CS5228 Final Project

Due date: 16 April 2023 11.59pm

Instructions

This final project involves two options, to give you the flexibility to either (1) apply your data mining skills on a practical real-world task; or (2) explore an area of your interest through an open-ended project, which can be in a domain of your choice.

If you have any questions, feel free to either post your questions on the Canvas forums, or email myself (bhooi@comp.nus.edu.sg), or He Xiaoxin (he.xiaoxin@u.nus.edu) who is the TA in charge of this project.

Option A: Kaggle Competition: Location, Location, Location?

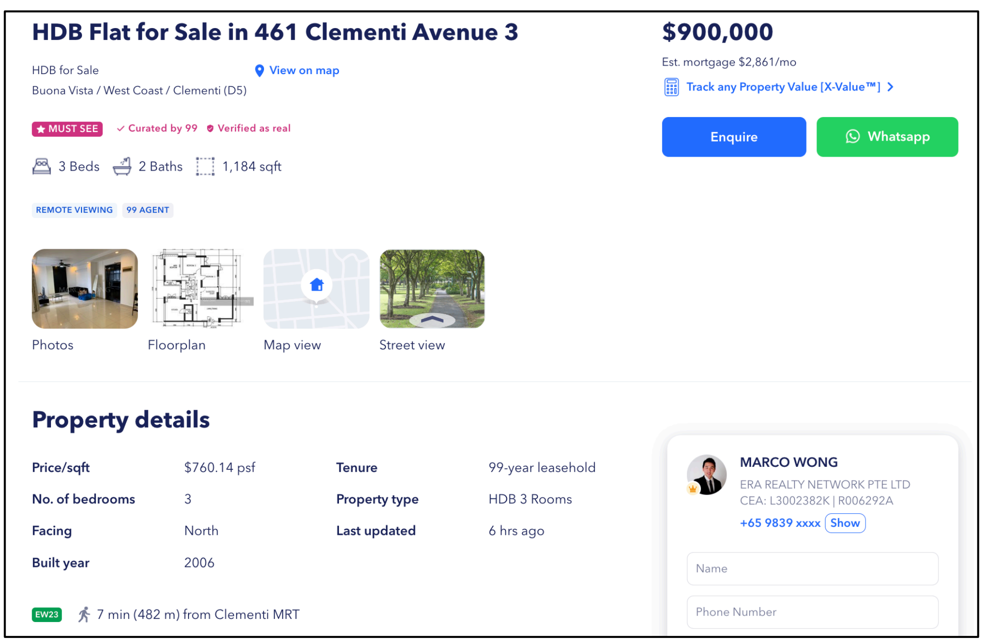

Figure 1: HDB listing on 99.co.

The goal of this project is to give you experience in a real-life regression task, which is one of the most practical and commonly encountered type of data-related task. It will give you experience applying approaches such as exploratory data analysis, visualization, preprocessing, and how to select and evaluate various data mining approaches.

The task is to predict the resale price of HDB flats. A full description of the dataset and evaluation metrics can be found at the Kaggle InClass competition page. The dataset comes from data.gov and 99.co, with around 20 attributes (such as asking price, property type, address, etc.)

In this project, we look into the Singapore housing market. Home ownership in Singapore is rather expensive, making buying a home one of the most significant financial decision for most people in their lives. Buyers therefore want to know what they can get for their money, where they can best save money, and simply spot bargains rip-offs. On the other hand, sellers and real estate agents aiming to maximize prices want to know how to best present and advertise their properties. Also, with the limited amount of available land, affordable housing is a major issue of the Singaporean government, which may deploy ”cooling measures” to influence the housing market. In short, there are many stakeholders that rely on and benefit from a deeper understanding of the Singaporean housing market.

Submitting your Predictions on Kaggle

The Kaggle page (https://www.kaggle.com/t/f88bbbd685514d4aa0a2cf161a699e6e) contains training data `train.csv`, with the associated labels in the form of the “*resale_price*" variable. The test data `test.csv` contains the test set which your model should predict. The predictions you submit should be a csv file: for reference, see the sample predictions file `sample_output.csv`, which has been included in the dataset package, and shows what your submission should look like. Additionally, considering the importance of location and nearby amenities, an auxiliary data file `auxiliary_data.zip` has also been provided, which contains some csv files with additional data like the locations of MRT stations, shopping malls, schools, etc. It is up to you if and how you want to consider integrating this data for training your model.

