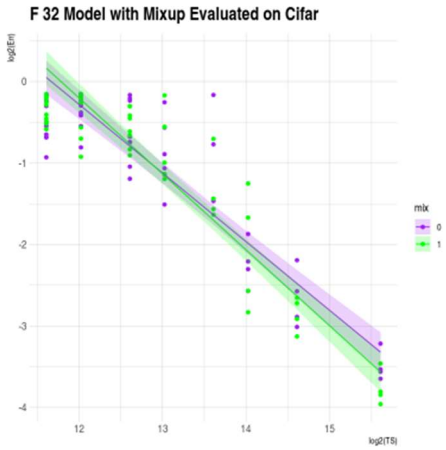
Mixup is a data augmentation technique, by mixing multiple images in a batch by a linear combination and having the model predict both images present. This has the potential to change the power law slope, since the number of combinations of images grows asymptotically faster than the number of images.

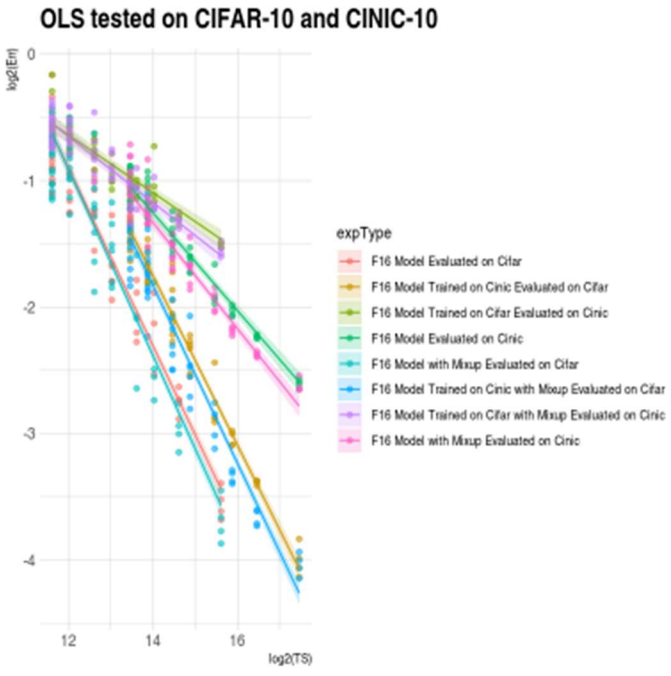
Mixup's effect on the power law was tested across a variety of groups training and testing on Cifar and Cinic, and 16 and 32 Filter Models.



For 32 filter models it appears that the error is not related to the training sizes for small sizes when training on Cifar (n > 4). This part of the dataset may not be in the power law region. Removing these points results in too small of a sample size for a good regression, and removing all the Cifar points causes issues with the interaction coefficients since 1 combination is missing. 32 Filter model data was excluded from the dataset for the regressions.

This is a graph is an OLS regression on the data, to estimate the effect of mixup on the slope and intercept of the trend, while blocking all of the other variables.

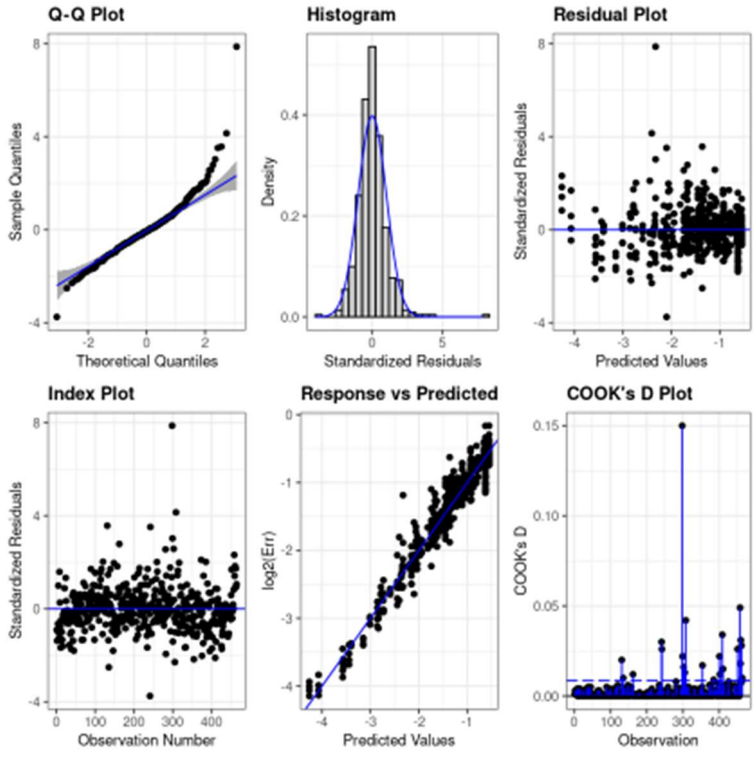
The regression has interaction terms between log trainsize and mixup, train and test dataset (Cifar-10 or Cinic-10). This results in each group having its own intercept and slope, except they all share the term of mixup's effect on those values.



