# Intro

This doc contains notes I took while figuring out how to make a mixed data net of 1d conv and dense layers in tensorforce. The target net looks like image below. It is highly likely that lots of the notes here relate to me learning tensorflow, since I was starting with minimal understanding.

Resampled track profile

Height, speed, travel

State(53,1) [resampled track profile(50,1), height off ground(1,1), speed(1,1), suspension travel(1,1)]

Conv1d

output

dense

I should note that initially, with a network spec of multiple dense layers, the PPO agent was learning something, so I had something working with tensorforce.

Disclaimer: why did I chose that state formulation? I wanted to create a state that provided the most important information that a human rider would use, but in the most compact form, e.g. a 1d array of scalars.

# Environment state shape

Through many trials, I learned that when the first layer in network\_spec is a dense layer, the state shape should be (53, ), while if the first layer is a conv1d, the state shape should be (53,1).

At model.py line 281, the state shape passed from the env is modified. In my case, (53, ) => (1,53, ), and (53,1)=> (1,53,1).

states, actions, reward = self.fn\_preprocess(states=states, actions=actions, reward=reward)

at line 538 does this:

self.fn\_preprocess = tf.make\_template(

name\_='preprocess',

func\_=self.tf\_preprocess,

custom\_getter\_=custom\_getter

)

See [link for make\_template](https://www.tensorflow.org/api_docs/python/tf/make_template).

Where self.tf\_preprocess is defiend at line 885:

# States preprocessing

for name in sorted(self.states\_preprocessing):

states[name] = self.states\_preprocessing[name].process(tensor=states[name])

to be continued …..

# trials

each trial below provides the states and network\_spec args passed to PPOAgent, and the resulting errors and locations in the stack where state is modified.

## t001

Env.states= {'shape': (53, 1), 'type': 'float'}

single conv1d layer.

network\_spec = [

dict(type='conv1d', size=32, window=3,stride=1,padding='SAME',bias=False,activation='relu',l2\_regularization=0.0,l1\_regularization=0.0),

]

Error:

File "C:\Users\albert.mathews\AppData\Local\Continuum\anaconda3\lib\site-packages\tensorforce\core\networks\layer.py", line 577, in tf\_apply

'Invalid input rank for linear layer: {}, must be 2.'.format(util.rank(x))

TensorForceError: Invalid input rank for linear layer: 3, must be 2.