جامعة الملك عبدالله
للعلوم والتقنية
King Abdullah University of
Science and Technology

أكاديمية كاوست
KAUST ACADEMY

How to Win a Data Science Competition

King Abdullah University of Science and Technology (KAUST)
KAUST Academy

Table of Contents

1. Introduction to Kaggle
2. Exploring AI Competitions
3. Mastering AI Competitions: Understand your Toolkit
4. Choosing the Best Models
5. Practical Tips and Strategies
6. Career Options in AI
7. 👀

Introduction to Kaggle

Welcome onboard kagglers

Kaggle.com

The Biggest AI community!

- Hosting AI competitions, datasets, notebooks, models, tutorials and a lot more.
- Best place to learn the practical part of AI.
- Ranks and Tiers.

Why do you have to compete in Kaggle?

- Improve your AI practical skills.
- Learn how to apply the state of the art.
- See the implications of your decisions during building the model (i.e. learn what overfitting really mean.) (btw, have you heard about shake up?👀)
- Compare yourself against the tops in the world.

  (You may be able to get over them).
- Get some prizes👀
- Fun :)

People talking nonsense about Kaggle on social media

Same people trying to take part in an actual Kaggle competition

Exploring AI Competitions

Have you heard about shakeup?👀

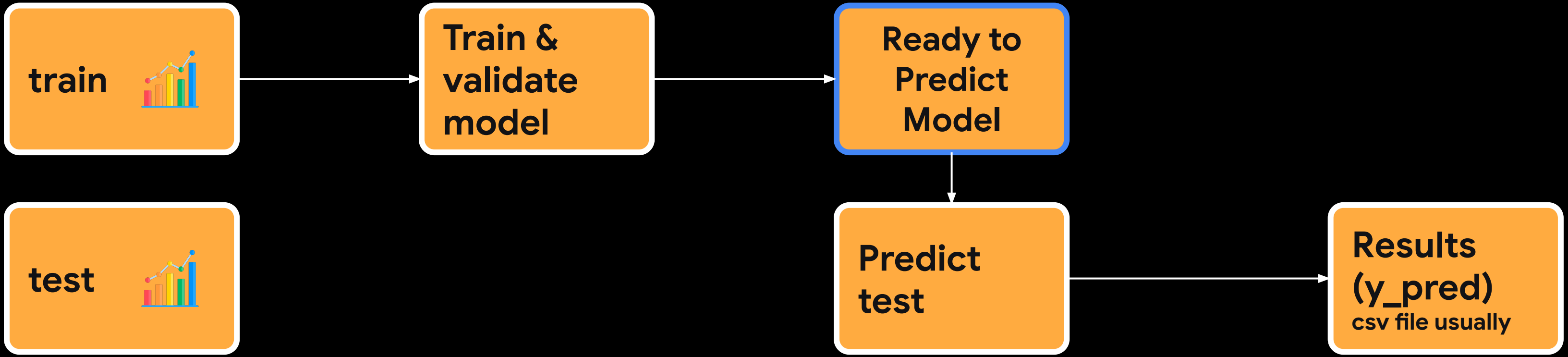
What are AI comps?

- You will be given a real-world task coming from any discipline (e.g. medicine, finance, sports, astronomy,..)
- You should gather some domain knowledge about the topic, read about it, then choose the model that best solves the problem.

You after the first submission:

What are AI comps?

- Dataset:
    - You will be given a two datasets for this task (or asked to gather a one lol).
    - Training and testing datasets.
    - You should train your model on the training data then make predictions for the testing data.
    - These predictions are the submission file for the competition.



train

Train & validate model

Ready to Predict Model

test

Predict test

Results (y_pred)
csv file usually

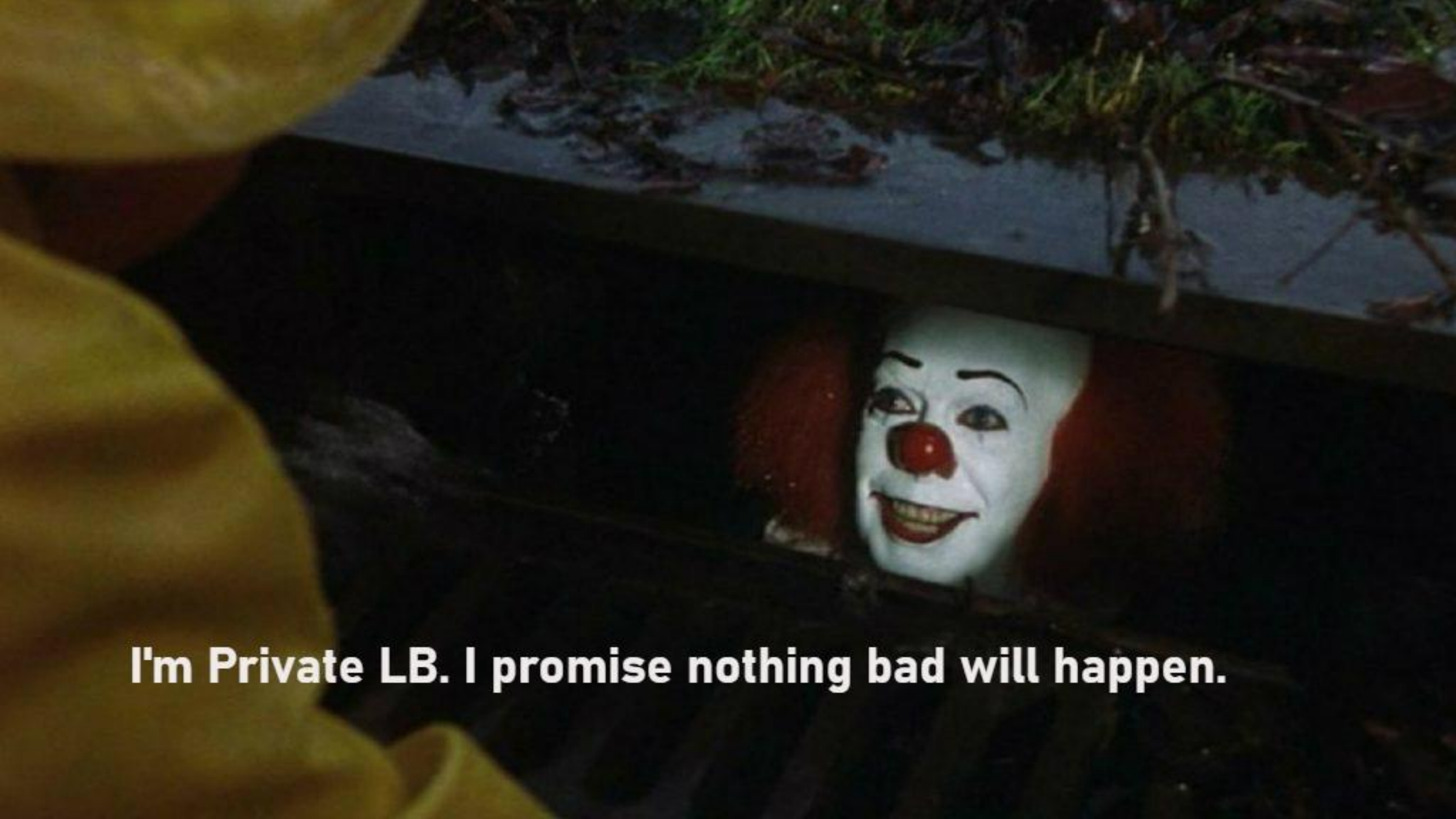
Example of Dataset samples

Index	Weight (g)	Wingspan (cm)	Price?	Back color	Species
1	100.1	125.5	10	Brown	Buteo jamaicensis
2	3000.7	200.0	98.1	Gray	Sagittarius serpentarius
3	3300.0	220.3	110.2	Gray	Sagittarius serpentarius
4	4100.0	136.0	154	Black	Gavia immer
5	3.0	11.0	2	Green	Calothorax lucifer

Leaderboard 🤔

- Leaderboard:
    - Oh bro, have you heard about shakeup?👀
    - There are 2 leaderboards. Public and Private.
    - Public LB: running while the competition is running.
    - Private LB: When the competition finishes the private lb will get revealed and the final ranking will depend on the private.
    - But why do we need this split for test data?🤨

	Price
Public Leaderboard	13495
	16500
	16500
Private Leaderboard	13950
	17450
	15250



I'm Private LB. I promise nothing bad will happen.

Leaderboard 🤔

- Leaderboard:
  - We need private LB to drop solutions that overfitted to the public LB.
  - This is to assure your solution is generalizable, working and reliable.
  - When many people overfits the public LB and the private LB get all over the place, we call this **shake up**.
  - But how to avoid overfitting?

    By having a robust validation (more details next).

You might overfit the leaderboard

	Price-True	Price_pred
Public Leaderboard	13495	13490
	16500	16562
	16500	165315
Private Leaderboard	13950	35134
	17450	20518
	15250	29516

AI Competitions Goals

- **Performance**: The current state of the art is the baseline for the competition. So in many times you have to beat the SOTA to win (or maybe not).
- **Efficiency**: There are many constraints in the competitions (e.g. limited GPUs, limited inference time,...)
- **Handle Big Data**: Sometimes data sets can go up to 700GB. How could you decrease this data size with minimal loss in information so you can handle it in your device? How to load it to your model fast enough to finish within a reasonable time? How to do inference later with this amount of data?

AI Competitions Goals

- **Ideal Dataset:** Sometimes the dataset for the competition is not enough for winning. You have to search for datasets in the internet or check pretrained models.
- **Avoid overfitting:** Having robust validation split would make your model robust against shakeups (I hope lol).

Rank 67 on Kaggle competition public leaderboard
Rank 212 on private leaderboard

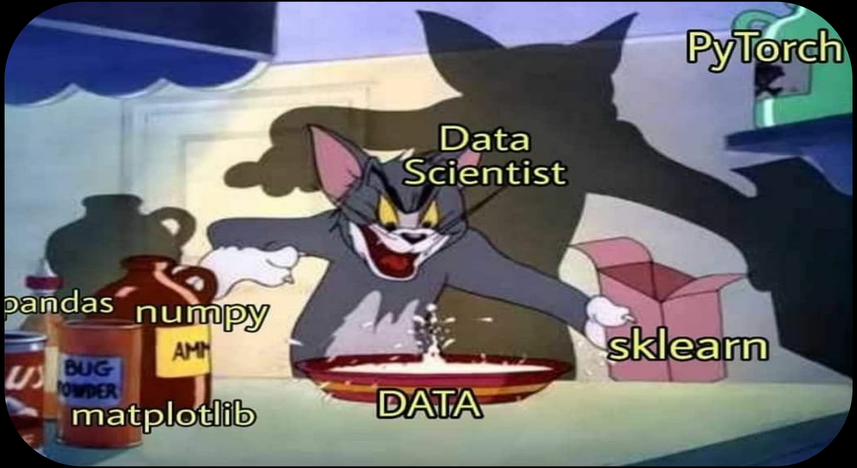
What are AI comps?

- Let’s explore it a bit:

  [Kaggle: Your Home for Data Science]

Mastering AI Competitions

Understand your Toolkit



Understand Your Toolkit

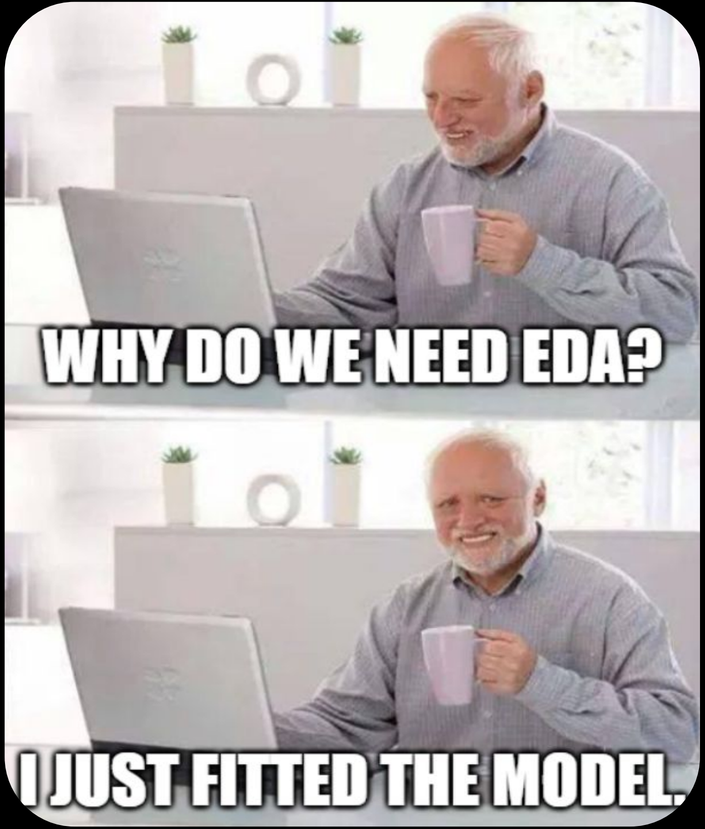
Let’s start by a quick revision for data types...