Below is the table of regression results sorted by the t value, so insignificant params will end up at the bottom. The intercept represents the log error at a dataset size of 3125. The row I(log2(TS) - lXMin):mix1 is the param of how much mixup changes the power law slope compared to the corresponding experiment without mixup. The value is -0.0325, and is statistically significant.

There are n trials when testing on 1/nth of the dataset, except for the larger training sizes where the same data is repeated with different model initializations for 4 trials. (ex 2 trials on each half for n = 2).

	Estimate	2.5 %	97.5 %	Std. Error	t value	Pr(>|t|)
I(log2(TS) - lXMin)	-0.7038168587	-0.734686756	-0.672946962	0.01570825	-44.805558986	9.226088e-169
I(log2(TS) - lXMin):CinicEv1	0.4752733303	0.434779454	0.515767207	0.02060544	23.065427314	1.764288e-78
(Intercept)	-0.6266443677	-0.698702683	-0.554586052	0.03666711	-17.090093852	6.395764e-51
CinicTr1	0.4737666500	0.347328802	0.600204498	0.06433831	7.363678247	8.485329e-13
I(log2(TS) - lXMin):CinicTr1:CinicEv1	-0.1921994819	-0.249467150	-0.134931814	0.02914084	-6.595536467	1.181194e-10
CinicTr1:CinicEv1	-0.2542266223	-0.433036741	-0.075416503	0.09098812	-2.794063985	5.425290e-03
I(log2(TS) - lXMin):mix1	-0.0325890841	-0.055660583	-0.009517585	0.01174001	-2.775899573	5.732395e-03
CinicEv1	0.0769843330	-0.011277409	0.165246075	0.04491228	1.714104390	8.719208e-02
I(log2(TS) - lXMin):CinicTr1	0.0337692580	-0.006725098	0.074263614	0.02060569	1.638831978	1.019410e-01
mix1	0.0001127254	-0.071924300	0.072149751	0.03665628	0.003075201	9.975477e-01



The minimum offsets on log trainsize and log params are to make the intercept represent the smallest model on smallest train set, instead of a 1 element trainset, and 1 param model.

The QQ plot shows the distribution of errors compared to a normal distribution, the nonlinearity shows the assumptions of OLS may not be true. On the histogram, the errors appear to have a skewed distribution with a heavier right tail than left tail.

OLS and IRWLS were tested on 1000 trials of synthetic distributions of residuals that are similar to this observed distribution of residuals. Graphs for these tests on the next 2 pages. The measured standard error and the estimated standard error are compared to see how accurate the estimator is, and if it is underestimating the error. The Green line is the ground truth value, and red is the mean estimate.

For OLS the estimate is unbiased, but the standard error of the estimate may be underestimated, and/or higher than the IRWLS estimator.

The same test with IRWLS accurately estimated the coefficient value and its standard error.

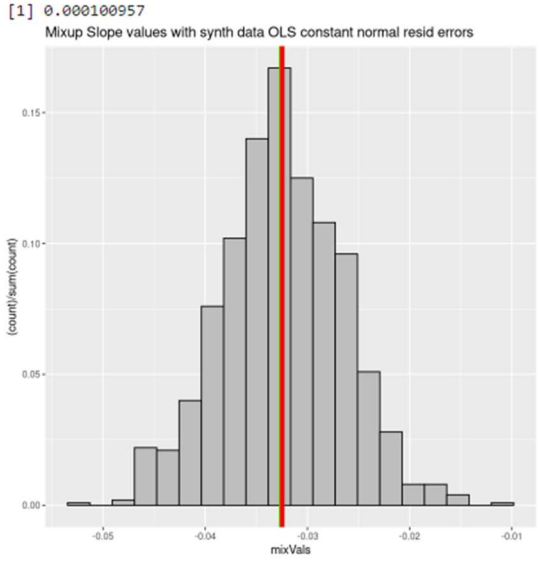
This shows that even for unbalanced designs and non-normal heteroscedastic distributions of residuals, IRWLS remains a reliable method for estimating the slope.

