Two-attribute and Single-attribute Comparisons of Data

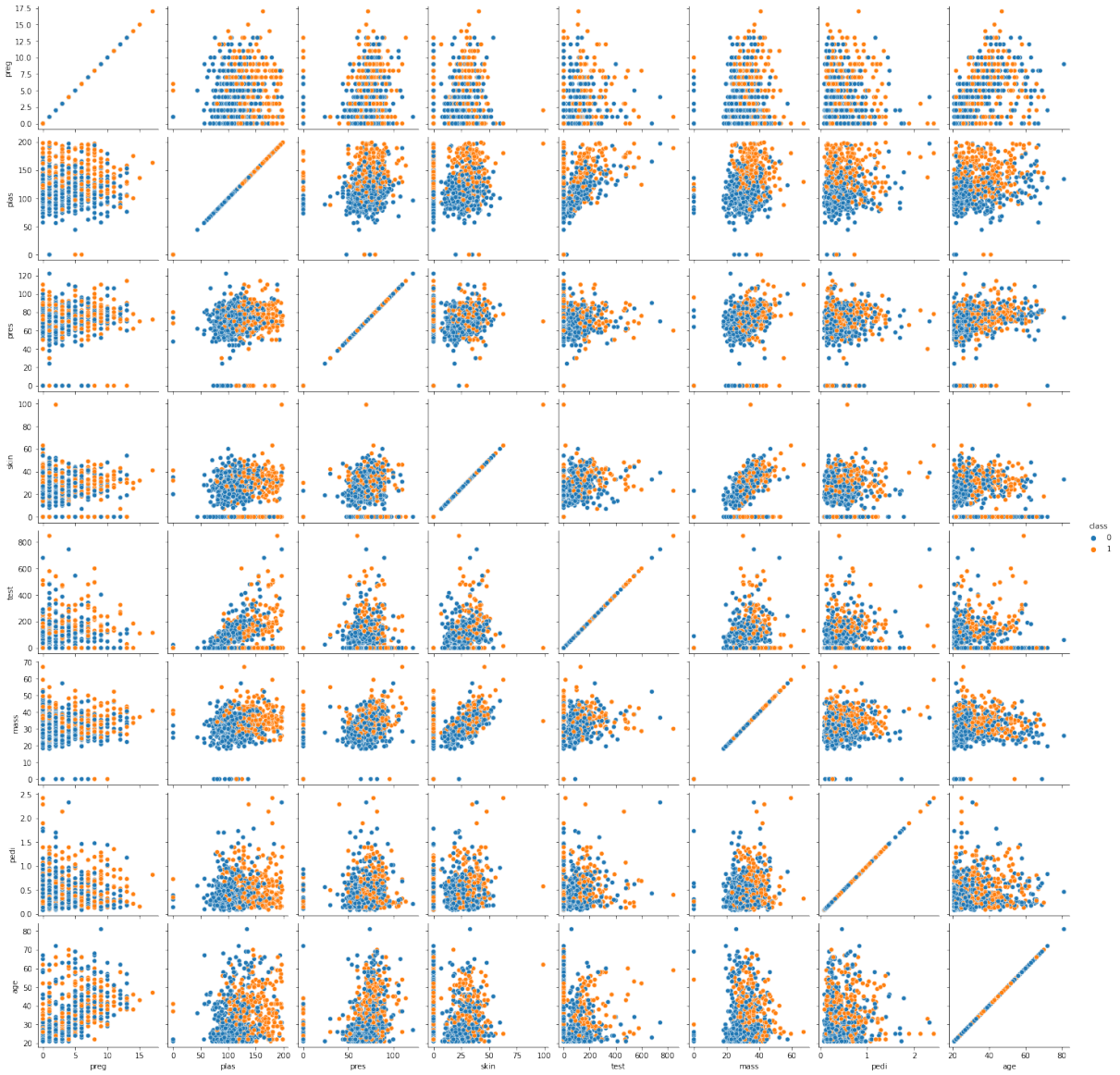

Figure 1: img

We can see in this plot that the diagonal has plots that describe a straight line. This happens because the pair plots are describing the relationship between **all** combinations of attributes, and eventually there is a combination of an attribute against itself. When this happens, both the x axis and y axis coordinates are given by **the same** column. So for example, an instance at attribute 1 has a 3 in x axis and a 3 in y axis, and therefore **when you compare an attribute against itself in a pair plot, a straight line (y = x) is shown**.

```
x = [6, 7, 24, 8, 12]
plt.rcParams["figure.figsize"] = [5,5]
plt.plot(x, x);
```

x	y
6	6
7	7
24	24
8	8
12	12

we would be getting the following.

