

ZACHARY CHEN-WISHART 30/10/2019

SUPER BIAS UPDATE

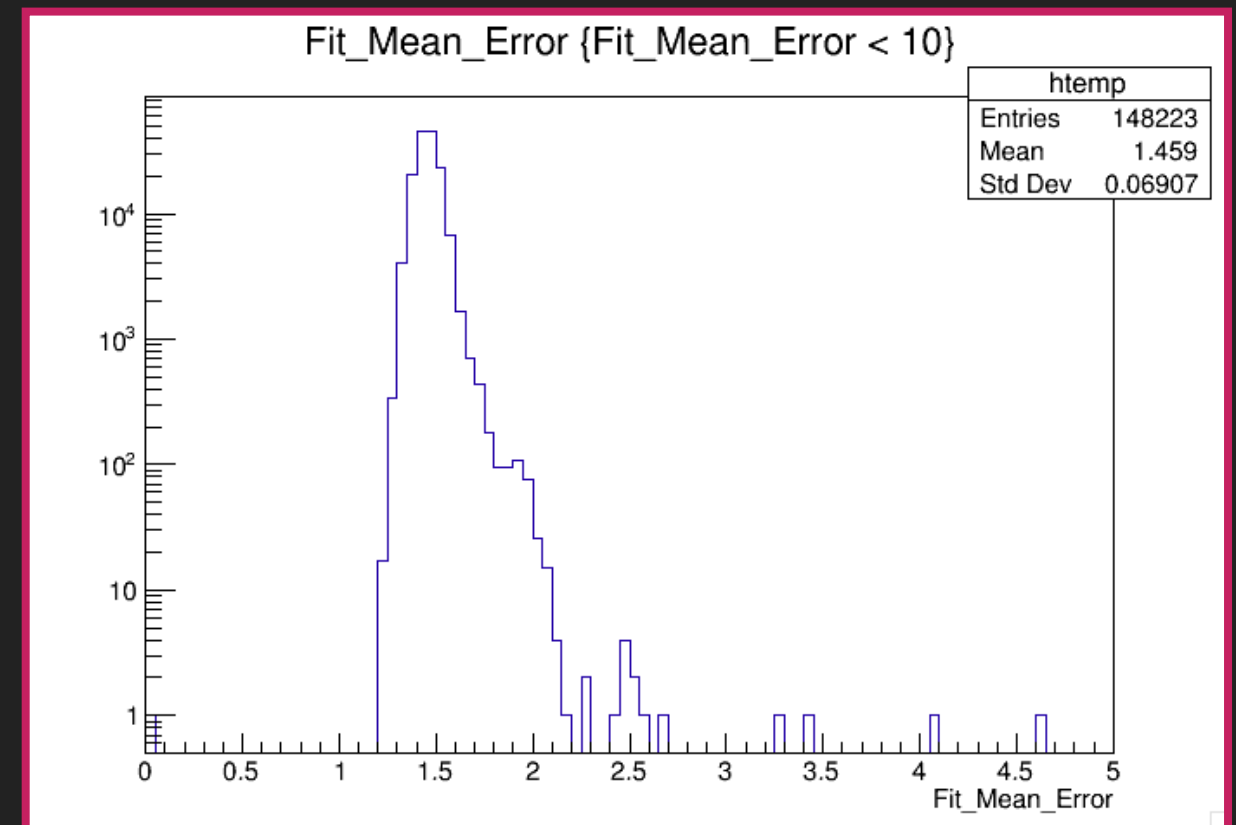
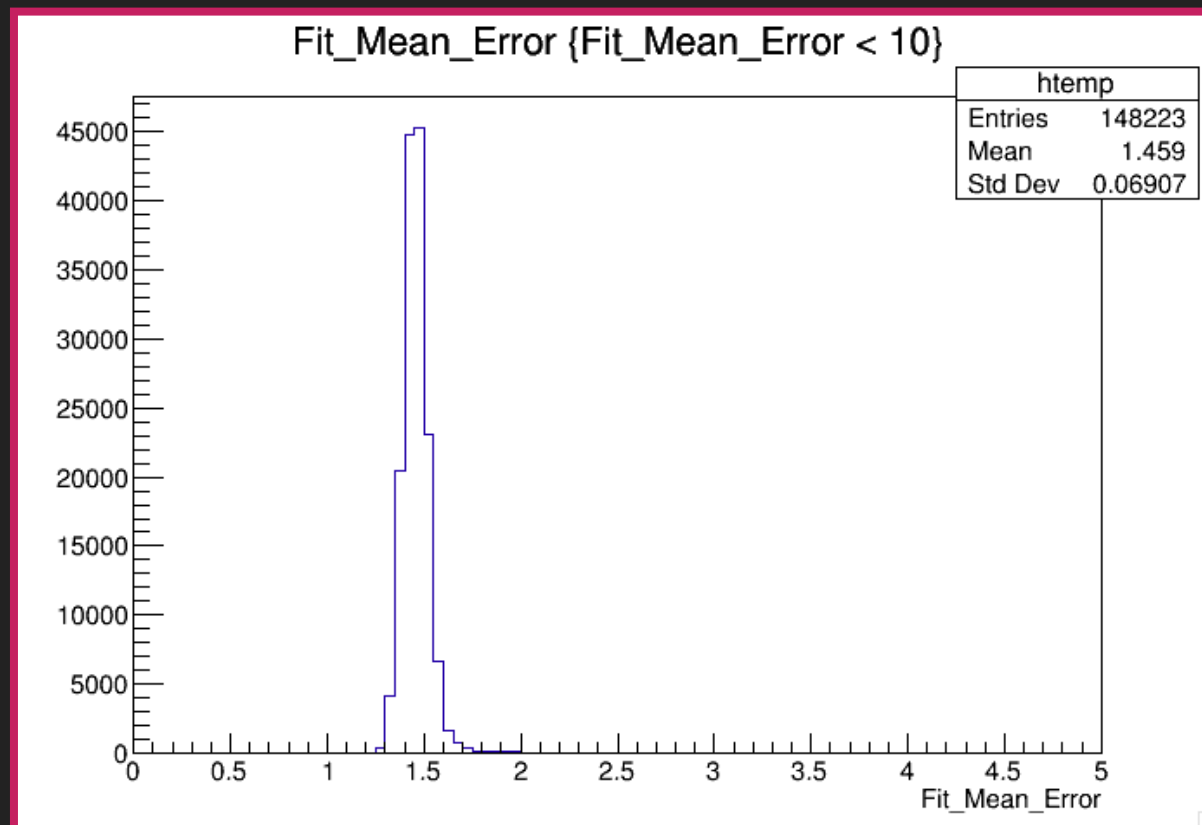
LIGHT SUM SQUARE - SUPER BIAS FRAMES

- ▶ I have written version 2 of my Hack_SparkKilledApp which now tracks all pixels from all Bias Frames instead of just pixels within boxes
- ▶ I have been working on and optimising (from 40 days run time to 5 mins) a macro `make_full_mask.cpp` ->

It does this by fitting for each pixel over 200 runs (800 bias frames) producing a super bias frames and outputting relevant info to look into the quality of the frame and stability of the pixels