To prevent overfitting to the leaderboard, you are allowed to make at most 5 submissions to the leaderboard per day. Note that you are expected to write your own procedures for evaluating how well your method is performing using the given datasets: for example, this can be done by cross-validation, splitting the data into a training and a validation set, or similar approaches. Do not mainly rely on the public leaderboard to evaluate your model: this could potentially overfit to the public leaderboard, resulting in poor performance on the private leaderboard, and the submission limit would hinder your model development process; instead, you should utilize your own evaluation procedures such as cross-validation (see the “Evaluation using Cross-Validation” link in the Helpful Resources section below for more details). We will place greater consideration on the private rather than the public leaderboard scores.

Final Report

One member of your team should submit a zipped folder to Canvas (under "Assignments") containing your final report (in PDF format) describing your data preprocessing steps, exploratory data analysis, the steps you have done in model fitting and evaluation, and the evaluation results (include both your public leaderboard score, as well as any evaluation metrics you have computed on your own to test and compare different approaches.) The report should be at most 8 pages, including figures. The zipped folder should also contain the code you use in your approach, or include a link to your github repository in your final report. Note that 10% of your grade will be based on the reproducibility of your approach, which includes the organization and readability of your code. Your report should include the names and student IDs (e.g. e1234567) of all your team members, and your Kaggle team name.

While the exact format of the report is up to you, you can structure it based on the following:

- **Motivation:** Motivate and outline the goals and questions you address.
- **Preprocessing and Exploratory Data Analysis**: Explain how you tried to understand the data, e.g. through plotting or exploratory data analysis. Explain all preprocessing steps taken; e.g. what features did you use, and did you perform any feature transformations, and why were these transformations done?
- **Data Mining**: Clearly describe how you ran your model(s). You do not need to explain the model itself (e.g. what a random forest is), but should explain all other relevant details for running it. Which approaches did you try? What values did you use for its hyper-parameters (or other choices involved in the model)? How did you select these hyper parameters?
- **Evaluation and Interpretation**: How did you compare between different approaches? Which approaches performed the best, and why?

There is no strict requirement for format, and many ways to write a good report, as long as it is clear and comprehensive; an example of a good report (from a previous semester of the class) can be found in the Canvas folder for this final project. Note that this was based on a different dataset, with its own characteristics, so the steps involved are probably not appropriate for the current dataset and task. Also, there is no need to follow the formatting of this report, as long as you have described your approach clearly and fully.

Grading

Grading will be primarily based on the quality of the methodological approach and the final report – i.e. whether your approach is methodologically sound, whether you can understand and clearly describe each step of the data mining process, and whether you are able to build an effective model (and clearly explain the steps you have taken to do so).

Your scores on the public and private leaderboards will be considered only used as part of the whole picture, i.e. as one indication of the quality of your methodology (which will also be assessed based on your report). Scoring well on the leaderboard is not a requirement for getting very good scores - it is possible to do very well as long as your overall approach as detailed in the report is methodologically sound and of high quality, and meets a reasonable standard of prediction accuracy, even if it does not get top positions on the leaderboard. The main purpose of the competition is to provide a relatively objective measure of methodological effectiveness (for example, methods which score significantly below that of simple baselines may indicate some issues with methodology), and to hand out bragging rights to the top competitors.

Grading will be based on:

1. **Methodological quality: 60%**
    a. **Preprocessing:** appropriate preprocessing methods are chosen and correctly implemented; e.g. missing values and categorical variables are handled appropriately
    b. **Visualization**: appropriate and informative use of visualization and plots, leading to good data understanding
    c. **Methods**: methods are well motivated and correctly implemented, in a methodologically sound manner
    d. **Evaluation**: methods are compared or evaluated in an effective manner, with the use of appropriate metrics, and with appropriate experimental setups (e.g. cross-validation, splitting into training and testing sets, or other approaches)

2. **Quality of report**: **30%**
    Report explains the results in a clear and comprehensive manner, demonstrating and communicating correct understanding of the various steps you have done

3. **Reproducibility: 10%**
    Code is included, and is sufficiently well-organized and readable so as to be usable by an outsider

Helpful Resources

Some useful resources include the following:

1. **Getting Started Guide on Kaggle**: https://www.kaggle.com/getting-started/45113

2. **Exploratory Data Analysis**: https://www.kaggle.com/kashnitsky/topic-1-exploratory-data-analysis-with-pandas

3. **Evaluation using Cross Validation**: https://www.kaggle.com/dansbecker/cross-validation: Cross validation (or similar tools) are important in evaluating your approach to see which of various approaches works better. Note that you should not rely solely on the public leaderboard for this, as you are only allowed 5 submissions per day (and also, this may overfit to the public leaderboard, which would affect your score on the private leaderboard).

