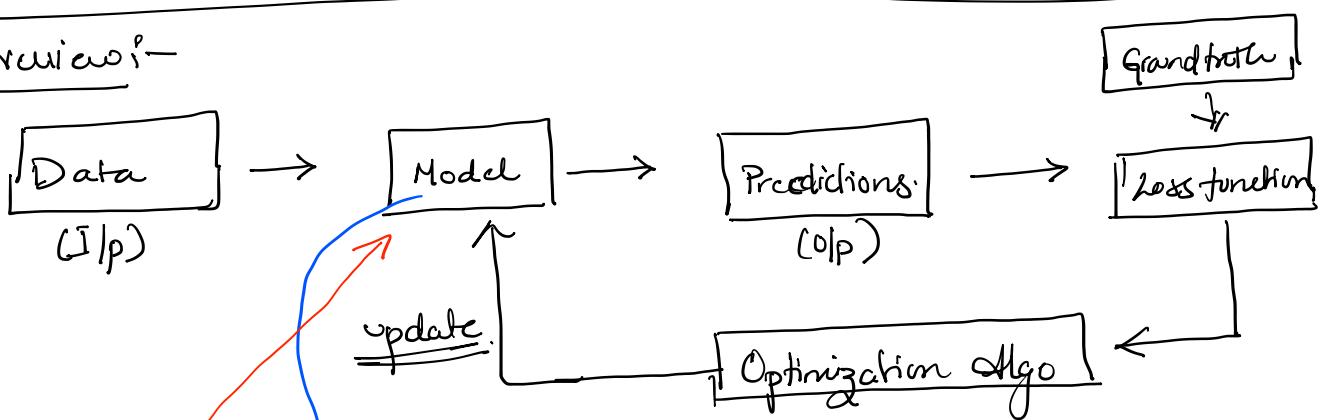


Hyperparameters :-

Overview:-



① Model parameters: $\Rightarrow (w)$ $\rightarrow y = \underline{\underline{w}}x + b$ } model parameters.

Pipeline : \rightarrow Parameters

Parameters:

- 1) Epochs.
- 2) Batch size
- 3) Type of optimizer.
- 4) Learning rate
- 5) LR decay.
- 6) LR scheduler.
- 7)
- ⋮

not learnable

learnable
(w)