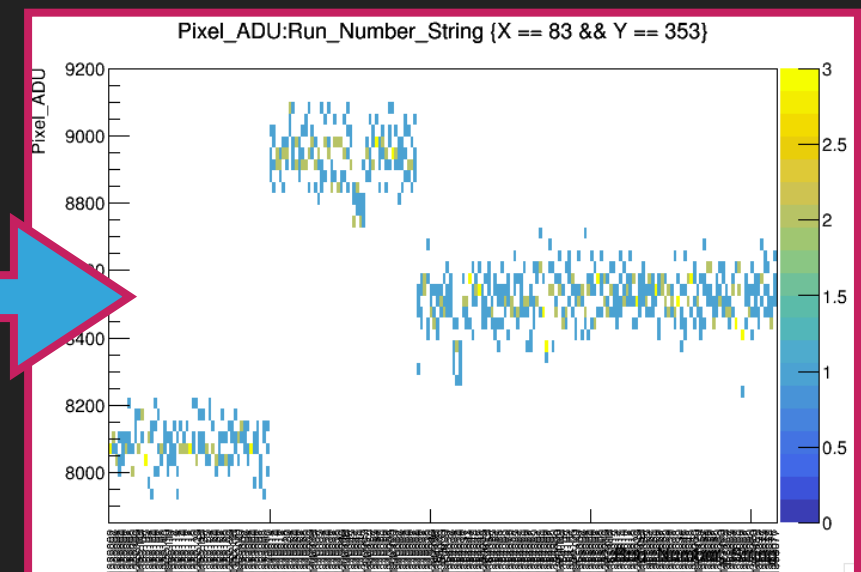
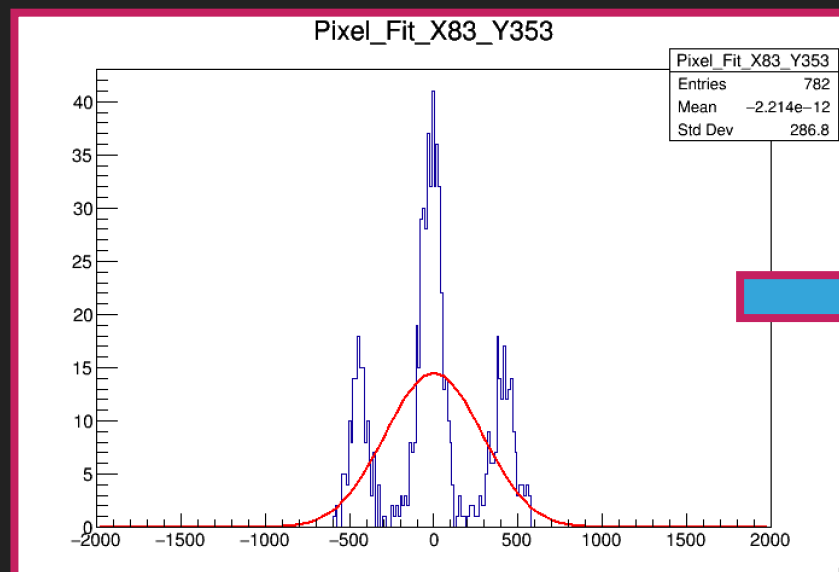
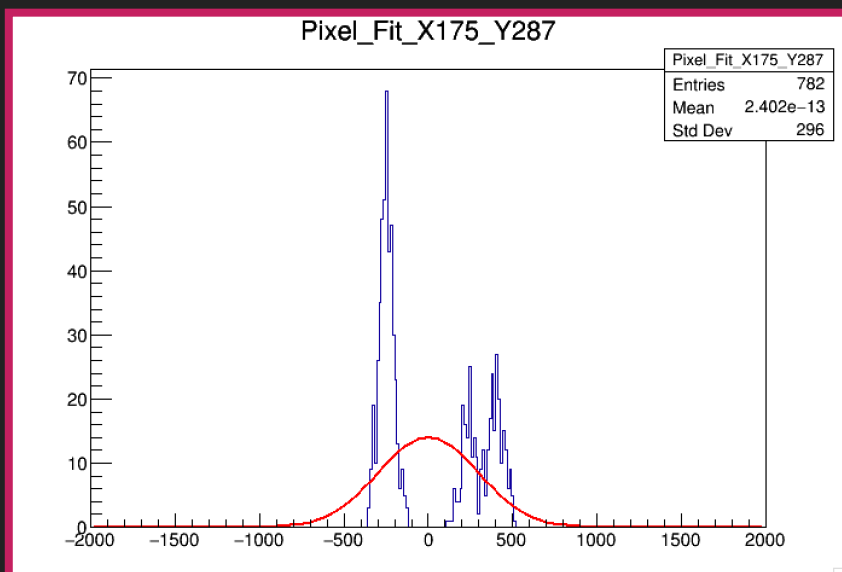
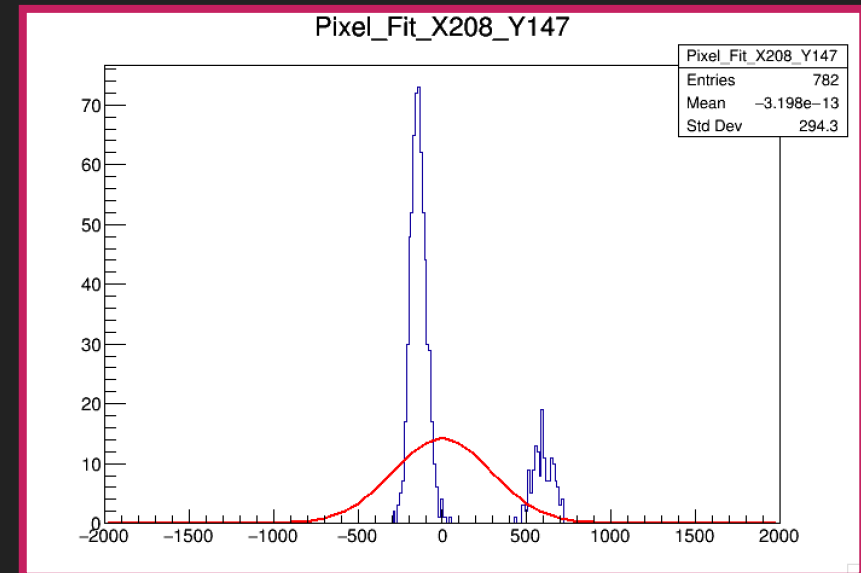
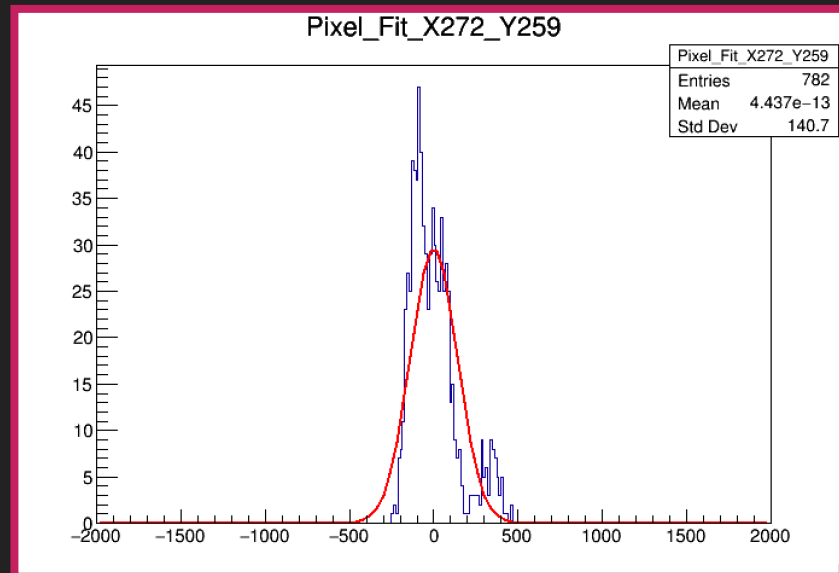
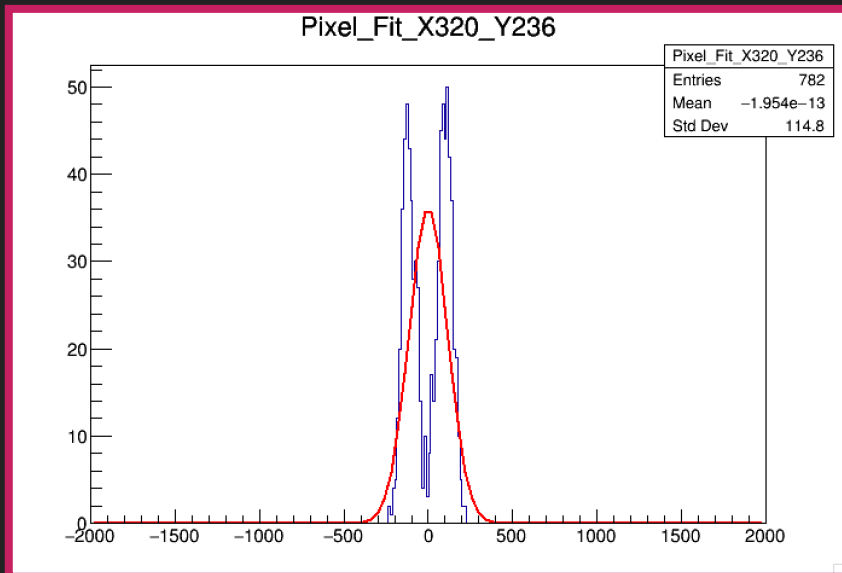
LIGHT SUM SQUARE – SUPER BIAS FRAMES

