JS Corrections of Neural Networks

The Neural Net

The model we work with is RegNetX 32GF by Radosavovic et al. This is a image classification model, which assigns an image to one of the 1000 labels.
Suppose we input the following image:

The network will then classify this image and give a 89.76% certainty that this is an image of a tench (a type of fish).

The Neural Net

The basic architecture of this model is stem (conv layer) + body (4 stages of sequences of identical blocks) + linear layer.
The identical blocks in RegNetX is given by residual bottleneck block given by 1x1 conv + 3x3 group conv + 1x1 conv, with 3 parameters for the block architecture.
The RegNet paradigm is to look at the design space and optimize the network structure. This design space has 16 degrees of freedom (number of blocks + block width + bottleneck ratio + group width).
The RegNet architecture provides a family of networks in different flops and are comparable to state of the art models (in 2020).

Preview of Results

We evaluate our model using the test set MatchedFrequency of ImageNetV2, which contains 10000 images divided into 1000 classes. Here the results are in net change of correct predictions.

A, q	0	$e, 1$	$e, 2$	$e, 3$	$e, 4$	$m, 1$	$m, 2$	$m, 3$	$m, 4$	$(m, e), 1$	$(m, e), 2$	$(m, e), 3$	$(m, e), 4$
Top1	0	+1	+10	+4	+3	+1	+10	+4	+5	0	+11	+2	-7
Top5	0	+1	+10	0	-1	+2	+13	-3	-2	0	+10	0	-11

Spectral Properties

The model has one linear layer in the end, in the form of

$$y = W^T x + b,$$

with W a 2520×1000 matrix. So $p = 2520, n = 1000$.
We have the spectral decomposition:

$$\frac{1}{n} W W^T = \sum_{(s,h)} s^2 h h^T$$

where (s, h) are singular value and left singular vector pairs.

