

Differentially Private ANOVA

August 3, 2017

1 The ANOVA Framework

The setting is that we have a dataset of k groups, each populated with an arbitrary number of individuals. Let n be the total number of individuals. The goal of ANOVA is to test if there is a statistical difference between the means of these groups. This test is ubiquitous in the social sciences, as well as in biology. Our goal is to develop a framework for executing ANOVA tests in a differentially private manner, as the data being evaluated is often sensitive. To that end, we will define the specific terms of an ANOVA.

Let \mathcal{D} be our database. Let $\{\mathcal{D}_i : 1 \leq i \leq k\}$, be a partition of \mathcal{D} such that $\bigcup_{i=1}^k \mathcal{D}_i = \mathcal{D}$. We will denote entries y_{ij} , which says that this is the j th entry in the i th group. Let $x_i = |\mathcal{D}_i|$, $y_{ij} \in [0, 1]$, \bar{y}_i the mean of \mathcal{D}_i , and \bar{y} , the mean over all of the data. We will not assume that every group has the same size. There are three important quantities in the ANOVA framework:

$$\begin{aligned} \text{SST} &= \sum_{i=1}^k \sum_{j=1}^{x_i} (y_{ij} - \bar{y})^2; \\ \text{SSA} &= \sum_{i=1}^k x_i (\bar{y}_i - \bar{y})^2; \\ \text{SSE} &= \sum_{i=1}^k \sum_{j=1}^{x_i} (y_{ij} - \bar{y}_i)^2. \end{aligned}$$

These are the sum of squares total, treatment, and error, respectively. We further define two more terms:

$$\text{MSA} := \frac{1}{k-1} \sum_{i=1}^k \sum_{j=1}^{x_i} (\bar{y}_i - \bar{y})^2 = \frac{\text{SSA}}{k-1},$$

and

$$\text{MSE} := \frac{1}{n-k} \sum_{i=1}^k \sum_{j=1}^{x_i} (y_{ij} - \bar{y}_i)^2 = \frac{SSE}{n-k}.$$

These are the mean squared treatment and mean squared error, respectively. We use these to calculate the F-ratio, $F = \frac{\text{MSA}}{\text{MSE}}$. This gives us a p-value, based on the F-distribution.

As a first attempt at making this framework differentially private, we will support releasing noisy versions of \bar{y} and \bar{y}_i . We will also support noisy versions of the following queries:

$$d : \mathcal{D} \times [0, 1] \rightarrow \mathbb{R}$$

given by

$$d(\mathcal{D}, c) := \sum_{i=1}^k \sum_{j=1}^{x_i} (y_{ij} - c)^2,$$

and

$$g : \mathcal{D} \times [0, 1] \rightarrow \mathbb{R}$$

given by

$$g(\mathcal{D}, c) := \sum_{i=1}^k x_i (\bar{y}_i - c)^2.$$

In this way, if the data analyst chooses c as the noisy \bar{y} or \bar{y}_i , we support noisy versions of SST, SSA, and SSE.

2 Naive Approach

Our first attempt will be to do a straightforward worst-case analysis of the sensitivity of these various parts of ANOVA, and add the corresponding Laplacian noise.

2.1 Releasing noisy \bar{y}

Define the mean query,

$$m : \mathcal{D} \rightarrow \mathbb{R}$$

as

$$m(\mathcal{D}) = \frac{1}{n} \sum_{i=1}^k x_i \bar{y}_i$$

We will analyze the global sensitivity of m . In the worst case there is one group, say \mathcal{D}_k , with one member whose value is zero. This means there exists a neighboring database \mathcal{D}' where this member is in a different group, say \mathcal{D}_{k-1} , with value 1. So we get the following:

$$\begin{aligned} \Delta m &= \max_{\mathcal{D}, \mathcal{D}' \text{ neighbors}} \left\| m(\mathcal{D}) - m(\mathcal{D}') \right\|_1 \\ &= \left| \frac{1}{k} \sum_{i=1}^k \left(\frac{1}{x_i} \sum_{j=1}^{x_i} y_{ij} \right) - \frac{1}{k-1} \sum_{i=1}^{k-1} \left(\frac{1}{x'_i} \sum_{j=1}^{x'_i} y'_{ij} \right) \right| \\ &= \left| \frac{1}{k} \left(\sum_{i=1}^{k-2} \frac{1}{x_i} \sum_{j=1}^{x_i} y_{ij} + \bar{y}_{k-1} \right) - \frac{1}{k-1} \left(\sum_{i=1}^{k-2} \frac{1}{x'_i} \sum_{j=1}^{x'_i} y'_{ij} + \bar{y}'_{k-1} \right) \right| \\ &= \left| \frac{1}{k} \bar{y}_{k-1} - \frac{1}{k-1} \bar{y}'_{k-1} + \frac{1}{k} \sum_{i=1}^{k-2} \frac{1}{x_i} \sum_{j=1}^{x_i} y_{ij} - \frac{1}{k-1} \sum_{i=1}^{k-2} \frac{1}{x'_i} \sum_{j=1}^{x'_i} y'_{ij} \right| \\ &= \left| \frac{1}{k} \bar{y}_{k-1} - \frac{1}{k-1} \bar{y}'_{k-1} + \sum_{i=1}^{k-2} \frac{1}{x_i} \sum_{j=1}^{x_i} y_{ij} \left(\frac{1}{k} - \frac{1}{k-1} \right) \right| \\ &= \left| \frac{1}{k} \bar{y}_{k-1} - \frac{1}{k-1} \bar{y}'_{k-1} - \frac{1}{k(k-1)} \sum_{i=1}^{k-2} \frac{1}{x_i} \sum_{j=1}^{x_i} y_{ij} \right| \\ &\leq \left| -\frac{1}{k-1} - \frac{1}{k(k-1)} \sum_{i=1}^{k-2} \frac{1}{x_i} \sum_{j=1}^{x_i} y_{ij} \right| \\ &\leq \left| -\frac{1}{k-1} - \frac{1}{k-1} \right| \\ &= \left| \frac{-2}{k-1} \right| \\ &= \frac{2}{k-1} \text{ for } k > 1. \end{aligned}$$

We can now use the Laplace Mechanism of Dwork[citation needed] to create a differentially private algorithm for releasing \bar{y} .

Algorithm 1

Input: Database $\mathcal{D} = \{\mathcal{D}_i \mid 1 \leq i \leq k\}$
Output: Noisy \bar{y}
for $i = 1$ **to** k **do**
 Compute \bar{y}_i
end for
Compute $\bar{y} = \frac{1}{k} \sum_{i=1}^k \bar{y}_i$
Compute $x = \bar{y} + Y$ where $Y \sim \text{Lap}(\frac{2}{\epsilon})$
return x

Theorem 1. *Algorithm 1 preserves $(\epsilon, 0)$ -differential privacy.*

2.2 Releasing noisy \bar{y}_i

We will now calculate the sensitivity of \bar{y}_i . Model this as a database query

$$m_i : \mathcal{D} \rightarrow \mathbb{R}$$

$$m_i(\mathcal{D}) = \frac{1}{x_i} \sum_{j=1}^{x_i} y_{ij}.$$

In the worst case \mathcal{D}_i has a member with value 1. This means there exists a neighboring database with \mathcal{D}'_i that is the same as \mathcal{D}_i except that that one member now has value 0. This gives us the following sensitivity bound.

$$\begin{aligned} \Delta m_i &= \max_{\mathcal{D}, \mathcal{D}' \text{ neighbors}} \left\| m_i(\mathcal{D}) - m_i(\mathcal{D}') \right\|_1 \\ &= \left| \frac{1}{x_i} \sum_{j=1}^{x_i} y_{ij} - \frac{1}{x_i} \sum_{j=1}^{x_i} y'_{ij} \right| \\ &= \left| \frac{1}{x_i} \left(\sum_{j=1}^{x_i-1} y_{ij} + 1 \right) - \frac{1}{x_i} \left(\sum_{j=1}^{x_i-1} y_{ij} + 0 \right) \right| \\ &= \frac{1}{x_i}. \end{aligned}$$

So again, we can add the corresponding noise given by $\text{Lap}(\frac{1}{\epsilon x_i})$.

2.3 Releasing noisy d

We now analyze the sensitivity of the query $d : \mathcal{D} \times [0, 1] \rightarrow \mathbb{R}$ given by $d(\mathcal{D}, c) = \sum_{i=1}^k \sum_{j=1}^{x_i} (y_{ij} - c)^2$. In the worst case one person's value changes from 0 in \mathcal{D} to 1 in a neighboring database \mathcal{D}'

