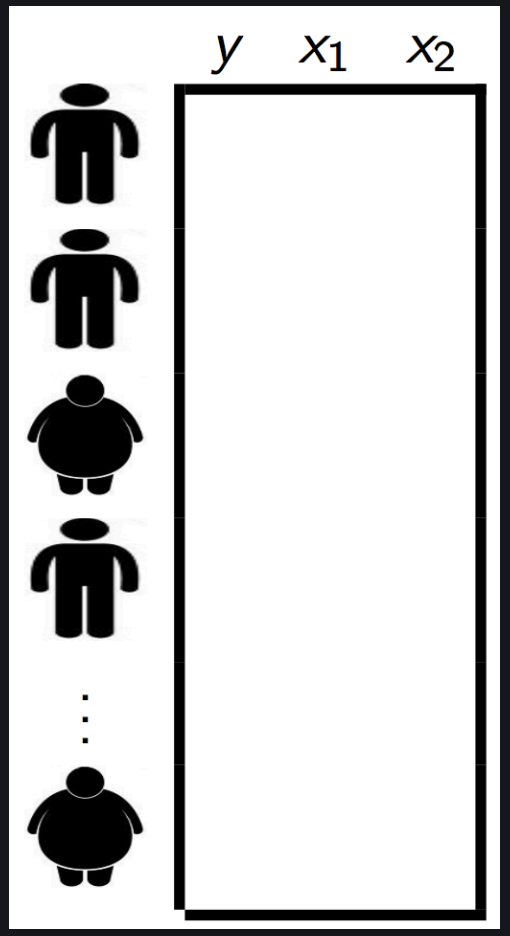
Multiple Comparisons

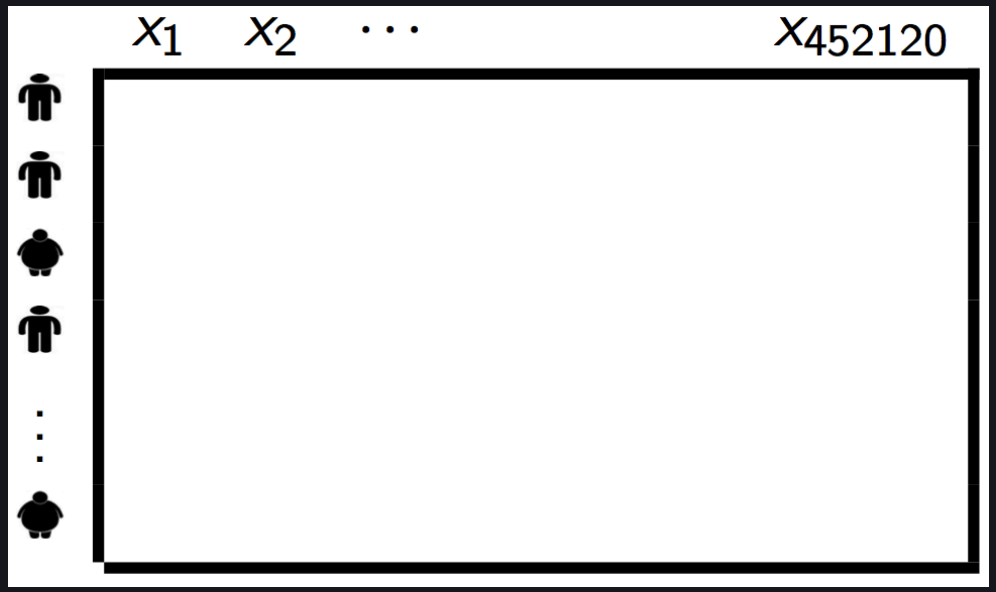
Statistical Methods in Bioinformatics

Claus Thorn Ekstrøm
UCPH Biostatistics
ekstrom@sund.ku.dk

Slides: biostatistics.dk/teaching/bioinformatics/

Data sizes. The $N \ll P$ problem





The "Big Data" revolution

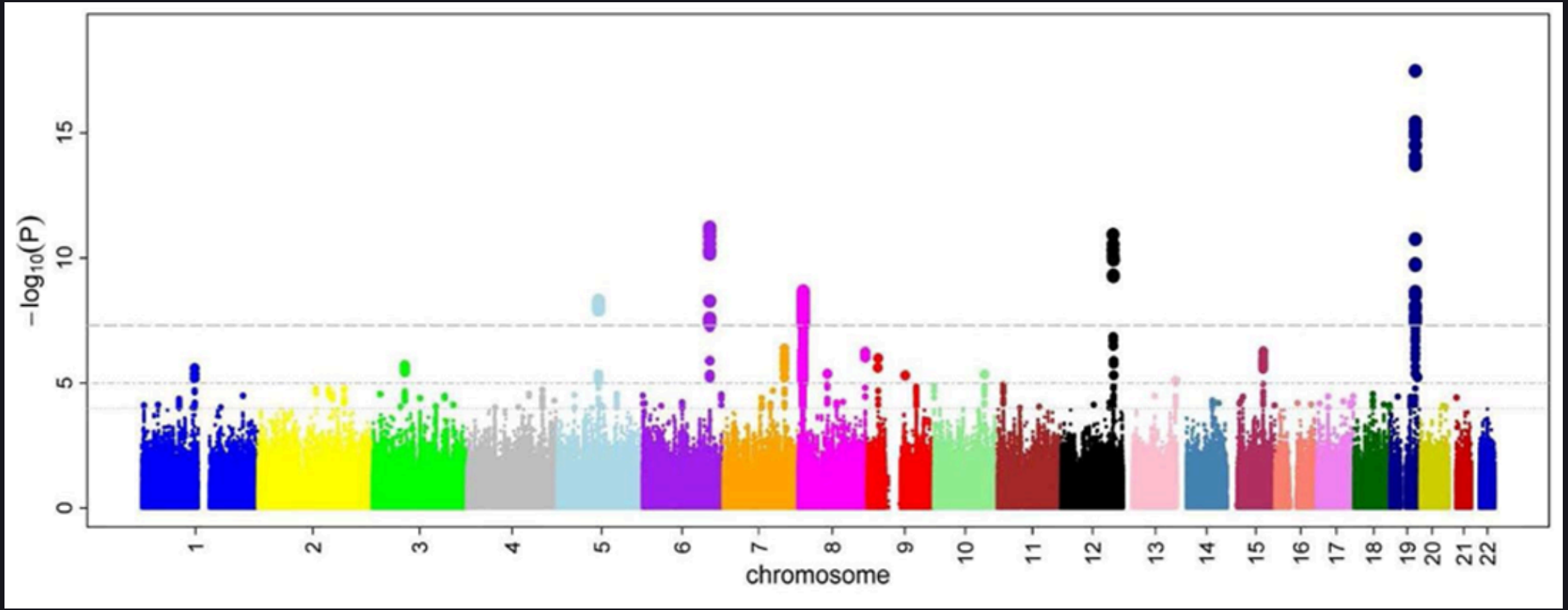
1. "Big P small N" problem with many modern large-scale-datasets: registers, images, text, *-omics, ...
2. Need to reduce the dimension in some way
3. How do we evaluate significance when we have used the data for feature selection?
4. Multiple testing becomes an issue --- not just for high-dimensional data

Example: Easy to find something "interesting"

```
sim <- function(n, p) { x  <- matrix(rnorm(n*(p+1)), ncol=(p+1)) ;
                        DF <- data.frame(x) ;
                        names(DF)[p+1] <- "Y"; DF  }
sim(100, 5) %>% lm(Y ~ ., data=.) %>% broom::tidy()
```

```
# A tibble: 6 × 5
  term        estimate std.error statistic p.value
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)  -0.149     0.104     -1.44   0.154
2 X1            0.0373    0.0954     0.391 0.697
3 X2            0.0243    0.0980     0.248 0.805
4 X3            0.0668    0.124      0.538 0.592
5 X4            0.270     0.0931     2.90  0.00468
6 X5            0.0360    0.103      0.349 0.728
```