- ▶ We can see we have a mean fit error about 1.46 ADU per pixel which should provide us with a vastly superior error on the gain measurement due to the bias
- ▶ We still have some pixels with larger errors which I've spent some time looking into:
 - ▶ 1. Individual pixel pedestal jumping - Estimation of 20-50 pixels jumping 100 -1000 ADU
 - ▶ 2. Column pedestal jumping - Estimation 10 columns jumping 50 - 100 ADU



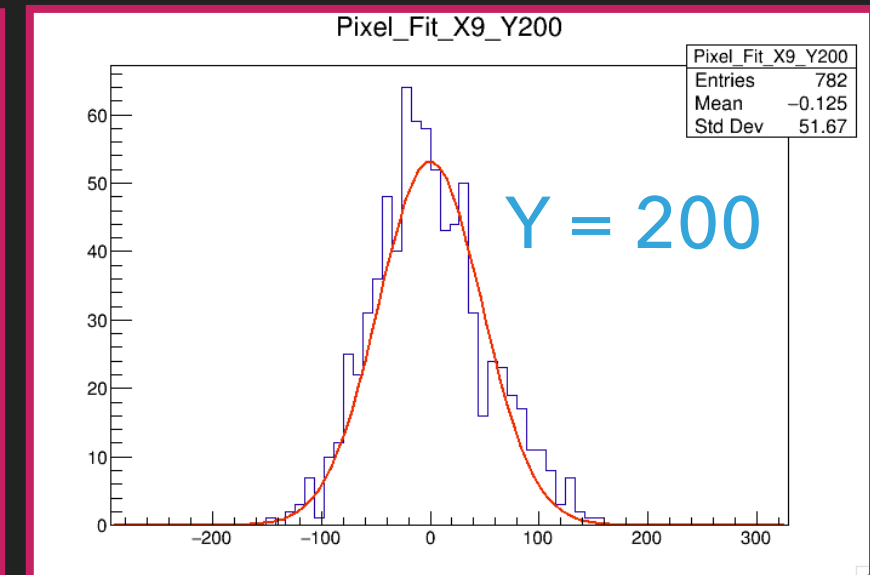
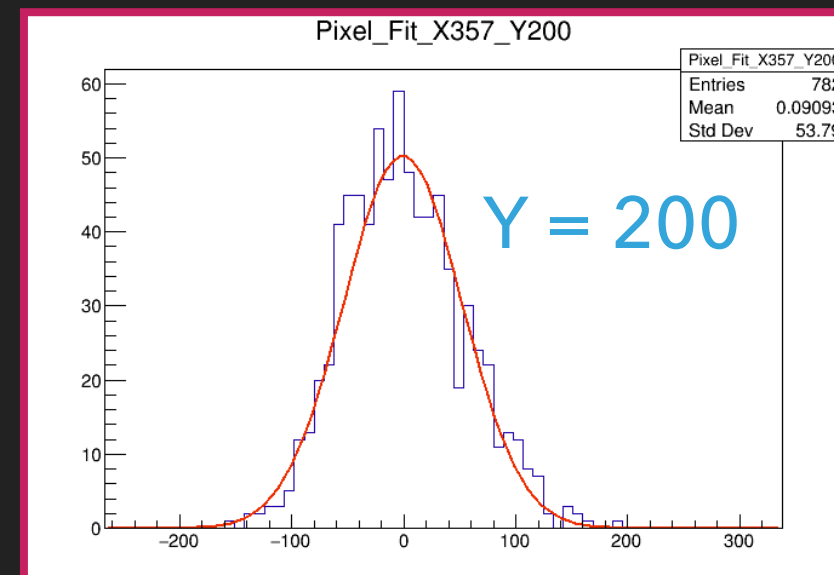
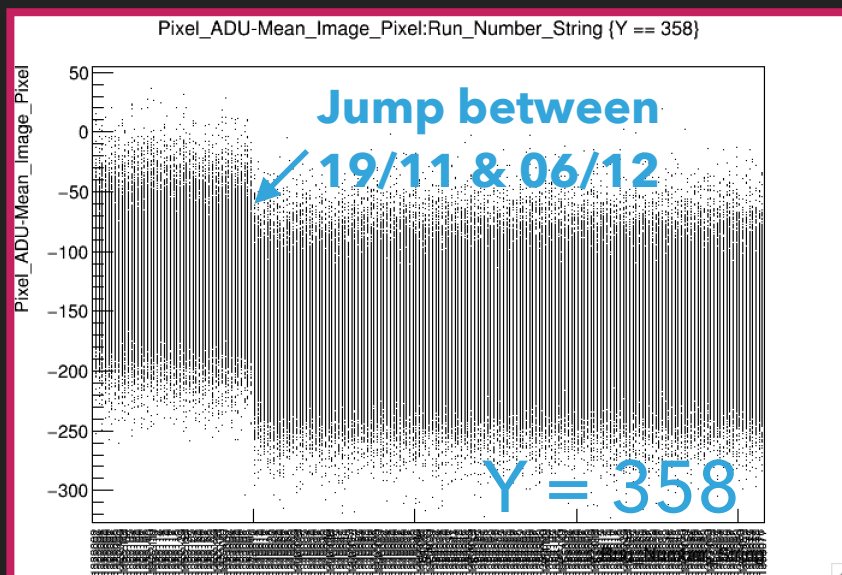
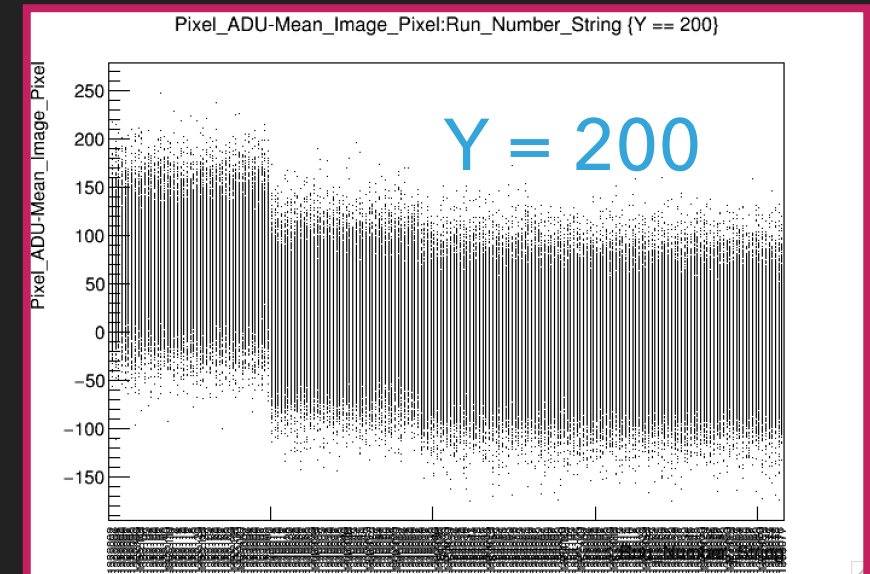
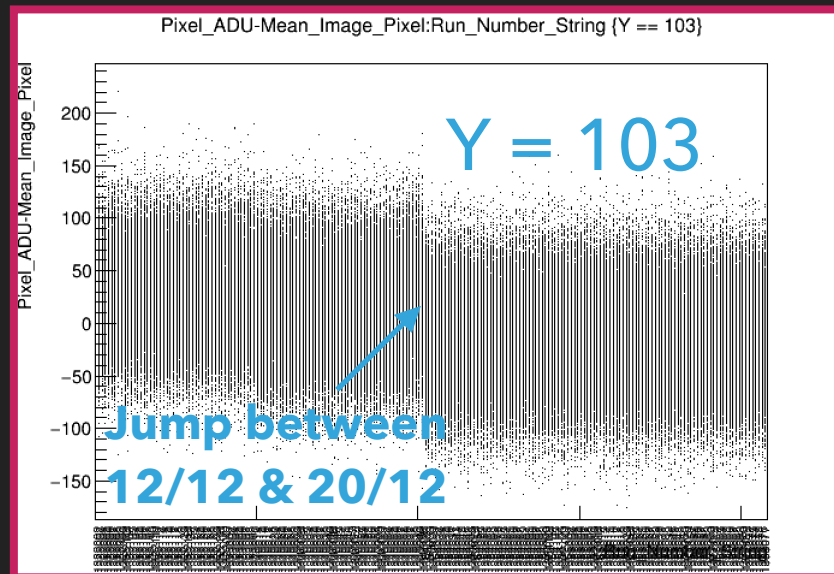
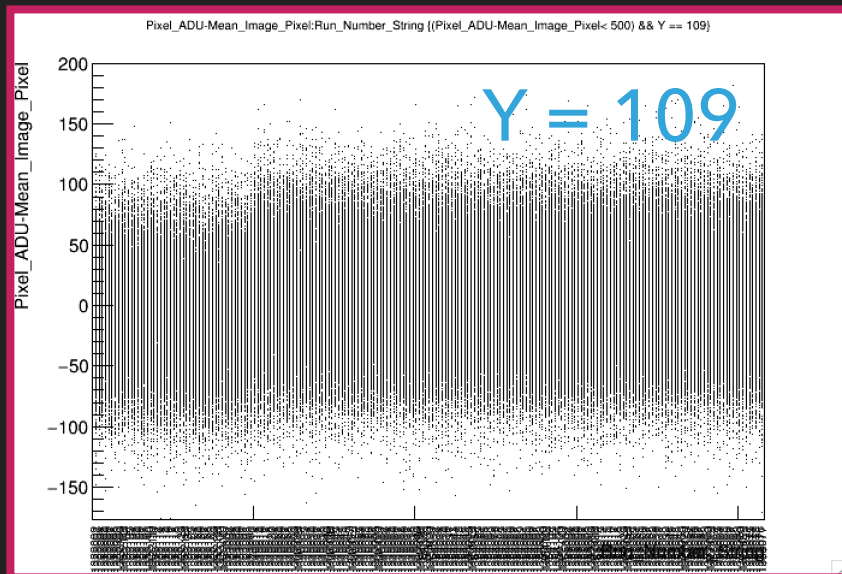
LIGHT SUM SQUARE - INDIVIDUAL PIXEL PEDESTAL JUMP

