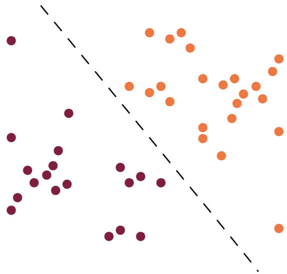
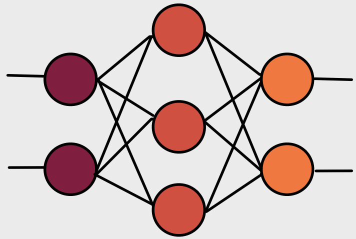
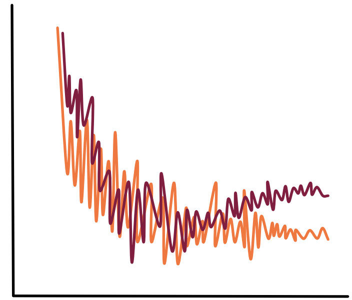
Overview



1. Find some data

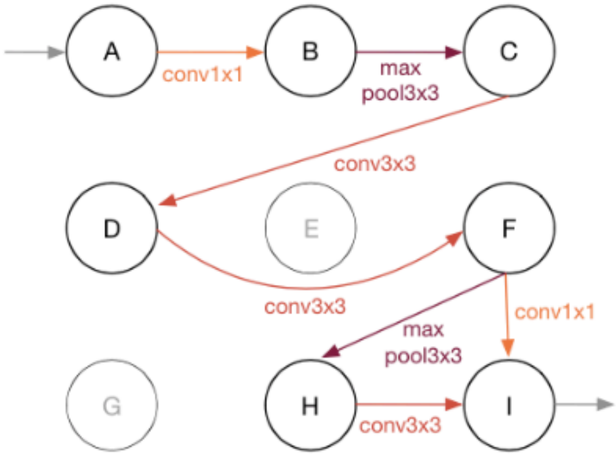


2. Design a network

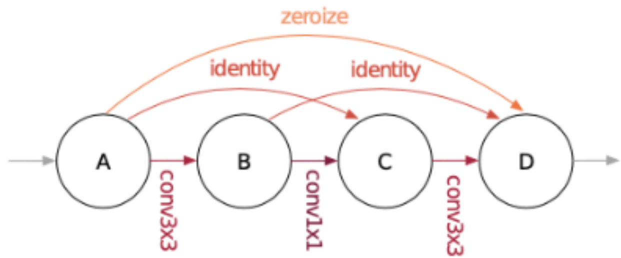


3. Train the network

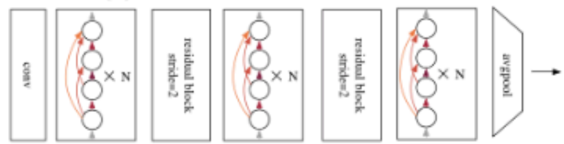
Automating Architecture Search



(a) A NAS-Bench-101 cell.

