

False Discovery Rate in

- Situation: making *many* statistical tests at once
 - e.g., Image voxels in FMRI; associating genes with disease
- Want to set threshold on statistic (e.g., F - or t -value) to control **false positive** error rate
- Traditionally: set threshold to control probability of making a **single** false positive detection
 - But if we are doing 1000s (or more) of tests at once, we have to be very stringent to keep this probability low
- **FDR**: accept the fact that there will be erroneous detections when making lots of decisions
 - Control the **fraction** of positive detections that are wrong
 - Of course, no way to tell which individual detections are right!
 - Or at least: control the expected value of this fraction

FDR: q and $z(q)$

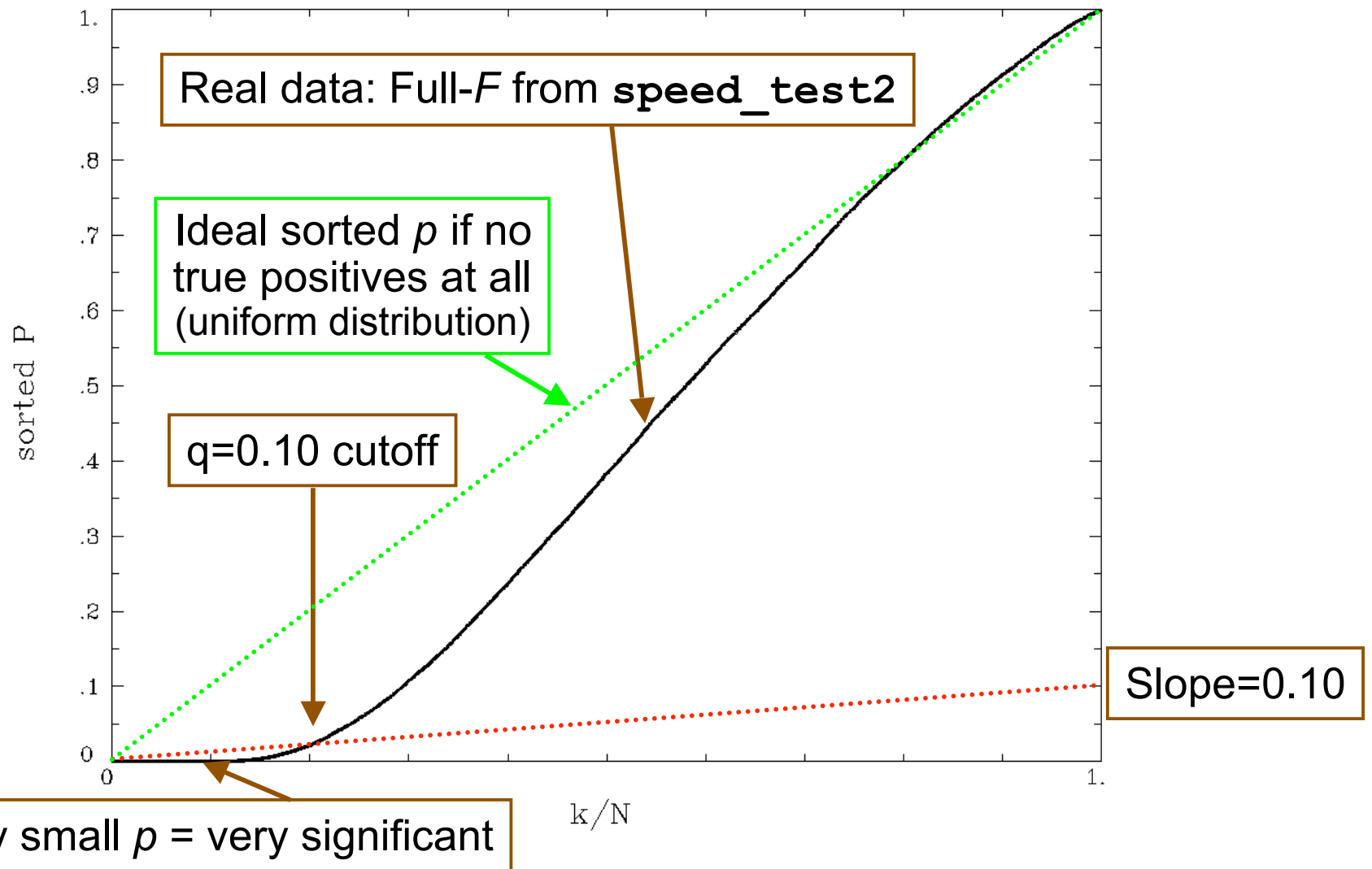
- Given some collection of statistics (say, F -values from **3dDeconvolve**), set a threshold h
- The **uncorrected p -value** of h is the probability $F > h$ when the null hypothesis is true (no activation)
 - “Uncorrected” means “per-voxel”
 - The “corrected” p -value is the probability that *any* voxel is above threshold in the case that they are all *unactivated*
 - If have N voxels to test, $p_{\text{corrected}} = 1 - (1 - p)^N \approx Np$ (for small p)
 - Bonferroni: to keep $p_{\text{corrected}} < 0.05$, need $p < 0.05 / N$, which is very tiny
- The FDR **q -value** of h is the fraction of false positives expected when we set the threshold to h
 - Smaller q is “better” (more stringent = fewer false detections)
 - $z(q)$ = conversion of q to Gaussian z -score: e.g, $z(0.05) \approx 1.95996$
 - So that larger is “better” (in the same sense): e.g, $z(0.01) \approx 2.57583$