The problem for OLS of underestimating of the standard error is mitigated by having 4 samples of the larger training size instead of 1 or 2, but IRWLS is still the better estimator, as it has a lower standard error for this data.

N samples for each size

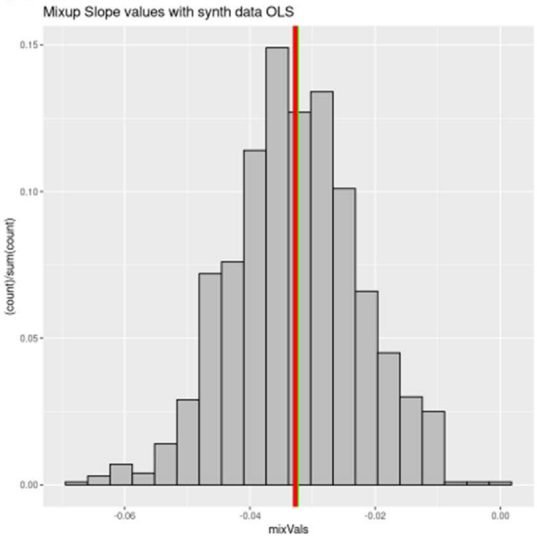
OLS Normal Residuals

```
[1] "measured se"
[1] 0.005842586
[1] "mean estimated se"
[1] 0.003329001
[1] "sd of estimated se"
[1] 0.0001554026
[1] "estimate bias"
[1] 0.000100957
```



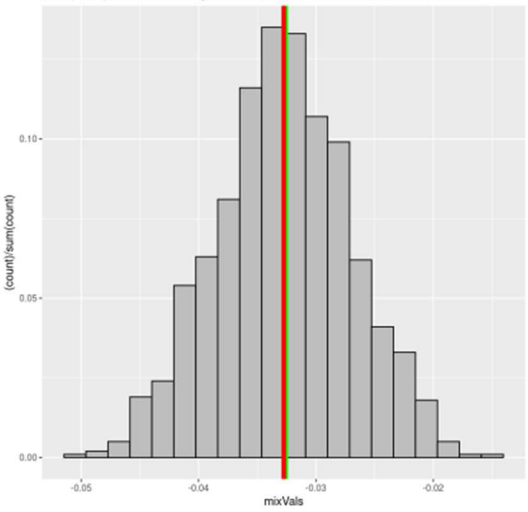
Non-Normal Heteroscedastic Residuals

```
[1] "measured se"
[1] 0.01017801
[1] "mean estimated se"
[1] 0.01141031
[1] "sd of estimated se"
[1] 0.0005299589
[1] "estimate bias"
[1] -0.0002071115
```



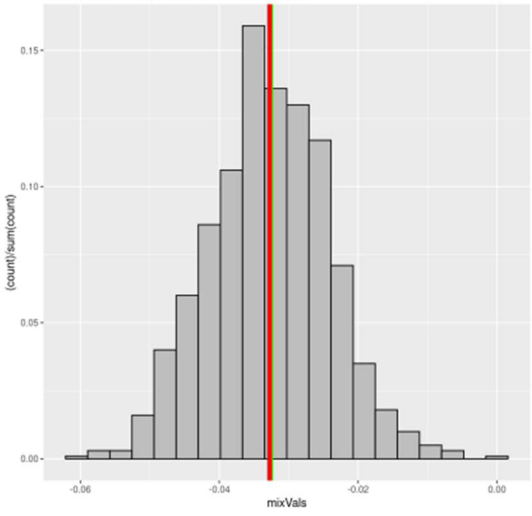
IRWLS Normal Residuals

```
[1] "measured se"
[1] 0.005688416
[1] "mean estimated se"
[1] 0.00490566
[1] "sd of estimated se"
[1] 0.00161533
[1] "estimate bias"
[1] -0.0001723415
```



