CS5228 – Tutorial 1

Data, Data Cleaning, Data Preprocessing

Figure 1 shows the first 20 data samples from the Students Performance in Exams dataset freely available on Kaggle. However, for the purpose of this tutorial, we tweaked it a little bit. After all, we don't really perform any analysis on that data here. Although only 20 samples are shown, just by looking at them, we can already gain some very basic insights that can and/or should guide any subsequent processing steps. As additional information, there are a total of 1,000 samples in the dataset.

id	gender	ethnicity	state	parent_education	email	sleeping_hours	prep_course	math_score	read_score	write_score
1	female	group b	MO	Bachelor's Degree	xxxxx	7 to 8	none	72.0	72.0	0.74
2	female	group c	VA	Some College	xxxxx	< 6	completed	69.0	90.0	0.88
3	female	group b	PR	master's degree	xxxxx	7 to 8	none	90.0	95.0	0.93
4	male	group a	CT	associate's degree	xxxxx	7 to 8	none	47.0	57.0	0.44
5	male	group-c	UM	some college	xxxxx	8 to 9	none	76.0	78.0	0.75
6	female	group b	OK	Associate's Degree	xxxxx	6 to 7	none	71.0	83.0	0.78
7	female	group-b	PA	Some College	xxxxx	6 to 7	completed	88.0	95.0	0.92
8	male	group-b	GA	Some College	xxxxx	> 9	none	40.0	43.0	0.39
9	male	group d	LA	high school	xxxxx	6 to 7	completed	64.0	64.0	0.67
10	female	group b	KY	High School	xxxxx	7 to 8	none	38.0	60.0	0.50
11	male	group c	AZ	Associate's Degree	xxxxx	< 6	none	58.0	54.0	0.52
12	male	group d	OK	associate's degree	xxxxx	8 to 9	none	40.0	52.0	0.43
13	female	group-b	OR	High School	xxxxx	7 to 8	none	65.0	81.0	0.73
14	male	group-a	SD	some college	xxxxx	6 to 7	completed	78.0	72.0	0.70
15	female	group a	KY	master's degree	xxxxx	8 to 9	none	50.0	53.0	0.58
16	female	group-c	CO	Some High School	xxxxx	< 6	none	69.0	75.0	0.78
17	male	group-c	UM	high school	xxxxx	7 to 8	none	88.0	89.0	0.86
18	female	group b	KY	Some High School	xxxxx	> 9	none	18.0	32.0	0.28
19	male	group-c	WI	Master's Degree	xxxxx	6 to 7	completed	46.0	42.0	0.46
20	female	group-c	NE	associate's degree	xxxxx	6 to 7	none	54.0	58.0	0.61

Figure 1: 20 Samples of Students Performance in Exams (modified).

Data preparation. The following questions are not "exam questions" as there can be quite some room for discussion with no single correct answer. This is particularly true since (a) we only see a small snippet of the whole dataset, (b) we have no means to perform an EDA beyond eye-balling the data, and (c) we did not specify the exact data mining task we want to solve.

1. **Types of Attributes.** For each attribute, decide whether it is *nominal*, *ordinal*, *interval*, or *ratio*. For which attributes might this decision not be so clear?

Solution:

- id: nominal
- gender: nominal
- ethnicity: nominal
- state: nominal
- parent_education: ordinal
- email: nominal
- sleeping_hours: ordinal/ratio
- prep_course: nominal
- *_score: ratio

As strings, the values for sleeping_hours can be considered ordinal, but strictly speaking, hours is a numerical measure. So sleeping_hours can be processed to result in numerical values, making this a ratio attribute. Also, whether an ordinal attribute such as parent_education might better be considered only a nominal attribute may depend on the exact task or other constraints.

2. **Data Cleaning.** Just by looking at these 20 samples, which data cleaning steps seem recommended? Note that this refers only to preprocessing steps to remove potential noise from the dataset, not any steps that might further benefit a subsequent analysis (arguably, there is no clear distinction, but we cover these steps in the next questions).

Solution:

- Normalize ethnicity (e.g. "group c" vs "group-c")
- Normalize parent_education (e.g., convert to all lowercase)
- Adjust scale of write_score

3. **Attribute/Feature Importance.** For each attribute, assess its importance (or relevance, usefulness, etc.) for a subsequent analysis. Let's assume we want to predict students' math, reading, and writing scores based on the other attributes.

Solution:

- remove id (just an "artificial" attribute)
- remove email (it's redacted anyway, and so of no use)
- probably remove state (we only have 1,000 sample and there a 50+ states, so the information w.r.t. students' state is very sparse and far from representative)

4. **Additional Data Preprocessing.** Many to most off-the-shelf data mining algorithms for clustering or classification/regression require numerical data as input. How does this affect the analysis of this dataset, and what can we do to address this using data preprocessing?

   (For this question, you are encouraged to look up alternative encoding strategies for converting categorical attributes into numerical ones. In the lecture, we only covered one-hot encoding for nominal attributes. However, there are other strategies for nominal and ordinal data, which can be useful for the project. Also, check and appreciate the pros and cons of different strategies.)

Solution:

- gender and prep_course seem to be both binary attributes, so 0/1 encoding should do just fine
- parent_education can be converted into a simple ranking of numerical values (e.g., 1, 2, ...). Many tools support **ordinal encoding** but users have to ensure that the labels reflect to semantic order of the original values (otherwise Master's Degree might be 0, and High School might be 1)
- state is definitely nominal, so we need to encode it. **One-hot encoding** is always applicable but that would result in 50+ new attributes that would also be very sparse. For classification/regression task, we could go with **target encoding** where we replace each state value with the mean of the output values over all sample with that state value (e.g., we replace "MO" with 70.3, assuming that's the mean of all math course of tuples with the state "MO"). But again, the data for state is very sparse, so it's really better to just ignore this attribute. Alternatively, maybe a generalization approach might be interesting (e.g., mapping all states to Democratic and Republican, yielding a convenient binary attribute). However, target encoding is not uncontroversial as it allows for *"data leakage"*. Simply speaking, we create

a feature based on the target, which is a kind of cheating. We look into the issue of data leakage a bit more when we cover the topics classification and regression.

- converting the sleeping_hours strings into a numerical estimate would probably be useful.

Optional "homework". In the lecture, we focused on the common challenges when working with real-world data, e.g.: noise, outliers, inconsistencies, missing values, misleading default values, imbalanced class labels, etc. However, there is also the potential case where the data was intentionally manipulated or tampered with. The purpose of such tampering is typically (a) to "inject" a pattern into the data that was not there or (b) to remove or obfuscate a pattern in the original data. Of course, both cases are highly unethical. Have a look at the website of DataColada. This blog is run by a group of researchers to identify and find evidence for data fraud, and they uncovered some very high-profile cases.