Lab 7 - Normality and Sample Properties 2

Nick Sumpter (Edited by Eddie-Williams Owiredu & Guy Twa)

2023-09-11

Today's Lab

Today, we will be extending last lab to testing normality across multiple groups, and determining methods for correction of non-normal data. We want to emphasize how important these type of tests are for parametric tests. **You will need to do this in your exams.**.

Loading Packages and Data

The dataset for today is called ABI and is stored in the `ABI.RData` file. This dataset comes from the Allen Brain Institute and describes various protein concentrations in post-mortem brain tissue from two sites: the Hippocampus (HIP) and Parietal Cortex (PCx). We will use the `car`, `pastecs` and `tidyverse` packages for our analyses today. Also we should load in the new functions as we did last lab.

```
library(car)
library(pastecs)
library(tidyverse)

setwd("~/Documents/GRD770/Lab 7 - Normality and Sample Properties 2")

load("ABI.RData")

source("../functions/functions.R")
```

Getting to Know Your Data

Have a look at the ABI dataset in your Environment. As you can see, there are 175 observations (rows) of 5 variables (columns):

1. `ID` = a factor describing which donor the sample came from (note there are 101 levels, but 175 rows, so there are some repeated IDs)
2. `struct` = a factor describing which tissue the sample came from (2 levels = HIP and PCx)
3. `mip_1a` = a numeric vector describing the concentration of macrophage inflammatory protein 1 alpha in pg per mg

4. `tau` = a numeric vector describing the concentration of non-phosphorylated tau protein in pg per mg
5. `ptau` = a numeric vector describing the concentration of phosphorylated tau protein in pg per mg

For a quick look at the distribution of each variable, we can run the `summary` command over the whole dataset:

```
summary(ABI)
```

```
##           ID      struct       mip_1a            tau              ptau          
##  H14.09.001:  2   HIP:89   Min.   :  0.00   Min.   : 724.1   Min.   :  16.89  
##  H14.09.002:  2   PCx:86   1st Qu.:  9.53   1st Qu.:1078.7   1st Qu.: 963.24  
##  H14.09.004:  2            Median : 15.84   Median :1148.3   Median :1516.30  
##  H14.09.006:  2            Mean   : 31.79   Mean   :1200.3   Mean   :2073.20  
##  H14.09.007:  2            3rd Qu.: 24.59   3rd Qu.:1347.9   3rd Qu.:2959.95  
##  H14.09.009:  2            Max.   :789.64   Max.   :1793.0   Max.   :6443.01  
##  (Other)   :163            NA's   :20       NA's   :17       NA's   :17
```

This immediately tells us that there are 89 HIP and 86 PCx samples, and describes the distribution of each of the three proteins. Pay careful attention to the presence of `NA` in the three protein columns. This is a common occurrence and represents missing data. We have two options:

