

# \* PET PROJECT \*



Happy Learning  
Dheeraj Pattnaik

# 3 INGREDIENTS OF SUCCESSFUL PET-PROJECT

- YOUR **MINDSET** [IT'S YOUR LIFE · PLAN]
- YOUR **STRATEGY** [WHERE YOU WANT TO GO]
- YOUR **EXECUTION** [HOW YOU WILL GET THERE]

## WHY \*PET-PROJECTS IN FIRST PLACE ?

- EXPERIMENTATION
  - NEW SKILLS
  - JOB
  - STRETCH GOALS
  - FUN
- NEW LEARNING
  - YOUR STORY
  - SHOWCASE
  - GAIN WORK EXPERIENCE
- INSPIRATION
  - GREAT THING TO TALK ABOUT

## WHAT SHOULD BE YOUR GOAL \*

- MAKE IT YOUR **VOICE** EVERYWHERE YOU GO (SOFT SKILL - PRACTISE)
- YOUR **BRAND IDENTITY** (STRATEGIZE. KNOW WHO YOU WANT TO BE)
- FOCUS ON CREATING **VALUE & SOLVING NEEDS** (KEY)
- HAVE **STRONG FOUNDATION** (TECHNICAL AND BEHAVIORAL)
- BUILD TANGIBLE AND SCALABLE **PRODUCTS** (CAN YOU BUILD A BUSINESS OUT OF YOUR PROJECT IDEA?)
- ADD YOUR **UNIQUE PERSPECTIVE**
- CREATE TOUCHPOINTS TO **TALK & SHOWCASE** (IN INTERVIEWS)
- TELL A **COMPELLING STORY** (EMOTIONAL CONNECT)
- HAVE **FUN & LOVE** WHAT YOU DO
- BRINGING **EVERYTHING TOGETHER**

LESSONS  
FROM MY  
JOURNEY  
LEARNING  
EXPERIENCE

# #1. DEFINE PROBLEM\*

CLEAR & CONCISE  
ACTIONABLE  
MEANINGFUL  
MEASURABLE  
STRUCTURED APPROACH

## \* 2 SIMPLE WAYS TO DEFINE

### POINT OF VIEW STATEMENT [POV]

 (USER) NEEDS A WAY TO (VERB) BECAUSE (COMPELLING INSIGHT)

### Why-How LADDERING

ASK "WHY" TO GET MORE ABSTRACT STATEMENTS  
ASK "HOW" TO GET MORE SPECIFIC STATEMENTS

## PROBLEM SOLVING TECHNIQUES/WAYS

- ZOOM IN & OUT OF THE PROBLEM (5 WHYS → GO DEEPER LEVELS)
- REDUCE PROBLEM TO ONE-WORD AND FIND RELATED CONCEPTS (MIND MAPS)
- SWITCH DIFFERENT ROLES AND PERSPECTIVES (ROLE PLAYING)
- DEFINE CAUSE-AND-EFFECT AND EXPAND ON SUB-CAUSES (FISHBONE DIAGRAM)
- GATHER SIMILAR ITEMS, TYPES, NEEDS, FEATURES (AFFINITY DIAGRAM)
- CONTINUOUS IMPROVEMENT OF PROCESSES, PRODUCTS (DEMING'S WHEEL - PDCA)  
PLAN - DO - CHECK - ACT

## MY CHECKLIST

- |   |  |
|---|--|
| <input type="checkbox"/> PROBLEM STATEMENT*   | <input type="checkbox"/> RETURN ON INVESTMENT*<br>(ROI FOR BUSINESS)   |
| <input type="checkbox"/> PROBLEM MINDMAPS   | <input type="checkbox"/> NEXT BEST ALTERNATIVE                         |
| <input type="checkbox"/> STAKEHOLDER MAPS   | <input type="checkbox"/> EARLY HYPOTHESIS*                             |
| <input type="checkbox"/> ASSUMPTIONS & QUESTIONS  | <input type="checkbox"/> POTENTIAL SOLUTION                            |
| <input type="checkbox"/> POSSIBILITIES LIST*<br>(ALL POSSIBLE USECASES<br>FOR YOUR PROJECT) | <input type="checkbox"/> INTEGRATION INTO BUSINESS*<br>(HOW TO SCALE ) |

\* INTERVIEW  
TOUCHPOINTS

LESSONS  
FROM MY  
JOURNEY  
LEARNING  
EXPERIENCE

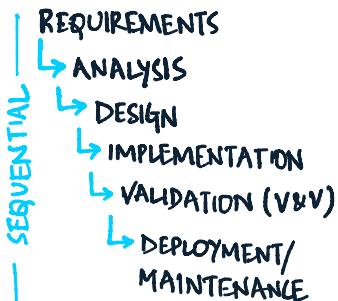
## #2 CHOOSE A METHODOLOGY\*

**WHY?**

- LIFE CYCLE OF ANY PIECE OF SOFTWARE
- PROCESS BRINGS STRUCTURE & CLARITY
- CONTROL DEVELOPMENT OF ANY INFORMATION SYSTEM
- FRAMEWORK FOR SUCCESSFUL PRODUCTS/PROJECTS/SERVICES

## 3 DEVELOPMENT METHODOLOGIES

### TRADITIONAL



### AGILE/SCRUM



### RAPID APPLICATION DEVELOPMENT (RAD)



## MY RECOMMENDATION [BASED ON MY WORK EXPERIENCE]

IMPROVING PROCESS ?

**SIX SIGMA / LEAN**

LONG TERM & FIXED PROJECT ?

**TRADITIONAL**

BUILDING PRODUCT (INCREMENTAL UPDATES) ?

**AGILE / SCRUM**

HIGH STAKES & HIGH VISIBILITY ?

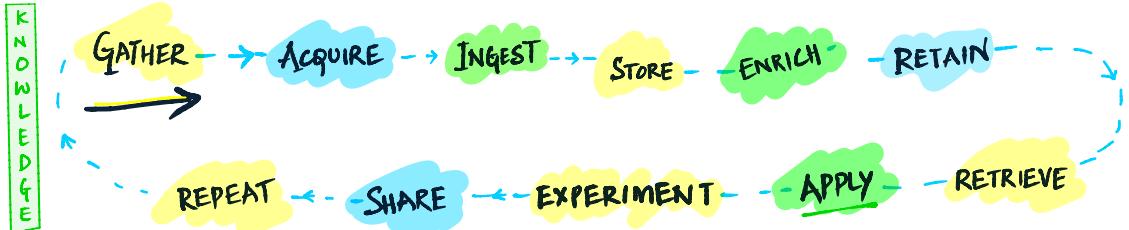
**HYBRID (AGILE + ITERATIVE)**

\* MAKE A POINT TO DISCUSS ABOUT YOUR METHODOLOGY IN YOUR INTERVIEWS \*

# #3 GATHER

# KNOWLEDGE & IDEATE

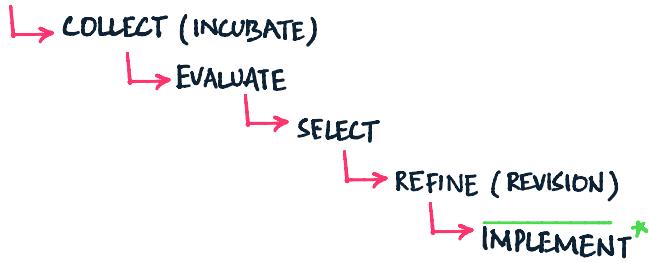
## KNOWLEDGE PHASES



ELEMENT OF THOUGHT  
• VISUAL • CONCRETE • ABSTRACT

## IDEA MANAGEMENT

### GENERATE



### MY IDEA GENERATION TECHNIQUES

- MINDMAPS\*
- DOODLING
- DESIGN THINKING\*
- 5WHY'S
- REVERSE ENGINEER
- COLLABORATION\*

## MY CHECKLIST [FOR PET PROJECTS]

- |   |   |  |
|---|---|--|
| <input type="checkbox"/> ELEVATOR PITCH*<br>(15 SECS TO EXPLAIN YOUR PET PROJECT)                 | <input type="checkbox"/> IDEA POOL<br>(LIST OF ALL IDEAS GOOD, BAD, OBVIOUS)                      | <input type="checkbox"/> OPPORTUNITY STATEMENT     |
| <input type="checkbox"/> TAG LINE (SINGLE LINE)   | <input type="checkbox"/> USECASES*  | <input type="checkbox"/> OSBORN CHECKLIST (GOOGLE) |
| <input type="checkbox"/> IDEATION TECHNIQUES*   | <input type="checkbox"/> IDEA STACKING*<br>(STACK MULTIPLE IDEAS TO FORM BIGGER AND BETTER IDEAS) | <input type="checkbox"/> PRODUCT OR BUSINESS IDEAS |
| <input type="checkbox"/> LANDSCAPING (IDENTIFY GAPS)<br>(USED WHILE GENERATING IDEAS FOR PATENTS) | <input type="checkbox"/> MOODBOARDS (FOR INSPIRATION)   |  |