- Tabular Data
- Time Series Data
- Waves (e.g. Audio)
- Text
- Images
- Videos

Understand Your Toolkit: EDA

- **Exploratory Data Analysis (EDA)** is to explore your data by looking into raw samples, statistics or plots to gain useful insights about your task.

- Why do we need it?🤨



Understand Your Toolkit: EDA

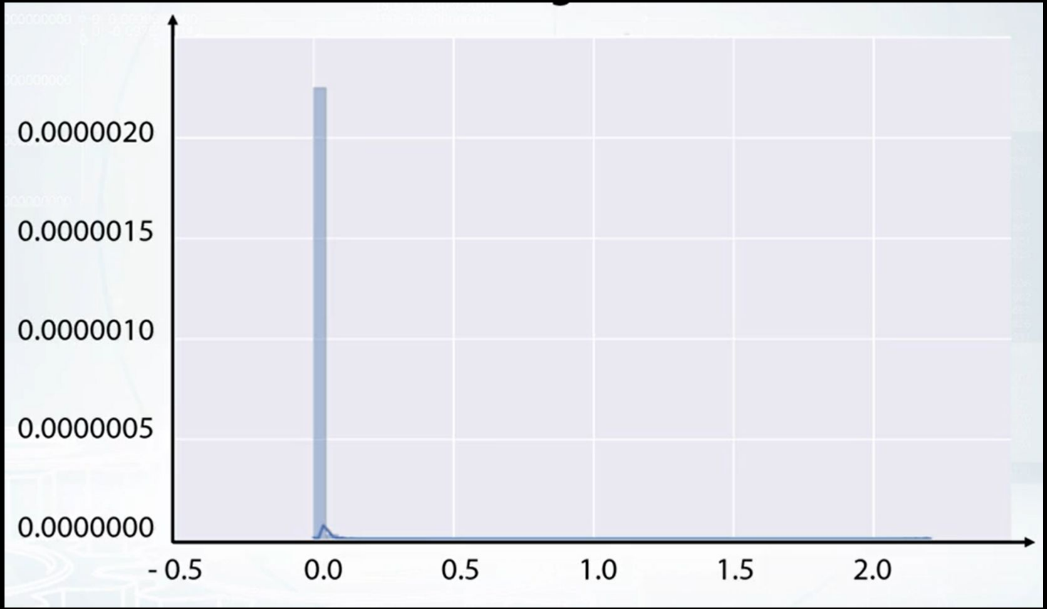
- **Exploratory Data Analysis (EDA)** is to explore your data by looking into raw samples, statistics or plots to gain useful insights about your task.
- Why do we need it?🤨
  - Get comfortable with the data.
  - Determine how to approach the problem.
  - Determine what is the best cv split.
  - Determine the most important features (How can you do that?👀).
  - Determine any strange behaviour in features' distributions or features' correlation with each other.
  - Discover serious problems with the data (e.g. Leakage).
  - Find magic features👀

Understand Your Toolkit: EDA

- Let’s see some examples👀…

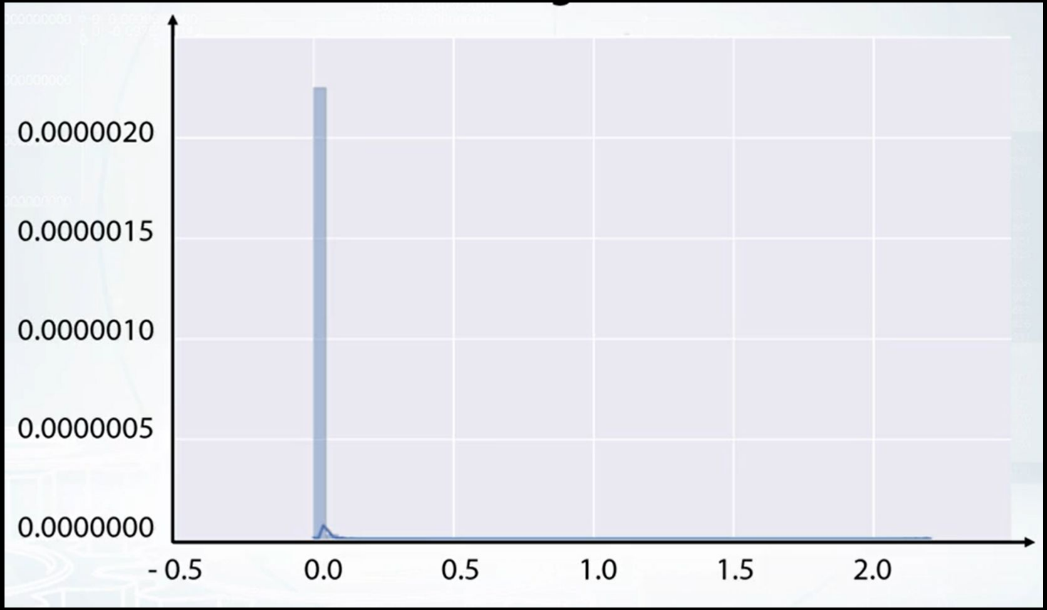
Understand Your Toolkit: EDA

- Plotted the histogram of a feature and saw this:
- Any problems?



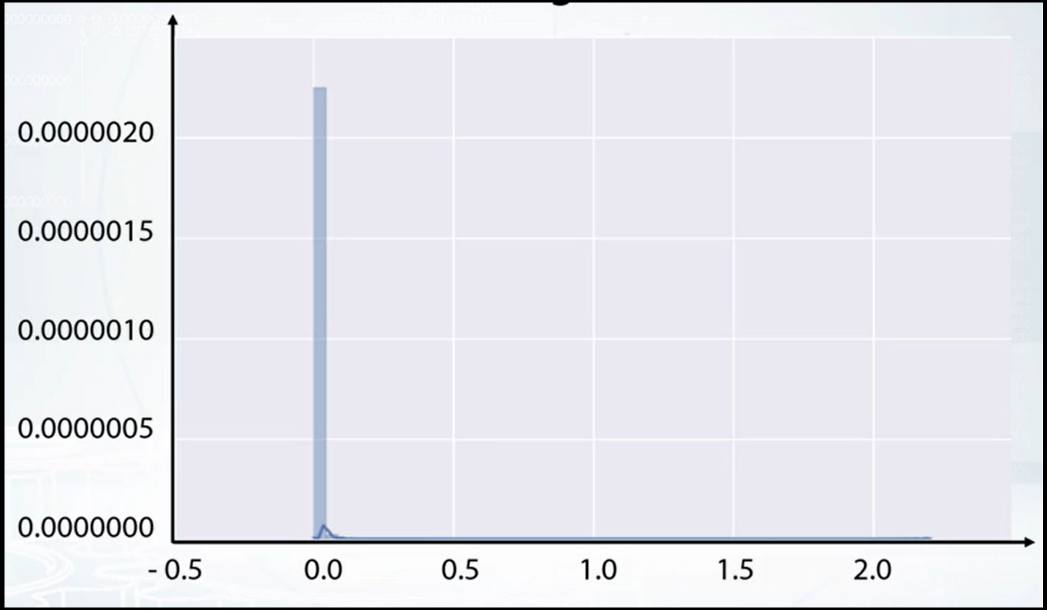
Understand Your Toolkit: EDA

- Plotted the histogram of a feature and saw this:
- Any problems?
- Long tail (sign of outliers).



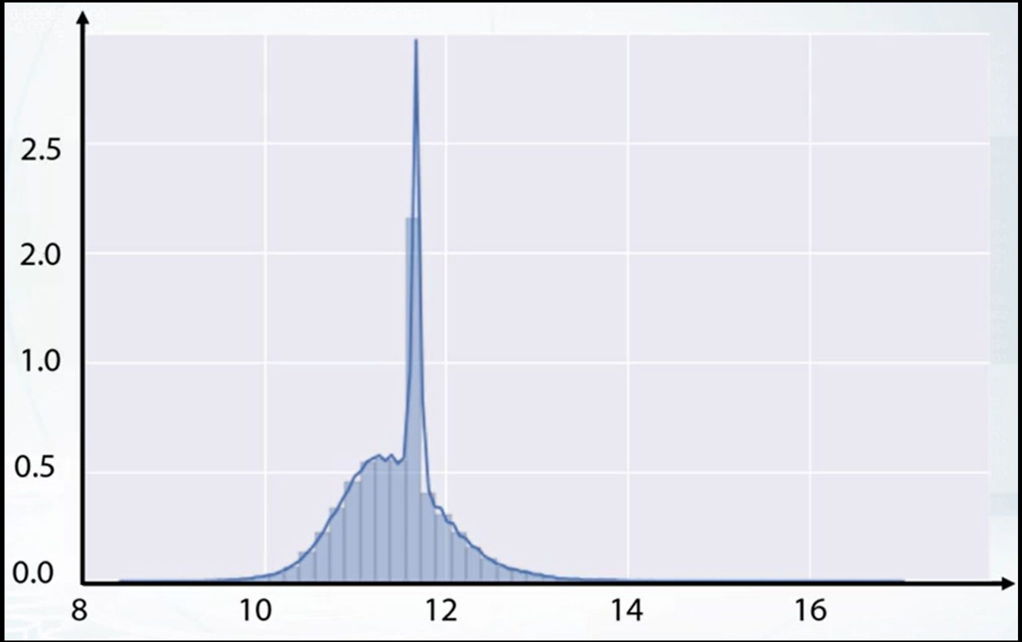
Understand Your Toolkit: EDA

- Plotted the histogram of a feature and saw this:
- Any problems?
- Long tail (sign of outliers).
- Let’s try take the logarithm👀



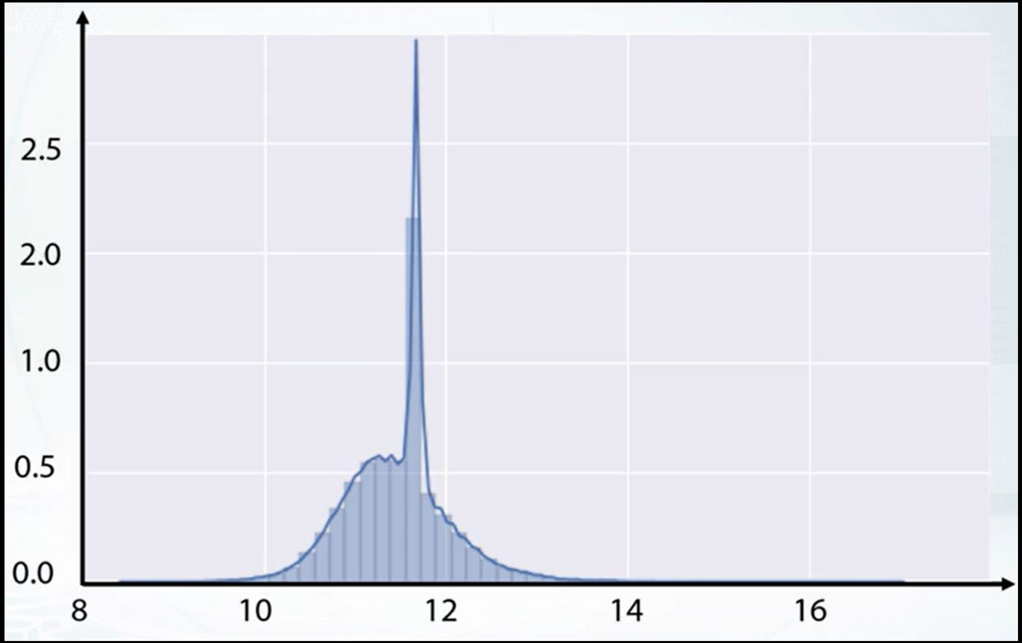
Understand Your Toolkit: EDA

- Plotted the histogram of a feature and saw this:
- Any problems?



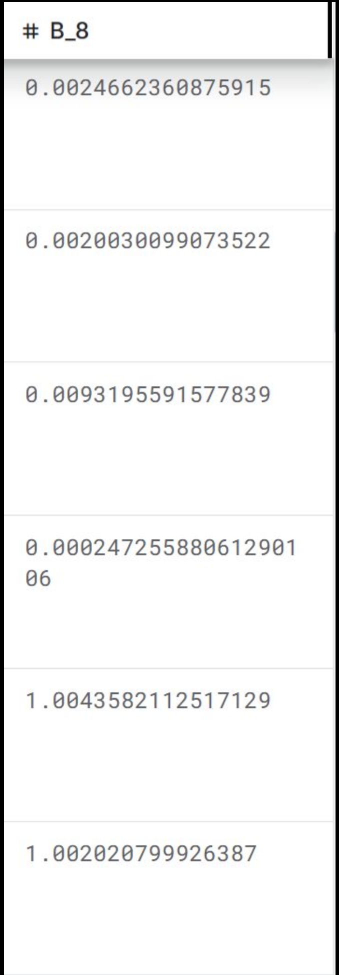
Understand Your Toolkit: EDA

- Plotted the histogram of a feature and saw this:
- Any problems?
- Missing values filled with the mean!