Manhattan plot



Multiple comparison problems

Errors committed when testing a single null hypotheses, H_0

Analysis result	H0 true	H0 false
Reject	α	1-β
Don't reject	1-α	β

α is the significance level

$1 - \beta$ is the power

Multiple comparison problems

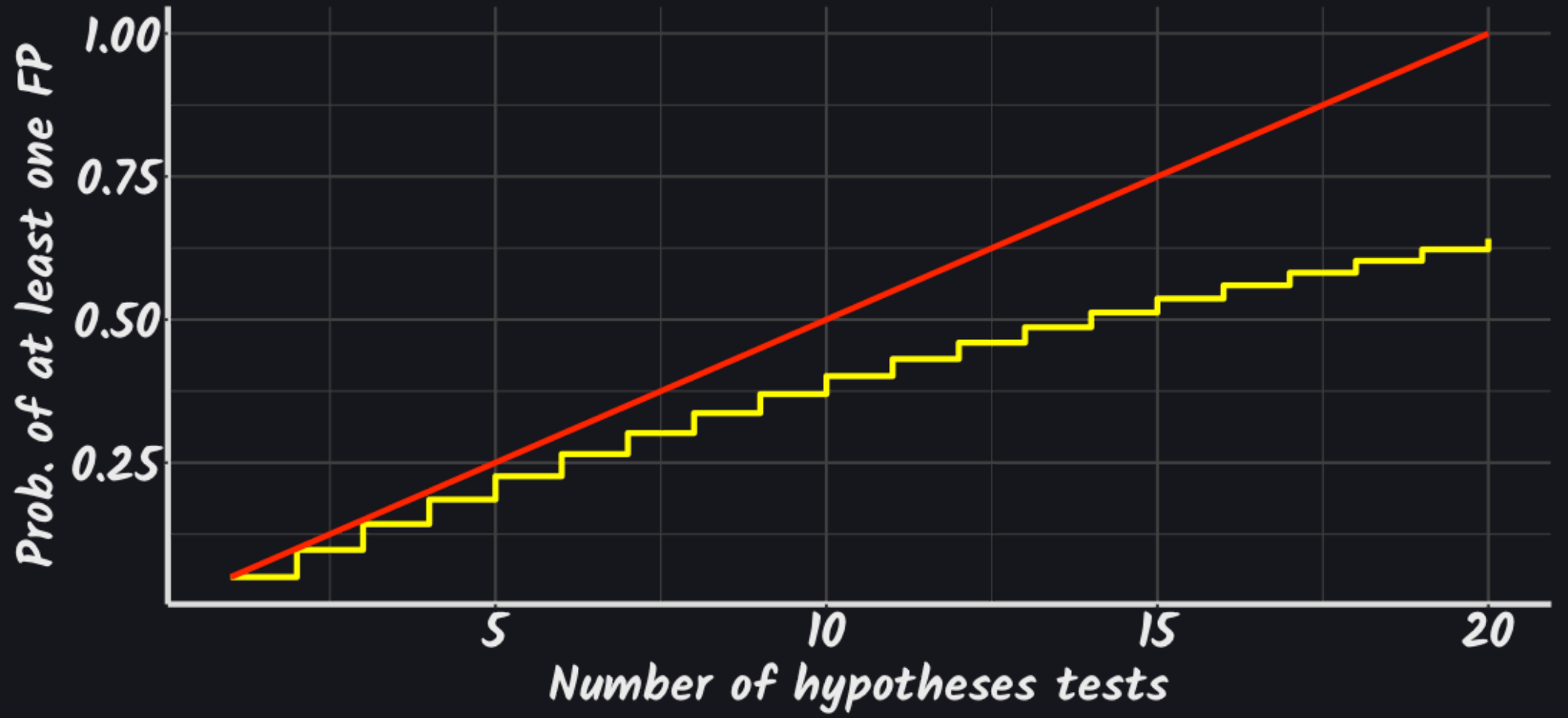
The family-wise error rate (FWER) is the probability of making at least one type I error (false positive).

For m tests we have

$$FWER = P(\cup(p_i \leq \alpha))) = 1 - P(\text{no false positives}) = 1 - (1-\alpha)^m \leq m\alpha$$

where the third equality only holds under independence, but the inequality holds due to Boole's inequality.

