# CSCI433/CSCI933
# Machine Learning Algorithms and Applications
# Assignment Problem Set #3

Lecturer: Prof. Philip O. Ogunbona(philipo@uow.edu.au)
School of Computing and Information Technology
University of Wollongong

Due date: Friday June 3, 6:00 p.m.

## Introduction

This is a group assignment and can be considered a "micro" research project (roughly 4 weeks). The idea is for you and your group to explore the interplay between the choice of loss function, optimization scheme and deep network performance. You will also have opportunity to read some research papers, albeit not at a very deep level. The research papers discuss various aspects of the loss function. Read as much of the papers as you can understand and discuss within your group. Some of the fundamentals we have discussed in the class should help you understand the "language" and argument used in the papers.

You are given a classification task using the MNIST-FASHION dataset. Your baseline classifier architecture is a convolutional neural network (CNN) with the following layers:

- Input layer
- Batch normalization
- Conv2D(filters= ??,kernel_size=[?, ?], padding="same", activation=relu OR some other activation)
- MaxPooling2D(pool_size=[?, ?], strides=?)
- BatchNormalization
- Dropout (select a value between 0.0 and 1.0)
- Conv2D(filters= ??,kernel_size=[?, ?], padding="same", activation=relu OR some other activation)
- MaxPooling2D(pool_size=[?, ?], strides=?)
- BatchNormalization
- Dropout (select a value between 0.0 and 1.0)
- Flatten
- Dense layer

Note that the question marks represent your design choices.

However, you will extend the task beyond classification and explore performance issues including convergence and accuracy as they are affected by network architecture (number of layers), optimizer, and loss function. You will implement appropriate loss functions and optimizers (based on the TensorFlow deep learning framework - `https://www.tensorflow.org/api_docs/python/tf/keras/losses`). Not all the loss functions implemented in TensorFlow are appropriate for this classification task. You will use literature provided and TensorFlow to select three (3) appropriate loss functions and justify your choice. The optimizers of interest are implemented in TensorFlow (`https://www.tensorflow.org/api_docs/python/tf/keras/optimizers`) and you will choose four (4) for your experiments. You must justify your choice. As in Assignments 1 & 2, you will find the books by Chollet (2021) and Géron (2019) useful while implementing the network and conducting the experiments.

### Papers provided for reading and insights

The following papers will be found useful for this assignment. You may not fully understand them (do your best), but various parts of the papers use language and concepts you are already familiar with.

1. Janocha and Czarnecki (2017) - "On Loss Functions for Deep Neural Networks in Classification"
2. Kukačka, Golkov, and Cremers (2017) - "Regularization for Deep Learning: A Taxonomy"
3. Rosasco, Vito, Caponnetto, Piana, and Verri (2003) - "Are Loss Functions All the Same?"

## Experimental design

From an experimental design perspective you can consider this to be 3x4 factorial design (see Table 1). This means that you have at least 24 results to record from the experiments your group conducts. For example, when you consider convergence rate you will record this value for each of the combinations ([loss type, optimizer]) in Table 1. Consider spreading the work equally among your group members so everyone gain experience and the workload is even. You may also consider using Google Colaboratory for your experiments.

If we denote the convergence rate (during training) for a particular combination of loss function ($l$) and optimizer ($o$), at a given network depth ($d$) by $\mathcal{C}_i(l, o)$, you will need to obtain $i = 12$ measurements. Similarly, you will record your results for accuracy of classification. Note that it is up to you to decide how you want to vary the depth of the network or the number of epochs to use during training. All these decisions must be reported with some justification. In any machine learning task, you must split the data for training, validation and testing. The accuracy you record is for **test data**.

### Summary of task

1. Form a group of four (4). Your group may conatin a mixture of Masters and Senior Undergraduate students. You need special permission to form a group of 5 students.

Table 1: Table showing the experimental design

| | | Optimizer ($o$) | | | |
|---|---|---|---|---|---|
| | | Optimizer 1 | Optimizer 2 | Optimizer 3 | Optimizer 4 |
| Loss function($l$) | Loss type 1 | Perf | Perf | Perf | Perf |
| | Loss type 2 | Perf | Perf | Perf | Perf |
| | Loss type 3 | Perf | Perf | Perf | Perf |

2. Using **TensorFlow** build a CNN-based classifier for the Fashion MNIST dataset.

3. Train the model induced by your network. Take into consideration data split for training, validation and testing. Consider the optimizer and the loss function (see the links given above).