4. **Avoiding Data Leakage**: https://www.kaggle.com/dansbecker/data-leakage/: Data leakage is a commonly encountered problem on Kaggle (and similar settings). Informally, for test accuracy to be meaningful, the algorithm being tested should generally not be trained on information from the test data. Data leakage means that there is "leakage of information" from your test set to your training set, which makes test accuracy an unreliable metric. This can happen in subtle ways, e.g. when the training and test set columns are preprocessed together, such as through normalization.

5. **Additional Learning Materials on Kaggle:** see https://www.kaggle.com/dansbecker/learning-materials-on-kaggle for a comprehensive and useful list of resources, e.g. on handling categorical data.

Frequently Asked Questions

- **What size groups are allowed?**

Group size is 3-4. If you work in groups, only one member needs to perform the final submission, but please include all your names in the final report.

- **How does the leaderboard on Kaggle work?**

The test set has been partitioned randomly: 35% into a **public test set,** and 65% on a **private test set**. When you upload your predictions, Kaggle computes the accuracy of your predictions against the public test set, which is used to determine your score on the leaderboard. At the end of the competition, your score on the private test set will also be revealed.

To prevent overfitting on the public test set, Kaggle allows you to make up to 5 submissions per day.

At the end of the competition, Kaggle will allow you to choose your 3 preferred submissions, which will be used for evaluation on the private leaderboard (Kaggle defaults to using your top scoring submissions on the public leaderboard).

- **What kind of additional resources am I allowed to use or refer to?**

To keep the leaderboard fair, please do not make use of code which performs modelling specifically on this dataset, or refer to external sources which perform analysis on this dataset. Note that we have made some minor modifications to the dataset to make it harder to directly apply existing code which is designed for the original dataset.

Other than the above, you may use any available software libraries and APIs. For **python**, commonly used packages include `pandas` for data processing, `matplotlib` and `seaborn` for data exploration and plotting, `scikit-learn`, `xgboost`, `catboost` and `lightgbm` for modelling. For **R**, commonly used packages include `dplyr` for data processing, `ggplot2` for data visualization, and `caret`, `randomForest`, `gbm`, `xgboost`, `catboost` and `lightgbm` for modelling. These are the 2 most commonly used languages with rich libraries, but you are allowed to use other languages if you are familiar with them. Moreover, you are allowed to refer to resources as long as they are not specifically designed for our dataset: for example, you can refer to stackoverflow, guides explaining how to perform exploratory data analysis, cross validation etc.

- **Will we cover all the needed regression / classification approaches in class?**

While we will cover some commonly used approaches in class (e.g. regression, random forests, etc.), you should expect to have to do a significant amount of learning on your own, particularly in the practical and implementation aspects. Feel free to email the course staff if you have any questions. For a set of tutorials to read, please refer to the list of helpful resources above for this project.

Option B: Open-Ended Project

The goal of this project is to allow you to explore an area of your interest in the form of a data analysis (or data-related) project. For example, if you are currently working in a particular industry area, or doing research within a particular academic area, it is advisable to pursue a topic in those areas. This could take the form of performing some data analysis on a dataset from your domain of interest, or proposing a new method relevant to data mining on a particular type of data. If you are working on a fairly new dataset, please start by making sure the dataset is available and clean enough for analysis – it is likely not optimal for most of your effort to be spent on acquiring or cleaning the dataset. Please talk to the course staff if you are unsure with regard to any potential topics.

You are encouraged to work in groups of 3-4. If you work in groups, only one member needs to perform the final submission, but include all your names in the final report. In special cases, depending on the scope and goals of the particular project, larger groups may be considered, but please talk to the course staff first.

Final Report

One member of your team should upload to Canvas (under "Assignments") a zipped folder containing your final report (in PDF format) of at most 8 pages, including figures. If applicable, your zipped folder should also include the code you use in your approach, or include a link to your github repository in your final report. Include the names and student IDs (e.g. e1234567) of all your team members in the report.