Non-Normal Heteroscedastic Residuals

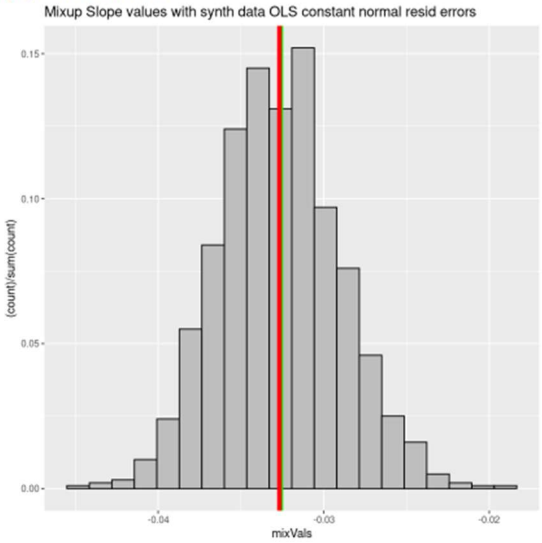
```
[1] "measured se"
[1] 0.008597616
[1] "mean estimated se"
[1] 0.008686358
[1] "sd of estimated se"
[1] 0.001494603
[1] "estimate bias"
[1] -0.0001830715
```



Minimum 4 trials for large training size (small N)

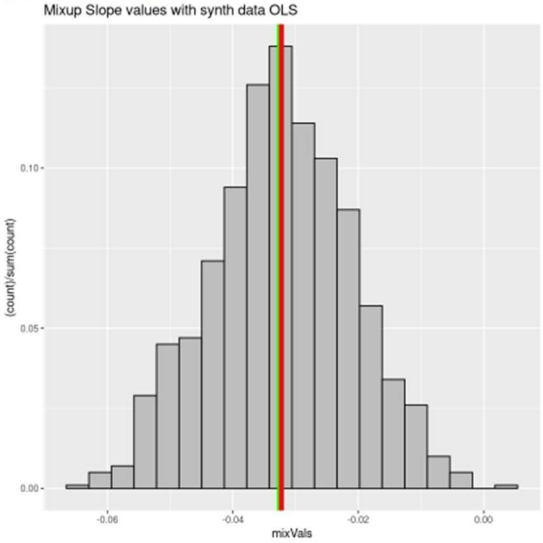
OLS Normal Residuals

```
[1] "measured se"
[1] 0.003636302
[1] "mean estimated se"
[1] 0.00320905
[1] "sd of estimated se"
[1] 0.000123274
[1] "estimate bias"
[1] -0.00008287347
```



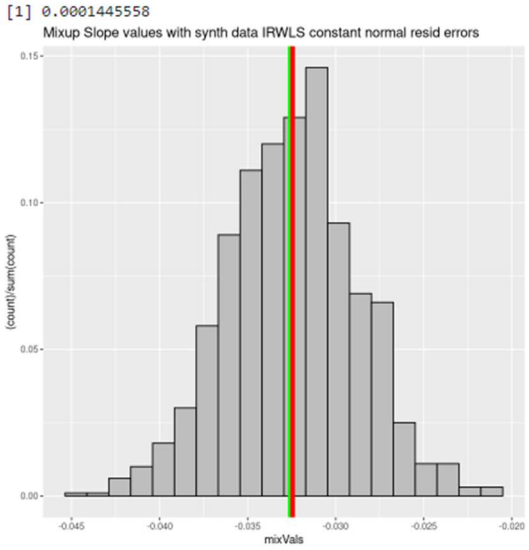
Non-Normal Heteroscedastic Residuals

```
[1] "measured se"
[1] 0.01120622
[1] "mean estimated se"
[1] 0.01133936
[1] "sd of estimated se"
[1] 0.000501306
[1] "estimate bias"
[1] 0.0003499316
```



