Artificial Intelligence & Data Analytics

Ng Hun Duc

L1, Intro

L2, Python

- AI: tools and methods which enable machines to succeed at tasks normally limited to human
- Data Analytics: raw data ⇒ insights ⇒ | actions / decisions

- Panda.

```
import pandas as pd
df = pd.read_csv('data.csv')
df[df.name == "tom"]
df[df.age > 15]
df["age"].min()
```

Q&A:

L3, - What are the steps / tasks in Data Preparation?
- What is data exploration? Cleaning? Transformation?
- What do we do in Cleaning Data?
How to deal with Missing Values, Outliers, inconsistencies?
- What do we do in Data Transformation?
What are Normalization, Aggregation, Discretization and their purposes?
- What are the common operation to prepare image data?
- What about Text Data?
What are tokenization, normalization, noise removal here?
What to notice with sensitive data?

L4, - What is the motivation behind the need for Data Integration
- What is Data Integration, Data Pipeline?
- What is the advantages of Database against conventional file-based system?
- Describe and compare Relational DB and Document-oriented DB.
- Briefly describe SQL.
- What is ETL?
- Describe and compare Data Warehouse & Data Lakes?

L5, - How do you understand feature extraction, selection, dimensionality reduction, feature projection, feature elimination?
- How do you understand the curse of dimensionality?
- Explain Maximum Relevancy Minimum Redundancy!
- Explain briefly PCA, LDA, & compare them.
What is the assumption in PCA?
goal of LDA?

AIDAE

Data Prepagation

L3/

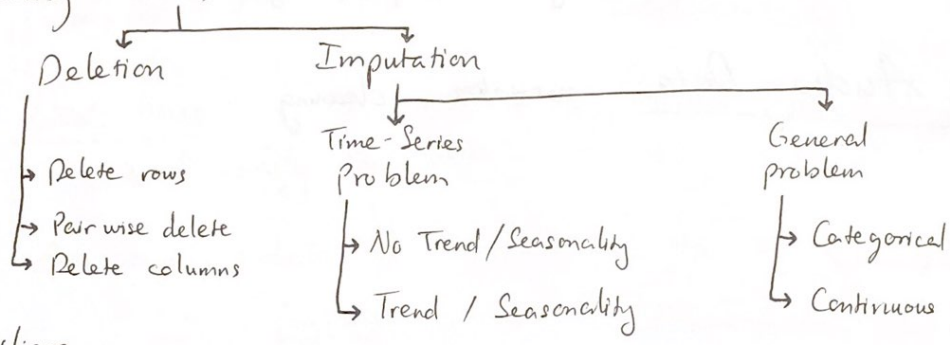
Data Prepagation = Exploration + Cleansing / Transformation ×

= Cleaning + Transformation + Integration + Reduction

⊛ Exploration: Creating initial understanding

⊛ Cleaning: Manual ⇒ Automatic

1/ - Missing Values



- Outliers:

- Inconsistencies: Ex: Semantic diff. Colour: ABB, Fanuc .. ⇒ Manually handle
Quality: Good, Poor, 2, 5 .. ⇒ Detect manually
Value diff

⊛ Transformation

2/ - Normalization

- Aggregation: combine >2 attributes into 1 reduce < dimension variability

- Discretization: ~~Ex~~ Age: { 1, .. , 10 ⇒ Young
60 ... 80 ⇒ Old
(Int ⇒ Word)

convert attributes ⇒ 1 discrete value

Image Data: transform, equalize, segment, augment

- Most common operations: resize, denoise, thresholding, light correction, segmentation ..

Text Data:

- Web page usually has API
- Operations:
  - tokenization (segment into word strings)
  - normalization (not upper/lower case, "one" vs "1"
  - noise removal (headers, footers, tag, ..)
- Sensitive data:
  - identification (name, bank account, personal info..)
  - anonymization / pseudonymization

Audio Data: normalize, cleaning

AIDAE

L4,

Data integration

- Motivation: variety of data sources (machines, data systems..)
  ⇒ need a transparent connection, pipeline..
  a consolidated system architecture

- Definition: +) database integration: combining heterogeneous data from various sources, to enable users a unified access

  +) data pipeline: software system (ex: code program)
  takes data (from 1→many source) ⇒ transform → write to output(s)

⊛ Data base: - has data base - management system (DBMS) to handle / modify the data

- Advantage: - data is searchable, can be cross-gathered
  - failsafe .. ACID (Atomicity Consistency Isolation Durability

- Types:

Relational DB	Document-oriented DB
Tables with columns having relation	Documents, similar to JSON
Predefined schema	Each doc can be complex, different
Uses SQL	No SQL
Not as flexible/scalable	High flexibility

schema = structure

- SQL (Structured Query Language)
  Based on Relational Algebra
  Close to Natural English language

Select * from Table;
Select Column1 from Nam
Select * from .. where condition?