While the format of the report has flexibility depending on the project focus, the following are some examples of possible structure. For a data analysis project, you could structure it as:

- **Introduction / Motivation**: explain why the problem is important and needs to be solved.
- **Background**: if applicable, explain what other work has been done in this area.
- **Dataset**: describe the dataset clearly, i.e. its dimensions, what its variables mean, etc.
- **Findings**: perform exploratory data analysis on the dataset and explain your analysis steps and findings.
- **Conclusion**: summarize your main findings, provide your interpretation, and explain their implications (if appropriate)

For a project focused on proposing a new method, you could structure it as:

- **Introduction / Motivation**: explain why the problem is important and needs to be solved.
- **Background**: explain what other work has been done in this area.
- **Method**: clearly explain the intuition for your method and how your method works.

- **Experiments**: explain experiments to show how your method improves on simpler existing methods.
- **Conclusion**: summarize your main findings, provide your interpretation, and explain their implications (if appropriate)

Grading

Grading will be based on:

1. **Methodological quality: 60%**
    a. **Preprocessing:** appropriate preprocessing methods are chosen and correctly implemented; e.g. missing values and categorical variables are handled appropriately
    b. **Visualization**: appropriate and informative use of visualization and plots, leading to good data understanding
    c. **Methods**: methods are well motivated and correctly implemented, in a methodologically sound manner
    d. **Evaluation**: methods are compared or evaluated in an effective manner, with the use of appropriate metrics, and with appropriate experimental setups (e.g. cross-validation, splitting into training and testing sets, or other approaches)

2. **Quality of report**: **30%**
    a. Report explains the results in a clear and comprehensive manner, demonstrating and communicating correct understanding of the various steps you have done

3. **Reproducibility: 10%**
    a. Code is included, and is sufficiently well-organized and readable so as to be usable by an outsider

Due to the flexibility of the project, in some cases, some categories may be adjusted by considering the goals of the specific project, and evaluating the extent to which these goals have been achieved (taking into consideration how ambitious the goals were).

Helpful Resources

Some useful resources include the following (credit to Srijan Kumar / Georgia Tech CSE 6240, and Stanford CS224N, CS224W and CS341 classes):

Dataset repositories:

1. Kaggle Public Datasets https://www.kaggle.com/datasets

2. Subreddit of Datasets https://www.reddit.com/r/datasets/
3. Google Dataset Search https://toolbox.google.com/datasetsearch
4. Google Public Datasets https://www.google.com/publicdata/directory
5. Github page of Public Datasets https://github.com/awesomedata/awesome-public-datasets
6. Large Datasets publicly available https://www.quora.com/Where-can-I-find-large-datasets-open-to-the-public
7. European Union Open Data Portal https://data.europa.eu/euodp/en/data/
8. US Healthcare Data https://healthdata.gov/
9. Microsoft Open Datasets https://msropendata.com/
10. Singapore Open Datasets https://data.gov.sg

Sample Papers and Class Projects:

1. [Paper] LUNAR: Unifying Local Outlier Detection Methods via Graph Neural Networks
2. [Paper] Flashlight: Scalable Link Prediction with Effective Decoders
3. [Paper] MIDAS: Microcluster-Based Detector of Anomalies in Edge Streams
4. [Paper] Graph Neural Network-Based Anomaly Detection in Multivariate Time Series
5. Financial News in Predicting Investment Themes
6. Sarcasm Detection
7. Humor Classification on Yelp reviews
8. Detection and Analysis of Hateful Users on Twitter
9. Fake News detection using Machine Learning on Graphs
10. Fraud Detection in Bitcoin Networks
11. Weighted Signed Network Embeddings
12. Anomaly Detection of Computer Health

Other examples of possible project topics can be found at the following class webpages:
- http://web.stanford.edu/class/cs341/projects.html
- http://snap.stanford.edu/class/cs224w-2019/projects.html
- https://web.stanford.edu/class/cs224w/projects.html
- https://nlp.stanford.edu/courses/cs224n/

Other Additional Datasets:

- (Finance) https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting/data
- (Movies) https://www.reddit.com/r/datasets/comments/b4yy6p/480000_rotten_tomato_critic_reviews/
- (Airbnb) https://www.kaggle.com/c/airbnb-recruiting-new-user-bookings
- (Enron Emails) https://www.cs.cmu.edu/~enron/
- (Music) https://components.one/datasets/billboard-200/