Assignment-3 (View only Draft)

Introduction

In this assignment, you will be using the loan dataset provided and the machine learning algorithms you have learned in this course in order to predict:

1. If a loan applicant will be able to repay the loan or not
   - This will help the bank to decide if it is risky to approve the loan application
2. Predict the client's income based on the information provided in the application
   - This can help the bank to further investigate if the provided documents for payslips are fishy or not.

   NOTE: this is a very challenging problem and we are not expecting very high accuracy in your predictions. However, you must apply all your analytic skills to build decent ML models;

Datasets

In this assignment, you will be given two datasets training.csv & test.csv (https://webcms3.cse.unsw.edu.au/COMP9321/22T1/resources/74268)

Here is the description of the columns in these datasets:

Row	**Description**
SK_ID_CURR	ID of loan in our sample
TARGET	Target variable (1 - client with payment difficulties: he/she had late payment more than X days on at least one of the first Y installments of the loan in our sample, 0 - all other cases)
NAME_CONTRACT_TYPE	Identification if loan is cash or revolving
CODE_GENDER	Gender of the client
FLAG_OWN_CAR	Flag if the client owns a car
FLAG_OWN_REALTY	Flag if client owns a house or flat
CNT_CHILDREN	Number of children the client has
AMT_INCOME_TOTAL	Income of the client
AMT_CREDIT	Credit amount of the loan
AMT_ANNUITY	Loan annuity
AMT_GOODS_PRICE	For consumer loans it is the price of the goods for which the loan is given
NAME_TYPE_SUITE	Who was accompanying client when he was applying for the loan
NAME_INCOME_TYPE	Clients income type (businessman, working, maternity leave,…)
NAME_EDUCATION_TYPE	Level of highest education the client achieved
NAME_FAMILY_STATUS	Family status of the client
NAME_HOUSING_TYPE	What is the housing situation of the client (renting, living with parents, ...)

REGION_POPULATION_RELATIVE	Normalized population of region where client lives (higher number means the client lives in more populated region)
DAYS_BIRTH	Client's age in days at the time of application
DAYS_EMPLOYED	How many days before the application the person started current employment
DAYS_REGISTRATION	How many days before the application did client change his registration
DAYS_ID_PUBLISH	How many days before the application did client change the identity document with which he applied for the loan
OWN_CAR_AGE	Age of client's car
FLAG_MOBIL	Did client provide mobile phone (1=YES, 0=NO)
FLAG_EMP_PHONE	Did client provide work phone (1=YES, 0=NO)
FLAG_WORK_PHONE	Did client provide home phone (1=YES, 0=NO)
FLAG_CONT_MOBILE	Was mobile phone reachable (1=YES, 0=NO)
FLAG_PHONE	Did client provide home phone (1=YES, 0=NO)
FLAG_EMAIL	Did client provide email (1=YES, 0=NO)
OCCUPATION_TYPE	What kind of occupation does the client have
CNT_FAM_MEMBERS	How many family members does client have
REGION_RATING_CLIENT	Our rating of the region where client lives (1,2,3)
REGION_RATING_CLIENT_W_CITY	Our rating of the region where client lives with taking city into account (1,2,3)
WEEKDAY_APPR_PROCESS_START	On which day of the week did the client apply for the loan
HOUR_APPR_PROCESS_START	Approximately at what hour did the client apply for the loan
REG_REGION_NOT_LIVE_REGION	Flag if client's permanent address does not match contact address (1=different, 0=same, at region level)
REG_REGION_NOT_WORK_REGION	Flag if client's permanent address does not match work address (1=different, 0=same, at region level)
LIVE_REGION_NOT_WORK_REGION	Flag if client's contact address does not match work address (1=different, 0=same, at region level)
REG_CITY_NOT_LIVE_CITY	Flag if client's permanent address does not match contact address (1=different, 0=same, at city level)
REG_CITY_NOT_WORK_CITY	Flag if client's permanent address does not match work address (1=different, 0=same, at city level)
LIVE_CITY_NOT_WORK_CITY	Flag if client's contact address does not match work address (1=different, 0=same, at city level)
ORGANIZATION_TYPE	Type of organization where client works
EXT_SOURCE_1	Normalized score from external data source
EXT_SOURCE_2	Normalized score from external data source
EXT_SOURCE_3	Normalized score from external data source
APARTMENTS_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
BASEMENTAREA_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
YEARS_BEGINEXPLUATATION_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor

