DS 102 Data, Inference, and Decisions **Fall 2019**

Lecture 4: Online False Discovery Rate Control

Lecturer: Michael I. Jordan

1 Uniformity of Null P-values

We return to the concept of p-values in order to prove one of their important properties.

First we recall the definition of the p-value. Let $\mathcal{S} \sim \mathcal{P}_\theta$ be the observed data set, and let $T(\mathcal{S})$ be some real-valued summary statistic obtained from $\mathcal{S}$. For example, $\mathcal{S}$ could be a set of n i.i.d. samples $X_1, \ldots, X_n$, and $T(\mathcal{S})$ could be their average $\frac{1}{n}\sum_{i=1}^n X_i$.

In the previous lecture, we assumed that the null hypothesis was $\theta \in \Theta_0$, and that the alternative was $\theta \in \Theta_1$, for some arbitrary sets Θ_0 and Θ_1. In this section, we assume $\Theta_0 = \{\theta_0\}$ and $\Theta_1 = \{\theta_1\}$, i.e. there is a single null and a single alternative distribution.

As usual, we assume that we can compute the distribution of $T(\mathcal{S}_0)$, where $\mathcal{S}_0 \sim \mathcal{P}_{\theta_0}$. Then, the p-value is defined as:

$$P := \mathbb{P}(T(\mathcal{S}_0) \geq T(\mathcal{S}) \mid T(\mathcal{S})),$$

where the "hallucinated" sample $\mathcal{S}_0$ is independent from the observed sample $\mathcal{S}$. We proclaim a discovery if P is less than or equal to the significance level α.

Denote by $F(\cdot)$ the *tail* CDF of $T(\mathcal{S}_0)$:

$$F(t) = \mathbb{P}(T(\mathcal{S}_0) > t).$$

Notice that the p-value can be written as

$$P = F(T(\mathcal{S})),$$

where $T(\mathcal{S})$ is the test statistic computed from data.

Suppose that the ground truth is null, that is $\mathcal{S} \sim \mathcal{P}_{\theta_0}$. Then, the p-value has a uniform distribution:

$$\mathbb{P}(P \leq u) = \mathbb{P}(F(T(\mathcal{S})) \leq u) = \mathbb{P}(T(\mathcal{S}) > F^{-1}(u)) = F(F^{-1}(u)) = u,$$

where we assume certain regularity condition, like F is invertible, and $T(\mathcal{S})$ has a continuous distribution. The second equality follows because F^{-1} is a monotone decreasing transformation, due to F being decreasing. The second to last equality uses the assumption that the ground truth is null. If the ground truth was alternative, we would have obtained the CDF of P at u to be $F_1^{-1}(F(u)) \neq u$, where F_1 is the tail CDF under the alternative.

It is important to remember that p-values are random variables, because they are equal to a deterministic function F applied to a random variable $T(\mathcal{S})$, whose randomness comes from the randomness in the data.

If we assume that Θ_0 is not a single point θ_0, but a possibly larger set, we can show that the distribution of p-values is dominated by a uniform $\mathbb{P}(P \le u) \le u$. This property is sometimes referred to as *super-uniformity*, and is enough to show that all multiple testing procedures we consider in this class control FDR/FWER for arbitrary sets Θ_0.

Given that null p-values are uniform, it is now clear why the simple decision rule of rejecting the null if $P \le \alpha$ controls the probability of a false discovery under α (note that now we are talking about a single test, not multiple testing). That is because

$$\mathbb{P}(\text{false discovery}) = \mathbb{P}(P \le \alpha \mid \text{ground truth is null}) = \alpha.$$

2 Online False Discovery Rate Control

Classical statistics, including the Benjamini-Hochberg (BH) algorithm, focused on a batch setting in which we make decisions after all necessary data has already been collected. For instance, the BH procedure needs the p-values for all hypotheses one wishes to test *before* making any discoveries.

In turns out that it is possible to construct procedures that make decisions sequentially in time, while controlling the FDR at any given moment. In particular, these procedures allow performing tests one after the other, and at any given time they only need information about past tests, and do not require any knowledge about future tests, such as their number of the hypotheses they wish to test. Even more remarkably, such methods can provide FDR control over a lifetime, for a possibly infinite number of tests.