- ▶ These pixels changed pedestal over the course of the data range (Nov-Dec 2018) -> Producing large fit mean error -> These can be masked easily however cause problems in light gain measurement



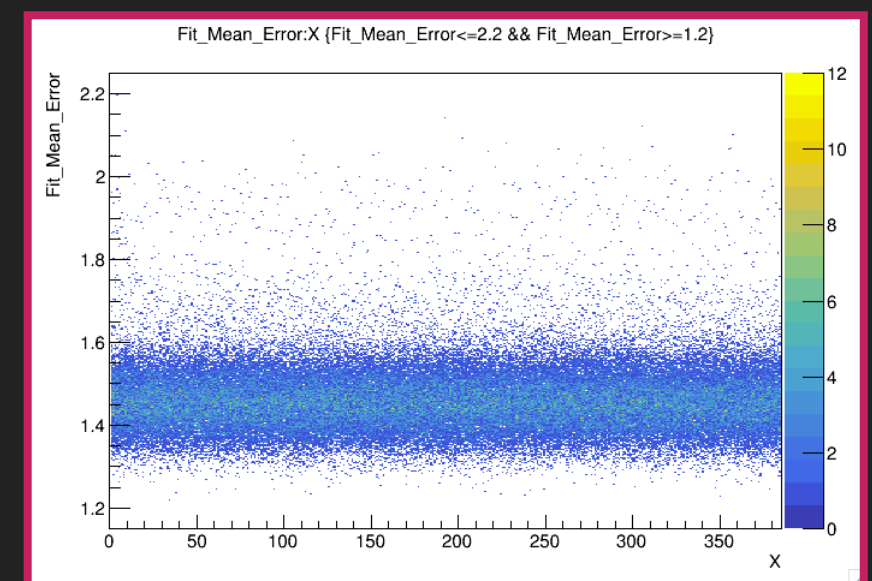
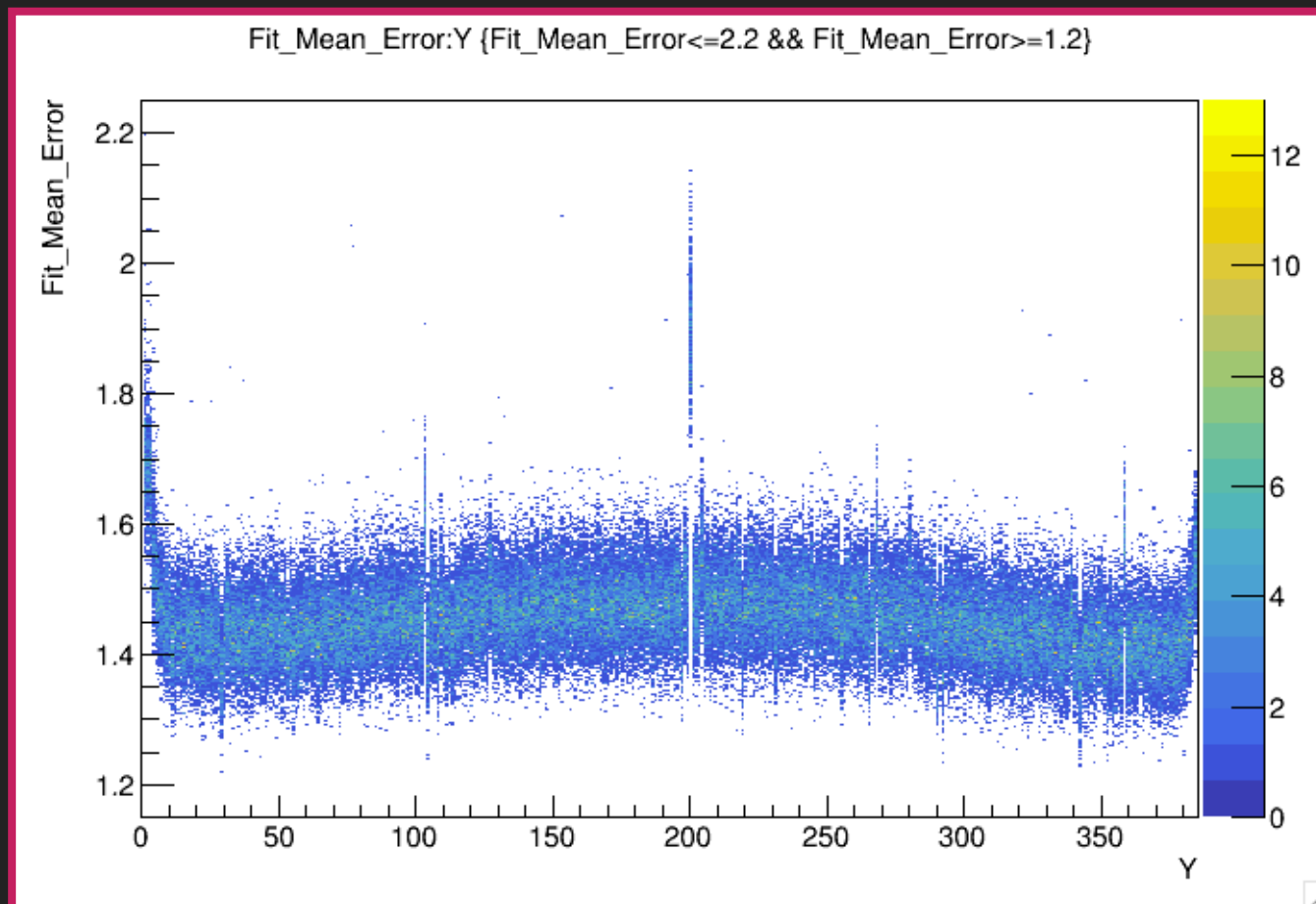
LIGHT SUM SQUARE - COLUMN PIXEL PEDESTAL JUMP