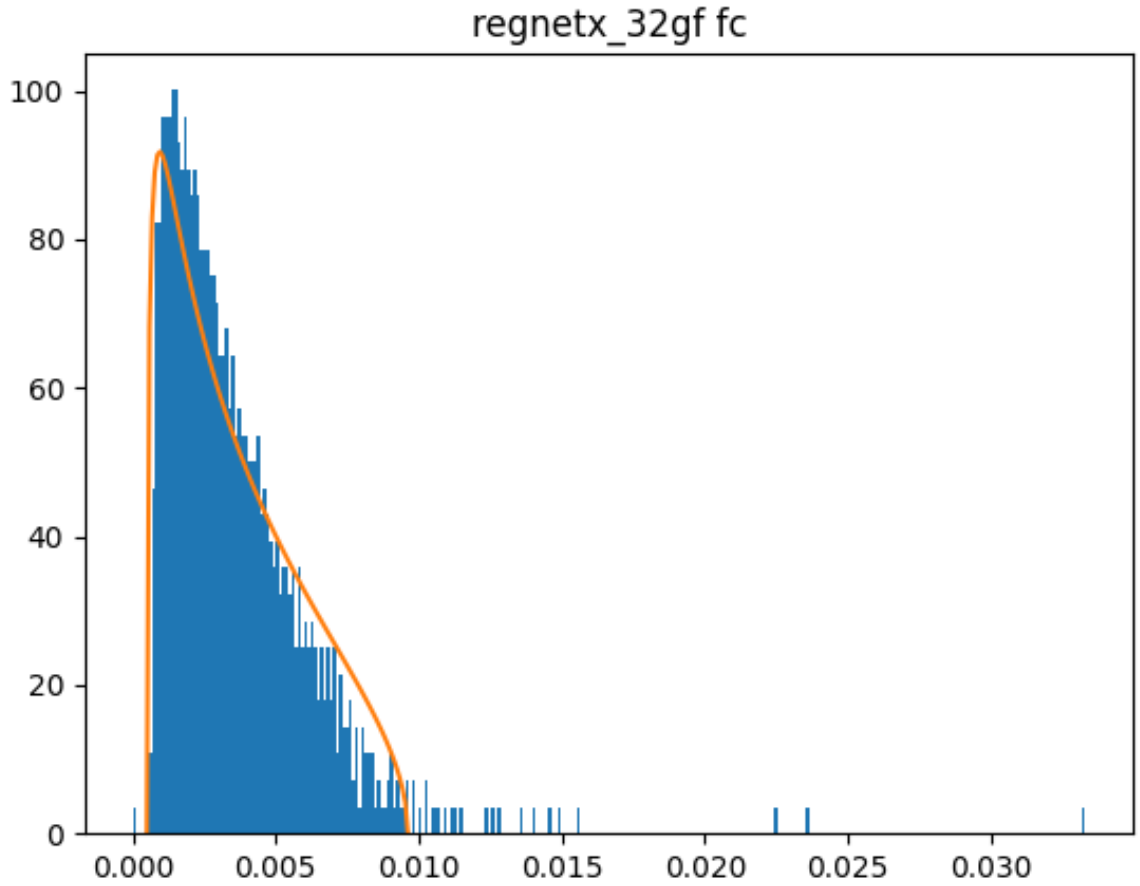
Spectral Properties

We consider a thought experiment in which we have an infinite training set which produces the weights Q satisfying:

$$QQ^T = \frac{1}{n}\mathbb{E}[WW^T].$$

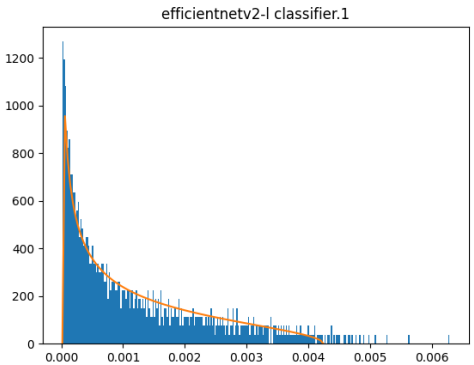
Spectral Properties

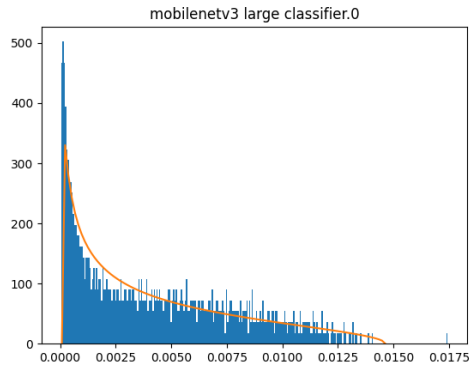
We examine the eigenvalues of the covariance matrix of W. Our crude estimate of the number of spikes $= 23$.

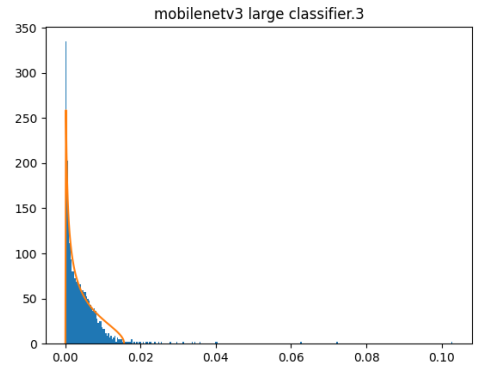


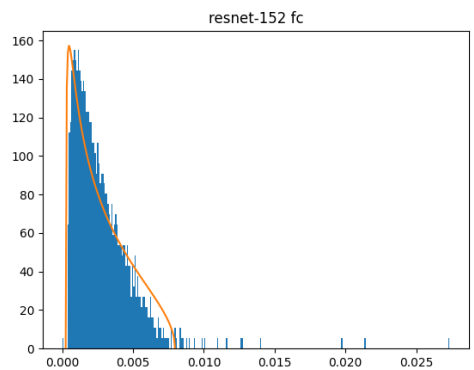
Spectrum of Linear Layers of Neural Networks

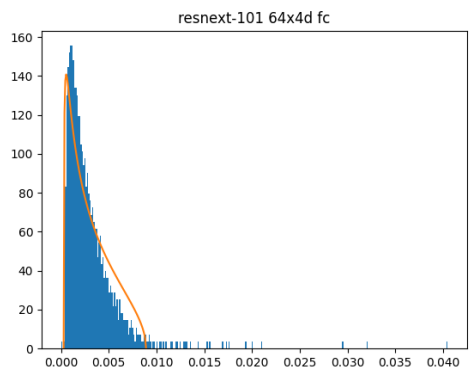
This behavior is not surprising and is quite common in neural networks (paper by Martin & Mahoney for reference):