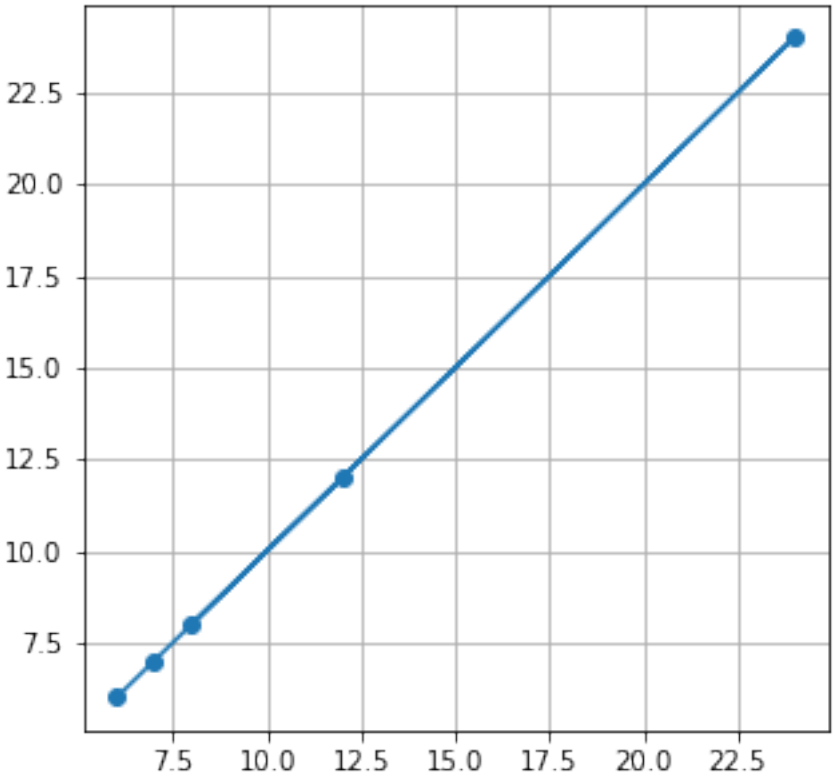

Figure 2: img

By default, the format that seaborn takes as diagonal is the **density** of the data, which is generated if

we run:

```
1  sea.pairplot(data, hue='class')
```

Figure 3: img

Whereas we put:

```
1  sea.pairplot(data, hue='class', diag_kind=None)
```

And we would get the image where the diagonals are y = x lines as we talked before. We can even run:

```
1 sea.pairplot(data, hue='class', diag_kind='hist')
```

And we would be seeing the diagonal show the separate histograms of the data.

