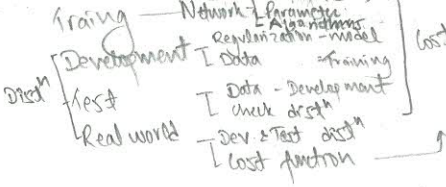


# Machine learning Strategy

ML Strategy - evolving

Orthogonalization - Early stopping

Data set



Loss function / Evaluation Metric

single numbers - Quick iteration

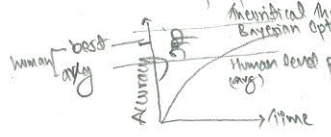
Regression - Square Error

Classification - Precision, Recall, Average

if already weighted accuracy

optimizing & satisficing

Human level performance



Advances in ML - comparable workflow - efficient

why compare

Human level performance

1 - Get more labelled data

2 - Manual Error Analysis

3 - Analysis of bias/variance

Super human level performance

Performance

Structure of Data

Amount

Natural Language perception task

Evaluation Metric - Place target

Rank order preference - Aim accurately

Orthogonalization

Online advertisement

Product recommendation

Loan approvals

Speech

Image

Medical

Recognition

Surpass human level performance

Error Analysis

Performance (Dev Set)

Data

Features

New problem - Data

Hand engineering features

focus

Specific type

Sense of option to pursue

Mathematical rigid formula

Error analysis

Incorrect prediction (Dev Set) & reason (FP/FN)

Absolute Error

Relative Error

# cases

Dev Error

Incorrect prediction (Dev Set) & reason (FP/FN)

ceiling on performance

Incorrect labelled data

Training set

quite robust to random error

no error

data set size

systematic error

effort ↑

Development set

Error analysis + Incorrect Label

Error (Summed Label)

Error (Other reason)

overall dev set error

guide - choice of algorithms

Action

dev set & test set ~ same dist<sup>n</sup>

apply changes in test set

a/c = right & wrong cases (predicted)

↑ data (skip)

biased error

Project

1) Build system quickly & iterate → Many direction → bias + variance error analysis → pick → continue

2) Prior Experience + Academic literature → start with some base

Dist<sup>n</sup>

Training

Dev/test ~ Same dist<sup>n</sup> target

different dist<sup>n</sup> sources → Bias & Variance (guidance)

Training → Dev Set

Data mismatch error - Manual Error Analysis

Data with similar dist<sup>n</sup>

Artificial data synthesis

Learning

sequential - Transfer learning - retain layer from pretrained model

fine tune

existing layers / model

add

pretrained model - same data / not sufficient data

features (low level)

multi task learning

different class in same model = loss function

- shared low level features

- Data - each task - similar

- Big single network

End to End learning

multiple stage of processing

intermediate component

INN

individual problem - simple

Amount of data

↓ Hand design features