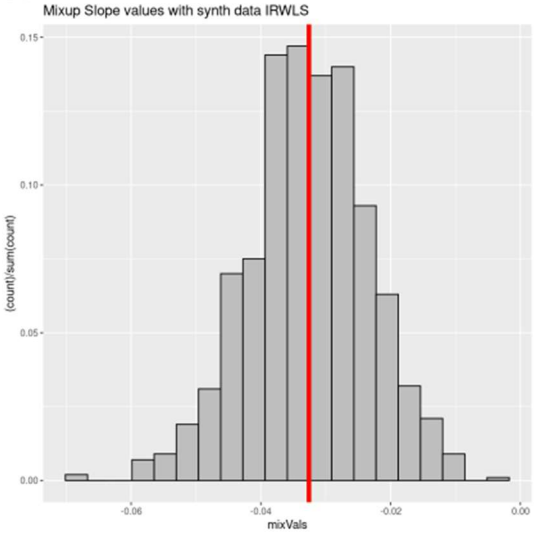
IRWLS Normal Residuals

```
[1] "measured se"
[1] 0.003710258
[1] "mean estimated se"
[1] 0.003804089
[1] "sd of estimated se"
[1] 0.0004540739
[1] "estimate bias"
[1] 0.0001445558
```



Non-Normal Heteroscedastic Residuals

```
[1] "measured se"
[1] 0.009173447
[1] "mean estimated se"
[1] 0.009519041
[1] "sd of estimated se"
[1] 0.000964091
[1] "estimate bias"
[1] -0.0000006333971
```

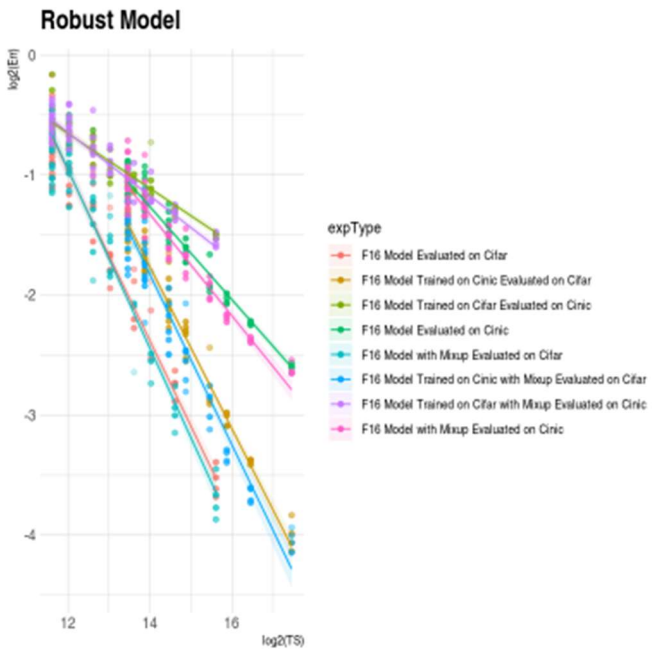


The studentized Breusch-Pagan test is for Heteroscedasticity of residuals. A result of a very small p value 2.3*10^-8, rejects the null hypothesis of constant variance of residuals, invalidating the results of ordinary linear regression. This may be a result of the unbalanced design.

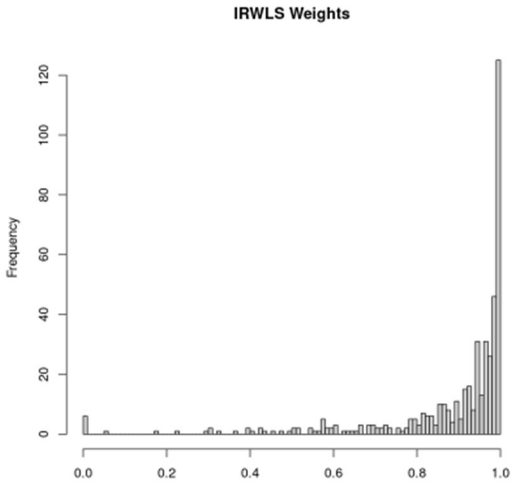
Robust Regression using Iteratively reweighted
least squares, detects and reduces the influence of outliers, deals with Heteroscedasticity and non-normally distributed residuals. This comes at a tradeoff of when the assumptions of OLS are true, the variance of OLS estimations are less than the variance of IRWLS estimations.

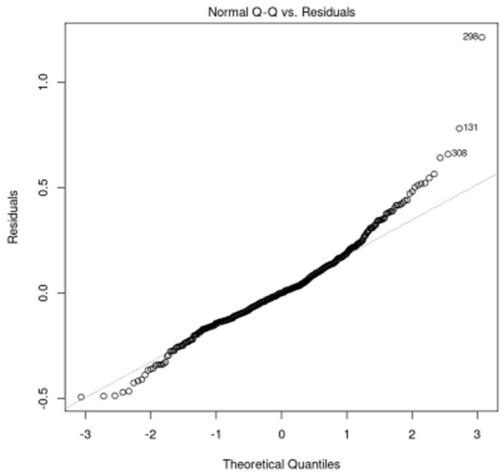
This model was with data weighted by inverse of the number of trials (ex, 1/8th of trainset would have 8 trials, and each datapoint is weighted by 1/8). Points are shaded by their IRWLS weight. The interaction coeficient of mixup and trainsize is -0.0356.

