Zakaria Kortam

Evergreen Valley College
Computer Science 28
Data Science
Professor Estrada
Final Exam

1. Find the name of the pokemon of type Water that has the highest HP.
    a. **water_pokemon = pokemon[pokemon['Type'] == 'Water']**
    b. **name = water_pokemon.loc[water_pokemon['HP'].idxmax(), 'Name']**
2. Find the proportion of pokemon of type Fire In the dataset whose Speedis strictly less than 100.
    a. **fire_pokemon = pokemon[pokemon['Type'] == 'Fire']**
    b. **prop = len(fire_pokemon[fire_pokemon['Speed'] < 100]) / len(fire_pokemon)**
3. Create a table containing Type and Generation that is sorted in decreasing order by the average HP for each pair of Type and Generation
    a. **typeGen = pokemon.groupby(['Type', 'Generation']) .agg({'HP': 'mean'}) .reset_index() .sort_values('HP', ascending=False)**
4. User Return an array that contains ratios of legendary to non-legendary pokemon for each generation.
    a. **count = pokemon.groupby(['Generation', 'Legendary']).size().unstack(fill_value=0)**
    b. **ratios = ((count[True] / count[False]).values**

Histograms

1. Calculate each quantity described below or write Unknown if there is not enough information above to express the quantity as a single number (not a range). Show your work!
    a. **0.035 != ~0.055. False**
    b. **Can only be found if averaged, and not reliably due to scaling.**
    c. **~0.010**
    d. **0.096**
2. If the PLU 4225 histogram were redrawn, replacing the three bins from 0-1000, 1000-2000, 2000-3000 with one bin from 0-to-3000, what would be the height of its bar?
    a. **About 0.064**
3. A fair coin is tossed five times. Two possible sequences of results are HTHTH and HTHHH. Which sequence of results is more likely? Explain your answer and calculate the probability that each sequence appears
    a. **Probability is independent of outcome. Neither is more likely. Each sequence has a 1/32 chance in occurring.**
4. For questions 2 - 4, assume we have a biased coin such that the probability of getting heads is ⅕ and the probability of getting tails is 4/5. The coin is tossed 3 times. What is the probability that you get exactly 2 heads?
    a. **(⅕)*(⅕)*(⅘) = 4/125 chance**
5. Once again, we toss the same biased coin 3 times. What is the probability I get no heads?
    a. **(⅘)^3 -> 64/125**
    b. **1-(64/125) -> 0.488**

6. Assuming that Achilles' coin is fair, write a function called one_walk that simulates one random walk of 100 time steps and returns how far from the origin Achilles ends up at the end of his walk. You may assume that Achilles always starts from the origin.
    a. **def one_walk():**
    b. **position = 0**
    c. **for x in range(100):**
    d. **flip = random.choice(['H', 'T'])**
    e. **position += 1 if flip == 'H' else -1**
    f. **return position**
7. Assuming that Achilles' coin is fair, we would like to simulate what would happen if Achilles took 10000 different random walks. Complete the simulation below and keep track of how far Achilles ends up from the origin in each of his walks in an array called distances. The histogram shown below is an example of a histogram plotted from distances
    a. **distances = make_array()**
    b.
    c. **for i in np.arange(10000):**
    d. **new_distance = one_walk()**
    e. **distances = np.append(distances, new_distance)**
8. Achilles goes for a walk and claims that at the end of his walk, he ended up 30 steps away from the origin. You notice this is strange, so you want to run a hypothesis test to test whether or not Achilles used a fair coin. Fill in the blanks below for the null and alternative hypotheses as well as a good test statistic for this experiment
    a. **Null Hypothesis: He used a fair/even coin.**
    b. **Alternate Hypothesis: The coin was not fair.**
    c. **Test statistic: Perhaps the distance from the start point after 100 or so flips.**
9. **p_value = np.mean(np.abs(distances) >= test_statistic)**
10. True or False
    a. **False, the death increase could have been from anything, not specifically public health.**
    b. **False, an A/B test is for experimental interventions, not for comparing existing salary data.**
    c. **True**
    d. **True**
    e. **False, It's suggestive, but not certain.**
    f. **False, the histogram will converge to the probability distribution, not the other way around.**
11. For each of the alternative hypotheses listed below, determine whether or not the test statistic is valid
    a. **Valid**
    b. **Valid**
    c. **Valid**
    d. **Valid**
    e. **Not Valid**
12. Write a function called compute_pvalue that, given an empirical distribution in the form of an array and the observed value of your test statistic, calculates the p-value for that test statistic.