Understand Your Toolkit: EDA

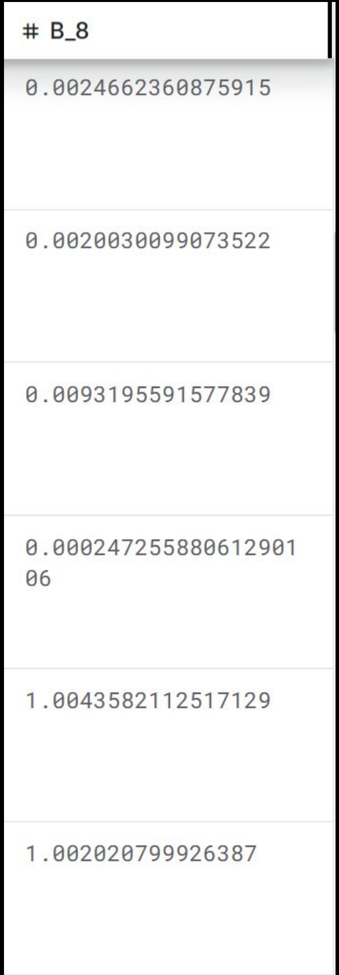
- Some values of a feature.
- Nothing strange right? Just an ordinary continue feature.



B_8
0.0024662360875915
0.0020030099073522
0.0093195591577839
0.00024725588061290106
1.0043582112517129
1.002020799926387

Understand Your Toolkit: EDA

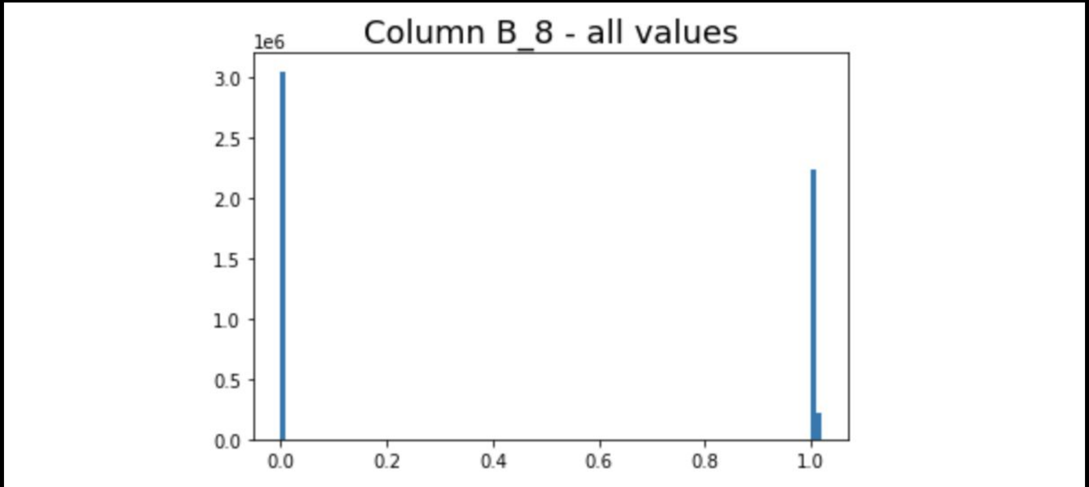
- Some values of a feature.
- Nothing strange right? Just an ordinary continue feature.
- Let’s plot its histogram.

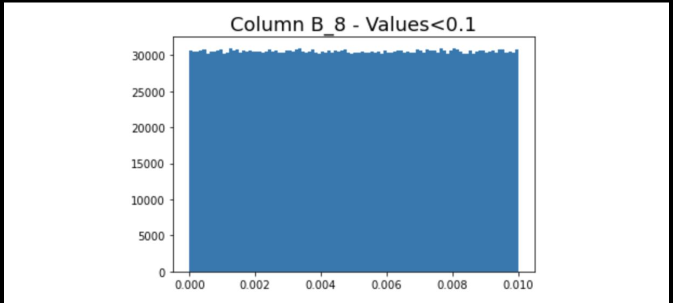


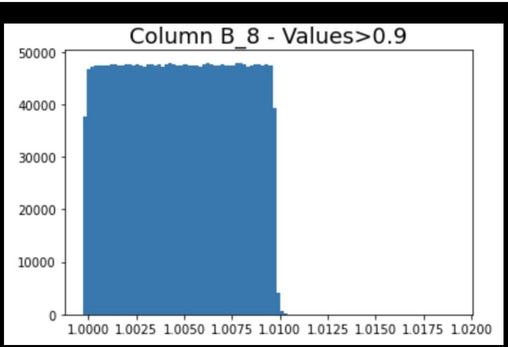
B_8
0.0024662360875915
0.0020030099073522
0.0093195591577839
0.00024725588061290106
1.0043582112517129
1.002020799926387

Understand Your Toolkit: EDA

- Does this looks like a continuous feature?

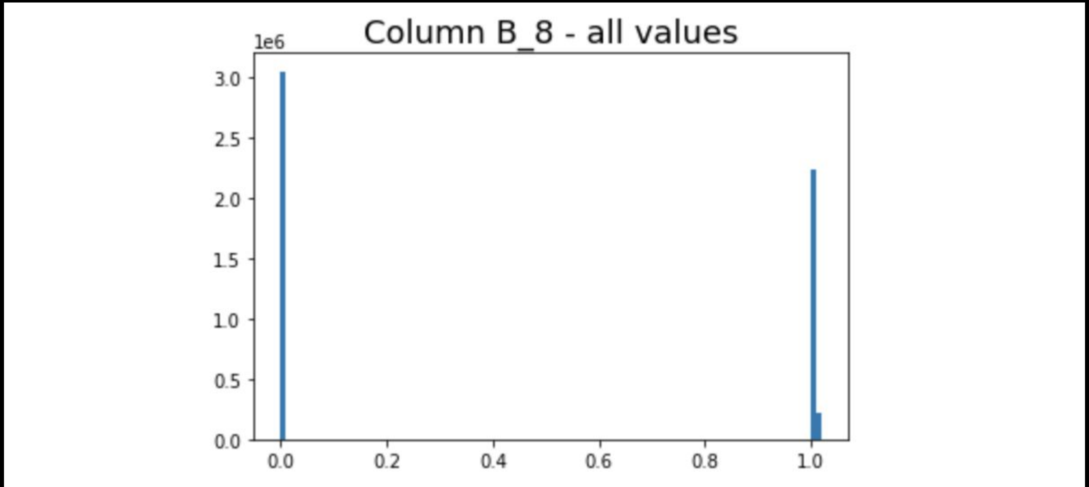


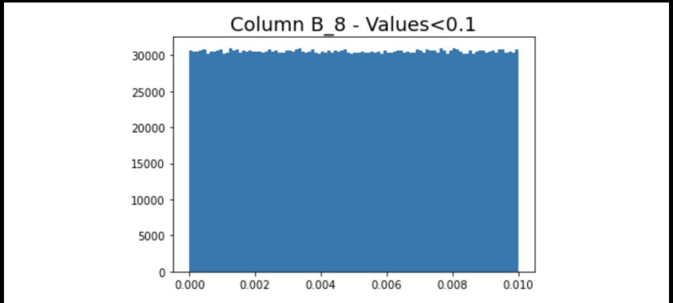


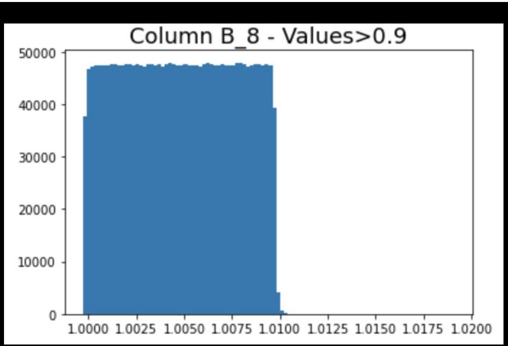


Understand Your Toolkit: EDA

- Does this looks like a continuous feature?
- Finding: Noise was injected into the feature!!







Understand Your Toolkit: EDA

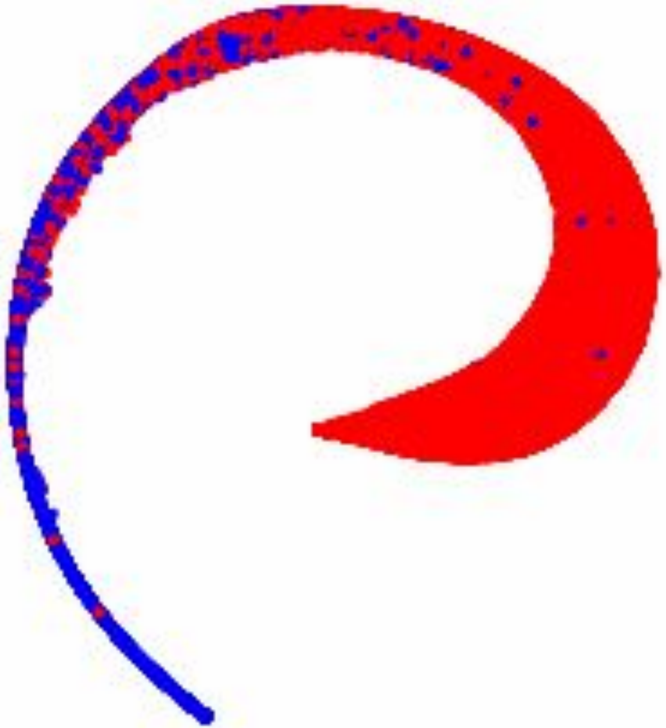
- **Good EDA could be the key for winning a competition.**
- **Examples:**
  - AMP®-Parkinson's Disease Progression Prediction | Kaggle
  - Google Research - Identify Contrails to Reduce Global Warming | Kaggle

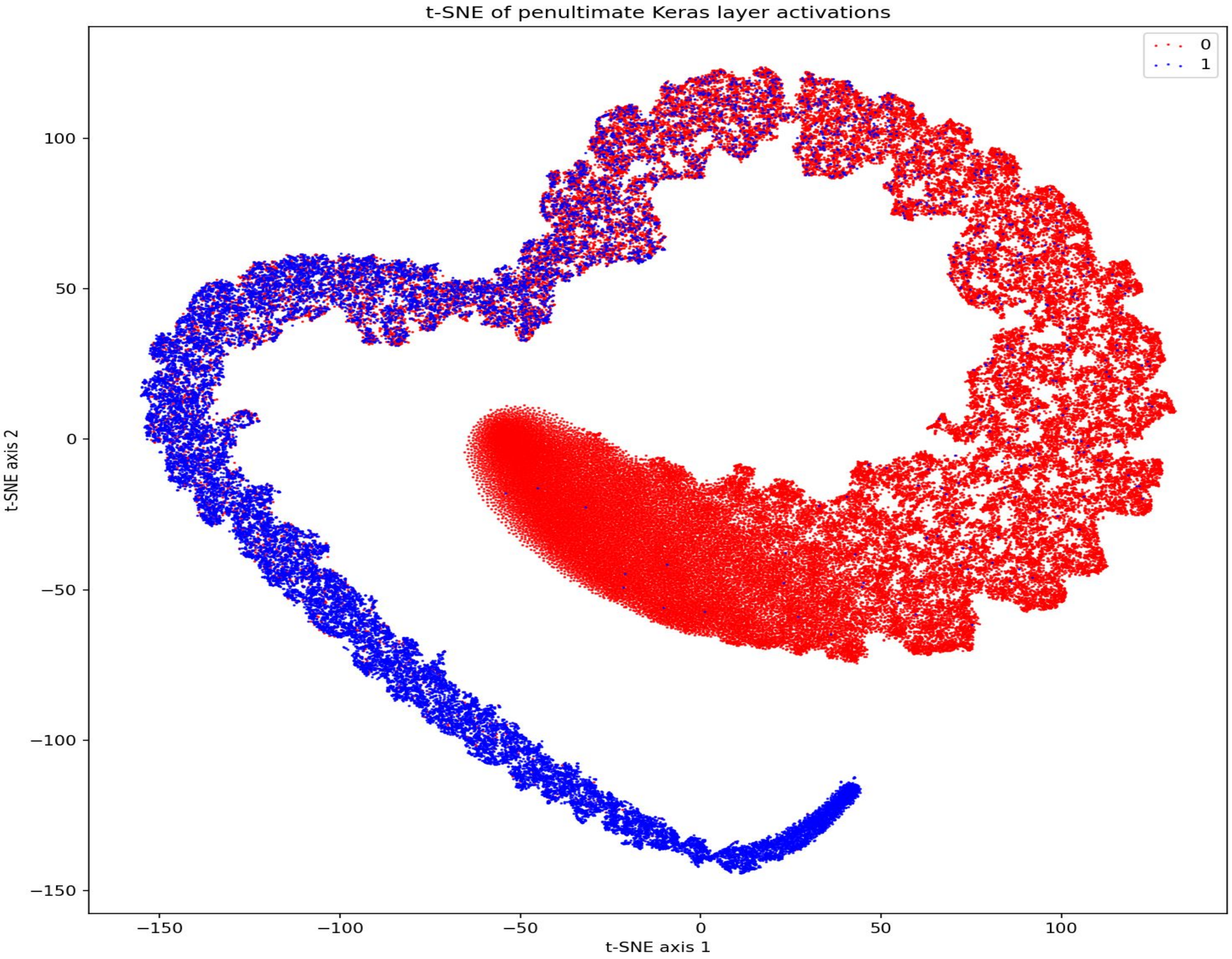
Understand Your Toolkit: EDA

- Tools & Techniques:
  - Plots: Ordinary EDA.
  - T-SNE/UMAP: Visualize high dimensional data. Link
  - Trees Models: Detect important features. Link
  - LOFO/SHAP/Permutation Importance: Better features selection tools. Link
  - Adversarial Validation: Detect Distribution Shift. Link
  - ...