IRWLS does not always converge to the same minimum, though the difference is small, and sometimes fails to converge.

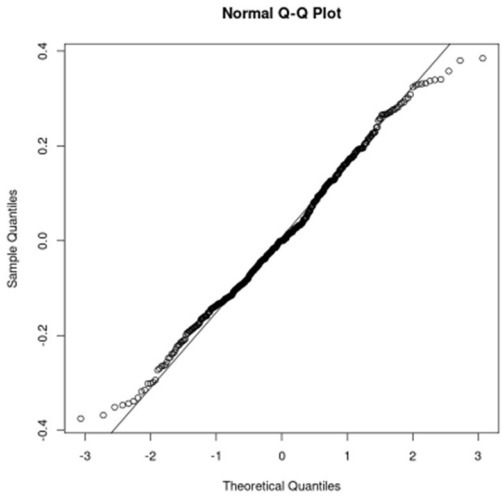


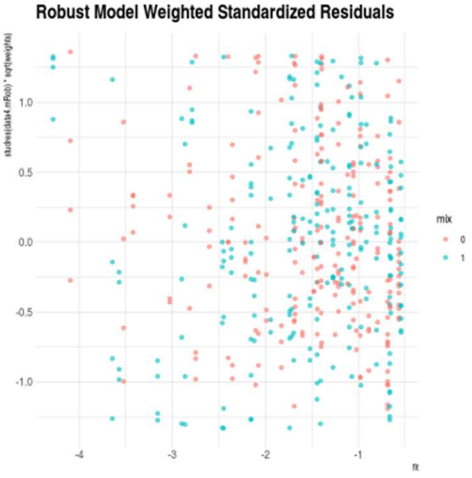
	Estimate	2.5 %	97.5 %	Std. Error	t value	Pr(>|t|)
I(log2(TS) - IXMin)	-0.70940780	-0.74745501	-0.67136059	0.01936045	-36.642117	1.058915e-137
I(log2(TS) - IXMin):CinicEv1	0.48100068	0.44278339	0.51921798	0.01944700	24.733932	3.675491e-86
(Intercept)	-0.68695113	-0.78517545	-0.58872681	0.04998177	-13.744035	3.469150e-36
I(log2(TS) - IXMin):CinicTr1:CinicEv1	-0.19440632	-0.25157611	-0.13723654	0.02909103	-6.682689	6.891392e-11
CinicTr1	0.52892822	0.36684935	0.69100709	0.08247437	6.413244	3.580817e-10
CinicTr1:CinicEv1	-0.30351683	-0.47727982	-0.12975383	0.08841987	-3.432677	6.524032e-04
I(log2(TS) - IXMin):mix1	-0.03568192	-0.06085144	-0.01051240	0.01280759	-2.785997	5.559783e-03
CinicEv1	0.12077372	0.02844494	0.21310249	0.04698180	2.570649	1.046844e-02
I(log2(TS) - IXMin):CinicTr1	0.03596145	-0.01830742	0.09023032	0.02761489	1.302248	1.934917e-01
mix1	0.02256919	-0.04786914	0.09300753	0.03584278	0.629672	5.292260e-01





The histogram of the weights shows that most points are near 1, and only a few are reduced, or eliminated (5 points with 0 weight). The weights that are not 1 or 0, have a median of 0.95 and mean of 0.88.