$$\begin{aligned}
\Delta d &= \max_{\mathcal{D}, \mathcal{D}' \text{ neighbors}} \left\| d(\mathcal{D}) - d(\mathcal{D}') \right\|_1 \\
&= \left| \left(\sum_{i=1}^k \sum_{j=1}^{x_i} (y_{ij} - c)^2 \right) - \left(\sum_{i=1}^{k-1} \sum_{j=1}^{x'_i} (y'_{ij} - c)^2 \right) \right| \\
&= \left| \sum_{i=1}^{k-1} \sum_{j=1}^{x_i} (y_{ij} - c)^2 - \sum_{i=1}^{k-1} \sum_{j=1}^{x'_i} (y'_{ij} - c)^2 - c^2 \right| \\
&= \left| \left(\sum_{i=1}^{k-2} \sum_{j=1}^{x_i} (y_{ij} - c)^2 + \sum_{j=1}^{x_{k-1}} (y_{(k-1)j} - c)^2 \right) - \left(\sum_{i=1}^{k-2} \sum_{j=1}^{x'_i} (y'_{ij} - c)^2 + \sum_{j=1}^{x_{k-1}} (y'_{(k-1)j} - c)^2 \right) - c^2 \right| \\
&= \left| \sum_{j=1}^{x_{k-1}} (y_{(k-1)j} - c)^2 - \sum_{j=1}^{x'_{k-1}} (y'_{(k-1)j} - c)^2 - c^2 \right| \\
&= \left| \sum_{j=1}^{x_{k-1}} (y_{(k-1)j} - c)^2 - \left(\sum_{j=1}^{x_{k-1}} (y_{(k-1)j} - c)^2 + (1 - c)^2 \right) - c^2 \right| \\
&= |-1 + 2c - 2c^2| \\
&\leq 1 \text{ for } c \in [0, 1].
\end{aligned}$$

We can use this to get an algorithm for releasing d with added noise from $\text{Lap}(\frac{1}{\epsilon})$.

Algorithm 2

Input: Database $\mathcal{D} = \{\mathcal{D}_i \mid 1 \leq i \leq k\}$, constant $c \in [0, 1]$

Output: Noisy $d(\mathcal{D}, c)$

$y = 0$

for $i = 1$ to k **do**

for $j = 1$ to x_i **do**

 Compute $y = y + (y_{ij} - c)^2$

end for

end for

Compute $x = y + Y$ where $Y \sim \text{Lap}(\frac{1}{\epsilon})$

return x

Theorem 2. *Algorithm 2 preserves $(\epsilon, 0)$ -differential privacy.*

2.4 Releasing noisy g

We now analyze the sensitivity of g . In the worst case we have a group, say \mathcal{D}_k , with one member, whose value is 1. Then there exists a neighboring database \mathcal{D}' that is the same as \mathcal{D} , except that the member in \mathcal{D}_k has moved to \mathcal{D}'_k and has value 0. This gives us the following bound.

$$\begin{aligned} \Delta g &= \max_{\mathcal{D}, \mathcal{D}' \text{ neighbors}} \left\| g(\mathcal{D}) - g(\mathcal{D}') \right\|_1 \\ &= \left| \sum_{i=1}^k \sum_{j=1}^{x_i} (\bar{y}_i - c)^2 - \sum_{i=1}^{k-1} \sum_{j=1}^{x'_i} (\bar{y}'_i - c)^2 \right| \\ &= \left| \sum_{i=1}^{x_{k-1}} \sum_{j=1}^{x_i} (\bar{y}_i - c)^2 + (1 - c)^2 - \sum_{j=1}^{x_{k-1}} (\bar{y}'_{k-1} - c)^2 \right| \\ &= \left| \sum_{j=1}^{x_{k-1}} (\bar{y}_{k-1} - c)^2 + (1 - c)^2 - \sum_{j=1}^{x_{k-1}+1} (\bar{y}'_{k-1} - c)^2 \right| \\ &= \left| x_{k-1} \left(\frac{1}{x_{k-1}} \sum_{j=1}^{x_{k-1}} y_{(k-1)j} \right)^2 - (x_{k-1} + 1) \left(\frac{1}{x_{k-1} + 1} \sum_{j=1}^{x_{k-1}} y_{(k-1)j} \right)^2 + (1 - c)^2 \right| \\ &\leq \frac{1}{x_{k-1} + 1} + 1. \end{aligned}$$

This gives us an algorithm for releasing g with noise added from $\text{Lap}\left(\frac{\frac{1}{x_{k-1}+1}+1}{\epsilon}\right)$.

Algorithm 3

Input: Database $\mathcal{D} = \{\mathcal{D}_i \mid 1 \leq i \leq k\}$, constant $c \in [0, 1]$

Output: Noisy $g(\mathcal{D}, c)$

$y = 0$

for $i = 1$ to k **do**

for $j = 1$ to x_i **do**

 Compute $y = y + (\bar{y}_i - c)^2$

end for

end for

Compute $x = y + Y$ where $Y \sim \text{Lap}\left(\frac{\frac{1}{x_{k-1}+1}+1}{\epsilon}\right)$

return x

2.5 Releasing the F-ratio

We can now use everything above to release the F-ratio. A user can first query for noisy \bar{y} , noisy \bar{y}_i s, then use these as input for the noisy d and g queries to get noisy SSE and SSA. We assume that k and n are public, which allows the analyst to compute MSA and MSE , the F-ratio, and thus the p-value.