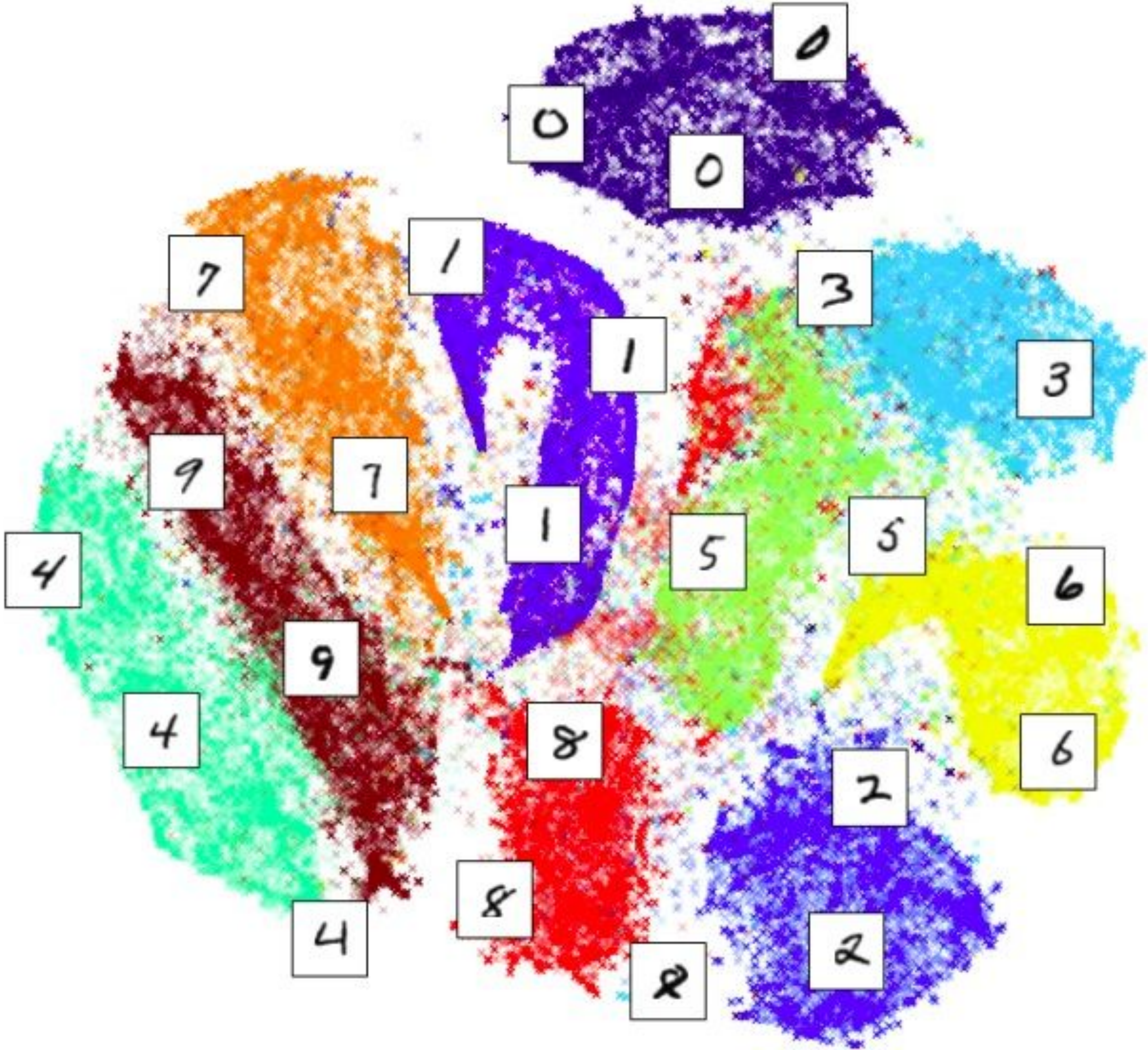
t-SNE short-movie showing how the NN behave during training.

Reference: American Express - Default Prediction | Kaggle





t-SNE of penultimate Keras layer activations
0
1
t-SNE axis 2
100
50
0
−50
−100
−150
−150
−100
−50
0
50
100
t-SNE axis 1


0
0
0
7
1
1
3
3
9
7
1
5
5
4
6
9
4
8
2
6
4
8
8
2

Understand Your Toolkit: Data Preprocessing

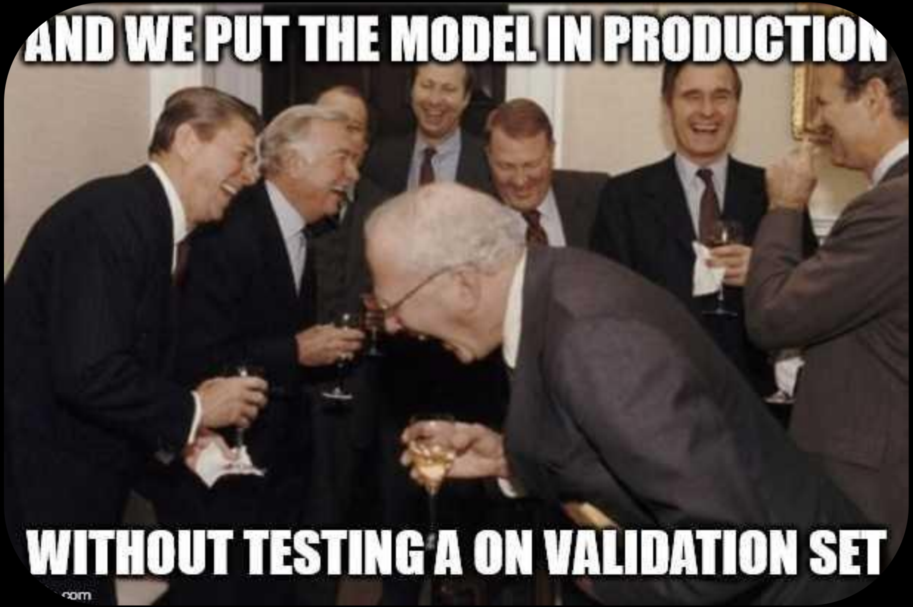
- Tabular/Time-series Data:
    - Encoding (Label, One-Hot,Frequency,...)
    - Feature Engineering (Aggregations, Lags, Differences,...). Link
    - Features Selection (Trees, LOFO, SHAP,...)
    - Normalization (maybe not?)
    - ...

Understand Your Toolkit: Data Preprocessing

- Images/text data:
  - Images Normalization.
  - Augmentations
  - Text Cleaning
  - Text Tokenization
  - ...

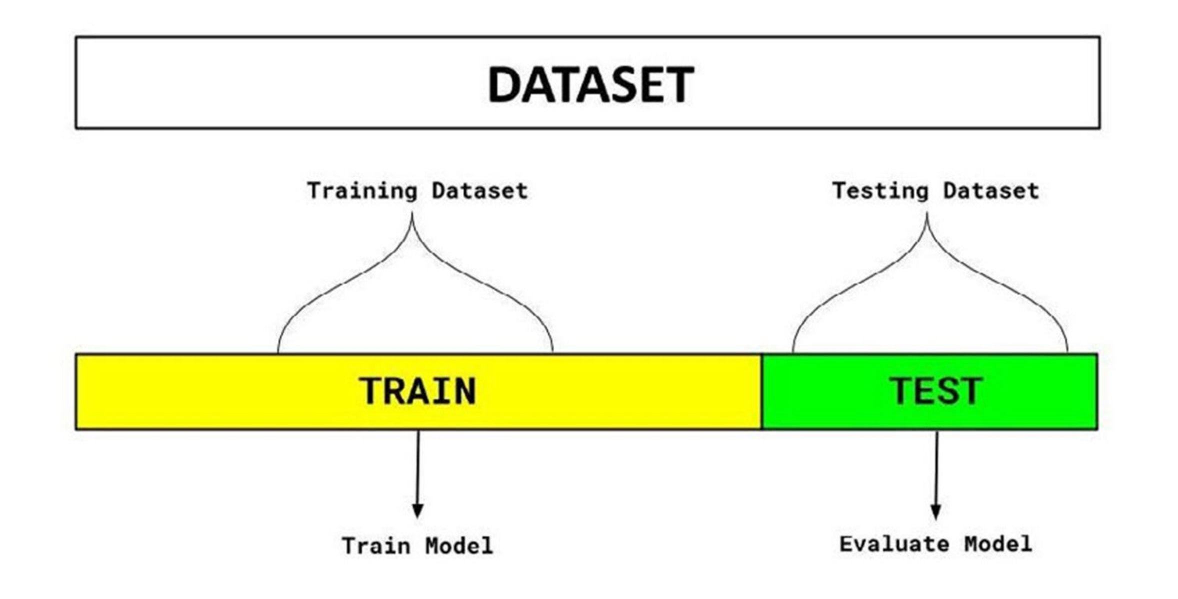
Understand Your Toolkit: Validation

- The most important step in the beginning of any project/competition is to choose the correct train/validation split.
- Choosing the incorrect split would lead to wrong evaluation making the all the next steps useless.


AND WE PUT THE MODEL IN PRODUCTION
WITHOUT TESTING A ON VALIDATION SET
.com

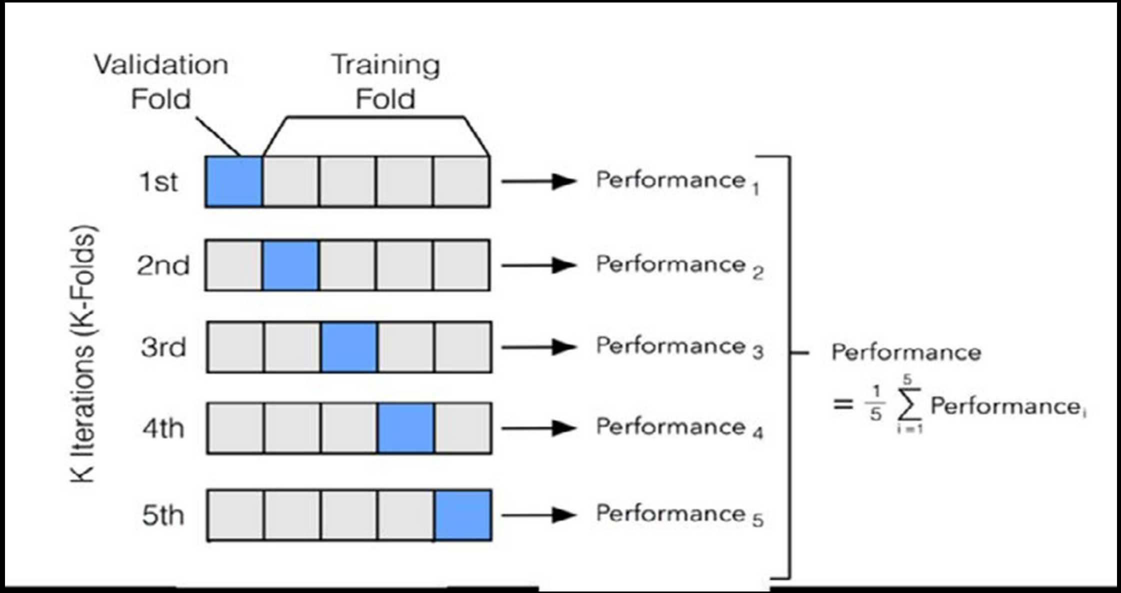
Understand Your Toolkit: Validation

- **Types of Data Split:**
  - **Hold-out fold**: Splits data into a single training set and a single validation set.