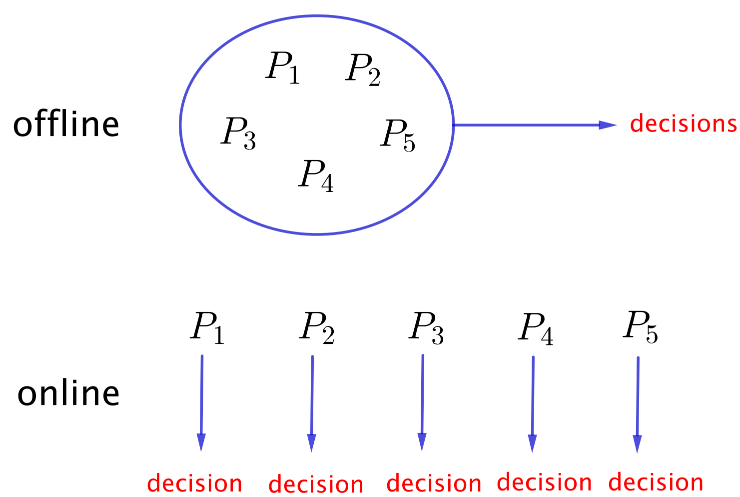

Figure 1.1: Illustration of offline versus online FDR control.

More formally, the setup of online FDR control is as follows. At every time $t \in \mathbb{N}$, a new p-value

P_t arrives. As soon as this p-value arrives, a decision has to be made of whether the corresponding hypothesis should be rejected. Importantly, this decision does not depend on any future tests. More precisely, online FDR algorithms determine an appropriate test level α_t, dependent on the tests from time 1 up to time $t-1$, and reject the t-th hypothesis if $P_t \leq \alpha_t$.

It is not clear how a procedure like BH would be adjusted to run in an online fashion. The Bonferroni correction, on the other hand, could be adjusted. The main idea behind the Bonferroni correction was to have individual test levels α_i sum up to α. We can do this for infinitely many p-values by picking a sequence $\{\gamma_i\}_{i=1}^{\infty}$ such that $\sum_{i=1}^{\infty} \gamma_i = 1$. If we set $\alpha_i = \gamma_i \alpha$, we get FWER for infinitely many tests. This procedure is sometimes referred to as alpha-spending. However, notice that its construction necessarily implies that $\alpha_i \to 0$ as $i \to \infty$. In other words, after a large enough number of tests, we are very unlikely to make any discoveries.

Here is a glimmer of hope though: FDP is a ratio of two counts: the number of false discoveries, and the total number of discoveries. We can make the ratio small in one of two ways - either by making the numerator small, or by making the denominator big. We can make the denominator big by making lots of discoveries.

The first online FDR algorithm was due to Foster and Stine and is called alpha-investing, to contrast the difference with alpha-spending. The authors gave an economic interpretation to the algorithm, which also transferred to more recent online FDR algorithms. The basic idea is as follows. When initially setting the target FDR level, we give some "initial wealth" to the algorithm. For every test we start, we "invest" some of the wealth into that test. If we don't make a discovery, we end up losing our investment. If, on the other hand, we do make a discovery (regardless of whether it's a true or false one), we get rewarded for that discovery and earn back some wealth. Online FDR algorithms are constructed in a way that ensures we never run out of wealth, by never investing all the wealth we have at a given moment.

A more recent (and slightly simpler) online FDR algorithm is due to Javanmard and Montanari and is called LORD. We state the steps of this procedure below.

Algorithm 1 The LORD Procedure

input: FDR level α, non-increasing sequence $\{\gamma_t\}_{t=1}^{\infty}$ such that $\sum_{t=1}^{\infty} \gamma_t = 1$, initial wealth $W_0 \leq \alpha$
Set $\alpha_1 = \gamma_1 W_0$
for $t = 1, 2, \ldots$ **do**
  p-value P_t arrives
  if $P_t \leq \alpha_t$, reject corresponding hypothesis
  $\alpha_{t+1} = \gamma_{t+1} W_0 + \gamma_{t+1-\tau_1}(\alpha - W_0)\mathbf{1}\{\tau_1 < t\} + \alpha \sum_{j=2}^{\infty} \gamma_{t+1-\tau_j} \mathbf{1}\{\tau_j < t\}$,
  where τ_j is time of j-th rejection $\tau_j = \min\{k : \sum_{l=1}^{k} \mathbf{1}\{P_l \leq \alpha_l\} = j\}$
end

Immediately we see that the LORD procedure is an online algorithm, as p-values P_t arrive at each time step, and decisions about whether to accept or reject the corresponding hypothesis must also be made at time step t. As discussed above, this is done by comparing P_t to a time-dependent significance threshold α_t.

To gain some intuition, let's take a closer look at the update step that determines α_{t+1} in every round. The last term is a sum over all of the previous discovery declarations, indexed by j. Each discovery gains α total wealth, and the $\gamma_{t+1-\tau_j}$ term in the sum serves to spread that wealth over all future time steps. The first term can be thought as doing the same for a "borrrowed" initial wealth W_0, whereas the second term pays off that borrowing over time.

We'll investigate this algorithm further in the next lecture, but for now there are two major take-aways: (1) online control of the FDR is possible, and (2) there are in fact implementable algorithms for achieving online FDR control arbitrarily far into the future!