Insert into — Values —
Update — set —

+, Approaches to Data Integration

- ETL (Extract, Transform, Loaded) into target system / data warehouse

- Data Warehouse: a central database system which integrates data from all kinds of company-wide op. data sources for subsequent analysis purpose

- Schema-on-write vs Schema-on-read

Define schema before any data is written to it

Data Lakes natural format (raw ..)

= Data Warehouse vs Data Lakes

	Data Warehouse	Data Lakes
Data		Central repository for storing data in its natural format
Schema	schema-on-write	schema-on-read
Quality	↑	↓ raw data ..
Users	Business Analysts	Scientist, developers
Analytics	BI ..	ML ..
Price	↑	↓

= Publish / Subscribe mechanism: to transport the data

Data Source publish ⇒ Cloud..
Consumers subscribe to topic

- MQTT: (Message Queue Telemetry Transport)
a easy to use publish / subscribe real-time protocol

use a broker to address messages to the right channels
enable real-time push
most promissing for IoT

Feature

L5,

we have many different representations of data representing the world

- Extract feature: process of deriving features from an real world object; Should be < informative, non-redundant

Bag of word: feature for texts

- Feature Selection

+) Curse of Dimensionality: the exp growths of feature space ..

⇒ Only pick a subset of the features

+) Benefit: moderate the curse of dimensionality
- reduction of training time
- generalization

+) Dimensionality reduction: ≠ is different thing (feature projection) = represent well data in a lower dim space

+) Maximum Relevancy Minimum Redundancy

Mutual Info?

$$MI(X_i, C) \qquad MI(X_i, X_j)$$

$$V_I(S) = \frac{1}{|S|} \sum_{x_i \in S} MI(X_i, C); \quad W_I(S) = \frac{1}{|S|^2} \sum_{x_i, x_j \in S} MI(X_i, X_j)$$

$$S^* = \underset{S \subseteq U}{\operatorname{argmax}} \left(V_I(S) - W_I(S) \right)$$

$$MI(X, Y) = I(X) + I(Y) - I(X, Y) \geq 0$$

$$I(X_1, Y_2) = \sum_{x_1} \sum_{x_2} p(x_1, x_2) \log \frac{p(x_1, x_2)}{p(x_1)\, p(x_2)}$$

⊗ PCA Dimensionality reduction can be divided into { feature elimination / feature extraction (create new variables from old ones)

PCA is a technique for feature extraction

Reality implementation: $N < D$ Gram-Schmit process; $D, N >>$ Power method

LDA Problem with PCA: Assumptions: "Dim with highest σ^2 is most imp"

⇒ LDA

⊗ Compare PCA & LDA

	PCA	LDA
Requirements	Unsupervised	Supervised
Optimization goal	Maximize variance	..
Technique		

⊗ Neural network uses part of its layers to perform feature extraction

⊗ Then definitions are .. varying

+) Feature selection ≈ feature ~~extraction~~elimination
However, in some algo., f. elimination refers backward algo., start with complete set of feature, then eliminate till criterion are reached
f. selection ——forward——, ——— empty set ———, —— add features ————

+) F. extraction ≈ F. projection

+) PCA is just extraction, the elimination step is not officially in PCA

PCA

$$X_{(D\times N)} = \begin{bmatrix} U_K & \bar{U}_K \\ (D\times K) & [D\times(D-K)] \end{bmatrix}_{(D\times D)} \cdot \begin{bmatrix} Z_{(K\times N)} \\ Y_{[(D-K)\times N]} \end{bmatrix}_{(D\times N)}$$

$$X = U_K . Z + \bar{U}_K . Y \Leftrightarrow \begin{cases} Z = U_K^T . X \\ Y = \bar{U}_K^T . X \end{cases}$$