Multiple testing



Multiple comparison problems

Number of errors committed when testing m null hypotheses.

Analysis result	H_o true	H_o false	Total
Reject	V	S	R
Don't reject	U	T	m-R
Total	m_0	$m - m_0$	m

Here R, the number of rejected hypotheses/discoveries. V, S, U and T are unobserved. The FWER is

$$FWER = P(V > 0) = 1 - P(V = 0)$$

Bonferroni correction

The most conservative method but is free of dependence and distributional assumptions.

$$FWER = 1 - P(V = 0) = 1 - (1 - \alpha)^m \leq m\alpha$$

So set the significance level for each individual test at α/m.

In other words we reject the ith hypothesis if

$$mp_i \leq \alpha \Leftrightarrow p_i \leq \frac{\alpha}{m}$$

Sidak correction

$$1 - (1 - \alpha)^m = \alpha^* \Leftrightarrow \alpha = \sqrt[m]{1 - \alpha^*}$$

Slightly less conservative than Bonferroni (but not much). Requires independence!

Holm correction

1. Compute and order the individual p-values: $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(m)}$.
2. Find $\hat{k} = \min\{k : p_{(k)} > \frac{\alpha}{m+1-k}\}$
3. If $\hat{k}$ exists then reject hypotheses corresponding to
   $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(\hat{k}-1)}$

Holm correction

Controls the FWER: Assume the (ordered) k is the first wrongly rejected true hypothesis. Then $k \leq m - (m_0 - 1)$.

Hypothesis k was rejected so

$$p_{(k)} \leq \frac{\alpha}{m + 1 - k} \leq \frac{\alpha}{m + 1 - (m - (m_0 - 1))} \leq \frac{\alpha}{m_0}$$

Since there are m_0 true hypotheses then (Bonferroni argument) the probability that one of them is significant is at most α so FWER is controlled.

Practical problems

- While guarantee of FWER-control is appealing, the resulting thresholds often suffer from low power.

  In practice, this tends to "wipe out" evidence of the most interesting effects

- FDR control offers a way to increase power while maintaining some principled bound on error

False discovery rate

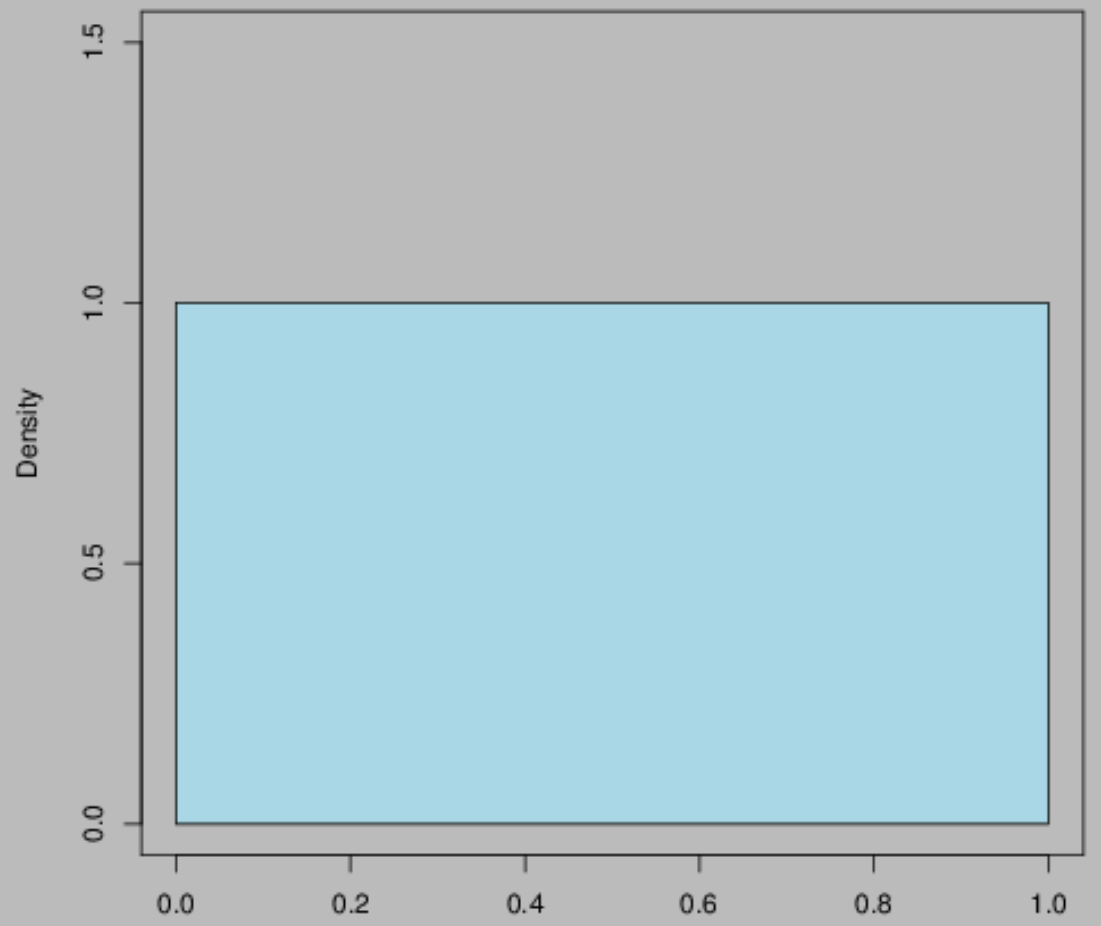
Number of errors committed when testing m null hypotheses.

