BITS Pilani
Pilani Campus

Introduction to Data Science

(BA ZG523 / CSI ZG523)

Tirtharaj Dash
Dept. of Comp. Sc. and APP Center for A.I. Research
BITS Pilani, K.K. Birla Goa Campus

innovate achieve lead

BITS Pilani
Pilani Campus

Introduction to Data Science

Lecture No. 04 (C.H.: 7, 8)

Previous Lecture:

- Various roles in a data science project
  - Project sponsor
  - Client
  - Data scientist
  - Data architect
  - Operations
- Stages of a data science project
  - Define the goal
  - College and manage data
  - Build the model
  - Evaluate and critique model
  - Present results and documents
  - Deploy model
  - [Repeat the stages in sequence, if needed]

Data

- Key ingredient of any analytical exercise.
- A rule: “The more data, the better”
- A principle: “Garbage In, Garbage Out (GIGO)”

Types of Data Sources

- Transactions
  - Consist of structured, low-level, detailed information capturing the key characteristics of a customer transaction (e.g., purchase, claim, cash transfer, credit card payment).
  - This type of data is usually stored in massive online transaction processing (OLTP) relational databases.
- Unstructured data
  - Embedded in text documents (e.g., emails, web pages, claim forms) or multimedia content
  - These sources typically require extensive preprocessing.
- Qualitative, expert-based data
  - An expert is a person with a substantial amount of subject matter expertise within a particular setting (e.g., credit portfolio manager, brand manager).
  - E.g. One would expect *a priori* that higher debt has an adverse impact on credit risk.

Cont.

- Publicly available data
  - Data originating from social networks (e.g. Facebook, Twitter)
  - Data originating from government sources (e.g. unemployment data, inflation data)

Sampling

- Drawing a subset of data instances (sample) from a pool of data (source)
- Source → (sampling procedure) → Sample
- Avoid sampling bias
- Stratified sampling
  - maintain the same predefined distribution of classes in the sample

Types of Data Elements

- Continuous:
  - These are data elements that are defined on an interval that can be limited or unlimited.
  - Example: income, sales, monetary.
- Categorical:
  - **Nominal**: These are data elements that can only take on a limited set of values with no meaningful ordering in between. (Example: marital status, profession, purpose of loan)
  - **Ordinal**: These are data elements that can only take on a limited set of values with a meaningful ordering in between. (Example: credit rating; age coded as young, middle aged, and old.)
  - **Binary**: These are data elements that can only take on two values. (Example: gender, employment status)

Visualizing Data

- Exploration: Plotting
  - Getting to know data in an “informal” way.
  - Different plots and graphs can be useful.
- Analysis: Inspect basic statistical measurements
  - Averages
  - Standard deviations
  - Minimum
  - Maximum
  - Percentiles
  - Confidence intervals

Missing Values

- Replace (impute)
  - Replacing the missing value with a known value
  - Usually: mean, median, mode, etc. of the features/attribute
  - E.g. marital status if empty can be replaced by mode of the column (the most repeated status)
- Delete
  - Most straightforward option
  - Deleting observations or variables with lots of missing values
  - Assumes: information is missing at random, no meaningful interpretation
- Keep
  - Missing values can be meaningful
  - E.g., a customer did not disclose his or her income because he or she is currently unemployed.

Today’s Practical

- We will learn about numpy
  - This is the core library for scientific computing in Python
  - some basic operations and functions
- We will learn about pandas
  - Fast, powerful, flexible, open data analysis and manipulation tool.

For your practice

If you have not done:

- Install Python 3.x [Link]
- Install anaconda environment [Link]
- Install jupyter notebook environment [Link]
- Install library numpy [Link]
- Install library pandas [Link]
- Install library scipy [not included today] [Link]

[*See the pipeline next*]

Cont.

I would follow the following pipeline for installation:

- Python (Download the file and install; or use command)
- Anaconda (File based installation)
  - Pip (conda install pip)
    - Scipy (pip install scipy)
  - jupyter (conda install -c conda-forge jupyterlab)
  - numpy (conda install numpy)
  - pandas (conda install pandas)

Lab repo

- Take the Jupyter notebooks from our lab repo and practice. This is useful: https://github.com/tirtharajdash/IntroductionToDataScience/tree/master/L04