300-300-15
625
B:500

Config 1: pD tuned back 1000

Config 2: // but phi x 2
↳ still overtraing but phi is much better!

still overtraing...
(Try next config 3: simple mse
(can also try, and this is next simple mse with
only 2, ϕ & p)

config 3: pD tuned back 1000
phi is now x 10
↳ doesn't do that much better in ϕ!

Issue: we have no proper overtraing in loss function
but we have one in mae (is it okay?)
and also val loss is very jumpy...

config 4: Only mse over all the variables
↳ still overtraing...
Ok Not overtraing less than config 2, however
it has better overall regression
(ϕ less good with D)

200-200-6

config 5: use mse to only regress the neutrino 3 vectors

still overtraining, very similar to config 2, better overall but less good

Next step: do all you can to stop this most simple configuration from overtraining.

config 6: 64-64-0.1-6 to check if that could keep from overtraining

Phi & eta are well reconstructed!
Always a difference of
11% between val & all for p components
5-3%. for ϕ & η components

Next: continue this search (and fix the issue) as well as playing w/ weights → prove we can get better ϕ & η out

Great for the | test file! (Will be shown & shared
end

9/12 - Aims: - Continue trying to fix overtraining
- When this is done: play with weights & follow planning on update 7.

config 7: Try to regress only 1 parameter
↳ still overtraining, just too many parameters?

Try kernel regularizer 0.2 L2 → still overtrain
" " 0.2 L1 L2 → still
2 L1 L2 → still.

Try with less inputs. → still

One more drop out layer 0.5 in between → still

Change size of train set: 80%. → still

Include the regressed vars in inputs → Not anymore but weird
val loss << loss
0.44 ↗ ↖ 2.6
could mse be calculated differently?
Like I don't understand.
See if regularizers save the day

try with regularisers 0.1 L1L2

loss = 3.45
val_loss = 1.16 } so weird

Try with less nodes: 32

loss = 5
val_loss = 0.7 } what

Changed batch size to 1000

loss = 5.4
val loss = 1.37

(according to Google: Larger batch size = smoother training).

changed L1L2 to 0.4

l = 6
vl = 1.8

Is the data ordered a certain way? Coz that the two wouldn't be the same?

And then back to 0.1

Batch size: 2^6 ← much slower training (too small)

l = 5.5
vl = 1.1

Now: Tune down

(100) d_p : 147706 → (1/150000) →

(1/1000) d_{all} : 8.6 → (1/800) → config 10

(100x) $d\phi$: 460 → (1/4.5) →

Have everything around 1 s and then do again with m_H & m_C → all same weights & then change.

m_H : 0.91 (1)

m_C : 0.05 (1)

d_p : 0.78 (1/1500)

d_{all} : 0.18 (1/800)

$d\phi$: 0.67 (1/4.5)

} config 10

Next we want: more d_p, more H & C
more ϕ, less d_{all}

eta is always well reco!

Results

m_H : (10/4) ~0.5

m_C : (4) 0.36

d_{all} : (1/10000) 0.14

d_p : (1/150) 7.3

$d\phi$: (4/4.5) 2.2

} config 11
↳ good on η & ϕ (similar to c10)

? looks complicated...
need to go through ⊖... will attempt but since ? is naturally well ~~res~~ is it an issue?

ρ is still very poorly reconstructed...
check

$d\rho =$	~~0.8~~ 1003	(1)
$d\text{all} =$	0.8	(1/10000)
$d\phi =$	3.5	(4/4.5)
$d\text{MH} =$	~~6.4~~ 6.7	(4)
$d\text{mC} =$	10.2	(10)

} config 12 (not amazing)

Fetch → need to re-run each time...

config 14 (w/ ρ)

8.8	$d\rho$	8.7	(1)
0.18	$d\text{all}$	0.18	(1/10000)
7.0	$d\phi$	7.0	(4)
0.7	$d\text{MH}$	0.7	(4)
0.55	$d\text{mC}$	0.6	(10)

} config 13 (still less good than 6 std use but better means use)

Next: include ZP & car matrix next

Identical don't even mention

Done: ~~config 14~~ - same as 13 but with ZP as well (shouldn't change much, it is the same as TS in this case)

Next: Include not cov (2)
ip cov } (3)
ip sig } (1)
ip alg }

Take channel:
nva_dm_1 = 2
nva_dm_2 = 1

$a_1 \rightarrow 1pi + 2pi \odot$
$\int$

First check: - The exact same way: vis 1 = pi + p⊙
vis 2 = pi + p⊙

- Changing vis 1 for: pi + 2pi⊙ (also in loss!)
- What hang IP & not hang it changes
- What hang CoV and not hang it changes.

Config 16: Exact same way but with a_1

Maybe use 3 arrays instead?
As seems like we know well which one to take here?
Or impact parameter?

Try nva 1 : 10 → and then nis 1 = pi_1 / pi 2_1 / pi 3_1 } corner

nothing?

wrong decay mode

and ask about other decay

Config 15: a1_rho 'normal'
+ IP

Config 16: nis = pi_1 + pi2_1 + pi3_1

config 17: config 16 but add ipsy 1 & 2

config 18: add SV info + Cov matrix.

(about other channels)

Questions:

- Is the nva decay mode 10 filled?
- For the dm 2: How do we form the a1?

plot it! → there seems to be stuff!

So we only have some combinations?
is very important!

Next: config 15 - 18

- clean complete comparision with nice results
- Read & understand Espinner paper(s)

according to graph

1	2
1	10

is populated

Why can't we see anything? → check their git ...

Nothing → could it be because no ~~graph~~ gen vings?