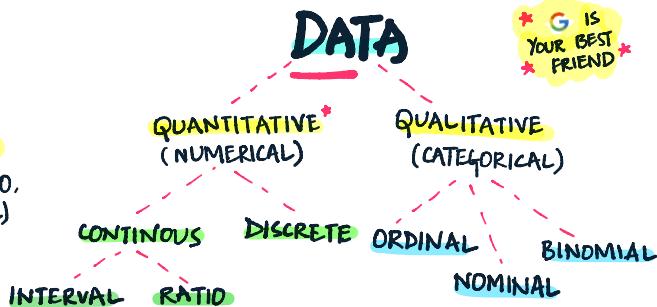
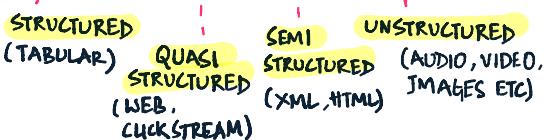
\* MAKE A POINT TO DISCUSS YOUR IDEATION TECHNIQUES IN YOUR INTERVIEWS \*

#4

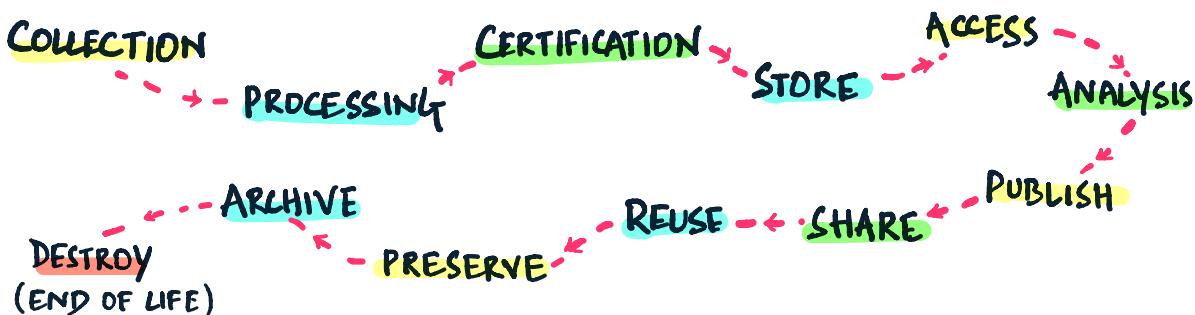
# DATA ESSENTIALS 101\*

- INDIVIDUAL PIECES OF FACTUAL INFORMATION
- BASIS OF REASONING AND INTERPRETATIONS
- QUINTESSENTIAL COMPONENT OF DECISION SCIENCE

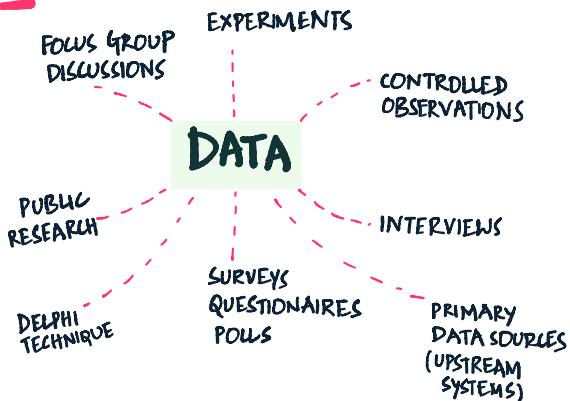
## DATA



## DATA LIFE CYCLE



## DATA COLLECTION METHODS

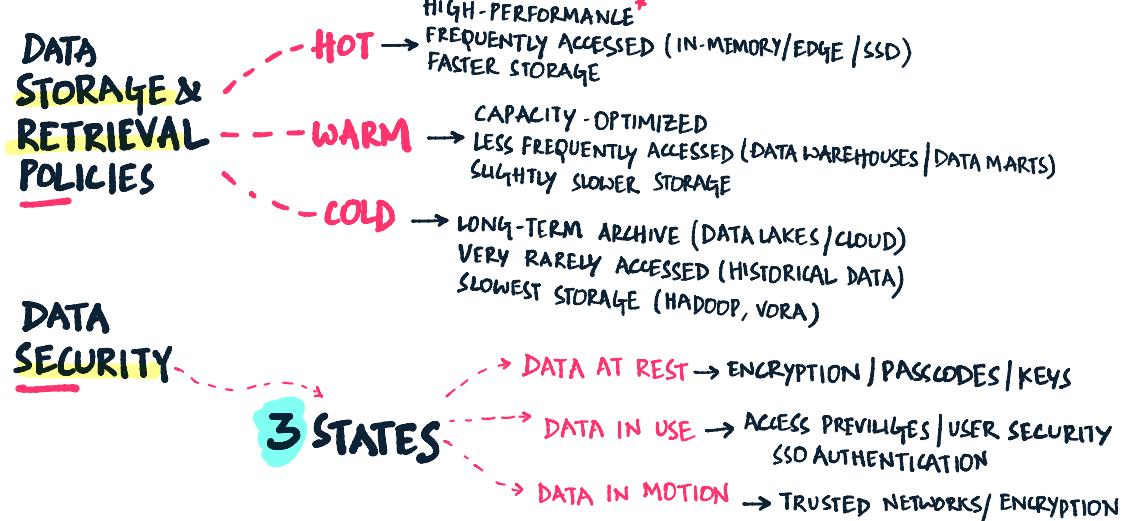


## DATA PROCESSING TYPES

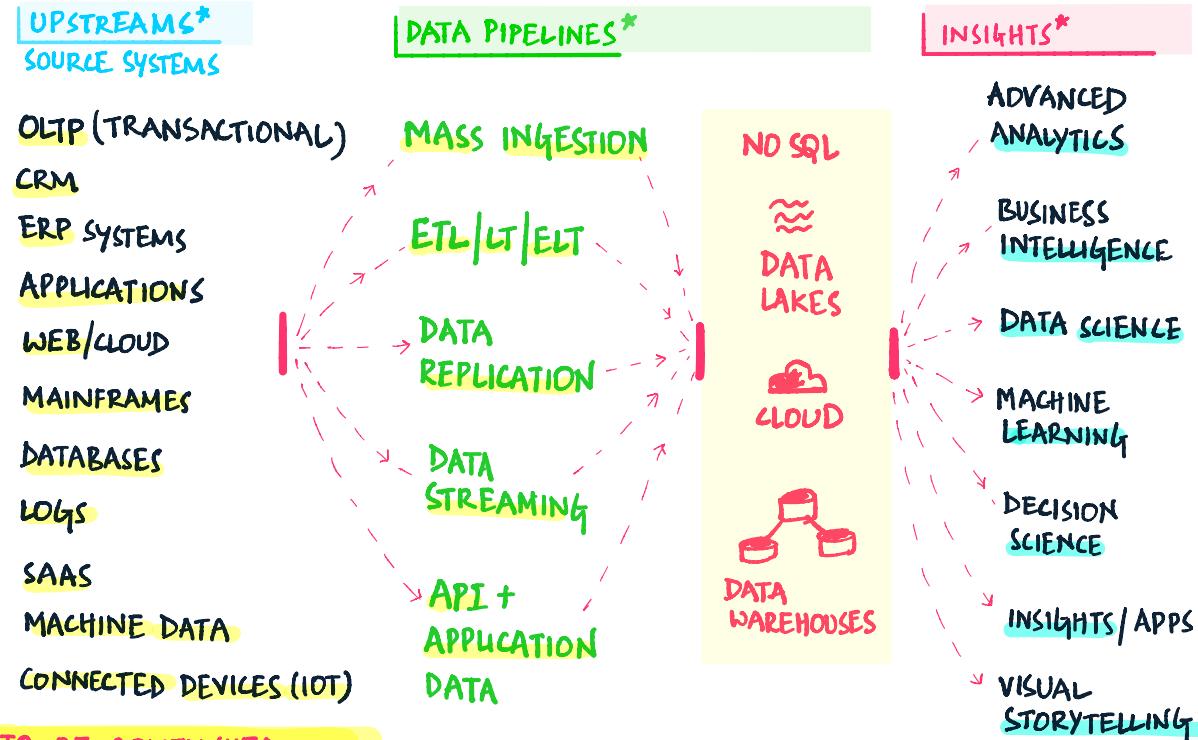
- BATCH PROCESSING
- REAL-TIME
- STREAMING
- MULTI PROCESSING
- TIME SHARED
- AUTOMATIC PROCESSING