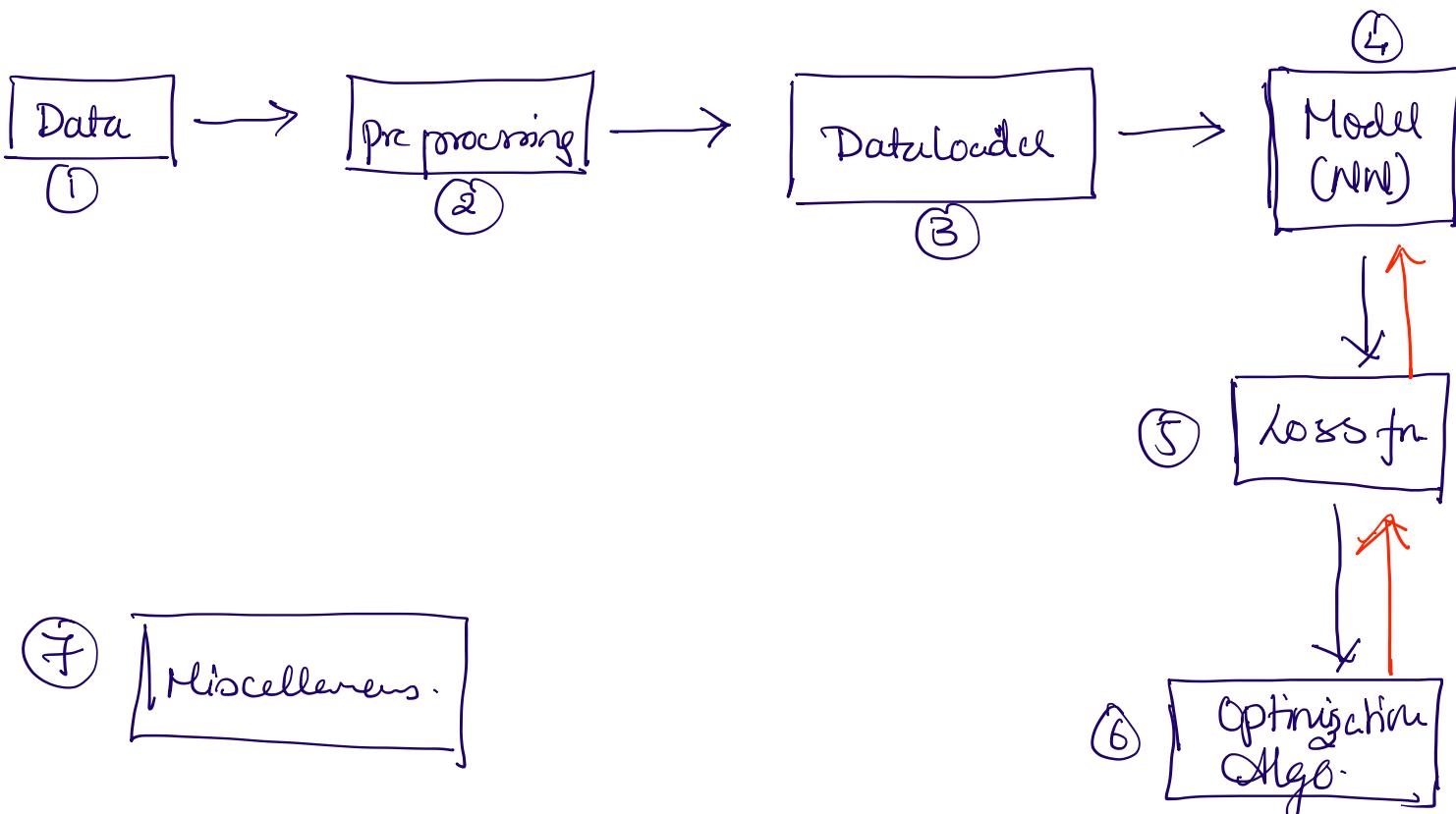
User defined / derived.

They control the overall training process but are not directly learned as a part of learning.

Can be grouped into various sections (parts) on the overall pipeline.)

Detailed pipeline Overview:

(for hyperparameter)



① -

② \rightarrow scaling
 \rightarrow augmentation. (Preprocessing)

choice / type
of augmentation parameters of
augmentation.

rotation \rightarrow degree ($30^\circ | 40^\circ \dots$)

flip \rightarrow type (HP / VP / both).

③ Dataloader.

\rightarrow BS \rightarrow shuffle.

helps in generalizing /
model not memorizing
the seq. of i/p's.

Not Hps
Not Hps \rightarrow #Workers \rightarrow time of training

pin memory (u o u)