4. Conduct experiments according to the schedule illustrated in Table 1. Three, You will write a report based on result of your experiments:

   (a) Summarize your understanding of loss functions using the papers provided and any other papers you may find. Ensure that you appropriately reference the sources of your information. One page.

   (b) Summarize your understanding of optimizers using the papers, materials at the links and any other papers you may find. Ensure that you appropriately reference the sources of your information. One page.

   (c) Write appropriate code to conduct your experiment.

   (d) Based on the results obtained form your experiments discuss your findings and trends. You must aim to provide some theoretical explanation for your results. The papers provided may be helpful. Three pages.

# Report format

### Title page - 5 marks

Clearly print the full names (Surname, other names) and student ID number of all members of your group. You may give your report a title that reflects what you have done. For example, "A study of the influnce of loss function and optimizers on classification performance".

### Introduction to loss functions - 10 marks

This should be no more than one page and provides a description of loss functions and how they relate to classification task and network model. Provide example loss functions and their meaning. Which loss functions are appropriate for classification task.

### Introduction to optimizers - 10 marks

This should be **no more than one page** and must describe optimizers and their role in network model training. Provide example optimizers and their importance/properties. How does the opti-

mizer relate to loss function?

**Experiment - 40 marks**

This section should be **no more than three pages** and must include a description of your experimental setup. Such description must justify the choice of your loss function and optimizers. You will clearly present the results (not code) of your experiments in a table or graph to demonstrate that you have conducted the experiments. Remember that each experiment answers a question.

**Discussion - 40 marks**

This section must provide a **detailed discussion** of the results of your experiment. Pay attention to trends in performance as the factors change. You will provide explanations for the trend you have noticed. There must be a conclusion or recommendation about the optimum combination of loss function, optimizer and depth of network. The definition of optimum in this assignment is a compromise between convergence rate and classification accuracy. The papers provided and the books (already on Moodle) may be found useful in writing a good discussion.

# What needs to be submitted and things to consider

1. If your code does not run, you can only score a maximum of 60% of the maximum marks available for this assignment.

2. Submit a report (PDF format only) with sections as specified above. Name your report PDF file `student-name_student-number.pdf` (e.g. good_student_123456.pdf) . This is very important.

3. Submit a Python code (**NOT a jupyter notebook**) named `optimizer_loss.py` which must run on command line as

   ```
   python3 optimizer_loss.py
   ```

   Your code should have code snippet that looks like:

   ```
   import tensorflow as tf
   from tensorflow import keras
   ...
   fashion_mnist = keras.datasets.fashion_mnist
   (train_images, train_labels), (test_images, test_labels) = fashion_mnist.load_data()
   ```

   When your code runs, it will print out the results or your experiment.

4. You will lose marks if your use of English language in your report is unintelligible.

5. You will add the two items to be submitted in an archive file (".zip" or ".rar"). Give your archive file your `student-name_student-number.[zip][.rar]`. This is very important.

# References


Chollet, F. (2021). *Deep learning with python* (2nd ed.). Shelter Island, NY, USA: Manning Publications Co.

Géron, A. (2019). *Hands-on machine learning with scikit-learn, keras &and tensorflow: Concepts, tools, and techniques to build intelligent systems* (2nd ed.). CA, USA: O'Reilly Media, Inc.

Janocha, K., & Czarnecki, W. M. (2017). On loss functions for deep neural networks in classification. *arXiv*, *1702.05659*(v1), 1-10.

Kukačka, J., Golkov, V., & Cremers, D. (2017). Regularization for deep learning: A taxonomy. *arXiv*, *1710.10686*(v1), 1-23.

Rosasco, L., Vito, E. D., Caponnetto, A., Piana, M., & Verri, A. (2003). *Are loss functions all the same?* (Paper). Università di Genova, Via Dodecaneso 35,.