CERTIFAI
powered by Skymind

Data Analysis and Machine Learning

Building AI Projects with Machine Learning

Data Analysis for AI

How to Build an AI Project?

AI Project Workflow

Define Problem

Define problem statement

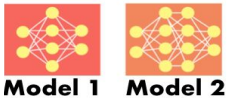
Collect data

Data Collection

Preprocess data

Clean and preprocess data

Train models



Build models depends on

- Problem domain
- Amount of data

Decide whether to build

- Classification
- Regression

Evaluate models

Monitor the performance of the model (Retrain the model if necessary)

Deploy models

Apply the trained model to predict new data

Monitor predictions

Monitor the production environment to ensure relevant to get accurate predictions

STEP 1: Problem Definition

AI project starts with defining what problems to solve

Define Problem Statement

Define Problem Statement

Which Vertical?

Industry

Medical

Manufacturing

Retail & Commerce

Financial

Others

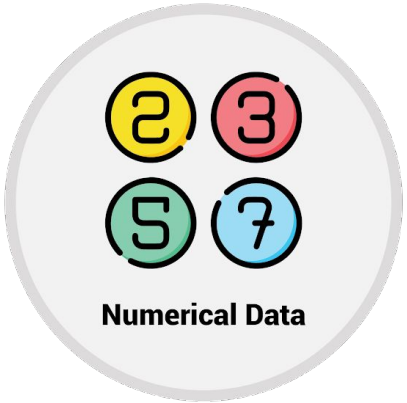
CERTIFAI
powered by Skymind

STEP 2: Data Collection

AI project continue with searching for data to work on

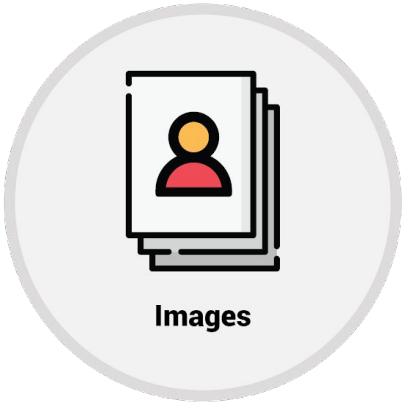
Which data

Types of Data & AI Use Cases



Numerical Data

Measurable attributes of an item. Normally stored as txt or csv format. People might refer to as csv data.



Images

Abundant pictures about objects or scenes. Normally grey images is sufficient.

Text

Language-based unstructured data. Example: English, Malay and Spanish.

Sequential / Time Series Data

Aforementioned data with the addition of time element.

Which Data?

Finding data sources

- Obtain datasets from research labs, project owner
- Ownself curate dataset (Setback: Difficult to obtain large amount of dataset)
  - Survey
  - Web scraping/crawling
- Open source dataset
  - https://www.kaggle.com/datasets
  - https://datasetsearch.research.google.com/
  - http://mlr.cs.umass.edu/ml/
  - https://www.visualdata.io/

kaggle

VisualData

Google Dataset Search

Data Analysis for AI

Data Preprocessing

STEP 3: Data Cleaning / Preprocessing

Reality of AI Project

> 80 percent of a data scientist's valuable time is spent simply finding, cleansing, and organizing data, leaving only 20 percent to actually perform analysis...
>
> IBM Data Analytics

Missing Data

Typical reasons why data is missing:

1. User forgot to fill in a field
2. Users chose not to fill out a field tied to their beliefs about how the results would be used or interpreted.
3. Data was lost while transferring manually
4. System error

Data Cleaning / Preprocessing

How to Clean / Preprocess data

1. Remove invalid value, NA or 0
2. Replace invalid value, NA with 0 or average value
3. Regular expression to remove unwanted symbols, characters / generalize terms
4. Categorize multiple data points into discrete groups
5. Encoding categorical data
    a. Convert Positive / Negative into 1/0