YEARS_BUILD_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
COMMONAREA_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
ELEVATORS_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
ENTRANCES_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
FLOORSMAX_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
FLOORSMIN_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LANDAREA_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LIVINGAPARTMENTS_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LIVINGAREA_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
NONLIVINGAPARTMENTS_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
NONLIVINGAREA_AVG	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
APARTMENTS_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
BASEMENTAREA_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor

YEARS_BEGINEXPLUATATION_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
YEARS_BUILD_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
COMMONAREA_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
ELEVATORS_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
ENTRANCES_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
FLOORSMAX_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
FLOORSMIN_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LANDAREA_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LIVINGAPARTMENTS_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LIVINGAREA_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
NONLIVINGAPARTMENTS_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
NONLIVINGAREA_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
APARTMENTS_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor

BASEMENTAREA_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
YEARS_BEGINEXPLUATATION_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
YEARS_BUILD_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
COMMONAREA_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
ELEVATORS_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
ENTRANCES_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
FLOORSMAX_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
FLOORSMIN_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LANDAREA_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LIVINGAPARTMENTS_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
LIVINGAREA_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
NONLIVINGAPARTMENTS_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
NONLIVINGAREA_MEDI	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor

FONDKAPREMONT_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
HOUSETYPE_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
TOTALAREA_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
WALLSMATERIAL_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
EMERGENCYSTATE_MODE	Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor
OBS_30_CNT_SOCIAL_CIRCLE	How many observation of client's social surroundings with observable 30 DPD (days past due) default
DEF_30_CNT_SOCIAL_CIRCLE	How many observation of client's social surroundings defaulted on 30 DPD (days past due)
OBS_60_CNT_SOCIAL_CIRCLE	How many observation of client's social surroundings with observable 60 DPD (days past due) default
DEF_60_CNT_SOCIAL_CIRCLE	How many observation of client's social surroundings defaulted on 60 (days past due) DPD
DAYS_LAST_PHONE_CHANGE	How many days before application did client change phone
FLAG_DOCUMENT_2	Did client provide document 2
FLAG_DOCUMENT_3	Did client provide document 3
FLAG_DOCUMENT_4	Did client provide document 4
FLAG_DOCUMENT_5	Did client provide document 5
FLAG_DOCUMENT_6	Did client provide document 6
FLAG_DOCUMENT_7	Did client provide document 7
FLAG_DOCUMENT_8	Did client provide document 8
FLAG_DOCUMENT_9	Did client provide document 9
FLAG_DOCUMENT_10	Did client provide document 10
FLAG_DOCUMENT_11	Did client provide document 11
FLAG_DOCUMENT_12	Did client provide document 12
FLAG_DOCUMENT_13	Did client provide document 13
FLAG_DOCUMENT_14	Did client provide document 14
FLAG_DOCUMENT_15	Did client provide document 15
FLAG_DOCUMENT_16	Did client provide document 16
FLAG_DOCUMENT_17	Did client provide document 17
FLAG_DOCUMENT_18	Did client provide document 18
FLAG_DOCUMENT_19	Did client provide document 19
FLAG_DOCUMENT_20	Did client provide document 20
FLAG_DOCUMENT_21	Did client provide document 21
AMT_REQ_CREDIT_BUREAU_HOUR	Number of enquiries to Credit Bureau about the client one hour before application

AMT_REQ_CREDIT_BUREAU_DAY	Number of enquiries to Credit Bureau about the client one day before application (excluding one hour before application)
AMT_REQ_CREDIT_BUREAU_WEEK	Number of enquiries to Credit Bureau about the client one week before application (excluding one day before application)
AMT_REQ_CREDIT_BUREAU_MON	Number of enquiries to Credit Bureau about the client one month before application (excluding one week before application)
AMT_REQ_CREDIT_BUREAU_QRT	Number of enquiries to Credit Bureau about the client 3 month before application (excluding one month before application)
AMT_REQ_CREDIT_BUREAU_YEAR	Number of enquiries to Credit Bureau about the client one day year (excluding last 3 months before application)

You can use the **training** dataset (but not validation) for training machine learning models, and you can use the test dataset to evaluate your solutions and avoid over-fitting.