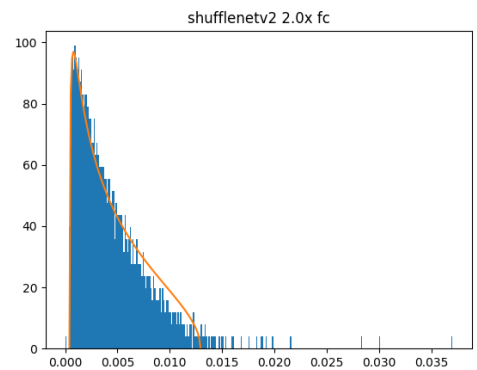






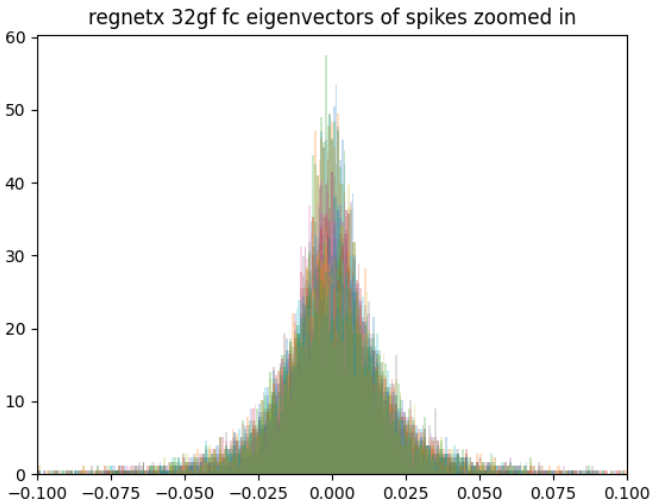


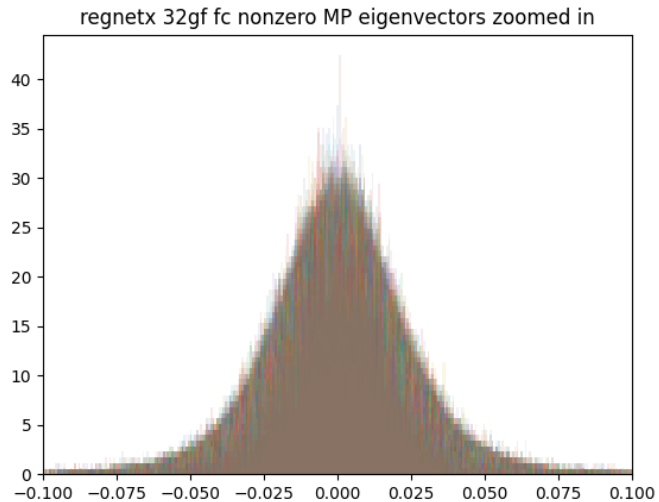




Spectral Properties

We see a qualitative difference between the eigenvectors of the spikes and the eigenvectors of the MP bulk:

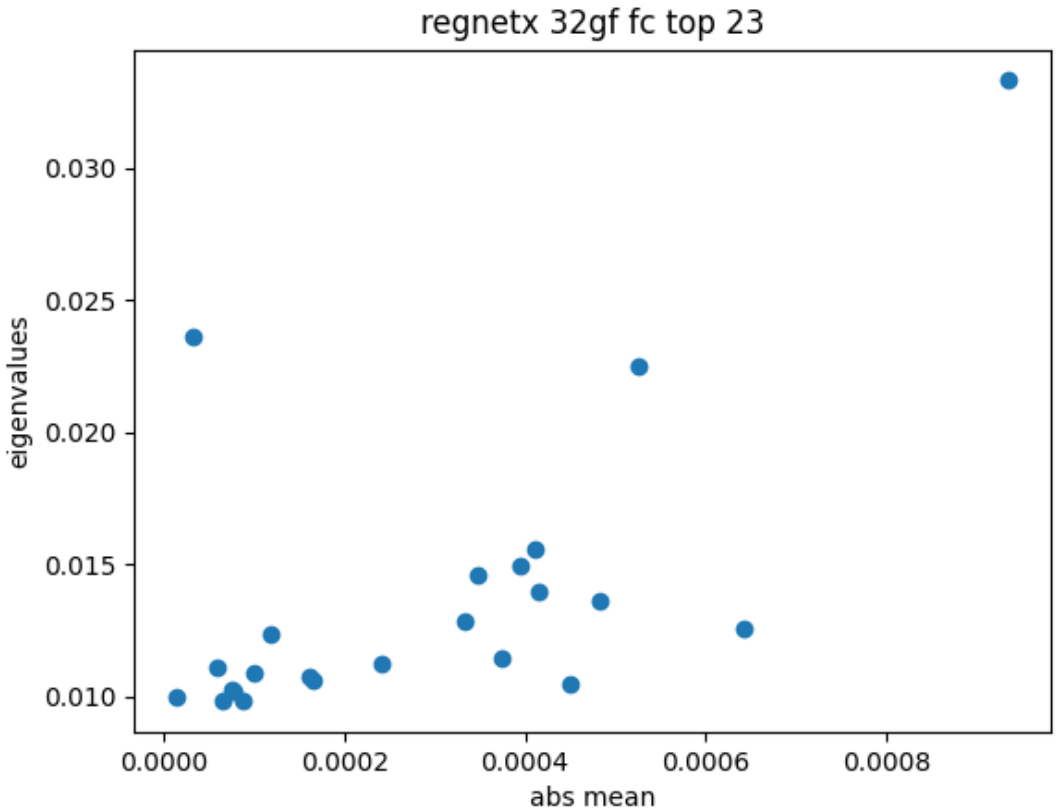




The eigenvectors of the MP bulk looks like pure noise, i.e. uniformly distributed vectors on the $p-1$ dimensional sphere.

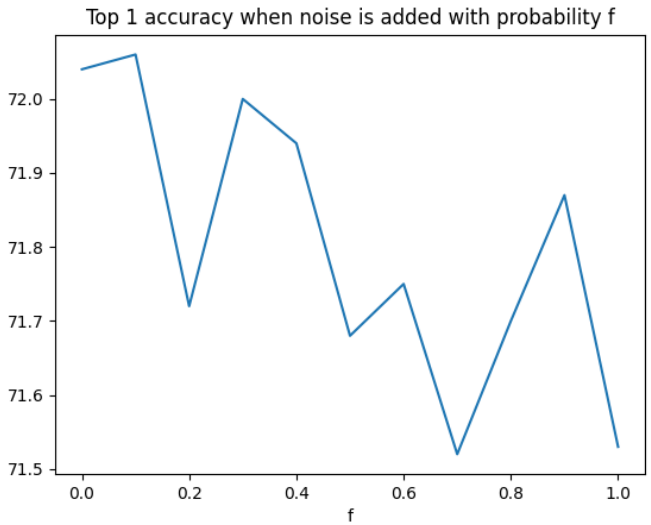
Spectral Properties

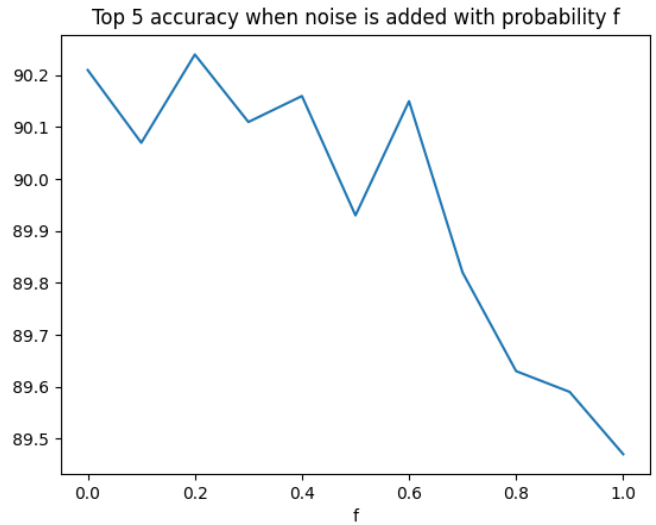
This scatter plot of eigenvalue vs absolute value of the mean of the eigenvector entries shows that large eigenvalues are associated with eigenvectors that are not noise like those in the MP bulk.



Tweaking the Weights

Our goal is to adjust the weights W to improve the performance of the network.
This adjustment has to be done in a specific way, or else performance would degrade.





f	0.00	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
Top1	72.04	72.06	71.72	72.00	71.94	71.68	71.75	71.52	71.70	71.87	71.53
Top5	90.21	90.07	90.24	90.11	90.16	89.93	90.15	89.82	89.63	89.59	89.47

Tweaking the Weights

We first perform SVD on the $p \times n$ weight matrix W:

$$W = U\Sigma V^T = U_{p\times q}\Sigma_{q\times q}V^T_{n\times q} + Z,$$

with the singular values in Σ sorted from greatest to smallest and Z is the residual matrix. So

$$\frac{1}{n}WW^T = \frac{1}{n}U\Sigma^2U^T = HH^T + N$$

where

$$H = \frac{1}{\sqrt{n}}U_{p\times q}\Sigma_{q\times q}.$$

Here $U_{p\times q}$ consists of the top q left singular vectors and $\Sigma_{q\times q}$ consists of the top q singular values.

