Discussion 3 Solutions

Note: Starting this week, I'm going to release solutions as an answer document instead of the filled worksheet to have space to explain everything, but just FYI: all of the following is from practice.dsc80.com — the purpose of this is just so you don't have to cross-reference anything yourself!

WI23 Midterm Problem 6

Problem:

Given the above information, what does the following expression evaluate to?

`tv_excl.groupby(["Age", "Service"]).sum().shape[0]`

Solution:

Note that the DataFrame `counts` is a pivot table, created using `tv_excl.pivot_table(index="Age", columns="Service", aggfunc="size")`. As we saw in lecture, pivot tables contain the same information as the result of grouping on two columns.

The DataFrame `tv_excl.groupby(["Age", "Service"]).sum()` will have one row for every unique combination of `"Age"` and `"Service"` in `tv_excl`. (The same is true even if we used a different aggregation method, like `.mean()` or `.max()`.) As `counts` shows us, `tv_excl` contains every possible combination of a single element in {`"13+"`, `"16+"`, `"18+"`, `"7+"`, `"all"`} with a single element in {`"Disney+"`, `"Hulu"`, `"Netflix"`, `"Prime Video"`}, except for (`"13+"`, `"Disney+"`) and (`"18+"`, `"Disney+"`), which were not present in `tv_excl`; if they were, they would have non-null values in `counts`.

As such, `tv_excl.groupby(["Age", "Service"]).sum()` will have $20 - 2 = 18$ rows, and

`tv_excl.groupby(["Age", "Service"]).sum().shape[0]` evaluates to 18.

Problem:

Tiffany would like to compare the distribution of Age for Hulu and Netflix. Specifically, she'd like to test the following hypotheses:

- **Null Hypothesis**: The distributions of Age for Hulu and Netflix are drawn from the same population distribution, and any observed differences are due to random chance.

- **Alternative Hypothesis**: The distributions of Age for Hulu and Netflix are drawn from different population distributions.

Is this a hypothesis test, or a permutation test? Why?

Solution:

Answer: **Permutation test**

A permutation test is a statistical test in which we aim to determine if two samples look like they were drawn from the same unknown population. Here, our two samples are the distribution of

`"Age"`s for Hulu and the distribution of `"Age"`s for Netflix.

Problem:

Consider the DataFrame distr, defined below.

```
hn = counts[["Hulu", "Netflix"]]
distr = (hn / hn.sum()).T # Note that distr has 2 rows and 5 columns.
```

To test the hypotheses above, Tiffany decides to use the total variation distance as her test statistic. Which of the following expressions DO NOT correctly compute the observed statistic for her test?

Solution:

Answer:

`distr.diff().sum().sum().abs() / 2` only

First, note that the difference between the TVD calculation here and those in lecture is that our pivot table contains one **row** for each distribution, rather than one **column** for each distribution. This is because of the `.T` in the code snippet above. `distr` may look something like:

Age	13+	16+	18+	7+	all
Service					
Hulu	0.004103	0.415385	0.228718	0.252308	0.099487
Netflix	0.001720	0.275150	0.382631	0.210662	0.129837

As such, here we need to apply the `.diff()` method to each column first, not each row (meaning we should supply `axis=0` to `diff`, not `axis=1`; `axis=0` is the default, so we don't need to explicitly specify it). `distr.diff()` may look something like:

Age	13+	16+	18+	7+	all
Service					
Hulu	NaN	NaN	NaN	NaN	NaN
Netflix	-0.002383	-0.140234	0.153913	-0.041646	0.030349

With that in mind, let's look at each option, remembering that the TVD is the **sum of the absolute differences in proportions, divided by 2**.

- `distr.diff().iloc[-1].abs().sum() / 2`:
  - `distr.diff().iloc[-1]` contains the differences in proportions.
  - `distr.diff().iloc[-1].abs()` contains the absolute differences in proportions.
  - `distr.diff().iloc[-1].abs().sum() / 2` contains the sum of the absolute differences in proportions, divided by 2. **This is the TVD.**
- `distr.diff().sum().abs().sum() / 2`:
  - `distr.diff().sum()` is a Series containing just the last row in `distr.diff()`; remember, null values are ignored when using methods such as `.mean()` and `.sum()`.
  - `distr.diff().sum().abs()` contains the absolute differences in proportions, and hence

    `distr.diff().sum().abs().sum() / 2` contains the sum of the absolute differences in proportions, divided by 2. **This is the TVD.**

