Data Analysis 1

Class 2

Variable types:

unique: time

* **Quantitative**
  + numeric value
  + take many values
* **Qualitative**
  + categorical variable
  + specific interpretation
  + types:
    - **categorical/factor**
      * **nominal** (Milka, Tibit)
        + Nominal qualitative variables take on values that cannot be unambiguously ordered.
      * **Ordinal** (ratings)
        + Ordinal, or ordered variables take on values that are unambiguously ordered. All quantitative variables can be ordered; some qualitative variables can be ordered, too.
    - **interval –** time intervals (commuting times: 0-10 min)
      * Interval variables are ordered variables, with a difference between values that can be compared.
    - **ratio/scale**: Ratio (=scale) variables are interval variables with the additional property: their ratios mean the same regardless of the magnitudes. This additional property alsoimplies a meaningful zero in the scale.
      * discrete variable : age (32 yo)
      * continuous (price)
* **binary:** 0/1 (Yes/No)

Data wrangling (data munging):

Data wrangling is the process of transforming raw data to a set of data tables that can be used for a variety of downstream purposes such as analytics.

1. understanding and storing:
   1. start from raw data
   2. understand the structure and content
   3. create tidy data tables
   4. understand the links between tables
2. Data cleaning
   1. understand features, variable types
   2. filter duplicates
   3. look for and manage missing obs.s
   4. understand limitations

The tidy data approach:

tidy table is tidy if every observation seperated uniquely

A useful concept of organizing and cleaning data is called the tidy data approach:

1. Each observation forms a row.

2. Each variable forms a column.

3. Each type of observational unit forms a table.

4. Each observation has a unique identifier (ID)

Advantages:

* standard data tables that turn out to be easy to work with.
* finding errors and issues with data are usually easier with tidy data tables
* transparent, which helps other users to understand
* easy to extend. New observations added as new rows; new variables as new columns.
* easy to understand

Relational database:

multiple data

* The relational database is a concept of organizing information.
* It is a data structure that allows you map a concept set of information into a set of tables
* Each table is a made up of rows and columns
* Each row is a record (observation) identified with a unique identifier ID (also called key).
* Rows (observations) in a table can be linked to rows in other tables with a column for the unique ID of the linked row (foreign ID)
* Define these tables, understand structure
* Merge tables when needed

Matching:

Matching (joining) depends on data structure

* **one-to-one (1:1)** matching: merging tables with the same type of observations.
  + Football teams and stadium now.
* **many-to-one (m:1)** or one-to-many (1:m) matching when in one of the data tables, a value may be matched to more than one values in the other table.
  + Football teams and their players now - many players in a team.
* **many-to-many (m:m)** matching when values in both tables could be matched to many others.
  + Football teams and their manager ever - some managers worked for multiple teams

**Data cleaning:**

Entity resolutaion:

* Duplicates: some observations appearing more than once in the data.
* Duplicates may be the result of human error (when data is entered by hand), or the features of data source (e.g., data scraped from classified ads with some items posted more than once).
* Often, easy process. Just check and get rid of repeated observations
* Sometimes, same observation is featured number of times.
  + Need to investigate. Makes sense / an error?
  + Example: Daily stock quotes, some stock features twice, with different price.
  + Decision- what to keep. Sometimes no clear-cut way, but usually no big deal.

Entity identification:

* More generally, you would need to have unique IDs
* It could be that two observations belong to two entities although ID is the same.
  + example: John Smith – there may be many
  + need to figure out, maybe assign unique IDs in raw data
* It could be that two observations have different ID but belong to same entity
  + need to figure out and have a single ID
* Unique IDs crucial. Numerical IDs are better

Non-entity observation:

* Rows that do not belong to an entity we want in the data table.
* Find them and drop them
* Such as: a summary row in a table that adds up, or averages, variables across all,
* or some, entities.
* Case study: a data table downloaded from the World Bank on countries often
* includes observations on larger regions, such as Sub-Saharan Africa

Missing values:

* Missing values mean that the value of a variable is not available for some, but not

all, observations.

* Key issues

1. Look at content of data - related to data quality (esp. coverage)

2. Missing values need to be identified.

* + Easy: “NA” (for “not available”), a dot “.”, an empty space “”.
  + Hard - binary 0 for no, 1 for yes, 9 for missing
  + Hard - percent 0-100, 9999 for missing
  + Hard numeric, range is 1-100000, 9999999999 for missing

3. Most software (Stata, R, etc) can have a value=missing. But when aggregate must pay attention.

4. Missing values should be counted. Missing values mean fewer observations with valid information. May actually have a lot fewer observations to work with than the size of the original dataset.

5. The third issue is potential selection bias. Is data missing at random?

Missing values - Understanding the selection process

* Random: When missing data really means no information, it may be the result of errors in the data collection process. Rare.
* In some other cases, missing just means "zero" or "no". In these instances, we should simply recode (replace) the missing values as "zero" or as "no".
* Often, values are missing systematically. Some survey respondents may not know the answer to a question or refuse to answer it, and such respondents are likely to be different from those who provide valid answers.,

Missing values: Some practical advice

* Focus on more fully filled variables. Often, simpler.
* Sometimes, informative if missing - create a new variable (called flag) to capture missing value and use this variable instead of the original.
  + For example, number of star for a restaurant.
  + Here flag/binary variable is 1 if restaurant does not have a star and 0 otherwise.
* Avoid automatic missing variable filling packages.
* Always be conservative, impute if absolutely necessary!
* For qualitative nominal variables, you may add missing as a new value: white, blue red and missing.
* For ordinal variables, you may add missing as new value or recode missing to a neutral variable: high, average, low, with missing recoded as average.
* For quantitative variables - you may recode with mean or median
* if impute, create a flag and use it analysis
* Always be conservative, impute if absolutely necessary!

Structure of files

It is good practice to structure the data files at three levels. These are

* Raw data files
* Clean and tidy data files
* Workfile(s) for analysis
* Output: graphs, tables

Data wrangling: common steps:

1. Write a code - it can be repeated and improved later
2. Understand the structure of the dataset, create data tables, recognize links. Draw a schema.
3. Start by looking into the data table(s) to spot issues
4. Store data in tidy data tables. Make sure one row in the data is one observation and manage duplicates
5. Get each variable in an appropriate format
6. Have a description of variables
7. Make sure values are in meaningful ranges; correct non-admissible values or set them as missing
8. Identify missing values and store them in an appropriate format. Make edits ifneeded.
9. Document every step of data cleaning