- ▶ The next to leading order subset of pixels has a strong correlation in y
- ▶ Column pedestal jumping -> more pixels effected but less error per pixel
- ▶ Hit large number between Nov and Dec and smaller number mid Dec



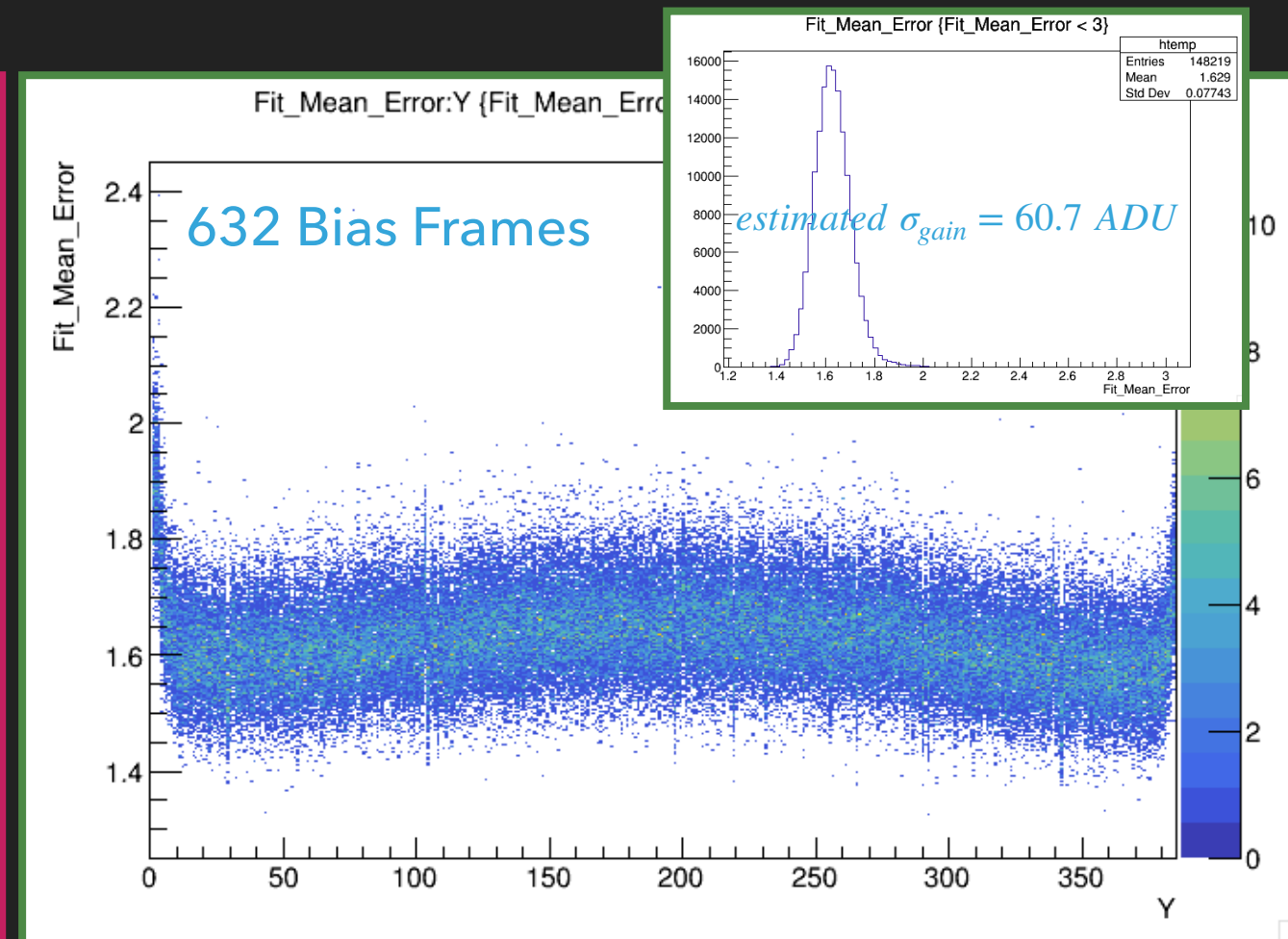
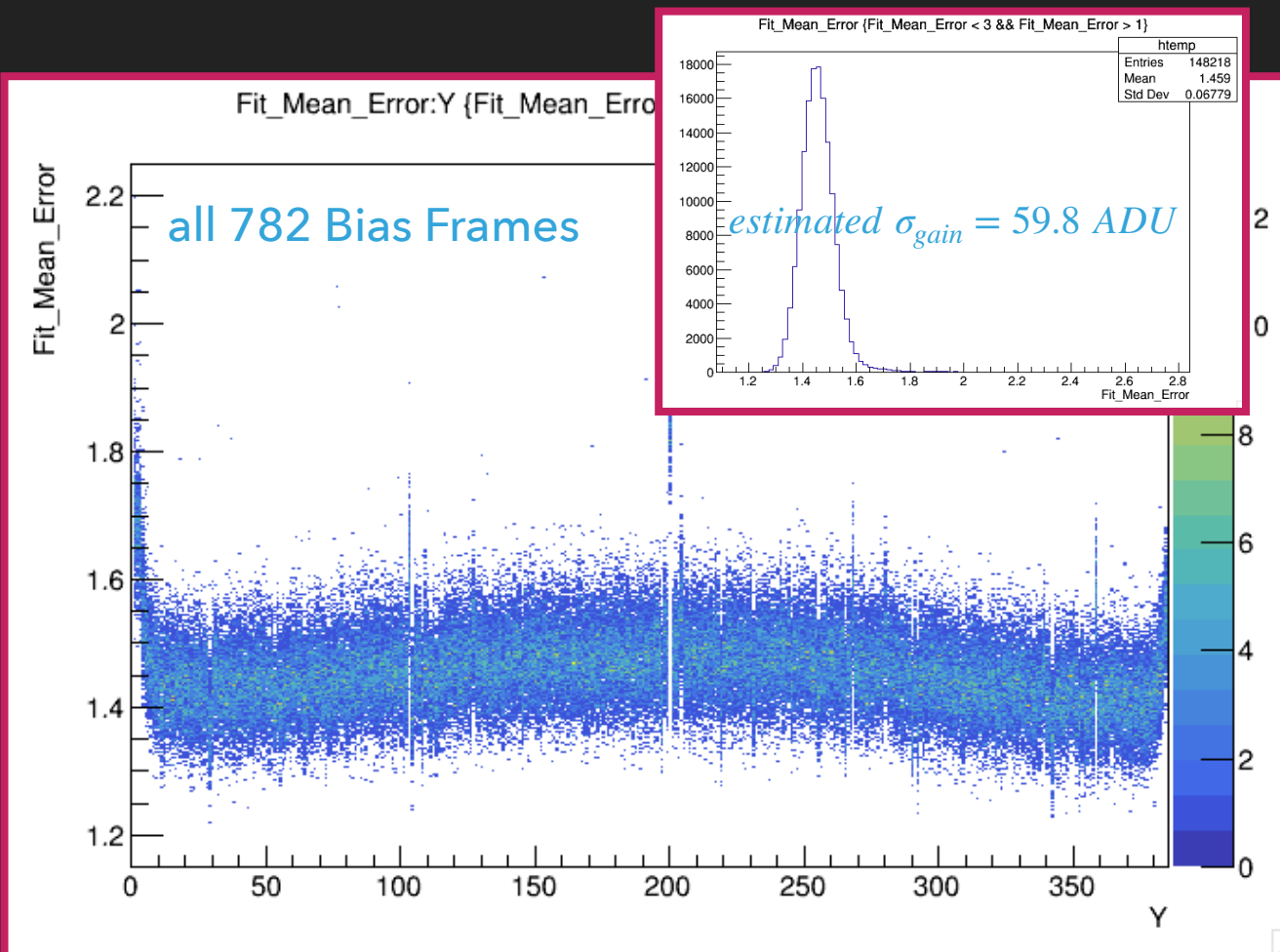
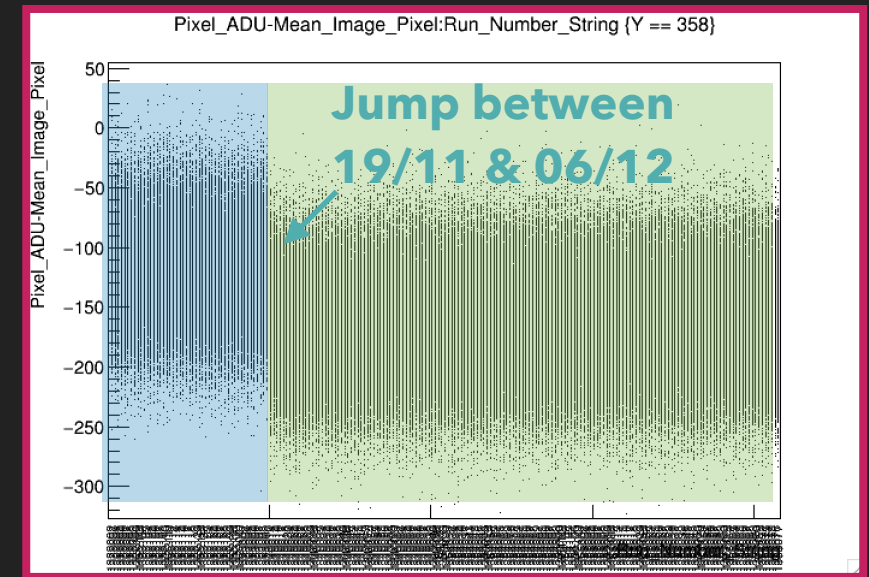
LIGHT SUM SQUARE – COLUMN PIXEL PEDESTAL JUMP