Tweaking the Weights

Provided that Z is noise, and that $\mathbb{E}W = B\Lambda\mathcal{V}^T$, then the left singular vectors of H_{JS} which is the JS correction of H will be a better estimator of B then $U_{p\times q}$.

Tweaking the Weights

To correct the weights W, we apply JS correction to the space spanned by the columns of H. Here we are not scaling any directions (yet). Given priors $v_1, \ldots, v_l$ (linearly independent) p-vectors, take A to be the matrix with the prior vectors as columns.
The shrinkage target is given by

$$M = A(A^T A)^{-1} A^T H$$

which is the projection of the columns of H to the space spanned by the priors.

Tweaking the Weights

The variance of the noise is estimated by

$$\nu^2 = \frac{\mathrm{Tr}(N)}{n_+ - q} \qquad \left(\text{or} \quad \frac{Tr(N)}{n_+ - (1 + \frac{n_+}{p})q} \right)$$

where n_+ is the number of nonzero singular values. Our shrinkage parameter is given by

$$C = I - \nu^2 J^{-1}, \;\; J = (H - M)^T (H - M).$$

Our JS correction of H is given by

$$H_{JS} = HC + M(I - C).$$

Tweaking the Weights

We take S^2 to be the largest q sample eigenvalues, i.e.

$$S^2 = H^T H = \frac{1}{n}\Sigma^2_{q\times q}.$$

We correct the singular values by taking

$$\Phi = S^2\Psi^2, \;\; \Psi^2 = I - \nu^2 S^{-2}$$

where S^{-2} is the inverse of S^2. So the q corrected singular values are given by

$$\Sigma_{JS,q\times q} = (n\Phi)^{1/2}.$$

Many other eigenvalue shrinkages are possible.

Tweaking the Weights

To correct the weights W, we again apply SVD to H_{JS} to get the left singular vectors β_{JS}.
We replace the top q left singular vectors of U in the SVD of W by β_{JS} to get a new matrix U_{JS}.
The JS corrected weights W_{JS} is given by

$$W_{JS} = U_{JS}\Sigma V^T$$

without singular value corrections. With singular value corrections, we have

$$W_{JS} = U_{JS}\Sigma_{JS} V^T.$$

Some Results

Taking $A = (e)$ which corresponds to taking the grand mean shrinkage, without singular value corrections, we have

q	0	noise	1	2	3	4	5	6	7	8	9	10	11
Top1	72.04	0.13	72.05	72.14	72.08	72.07	71.98	72.05	71.94	71.65	71.95	71.81	71.77
Top5	90.21	0.53	90.22	90.31	90.21	90.20	90.10	90.16	90.04	89.72	89.92	89.94	89.67

q	12	13	14	15	16	17	18	19	20	21	22	23	24
Top1	71.76	71.67	71.62	71.91	71.90	71.80	71.50	71.82	71.63	71.59	71.84	71.50	71.47
Top5	89.29	89.65	89.42	89.75	89.68	89.41	89.00	89.84	89.08	89.11	89.45	89.11	88.37

With singular value corrections, we have

q	0	1	2	3	4	5	6	7	8
Top1	72.04	72.05	72.11	72.07	72.07	71.98	72.06	71.95	71.66
Top5	90.21	90.22	90.32	90.20	90.23	90.07	90.15	90.03	89.81