Analysis result	H_0 true	H_0 false	Total
Reject	V	S	R
Don't reject	U	T	m-R
Total	m_0	$m - m_0$	m

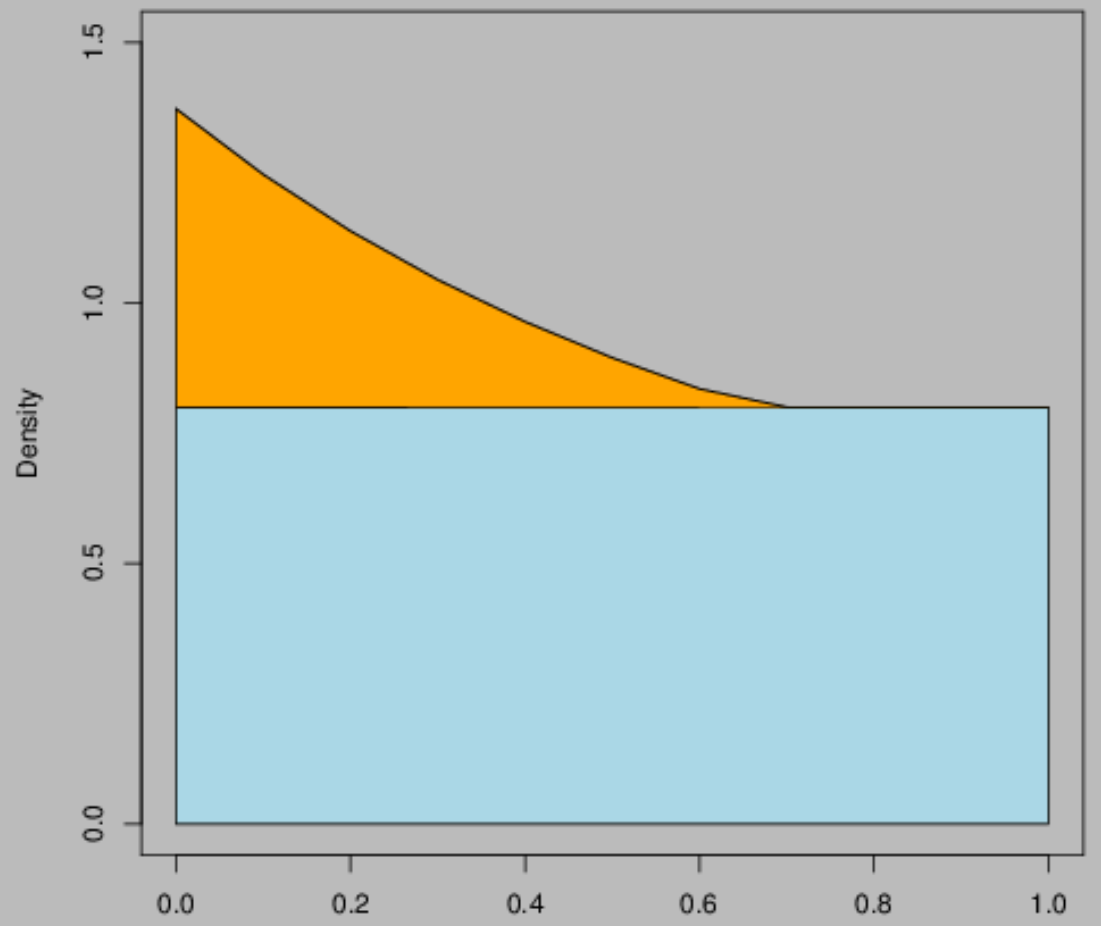
Proportion of false discoveries is $Q = \frac{V}{R}$. [Set to 0 for $R = 0$]