- ▶ We can plot the error on the fit mean vs y to highlight these y values
- ▶ Note: for the jumped pixels the error on the fit is misleading as it is fitting for one instead of multiple Gaussians -> they systematic this could produce would be on the order of 100's of ADU
- ▶ Also note the upwards tails effecting the first and last 10 columns of pixels



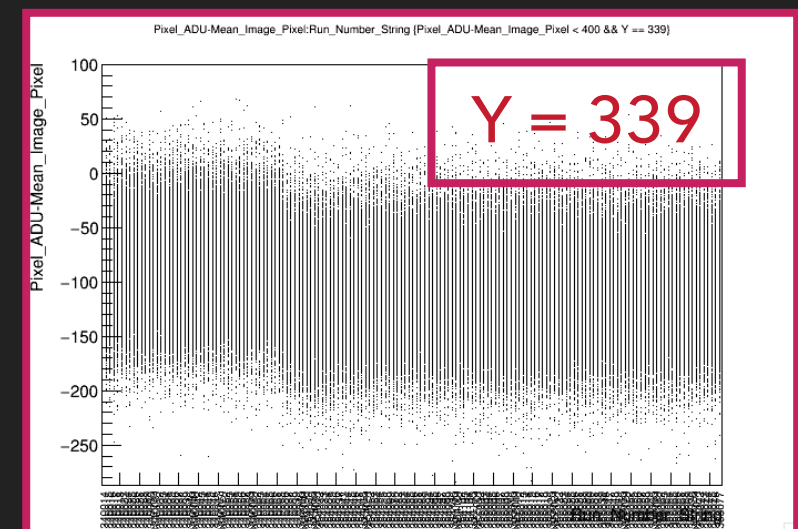
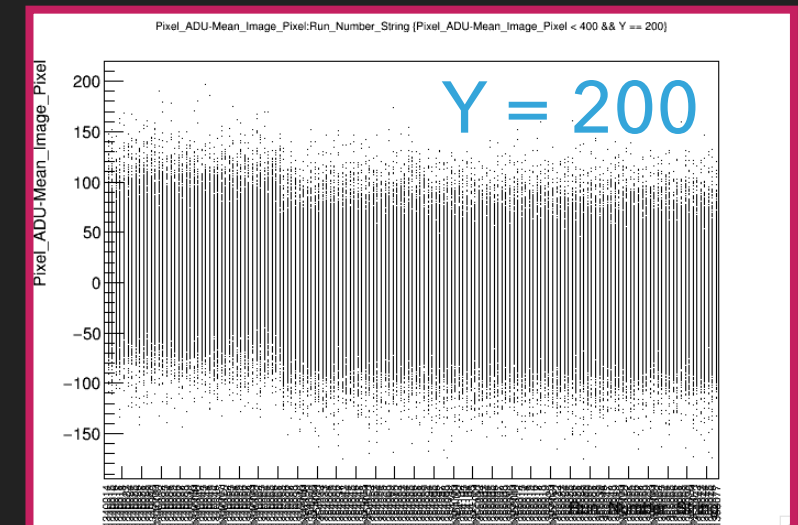
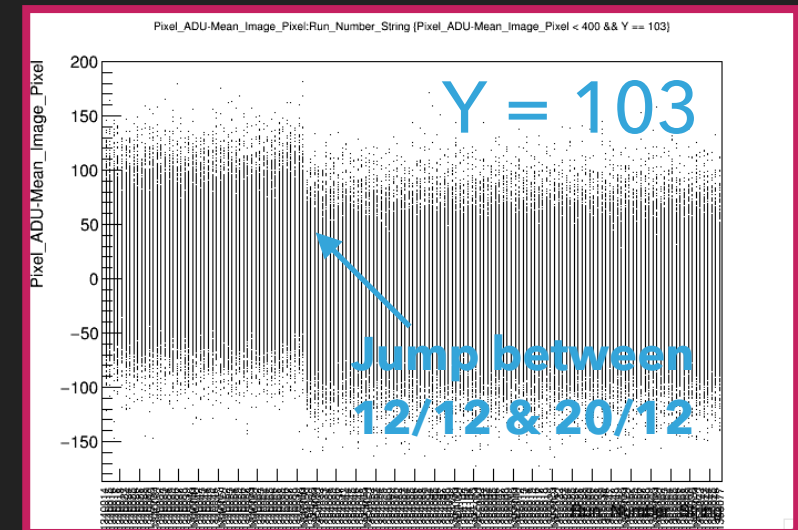
LIGHT SUM SQUARE – COLUMN PIXEL PEDESTAL JUMP

