Hyperiondev

TASK

Data Analysis II

Visit our website

Introduction

WELCOME TO THE DATA ANALYSIS II TASK!

In this task, we are going to work on how to handle missing data as well as Data Normalisation. Data can have missing values for a number of reasons. For example, an observation (a piece of data) may not have been collected or recorded, or perhaps data corruption may have occurred. Data corruption is when data becomes unusable, unreadable or in some other way inaccessible to a user or application.

Get in touch

Connect for support

Remember that with our courses, you're not alone! You can contact an expert code reviewer to get support on any aspect of your course.

The best way to get help is to login to Discord at **https://discord.com/invite/hyperdev** where our specialist team is ready to support you.

Our team is happy to offer you support that is tailored to your individual career or education needs. Do not hesitate to ask a question or for additional support!

INTRODUCTION TO VARIABLES

In the field of Data Science, we tend to work with two main categories of variables: categorical and continuous variables.

Categorical and Continuous Variables

Categorical variables are also known as discrete or qualitative variables. Examples of categorical variables are race, sex, age group, and educational level. According to Laerd (n.d.), these variables can be further divided into the following categories: *nominal*, *ordinal* or *dichotomous*.

- **Nominal variables** have two or more categories, which do not have a specific/predefined order. For example, properties could be classified as houses, condos or bungalows. Therefore, the variable that holds the property type is a nominal variable. Which State in the USA a person lives in would also be a nominal variable.

- **Dichotomous variables** are nominal variables, which have only two categories or levels. For example, we could use a dichotomous variable to describe whether a person is a pensioner or not. In this case, the categories would be "True" or "False".

- **Ordinal variables** are nominal variables, but the categories can be ordered or ranked. For example, if you were asked to rate your satisfaction with this course, your responses could be "Completely satisfied", "Mostly satisfied", "A little dissatisfied" or "Very dissatisfied".

A continuous variable is one which can take on infinitely many, uncountable numerical values, both integers and floating points. They are also known as quantitative variables. Continuous variables can be further categorised as either interval or ratio variables:

- **Interval variables** are "variables for which their central characteristic is that they can be measured along a continuum and they have a numerical value (for example, temperature measured in degrees Celsius or Fahrenheit)" (Laerd, n.d.).

- **Ratio variables** are interval variables, but a 0 value means that there is none of that particular variable. Therefore, "weight" would be a ratio variable because something that weighs 0 kgs has no weight. Temperature

(measured in degrees Celsius) would not be a ratio variable because 0 C does not mean that there is no temperature.

WORKING WITH CATEGORICAL DATA

As discussed previously, categorical data deals with discrete (individually separate and distinct) data that fits neatly into a number of categories. To analyse this type of data, data tables and visualisations are often used.

Data tables are often used to count the number of variables that fall within a certain category. For example, you may want to analyse a data set that stores data about HyperionDev students. The data set could contain categorical data such as which Bootcamp a student is currently registered for. Therefore, it could be useful to create a data table that displays the total number of students registered for each Bootcamp to help analyse this data set.

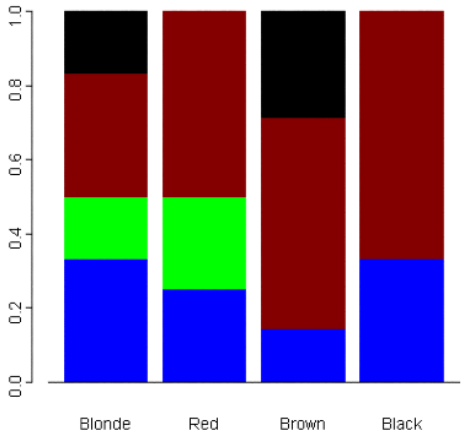
Two-way tables are useful when trying to analyse how data in two categorical variables relate. Consider this example by Lacey: "suppose a survey was conducted of a group of 20 individuals, who were asked to identify their hair and eye colour. A two-way table presenting the results might appear as follows":

Hair Colour	**Eye Colour**				
	Blue	*Green*	*Brown*	*Black*	**Total**
Blonde	2	1	2	1	**6**
Red	1	1	2	0	**4**
Brown	1	0	4	2	**7**
Black	1	0	2	0	**3**
Total	**5**	**2**	**10**	**3**	**20**

(Lacey, n.d)

The totals for each category are known as marginal distributions. To make this data easier to analyse, two-way tables are often converted to percentages. For example, it may be more helpful to know what percentage of people surveyed have blue eyes or what percentage of people with red hair have green eyes.