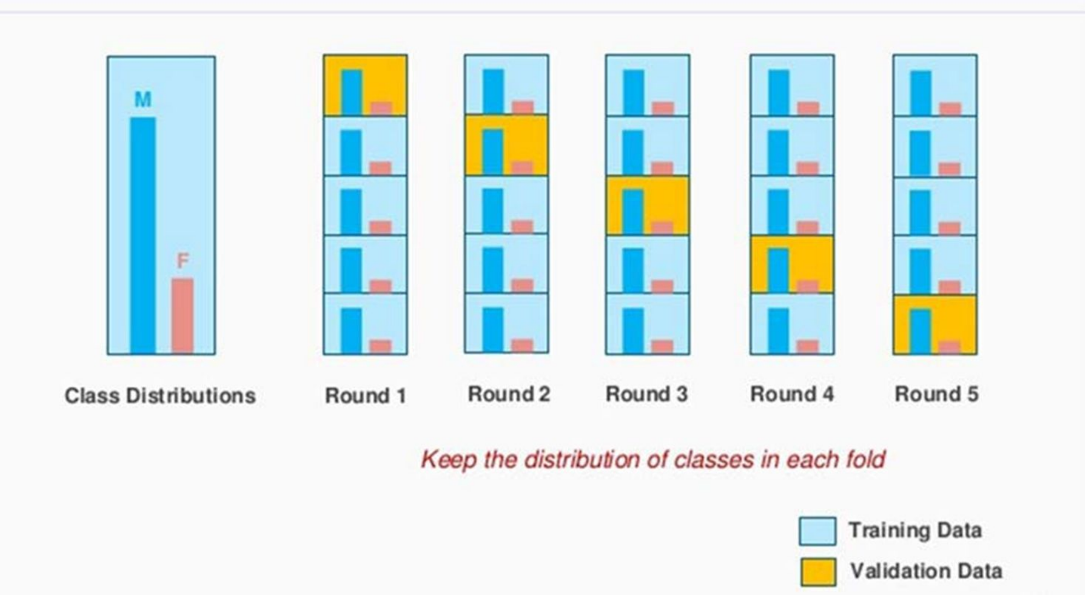
Understand Your Toolkit: Validation

- **Types of Data Split:**
  - **KFold**: Divides data into k subsets and uses each subset as a validation set while training on the remaining k-1 subsets



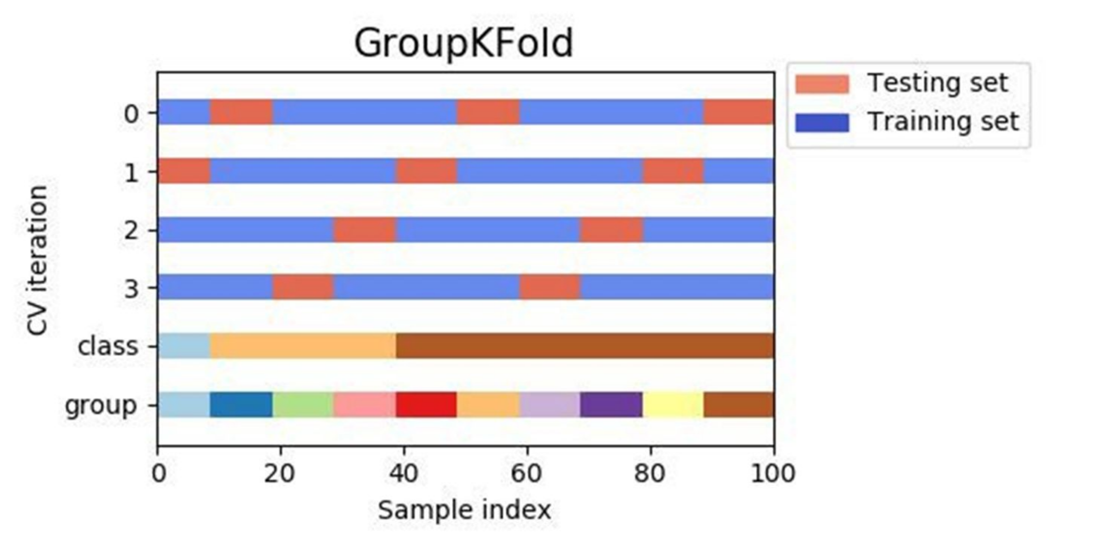
Understand Your Toolkit: Validation

- **Types of Data Split:**
  - **StratifiedKFold**: Similar to KFold, but ensures that each fold maintains the same proportion of class labels as the entire dataset.



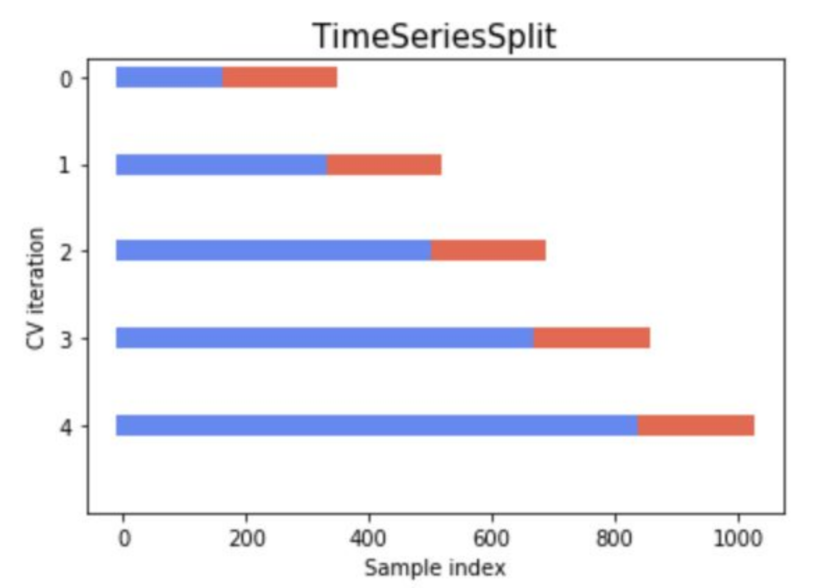
Understand Your Toolkit: Validation

- **Types of Data Split:**
  - ○ **GroupKFold**: Splits data into k folds while preserving the grouping of samples within each fold.



Understand Your Toolkit: Validation

- **Types of Data Split:**
  - **TimeSeriesSplit:** Divides data into training and validation sets in a manner that respects the temporal order, suitable for time series data.

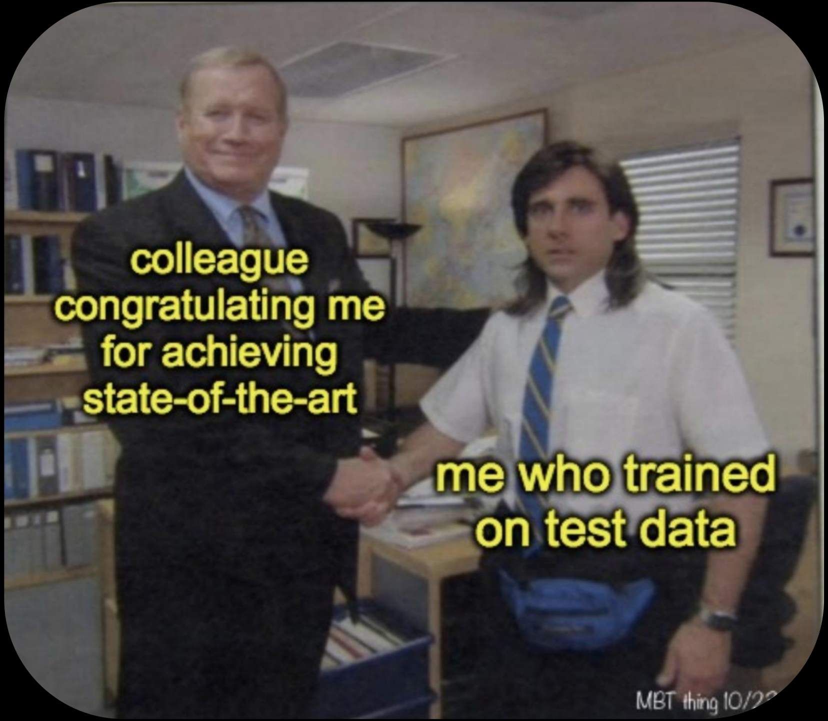


Understand Your Toolkit: Validation

- **Types of Data Split:**
  - Their Combinations...Link

Understand Your Toolkit: Validation

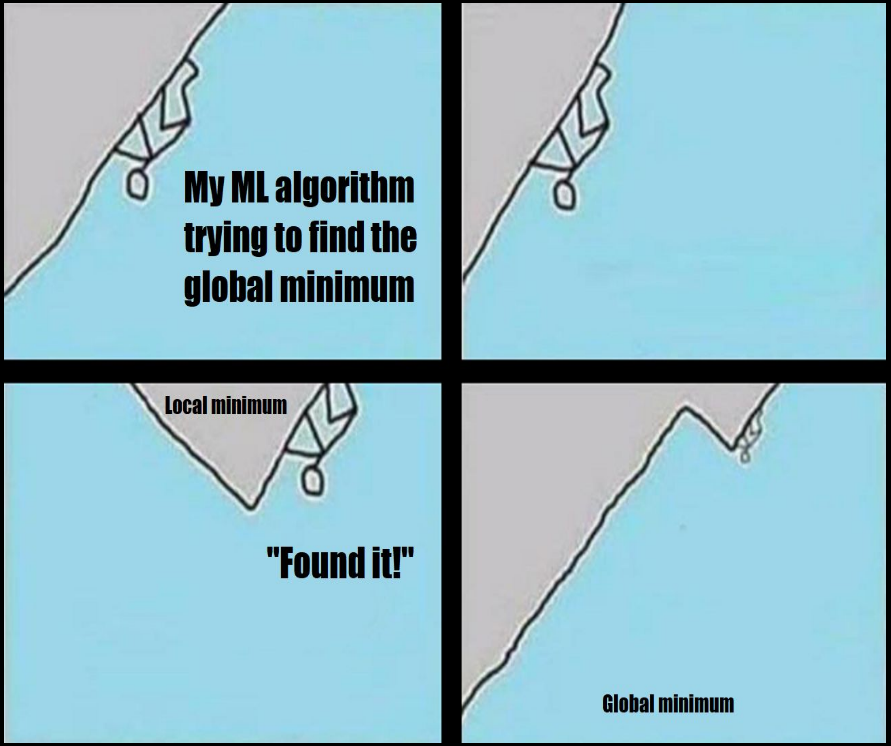
- How to know if your score is good or not?

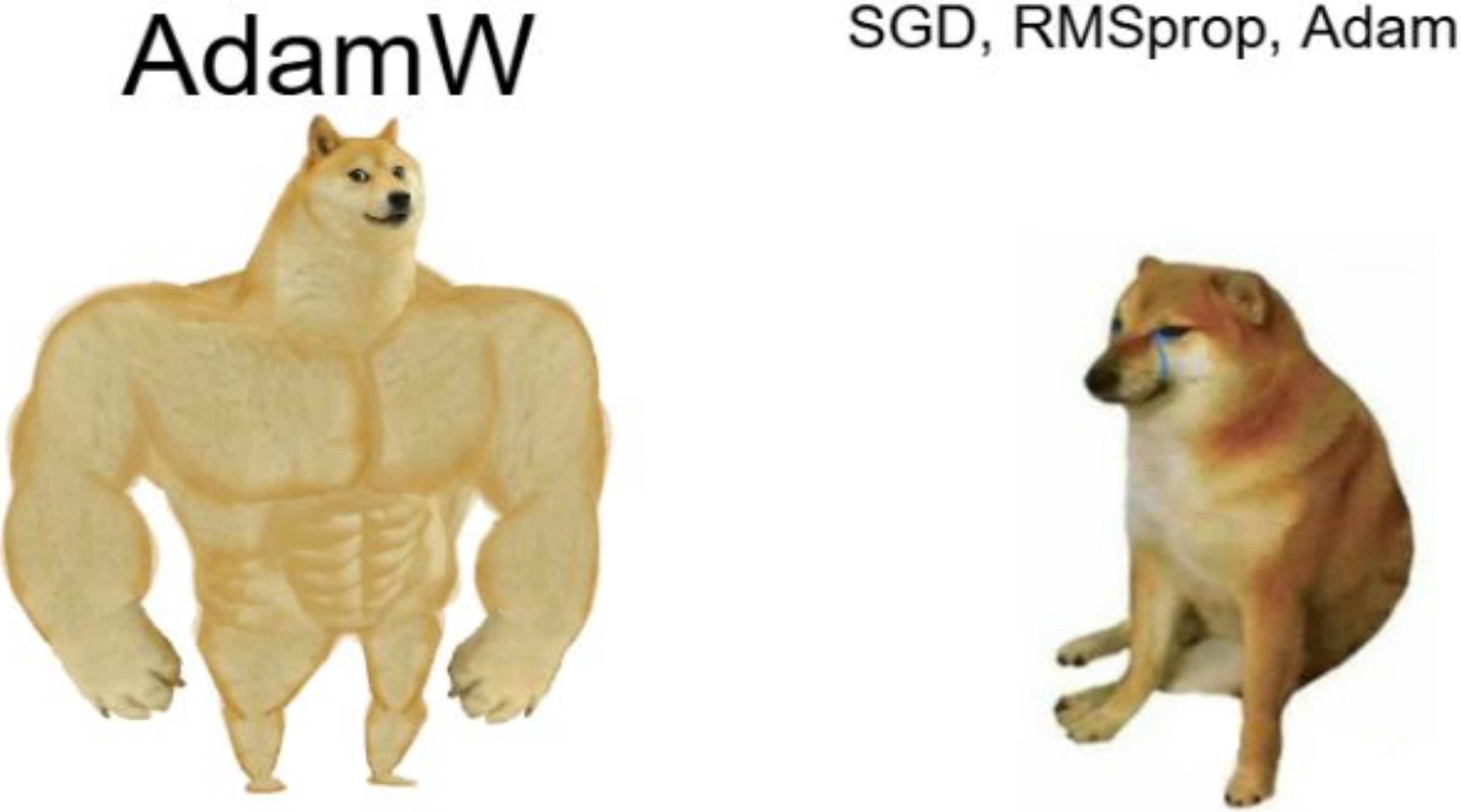

colleague congratulating me for achieving state-of-the-art
me who trained on test data
MBT thing 10/2

Make a Baseline then compare your subsequent scores against it.