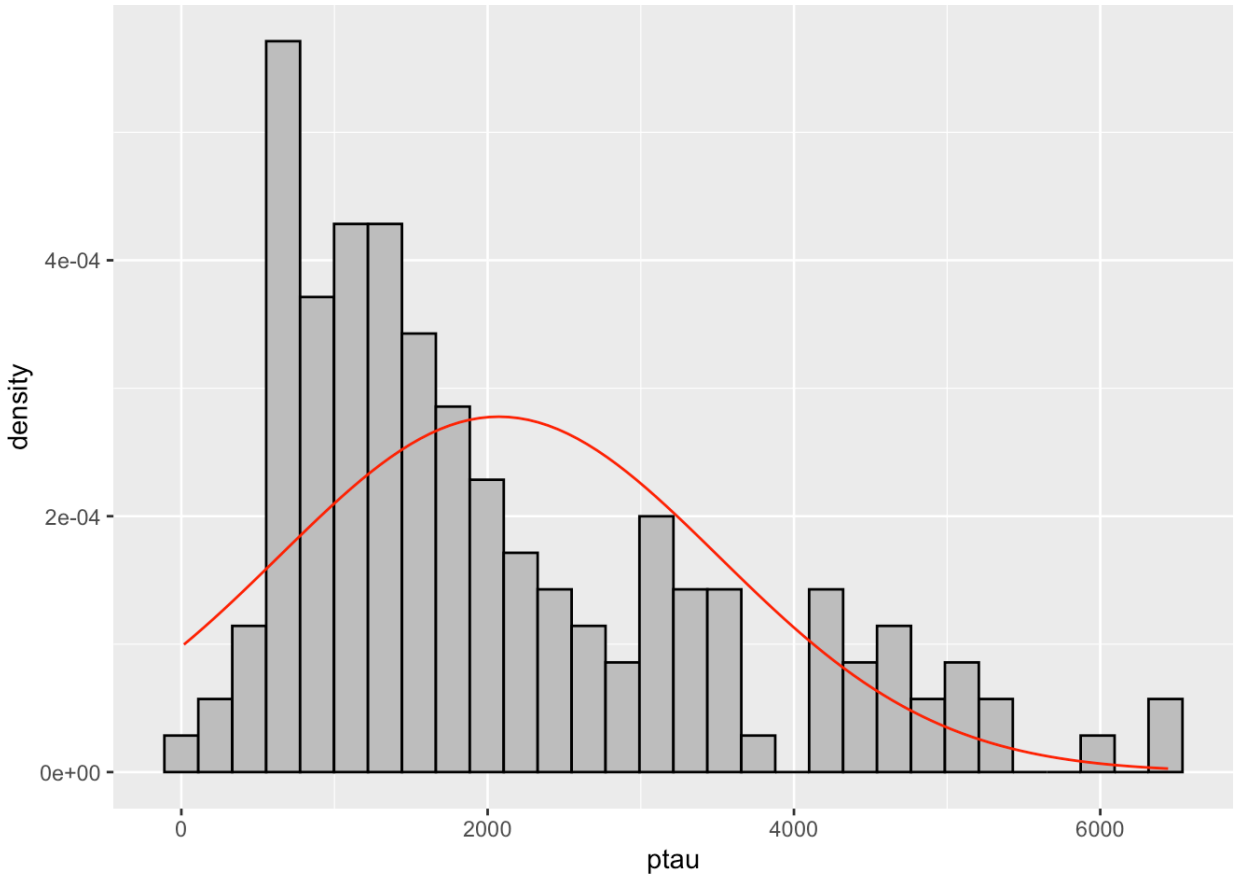
1. Remove all rows with an `NA` in any column (more conservative)
2. Tell our functions to ignore these rows where appropriate (better, as even if an individual is missing `mip_1a` for example, we may still be interested in their measurement for `tau`)