- ▶ If we cut these Nov runs we can omit the region where we have the most jumps and only lose 20% of our bias frames (note - I removed these runs out of the analysis anyway due to odd RMS behaviour) - Easy Choice
- ▶ We still have a few rows that have small jumps



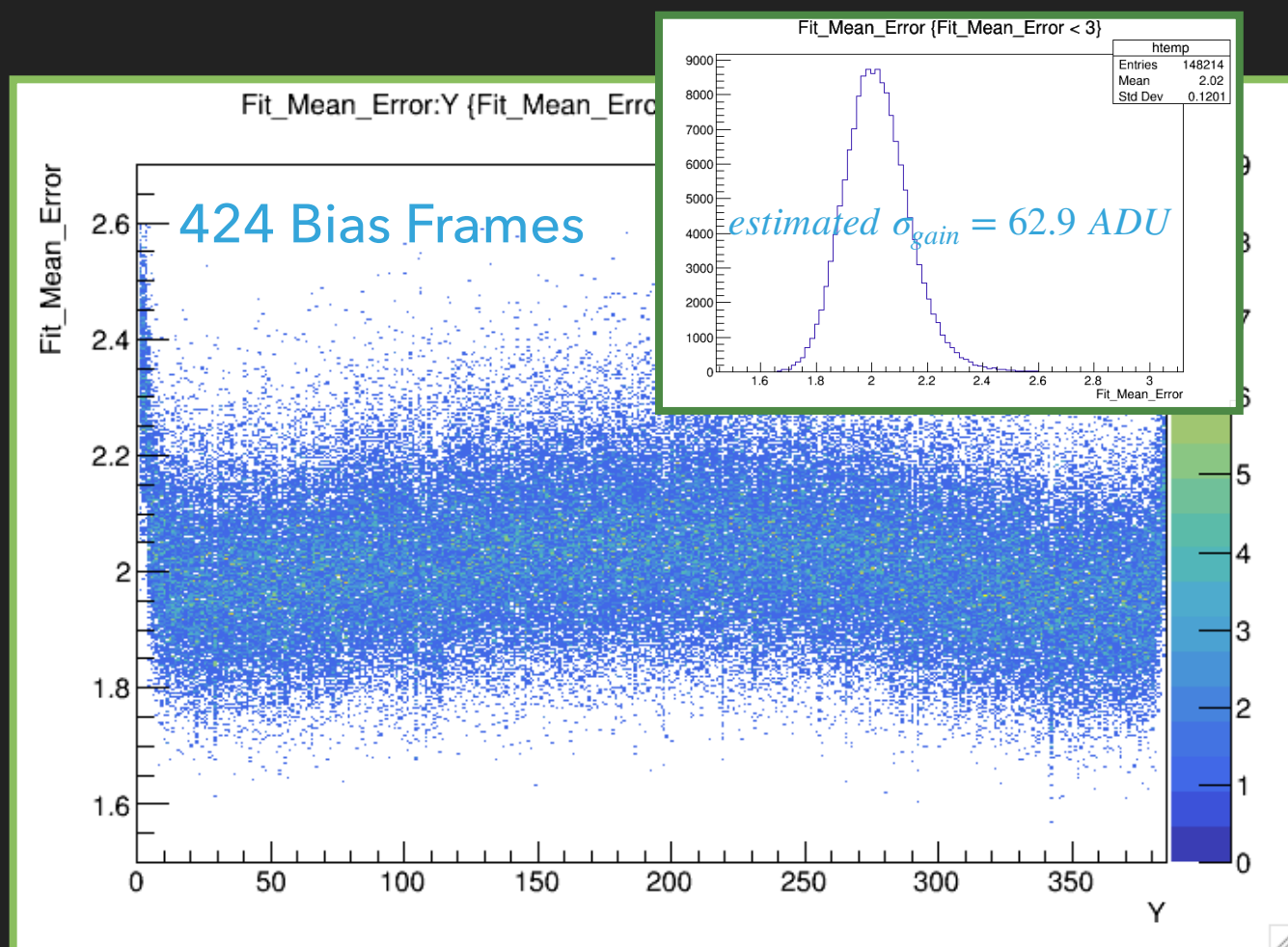
LIGHT SUM SQUARE - BAD PIXELS

- ▶ After this November cut we still have:
 - ▶ three columns of pixels with jumps between the 12 and 20th of Dec - One of which goes through the Source 3 & 5 box :(
 - ▶ one 30 ADU jump later in Dec, $Y = 217$; and
 - ▶ 20-30 (possibly more) single pixels with jumps
- ▶ I believe however that as these jumps seems to effect all pixels in the column by the same amount this could be corrected in analysis - the same method could be used for single pixels also

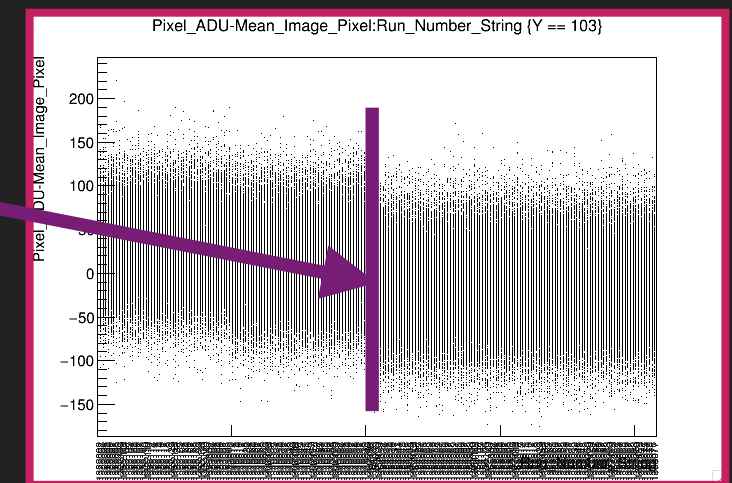


LIGHT SUM SQUARE - EFFECT OF CUTTING MORE BIAS FRAMES

- ▶ If we brought the cut later we would reduce the systematic in jump columns but increase the error on the mean for other pixels - reasonable trade off except;
- ▶ We only get rid of the systematic in jump columns for green runs and make it worse for blue ones (see table below)



