar depends on the minutes watched by each country.
    d. Stack all 5 horizontal bars and sort the chart in descending order according to the number of users.
3. A **bar-and-line chart** showing the minutes watched:
    a. The height of the bars depends on the number of minutes watched.
    b. The line represents the average number of minutes watched.

    c. Visualize it monthly.
4. A **bar chart** showing the number of registered users:
    a. The height of the bar represents the number of newly registered users.
    b. A number of the students in a given bar have also onboarded (following the definition of an onboarded student). These are to be colored differently so we can visually assess how this number compares to the total number of registered users.
    c. Visualize it monthly.

2. Tables

1. A **table** with 5 columns showing the top 5 most watched courses:
    a. The first column shows the courses' name.
    b. The second column shows the total number of minutes watched from each course.
    c. The third column shows the average minutes watched (number of minutes divided by the unique number of users that have watched the course).
    d. The fourth column shows the number of ratings for each of these courses.
    e. The fifth and final column shows the average rating for each course.

3. KPIs

1. A **'Registered Students' field** that shows the number of registered users
2. A **'Minutes Watched' field** that shows the number of minutes watched on the platform
3. An **'Average Minutes Watched' field** that shows the number of minutes watched on the platform, divided by the number of unique users who have contributed
4. A **'% Onboarded from Registered' field** that shows the percentage of registered users who have onboarded (out of all registered users)

4. Parameters

1. **'Start Date'** and **'End Date' parameters** that determine the time period of the collected data (**January 1, 2022 – October 20, 2022)**
2. A **'User Type' parameter** that allows for filtering by the following criteria:
    a. All
    b. Free

    c. Paid
3. A **‘Subscription Type’ parameter** that allows for filtering by the following criteria:
    a. All
    b. Monthly
    c. Quarterly
    d. Annual
4. A **‘User Country’ parameter** that allows for filtering by one or more countries

2. Data Preparation

Instructions

You must carefully prepare the data so that you feed it into your favorite visualization program and create the plots.

You can use MySQL to retrieve and join information from multiple tables, creating a sophisticated analysis. You are also welcome to perform this pre-processing step in Python or any other programming language you feel most comfortable with. You can also manipulate the relevant data directly in Tableau, Power BI, or another visualization tool.

Optional Instructions

Let’s think about the information we need to retrieve for each of the visualizations, following the numbering from the Dashboard Structure section and the dashboard skeleton. The following tables can be constructed in, for example, MySQL.

- Construct a data source (**Source 1**), which will later form **Table 1** in the dashboard. The source should contain the following columns:

        - Course name (Varchar)
        - Total minutes watched for each course (Decimal)
        - Average minutes for each course (Decimal)
        - Number of ratings for each source (Integer)
        - Average rating for each course (Decimal)
- Choose the User Type definition you want to work with from the **Data Dictionary** Construct a data source (**Source 2)**, which will later form **Chart 1.4**, **KPI 3.1**, and **KPI 3.4** in the dashboard. The source should contain the following columns:
    - User ID (Integer)
    - Date of the user's registration (Date)
    - Which country the user comes from (Varchar)
    - Whether the user is onboarded or not (Boolean)
    - Whether the user is paid or not (Boolean)
    - The subscription type of the user (Integer or NULL if the user is free)
- Construct a data source (**Source 3)**, which will later form **Chart 1**, **Chart 1.2**, **Chart 1.3**, **KPI 3.2**, and **KPI 3.3** in the dashboard. The source should contain the following columns:
    - User ID (Integer)
    - When a video was watched (Date)
    - Which course the video is from (Integer)
    - The number of minutes watched (Decimal)
    - Whether the user is paid or not (Boolean)
    - The subscription type of the user (Integer or NULL if the user is free)
    - Date of the user's registration (Date)
    - Which country the user comes from (Varchar)
    - Whether the user is onboarded or not (Boolean)
- The following columns from **Source 2** and **Source 3** would be used when filtering with **Parameters 1-4.4**:
    - Date of registration
    - Date in which a video is watched
    - Whether the user is paid or not
    - The subscription type of the user (null if the user is free)
    - Which country the user comes from

3. Dashboard Construction

Instructions

This is where you reveal your most creative side.

Create each visualization separately, then build your single-page dashboard chart by chart. Make sure it is visually attractive, intuitive, and practical so that people can easily interact with your findings.

Optional Instructions

Let’s build all visualizations individually in Tableau.

- Visualizations that use **Source 1**:
    - **Table 2.1**
        - Create a table that displays all 5 metrics for all courses on the platform (course name, minutes watched, average minutes watched, number of ratings, and average rating).
        - Display only the top 5 courses by minutes watched.
- Visualizations that use **Source 2**:
    - **Chart 1.4**
        - Place the distinct count of users onto Rows as a continuous measure.
        - Place the date of registration onto Columns as a discrete month.
        - Drag the Onboarded field onto the Color mark to color the portion of registered and onboarded students.
        - Drag the user count onto Label. Represent the count as Percent of Total, computed using Cell. That way, we will display the onboarded and not-onboarded students as a percentage of all registered users in a given month.
    - **KPI 3.1**

        - Drag the count of users onto the Text mark.
    - **KPI 3.4**
        - Calculate the ratio between the students who have both registered and onboarded, and all registered students.
        - Represent it as a percentage.
        - Drag it onto the Text mark.
- Visualizations that use **Source 3**:
    - **Chart 1.1**
        - Create a horizontal bar chart that shows the number of users from each country.
        - Display only the top 5 countries by the **number of users**.
        - Modify the chart such that it is in the form of a funnel.
    - **Chart 1.2**
        - Create a horizontal bar chart that shows the minutes watched from each country.
        - Display only the top 5 countries by the **number of users**.
        - Modify the chart in the form of a funnel.
    - **Chart 1.3**
        - Create a bar chart with 12 bars, each representing a different month of the year.
        - The height of the bars represents the number of minutes watched during a given month.
        - Create a line chart overlaid with the bar chart, whose values represent the average minutes watched in a given month—the number of minutes watched divided by the number of users who've watched content during the month in question.
    - **KPI 3.2**
        - Drag the sum of all minutes watched onto the Text mark.
    - **KPI 3.3**
        - Drag the sum of all minutes watched, divided by the distinct count of users, onto the Text mark.