Feature Engineering

Features are measurable piece of data that can be used for modeling

Images

- Colors
- Edges
- Texture
- Outline of the object

Text Data

- Keywords
- Numbers
- Length of the text
- Position of the words

Speech

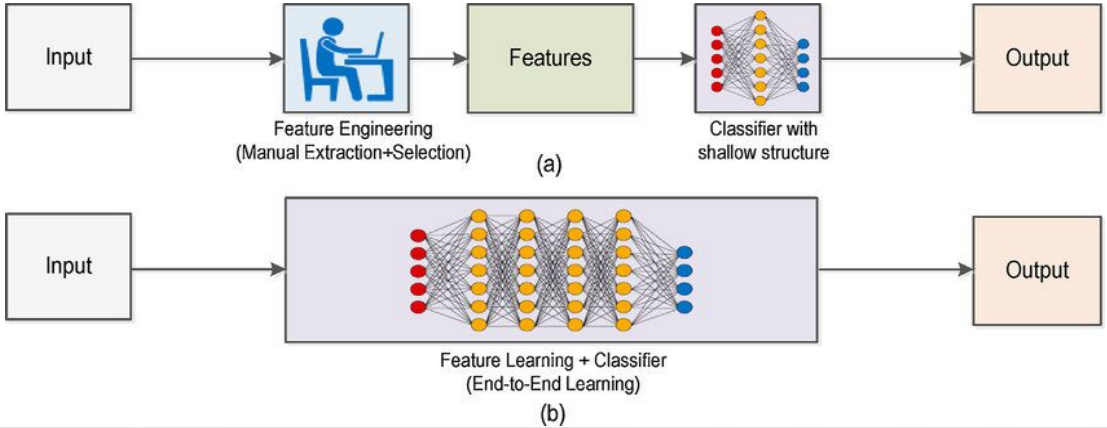
- Pitch
- Pause
- Use of words

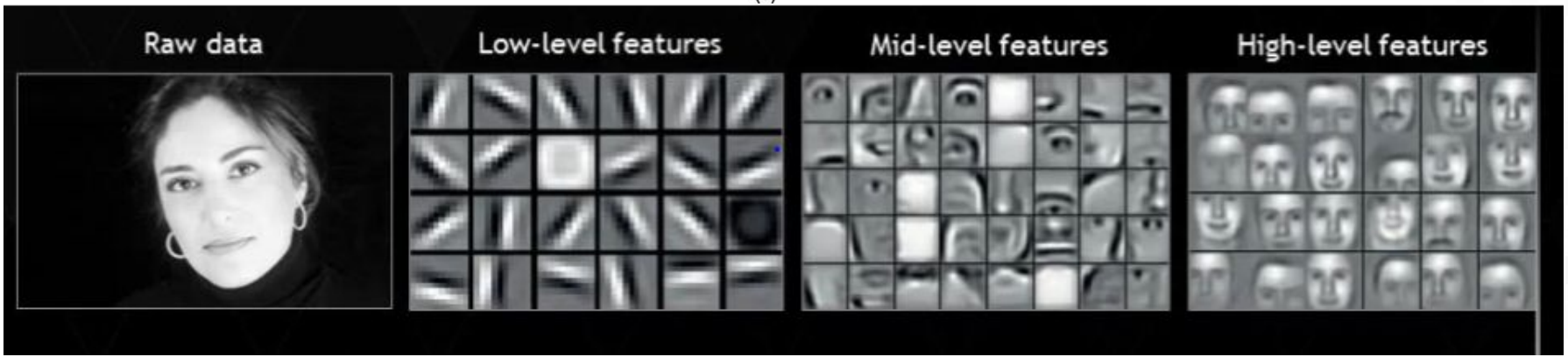
Feature Engineering

Feature Engineering - Putting domain knowledge into the selection or extraction of features and make patterns more visible to learning algorithms to work

Machine Learning	Deep Learning
Features to be selected and extracted by an expert	Automates the task of feature extraction
Example: Edges and lines in an image are extracted by calculating the gradients of pixel values i.e. Histogram of Oriented Gradients (HOG)	Example: CNN learns low-level features such as edges in early layers, to more complex features like shapes, and then high-level visual representations like faces
Performance mostly depends on how accurately the features are identified and extracted	Performance can be improved by training with more high quality data and fine-tuning hyperparameters