Understand Your Toolkit: Optimizers

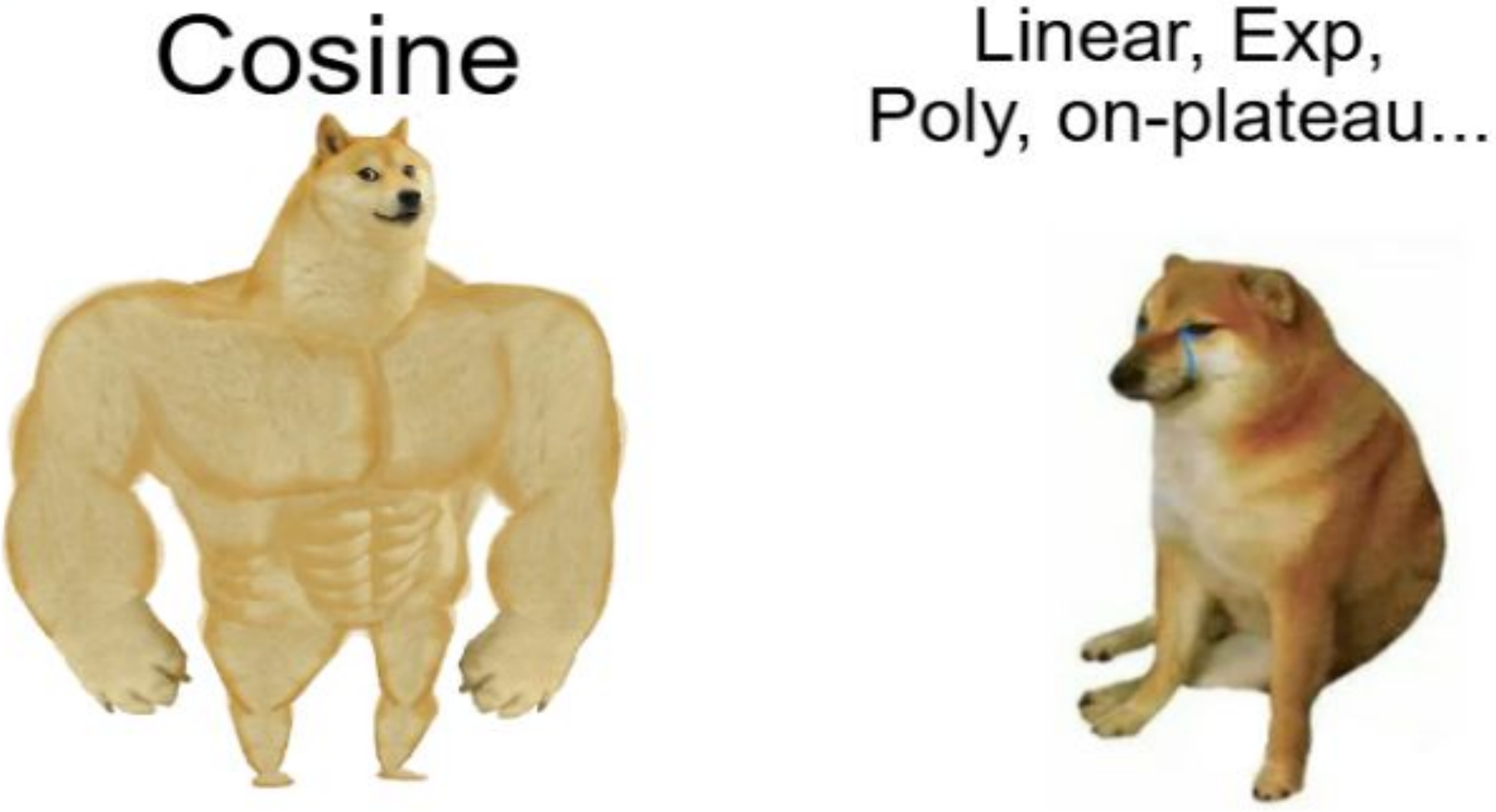
- Optimizers: Algorithms used to minimize your loss and update your model parameters.
- Types: SGD, SGD with Momentum, RMSprop, Adam, AdamW...etc
- The best one is ...?🤨




AdamW
SGD, RMSprop, Adam

Understand Your Toolkit: Schedulers

- Schedulers: Decay (decrease) the learning rate over time.
- Why? Increases the stability of the model a lot.
- Types: Linear (step decay), Exponential, Cosine, Polynomial, Reduce-on-plateau...etc
- The best one is ...?🤨


Cosine
Linear, Exp,
Poly, on-plateau...

Understand Your Toolkit: Training Stability

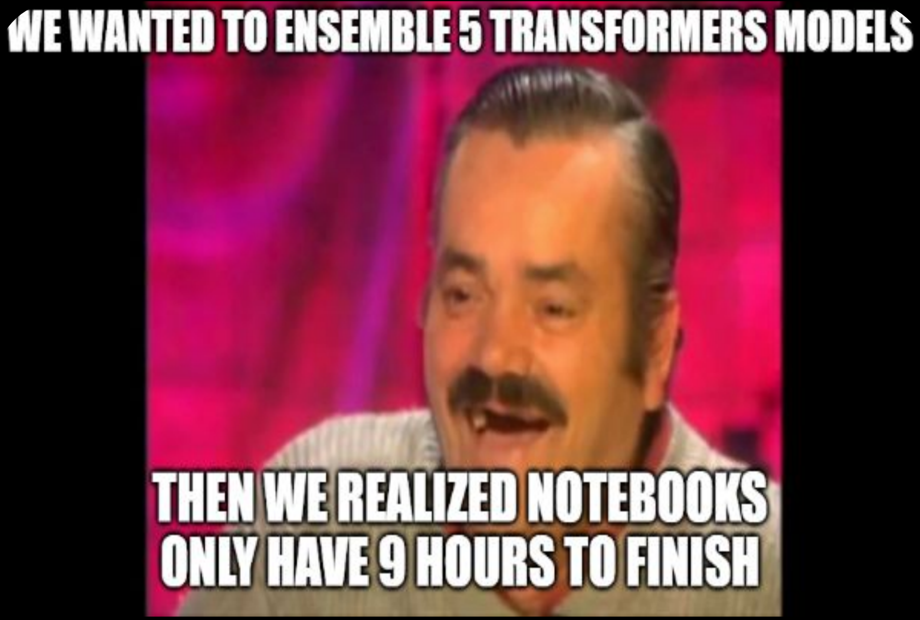
- To make the training stable we have these components work together:
    - Optimizer
    - Scheduler
    - Batch size
    - Initial Learning Rate
    - Warm-up steps
    - Number of Epochs

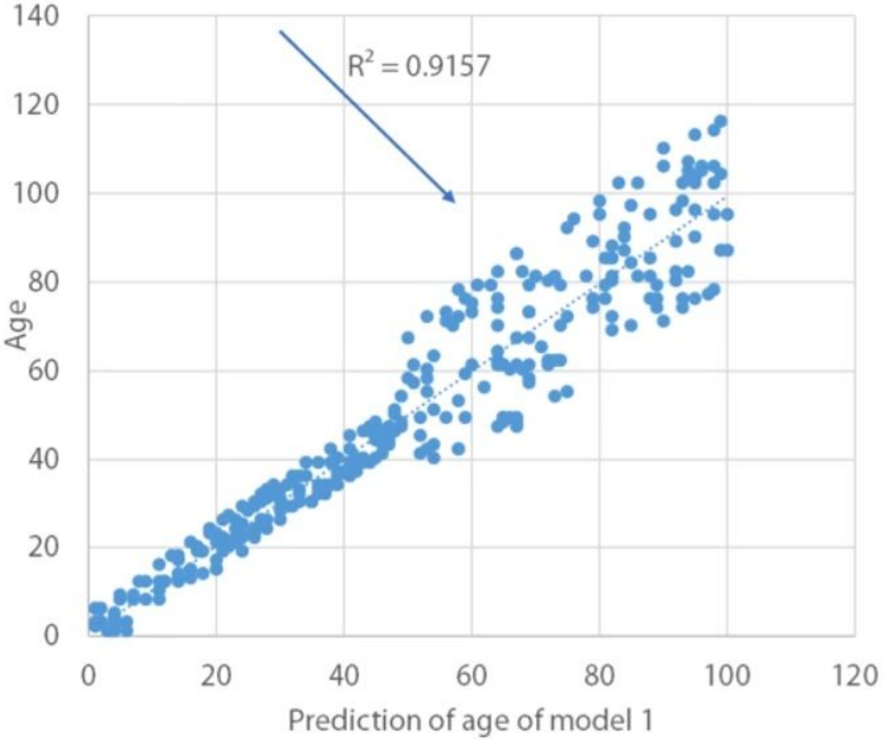
Understand Your Toolkit: Efficiency

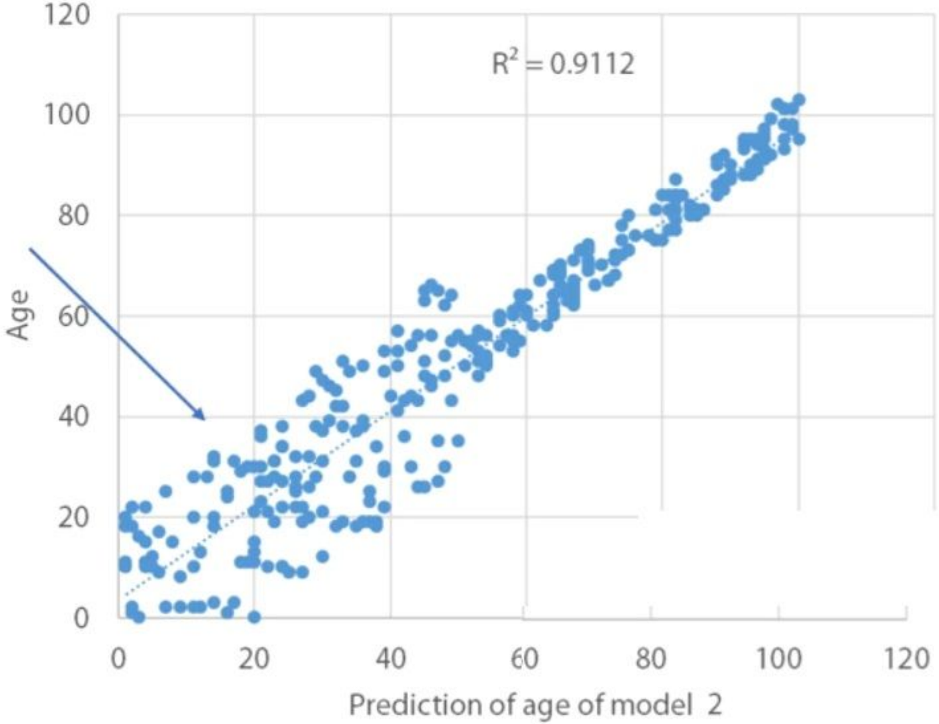
- Data Processing:
  - Cudf: Pandas on GPU
  - Polars: Pandas with multi-thread
- ML Algorithms:
  - Cuml: Sklearn on GPU
- DL Algorithms:
  - ONNX: Speed up for inference in CPU
  - Mixed Precision / Quantization: Lower size
  - Knowledge Distillation: Student model (small) try to learn from (replicate) a teacher model (big).
  - ...