We want to replace $Y \approx b . \underline{1}^T_{(1\times N)}$

$$\Rightarrow b = \underset{b}{\operatorname{argmin}} \| Y - b.\underline{1}^T \|_F^2 = \underset{b}{\operatorname{argmin}} \| \bar{U}_K^T . X - b.\underline{1}^T \|_F^2$$

Set $\frac{\partial f(b)}{\partial b} = 0 \Rightarrow$ Find $b = \bar{U}_K^T . \bar{x}$ with $\bar{x}$ is expectation of x

$$\Rightarrow X \approx \tilde{X} = U_K . Z + \bar{U}_K \bar{U}_K^T \bar{x} . \underline{1}^T$$

Loss func: $J = \frac{1}{N} \| X - \tilde{X} \|_F^2 = \frac{1}{N} \| \bar{U}_K \bar{U}_K^T X - \bar{U}_K \bar{U}_K^T \bar{x} . \underline{1}^T \|_F^2$

If U is orthorgonal:

$$\Rightarrow J = \dots \sum_{i=K+1}^{D} u_i^T S u_i$$

$$L = \sum_{i=1}^{D} u_i^T S u_i = \sum_{\lambda}^{D} \lambda_i$$

$\Rightarrow L$ doesn't depend on how U is chosen

$$\Rightarrow J_{min} \Leftrightarrow F_{max} = L - J = \sum_{i=1}^{K} u_i S u_i^T$$

$\Rightarrow$ Choose K maximum eigen values λ_i

PCA algorithm step

1) Calculate expectation / mean: $\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i$

2) Translate to 0: $\hat{x}_n = x_n - \bar{x}$

3) Covariance matrix: $S = \frac{1}{N} \hat{X} \cdot \hat{X}^T$

4) Find λ_i of S, rank / sort from greatest to smallest

5) Find respective eigenvector & normalize $\Rightarrow U_k$ each column is an eigenvector

6) $Z = U_k^T \cdot \hat{X}$

7) $x \approx U_k \cdot Z + \bar{x}$

⊗ Assumption: Dimension with the highest σ is the most important which might not always true

$\Rightarrow$ LDA

LDA

LDA algorithm steps:

1) Calculate mean μ_i of each class c_i, and of all data

$$\underset{[D\times 1]}{\mu_{c_i}} = \frac{1}{N_{c_i}} \sum x_{c_i} \; ; \quad \mu = \frac{1}{N} \sum x_i$$

2) Scatter matrix

+) Within-class scatter matrix: $S_W = \sum_i^c S_i$

$$S_i = \sum_{x \in C_i}^{n} (x - \mu_{c_i})(x - \mu_{c_i})^T$$

Or calculate covariance matrix: $\Sigma_{c_i} = \frac{1}{N_{c_i} - 1} \sum_{x \in C_i}^{n} (x - \mu_{c_i})(x - \mu_{c_i})^T$

then $S_W = (N_i - 1) \cdot \Sigma_i$

+) Between class

$$S_B = \sum_{i=1}^{c} N_i (\mu_{c_i} - \mu)(\mu_{c_i} - \mu)^T$$

3) Find eigenvalues / eigenvectors of $S = S_W^{-1} \cdot S_B$

4) Carry on as PCA

Supervised Learning

L6)

⊛ Definition

AI: Any technique which enables a computer to mimic human behavior

- General AI (GAI): Transfer knowledge across domains
  ↳ system that can do many tasks
- Narrow AI: Perform a single task extremely well
  - ↓ Machine Learning: Algorithms whose performance improve as they are exposed to more data
    - ↓ Supervised learning: labeled data
      - Unsupervised learning
      - Re-inforcement learing

- Regression V/S classification
  value ⟵ output ⟶ category

Linear Regression

Logistic Regression

K-nearest neighbors, SVM

Decision Tree Regressor

Decision Tree

Perceptron..

CNN..

- Confusion matrix & metrics: $ACC = \frac{TP + TN}{P + N}$

$$F_1 = \frac{2TP}{2TP + FP + FN}$$

- Gradient descent, overfitting, cross validation..

