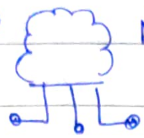
PRACTICAL DATASCIENCE ON AWS - CLOUD

- Local Data Science → limited by Hardware & Memory
  ↓



- Data Science on cloud → Agility | Elasticity
  → Scale up | Scale out

- Scaling up → If Model training takes too long → means u've exhausted your CPU limit on the current compute instance, so you can
  ↳ pick a compute instance with higher CPU resources.
  ↳ pick up a GPU based compute instance

  ↓ Ise size of compute instances (single CPU)

- Scaling out → Move from single CPU to PARALLEL computing CPU instances.

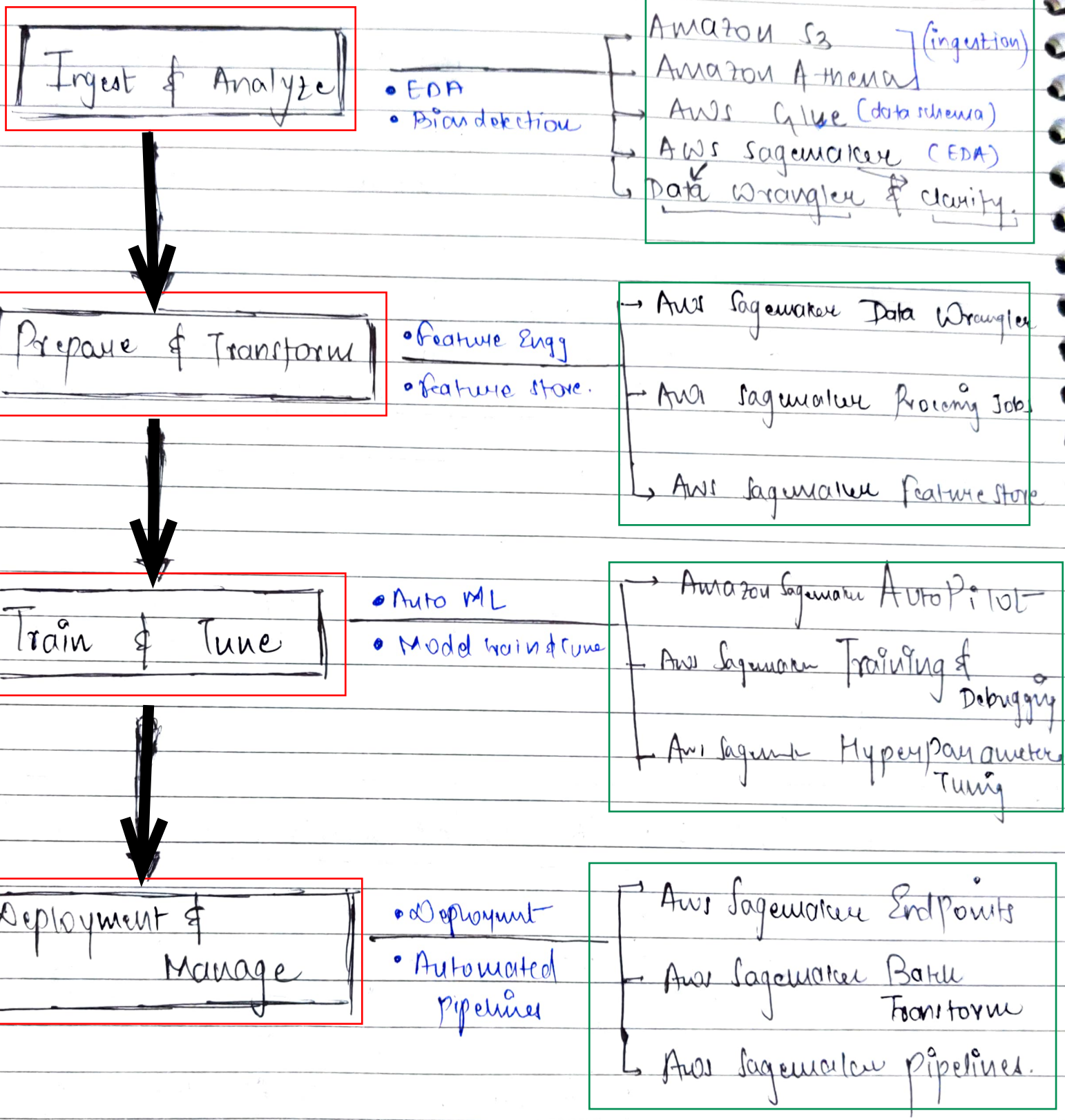
  ↓ (parallel X CPUs)