Understand Your Toolkit: Ensembling

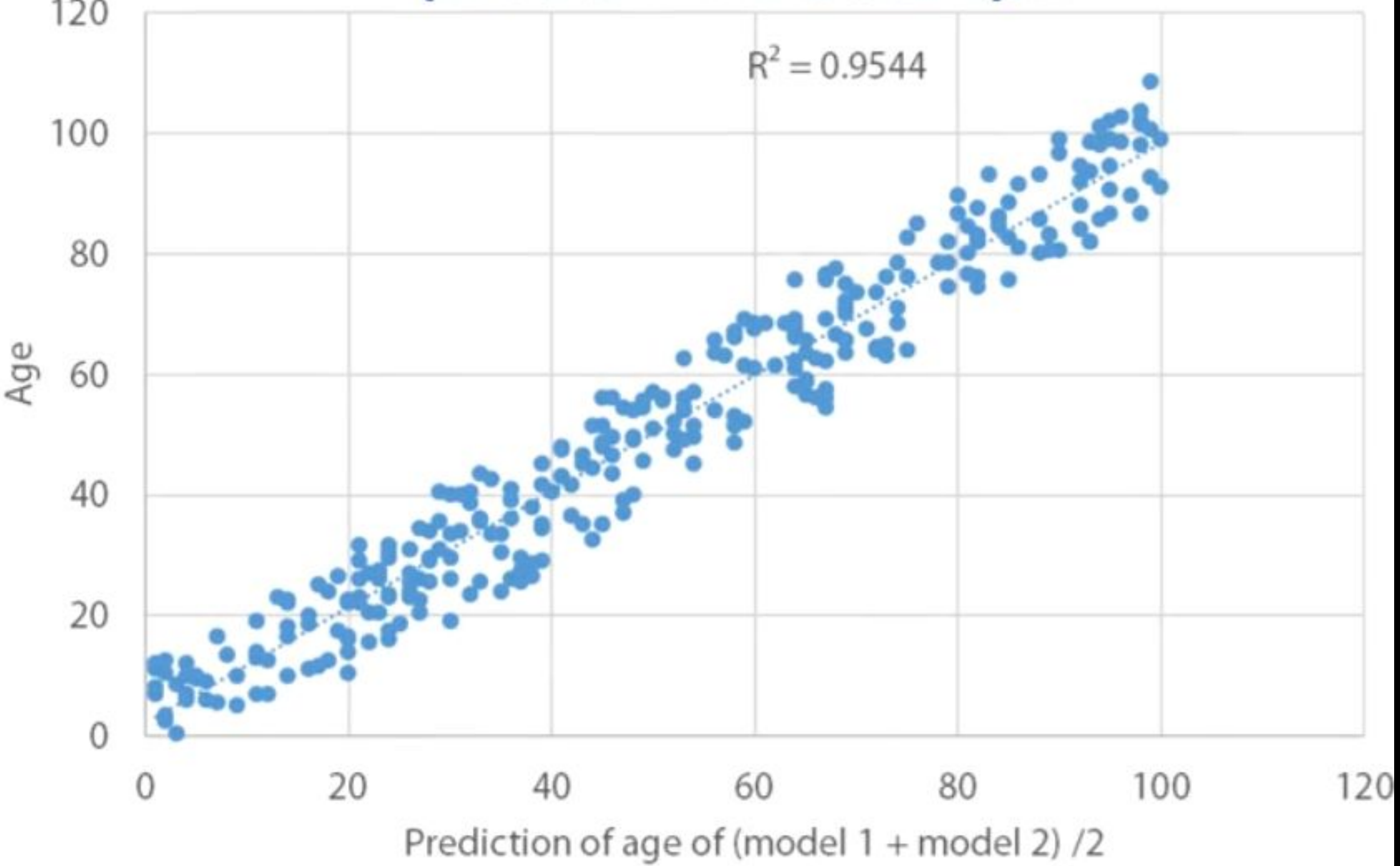
- Why using only one model? Let’s use several ones then use a weighted average!


WE WANTED TO ENSEMBLE 5 TRANSFORMERS MODELS
THEN WE REALIZED NOTEBOOKS ONLY HAVE 9 HOURS TO FINISH


140
120
100
80
60
40
20
0
Age
$R^2 = 0.9157$
0
20
40
60
80
100
120
Prediction of age of model 1


120
100
80
60
40
20
0
Age
$R^2 = 0.9112$
0
20
40
60
80
100
120
Prediction of age of model 2

(model 1 + model 2)/2



- Types of Ensembling:
    - Averaging (or Blending)
    - Bagging
    - Stacking

Choosing The Best Models

It’s all about the assumptions.

Revisit Data Types...

Do we have a more high level approach/view to see these types of data?

- Tabular Data
- Time Series Data
- Waves (e.g. Audio)

- Text
- Images
- Videos

Revisit Data Types...

- Classify the data types based on their properties (assumptions):
    - **Tabular (non-sequential)**: Order of features doesn't matter.
    - **Sequence**: Order does matter.
    - **Neighborhood**: Each feature has a relation with its neighbors.
    - **Graph**: Generalized Neighborhood.
    - **Sequential Decision-making**: Sequence + feedback influencing future decisions.

Data Assumptions

- Classify the data types based on their properties (assumptions):
    - **Tabular (non-sequential)**: Tabular Data.
    - **Sequence**: Time-series data, text data, waves data.
    - **Neighborhood**: Images.
    - **Graph**: Graph-based data (e.g. map, molecular structures,...)
    - **Sequential Decision-making**: Decision-making environments (e.g. playing games, robotics, ...)

WAIT A MINUTE,
WHERE ARE THE VIDEOS?

Data Assumptions

- Classify the data types based on their properties (assumptions):
    - **Tabular (non-sequential)**: Tabular Data.
    - **Sequence**: Time-series data, text data, waves data.
    - **Neighborhood**: Images.
    - **Graph**: Graph-based data (e.g. maps, molecular structures,...)
    - **Sequential Decision-making**: Decision-making environments (e.g. playing games, robotics, ...)
    - **Combination between them**: Videos (Sequence+Neighborhood)

Models Assumptions

- Classify the data types based on their properties (assumptions):
    - **Tabular (non-sequential)**: Machine Learning models.
    - **Sequence**: Sequence models (LSTM/Transformer/1D CNN)
    - **Neighborhood**: (1D CNN/2D CNN/3D CNN)
    - **Graph**: Graph NN
    - **Sequential Decision-making**: Reinforcement Learning
    - **Combination between them**: e.g. (for videos: CNN + Sequence)

Models & Data Assumptions

- So, to figure out the right model, you should know the properties/assumptions you have your dataset (by EDA!) then choos the appropriate models.

MY FRIEND FELL DOWN IN PRIVATE LB...
HE USED DEEP LEARNING WITH TABULAR DATA

Practical Tips and Strategies

It’s all about the assumptions.

ML Models:

- ML Models are the SOTA in tabular data.
- Types of models:
    - Linear Models.
    - Kernel Models.
    - Trees Models.

ML Models:

- **Linear Models**: Linear Regression / Logistic Regression
    - Best when there is a linear relationship between the features and the target variable.
    - Most stable model.
    - Used widely in time-series forecasting due to high relationship between target and lags.
    - Best Versions: Ridge, ElasticNet, Lasso. (in sklearn).
    - Forecasting variants: Arima, Facebook Prophet,...

ML Models:

- **Kernel Models**: SVM
    - Best when there is a very big number of features.
    - Quite slow for samples > 5K (How to solve this?👀).
    - Used sometimes as a head for embeddings extracted from the models because it can handle high dimensionality. Link
    - Best Versions: SVM (in sklearn), and ...?👀

ML Models:

- **Kernel Models**: SVM
    - Best when there is a very big number of features.
    - Quite slow for samples > 5K (How to solve this?👀).
    - Used sometimes as a head for embeddings extracted from the models because it can handle high dimensionality. Link
    - Best Versions: SVM (in sklearn), SVM (in cuml - very fast).

ML Models:

- **Trees Models:**
  - Generally the best ML algorithms.
  - Types:
    - Decision Trees
    - Random Forest
    - Gradient Boosting (GBDT):
      - XGBoost: Best Balance
      - LightGBM: Fast in CPU
      - Catboost: Best baseline

ML Models:

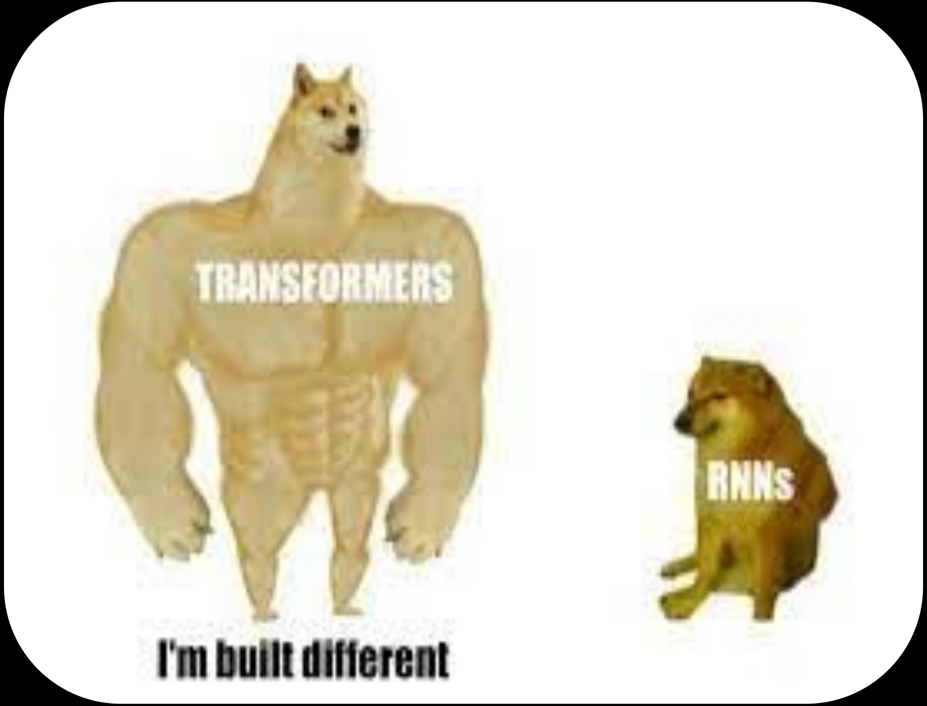
- **Examples:**
  - American Express - Default Prediction | Kaggle
  - ISIC 2024 - Skin Cancer Detection with 3D-TBP | Kaggle
  - Enefit - Predict Energy Behavior of Prosumers | Kaggle

DL Models:

- DL Models are the SOTA in everything except tabular (time series is in-between but mostly ML is better).

NLP Tasks

- **Best Models:** Transformers
  - Needs finetuning: (👑Deberta👑, Roberta,...)
  - Zero-shot: (Recent LLMs)
- **Best Library:** “transformers” by Huggingface🤗.



NLP Tasks

- NLP models are quite powerful already. Therefore, the improvements in LBs would be small.
- How to improve your results?
  - Clean your data properly.
  - Tune your parameters. They have high effects here.
  - Find models that are pre-trained on data similar to your task (you can use Huggingface).
  - Ensembling (averaging, stacking)
  - Manipulate the architecture and the embeddings. Link Link
  - Data-based ideas.
  - Check previous solutions.

NLP Tasks

- Notes:
  - If the task is regression, disable dropout to get more stable training. Link
  - Transformers needs very low learning rates (e.g. 1e-5, 2e-5)
  - Schedulers and warmup are widely used in NLP.
  - NLP models are usually finetuned using 1-3 epochs only before overfitting. (why? 👀)
  - LLMs are trained mostly on <= 1 epoch.
  - Scheduler should work on the steps, not the epochs.
  - NLP models are huge (e.g. xsmall deberta has ~70M params)
  - Recenely, a very interesting pattern appeared where Deberta could get beaten by LLMs in classification IF the data came from LLMs.

NLP Tasks

- **Examples:**
  - Feedback Prize - English Language Learning | Kaggle
  - Learning Agency Lab - Automated Essay Scoring 2.0 | Kaggle
  - LMSYS - Chatbot Arena Human Preference Predictions | Kaggle
  - AI Mathematical Olympiad - Progress Prize 1 | Kaggle
  - Kaggle - LLM Science Exam | Kaggle
- Some good baselines:
  - AES-2 | Multi-class Classification [Train] (kaggle.com)
  - [Training] Gemma-2 9b 4-bit QLoRA fine-tuning (kaggle.com)

Computer Vision

- **Best Models:**
  - EfficientNet (CNN-based)
  - ResNet (CNN-based)
  - ViTs - Swin Transformers
  - ConvNeXt
- **Best Library**:
  - **Models:** 👑Timm👑, Torchvision.
  - **Augmentations:** 👑Albumentations👑, Torchvision.

What normal people see when they walk on street
What Computer Vision folks see

Computer Vision Tasks

- **Classification:** EfficientNet, ResNet, ViT,..
- **Segmentation (2D/2.5D/3D):** 👑Unet-based archs👑, Mask RCNN,...
- **Detection:** 👑DETR (DEtection TRansformer), YOLO👑, EfficientDet, Faster RCNN,...
- How to improve your results? Everything in NLP +..
    - 👑Augmentations👑
    - TTA (Test-time augmentations).
    - Data-based ideas.