How q is Calculated from Data

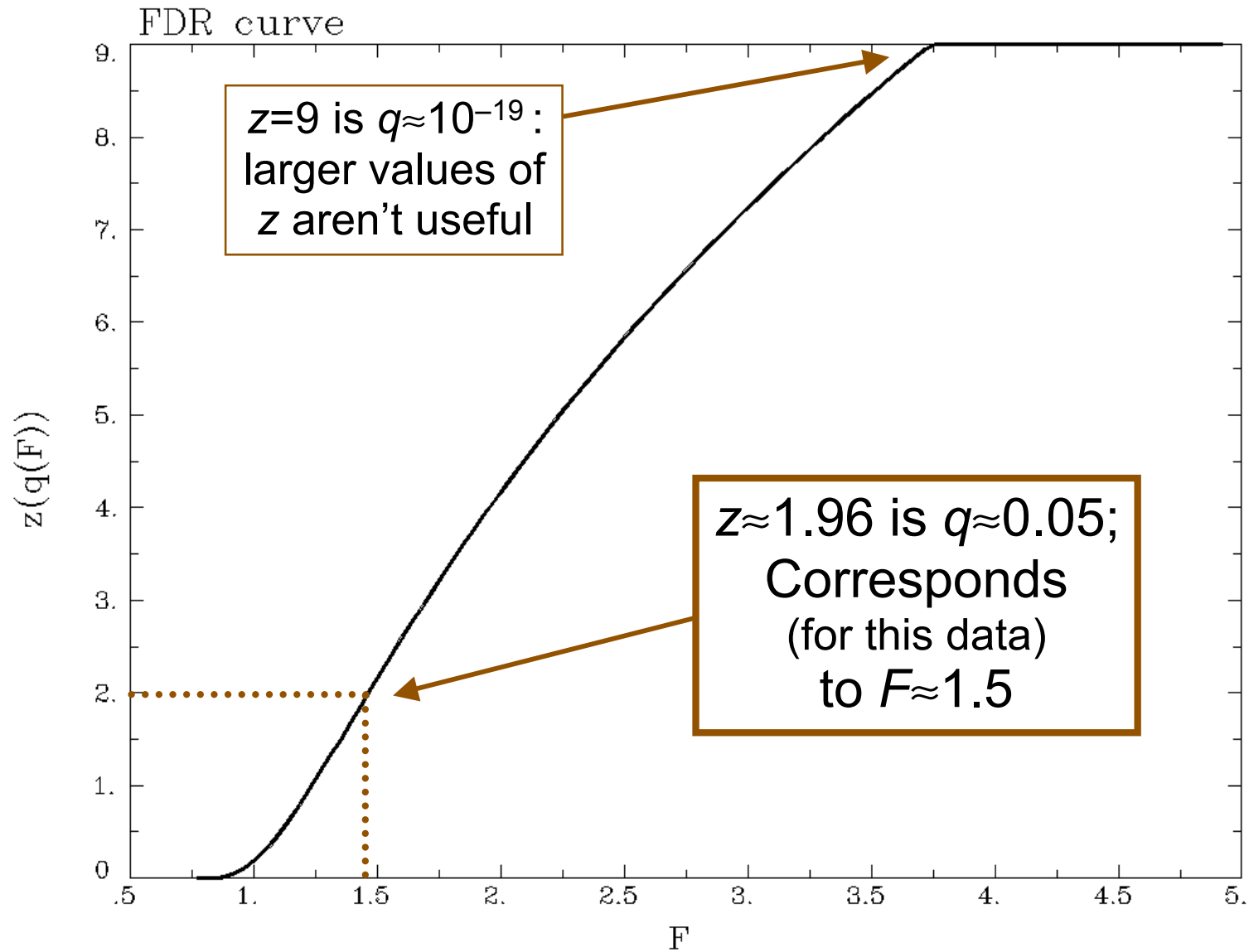
- Compute p -values of each statistic: $P_1, P_2, P_3, \dots, P_N$
- Sort these: $P_{(1)} \leq P_{(2)} \leq P_{(3)} \leq \dots \leq P_{(N)}$ {subscript_() \equiv sorted}
- For $k = 1..N$, $q_{(k)} = \min_{m \geq k} [N \cdot P_{(m)} / m]$
 - Easily computed from sorted p -values by looping downwards from $k = N$ to $k = 1$
- By keeping track of voxel each $P_{(k)}$ came from: can put q -values (or $z(q)$ values) back into image
 - This is exactly how program **3dFDR** works
- By keeping track of statistic value each $P_{(k)}$ came from: can create curve of threshold h vs. $z(q)$
- **N.B.:** q -values depend on the data in all voxels, unlike these voxel-wise (uncorrected) p -values!