You may assume that large values of your test statistic provide evidence against the null hypothesis.

a. **def compute_p_value(empirical_dist, observed_ts):**
b. **   p_value = np.mean(empirical_dist >= observed_ts)**
c. **   return p_value**

13. Now write a function called is_significant that takes in an empirical distribution, the observed test statistic and a p-value cutoff, returns True if the p-value of the observed test statistic is statistically significant based on the cutoff provided and False otherwise
    a. **def is_significant(empirical_dist, observed_ts, cutoff):**
    b. **   p_value = compute_p_value(empirical_dist, observed_ts)**
    c. **   return p_value < cutoff**
14. Chloe is suspicious about this distribution. After all, Velveeta is much cheaper to use than Gruyère, and she has also never bought a box that uses Gruyère. Chloe decides to buy many boxes throughout the next month and tracks the type of cheese used in each box. She uses this to conduct a hypothesis test
    a. **Null Hypothesis: The cheeses are used in Trader Joe's frozen mac n cheese according to the given probabilities**
    b. **Alternative Hypothesis: The cheeses are not used according to the given probabilities.**
    c. **observed_stat = sum(abs(observed_proportions - employee_proportions)) / 2**
    d. **def one_simulated_test_stat():**
    e. **   sample_prop = sample_proportions(20, employee_proportions)**
    f. **   return sum(abs(sample_prop - employee_proportions)) / 2**
    g. **p_value = np.count_nonzero(simulated_stats >= observed_stat) / len(simulated_stats)**
    h. **True: Using a 10% p-value cutoff, the null hypothesis should be rejected. Everything else is False.**
15. Choose True/False for each of the statements below, and explain your answer.
    a. True.
    b. False, that's not what permutation is.
    c. False, it's not limited to total variation distance.
    d. Null Hypothesis: Curry and Thompson both have similar field goal percentages.
    e. Alternate: They have very different field goal percentages
    f. Test stat: The difference in their field goal percentages
    g. I'd use the PT to shuffle, split, and calculate the test statistic within each group numerous times and then find the p value.

Feedback

1.What did you like best about this course?

I really appreciated the clear structure of the course. All the necessary materials were conveniently located in the modules, which made it straightforward to dive into the weekly labs and homework on Jupyter. The modules themselves were well-organized, and the content was engaging, offering a good mix of statistical concepts and practical software skills. It was particularly beneficial to learn about using software tools for data manipulation and visualization, as well as gaining exposure to important libraries like numpy and pandas. Another aspect that added to the relaxed atmosphere of the course was the absence of exams, albeit I understand that that will likely change considering that we were only the first iteration of the course. Regardless, the absence of exams allowed for more focus on learning and less on cramming information for the sake of exams. If exams are considered for future iterations of the course, I believe having them open-note, especially for programming sections, would be beneficial. This would support the course's goal of integrating coding with data and mathematics, giving students the freedom to use various resources to enhance their understanding. It's also more reflective of the real world where you really won't be doing everything off the top of your head, but rather, utilizing the best tools possible to make your work as efficient as possible.

How can this course, including the midterm, be improved (thoughtful answers will receive extra credit)?

To start, there does have to be some more refinement in the homework and labs themselves. I'm not sure as to how much customization access that we have, however, there are questions, for instance, where they'd give you questions such as "Consult with your partner" or "team," which doesn't make much sense, rough edges like that. Additionally, I spoke with you previously regarding some issues with the auto-grader where it would mark me off if I didn't capitalize a certain letter or if I wrote something in a different format from what it might be inspecting. This is just a matter for allowing more flexibility in the condition check. Another thing, especially for projects, and this applies to all CS courses, I think that it would be very useful to, at least, to a basic introductory extent, teach students how to use relevant parts of Git/Github. This will be incredibly useful for projects and project collaboration in async courses, because trying to organize all of the code can be a nightmare. It would also be useful if the group projects within this course were split into pre-defined portions that are assigned to individuals, where your portion would form like 80% of your project grade and the project as a whole from the other portions would be 20%.Something like that. This would make it easier to ensure that everyone pulls their weight.

Another thing I'd like to note is that make sure to warn students about choosing the correct login provider for Jupyterhub. They must choose Microsoft and not Google, because when I chose Google by accident, I was stuck and there was no way for me to change my provider, so I had to switch to another browser (Microsoft Edge) in order to use the Jupyter hub.

Eventually, after about a month, it asked me to enter my provider again, and then it allowed me to select Microsoft.

Another thing that I was thinking about, perhaps at the end of the course. Perhaps there could be a project that does something more involved with AI? If we could somehow perhaps train a primitive AI model using refined data, I think that would be a really awesome application of the stuff we learned, especially nowadays with the AI craze. Here's an example of a free online Harvard course regarding Data Science in Ml. You can check it out for inspiration if this is something you'd be interested in adding to the course. There are also many other courses like it.

https://www.edx.org/learn/machine-learning/harvard-university-data-science-machine-learning

Thanks a lot, Professor Estrada.