The false discovery rate is $FDR = E(Q) = E(\frac{V}{R})$

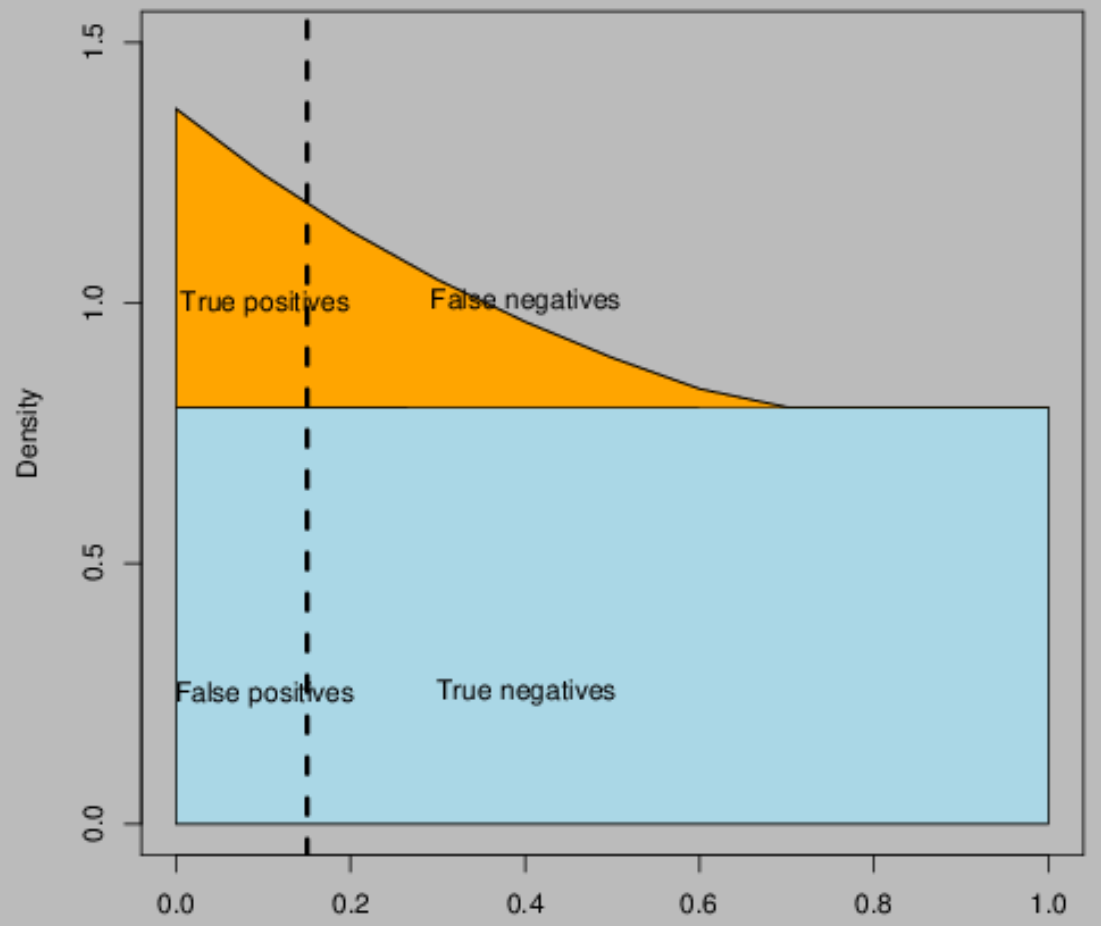
Estimating FDR



Estimating FDR



Estimating FDR



Estimating FDR — BH step-up

Benjamini-Hochberg step-up procedure to control the FDR at α.

1. Compute and order the individual p-values: $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(m)}$.
2. Find $\hat{k} = \max\{k : \frac{m}{k} \cdot p_{(k)} \leq \alpha\}$
3. If $\hat{k}$ exists then reject hypotheses corresponding to $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(\hat{k})}$

Estimating FDR — BH step-up

p-values

$$\begin{aligned} \tilde{p}_{(1)} &= \min\{\tilde{p}_{(2)}, mp_{(1)}\} \\ &\vdots \qquad\qquad \vdots \\ \tilde{p}_{(m-1)} &= \min\{\tilde{p}_{(m)}, \tfrac{m}{m-1}p_{(m-1)}\} \\ \tilde{p}_{(m)} &= p_{(m)} \end{aligned}$$

Note that each p_i is smaller or equal to the criterium in Holm's method so controls the FWER.

Estimating FDR — BH step-up

If iid of the m_0 tests (and all tests independent) and ordered so the m_0 true tests comes first. Control FDR at level q:

$$
\begin{aligned}
E(V/R) &= \sum_{r=1}^{m} E[\frac{V}{r} 1_{R=r}] = \sum_{r=1}^{m} \frac{1}{r} E[V 1_{R=r}] \\
&= \sum_{r=1}^{m} \frac{1}{r} E[\sum_{i=1}^{m_0} 1_{p_i \leq \frac{qr}{m}} 1_{R=r}] = \sum_{r=1}^{m} \frac{m_0}{r} [1_{p_1 \leq \frac{qr}{m}} 1_{R=r}] = \cdots \\
&= \sum_{r=1}^{m} \frac{m_0}{r} [\sum_{i=1}^{m_0} 1_{p_1 \leq \frac{qr}{m}} 1_{R=r}] \\
&= q \frac{m_0}{m} \leq q
\end{aligned}
$$

q values

The q-value is defined to be the FDR analogue of the p-value.

$$q \text{ value}(p_i) = \min_{t \geq p_i} \widehat{\text{FDR}}(t)$$

