Introduction to Machine Learning

From raw data to predictive models

Yordan Darakchiev

Technical Trainer

iordan93@gmail.com

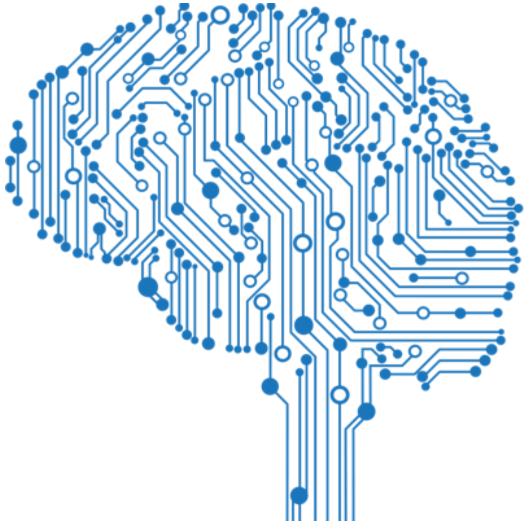


Table of Contents

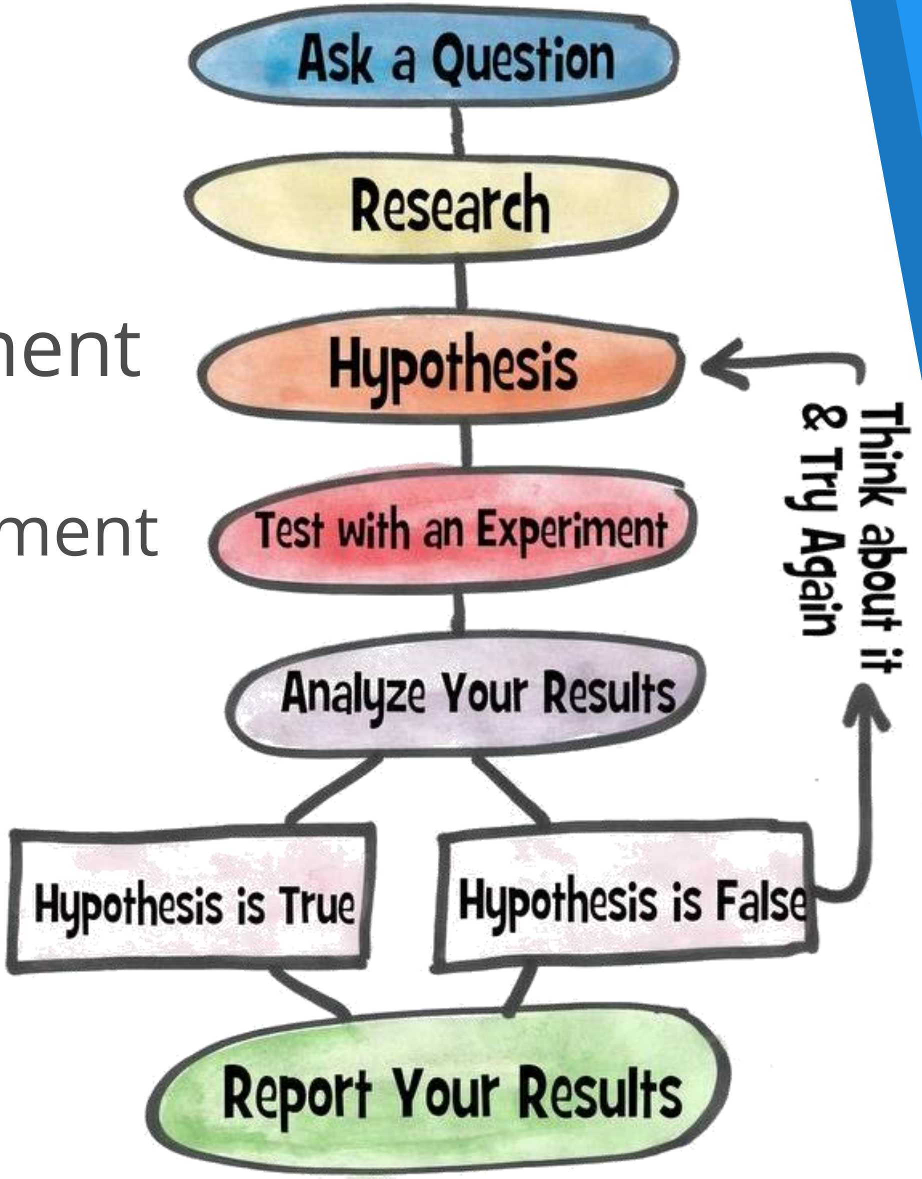
- sli.do: #ml-intro
- The scientific method – overview
  - Knowledge discovery from data
- Machine learning
  - Basic concepts
  - Algorithms (models) overview
- Getting, preparing and exploring data
  - Review
- Machine learning process overview

