**Q1. If you have any, what are your choices for increasing the comparison between different figures on**

**the same graph?**

**Q2. Can you explain the benefit of compound interest over a higher rate of interest that does not**

**compound after reading this chapter?**

**Q3. What is a histogram, exactly? Name a numpy method for creating such a graph.**

**Q4. If necessary, how do you change the aspect ratios between the X and Y axes?**

**Q5. Compare and contrast the three types of array multiplication between two numpy arrays: dot**

**product, outer product, and regular multiplication of two numpy arrays.**

**Q6. Before you buy a home, which numpy function will you use to measure your monthly mortgage**

**payment?**

**Q7. Can string data be stored in numpy arrays? If so, list at least one restriction that applies to this data.**

**SOLUTIONS**

*1. One option for increasing the comparison between different figures on the same graph is to use a common scale for the y-axis. This ensures that the data is normalized and the differences between the figures can be easily compared. Another option is to use different colors or patterns for each figure on the graph, or to add labels or annotations to the graph to clarify which figure represents what data.*

*2. Compound interest is beneficial because it allows for the interest to be earned not only on the principal amount, but also on the interest that has already been earned. This means that the interest earned in each compounding period is added to the principal amount, resulting in a higher amount of interest being earned over time. In contrast, a higher rate of interest that does not compound will only earn interest on the principal amount, which results in a lower amount of interest being earned over time.*

*3. A histogram is a type of graph that displays the distribution of a dataset. It is commonly used to show the frequency or proportion of data within specified intervals or bins. One numpy method for creating a histogram is* ***numpy.histogram()****, which takes an array of values and returns the counts and bin edges for each bin.*

*4. To change the aspect ratio between the x and y axes in a plot, you can use the* ***aspect*** *parameter in the* ***matplotlib.pyplot*** *function. The* ***aspect*** *parameter accepts a value that specifies the aspect ratio of the plot. For example, to set the aspect ratio to 1:2, you would use* ***plt.gca().set\_aspect(2)****.*

*5. The three types of array multiplication in numpy are the dot product, outer product, and regular multiplication. The dot product multiplies two arrays element-wise and then sums the result, resulting in a single value. The outer product multiplies each element of one array by each element of the other array, resulting in a new array with the shape of the two input arrays combined. Regular multiplication multiplies two arrays element-wise, resulting in a new array with the same shape as the input arrays.*

*6. To measure your monthly mortgage payment, you can use the numpy financial function* ***numpy.pmt()****. This function takes arguments for the interest rate, number of periods, and principal amount, and returns the monthly payment required to pay off the mortgage over the specified number of periods.*

*7. Yes, string data can be stored in numpy arrays. However, there are some restrictions that apply to this data. For example, numpy arrays have a fixed size, so the length of the strings stored in the array must be predetermined. Additionally, operations that are valid for numerical data may not work for string data, so it is important to ensure that the array is being used appropriately.*