"Scaling Up & Scaling Out Are possible within seconds in cloud"

→ Once model training is finished, the instance is killed.
↓
No extra payment. { Pay As you Use }

- Cloud has (AWS has) its own ML Toolbox that saves time & makes it Quick & easy.

Mapping Data-science Workflow to AWS Tools



Typical Data ingestion & Query / Explore Workflow

(1) INGEST : Ingest the Data with AWS S3

↓

(2) CATALOG : Catalog the data into desired schema by using AWS Glue

↓

(3) EXPLORE : Explore & run SQL Queries on data using AWS Athena

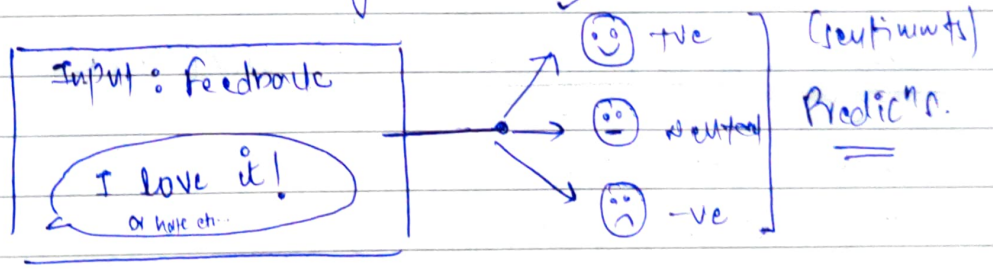
Use-Case Description → Multi Class Classification for sentiment Analysis of Product reviews.

- You work at an e-com company like Myntra/ flipkart.
- Your customers leave feedbacks via all online channels.

↓

Aim is to analyze these feedbacks quickly & alert if there are any product issues.

Example :



Observation 1 → The feedbacks generated/received can be millions every minute! So, we might need a data (ingestion) Repository that's ELASTIC enough to expand as per data size & formats.

↳ S3 ✓

Data Lakes : data is ingested into data lakes.

↳ Centralized & secure repository

↳ store / discover / share data @ ANY SCALE

- Structured data (CSV ...)
- Semi-structured data (XML, JSON ...)
- Unstructured (Image, video ...)
- Streaming data (LIVE).

→ • Needs to be governed ie managed to be used by further teams.

↳ Data lakes on **Aws : S3**

↳ stores data chunks as objects → each object with unique identifier & metadata for easy extraction.

⊛ Aws Data Wrangler (pip install awswrangler)

- is a python library
- connects pandas + Aws powers.

```
→ import awswrangler as wr
→ import pandas as pd
→ df = wr.s3.read_csv (path = '-f-f-')
```

⊛ Aws Glue Data ~~Wrangler~~ catalogue

→ Catalogue is the metadata of your data (path, schema et)

↳ info that you need to register before importing

→ import awswrangler as wr

→ wr.catalog.create_database(name = ---)

→ wr.catalog.create_csv_table(-----)

Catalog 2

Name = ~
Database = ~
Classification = csv
Locn = s3://< >/--

(*) SQL Queries using AWS ATHENA → (SQL Queries in Python ENV)

→ import awswrangler as wr

→ wr.athena.create_athena_bucket()

→ df = wr.athena.read_sql_query(sql = 'select * ---', database = '---')

↓

This queried df is first stored by Athena into (S3) & then stored in variable (df).