AIDAE

Unsupervised Learning

L7,

Doesn't have label ⇒ Model discover relations

1, Finding clusters:

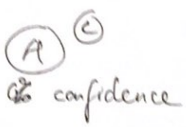
+, Clustering:

Hierarchical clustering	Partitioning clustering	Density-based Clustering
Single-Linkage	K-means	Mean-shift
Wards method	Fuzzy C-means	DBSCAN
	Affinity Propagation	OPTICS
	EM-Clustering	

+, Anomaly Detection: objects outside of clusters

2, Association Rule Mining: discover associations (relationship, dependency) between variables

- Support = Antecedent ∩ Consequent &
- Confidence = Support / Coverage



% confidence

100% Confidence

- from apyori import apriori
  transactions = [. .]
  association_rules = list(apriori(transactions))

3, Dimensionality Reduction:

Feature Selection	Linear Projections	Non-linear projections
Feature filtering	PCA SVD	Transform feature space
Random forest (Gini..)	ICA	while maintaining || local / global | distance metric

AIDA

Reinforcement Learning

Ls,

- Main idea: goal - feedback : reward
  learn from (state, action, reward)

+) MDP (Markov Decision Process): $\langle S, A, P, R, \gamma \rangle$

- Markov Property: $P[S_{t+1} | S_t] = P[S_{t+1} | S_1 .. S_t]$
- $G_t = R_{t+1} + \gamma \cdot G_{t+1}$

  $V_\pi(S_t) = E_\pi(G_t | S_t)$; $Q_\pi(S_t, a) = E_\pi(G_t | S_t, a)$
- Bellman equations: $V^*(s) = \max_a Q^*(s,a)$

  $Q^*(s,a) = \sum_{s'} T(s,a,s')[R(s,a,s') + \gamma \cdot V^*(s')]$

  $V^*(s) = \max_a \sum_{s'} T(s,a,s')[R(s,a,s') + \gamma \cdot V^*(s')]$

+) Classification:

- Offline vs Online

  Dynamic Programming — ~~on-policy~~ online [actor critic, Q learning]

  (Value iteration: synchronous / asynchronous)
- Model-based v/s Model-free

  Learn $P(s'|a,s)$, $R(s,a)$ — Learn $V(s)$, $Q(s,a)$

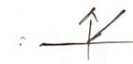
- Exploration v/s Exploitation

  ⇒ ε-greedy

Neural Networks

Lg,

+) Some functions:

- Sigmoid: $f(s) = \frac{1}{1+e^{-s}}$ ⇒ Logistics Regression (2 classes)
- Tanh: $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
- ReLU: 
- Leaky ReLU 

⇓

Softmax Regression

$$a_i = \frac{\exp(z_i)}{\sum_j^c \exp(z_j)}$$

- One-hot coding

+) Loss functions:

- L2, L4
- Cross entropy: $L = -\sum_i^c t_i \log(y_i)$
- Hinge loss: $L = \sum_n [1 - t_n \cdot y(x_n)]$

AI DA

Visual Analytics

L11

- To interprete data in an efficient way
  so that in can be understand, use, learn..
- Visualization: a way to communicate concrete, abstract ideas

- Visual Analytics: creation of tools, techniques enable people
  - synthesize info, insight from massive, ambiguous .. data
  - detect the expected & discover the unexpected
  - provide timely .. assessment ⇒ communicate ⇒ action

(Statistic, Graphics, Data mining ...) Data / Machine + Human (Cognitive, perception, visual intelligence.)

- Preprocess ⇒ apply algorithmic analysis ⇒ Visualize
  ⇒ generate insights ⇒ new hypotheses ⇒ updated visualization

- There are available tools:
  Tableau, Superset, Qlikq, Power BI

- Incomplete problem formulation ⇒ the need for visullizations

1, EDA & visual analytics — visual representation of raw data / underlying info

Exploratory data analysis

Ex: plot it .. to | see if we get the right predict / find outliers?

+) from pivottablejs import pivot_ui

import ggplot..

⊛ Challenges: Scalability with Data Volumes & Data Dimensionality

Quality of data & graphical representation

Visual representation & Level of Detail

User Interfaces, Interaction Styles & Metaphors

Display Devices

Evaluation & Infrastructure

SQL GUIDE

2, Modeling & visual analytics — visual representation of model and prediction / underlying info

Ex: Why given an image, a network classify as a "dog"

+) Target audience:

different audience

different need

- Domain experts (doctors, insurance agents..) trust?
- Affected users (..) ⇒ understand, verify
- Regulatory entities (gov..) ⇒ certify, audits..
- Data scientist, developer ⇒ efficiency, research
- Managers, EB ⇒ assess, understand

⇒ XAI (Explainable AI): ethics, fairness, privacy, trust, transparency

STEPFA

Ex: Activation Atlas

Tensorflow playground