TO BE CONTINUED xx

# #5 DATA ESSENTIALS 101\*

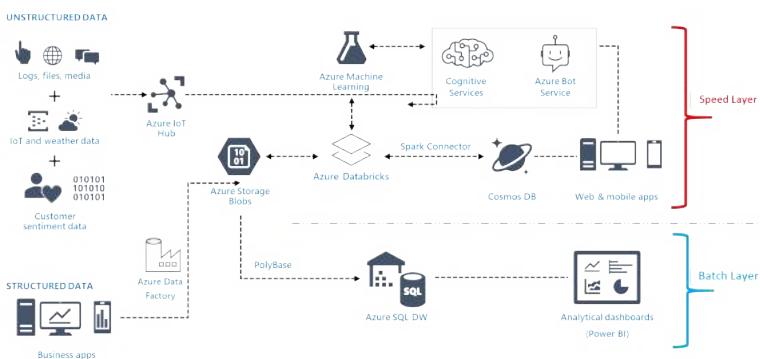
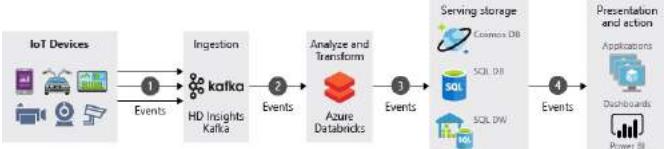
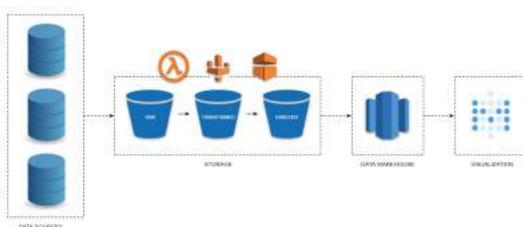
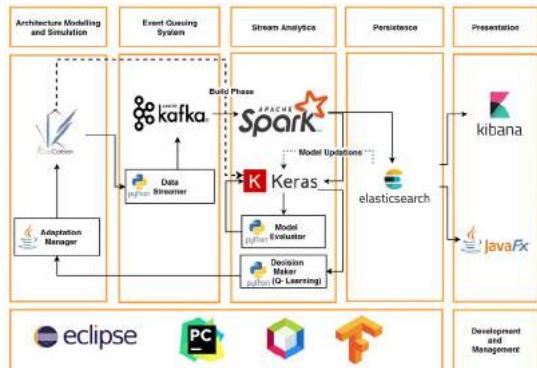


## ENTERPRISE DATA PIPELINE [GENERIC & APPLICABLE TO MOST ORGANIZATIONS]



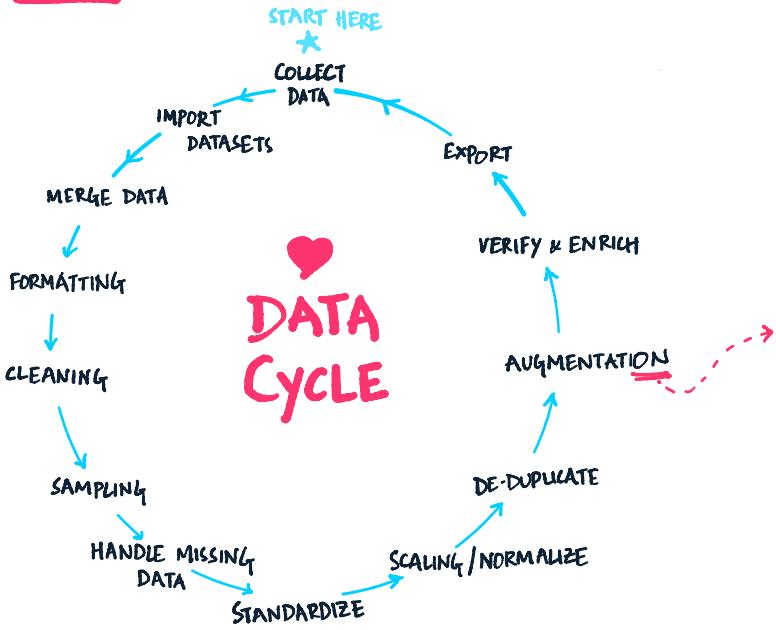
TO BE CONTINUED XX

# #6 KNOW YOUR DATA [IN THE CONTEXT OF PET PROJECT]



# #6 KNOW YOUR DATA [IN THE CONTEXT OF PET PROJECT]

## 3 STEP DATA MANAGEMENT



**DATA AUGMENTATION TECHNIQUES**  
(CREATE NEW USEFUL DATA)

### IMAGE

- CROPPING • SCALING
- PADDING • FLIPPING
- KERNEL FILTERS
- COLOR SPACE TRANSFORMATION

### AUDIO

- CHANGE PITCH • SPEED
- ADD NOISE • SHIFT TIME

### TEXT

- USE SYNONYMS • THESAURUS
- RANDOM SWAP • DELETION
- MASKED LANGUAGE MODEL
- BACK TRANSLATION • NOISE

### NUMERICAL DATA

- PCA • LOG TRANSFORMATIONS
- AUTO ENCODERS

**UNIVARIATE** • ONE VARIABLE OF INTEREST

**MULTIVARIATE** • MULTIPLE VARIABLES OF INTEREST

## EXPLORATORY DATA ANALYSIS

→ GAIN MAXIMUM INSIGHT INTO A DATASET

→ DISCOVER PATTERNS, TEST HYPOTHESIS

→ SUMMARIZE MAIN CHARACTERISTICS OF DATA VISUALLY

→ FIND ANOMALIES, CHECK ASSUMPTIONS

■ CENTRAL TENDENCY  
(MEAN, MEDIAN, MODE)

■ FREQUENCY DISTRIBUTION  
(BINS, HISTOGRAM/BOX)

■ VARIANCE  
(QUARTILES, STD DEV)

■ CO-LINEARITY

■ IDENTIFY RELATIONSHIPS

■ SPATIAL DEPENDENCIES

■ OUTLIERS

■ SKEWNESS

■ KURTOSIS  
(PEAKS)

■ CLUSTER ANALYSIS

■ DIMENSIONALITY

# #7 KNOW YOUR DATA [IN THE CONTEXT OF PET PROJECT]

## WHERE TO FIND BEST PUBLIC DATASETS



kaggle

Quandl

DATA.GOV

UCI Machine Learning Repository



Socrata.