Feature Engineering



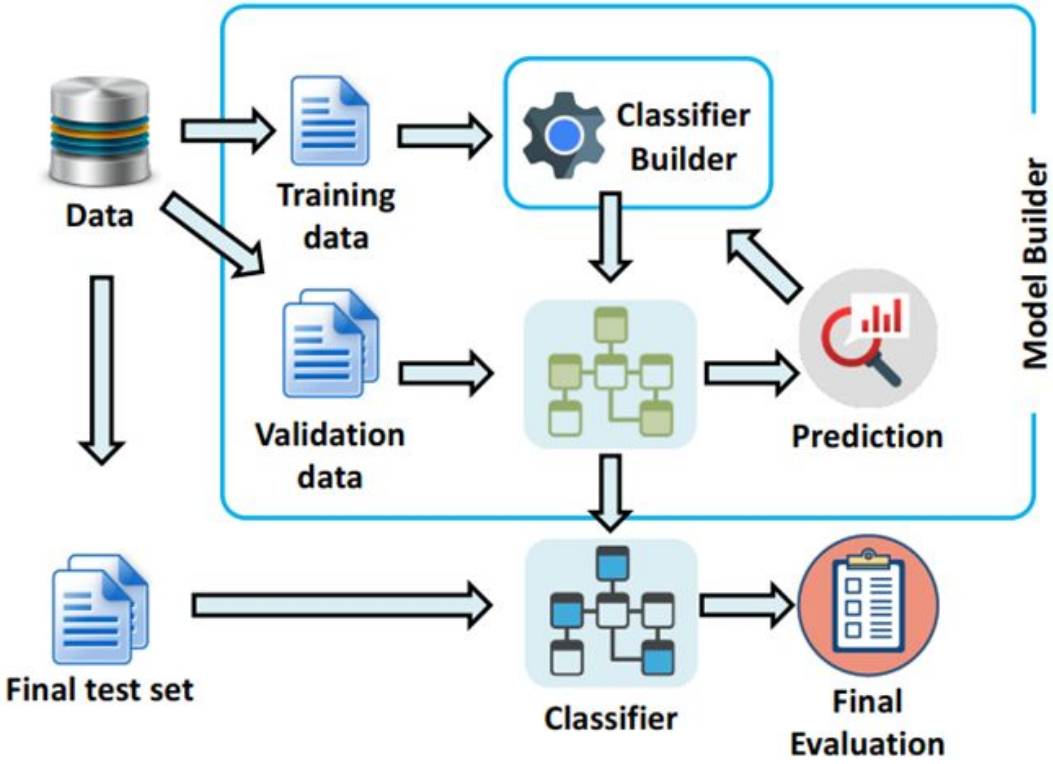


https://www.researchgate.net/figure/Comparison-between-two-techniques-a-traditional-machine-learning-b-deep-learning_fig2_322325843

Data Analysis for AI

Model Training

STEP 4: Training Models



Training, Dev and Test Sets

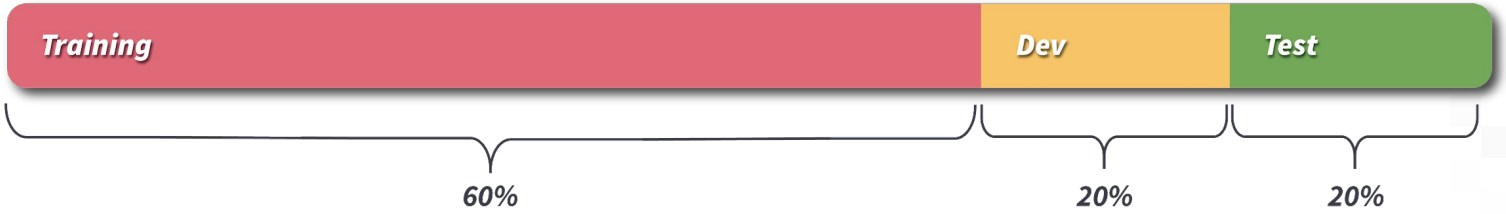
- In practice, data is splitted into training set, dev (development) set and test set.

- **Training Set** - Which our model is trained on.
- **Dev Set** - Which we use for parameter tuning, feature selection or changing to different models. Also called **hold-out cross validation set**.
- **Test Set** - Evaluate the performance of the models.