So dump: non wong the right one...

tau 1 x == 1

(tau 2 (looks like) all the time (because we do not need with 10)

is okay looks doable → it works

Nuttinos-a1-rho.ipynb → has $\begin{cases} 10 \\ 1 \end{cases}$ a1 → 3pt

Could c15 work because the first π is the most important? And the rest doesn't matter? (as much)

Issue: There will be less data → Higher risk of overtraining.

dp_all = 0.16
d_p = 8.1
d_phi = 6.6
dmC = 0.3
dmH = 1.7

(1/10000)
(1/150)
(4)
(10)
(4)

	1	2
ϕ	-0.02±0.9	0.02±0.7
η	-0.04±0.61	-0.04 ± 0.56
p	0.4 ± 29.0	0.78 ±29.68

Config 15

11/12/20 - Continue with the a, searches

Config 15: pretty good with all the σ too! (Large error on p but right approximation!)

Change vars1= pi2+pi3+pi (~~2~~ only)

Config16
- (same as 15)
- dpall = 0.15
- dp = 7.7
- dphi = 7.41
- dmH = 0.83
- dmC = 0.55

	1	2
ϕ	0.26 ± 1.2	0.03 ± 0.71
η	-0.07 ±0.63	0.01 ±0.55
p	8.17 ±28.7	1.32 ±28.3

5

clearly not correct way of estimating a_1...

Check the mass plot...

Some are at 0! (okay, it is good, right mass)

Is very different mass distributions...

Is Not happy with the hits a 0 → should we remove them?

Config 17: (config 16 but add ip-sgn)

(yeap all seems right, in the Loss function we are having the correct refs (to the iss products)).

Much better than 16

	1	e
ϕ	0.02 ±0.8	0.05 ±0.77
η	0.06 ±0.62	0.06 ±0.54
ρ	0.78 ±28	0.67 ±27

17

(same as previously)
$d_{pall} = 0.14$
$d\rho = 7.4$
$d\phi = 6.9$
$d_{mH} = 0.75$
$d_{mC} = 0.44$

Next: • I want to check what TauSpinner looks like
• Check config 19: $\pi^0 + \pi = a_1$ (wrong)
is and ip sign.
• Include an eta part?
• Add SV & Cov matrix info

Still ~12% difference between ~~train~~ all & val (okay?)

Okay so 19 is still worse than config 17 (not by far, it is actually better at all that is ρ (except for σ_ρ) but $a_1 = 3\pi^\pm$ makes sense → keep it).

dpall = 0.16
dp = 8.1
dϕ = 7.6
dmtl = 1.7
dmC = 0.41

} config 19

	1	2
ϕ	-0.22 ± 1.26	0.01 ± 0.77
η	0.04 ± 0.67	-0.13 ± 0.53
ρ	0.31 ± 29.6	-0.01 ± 29.4

Shall investigate next:
- sv_x_1 ✓
- sv_cov
- ip cov
- met cov

$d_{pall} = 0.14$ (same)
$d\rho = 7.4$
$d\phi = 6.8$
$d_{mH} = 0.76$
$d_{mC} = 0.485$

} config 98

	1	2
ϕ	0.86 ± 1.12	-0.03 ± 0.74
χ	0.01 ± 0.64	0.04 ± 0.55
ρ	2.13 ± 28.49	1.5 ± 28.05

Check also if we only put sv 1 (cause not as relevant for ρ I think?) config 20

We have some nans (very small deltas)

(could pump up the overall loss...)

config 20 {
$d_{pall} = 0.15$
$d\rho = 7.5$
$d\phi = 7.8$
$d_{mH} = 0.476$
$d_{mC} = 0.47$

	1	2
ϕ	0.01 ± 0.8	0.13 ± 0.91
χ	0.01 ± 0.51	0.05 ± 0.57
ρ	1.31 ± 28.68	2.68 ± 28.82

(everything is pretty similar... cs getty side tracked here)

Before moving on: check only regressing ϕ and η and p with NN → Ex is better ②

And taking ϕ, η and regressing only α.

↳ with same inputs.

① Add the covariance matrix info. (cov IP, SV and met) ✓ — is a lot of memory

(What about more layers?)

actually we have metcov (but not metCov!) → set up to run in bchg?

(Note: always EP sign)

	cov IP1	cov IP2	cov SV1	cov SV2	cov met	IP1	IP2	SV1	SV2	met
cA (0)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
cB (1)	×	×	×	×	×	✓	✓	✓	✓	✓
cC (2)	×	×	×	×	×	×	×	✓	✓	✓
cD (3)	×	×	✓	✓	×	×	×	✓	✓	✓
cE (4)	×	×	✓	✓	✓	×	×	✓	✓	✓
cF (5)	×	×	×	×	✓	×	×	×	×	✓
cG (6)	✓	✓	×	×	✓	✓	✓	×	×	✓
cH (7)	✓	×	✓	×	✓	✓	×	✓	×	✓

And in the meantime → I can do the mse regression of ϕ, γ, p.

Should I do it on the IC cluster?

(Maybe only download what I need?
(Well at some point we'll need everything...)

Okay check about the data saving as .txt file.

Is okay this works is now set up a .py

Remember to increase swap space

(added to .bash the line to make that swap permanent.)

Check now if it doesn't crash anymore

Is nope, we're good.

Have x_ref and then just keep the one we want.

Okay takes time to take everything in but apart from this we are good?

check if we can load everything in one ~~one~~ go

↳ there are no pi0 in the a, channel (makes sense)

↳ then set up the total bck runs (maybe leave them overnight rather than when I am working...)

571,000 events in the branch (but it looks like it's working!

Yep, should all be working and say the right stuff now - is awsome for testing the whole shff (run overnight & such).

↑ yes it works