The Scientific Method

How to work with data…
the right way

The Scientific Method Steps

- Ask a question
- Do a research
- Form a hypothesis
- Test the hypothesis with an experiment
  - Experiment works ⇒ Analyze the data
  - Experiment doesn't work ⇒ Fix experiment
- Results align with hypothesis ⇒ OK
- Results don't align with hypothesis ⇒ new question, new hypothesis
- Communicate the results



OSEMN Model

- Some guidelines on the process to extract meaningful information from data
  - Very similar to the scientific method
  - Can be viewed as a sequential process
    - Or just as some guidelines on how to do research
  - Read as "awesome"

1. **O**btain data
2. **S**crub data
3. **E**xplore data
4. **M**odel data
5. i**N**terpret the results

Applied Machine Learning Process

- This allows us to do our job faster and more reliably

1. Problem definition
   - Make sure the problem is well-defined and that you're solving the right problem
2. Data analysis
   - Get familiar with the available data
3. Data preparation
   - Get the data ready for modelling
4. Algorithm evaluation
   - Test and compare algorithms
5. Result improvement
   - Use results to create better models (e.g. fine-tuning, ensembles)
6. Result presentation
   - describe the problem and solution to non-specialists

Machine Learning

Fundamental concepts

Machine Learning

- We described a general process
  - We didn't explain ML in detail
- *"A computer program is said to learn from experience* E *with respect to some task* T *and some performance measure* P*, if its performance on* T*, as measured by* P*, improves with experience* E*."* – Tom Mitchell, Carnegie Mellon University
- More simply, **making computers learn from data**
  - And observing them getting better and better
  - Results: computers do things that they weren’t explicitly told
- The field is vast (and expanding)
  - There are many sub-fields, variations and algorithms
  - … but the basis is still the same

Types of Machine Learning Algorithms

- **Supervised learning**
  - We train the program on previously known (labelled) data
  - After training, we expect it to make predictions on new data
  - Examples: regression, classification
- **Unsupervised learning**
  - We leave the program to find patterns in data
  - Examples: clustering analysis, dimensionality reduction
- **Reinforcement learning**
  - A form of unsupervised learning
  - The program learns continuously
  - Examples: learning to play a game by observing other players, learning to drive a car

Algorithms by Task

- **Statistical algorithms**
- **Regression** – predicting a continuous variable
- **Classification** – predicting class labels
- **Clustering** – finding compact groups of data points
- **Dimensionality reduction** – simplifying the input data
- **Recommendation** – suggest items for users
- **Optimization** – minimize / maximize a target function
- **Testing and improvement algorithms** – helper algorithms to select, fine-tune and optimize other ML algorithms
- … and more :)

Getting and Preparing Data

Review: Preparing raw data for modelling

Common Libraries

- In Python, we use libraries to perform common operations
- `scikit-learn` – machine learning models
- **`pandas`** – working with data
  - Reading, tidying, cleaning, preparation
- **`numpy`** and **`scipy`** – numerical and scientific libraries
  - Contain a ton of useful functions for performing research
- **`matplotlib`** – plotting and data visualization
- There are many more we'd like to use but these are the most commonly used ones