Segmented bar graphs are also often useful for analysing categorical data. For example, notice how the segmented bar graph below can be used to represent the information about people's eye colour (colour coded) and hair colour:



(Lacey, n.d)

Often it is useful to introduce a categorical variable into a data set to make continuous variables easier to analyse. For example, if you had a data set that stores the weight for a number of boxers (continuous variable) it may be useful to introduce a weight range variable (e.g. minimumweight, light flyweight, flyweight, etc.).

INTRODUCTION TO MISSING VARIABLES

As you have already learned, before you can do any useful analysis, it is important to clean your data set. One problem that you will often encounter when cleaning data is missing data. There is no perfect way of dealing with missing data! The approach you use will depend on a good understanding of the data and of the effect that the approach you use will have on the data set. The common approaches to handling missing data are trying to find missing data, removing observations with missing data or using an imputation method to replace missing data with a substitute value. However, before deciding how to deal with missing data, it is important to understand why the data is missing.
Missing data can be categorised into three types:

- **Missing Completely at Random (MCAR)**: This means that the data that is missing is **not in any way related** to any other data in the data set.
    - For example, data may be missing because a test paper was lost, or a blood sample was damaged. This means that there is no way to

predict what was missing (and therefore no reasonable way to guess what those values are). This is entirely random.

- **Missing at Random (MAR)**: This is when the data that is missing is in some way related to other data in the data set. If data is MAR, it can sometimes be predicted based on other data in the data set. In other words, data that is MAR **can be explained by other data measured in the dataset**. This is due to implicit biases in the data itself. These biases may be used partially to try and gain a more realistic statistic.
    - For example, consider a survey on depression. Studies have shown that males are less likely to fill in surveys about depression severity. In other words, this missingness can be determined by their gender (which was noted in the dataset).
- **Missing Not at Random (MNAR)**: This occurs when there is a direct relation to some missing data that hasn't been measured by the researchers. Like with MAR, there are certain biases that affect the values in the dataset. However, these biases are from **factors not measured in the dataset.**
    - An example of this is in COVID reporting. Since restrictions were lifted and it had a lower impact on our lives, we saw a drop in the reported COVID cases. This is not because lifting restrictions makes the disease go away - it is just that people are less likely to get tested! This is something that cannot be measured by the scientists, and therefore can't be imputed.

Whether we remove observations with missing data or substitute the missing value with another value, we have to be very careful not to create bias! **Rumsey** defines statistical bias as the "systematic favouritism that is present in the data collection process, resulting in lopsided, misleading results." When we remove data or add substituted data we could cause bias by creating a data set that favours a certain idea. For example, if you remove mainly data about people with low income or if you assume that all those who have a missing value for income are high income, your findings could be distorted.

INTRODUCTION TO SCALING AND NORMALISATION

Another common problem that we encounter when trying to analyse data is having different units of measurement for a particular variable. For example, if you wanted to compare housing prices in different countries, you would probably have different data sets that store price values with different currencies. Clearly, it is not easy to compare these values. To get rid of the unit of measurement, we either 'normalise' or 'standardise' the data. In this section, we briefly consider some methods of doing this.