4. Results Discussion

Instructions

After you've completed your dashboard, the project requires you to discuss the plots you've built, the information they convey, and the conclusions we can draw from them.

This part is essential as it is the underlying reason for creating a dashboard of this sort. Given specific results and certain information from the dashboard, the 365 team can determine what actions can be undertaken to improve the company's growth and deliver its students the best learning experience.

Optional Instructions

Go back to the questions from the Dashboard Structure section and try to answer them with the respective visualizations:

1. Which courses are the most watched by students? How are they rated?
   Use **Table 2.1** to answer.
2. How would you define engagement (examples could be onboarding, minutes watched on the platform, exams/quizzes taken, etc.)?
   In this dashboard example, we've looked at engagement in two ways: looking into the fraction of onboarded students and studying the minutes watched per course, per month, and as a total.
3. What key performance indicators (KPIs) are relevant to the problem?
   We've decided it's essential to display the total number of registered students, the total number of minutes watched, the average minutes watched, and the fraction of registered students who have also been onboarded.
4. How many students register each month? What fraction of these students are also onboarded?
   Use **Chart 1.4** to answer.
5. Do students watch more content with time? Is this seasonally dependent? Does it depend on marketing campaigns, promo periods, etc.?
   Use **Chart 1.3** to answer the question.
6. How do the students engage with the platform based on user type (free or paid), subscription type (monthly, quarterly, or annual), and country?
   Use the dashboard filters to answer.
7. Which are the countries with the most students registered? Does this number scale proportionally with the number of minutes watched per country?
   Use **Charts 1.1** and **1.2** to answer the question.

Machine Learning Model

1. Building the Model

Instructions

The following task has you develop a machine learning model to predict whether a Free Plan user would convert to a paid subscriber or not.

Think of the features that could help determine the outcome. Share your results by supporting them with relevant metrics, a confusion matrix, or another industry-standard method for model evaluation. **Note that this classification problem deals with a heavily imbalanced dataset.**

Optional Instructions

The following instructions refer to Python's **imblearn** and **sklearn** libraries. The former is used for re-sampling imbalanced data points, while the latter is applied when we perform the machine learning part.

A list of useful tools and recommended libraries include:

- Python 3.8.8
- imbalanced-learn 0.9.1
- numpy 1.20.3
- pandas 1.3.4
- scikit-learn 1.1.2

You are encouraged to play around with the example solution described below, fiddle with the parameters, try out new approaches, think of different features that could be extracted from the data, etc. **Use the instructions as inspiration rather than a solution.**

- First, consider the features that could be used as a defining difference between users who are likely to subscribe and those who are not, including (but not limited to):
    - Minutes watched on the platform
    - Engagement with quizzes
    - Engagement with exams
    - Engagement in the Q&A hub

You can evaluate the metrics for a fixed period. Our experience has shown that students generally convert within 1 or 2 weeks after entering the platform. Therefore, it is reasonable to confine the features to within several days after registration—for example, minutes watched *N* days after registration.

- Generate SQL queries that create a table with the following columns as inputs:
    - Minutes watched on the platform (Decimal)
    - Number of days in which a student was engaged with the platform (Integer)
    - Engaged with quizzes (Boolean)
    - Engaged with exams (Boolean)
    - Engaged with the Q&A hub (Boolean)

And the following column as a target:

    - Subscribed (Boolean)

Note: *You can refer to our 'SQL' course, which covers queries in depth.*

- Export your table as a .csv file.
- Switch to Python and read the .csv file using, for example, the **pandas** Apply any preprocessing needed.

Note: *You can refer to our 'Data Cleaning and Preprocessing with pandas' course, which teaches you about reading and manipulating data with pandas.*

- Split your data into training and testing sets.

- Choose an approach that helps us handle the imbalance of the data. Choose an over-sampling method from the imblearn library. Fiddle around with the parameters—most importantly with **sampling_strategy**. Try to understand how this parameter changes the number of data points from each class.

Note: *Remember to apply the over-sampling only to the training dataset! The test dataset is reserved for testing the model's performance on data that the model has never encountered.*

- In general, applying an under-sampling technique right after over-sampling increases the model's performance. Choose an under-sampling method from the imblearn library and apply it on you over-sampled data. Fiddle around with the parameters—most importantly **sampling_strategy**. Try to understand how this parameter changes the number of data points from each class.
- Choose a machine learning model to work with. (You can test the performance of several different models.) Feed the re-sampled data into the algorithm.

Note: *You can refer to Module 3: Machine and Deep Learning on our platform, where we've covered many different machine learning algorithms.*

- Reduce the problem's dimensionality as much as possible—find any features that might not contribute to the model's performance and remove them.
- Apply a hyperparameter tuning technique, for example grid search.
- Apply a cross-validation technique to better train your model.
- Use your model to predict the outcome given the features from your test set.
- Construct a confusion matrix that would clearly show how many data points from the test set the model predicted correctly and how many of them were misclassified instead.
- Retrieve and discuss relevant metrics such as accuracy, precision, recall, F1 score, AUC, etc.