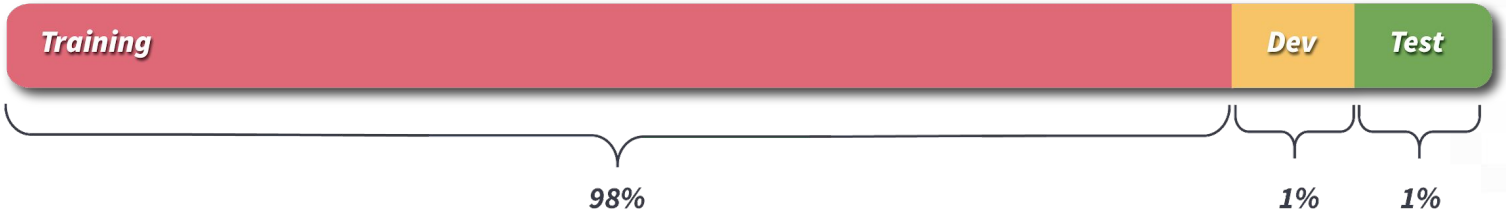
Setting Up the Right Dev and Test Sets

- **Choose the correct size for Train / Dev / Test Split**

Small Data Set (100 - 10,000 data points)



Big Data (1 Million data points)



Setting Up the Right Dev and Test Sets

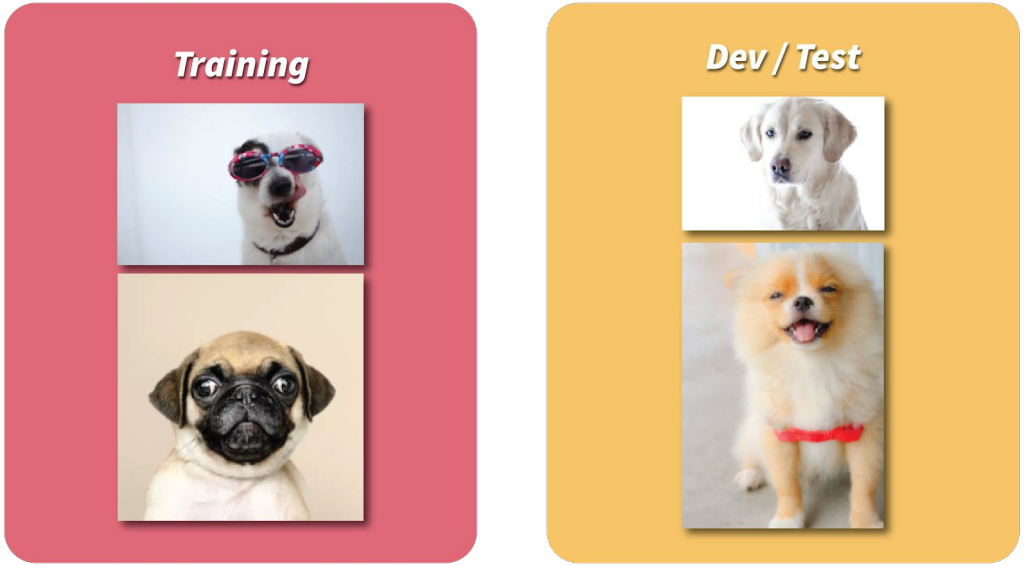
- **Make sure that the dev/test sets are sampled from the same distributions**
  - Let say you wish to build a vehicle classifier to classify boats, car, aeroplanes and motorcycle

Setting Up the Right Dev and Test Sets

- **Make sure that the dev/test sets are sampled from the same distributions**
  - But you use “boat” and “motorcycle” classes in the dev set, “car” and “aeroplane” classes in the test set.