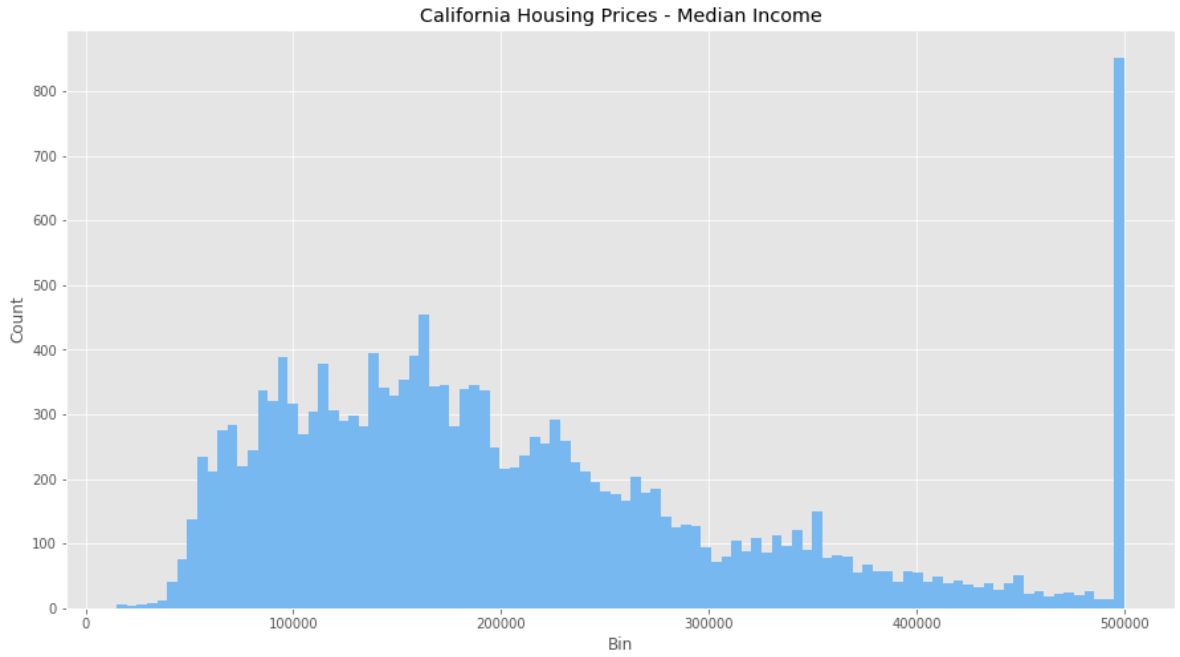
Normalisation

Normalisation usually involves scaling a variable to have a value between 0 and 1. Scaling is often done using the formula below:

$$z = \frac{x \text{ - } \min(x)}{\max(x) \text{ - } \min(x)}$$

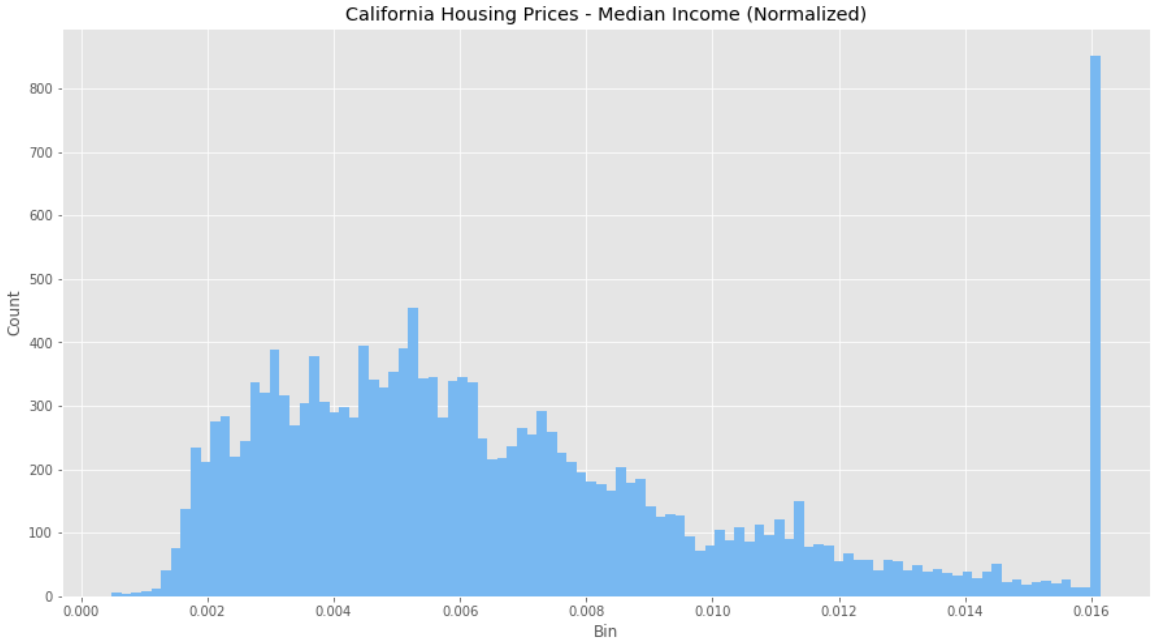
This helps to eliminate units of measurement for data. For example, if you are working with a dataset with variables containing different units of measures, such as kilometres, miles etc., using scaling reduces all the variables to a scale of 0 to 1 removing the need for units of measurement. This helps by allowing you to more easily compare data from different sources. For example, if you wanted to compare housing prices in America vs housing prices in Germany, you would have to change your data so that the prices are measured using the same scale.

