

# Project 2

You can choose between two options to complete the project. Details of each option are shown below:

- **Option 1:** Logistic Regression Implementation. Can be delivered as a two-student cooperative project or as a single student project.
- **Option 2:** Rain prediction Model delivered. Can only be delivered as a single student project.

## Logistic Regression implementation

In this project you will dive into the details of multinomial logistic regression and build a model that is able to classify the **Iris** dataset. All code must be implemented from scratch. To be considered as a successful implementation your model must produce the same results as the scikit-learn's Logistic Regression multinomial classification model.

## Objectives

This project will help you gain better understanding of the implementation of mathematical concepts using Python, apply object-oriented programming and explore the details of the entire ML model development process (from training with data to model evaluation).

## Dataset

Your classification model must be trained using all features and classes of the well-known **Iris** dataset compiled by R.A. Fisher. (<https://archive.ics.uci.edu/ml/datasets/iris>)

## Requirements

Your project is required to follow these guidelines. If one of the requirements is not met, the entire project will be considered as undelivered.

1. Implement <sup>Split data, Scaler</sup> **data preprocessing** and a <sup>Logistic</sup> **multinomial logistic regression** model using only your python code (your code can only have dependencies from the Numpy or Pandas packages). <sup>Metrics / co librerias accuracy, recall...</sup>
2. Use a scikit-learn's **metrics and logistic regression classifier** to compare the results of your model.
3. Your model must be implemented in a class and contain at least the following **methods**: <sup>Metodos</sup>
  - a) **.fit(x,Y)**: Train the model with samples x and labels Y.
  - b) **.predict(x)**: Predict the class labels of x. Returns a numpy array.
  - c) **.predict\_proba(x)**: Probability estimates of each class of x. Returns a numpy array.

## Rain prediction model

In this project you will explore the provided weather prediction dataset, verify data validity, create a **classification** model to predict the **Rain Tomorrow** target variable, analyze its performance, persist and deliver your best solution.

Your delivered model will be evaluated and ranked among all submissions using F1 score performance over a dataset of 4K reserved observations (Make sure you deliver the best performing model you can train).

## Objectives

During this project you will master the use of scikit-learn's pipelines and improve your confidence on model training techniques, evaluation metrics, and model selection process.

## Dataset

**The provided dataset contains about 10 years (2007-10-31 to 2017-06-24) of daily weather observations from many locations across Australia.**

RainTomorrow is the target variable to predict. It means -- did it rain the next day? Yes or No? This column is Yes if the rain for that day was 1mm or more.

## Source & Acknowledgements

Observations were drawn from numerous weather stations. The daily observations are available from <http://www.bom.gov.au/climate/data>.

An example of latest weather observations in Canberra:

<http://www.bom.gov.au/climate/dwo/IDCJDW2801.latest.shtml>

- Definitions adapted from <http://www.bom.gov.au/climate/dwo/IDCJDW0000.shtml>
- Data source: <http://www.bom.gov.au/climate/dwo/> and <http://www.bom.gov.au/climate/data>.
- Copyright Commonwealth of Australia 2010, Bureau of Meteorology.