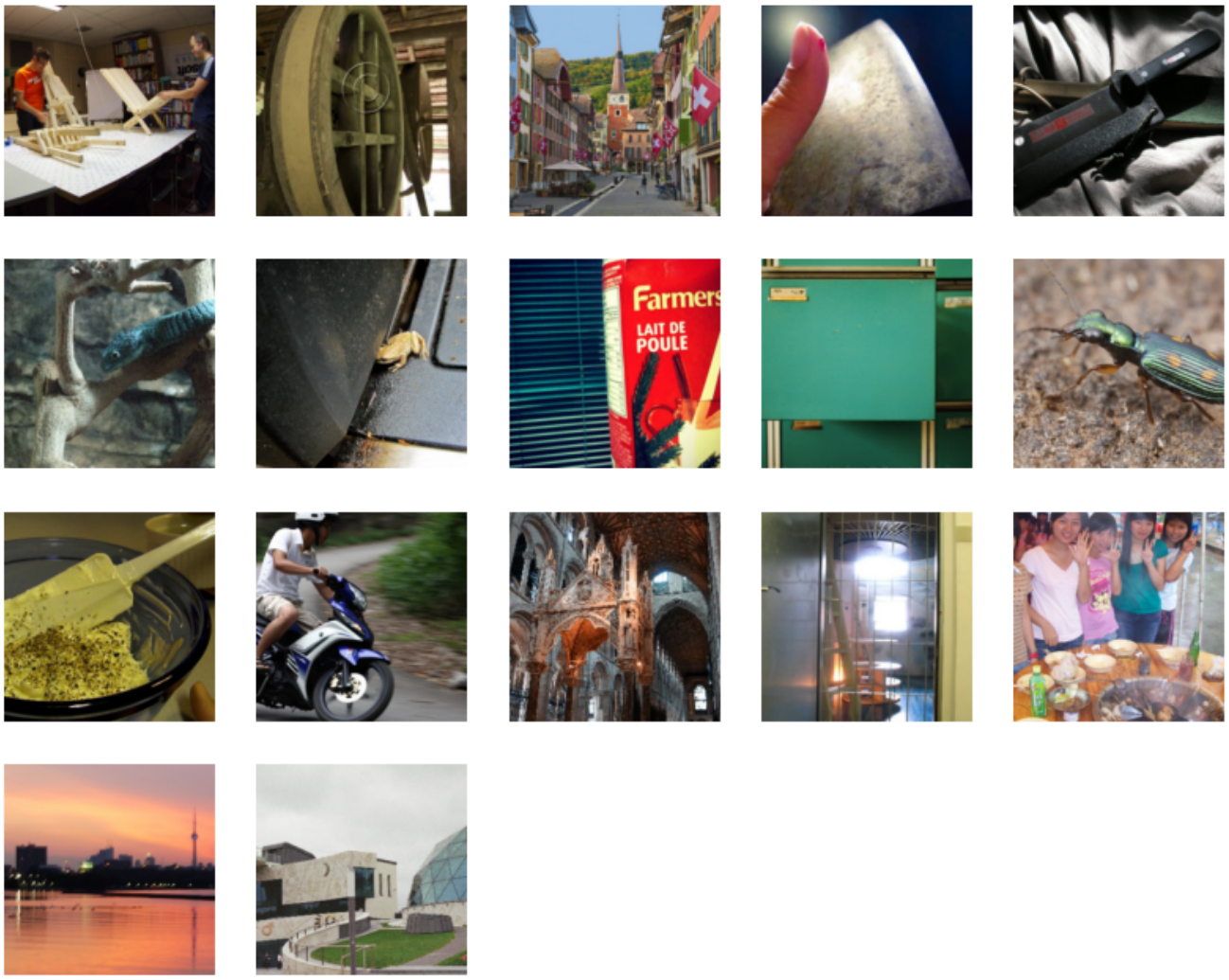
Some Results

Taking a closer look to the Top 1 prediction for $q = 1, 2, 3, 4$:

q	1	2	3	4
worse	0	17	11	12
better	1	27	15	15
net change	+1	+10	+4	+3

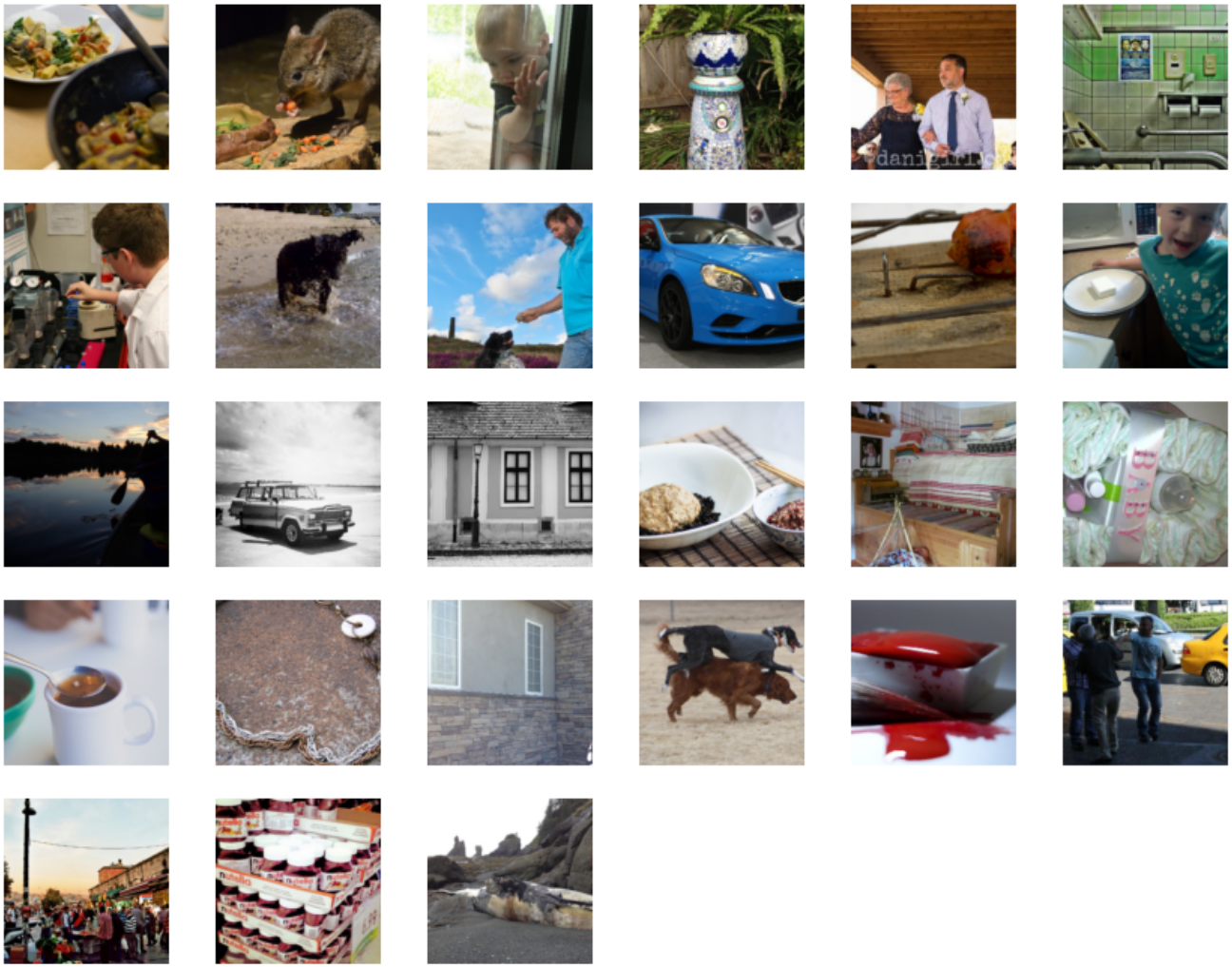
Some Results

Taking a closer look at $q = 2$, the following are the images that were originally correctly classified but now misclassified:



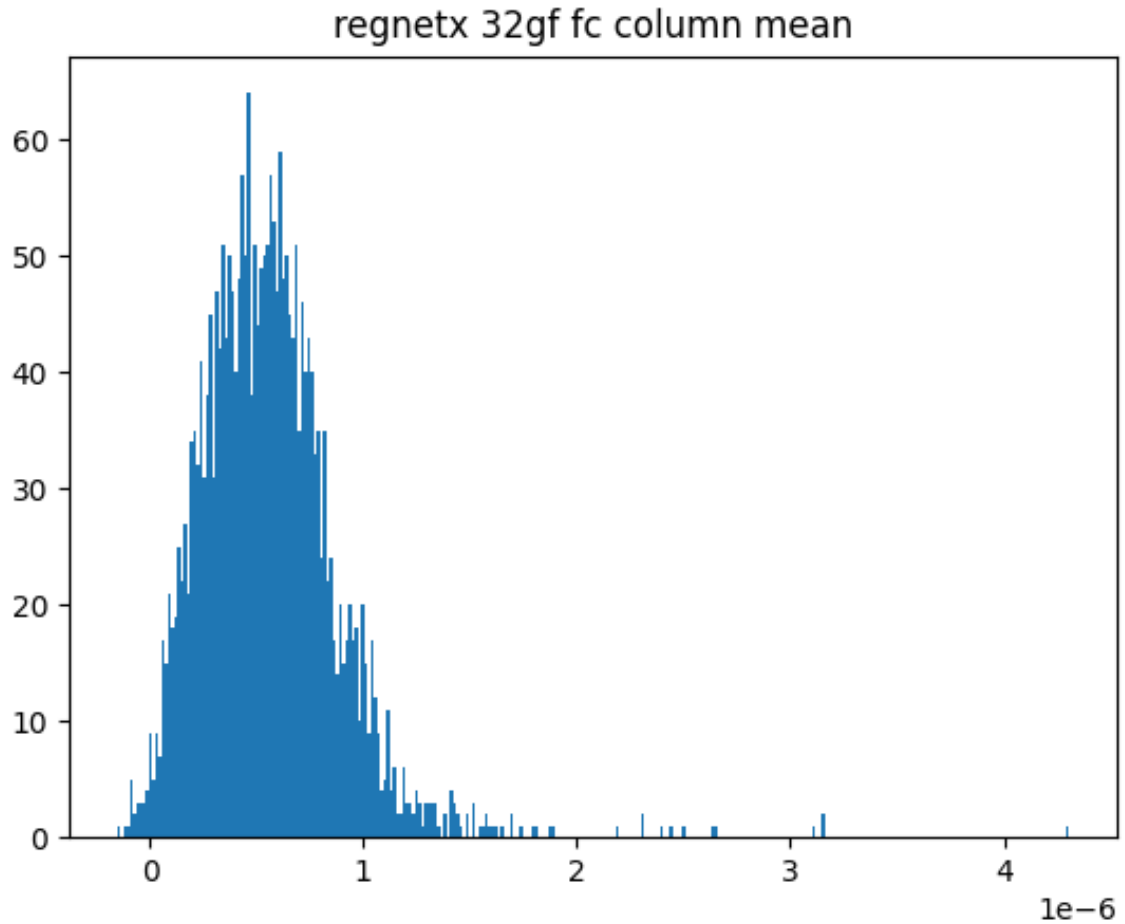
Some Results

The following are the images that were originally misclassified but now correctly classified:



Choice of A

To find better A, we explore the empirical properties of W.



Choice of A

Take β to be the left singular vectors of H, i.e. β is the matrix with normalized columns of H.
For z a unit p-vector, we look at the value

$$\sqrt{\mathrm{Tr}(\beta^T z z^T \beta)}$$

which is the length of the component of z in the column space of H.
We know that the length of the component of z in the column space of B is greater than of H for large p, so the greater the projection is, the more information it carries about B, the more justified we use such z inside A.

z, q	1	2	3	4
e	0.04694043225325887	0.0469690466771529	0.053850788947290174	0.05768234304901782
m	0.14742422943229017	0.15261738913083028	0.17775110605585345	0.18052871757787675
$\mu_{JS}(q)$	0.20988504238310507	0.224671248953613	0.26171971365607655	0.263107512010601

Results with different A and q

A, q	0	$(m, 1)$	$(m, 2)$	$(m, 3)$	$(m, 4)$	$(\mu_{JS}(1), 1)$	$(\mu_{JS}(2), 2)$	$(\mu_{JS}(3), 3)$	$(\mu_{JS}(4), 4)$
Top1	72.04	72.05	72.14	72.08	72.09	72.04	72.13	72.06	71.99
Top5	90.21	90.23	90.34	90.18	90.19	90.21	90.31	90.22	90.10

A, q	0	$(m, e), 1$	$(m, e), 2$	$(m, e), 3$	$(m, e), 4$	$(\mu_{JS}, e), 1$	$(\mu_{JS}, e), 2$	$(\mu_{JS}, e), 3$	$(\mu_{JS}, e), 4$
Top1	72.04	72.04	72.15	72.06	71.97	72.04	72.15	72.06	71.97
Top5	90.21	90.21	90.31	90.21	90.10	90.21	90.31	90.21	90.10

Future Work

1. Find better priors A to have a better shrinkage target
2. Find good weights to work on the weighted version of JS correction, emphasizing the more important directions
3. There should be correction to the complement left singular vectors since after the correction the vectors are not orthogonal anymore
4. Apply JS correction to other neural networks that have similar behavior

Future Work

5. Perform LW type shrinkage to WW^T/n towards sample correlation matrix and correct the top q singular vectors this way
6. Investigate the behaviors of the spiked eigenvectors during training, whether the self-regularization that happens during training is JS shrinkage, and whether this will give a hint about better targets

Future Work

7. Some layers of neural networks belong to another class of behavior, having a heavy tailed spectrum. Are there ways to correct these layers?