To understand scaling, look at the visualisation related to housing prices in California below. The range of income is between 0 and 500,000. We also have a large number of people in the category of income at 500,000, so the dataset probably puts anyone that earns more than 500,000 into that bracket.



(DeFilippi, 2018)

The graph below shows the data after scaling. All the values are now between 0 and 1, and the outliers are still visible within our scaled data. **The data retains the same structure.**



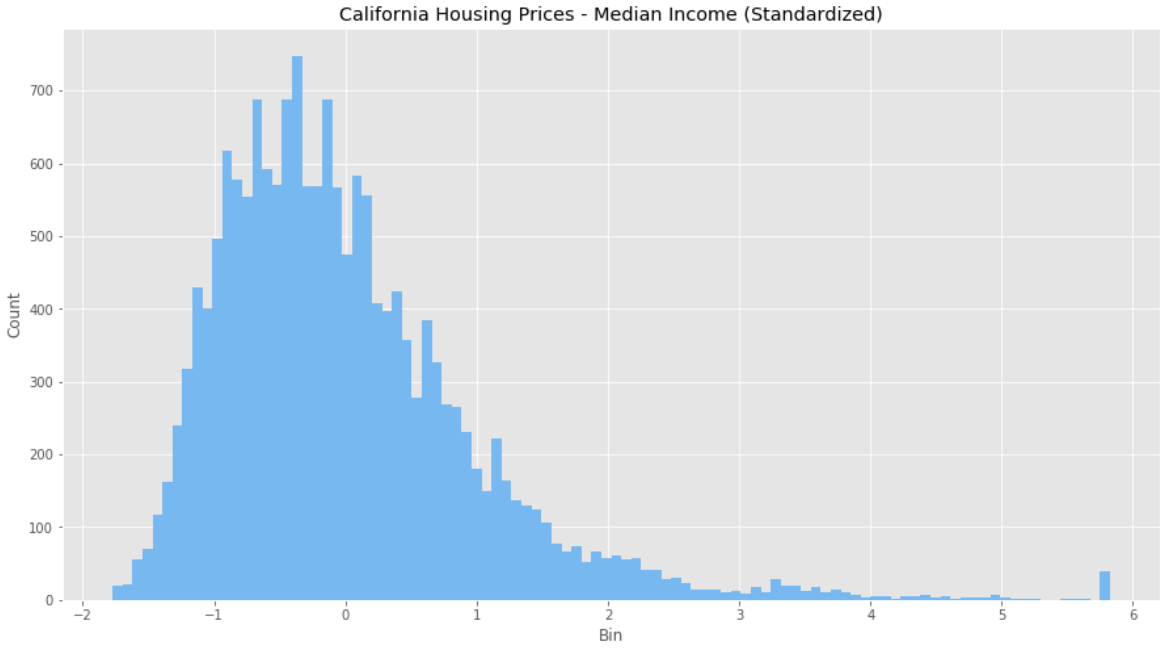
(DeFilippi, 2018)

Standardisation

Standardisation is when we transform data to have a mean (average) of zero and a standard deviation (the average deviation from the mean in distribution) of 1. Standardisation is done using this formula:

$$z = \frac{x_i - \mu}{\sigma}$$

The image below illustrates the effect of standardising data:



(DeFilippi, 2018)

Normalisation vs. Standardisation

Now we know the difference between normalisation and standardisation. Normalisation preserves the distribution of the data, and standardisation shifts the statistical distribution to follow a normal distribution. But, how do we know when to use which one? Well, it very much depends on where you're using it.