Testing Normality of the Phosporylated Tau Variable Within Each Tissue

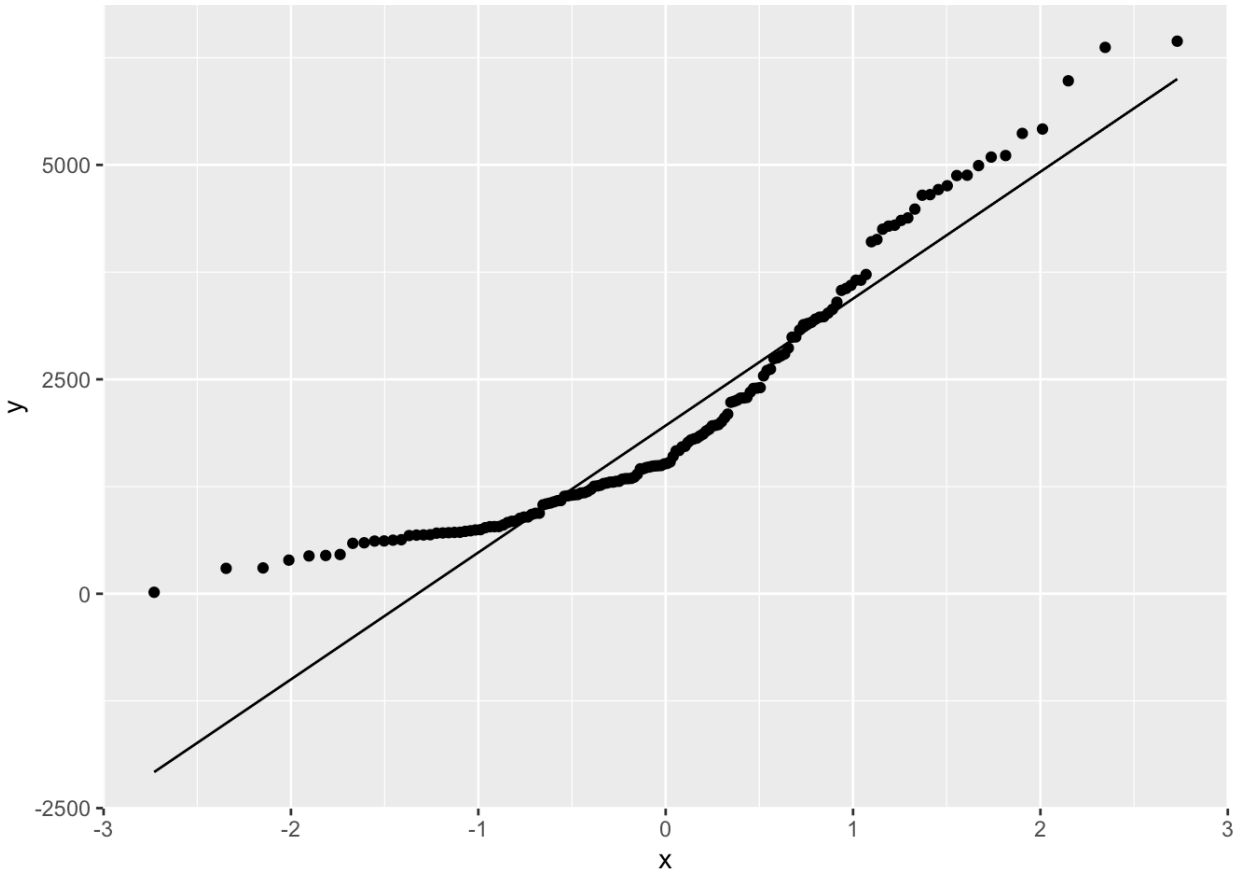
Graphical Normality tests

Let's start with assessing normality graphically for the `ptau` variable across all tissues. Note that for the `stat_function` function, we need to set the `na.rm` argument to `TRUE` in both the `mean` and `sd` functions so they ignore `NA`s.

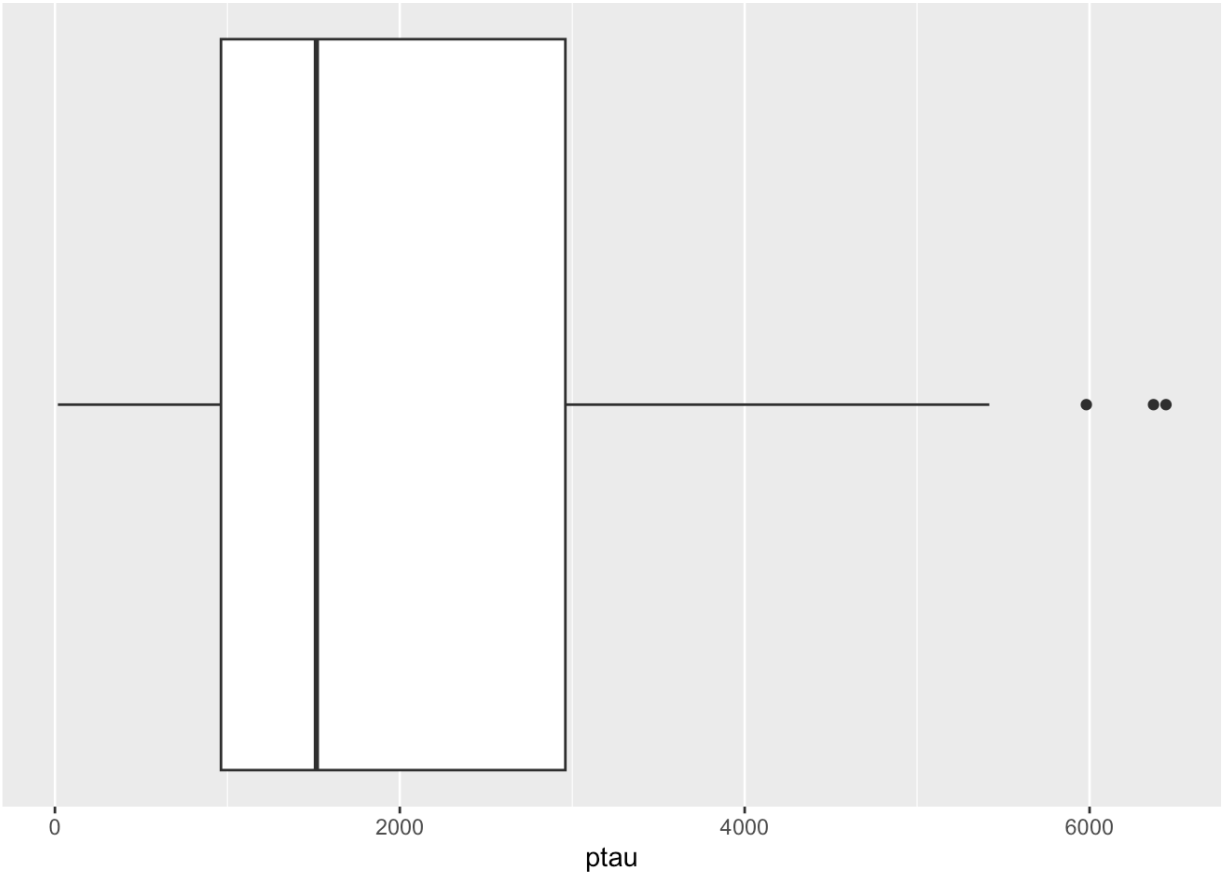
```
# Histogram
ggplot(data = ABI, mapping = aes(x = ptau)) +
  geom_histogram(mapping = aes(y = ..density..), bins = 30, fill = 'gray', color = 'black') +
  stat_function(fun = dnorm, args = list(mean = mean(ABI$ptau, na.rm = TRUE), sd = sd(ABI$ptau,
na.rm = TRUE)), color = 'red')
```




```
# Q-Q plot
ggplot(data = ABI, mapping = aes(sample = ptau)) +
  geom_qq() +
  geom_qq_line()
```