Computer Vision Tasks

- **How to choose the appropriate augmentations?** 👑Error Analysis👑
  - Train a baseline.
  - Make predictions on your validation data.
  - Inspect the worst predicted images, these predictions should guide you to the problems the model facing.
  - Examples:
    - Failure with very small objects: Scale augmentation.
    - Failure with different colors / environment: Color augmentations.
    - Failure with rotated images: Rotation augmentations.
    - Failure with Blurry images: Noise augmentations.
    - ...etc

Computer Vision Tasks

- **Notes:**
  - Smaller models can be better than bigger models in many times.
  - Computer vision models are small. Training is fast compared to NLP.
  - You will need a bit bigger lr if you use CNN-based model (e.g. 1e-4,1e-3) and small one if you use Transformer-based model (e.g. 1e-5,2e-5).
  - You may need big number of epochs (e.g. 5-200). It depends on the task.
  - Using wrong augmentations can decrease performance significantly.
  - Scheduler based on epochs not steps.

Computer Vision Tasks

- **Examples:**
  - Google Research - Identify Contrails to Reduce Global Warming | Kaggle
  - RSNA 2022 Cervical Spine Fracture Detection | Kaggle
  - Vesuvius Challenge - Ink Detection | Kaggle
  - RSNA Screening Mammography Breast Cancer Detection | Kaggle
- **Resources:**
  - Kaggle Days Paris 2022_Philipp Singer & Yauhen Babakhin_Practical Tips for Deep Transfer Learning - YouTube

When you haven't even gone to sleep yet and you already can't wait to come home from work tomorrow

Waves

- **Best Models:**
  - 1D-CNN (e.g. WaveNet,..)
  - and... 👀👀

Waves

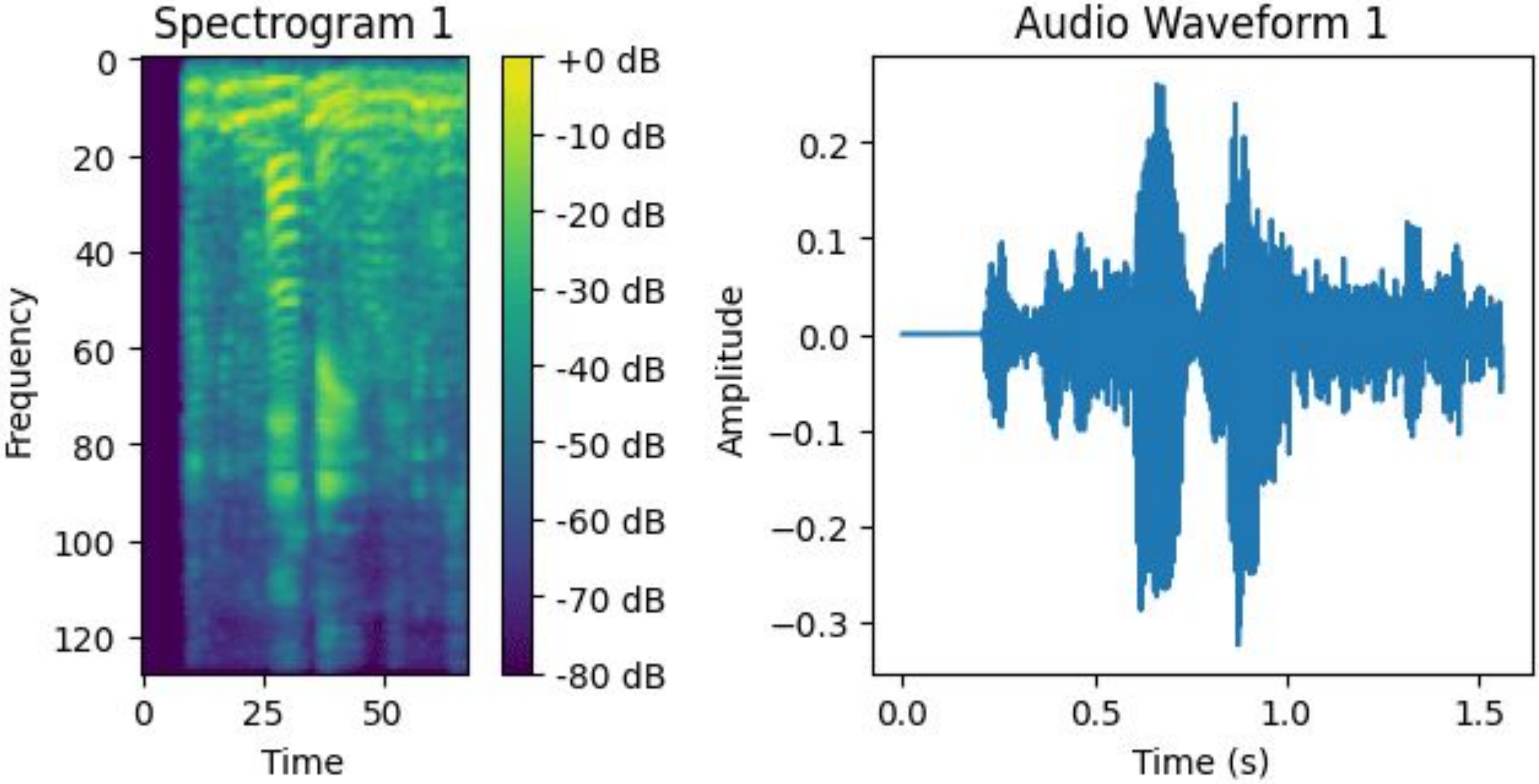
- **Best Models:**
    - 1D-CNN (e.g. WaveNet,..)
    - 2D-CNN (e.g. EfficientNet,..) (What?🤨)

Waves

- **Best Models:**
  - 1D-CNN (e.g. WaveNet,..)
  - 2D-CNN (e.g. EfficientNet,..) (What?🤨)

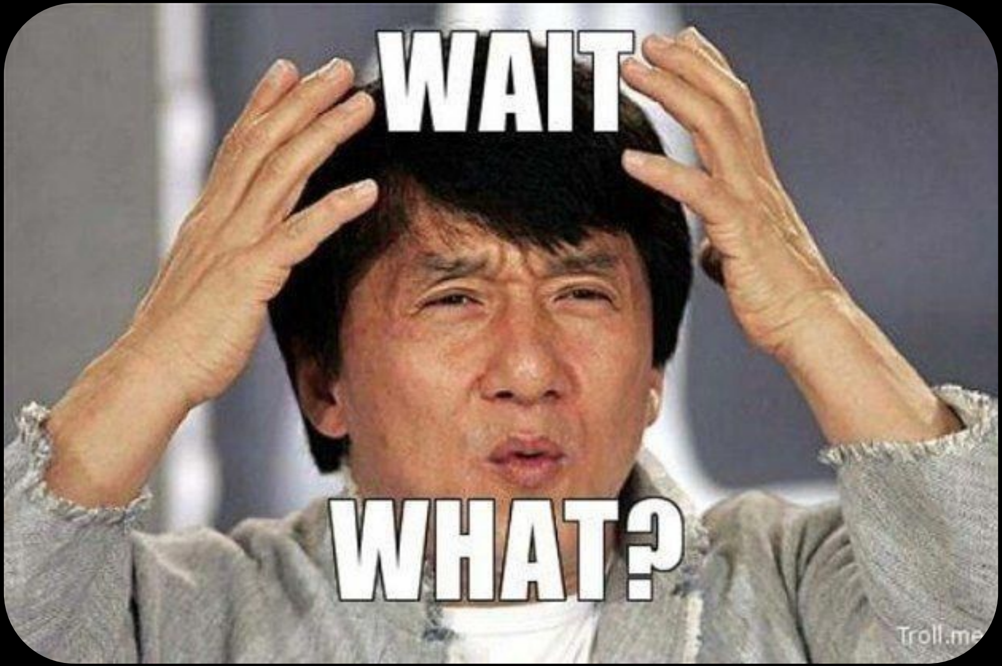
    Waves can be converted into an image representation called **Spectrograms.** Then we fit CNN on it.👀

Spectrogram Comparison for Class: puppy



Waves

- Fitting on spectrograms is the SOTA👀👀👀👀👀.

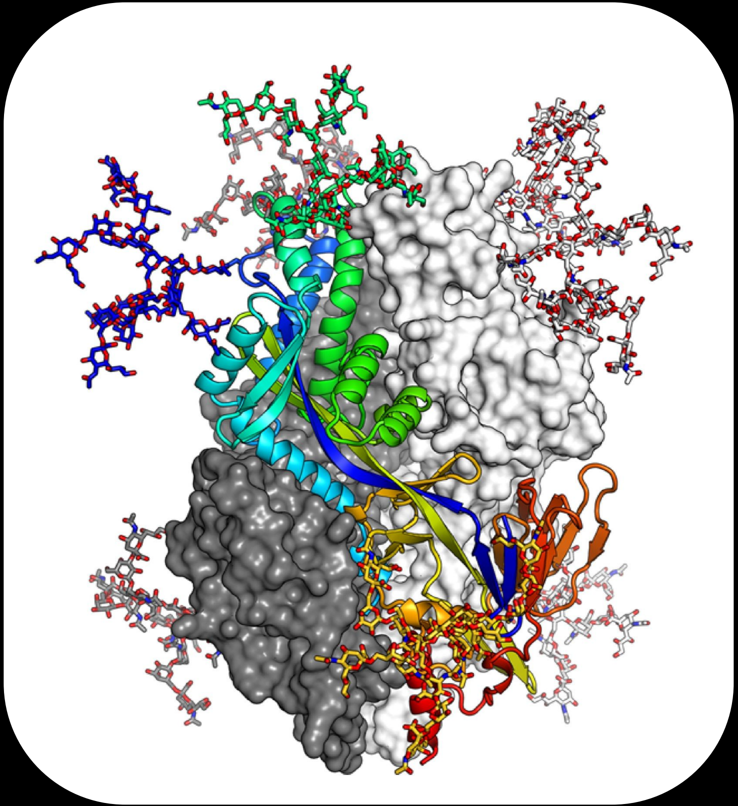


Waves

- **Examples:**
  - BirdClef 2023: Pytorch Lightning-Training w/ cMAP (kaggle.com)
  - HMS - Harmful Brain Activity Classification | Kaggle

Proteins & DNA Sequences

- Per-Char Sequences.
- Can be approached using NLP models.
- Sometimes, it can be converted to 3D images, then approached as 3D CNN.
- Can be approached using Graph NN as well.



Proteins & DNA Sequences

- Examples:
  - Novozymes Enzyme Stability Prediction | Kaggle
  - NeurIPS 2024 - Predict New Medicines with BELKA | Kaggle
  - Stanford Ribonanza RNA Folding | Kaggle

AI & Security

- CTF but for AI👀.
- Learn about Adversarial attacks, LLM jailbreaking and some random stuff👀.
- **Examples:**
  - AI Village Capture the Flag @ DEFCON31 | Kaggle
  - AI Village Capture the Flag @ DEFCON | Kaggle

How do you approach a new competition?

- Start with reading discussions, understanding the domain knowledge, then summarize everything important.
- Read the baselines, do eda and choose the best validation.
- if you have time, build your own baseline from scratch then replicate all the ideas in the best public nbs into yours. Otherwise, choose a public nb then build on it.
- Then add your ideas🫡.

(Don’t forget to eat and sleep btw👀)

Final Notes: Difference between Kaggle Competitions and real-world projects?

- Problem Definition.
- Data Availability.
- Model Deployment.
- Evaluation Metrics.
- Resources and time constraints.

Final Notes: Should i use GPT?

- Use it Extensively! But with two rules:

Final Notes: Should i use GPT?

- Use it Extensively! But with two rules:
    a. Don’t copy! Just write.

Final Notes: Should i use GPT?

- Use it Extensively! But with two rules:
  a. Don’t copy! Just write.
  b. Don’t write something you don’t understand!

AI career directions

Research Path

- ***Focus:*** *Formulate and test new hypotheses, design novel architectures, publish papers*
- ***Work Environment:***
  - *Universities labs (MIT, CMU, Stanford, KAUST, Top Universities).*
  - *AI labs (OpenAI, DeepMind, Microsoft Research).*

Industry Path

- ***Focus:*** *Build, deploy, and maintain ML systems in production, model training pipelines, inference APIs, monitoring, and scaling.*
- ***Work Environment:***
  - *Tech giants (Google, Amazon, Microsoft, Apple).*
  - *AI companies & startups.*

What to do next?

- ***Skills + Connections:*** *You need both!*
- ***Build a strong foundation:*** *Both theory and practical.*
- ***Blend both worlds:*** *Research and industry.*
- ***Hands-on projects:*** *Contribute to open-source, compete in competitions & hackathons, build end-to-end demos, contribute to research, and develop real-world projects.*
- ***Networking:*** *Attend workshops, join AI communities (Slack, Discord, conferences). These will offer you many opportunities!*

👀

KAUST vs KKU Kaggle Competitions!

- *Build the best AI model!*
- *Starts next week, one per week on Thursday (i hope?).*
- *4-5 hours (lab + enrichment time).*
- *Mainly to give you more hands-on experience.*