# CMPT 423-820 MINI PROJECT 2: FORMULA RECOGNITION

Team formation due date: February 27, 2025 at 4pm CST
Project due date: March 12, 2025 at 4pm CST

(a) "$2 + 0$" (b) "$5 + 6$" (c) "$2 \cdot 9$" (d) "$4 - 1$" (e) "$3/3$"

Figure 1: **Formula image dataset.** The provided datasets consists of images displaying simple formulas such as "$2+0$" or "$2 \cdot 9$". Each image consists of a digit ranging from zero through nine, an operator character, and a second digit. The used operators are addition, indicated with an "a" character, multiplication, indicated with an "m" character, subtraction, indicated with a "s" character, and division, indicated with a "d" character.

In this project, you will work in teams of three to four to develop a machine learning model that detects formulas displayed in an image. Figure 1 illustrates examples from the provided dataset: Each image contains a formula that either adds, multiplies, subtracts, or divides two single digit numbers. The goal of this project is to use probabilistic modelling techniques to derive, implement, and train a model that predicts the formula displayed.

Using the EMNIST dataset [1], the provided formula dataset was created by sampling digit and lower-case character images and composing them into the images shown in Figure 1. Each image is then labeled with a three character string, for example `2p0`, `5p6`, or `2m9`.

Formulas were sampled uniformly at random. For an image $\boldsymbol{x}$ that is labelled with a formula $y_1, y_2, y_3$, you can assume that $y_1$, $y_2$, and $y_3$ are independent from each other. The chance that a label $y_i$ is a specific digit or letter only depends on the image $\boldsymbol{x}$.

## Team formation

In the first phase of the project, you will have to form teams of three or four students. These teams must be submitted by February 27 at 4pm CST by submitting the *Mini Project 2 Team Formation* assignment on Canvas. Every team member must submit this assignment independently with identical name lists.

**Team Gitlab repository.** For this project, each team must create its own private team repository. This can be done by forking the template repository found in the course Gitlab group here (need to insert link). This fork must be created by one team member and this team member must then invite all other team members and the instructor (Lucas Lehnert) as a "Developer" to this repository. The link to the repository location must be submitted on Canvas as the solution to the *Mini Project 2 Team Formation*. Only the team members and the instructor are allowed to have read access to the repository.

**Late day policy.** You can use late days for this project. If a late day is used, then a late day will be deducted from every team member's account. If one team member has only three late days remaining, but another team member has still four late days remaining, then the whole team can use at most four late days. However, if all team members have only three or fewer late days in their account, then the team can use at most three late days for this project.

## Project instructions

The project goal is as follows:

1. Derive a loss objective for fitting a model that predicts formulas given images using the maximum likelihood method.

2. Fit a linear model (similar to linear regression, linear logistic regression, or linear softmax regression) to this dataset to predict formulas correctly. Such a linear model should predict about 75% of the test set correctly. This model is referred to as the *baseline model*. This model must be implemented in PyTorch and optimized using a gradient descent algorithm.
3. Develop an improved model that outperforms the baseline model in a statistically significant manner.
4. Submit a Gitlab repository with documentation for reproducing model training.
5. Submit a project report in this Gitlab repo that presents the following:
   (a) A derivation of the loss objective used for fitting both models together with a discussion of why this approach is appropriate.
   (b) A performance report of the baseline and improved models and argue why the improved model outperforms the baseline model in a statistically significant manner.
   (c) Documentation of how to launch the training loops for each of the two models and reproduce the experiments.

**Project report formatting instructions.** The project report must be written in Latex using the provided template. The report must be at most two pages long, including figures but excluding any references you may want to use. This report must be submitted as a `report.pdf` file stored in the private team repository. This project report must clearly specify the team members in the author section. *Make sure to use your full name as listed in Canvas.*

**Project implementation instructions.** Both baseline and improved models should be implemented in PyTorch and should be optimized using a gradient descent algorithm.

## Project evaluation

The project will be evaluated based on the following aspects:

| Grade weight | Aspect |
|---|---|
| 30% | Derivation of loss objective and explanation of why your approach is appropriate. |
| 15% | Description of how the two models are implemented and optimize the loss objective. |
| 5% | Description of different hyperparameters and how they are tuned. For full marks at least three different hyperparameters should be discussed and optimized for each model. The report must clearly document which hyperparameters resulted in best possible performance. |
| 10% | Explanation of how or why the improved model should outperform the baseline model. |
| 15% | Reporting of performance results. |
| 5% | Model performance. For full marks, the baseline model must predict at least $75\%$ of the test set correctly and the improved model must outperform the baseline model in a statistically significant manner. |
| 20% | Reproducibility of code. The project report must contain documentation about how to start each training loop and the provided code must start the training loop correctly. The code base must be documented (similar to how Mini Project 1 was documented) to explain how the methods discussed in the report are implemented. |

The same grade will be assigned to each team member.

## References

[1] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. Emnist: an extension of mnist to handwritten letters. *arXiv preprint arXiv:1702.05373*, 2017.