- `distr.diff().sum().sum().abs() / 2`:
  - `distr.diff().sum()` contains the differences in proportions (explained above).
  - `distr.diff().sum().sum()` contains the sum of the differences in proportions. **This is 0**; remember, the reason we use the absolute value is to prevent the positive and negative differences in proportions from cancelling each other out. As a result, this option **does not compute the TVD**; in fact, it errors, because `distr.diff().sum().sum()` is a single `float`, and `float`s don't have an `.abs()` method.
- `(distr.sum() - 2 * distr.iloc[0]).abs().sum() / 2`:
  - This option seems strange, but does actually compute the TVD. The key idea is the fact that $a - b$ is the same as $(a + b) - (2 \cdot b)$. `distr.sum()` is the same as `distr.iloc[0] + distr.iloc[1]`, so `distr.sum() - 2 * distr.iloc[0]` is `distr.iloc[0] + distr.iloc[1] - 2 * distr.iloc[0]` which is `distr.iloc[1] - distr.iloc[0]`, which is just `distr.diff().iloc[-1]`.
  - Then, this option reduces to `distr.diff().iloc[-1].abs().sum() / 2`, which is the same as Option 1. **This is the TVD.**
- `distr.diff().abs().sum(axis=1).iloc[-1] / 2`:
  - `distr.diff().abs()` is a DataFrame in which the last row contains the absolute differences in proportions.
  - `distr.diff().abs().sum(axis=1)` is a Series in which the first element is null and the second element is the sum of the absolute differences in proportions.
  - As such, `distr.diff().abs().sum(axis=1).iloc[-1] / 2` is the sum of the absolute differences in proportions divided by 2. **This is the TVD.**

WI23 Final Problem 2.5

Problem:

Suppose $\vec{a} = \begin{bmatrix} a_1 & a_2 & \dots & a_n \end{bmatrix}^T$ and $\vec{b} = \begin{bmatrix} b_1 & b_2 & \dots & b_n \end{bmatrix}^T$ are both vectors containing proportions that add to 1. As we've seen before, the TVD is defined as follows:

$\text{TVD}(\vec{a}, \vec{b}) = \frac{1}{2} \sum_{i=1}^{n} |a_i - b_i|$

The TVD is not the only metric that can quantify the distance between two categorical distributions. Here are three other possible distance metrics:

$$\text{dis1}(\vec{a}, \vec{b}) = \vec{a} \cdot \vec{b} = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$$

$\text{dis2}(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}||\vec{b}|} = \frac{a_1 b_1 + a_2 b_2 + \ldots + a_n b_n}{\sqrt{a_1^2 + a_2^2 + \ldots + a_n^2} \sqrt{b_1^2 + b_2^2 + \ldots + b_n^2}}$

$\text{dis3}(\vec{a}, \vec{b}) = 1 - \frac{\vec{a} \cdot \vec{b}}{|\vec{a}||\vec{b}|}$

Of the above three possible distance metrics, only one of them has the same range as the TVD (i.e. the same minimum possible value and the same maximum possible value) *and* has the property that smaller values correspond to more similar vectors. Which distance metric is it?

Solution:

Note: the solution here has a lot of inline LaTeX that's hard to copy — you can read the source solution here: https://practice.dsc80.com/wi23-final/index.html

FA23 Midterm Problem 4

Problem:

The `donkeys` table contains data from a research study about donkey health. The researchers measured the attributes of 544 donkeys. The next day, they selected 30 donkeys to reweigh. The first few rows of the `donkeys` table are shown below (left), and the table contains the following columns (right):

	id	BCS	Age	Weight	WeightAlt
0	d01	3.0	<2	77	NaN
1	d02	2.5	<2	100	NaN
2	d03	1.5	<2	74	NaN

`id`	A unique identifier for each donkey (`d01`, `d02`, etc.).
`BCS`	Body condition score: from 1 (emaciated) to 3 (healthy) to 5 (obese) in increments of 0.5.
`Age`	Age in years: <2, 2–5, 5–10, 10–15, 15–20, and over 20 years.
`Weight`	Weight in kilograms.
`WeightAlt`	Second weight measurement taken for 30 donkeys. NaN if the donkey was not reweighed.

Alan wants to see whether donkeys with `'BCS'` >= 3 have larger `'Weight'` values on average compared to donkeys that have
`'BCS'` < 3. Select **all the possible test statistics** that Alan could use to conduct this hypothesis test. Let μ_1 be the mean weight of donkeys with `'BCS'` >= 3 and μ_2 be the mean weight of donkeys with `'BCS'` < 3.

- A. μ_1
- B. $\mu_1 - \mu_2$

- C. $2\mu_2 - \mu_1$
- D. $|\mu_1 - \mu_2|$
- E. Total variation distance
- F. Kolmogorov-Smirnov test statistic

Answer: B and C

- A: Incorrect. μ_1 does not tell compare the two groups, and so cannot be used to see which is larger on average.
- B: Correct. $\mu_1 - \mu_2$ tells us the difference between the average weight of both groups as well as which group would be larger (a test statistic greater than zero means μ_1 is larger).
- C: Correct. $2\mu_2 - \mu_1$ tells us the difference between the average weight of both groups as well as which group would be larger (a test statistic greater than μ_2 means μ_2 is larger).
- D: Incorrect. $|\mu_1 - \mu_2|$ tells us the difference between the average weight of both groups, but we cannot tell which group is larger due to the absolute value sign.
- E: Incorrect. Total variation distance is defined as $\frac{1}{2}\sum_{i=1}^{k} \mid a_i - b_i \mid$. This has the same issue as D where we cannot tell which group is larger due to the absolute value sign.
- F: Incorrect. Kolmogorov-Smirnov is a measurement of the maximum absolute difference between two cumulative distribution functions. It does not look at the average, nor does it tell us which weight would be larger.