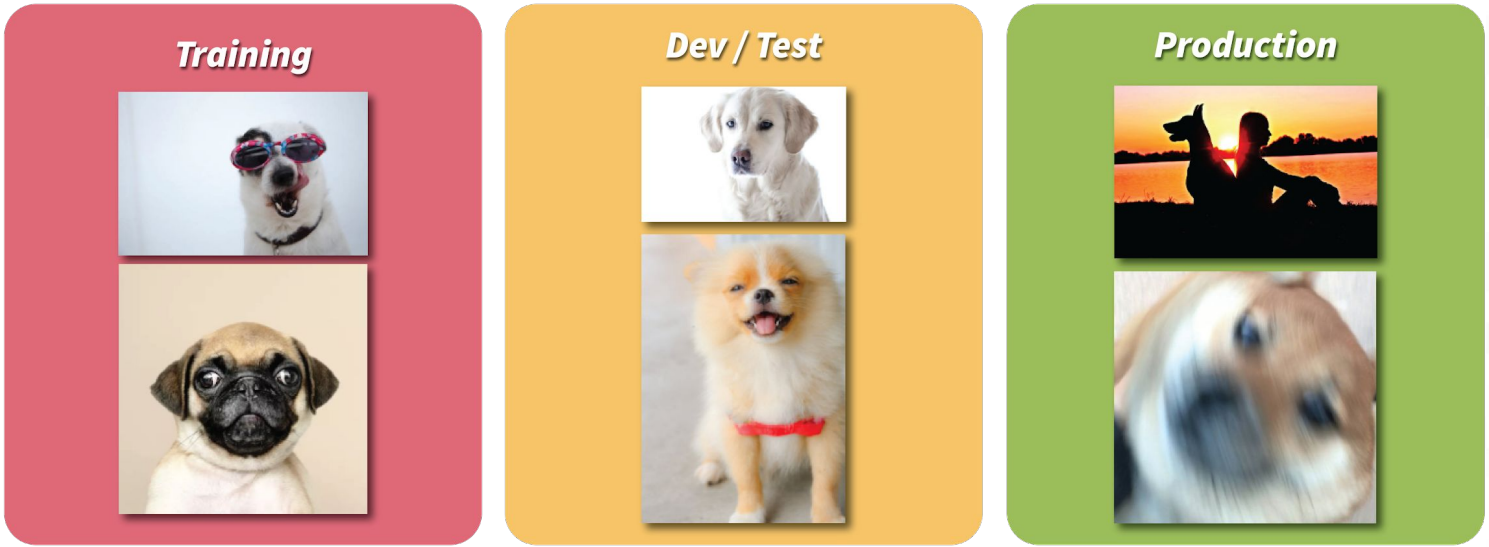
Setting Up the Right Dev and Test Sets

- **Make sure that the dev/test sets reflect the data in production**
  - Let say you want to build a dog classifier app for smartphone and achieve a high accuracy for the model:



Setting Up the Right Dev and Test Sets

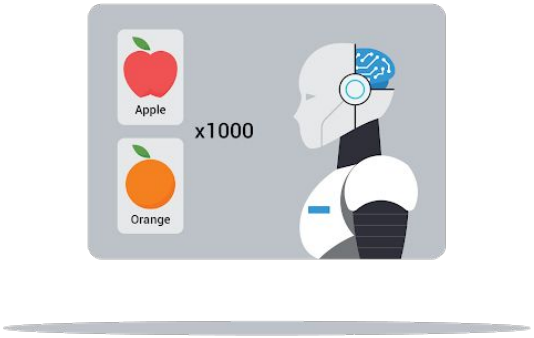
- **Make sure that the dev/test sets reflect the data in production**
  - During production, the image captured by users are blurry / differs from dev set, your model gets a very poor accuracy !



Modeling Techniques

SUPERVISED LEARNING

Learning from labeled data



A machine is trained on 1000 images of apples and oranges to identify type of fruit in new images.

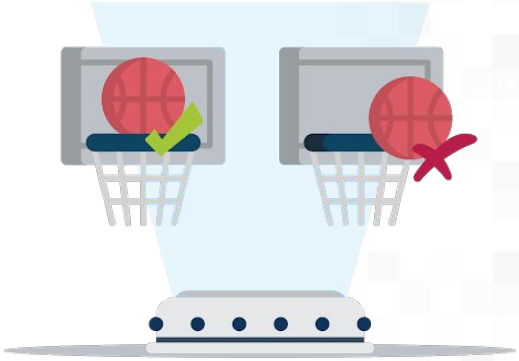
UNSUPERVISED LEARNING

Learning from unlabeled data

A machine groups objects based on the distribution of data.