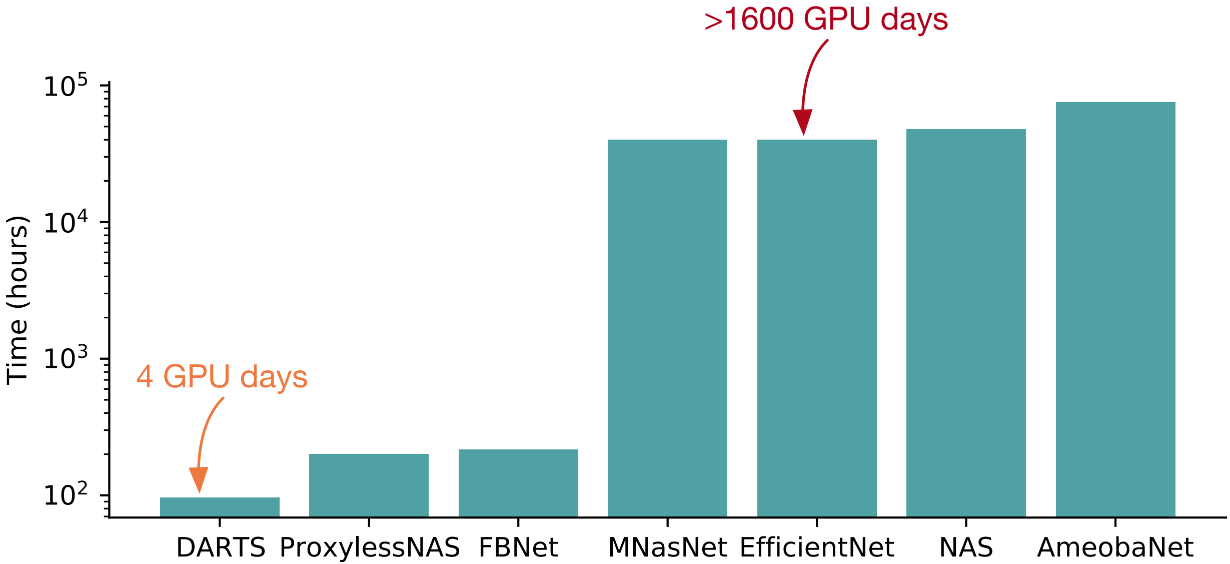


(b) A NAS-Bench-201 cell.



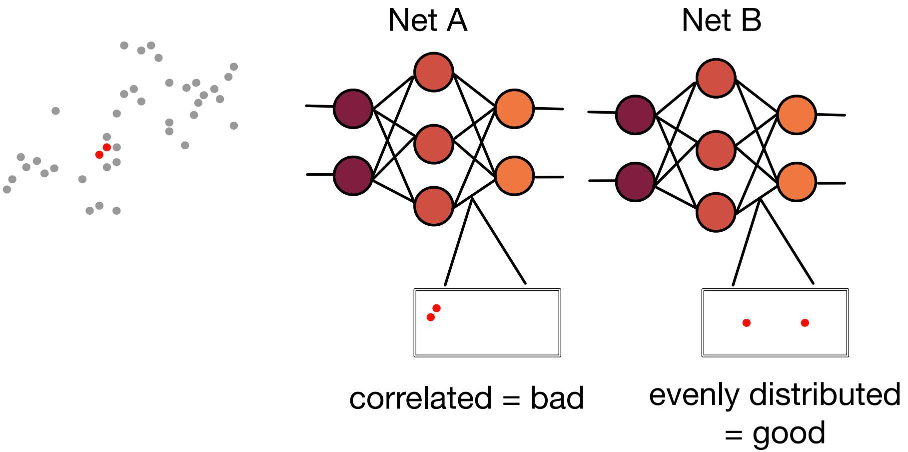
(c) The skeleton for NAS-Bench-101 and 201.

NAS is slow



>1600 GPU days
4 GPU days
Time (hours)
10^5
10^4
10^3
10^2
DARTS
ProxylessNAS
FBNet
MNasNet
EfficientNet
NAS
AmeobaNet

Is there a property of networks *at initialisation* that we can use instead of learning a policy?