Please Note:

- This assignment specification is deliberately left open to encourage students to submit innovative solutions.
- You can only use Scikit-learn to train your machine learning algorithm
- Your model will be evaluated against a different set of datasets (available for tutors, but not for students)
- You must submit your code and a report
- The due date is **22/04/2022 at 20:00**

Part-I: Regression (10 Marks)

In the first part of the assignment, you are asked to predict the client's "income" based on the information provided in their loan application. More specifically, you need to predict a client's income based on columns (or any subsets) provided in the dataset except for AMT_INCOME_TOTAL, which you are predicting.

- The minimum requirement for ***Correlation for this part is 0.20*** on the final test dataset (the dataset will not be public, and will be used by tutors to test your models- so do not try to overfit your models on the provided datasets)

- You should analyze and select features they think would improve your machine learning models (and filter out those that may not). You can also combine multiple features and create new ones.

Part-II: Classification (10 Marks)

Using the same datasets, you must predict if a loan application should be approved or not. For this part, you can use all columns (or any subset) of the dataset except "TARGET", the column that you are going to predict.

- The minimum requirement for ***Accuracy for this part is 0.85*** on the final test dataset (the dataset will not be public, and will be used by tutors to test your models- so do not try to overfit your models on the provided datasets)

- You should analyze and select features they think would improve your machine learning models (and filter out those that may not). You can also combine multiple features and create new ones.

Submission

You must submit two files:

- A python script z{id}.py
- A report named z{id}.pdf

Python Script and Expected Output files

Your code must be executed in CSE machines using the following command with three arguments:

```
$ python3 z{id}.py path1 path2
```

- **path1** : indicates the path for the dataset which should be used for training the model (e.g., ~/training.csv)
- **path2** : indicates the path for the dataset which should be used for reporting the performance of the trained model (e.g., ~/test.csv); we may use different datasets for evaluation

For example, the following command will train your models for the first part of the assignment and use the test dataset to report the performance:

```
$ python3 YOUR_ZID.py training.csv test.csv
```

Your program should create 4 files on the same directory as the script:

- z{id}.PART1.summary.csv
- z{id}.PART1.output.csv
- z{id}.PART2.summary.csv
- z{id}.PART2.output.csv

For the first part of the assignment:

" z{id}.PART1.summary.csv " contains the evaluation metrics (MSE, correlation) for the model trained in the first part of the assignment. Use the given validation dataset to compute the metrics. The file should be formatted exactly as follow:

```
zid,MSE,correlation
z123456,6.13,0.53
```

- **MSE** : the mean_squared_error in the regression problem
- **correlation** : The **Pearson correlation coefficient** in the regression problem (a floating number between -1 and 1)

" z{id}.PART1.output.csv " stores the predicted revenues for all of the movies in the evaluation dataset (not the training dataset), and the file should be formatted exactly as:

```
SK_ID_CURR,predicted_income
1,178000
2,256000
...
```

For the second part of the assignment:

" z{id}.PART2.summary.csv " contains the evaluation metrics (average_precision, average_recall, accuracy - the unweighted mean) for the model trained in the second part of the assignment. Use the given validation dataset to compute the metrics. The file should be formatted exactly as:

```
zid,average_precision,average_recall,accuracy
z123456,0.69.71,0.89
```

- **average_precision** : the average precision for all classes in the classification problem (a number between 0 and 1)
- **average_recall** : the average recall for all classes in the classification problem (a number between 0 and 1)

" z{id}.PART2.output.csv " stores the predicted ratings for all of the movies in the test dataset (not the training dataset) and it should be formatted exactly as follow:

```
SK_ID_CURR,predicted_target
1,1
2,0
...
```

Marking Criteria

You will be marked based on:

- **(4 marks)** Your code must run and perform the designated tasks on CSE machines without problems and create the expected files. Your submission will be penalized up to 50% if is not able to create the output files.
- **(8 marks)** How well your model (trained on the training dataset) performs in the test dataset (a different dataset not available for students - will be used for fair marking)
  A submission will get 0 if it does not pass the advertised baselines (minimum requirements). Tutors will judge how good are your models in each part of the assignment and give marks accordingly.
- **(3 marks)** You must correctly calculate the evaluation metrics (e.g., average_precision - 2 decimal places) in the output files (e.g., z{id}.PART2.summary.csv)
- **(5 marks)** A report
  - You should provide a report, containing your analysis of the dataset which helps you in the feature engineering of your machine learning models. For this, you must use Jupiter Notebook and export it as a PDF (https://towardsdatascience.com/jupyter-notebook-to-pdf-in-a-few-lines-3c48d68a7a63) file. Add comments in your notebook describing what are you concluding for each of your analyses. Use chars and any skill you have learnt in the course to support your decisions about features used in your ML models.
- The late penalty is 5% per day, and submissions after day 5 will not be marked.
- You will be penalized (1 mark per minute) if your models take more than 3 minutes to train and generate output files.
- Your assignment will not be marked (zero marks) if any of the following occur:
  - If it generates hard-coded predictions
  - If it also uses the second dataset (test/validation) to train the model
  - If it does not run on CSE machines with the given command (e.g., python3 zid.py training_dataset.csv test_dataset.csv)
    Do NOT hard-code the dataset names

FAQ

- **Can we define our own feature set?**
  Yes, you can define any features; make sure your features do not rely on the test datasets.
- **For the average precision/recall functions, should we use the unweighted ('macro') mean or the weighted mean?**
  Use the unweighted ('macro') mean

- **Should we calculate metrics to 1 Decimal Place?**
  2 Decimal Places
- **Can we use any machine learning algorithm?**
  Yes, as long as it is provided in sklearn.
- **What python modules can we use for developing our solutions?**
  You can use any modules presented in the lab activities; otherwise, you may get permission by asking ...
- **How should we calculate the Pearson correlation coefficient?**
  It is calculated between your predictions and the real values for the test dataset.
- **Will I get penalized for "Warnings" thrown by my code?**
  No, you will not get penalized

Plagiarism

This is an *individual assignment* . The work you submit must be your own work. Submission of work partially or completely derived from any other person or jointly written with any other person is not permitted. The penalties for such offense may include negative marks, automatic failure of the course, and possibly other academic disciplines. Assignment submissions will be checked using plagiarism detection tools for both code and the report and then the submission will be examined manually.

Do not provide or show your assignment work to any other person - apart from the teaching staff of this course. If you knowingly provide or show your assignment work to another person for any reason, and work derived from it is submitted, you may be penalized, even if the work was submitted without your knowledge or consent. Pay attention to that is **also your duty to protect your code artifacts** . if you are using an online solution to store your code artifacts (e.g., GitHub) then make sure to keep the repository private and do not share access to anyone.

Reminder: Plagiarism is defined as (https://student.unsw.edu.au/plagiarism) using the words or ideas of others and presenting them as your own. UNSW and CSE treat plagiarism as academic misconduct, which means that it carries penalties as severe as being excluded from further study at UNSW. There are several online sources to help you understand what plagiarism is and how it is dealt with at UNSW:

- Plagiarism and Academic Integrity (https://student.unsw.edu.au/plagiarism)
- UNSW Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

Make sure that you read and understand this. Ignorance is not accepted as an excuse for plagiarism. In particular, you are also responsible for ensuring that your assignment files are not accessible by anyone but you by setting the correct permissions in your CSE directory and code repository, if using one (e.g., Github and similar). Note also that plagiarism includes paying or asking another person to do a piece of work for you and then submitting it as your own work.

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW staff and students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW.

Resource created 10 days ago (Monday 28 March 2022, 10:50:39 AM), last modified about 23 hours ago (Wednesday 06 April 2022, 02:05:07 PM).

Comments

(/COMP9321/22T1/forums/search?forum_choice=resource/74265)

(/COMP9321/22T1/forums/resource/74265)

Add a comment

Tanya Hollis (/users/z5323560) about an hour from now (Thu Apr 07 2022 14:00:44 GMT+0800 (香港标准时间))

Hi tutors &/or Morty, I have 2 questions...

1) Regarding marking criteria "You will be penalized (1 mark per minute) if your models take more than 3 minutes to train and generate output files.".... On what kind of machine are you marking? If we are developing on a fast gaming enabled computer, and we carry out full testing at uni on a "good day" (low number of students) and the whole output is generated in well under 3 mins. Will we be marked down if the marker chose to mark on University machines on a "bad day" and it takes over 3mins? Should we retain timing "proof" that our assignments can run under 3mins on UNSW machines?

2) The assignment specification refers to a validation set. Should we be splitting off some of our training data for validation purposes? Or should we just assume by "validation" you mean "test"?

Reply