Getting Data

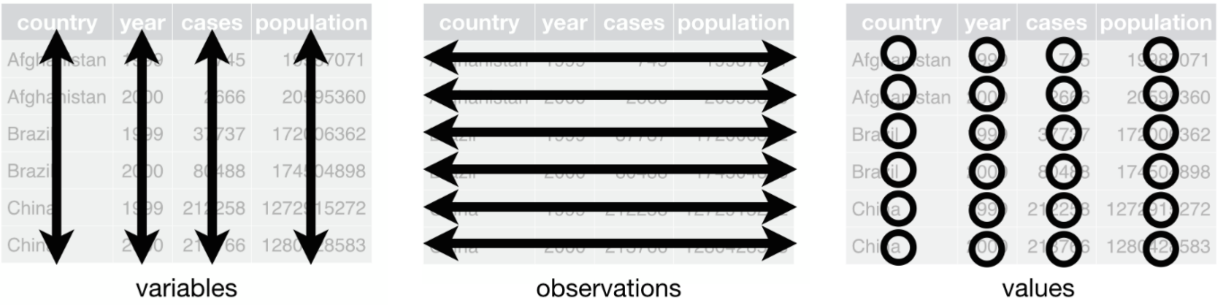
- Import pandas in your notebook or script
  - We usually give it an alias to make code shorter

```
import pandas as pd
```

- Read a dataset (table with data)

```
dataset = pd.read_table(...)
```

  - The method contains a lot of options
  - We can also read from other (non-local) sources
- Transform the data to make analysis easier
  - Tidy up the data
    - Attributes in columns
    - Observations in rows
    - Adding a new record = adding a single, complete row



Preparing Data

- Make other transformations as needed
  - Incorporate many datasets
  - Filter rows and columns
  - Group and aggregate values (e.g. sums by group)
  - Transform columns (e.g. apply a function to all values)
    - Change data types
    - Alter the distributions (e.g. `log`, `minmax`)
    - Calculate new columns (e.g. sum of two columns)
- All of these transformations are commonly used and have their own methods within pandas
  - 10 Minutes to pandas
  - Pandas Cheat Sheet
  - Full docs

Exploring Data

- Running an ML model is a small part of the process
  - Before that, we have to get to know our data
- **Exploratory data analysis (EDA)**
  - "Playing around" with the data using "mind power"

- An important part of EDA is creating graphs
  - With **`matplotlib`** or a similar library
  - Histograms and boxplots – to represent distributions
  - Line and bar charts – to represent relationships and allow comparisons
  - Scatterplots – to represent correlation
  - ... and many others, depending on the case
    - We can even create our own charts if we need that

Example: Getting and Exploring Data

- Passengers on the Titanic
  - Dataset provided in the additional materials: `titanic.csv`
  - Dataset info
- Read the data (using pandas)
- Tidy up and clean the data
  - While also exploring the information
  - No "hard and fast" rules – you've got to use intuition
  - **Usual workflow:** start by inspecting variables and data types, move to single-variable distributions, then try to find relationships between two or more variables; transform the data if needed
  - Deal with missing values and outliers, normalize the data if needed

Example: Preparing Data for Modelling

- Most models require two additional steps
  - **Convert categorical variables** into **indicator variables**

```
dataset = pd.get_dummies(dataset)
```

  - **Normalize values** if needed (e.g. scale all variables from 0 to 1 using minmax scaling, or use Z-scores)
- Perform other model-specific transformations
  - E.g. your model may not work well with highly imbalanced data (when you look for anomalies)
- If possible, prepare several versions of the dataset
  - To see how a transformation affects model performance
- **Describe and document the entire process!**
  - Don't forget the rules for reproducible research

Example: AzureML

- In this course, we'll be using Python code to run and evaluate models
    - We'll create a nice, structured pipeline
    - But there are other solutions
- Microsoft AzureML Studio: https://studio.azureml.net/
    - Good for a demo if you're not experienced in coding
    - Pros
        - Free to try
        - Easy, visual representation of the workflow
        - Has many predefined modules; can also execute Python code
        - Runs on the cloud – no need to throttle your machine
    - Cons
        - Hides away or obscures some important implementation details
        - Running on the cloud is too expensive sometimes

Summary

- The scientific method – overview
  - Knowledge discovery from data
- Machine learning
  - Basic concepts
  - Algorithms (models) overview
- Getting, preparing and exploring data
  - Review
- Machine learning process overview

Questions?