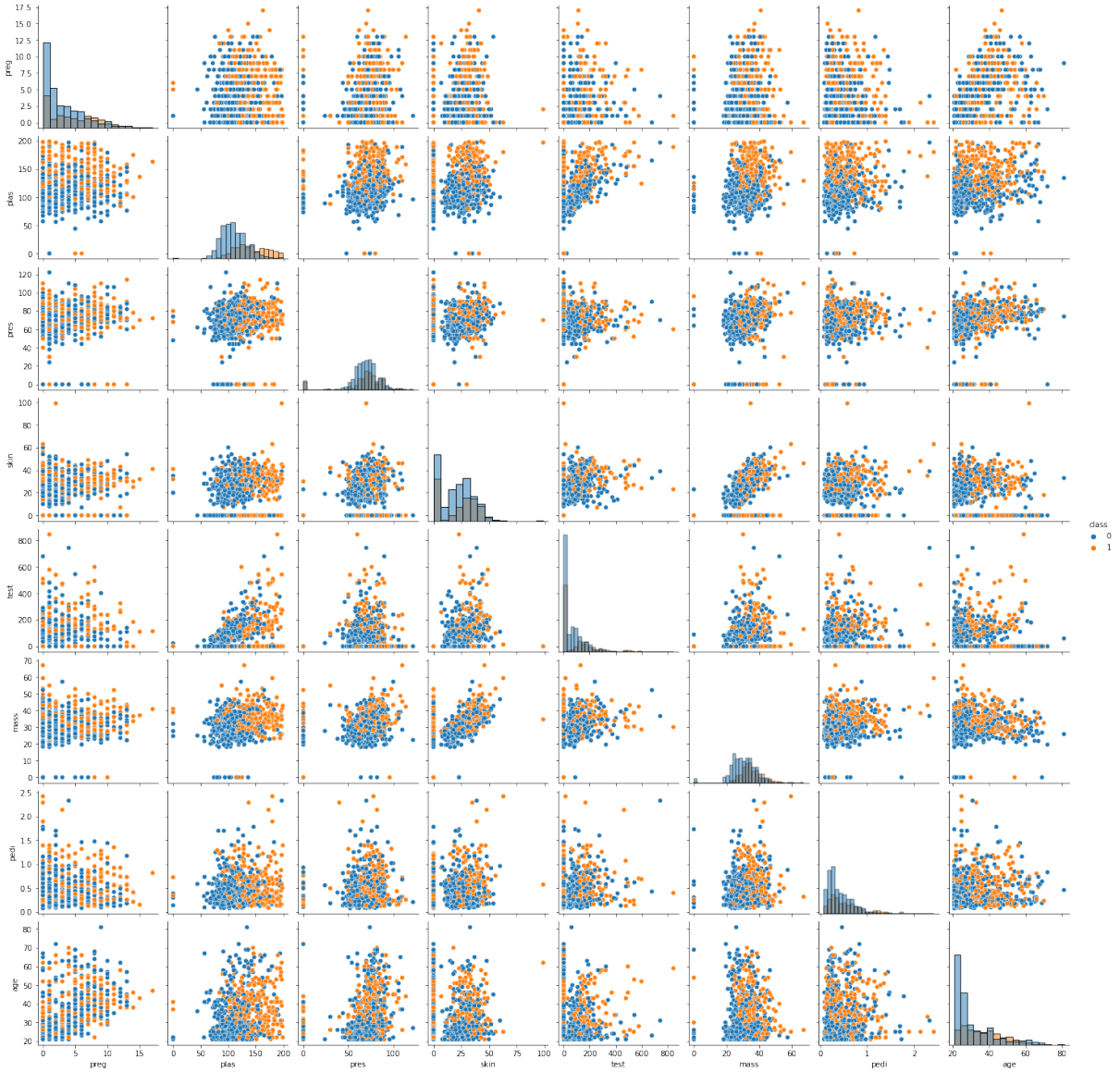

Figure 4: img

In general, we are seeing two coloured scatter plots because they are being divided into the corresponding value of the **class** column. That is, the paired data point that corresponds to a 0 is painted in one color and the ones who correspond to 1 in another color.

age	preg	…	class
25	2	…	0
34	4	…	1

What are we expecting? What would be the ideal model between two attributes, so that such comparison between those two attributes can have a better prediction model?

- Both point clouds (in respect to color) are completely separated. Ideally, this would look:

column 1	column 2	class
1	85	1
2	90	1
3	92	1
4	100	0
7	138	0
8	148	0

Range col 1: 1 - 8

Range col 2: 85 - 148

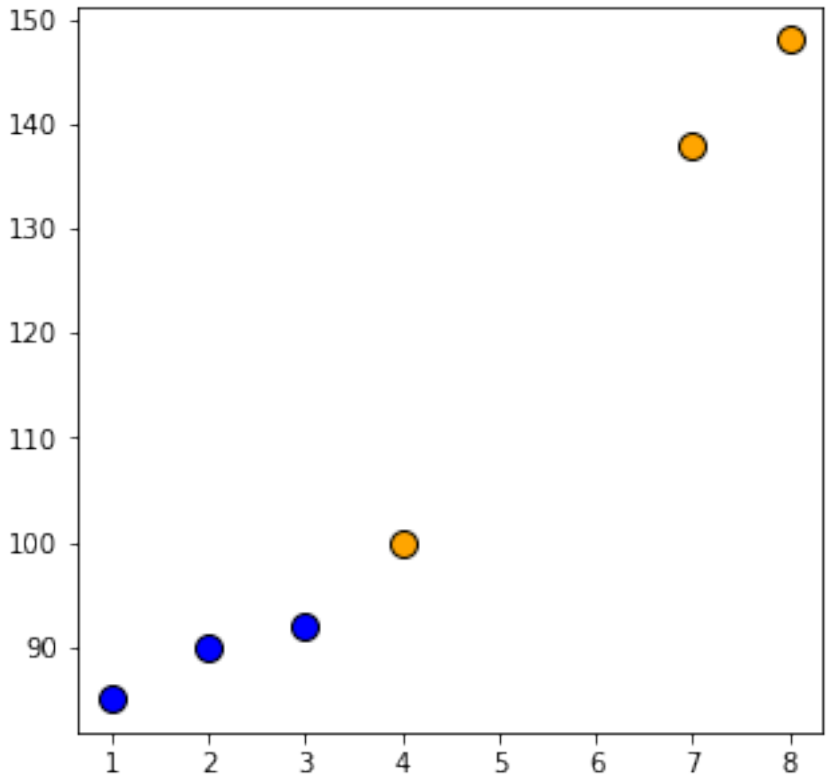

Figure 5: img

- This would be ideal because in the case of **new data**, if this new point lies in the blue area (considering both x's and y's), and if we are using a classification model, we would certainly know to which class it belongs to (0 or 1, in this case).
- This is ideal only in the **classification or clustering problems**, but if it is a **regression** problem, it would still be accepted if both clouds of points are overlapped, because the linear model is not affected if they are used separate.
- In the original pair plot, we are seeing that all of them are quite overlapped and lack this ideal separation.

Iris Database

The first we will do is compare the boxplots of each attribute with respect to its class (its corresponding **class** column value). We will use seaborn to plot.

In the code cells, when it says:

```
1 plt.subplot(4, 2, 1)
```

we are referring to a grid such as:

Figure 6: img

We are plotting the box plot of each class that is in each attribute. In this case we start with the first column (Sepal.length) and with respect to its class we will do the box plot.

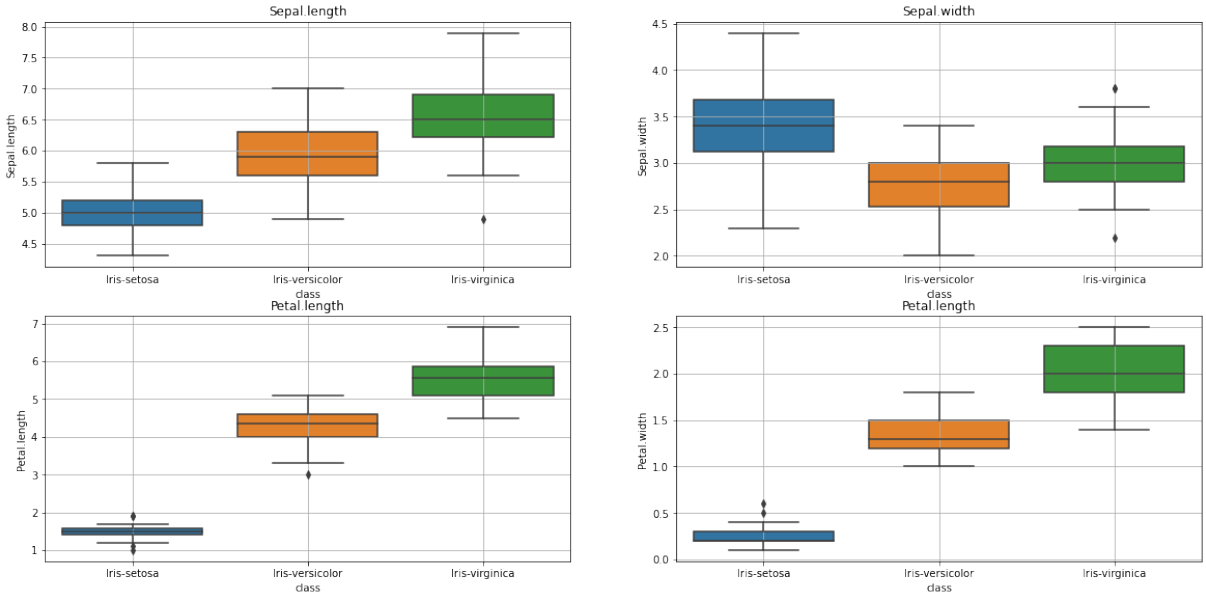

Figure 7: img

What can we say about these plots?