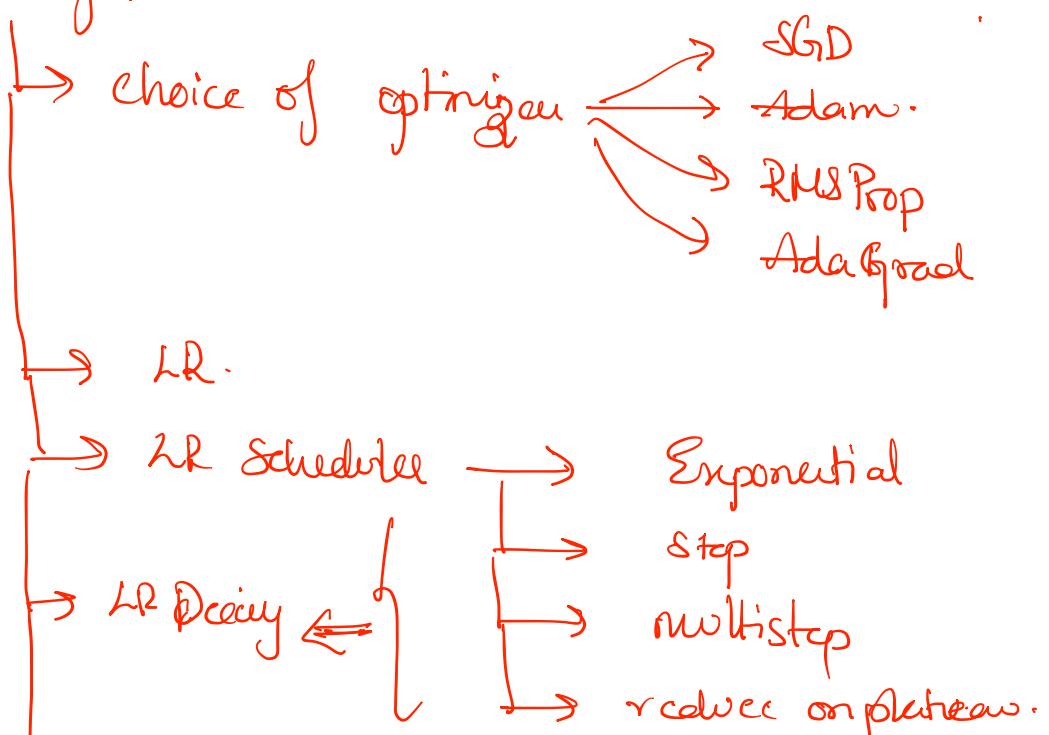
④ Model (Net) :-

- Neural Architecture Search (NAS)
- # layers.
 - # kernels.
 - # of neurons in FC layers.
 - Activation functions.
 - Batch norm parameters.
 - kernel sizes , stride, padding, channels.
 - Proportions.

⑤ Loss : → task dependent.

→ governs the surface of optimization.

⑥ Optimization Algo:-



Weight decay.
momentum.

7

Misc:

↳ Epochs.

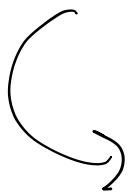
:

Q) How do we select the best HPs?