## Dataset Columns

**Date:** The date of observation

**Location:** The common name of the location of the weather station

**Location:** The minimum temperature in degrees celsius

**MinTemp:** The maximum temperature in degrees celsius

**MaxTemp:** The amount of rainfall recorded for the day in mm

**Rainfall:** The amount of rainfall recorded for the day in mm

**Evaporation:** The so-called Class A pan evaporation (mm) in the 24 hours to 9am

**Sunshine:** The number of hours of bright sunshine in the day.

**WindGustDir:** The direction of the strongest wind gust in the 24 hours to midnight

**WindGustSpeed:** The speed (km/h) of the strongest wind gust in the 24 hours to midnight

**WindDir9am:** Direction of the wind at 9am

**WindDir3pm:** Direction of the wind at 3pm

**WindSpeed9am:** Wind speed (km/hr) averaged over 10 minutes prior to 9am

**WindSpeed3pm:** Wind speed (km/hr) averaged over 10 minutes prior to 3pm

**Humidity9am:** Humidity (percent) at 9am

**Humidity3pm:** Humidity (percent) at 3pm

**Pressure9am:** Atmospheric pressure (hpa) reduced to mean sea level at 9am

**Pressure3pm:** Atmospheric pressure (hpa) reduced to mean sea level at 3pm

**Cloud9am:** Fraction of sky obscured by cloud at 9am. This is measured in "oktas", which are a unit of eighths.

**Cloud3pm:** Fraction of sky obscured by cloud (in "oktas": eighths) at 3pm. See Cloud9am for a **description** of the values

**Temp9am:** Temperature (degrees C) at 9am

**Temp3pm:** Temperature (degrees C) at 3pm

**RainToday:** Boolean: 1 if precipitation (mm) in the 24 hours to 9am exceeds 1mm, otherwise 0

**RainTomorrow:** The amount of next day rain in mm. Used to create response variable RainTomorrow. A kind of measure of the "risk". (Target variable)

# Requirements

Your project is required to follow these guidelines. If one of the requirements is not met, the entire project will be considered as undelivered.

1. Implement a classification model using only **one** of the following algorithms (No additional algorithms must be included or evaluated in the project):
  - a. Logistic regression classifier
  - b. SVM classifier
  - c. Decision tree classifier
2. Use a scikit-learn pipeline object which contains all pre-processing, feature selection and training stages.
3. Your trained model must be **persisted** as name\_lastname.pkl using **joblib** and submitted **with the provided model delivery template** with your additional code needed to load and predict data. (Ensure that the model will be executed and can predict without errors).

## Expected delivery for both options

You are expected to deliver in a zip file named as **name\_lastname.zip**:

1. A .pdf report with the sections described below.
2. .ipynb notebook with all your code.
3. Model selection code and main reasoning for each step of the process.\*
4. Model training and evaluation metrics.
5. Persisted model in name\_lastname.pkl format and delivery template with the required code to run the persisted model.\*
6. Presentation slides.

\*Only for option 2.

## Grading

**Note: Max possible grade can be up to 110**

Model ranking	F1 score model rank among all submissions	10	
Code		20	
	<ul style="list-style-type: none"><li>- High-level overview of dataset done</li><li>- Initial data visualizations performed</li><li>- Cleaned dataset used for further analysis</li><li>- Sections and comments that make it easy to understand what's being done.</li><li>- Algorithm selection reasons and interpretations are presented</li><li>- Hyper parameter tuning for best performance done</li></ul>		
Report		55	
	Introduction:		5

	<ul style="list-style-type: none"> <li>- Summarize the purpose of the report and summarize the data / subject.</li> <li>- Include important contextual information about the reason for the report.</li> <li>- Summarize your analysis questions, your conclusions, and briefly outline the report.</li> </ul>		
	Data: <ul style="list-style-type: none"> <li>- Include written descriptions of dataset(s).</li> <li>- Analyze any bias and specific attention to features.</li> </ul>		5
	Methods: <ul style="list-style-type: none"> <li>- Explain how you gathered and analyzed data.</li> <li>- Explain the algorithms you used.(Analysis, preprocessing, models)</li> <li>- Explain your training and parameter selection process.(Description and reasoning)</li> </ul>		15
	Results: <ul style="list-style-type: none"> <li>- Describe the results of your analysis.</li> <li>- Provide all relevant metrics to evaluate the performance of your model.</li> <li>- Present metrics in an easy-to-understand format.</li> </ul>		20
	Conclusion: <ul style="list-style-type: none"> <li>- Restate the questions from your introduction.</li> <li>- Restate important results.</li> <li>- Include any recommendations for additional data as needed.</li> </ul>		10
Presentation		25	
	Comprehensibility to those outside DS		5
	Logic of presentation flow		5
	Clear recommendations		5
	Validity of ideas and strength of recommendation support		10