The q-value of an individual hypothesis test is the minimum FDR at which the test may be called significant.

q values

- When all m null hypotheses are true then FDR control is equivalent to FWER control.
- FDR approach generally gives more power than FWER control and fewer Type I errors than uncorrected testing.
- The FDR bound holds for certain classes of dependent tests. In practice, it is quite hard to "break"

Evaluating complex methods and data

When we have complex data or complex procedures/algorithms (or perhaps just big data combined with simple methods) then we still with to evaluate their results.

How stable are the results?

Randomzation/simulation tests

Sanity check: how does the method perform under realistic situations where there are *nothing* to be found?

```
sim(100, 5) %>% lm(Y ~ ., data=.) %>% broom::tidy()
```

```
# A tibble: 6 × 5
  term        estimate std.error statistic p.value
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)  -0.0646    0.0953    -0.678  0.499
2 X1           -0.149     0.101     -1.48   0.142
3 X2            0.0749    0.0928     0.808  0.421
4 X3           -0.151     0.0849    -1.78   0.0784
5 X4            0.0464    0.0927     0.501  0.618
6 X5           -0.202     0.0949    -2.13   0.0358
```

Approximate the distribution

If we have information about the distribution under the null:

- Simulate data, run algorithm to get an idea about how it behaves

If we *don't* have information about the distribution under the null

- Permutations, randomizations
- Use bootstrap, subsampling

Exercises