Academic Torrents

WIKIPEDIA  
The Free Encyclopedia

VISUALDATA.IO

Google Dataset Search...  
Be the first to understand!Carnegie Mellon University  
Libraries

### SPECIFIC\*

COMPUTER VISION

- XVIEW
- KINETICS-700
- IMAGENET
- VISUAL DATA
- GOOGLE OPEN IMAGES

SENTIMENT

- SENTIMENT 140
- IMDB
- YELP
- STANFORD TREEBANK
- LEXICODER
- TWITTER US AIRLINE

TEXT

- 20 NEWS GROUP
- REUTERS NEWS
- WORDNET
- WIKI QA CORPUS
- UCI SPAM BASE

BLOGGER CORPUS

JEOPARDY

SMS SPAM

GUTENBURG

AUDIO

SELF-DRIVING

HOTSPOT QA

AMAZON REVIEWS

ROTEN TOMATOES

GOOGLE BOOKS N-GRAMS

NLP

LIBRI SPEECH

TIMIT

2000 HUB5 ENGLISH

SPOKEI JIKI CORPUS

FREE MUSIC ARCHIVE (FMA)

MILLION SONG DATASET

WAYMO OPEN DATA

BOSCH SMALL TRAFFIC

BERKELEY DEEP DRIVE

LARA TRAFFIC

WPI DATASETS

#8

# DATA IN A NUT-SHELL

[IN THE CONTEXT OF PET PROJECT]

## PLANNING

- COLLECTION
- DATA MERGING
- GENERATE NEW DATA
- FEATURE GENERATION
- DATA SELECTION

## PRE-PROCESSING

- PROFILING
- FORMATTING
- CLEANING
- FILTERING
- COMPLETION
- SAMPLING
- CORRECTION
- STANDARDIZE
- NORMALIZE
- TRANSFORM
- AUGMENT
- VALIDATE

## ANALYSIS

- VISUALIZATION
- CORRELATION
- SKEWNESS
- DISTRIBUTION
- CLUSTERING
- COHORTS
- DIMENSIONALITY
- OUTLIERS/NOISE
- TRENDS/PATTERNS
- STORY POINTS
- INTERPRETATION
- DOCUMENTATION

## \*ADDITIONAL STEPS\*

### REAL-WORLD ENTERPRISE DATA SETUP

#### AVAILABILITY (ON-PREM/CLOUD)

#### AUTOMATED DATA WORKFLOWS (DATA LOADS)

#### GOVERNANCE

#### SECURITY

#### INTEGRITY

#### OWNERSHIP

#### CERTIFICATION

#### INFRA STRUCTURE

#### STORAGE & COMPRESSION

## MY ARTIFACT CHECKLIST

---&gt; FOR DATA PHASE OF PET PROJECTS

- |   |  |  |
|---|--|--|
| <input type="checkbox"/> DATA SOURCES *               | <input type="checkbox"/> LIST OF ENRICHED/ NEW DATA/FEATURES * | <input type="checkbox"/> EXPLORATORY DATA ANALYSIS SUMMARY |
| <input type="checkbox"/> PROCESS FLOWS/ DATA WORKFLOW | <input type="checkbox"/> DATA QUALITY ISSUES LOG               | <input type="checkbox"/> EDA* KEY FINDINGS                 |
| <input type="checkbox"/> DATA ASSUMPTIONS*            | <input type="checkbox"/> QUICK SUMMARY OF TOOLS/PACKAGES       | <input type="checkbox"/> CONCLUSION ON DATASET             |
| <input type="checkbox"/> DATA PREP* TECHNIQUES        | <input type="checkbox"/> LIST OF DEPENDENCIES                  | <input type="checkbox"/> NEXT STEPS                        |

\* MAKE A POINT TO DISCUSS YOUR DATA &amp; ANALYSIS TECHNIQUES IN YOUR INTERVIEWS \*

#9

# DESIGN YOUR ARCHITECTURE

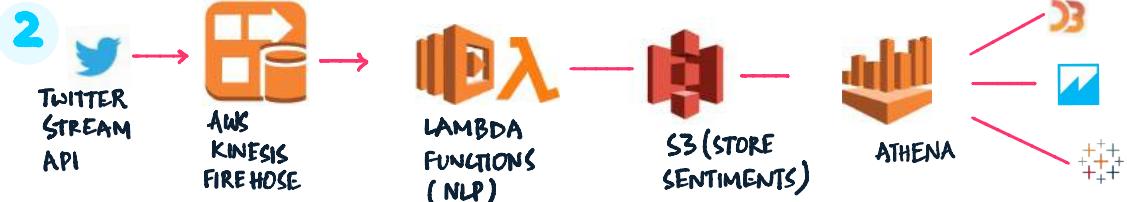
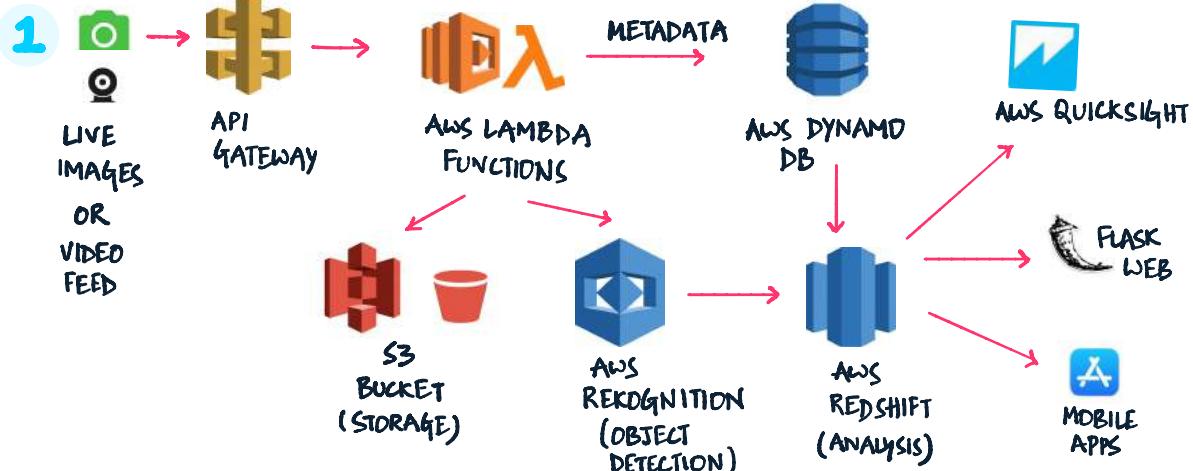


## \* DESIGN/SOLUTION ARCHITECTURE \*

DATA | SYSTEM | MODEL |  
ANALYTICS | INTELLIGENCE



## EXAMPLE SOLUTION DESIGNS (USING AWS)



\* DISCUSS YOUR DESIGN & ARCHITECTURE IN YOUR INTERVIEWS \*

# RECAP + UPCOMING



## Pet-Projects Post Series

Post 0 - Pet Project Intro

Post 1 - 3 Ingredients of a successful pet-project

Post 2 - Define Problem

Post 3 - Choose a Methodology

Post 4 - Gather Knowledge

Post 5 - Data Essentials

- Post 5.1 - Data Essentials I
- Post 5.2 - Data Essentials II
- Post 5.3 - Know your Data
- Post 5.4 - Know your Data - Public Datasets
- Post 5.5 - Data in a Nut-shell

Post 6 - Implementation

- Post 6.1 - Design Architecture - Using AWS examples (\*\* we are here \*\*)
- Post 6.2 - Dig Deeper into Design (Architecture)
- Post 6.3 - Model Know-how's - Essentials 101
- Post 6.4 - Model Engineering - Compute (GPU/TPU), ML Frameworks, AutoML
- Post 6.5 - Feature Engineering (An Art)
- Post 6.6 - Coding Essentials
- Post 6.7 - Model Artifacts / My checklist
- Post 6.8 - Deployment / Scale / CI-CD (Continuous Integration / Deployment pipelines)
- Post 6.9 - Model Inferences / Model Serving / End Points / Drift / Monitoring / Logging

Post 7 - Deliver Insights / Inferences - Visualizations / End User Interfaces - Interaction Touch points

BONUS - Bringing it all together - Final Pet-project Report\* (Showcase / Git / Portfolio)