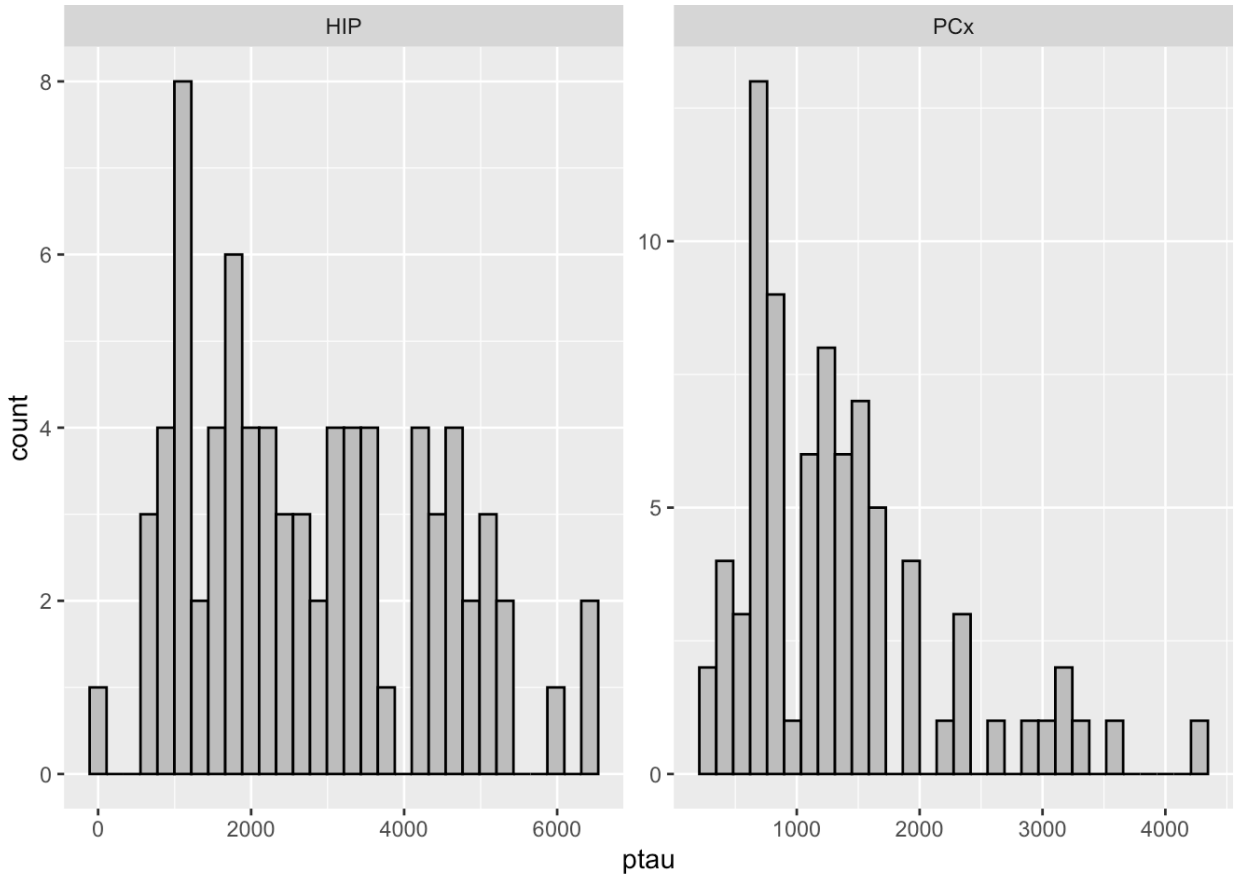

```
# Boxplot
ggplot(data = ABI, mapping = aes(x = ptau)) +
  geom_boxplot() +
  theme(axis.ticks.y = element_blank(),
        axis.text.y = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor.y = element_blank())
```