Remember, standardisation is used for Machine Learning (ML). This will be touched on later, but certain ML methods work on certain assumptions in the data. Methods that don't need to make assumptions about the distribution of the data will use normalisation. However, some methods do need the data to follow a normal distribution, so standardisation will be useful here.

When analysing data, normalising is important to ensure that no specific feature has dominance over another feature. For example, when looking at the number of rooms in a house vs. the price of that house, you'll notice that the numbers for house prices are orders of magnitude larger than the number of rooms. To fix this, simply impose a maximum of 1 for both features - this makes graphing and visualising the data easier.

Standardising data is useful when you want to examine specific attributes of a variable with respect to the overall statistical distribution. For example, if you are looking at exam marks, and you want to find out the number of standard deviations a student's mark lies from the mean, you could standardise the data.

Instructions

Before you get started, we strongly suggest that you start using the Jupyter Notebook to open all notebooks (.ipynb). Do not use the normal Windows notepad as it will be much harder to read.

You may run Jupyter Notebook to see the output and follow through the content. Feel free to write and run your own example code before doing the Task to become more comfortable with the concept.

Compulsory Task 1

Follow these steps:

- Open the **Data Analysis II.ipynb** file and within it read in the **Pakistan.csv** file and display the first 5 columns.
    - Get the number of missing data points per column.
    - Look at the number of missing points in the first ten columns. Write a note on the reason you think we have missing data on the three top columns: Islamic date, Holiday type and Time. Remember to classify them according to the three categories (types of missingness) we have considered. For more information regarding types of missingness, see **here**.

Compulsory Task 2

Follow these steps:

This task handles the normalisation and standardisation of variables. For more information about normalisation and standardisation, see **here**. Create a Jupyter Notebook in which you do the following:

- For the following examples, decide whether normalisation or standardisation makes more sense:
    - You want to build a linear regression model to predict someone's grades given how much time they have spent on various activities during a normal school week. You notice that your measurements for

how much time students spend studying aren't normally distributed: some students spend almost no time studying and others study for four or more hours every day. Should you normalise or standardise this variable?

- You're still working on your grades' study, but you want to include information on how students perform on several fitness tests as well. You have information on how many jumping jacks and push-ups each student can complete in a minute. However, you notice that students perform far more jumping jacks than push-ups: the average for the former is 40, and for the latter only 10. Should you normalise or standardise this variable?

- From the countries dataset, scale the "EG.ELC.ACCS.ZS" column. Visualise the scaled data.

- From the countries dataset, standardise the "SP.DYN.CBRT.IN" column. Visualise the data.

If you are having any difficulties, please feel free to contact our specialist team **on Discord** for support.

Completed the task(s)?

Ask an expert to review your work!

Review work

Rate us

Share your thoughts

HyperionDev strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think that the content of this task, or this course as a whole, can be improved? Do you think we've done a good job?

Click here to share your thoughts anonymously.

REFERENCES

DeFilippi, R. R. (2018, April 29). Standardize or Normalize? Examples in Python. Retrieved April 13, 2019, from Medium.com: **https://medium.com/@rrfd/standardize-or-normalize-examples-in-python-e3f174b65dfc**

Lacey, M. (n.d.). Categorical data. Retrieved April 30, 2019, from stats.yale.edu: **http://www.stat.yale.edu/Courses/1997-98/101/catdat.htm**

Laerd statistics. (n.d.). Types of variables. Retrieved April 30, 2019, from statistics.laerd.com: **https://statistics.laerd.com/statistical-guides/types-of-variable.php**

Kwak, S. K., & Kim, J. H. (2017). Statistical data preparation: management of missing values and outliers. Korean journal of anesthesiology, 70(4), 407–411. doi:10.4097/kjae.2017.70.4.407

Rumsey, D. (2020). How to Identify Statistical Bias - dummies. Retrieved 25 August 2020, from **https://www.dummies.com/education/math/statistics/how-to-identify-statistical-bias/**

Swalin, A. (2018, January 31). How to Handle Missing Data. Retrieved May 13, 2019, from Towards Data Science: **https://towardsdatascience.com/how-to-handle-missing-data-8646b18db0d4**