LESSONS  
FROM MY  
JOURNEY  
LEARNING  
EXPERIENCE

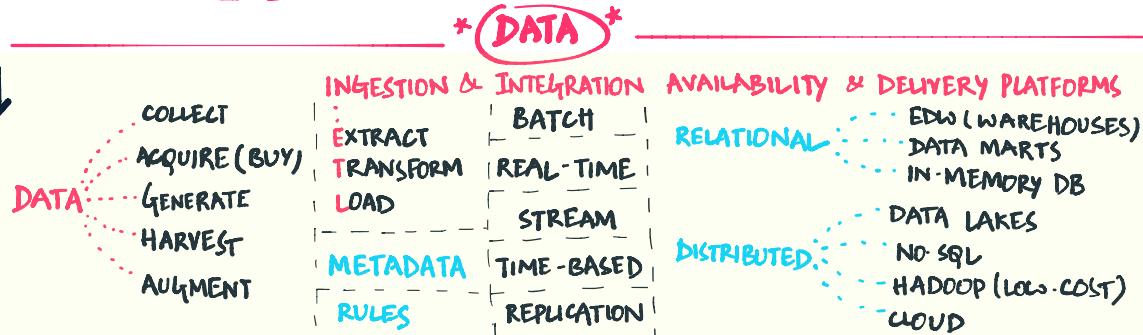
THANK YOU  
FOR YOUR INCREDIBLE  
SUPPORT\*\*

\*\* Thank you! \*\*

# #10 SYSTEM

- PARTS
- STEPS
- COMPONENTS
- MODULES
- PROCESS
- INTERFACES
- ARCHITECTURE
- INTERACTIONS

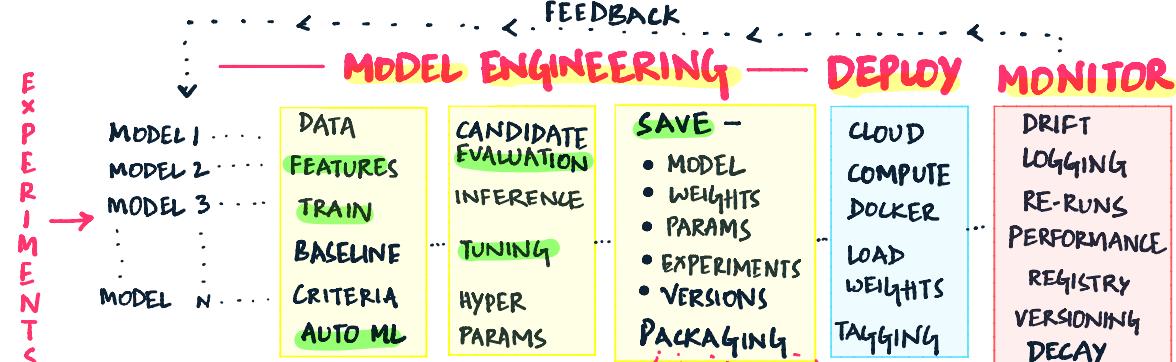
# HOLISTIC SYSTEM ARCHITECTURE [TOP-DOWN]



# DATA GOVERNANCE

- ## DATA MANAGEMENT

# DATA PREP & ANALYSIS



# JAR PKL ÖNNX 'HDF5 MODEL SERVING (ML OPS)

-

# #11 MODEL ESSENTIALS 101

G IS  
YOUR BEST  
FRIEND

- ARTIFACT TRAINED TO DETECT CERTAIN TYPE OF PATTERNS IN DATA
- ALGORITHMS THAT IMPROVE AUTOMATICALLY THROUGH EXPERIENCE

## KEY ELEMENTS OF ML

REPRESENTATION  
KNOWLEDGE

OPTIMIZATION  
SEARCH PROCESS

EVALUATION  
CANDIDATE MODELS (HYPOTHESIS/RECIPES)



## OPTIMIZERS

- ADAM
- GRADIENT DESCENT
- MOMENTUM
- RMSPROP
- SGD
- ADAGRAD

## ACTIVATION FUNCTIONS

- RELU
- TANH
- GAUSSIAN
- SIGMOID
- IDENTITY
- BINARY STEP

## EVALUATION METRICS (BASED ON MODEL FAMILY)

- AUC — AREA UNDER ROC CURVE
- MICRO AVERAGED F1 SCORE
- ACCURACY, PRECISION, RECALL
- LOG LOSS / ERROR
- MSE, RMSE, MAE, R-SQUARED
- CUT-OFF

## EXPLAINABILITY REPRODUCE

## MODEL

CHOOSE  
MODEL  
FAMILY

- SIZE
- TYPE OF PROBLEM
- RANDOMNESS
- PARAMETERS

MODEL  
PARAMETERS

- FEATURES
- LEARNED  
DIRECTLY FROM  
TRAINING

HYPER  
PARAMETERS

- LEARNING RATE
- EPOCHS / PASSES
- DECIDED BEFORE  
LEARNING

## TUNING

- CROSS-VALIDATION (K-FOLD)
- GRID SEARCH

## OVER-FITTING

- DROP-OUT
- REGULARIZATION (L1 & L2)
- DATA SHUFFLING

## PREDICTIONS

- REAL-TIME
- BATCH
- ONE-OFF / LOCAL

## ETHICS BIAS

#II

# MODEL ESSENTIALS 101\*

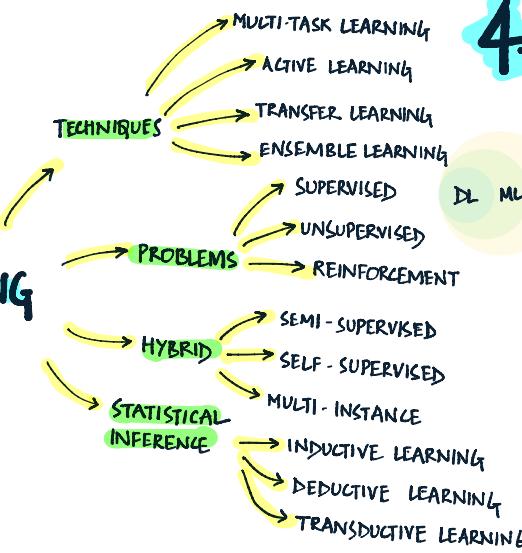
- ARTIFACT TRAINED TO DETECT CERTAIN TYPE OF PATTERNS IN DATA
- ALGORITHMS THAT IMPROVE AUTOMATICALLY THROUGH EXPERIENCE

## 4 TYPES OF AI



- REACTIVE MACHINES
  - PERCEIVE THE WORLD DIRECTLY AND ACT ON WHAT IT SEES.
  - NO ABILITY TO FORM MEMORIES OR USE PAST EXPERIENCES
- LIMITED MEMORY
  - TYPE-II MACHINES THAT CAN LOOK INTO PAST
  - MEMORIES AREN'T SAVED AS PART OF LIBRARY OF EXPERIENCE
- THEORY OF MIND
  - THOUGHTS, EMOTIONS, CO-EXISTANCE THAT EFFECT THEIR OWN BEHAVIOR
- SELF-AWARENESS
  - TYPE-III SENTIENT MACHINES (FUTURISTIC)
  - FORM REPRESENTATIONS ABOUT THEMSELVES
  - CONSCIOUS BEINGS - AWARENESS OF THEIR STATE & OTHERS FEELINGS

## LEARNING

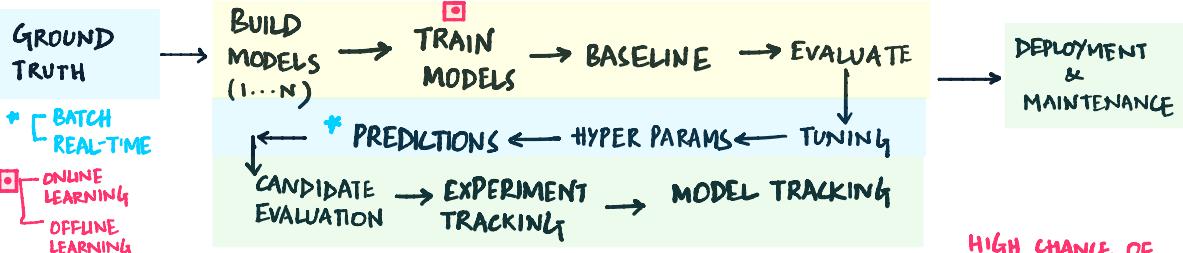


## 4 TYPES OF AI



- REACTIVE MACHINES
  - NO MEMORY / PAST EXPERIENCE
- LIMITED MEMORY\*
  - CAN LOOK INTO PAST.
- THEORY OF MIND
  - THOUGHTS, BEHAVIOR
- SELF-AWARENESS
  - CONSCIOUS · SENTIENT