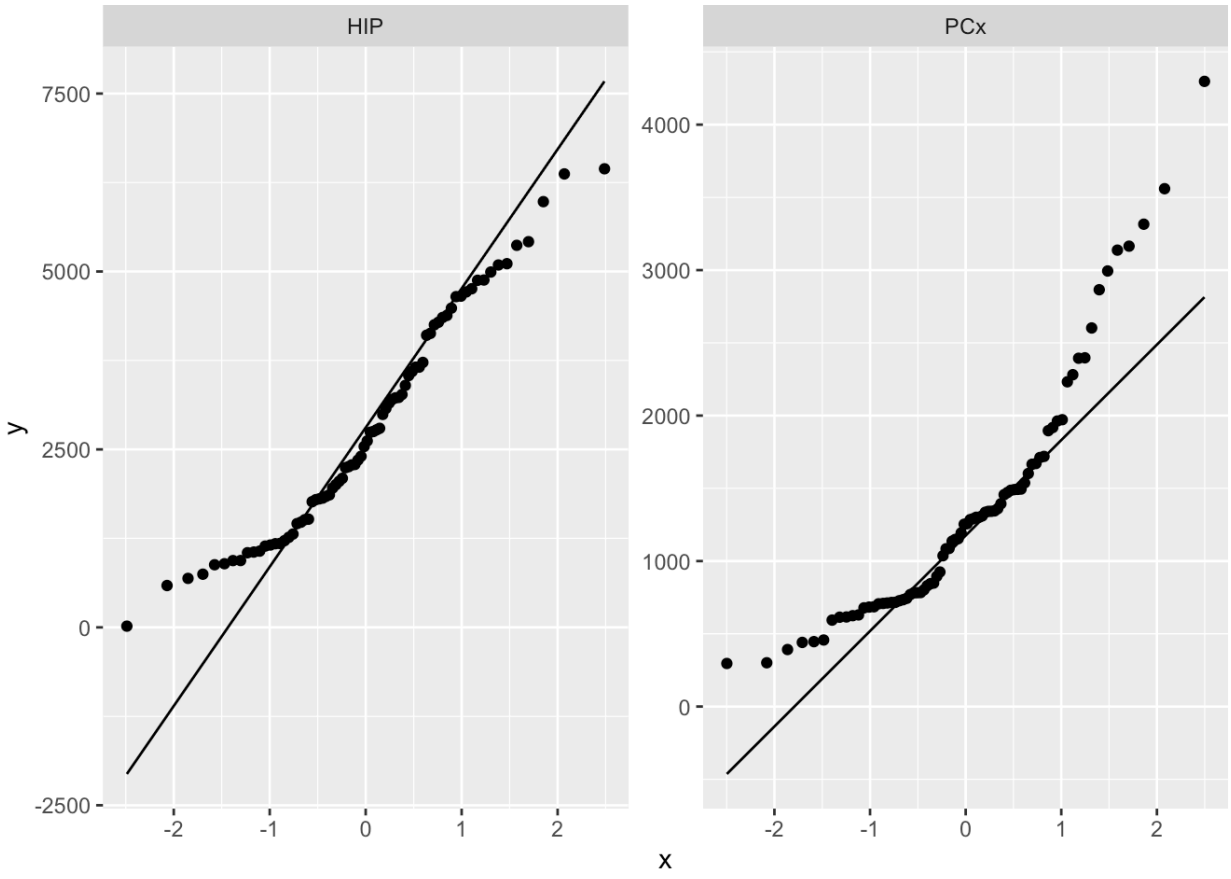
Note the warning you get when making these plots: “Warning: Removed 17 rows containing non-finite values ().”. This indicates that ggplot deals with the 17 rows containing `NA` for `ptau` by removing them completely from the plot.