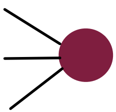


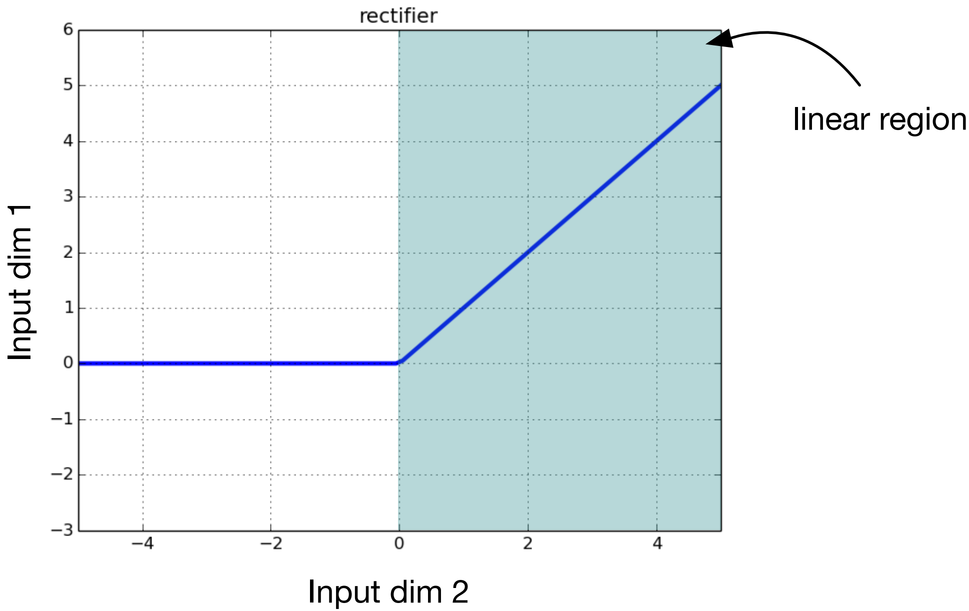
Linear Regions in Neural Networks

Input

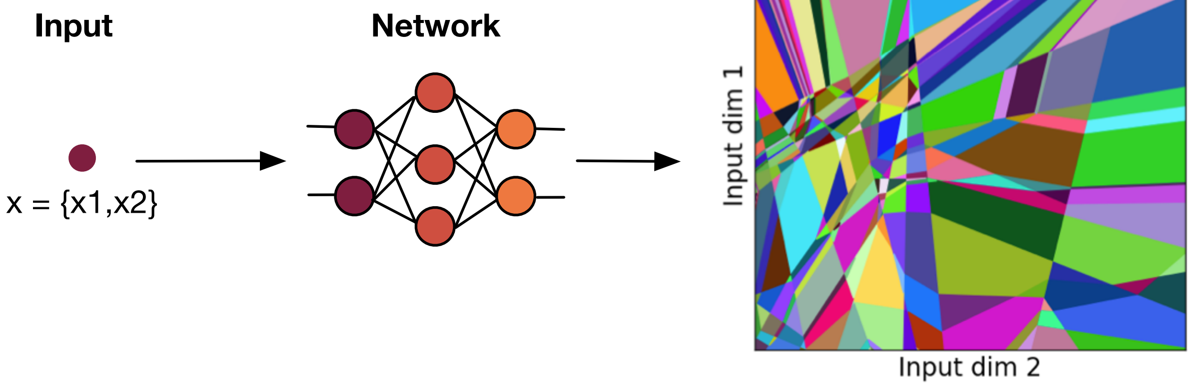
x = {x1,x2}

Neuron



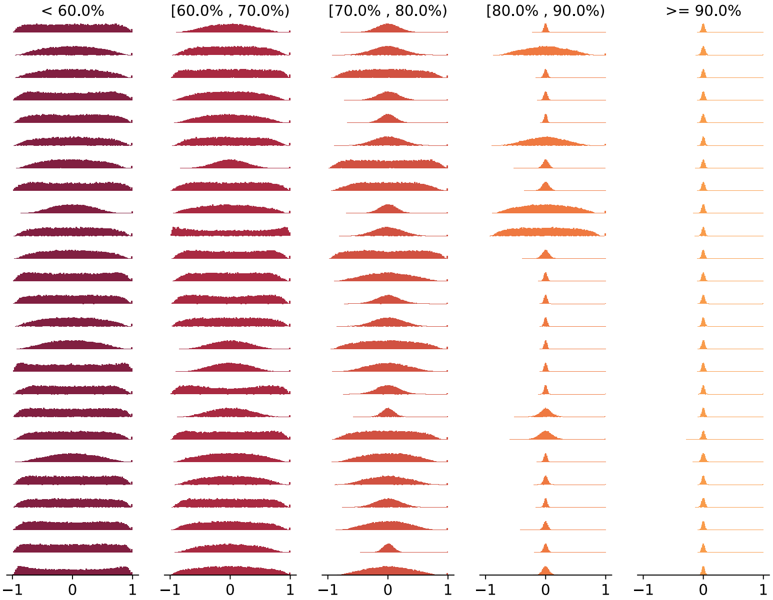


Linear Regions

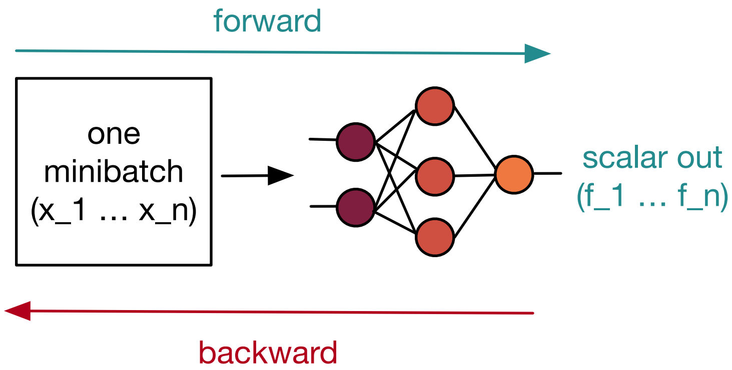


[1] Hanin, B. and Rolnick, D., 2019. Deep relu networks have surprisingly few activation patterns. In *Advances in Neural Information Processing Systems* (pp. 361-370).