# #12 MODEL\* ENGINEERING AT SCALE



## BASELINE

- TEST ERROR ... REDUCE
- VALIDATION SET ERROR
- TRAINING ERROR
- AUTO ML
- HUMAN ERROR
- BAYESIAN ERROR

## HYPER PARAM SEARCH\*

- LEARNING RATE\*
- MODEL SIZE
- NUMBER OF EPOCHS\*
- WEIGHTS
- MODEL DEPTH\*
- LAYER PARAMS
- OPTIMIZER CHOICE
- BATCH SIZE
- LOSS FUNCTION\*
- REGULARIZATION WEIGHTS

HIGH CHANCE OF SUCCESS  
AUTOMATED  
HYPERAS  
SIGOPT  
W&B

## RESOURCES (COMPUTE)

- ON-PREM (CUDA - GPU/LAMBDA)
- CLOUD\* (AWS, GCP, AZURE)

- DISTRIBUTED TRAINING (MULTIPLE NODES, GPUs, CPU)
- PARALLEL DATA
- MODEL PARALLELISM

## EXPERIMENTS

- RECORD & TRACK ALL EXPERIMENTS
- SYNCH & MANAGE (ML FLOW, TENSORBOARD)

\*MAKE A POINT TO DISCUSS ABOUT SCALING IN YOUR INTERVIEWS \*

## IN THE CONTEXT OF PET PROJECTS\*

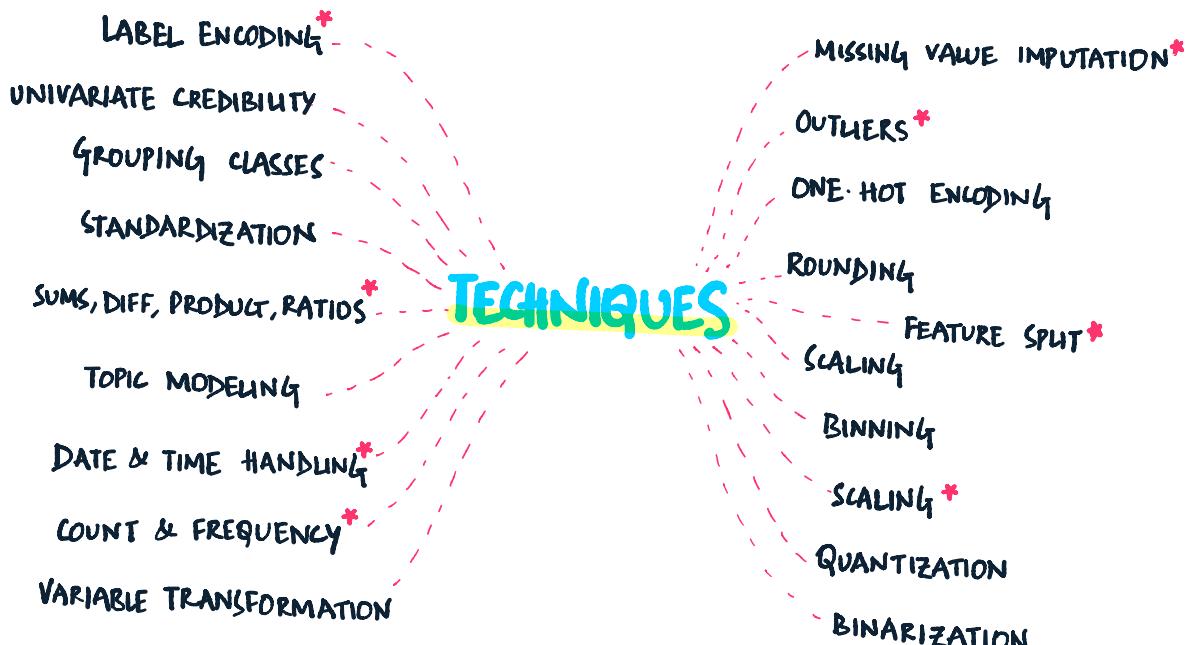
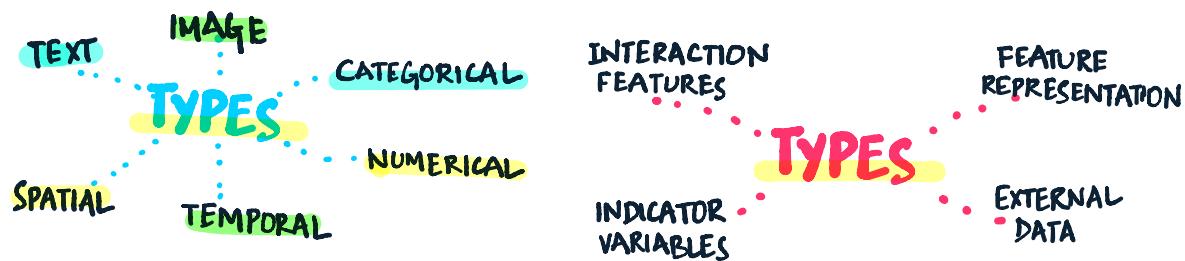
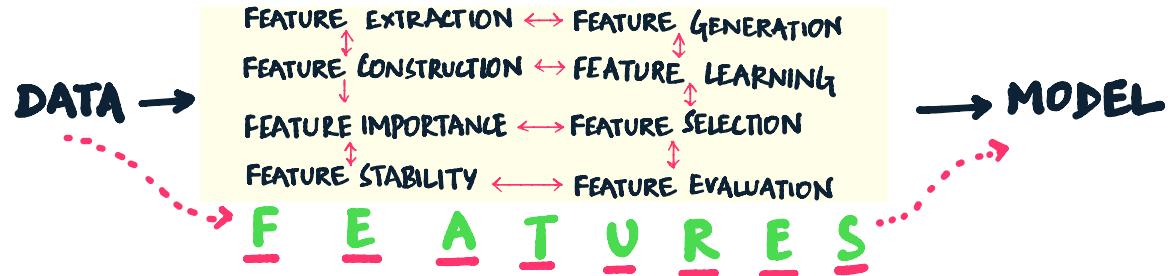
- GET BASELINE METRICS
- AUTO ML FRAMEWORKS
- BUILD MULTIPLE MODELS FROM DIFFERENT FAMILIES
- EVALUATE USING STANDARD METHODS (PRECISION, RECALL, CONFUSION MATRIX, LOSS ETC.)
- BIAS-VARIANCE TRADE OFF & DISTRIBUTION SHIFT
- ERROR ANALYSIS
- FEATURE SELECTION & ENGINEERING
- HYPER PARAMETER OPTIMIZATION
- ENSEMBLES (BAGGING, BOOSTING, STACKING)
- PICK THE BEST MODEL BASED ON YOUR SUCCESS CRITERIA
- USE COLAB/GPU SUPPORTED ENVIRONMENT

# #13 FEATURE ENGINEERING 101\*

G IS  
YOUR BEST  
FRIEND

## WHY?

- MAKES THE MODEL UNDERSTAND THE INTRICACIES OF THE PROBLEM
- REPRESENTS THE MOST MEANINGFUL ATTRIBUTES / RELATIONS / LEARNING



TO BE CONTINUED xx

# #14 FEATURE ENGINEERING DEEP DIVE



## 1 MISSING DATA IMPUTATION

- COMPLETE CASE ANALYSIS
- MEAN/MEDIAN/MODE
- RANDOM SAMPLE
- MISSING VALUE INDICATOR
- REPLACEMENT BY ARBITRARY VALUE
- MULTIVARIATE IMPUTATION

## 3 OUTLIERS

- REMOVAL
- TREAT AS NAN
- CAPPING
- WINSORIZATION (TOP/BOTTOM/ZERO CODING)

## 6 FEATURE SCALING

- STANDARDIZATION
- MIN-MAX SCALING
- MEAN SCALING
- MAX ABSOLUTE SCALING
- UNIT-NORM SCALING

## 9 DATE & TIME

- DAYS WEEKS
- MONTHS QUARTERS
- YEARS
- WEEKDAYS
- LEAP YEARS
- TIME DELTA
- DAY OF MONTH
- CALENDAR/FISCAL
- TIMESTAMP/SPECIAL