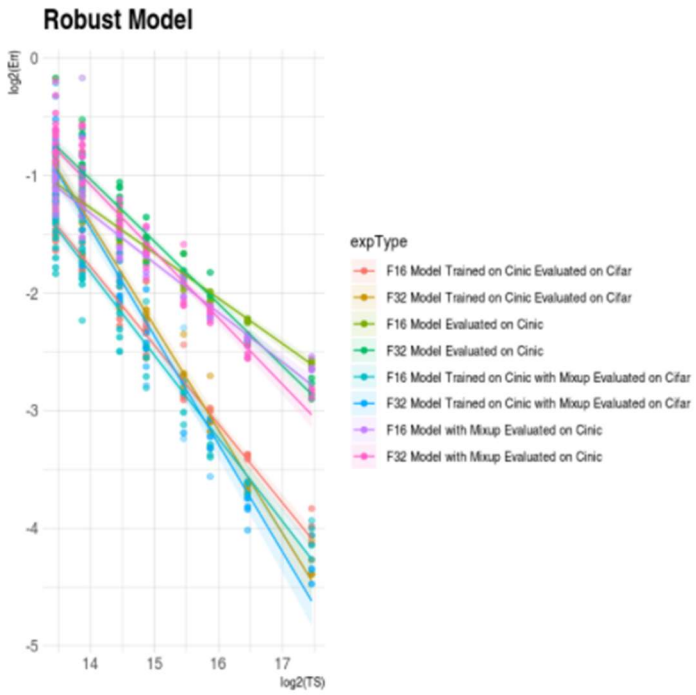




When making a QQ plot for the weighted residual distribution (lower plot), it looks normal except with some deviation on the tails, while the QQ plot for unweighted residuals looks similar to the OLS QQ plot and is non-normal. This shows how the weights adjust for non-normal distributions.

The residual plot above has the value of the residual and shows the weight by the transparency of the point. The weighted residual plot on the bottom shows the value of the weighted residual. This shows how the weights reduce or eliminate outlier points, and adjust for non-constant variance.

Another way to split the data is to remove training on Cifar and keep both F16 and F32 models. The estimate for mixup's effect on the slope is -0.0367, similar to above. Though it is not an independent estimate since half the data is shared.



The larger model has a higher error intercept, and increased magnitude power law slope. (note that for the Cinic Ev group the intercept and slope are the sum of both coefficients with and without the Cinic Ev term)

	Estimate	2.5 %	97.5 %	Std. Error	t value	Pr(>|t|)
(Intercept)	-1.41007471	-1.477460329	-1.3426890856	0.03428940	-41.1227638	3.455292e-155
I(log2(TS) - IXMin)	-0.67037254	-0.703465297	-0.6372797808	0.01683936	-39.8098612	3.548359e-150
I(log2(TS) - IXMin):CinicEv1	0.28466925	0.245722998	0.3236155049	0.01981793	14.3642279	7.753004e-39
CinicEv1	0.34683548	0.272514289	0.4211566702	0.03781858	9.1710333	1.642603e-18
I(log2(param) - IPMin)	0.24572172	0.182377070	0.3090663682	0.03223313	7.6232651	1.458763e-13
I(log2(TS) - IXMin):I(log2(param) - IPMin)	-0.10601174	-0.139287510	-0.0727359602	0.01693249	-6.2608482	8.880564e-10
CinicEv1:I(log2(param) - IPMin)	-0.08516369	-0.157240018	-0.0130873620	0.03667628	-2.3220375	2.067245e-02
I(log2(TS) - IXMin):mix1	-0.03676439	-0.073424919	-0.0001038599	0.01865483	-1.9707704	4.935720e-02
I(log2(TS) - IXMin):CinicEv1:I(log2(param) - IPMin)	0.03391547	-0.003345965	0.0711769103	0.01896061	1.7887337	7.432432e-02
mix1	-0.02723540	-0.092160142	0.0376893440	0.03303717	-0.8243866	4.101528e-01

It appears that Cifar is of insufficent size to scale the model and look at the power law since it can only be split a few ways before the trend is in the low data region instead of the power law region. Cinic is the better dataset to train on for scaling studies, though transfer learning from training on Cinic to the easier Cifar test set is still interesting. Cifar may be useful for studies on the transition of the low data region to the power law region, which occurs right in the middle of the dataset.

This shows that for small datasets of CIFAR formatted images, mixup increases the magnitude of the power law slope, accelerating the rate the model learns from data.

This process of multiple regression lets each experiment group have its own power law intercept and slope, but they all share the same parameters of interest, in this case mixup's change to the intercept and slope. This allows data from distinct experiments be used to estimate a single parameter with greater accuracy.

With changes that are expected to not influence the power law slope, something similar could be done by making an interaction of all the categorical factors, and not the training size. This would make every experiment group have its own intercept, but share the slope improving the accuracy of its estimate.