D207 CODES

Remember

-How do you prep your data: Talk about removing blanks, Null Values, NA, NANS,

-Remove categorical with high cardinality > 3 to 5 levels, therefore in your answer you say I am going to drop high categorical with high-level cardinality, only categorical with 3 levels of cardinality will be used.

-Consider removing outliers>3 standard deviations, however, check what is the impact on the mean, and median.

-Remember to convert chr and str data to categorical

-Once you convert to categorical then convert to numeric levels. E.g small-o, medium-1, large-2.

Only choose one statistical method either t-test or ANOVA or Chi square

4univariate and 2 bivariate

-Univariate one variable a graph of a categorical variable like a graph to improve understanding.

-Bivariate graphs use the same datatype: cont. v. cont. cat. V. cat

-Scale and title the x-y axis, main title for clarity.

Showing correlation between two variable

# Create a scatterplot of happiness\_score vs. life\_exp and show

sns.scatterplot(x='life\_exp', y='happiness\_score', data=world\_happiness)

# Show plot

plt.show()

# Create scatterplot of happiness\_score vs life\_exp with trendline

sns.lmplot(x='life\_exp', y='happiness\_score', data=world\_happiness, ci=None)

# Show plot

plt.show()

T TEST 04/23/2022

**Two sample mean test statistic**

The hypothesis test for determining if there is a difference between the means of two populations uses a different type of test statistic to the z-scores you saw in Chapter 1. It's called "t", and it can be calculated from three values from each sample using this equation.

t=(x¯child−x¯adult)schild2nchild+sadult2nadult

While trying to determine why some shipments are late, you may wonder if the weight of the shipments that were on time is **less than**the weight of the shipments that were late. The late\_shipmentsdataset has been split into a "yes" group, where late == "Yes" and a "no" group where late == "No". The weight of the shipment is given in the weight\_kilograms variable.

The sample means for the two groups are available as xbar\_no and xbar\_yes. The sample standard deviations are s\_no and s\_yes. The sample sizes are n\_no and n\_yes. numpy is also loaded as np.

* Calculate the numerator of the t test statistic.
* Calculate the denominator of the t test statistic.
* Use those two numbers to calculate the t test statistic.
* Subtract xbar\_yes from xbar\_no.
* Square s\_no and divide by n\_no, and do the same for the "yes" component.
* Add those two numbers together and take the square root.
* Take the ratio of the numerator and the denominator.

# Calculate the numerator of the test statistic

numerator = xbar\_no - xbar\_yes

# Calculate the denominator of the test statistic

denominator = np.sqrt(s\_no \*\* 2 / n\_no + s\_yes \*\* 2 / n\_yes)

# Calculate the test statistic

t\_stat = numerator / denominator

# Print the test statistic

print(t\_stat)

answer will show

Note: t

When testing for differences between means, the test statistic is called 't' rather than 'z', and can be calculated using six numbers from the samples. Here, the value is about -2.39 or 2.39, depending on the order you calculated the numerator.

'z', and can be calculated using six numbers from the samples. Here, the value is about -2.39 or 2.39, depending on the order you calculated the nu