Correlating Linear Regions



Correlating Linear Regions



$$\mathbf{J} = \left(\frac{\partial f_1}{\partial \mathbf{x_1}} \quad \frac{\partial f_2}{\partial \mathbf{x_2}} \quad \cdots \quad \frac{\partial f_N}{\partial \mathbf{x_N}} \right)^\top$$

Scoring Jacobians

Take the covariance of J:

$$(\boldsymbol{\Sigma}_J)_{i,j} = \frac{(\mathbf{C}_J)_{i,j}}{\sqrt{(\mathbf{C}_J)_{i,i}(\mathbf{C}_J)_{j,j}}}$$

Take the KL divergence between Gaussian with kernel Σ and Gaussian with uncorrelated kernel:

$$S = -\sum_{i=1}^{N}[\log(\sigma_{J,i} + k) + (\sigma_{J,i} + k)^{-1}]$$

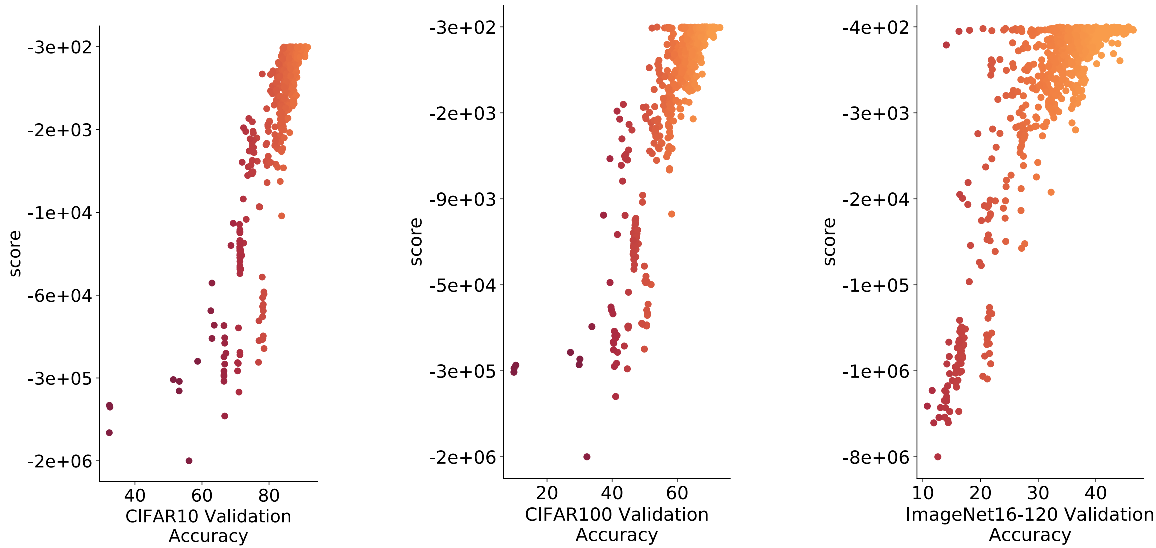
scalar quantifier of linear map similarity

In Code

All we need to do to get a score for a network is:

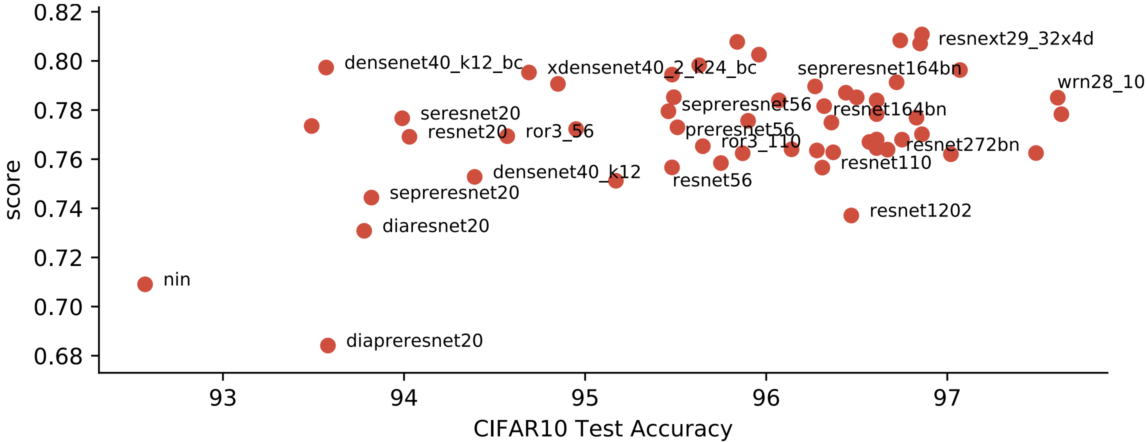
```
def eval_score(jacob):
    corrs = np.corrcoef(jacob)
    v, _  = np.linalg.eig(corrs)
    k = 1e-5
    return -np.sum(np.log(v + k) + 1./(v + k))
```

Results



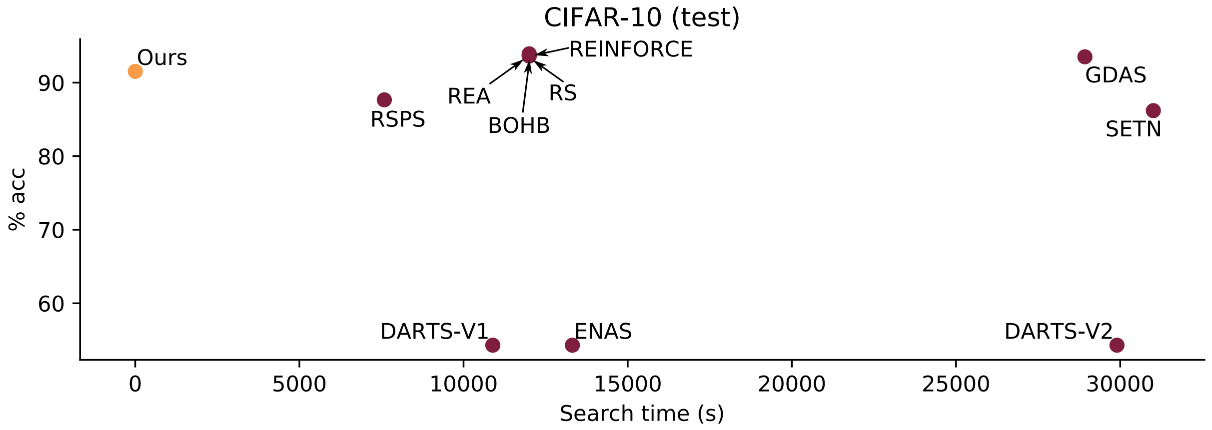
score
-3e+02
-2e+03
-1e+04
-6e+04
-3e+05
-2e+06
40
60
80
CIFAR10 Validation Accuracy
score
-3e+02
-2e+03
-9e+03
-5e+04
-3e+05
-2e+06
20
40
60
CIFAR100 Validation Accuracy
score
-4e+02
-3e+03
-2e+04
-1e+05
-1e+06
-8e+06
10
20
30
40
ImageNet16-120 Validation Accuracy

Results



score
0.82
0.80
0.78
0.76
0.74
0.72
0.70
0.68
93
94
95
96
97
CIFAR10 Test Accuracy
densenet40_k12_bc
xdensenet40_2_k24_bc
sepreresnet164bn
resnext29_32x4d
wrn28_10
seresnet20
resnet20
ror3_56
sepreresnet56
resnet164bn
preresnet56
ror3_110
resnet272bn
resnet110
densenet40_k12
resnet56
sepreresnet20
resnet1202
diaresnet20
nin
diapreresnet20

Results



CIFAR-10 (test)
Ours
RSPS
REA
BOHB
RS
REINFORCE
GDAS
SETN
DARTS-V1
ENAS
DARTS-V2
% acc
90
80
70
60
0
5000
10000
15000
20000
25000
30000
Search time (s)

Questions?

Code: https://github.com/BayesWatch/nas-without-training