- They are showing the variability of each attribute with respect to its class.

- It is a problem of **classification** (the other option was Regression, so not possible). There are three groups:
  - Iris-setosa
  - Iris-versicolor
  - Iris-virginica
- Those are the three types of flowers. Thus, it is a classification problem with 3 classes and 4 attributes. The 4 attributes are:
  1) sepal length
  2) sepal width
  3) petal length
  4) petal width

Plot 4,2,1

- Each plot corresponds to one attribute/column.
- There are three box plots, where each color corresponds to one class (3 in total). Therefore, we are separating from each column the classes that appear in those column instances. For example,

sepal length	class
4.5	setosa
…	setosa
5.8	setosa
4.9	versicolor
…	versicolor
7.0	versicolor
5.6	virginica
…	virginica
8.0	virginica

Where the first three instances of the two columns generate the first box plot, and so on.

- With respect to the range of data that we have in the table, it will be to which class it corresponds. The interesting part of this, is to determine which attribute is better for a problem of classification.
- On the blue box plot, we can see that the **almost all density** of data lies on the box, from 4.7 - 5.3. If I have a new data an it lies on 5.5, it is **most probable** that it belongs to the orange one, because 5.5 is closer to the orange's density (box). If it was 6.0, it would **exactly** be the orange, because there are no values of the other two classes that tell me that the new data could be of those other two.

Plot 4,2,3

- What happens if there is a new value of 2.5? because 2.5 lies in an empty space (it is outside of any box plot), we should analyze another characteristic/column. By itself, one attribute gives us a percentage or probability of which class the data belongs to. There will be some ranges, in which we cannot predict to which class it belongs to (2.5, for example).
- Thus, ideally what would be a model for this problem? Boxes arranged like below:

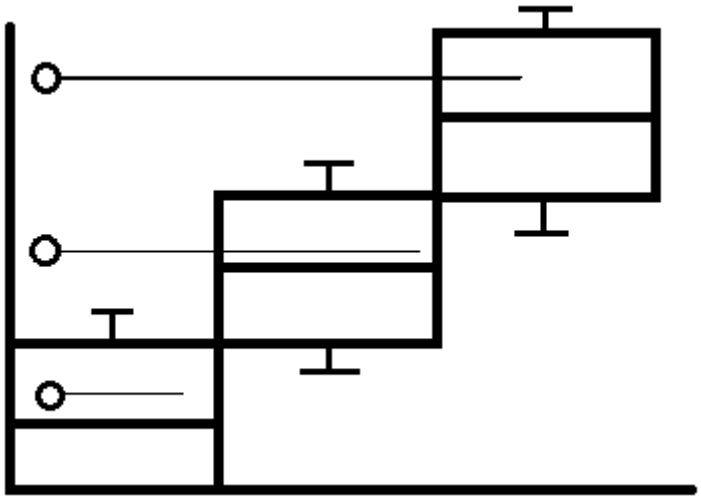

Figure 8: img

Note: we could have the whiskers anywhere, but the density (box) is the important. They would ideally be like above.

- With this, if there is a data in the range of density one, we know that it belongs to class 1, and so on.
- In which way we would not be able to predict?

  - Boxes too separate leaving black spaces. But even in this plot and 4,2,4 if a data point is 2.5 it is more **probable** that it belongs to setosa, as setosa owns most of the lower values. It gives you a probability because you are closer to being an atypical value of the setosa class than others. But still, it is advisable that you look into other attributes.
  - Boxes are exactly in the same range. This would not be a characteristic that can predict to which class it belongs to. In data cleaning, maybe this could be a column that we could delete.

- Thus, **the relationship between attributes, can help to have a better prediction**. We saw the pair plots to see if we could achieve a good prediction (in classificiation problems) taking 2 columns, or the box plots to see if we could achieve a good prediction (in classificiation problems) taking 1 column.

Plots 4,2,3 and 4,2,4

- Petal length and petal width are **highly correlated** and this can be seen because if we put plot 3 over plot 4, their boxes are almost exactly at the **same positions**. This also tells us that maybe we can erase one of the two columns, because a new data point that falls into one box of one plot (3), the data point will also fall on the same box but of the other plot (4).