## 10 FEATURE CREATION (GROUP OF FEATURES)

- SUM
- MINUS
- PRODUCT
- MEAN
- MIN
- QUOTIENT
- ABS
- RATIDS

## 2 CATEGORICAL ENCODING

- ONE-HOT ENCODING
- COUNT & FREQUENCY
- ORDINAL ENCODING
- TARGET ENCODING
- WEIGHT OF EVIDENCE
- RARE LABEL ENCODING
- BASE N
- FEATURE HASHING

## 4 IMAGES

- PIXELS
- LINES
- EDGES
- THRESHOLDS

## 5 VARIABLE TRANSFORMATION

- BOX-LOG
- LOG-LOG( $x$ )
- RECIPROCAL  $1/x$
- SORT  $\sqrt{x}$
- EXPONENTIAL
- YEO-JOHNSON

## 7 TEXT

- TOPIC EXTRACTION
- BAG OF WORDS
- TFI·DF
- N-GRAMS
- WORD2VEC

## 8 DISCRETISATION

- EQUAL FREQUENCY
- EQUAL LENGTH
- WITH TREES
- WITH CHIMERGE

## 11 FEATURE STABILITY

- PEARSON'S COEFFICIENT
- JACCARD'S INDEX
- SYMMETRICAL UNCERTAINTY
- SPEARMAN'S RANK
- CANBERRA DISTANCE

# #15 WRITE WORLD-CLASS CODE\*

## FULLY FUNCTIONAL

- ZERO TO MIN BUGS
- EXCEPTION HANDLING
- SOLVES THE PURPOSE
- SINGLE POINT OF FOCUS
- SIMPLIFY, SPLIT, KISS

## READABILITY

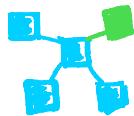
- INDENTATION
- STYLE GUIDES
- STANDARDS
- NAMING
- LINTING
- CODING PATTERNS
- TOOLS/IDE/AUTOMATION

## CLARITY OF THOUGHT/FLOW

- WORKFLOWS
- CONNECT THE DOTS
- SYSTEMS WORKING COHESIVELY

## SELF-DOCUMENTING

- MINIMIZE OVERHEAD
- STANDARDIZE COMMENTS
- EXPRESSIVE



## EXTENSIBILITY

## MODULARITY (PLUG & PLAY)

- SELF-CONTAINED

## REFACTORIZATION



## MAINTAINABILITY

- DECODEABLE
- UNDERSTANDABLE
- WELL-STRUCTURED
- LESS DEPENDENCIES
- LONG-LIFE

## DEBUGGING / LOGGING

- LOGGER MODULES
- CHECKPOINTS
- COMMON ERRORS/KNOWN ISSUES

## VERSIONING & BACKUPS (LOCAL & CLOUD)

## TESTABILITY

- UNIT TESTING
- INTEGRATION
- REGRESSION
- SMOKE TESTS

THINK OF ART. BEAUTY, FLOW, SERENITY, CALMNESS, CANDID, COMPOSED,  
PLEASURE & HEAVEN — THAT'S HOW YOUR CODE SHOULD FEEL — SPECIAL\*

#16

# DEPLOY FOR SCALE

IS THE  
"MAGIC BULLET"

MODEL  
PACKAGE

TRAINED  
TESTED  
SAVED

CONFIG & DEPENDENCY  
MANAGEMENT

DOCKER  
IMAGE  
REGISTRY

KUBE  
ORCHESTRATION

MICRO SERVICES

CI/CD



AWS  
FARGATE

DEPLOY & SERVE

ON-PREMISES

AWS (ECR & LAMBDA)  
GCP (GCR)

PYTORCH

TENSORFLOW  
(PROTOS)

AZURE

INFERENCE

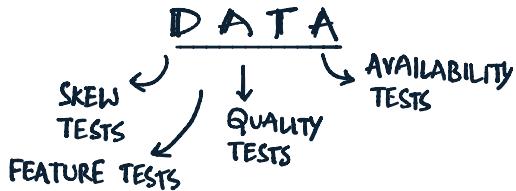
EDGE  
DEVICES

HTTPS  
REST API

APPS

DATA  
LAKES

END POINTS



- MODEL BUILD TESTING [AUTOMATED\*]
- UNIT TESTING
  - INTEGRATION TESTING
  - INFRASTRUCTURE TESTING
  - RELIABILITY TESTING
  - PERFORMANCE TESTING
  - SYSTEM TESTING
  - EVALUATION TESTS
  - BIAS/FAIRNESS
- CONTINUOUS INTEGRATION

DEV → QA → STAGING → PROD\*

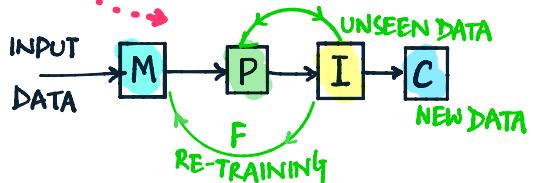
- PACKAGING
- DEPENDENCIES
- VERSIONING
- IMAGE/CONFIG CONTAINERS
- ORCHESTRATORS
- SCHEDULING / EVENT TRIGGERS
- REPRODUCIBILITY TESTING
- TEST SCORING (READINESS)
- SANITY CHECK POINTS
- AUTO SCALING
- FAULT TOLERANCE / Fallback

CONTINUOUS DELIVERY

\*KEYWORDS\*

SERVERLESS CONTAINERS SERVING REGISTRY LOAD BALANCING VERSIONS COST  
MICROSERVICES ORCHESTRATION REFITTING ACCELERATORS TAGGING SCORING