say, \hookrightarrow LR (n) : $\{0.1, 0.001, 0.005, \dots\}$

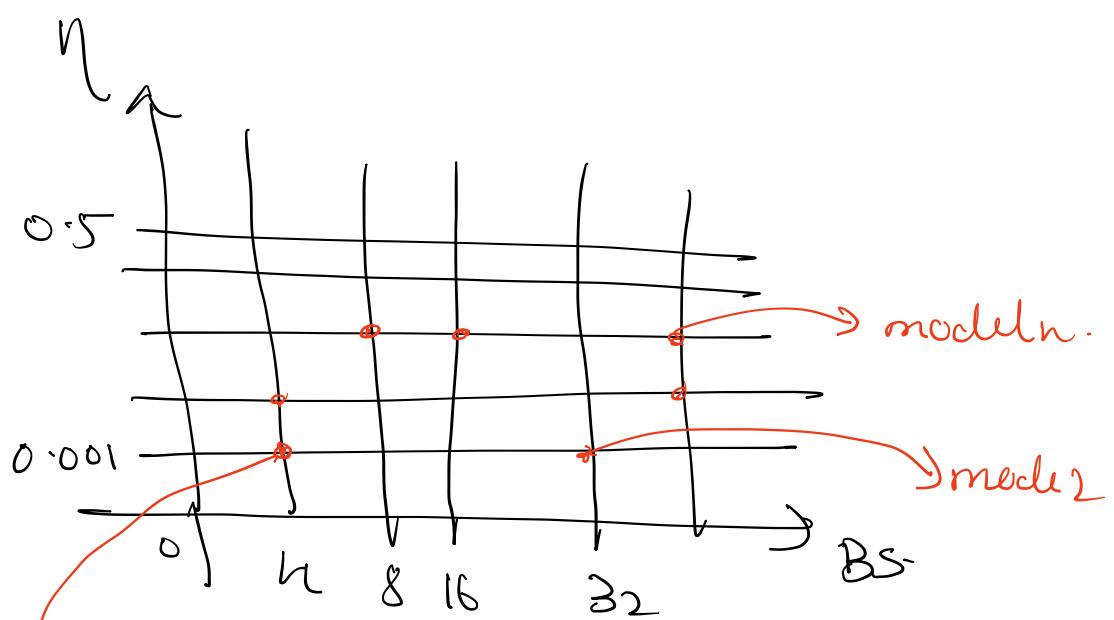
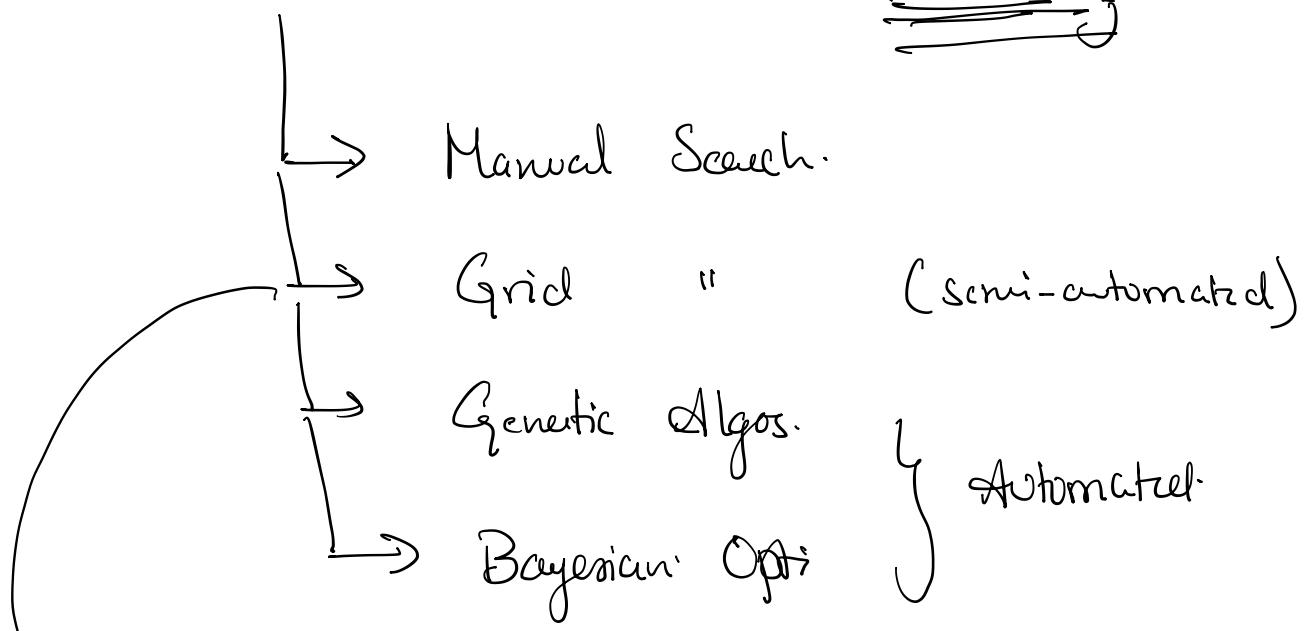
\hookrightarrow which one is best /
more suitable for a
task at hand?

{ Given data
Given algo
Given problem [task] }



Hyper parameter Optimization (HPO).

{
 → Optional
 AX }
 |



repeat
until
converge
in
HPS -

train model with setting (model1)

