BIRMINGHAM CITY
University

CMP6202 Report

DRAFT VERSION

Lewis Higgins - Student ID 22133848

Word count: XXXX

Contents

Introduction

Your EDA can be very extensive, and you could potentially have pages and pages and pages of it; this isn't a bad thing. The vast majority of any ML-related work is EDA because it gives you the background information on the dataset to then apply when training the model, such as the identification of non-numerical columns and encoding them into numerical equivalents where possible so that they become useful training data for the model, as ML models cannot interpret strings.
You should work with at least two ML algorithms. Compare them, pros/cons, why you settled on the one you did. F1-Score if you do a classification problem, otherwise r^2. A template exists for both the presentation and report. **You need to pay close attention to section 6.2, as he says it'll be at least 10%**. He mentions things like certificates, a linkedin profile and more can be relevant to this section. Sounds less like "individual learning reflection" and more "individual reflection".
You might be missing some LaTeXpackages, consult your lit review for the ones you might need. This also applies to the CMP6230 report.
Kaggle can give you certificates, as can GitHub. With Github, it could be advisable to make this repo public, or perhaps a sanitised duplicate.

Notes, remove before final

1.1 Week 7

1.1.1 Naive Bayes

This symbol | means "given that". Sentiment analysis, this could be linked to your dissertation too. Used in classification. After the formula, the larger number is chosen as the class (i.e. if spam scored 0.05 but good showed 0.12, good wins). Gaussian NB is best used for continuous data and text data classification. The prior probability is calculated by dividing the number of positive instances by the total number of instances. When classifying spam emails, the probability of the email being spam is the prior probability. Probability of a given event (A) given that (|) a previous event has occurred (B)

Cross Validation

Train test split has some issues, due to its random selection of data to put in each split, which are addressed by cross validation. Cross validation helps to reduce bias in an ML model.

K-Fold

Divides the dataset into "K" subsets/folds, then trains and evaluates the model "K" times, using a different fold as the test set while the others act as the test data.

Stratified K-Fold

A variation of K-Fold that ensures class balance, helping where the dataset is imbalanced. **Could be very relevant.** Week 7 PPT slides 17 through 20 help with this and regular K-Fold.

Accuracy score

Your accuracy score will differ from the cross-validation score because it doesn't account for the averaged and varied subsets used in cross-validation.

1.1.2 Decision tree

Another classification method, currently don't know much about it. Initialised in code via DecisionTreeClassifier().

1.2 Mihai W7

1.2.1 Overall notes - Beginning of talk

- Find a problem before a dataset (No)
- Identify if said problem is regression or classification
- It's hard to use a classification dataset where the target column is just a bunch of labels (???)
  - This may be in reference to Week 7's wine set where you had 7 different levels of quality but converted them to 2 for binary classification.

- Examples on Moodle
- Datasets on Moodle but it's almost 100% that someone else will have used them.
- I also don't know if you're even allowed to use them.
- A dataset can have more than one target column (though the ones you use likely won't).
-

1.2.2 Section 2

- After you have an identified dataset, begin EDA.
- Describe how you split the dataset into training and testing dataset.
- If the machine sees the test data, it may "overfit".
  - You've seen this before where the line of best fit isn't really a line and just connects every single point, meaning it's really good at guessing the data it already knows but not new data.
- You split your data *BEFORE* EDA to avoid "conceptual overfitting".
- When splitting data, you need to think about data imbalance i.e. training set having too much of the one option and few of the other (too many True, not enough False).
- SKLearn may try to fix that for you.
- Identify outliers, missing data.
- "Missing information can be information in itself". Consider the source of the data (not Kaggle but rather where the Kaggle author got it from)
  - You might be able to impute data rather than deleting it.
  - Conserve as much data as you can, deleting data should be a last option.
- Erroneous data
  - Another reason why your data source is important.
  - Could just be mistyped, see what the erroneous data actually is to see if you can correct it.
- Outliers
  - Is it significant enough to remove it? Is it definitely an outlier; could it feasibly be true? (speed camera example where one guy went 30 but another went 100, but that's still plausible.)
  - Boxplots can identify outliers.
- If your dataset is bad, your model will be, too.

1.2.3 Section 3

- Identify the right algorithm.
  - Decision trees are good at classification.
  - Random forest is also used for it.
  - Naive Bayes and KNN work for prediction and classification.
  - Some of these may perform worse with higher amounts of data, look into them.
- Performance won't matter a massive amount but you still need to be able to justify why it was good.
  - Also justify its downsides and limitations.
  - And the limtiations of the dataset itself.

1.2.4 Section 4

- Encoding is important here because ML doesn't use text.
  - Side-note: Even if it does that's actually just an abstraction and it's just encoding it under the hood.
- Fine-tuning
  - Playing around with parameters of the functions.
  - KNN Neighbours and such
  - Often comes after an initial test run
  - If your accuracy is really low (20% was the example), fine-tuning won't help and you just need to redo the entire work.
  - Could maybe give you an extra 5% accuracy.
  - Decision trees may have multiple versions. SKLearn's decision tree may be different from a CUDA one.
- Evaluation metrics
  - Accuracy is not Precision. Which do you need?
  - You want both to be high, if one lags massively behind the other then that's bad.
  - ML is an iterative process until you can get the best performing model.

1.2.5 Section 5

- Visualisation of evaluation results
  - For a correlation matrix, a heatmap is better than a bar plot for example.

1.2.6 Presentation

When writing this presentation, there's 3 questions at all times. What, why and how? What are you doing, why are you doing it, how are you doing it? Why is very important - why did I select my problem? Why do I think it's important.

Key note

You need quite a substantial amount of this actually done for **November 29th** rather than Dec 20th as the deadline states. This is because the presentation relies upon this report.