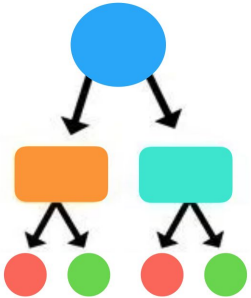
REINFORCEMENT LEARNING

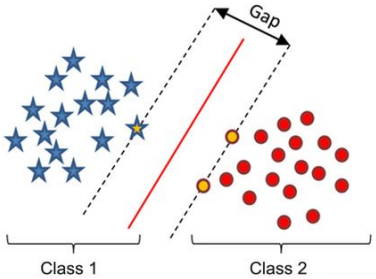
Software agents take actions to maximise cumulate award

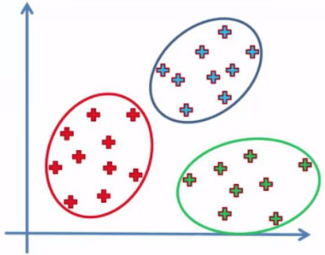


A machine is rewarded when it lands a ball into the basket, otherwise it is penalised.

Which Approaches?

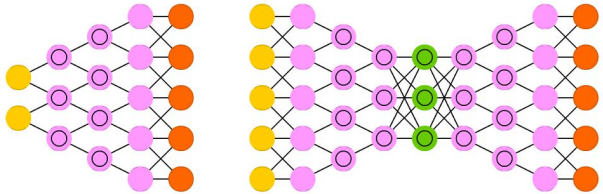


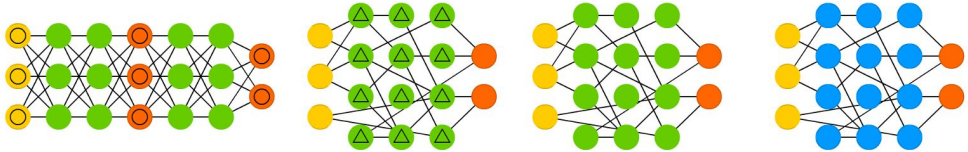




Machine Learning

Deep Learning





Deep Learning Frameworks

Made accessible to the public, via Open Source Software

They are written in different programming languages, some examples:

Data Analysis for AI

Model Evaluation and Deployment

STEP 5: Evaluate Model Performance

Predicted Value (predicted by the test)		Actual Value (as confirmed by experiment) positives	Actual Value (as confirmed by experiment) negatives
	positives	**TP** True Positive	**FP** False Positive
	negatives	**FN** False Negative	**TN** True Negative

Precision = $\frac{\text{True Positive}}{\text{Actual Results}}$ or $\frac{\text{True Positive}}{\text{True Positive + False Positive}}$

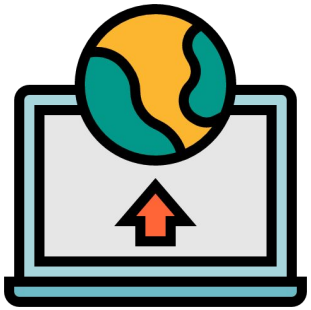
Recall = $\frac{\text{True Positive}}{\text{Predicted Results}}$ or $\frac{\text{True Positive}}{\text{True Positive + False Negative}}$

Accuracy = $\frac{\text{True Positive + True Negative}}{\text{Total}}$

STEP 6: Model Deployment

Key Steps:

- Plan for production deployment.
- Plan to monitor and maintenance.
- Organize the process and produce a report or prototype.
- Review project



Project Example: Predict Housing Prices

Data Collection	Data Pre-processing	Feature Selection + Data Splitting	Modeling	Model Evaluation
• Housing price • Number of rooms • Age • Weighted distance to employment centers • Accessibility index to highways / facilities • Population	• Remove invalid value • Feature selection	• Feature Selection • Split dataset into 70/30	• Linear Regression • Fit features to class to obtain coefficients	• MSE (Mean Squared Error) • Measure of quality of estimator, value close to 0 = good fit

Project Example: Classify Defective Objects

Data Collection	Image Pre-processing	Data Splitting	Build & Train model	Evaluate model
Sort photos into categories and name folders based on its class name (e.g.: good / defective)	• Convert to RGB • Resize image to 32x32 • Normalize pixels	Split training and test data into 80 /20	• Build convolution layer and NN layer • Train model with training dataset, tune parameters: batch_size, epochs	• Model Accuracy for classification • Overfitting • Model loss