Jump between 12/12 & 20/12



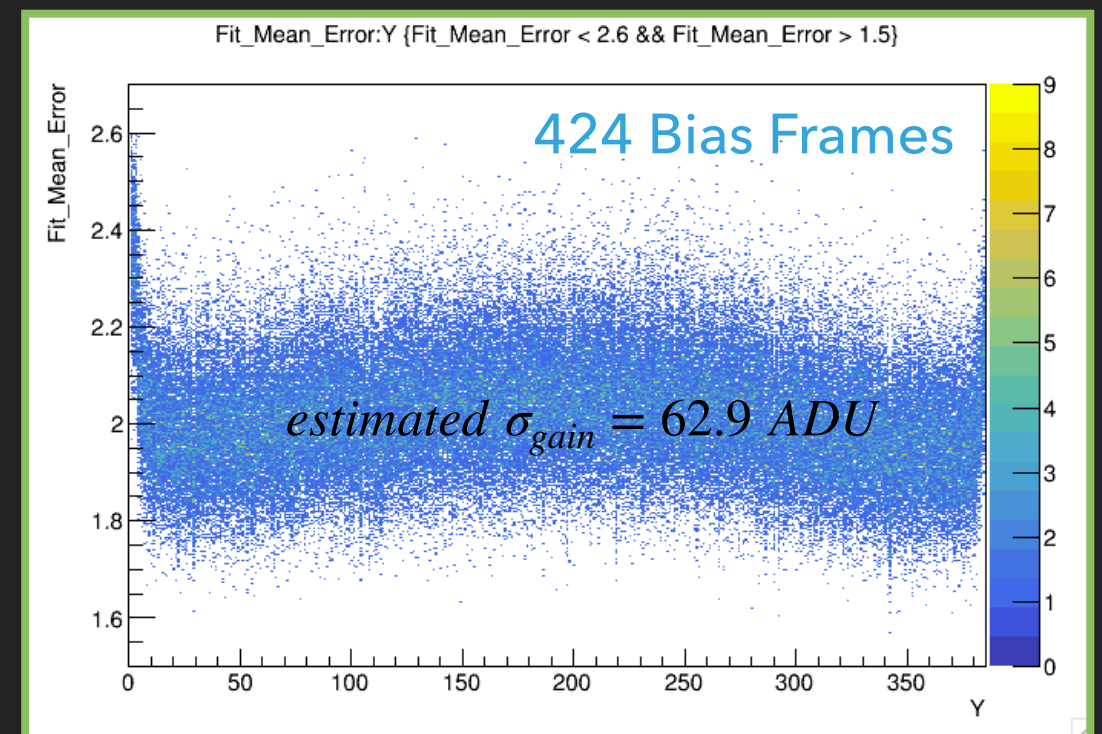
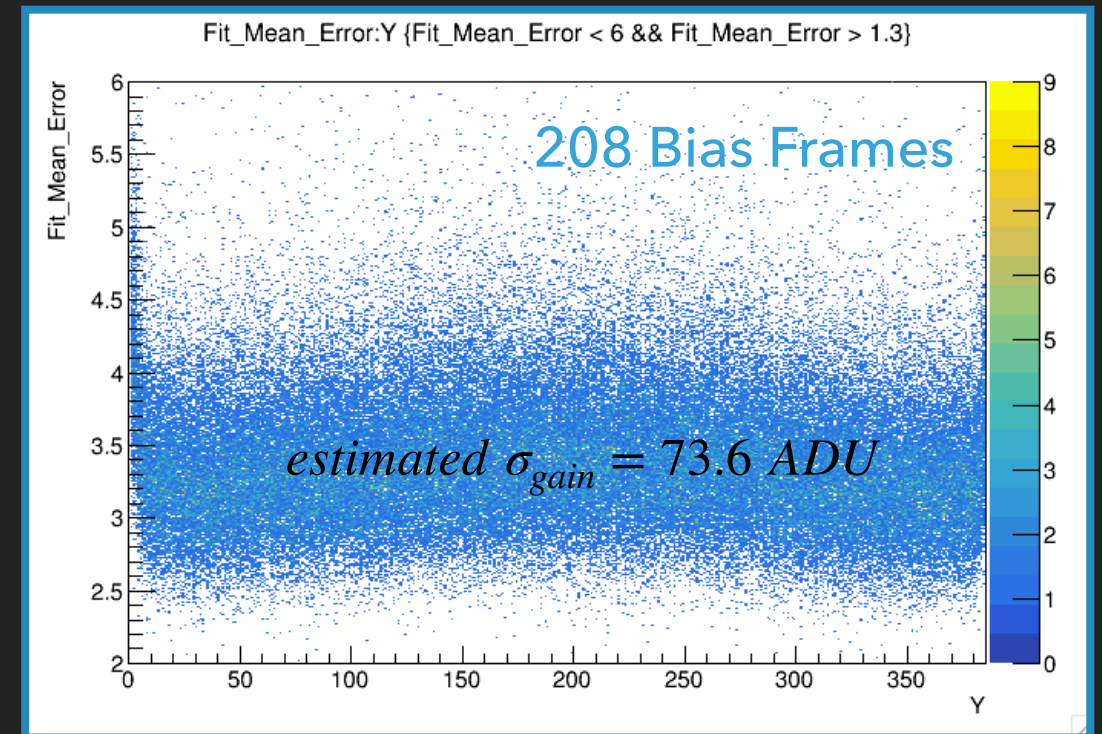
Run Number		mean_cathode_V	mean_anode1_V	mean_anode2_V	mean_anode3_V
First	Last				
1322092	1323010	9267	1699	3399	4250
1340014	1340018	6780	1000	2700	4000
1340023	1340047	6483	1179	2800	4150
1340051	1340058	6732	1399	2050	4448
1340061	1340068	6989	1416	3000	5000
1346036	1346041	6484	1399	3000	5000
1354029	1354037	6483	1639	3250	5249
1354043	1354048	6989	1759	3374	5375
1354051	1354055	6989	1879	3500	5498
1354068	1354070	6884	1999	3650	5689
1354082	1354089	6989	1999	3475	5400
1354094	1355030	6483	1999	3649	5690
1355037	1355052	6591	1749	2350	3699
1355055	1355077	6592	1749	2449	3699

LIGHT SUM SQUARE - EFFECT OF CUTTING MORE BIAS FRAMES

OPTIONS:

- ▶ Run two separate bias frames for blue and green runs:
 - ▶ fewer jumped pixels; but higher error per pixel
- ▶ Have a single bias frame but fix jumps in analysis:
 - ▶ would need to compose list of jump pixels and columns and refer back to green or blue pixel value depending on run number -> can write quick scrip to list pixels where blue - green ADU > some value
 - ▶ would give smaller error per pixel

Run Number		mean_cathode_V	mean_anode1_V	mean_anode2_V	mean_anode3_V
First	Last				
1322092	1323010	9267	1699	3399	4250
1340014	1340018	6780	1000	2700	4000
1340023	1340047	6483	1179	2800	4150
1340051	1340058	6732	1399	2050	4448
1340061	1340068	6989	1416	3000	5000
1346036	1346041	6484	1399	3000	5000
1354029	1354037	6483	1639	3250	5249
1354043	1354048	6989	1759	3374	5375
1354051	1354055	6989	1879	3500	5498
1354068	1354070	6884	1999	3650	5689
1354082	1354089	6989	1999	3475	5400
1354094	1355030	6483	1999	3649	5690
1355037	1355052	6591	1749	2350	3699
1355055	1355077	6592	1749	2449	3699



JUMP PIXELS

hBiasSubbed_run1323062_cam2