Graphical Calculation of q

- Graph $P_{(k)}$ vs. k/N and draw lines from origin



Same Data: threshold F vs. $z(q)$



Recent Changes to 3dFDR

- Don't include voxels with $p=1$ (e.g., $F=0$), even if they are in the **-mask** supplied on the command line
 - This change decreases N , which will decrease q and so increase $z(q)$: recall that $q_{(k)} = \min_{m \geq k} [N \cdot P_{(m)} / m]$
- Sort with Quicksort algorithm
 - Faster than the bin-based sorting in the original code
 - Makes a big speed difference on large 1 mm³ datasets
 - Not much speed difference on small 3 mm³ grids, since there aren't so many voxels to sort
- Default mode of operation is '**-new**' method
 - Prints a warning message to let user know things have changed from the olden days
 - User can use '**-old**' method if desired

FDR curves: h vs. $z(q)$

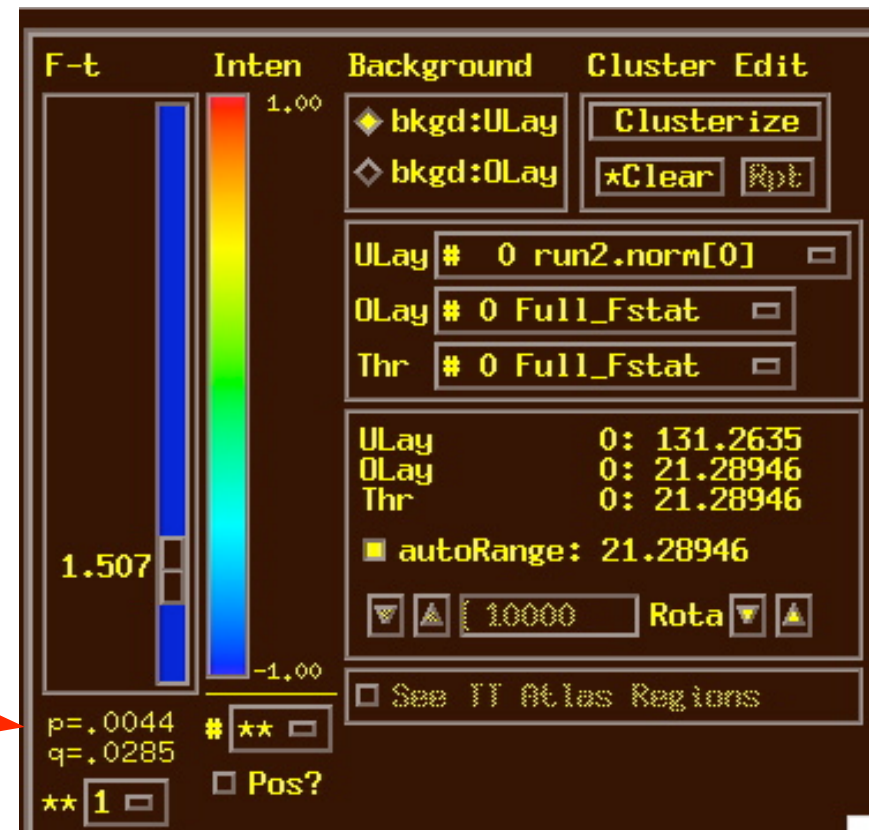
- **3dDeconvolve**, **3dANOVAX**, **3dttest**, and **3dNLFim** now compute FDR curves for all statistical sub-bricks and store them in output header
 - `THD_create_all_fdrcurves (dset)` does the work

- **3drefit -addFDR** does same for older datasets

- **3drefit -unFDR** can be used to delete such info

- **AFNI** now shows p - **and** q -values below the threshold slider bar

- Interpolates FDR curve from header (threshold $\rightarrow z \rightarrow q$)



FDR Statistical Issues

- FDR is conservative (q -values are too large) when voxels are positively correlated (e.g., from spatially smoothing)
 - Correcting for this is not so easy, since q depends on data, so a simulation like **AlphaSim** is hard to conceptualize
 - At present, FDR is alternative way of controlling false positives, vs. clustering and **AlphaSim**
 - Working on combining FDR and clustering (e.g., Pacifico, *JASA* 2004)
- Accuracy of FDR calculation depends on p -values being uniformly distributed under the null hypothesis
 - Statistic-to- p conversion should be accurate, which means that null F -distribution (say) should be correctly estimated
 - Serial correlation in FMRI time series means that **3dDeconvolve** denominator DOF is too large
 - \Rightarrow p -values will be too small, so q -values will be too small
 - Trial calculations show that this may not be a significant effect, compared to spatial smoothing (which tends to make q too large)