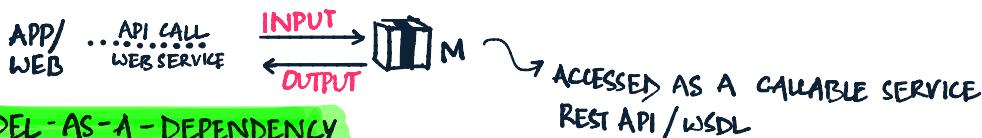
# #17 SERVE MODEL\*



M - MODEL P - PREDICTIONS F - FEEDBACK  
I - INSIGHTS C - CONSUMPTION

## TYPES OF SERVING (ML OPS)\*

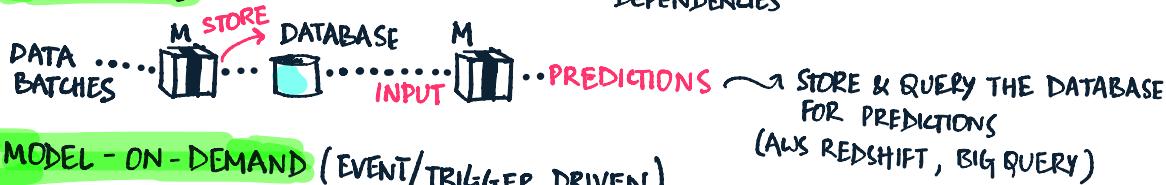
### MODEL-AS-A-SERVICE



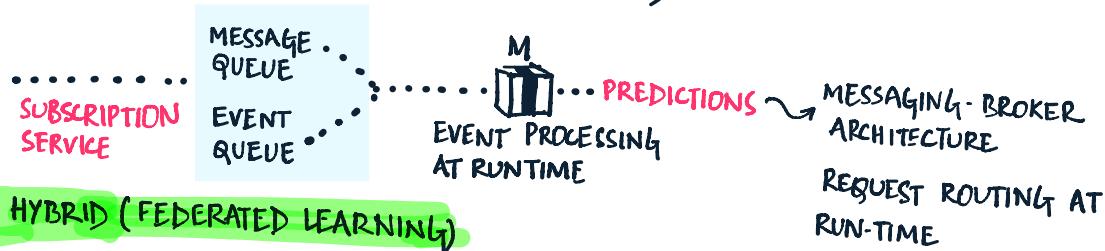
### MODEL-AS-A-DEPENDENCY



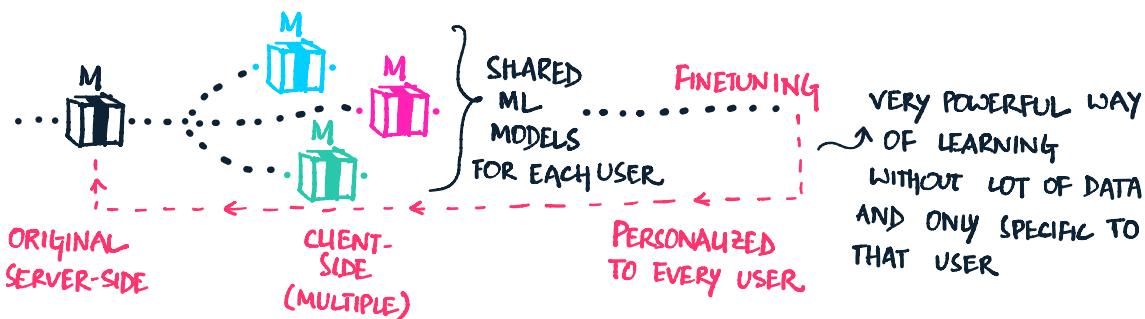
### PRECOMPUTE



### MODEL-ON-DEMAND (EVENT/TRIGGER DRIVEN)



### HYBRID (FEDERATED LEARNING)



# #18 CONTROL DRIFT

CHANGE IN THE DIRECTION OF THE PERFORMANCE OF MODEL

DEGRADATION OVER TIME

## TYPES OF DRIFT

### CONCEPTUAL / CONTEXTUAL

- TRAINING DATA NO LONGER REPRESENT THE PREDICTIONS OR CORRECT LEARNING
- CHANGE OF CONTEXT / APPLICABILITY OF PREVIOUS TRAINING RULES

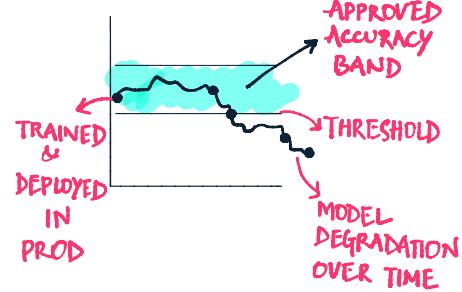
### DATA

- CHANGE IN THE BEHAVIOR OF DATA (AS MORE DATA GETS ADDED)
- DATA QUALITY ISSUES
- CHANGE OF ENVIRONMENT, POLICIES OR ANYTHING THAT IMPACTS DATA

## MODEL [DETECT DRIFT]

## MONITORING

- HYPOTHESIS TESTS
- CHI-SQUARED, P-VALUE, T-TESTS
- DISTANCE MEASURES
- KULLBACK-LIEBNER (KL) MEASURE
- JENSEN-SHANNON DIVERGENCE
- COMPARE WITH GROUND TRUTH
- SUMMARY STATS & INPUT DISTRIBUTIONS
- ROC CURVE, MANN-WHITNEY U & GINI
- DATA IN TRAIN VS. DATA IN PROD



### FEATURE / COVARIATE DRIFT

- CHANGE TO THE UNDERLYING FEATURES
- NUMERICAL VARIANCE
- DISTRIBUTION SHIFT

### DUAL DRIFT

[COMBINATION OF ABOVE]

## MODEL [KEEP IT GOING]

## MAINTENANCE

- AUDITING
- LOGGING
- PROFILING
- ALERTING
- EVENT/TRIGGERS
- A/B TESTS
- SPLIT TESTS

\* FAMOUS OPEN-SOURCE MONITORING METRICS TOOLS ARE PROMETHEUS & GRAFANA.  
CLOUD OFFERING AWS SAGEMAKER MODEL MONITOR

# #19 DELIVER INTELLIGENCE

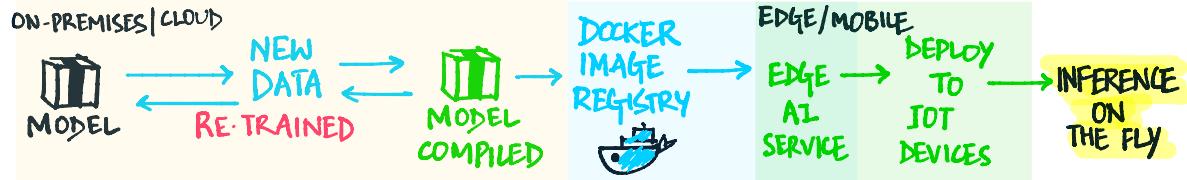


\* BUILD "INTELLIGENCE" END USER INTERFACE AS PART OF YOUR PET-PROJECT \*

# #20 GO LITE (EDGE)

DEPLOY YOUR PROJECT ON EDGE/MOBILE  
FOR ON-THE FLY PREDICTIONS

- INEXPENSIVE
- CONNECTIVITY
- LOW POWER DEVICES
- NO LATENCY
- PRIVACY & SECURE
- LIMITED PROCESSING POWER



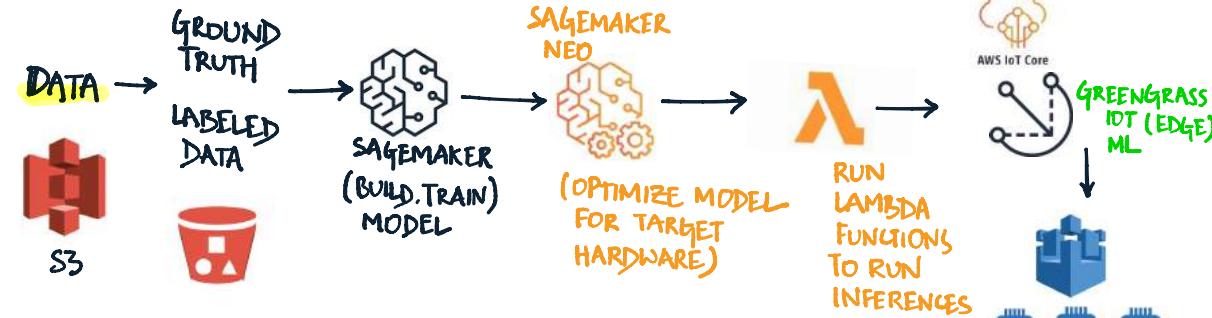
## EDGE AI FRAMEWORKS



## TF WORKFLOW



## AWS WORKFLOW



\* SHOW, DON'T TELL — BUILD AN EDGE INTERFACE TO SHOW YOUR PET-PROJECT \*

# #21 GO WEB (APPS, API's)

GIVE A FACE TO YOUR AI/ML PROJECT  
BUILD A USABLE WEB APP INTERFACE

START HERE

M  
...  
TF TRAINED  
PYTORCH  
PYTHON  
KERAS  
SPARK ...

ARCHITECTURE  
+ WEIGHTS + OPTIMIZED STATE (PICKLE, HDF5, PROTOBUF...)  
ONLY WEIGHTS  
JSON / YAML

DOCKER / KUBERNETES — CONFIGURE IMAGE REGISTRY

S E R V E

WEB FRAMEWORKS

FLASK \*  
DJANGO  
REACT / ANGULAR  
GOOGLE APP ENGINE  
HTML + CSS + JS  
BOOTSTRAP \*  
NODE.JS  
TF.JS

GET PUT POST DELETE  
FLASK  
REST API

CREATE CALLABLE API  
2 MAIN COMPONENTS  
APP.PY \*  
MAIN / INDEX.HTML

BOOTSTRAP / HTML UI TEMPLATE

POSTMAN (TEST THE END POINT WHICH IS EXPOSED VIA FLASK API)

\* STREAMLIT / DASH

BUILD WEB APP / VISUALIZATION APP (IGNORE IF GOING FLASK + BOOTSTRAP UI)

DEPLOY TO HEROKU USING GEMFURY OR \* GITHUB \* REPO OR GUNICORN

PROFILE  
(FOR BINDING)

HEROKU DYNOS (512MB)

REQUIREMENTS.TXT  
(FREEZE PACKAGES)

(PLATFORM AS A SERVICE)

SERVE PREDICTIONS

END · USER (BROWSER, MOBILE, APP) ← →