As you can see, the `ptau` variable is clearly positively skewed. The distribution also appears to have a second small peak around 4500 pg per mg. We know that this variable comes from two different tissues (based on the `struct` column). Perhaps we need to facet based on the `struct` column to make more sense of the distribution.

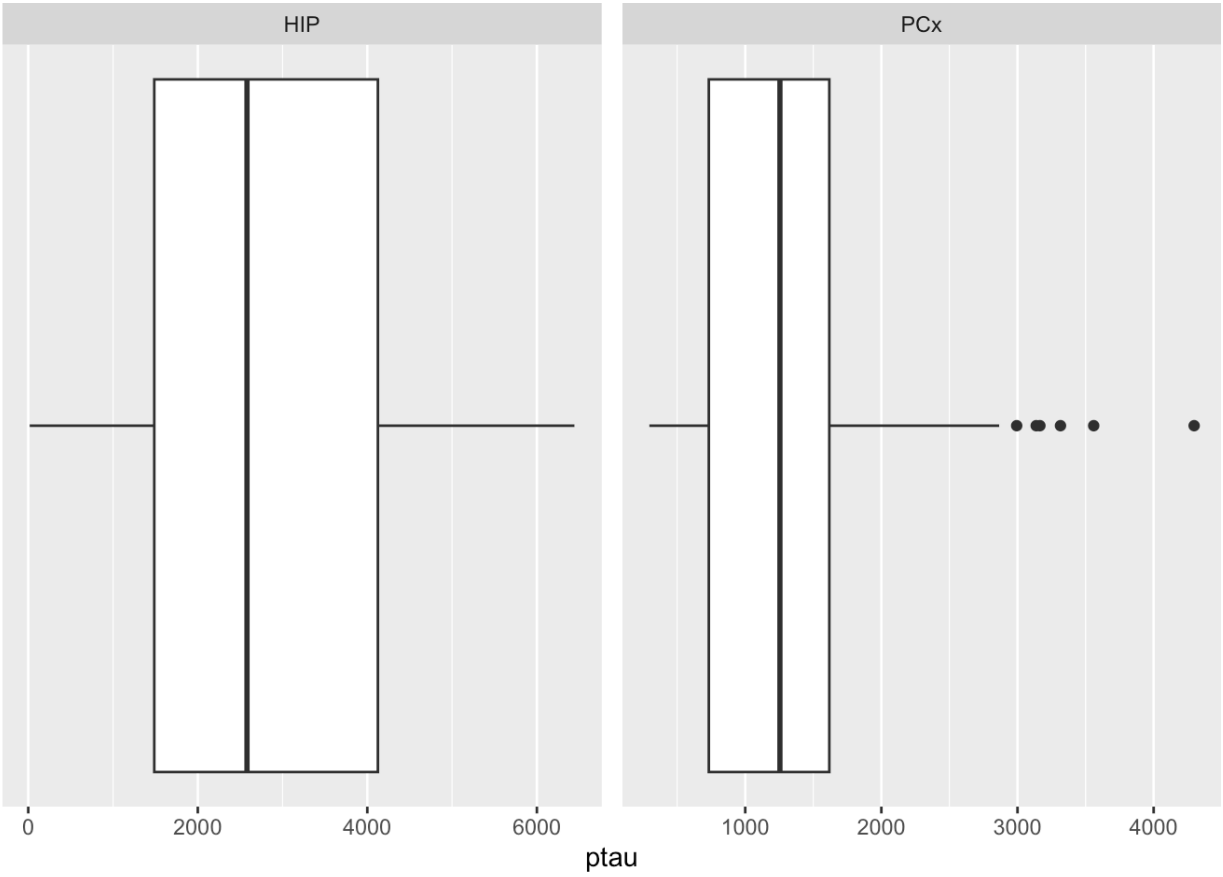
```
# Histogram
ggplot(data = ABI, mapping = aes(x = ptau)) +
  geom_histogram(bins = 30, fill = 'gray', color = 'black') +
  facet_wrap(~ struct, scales = "free")
```




```
# Q-Q plot
ggplot(data = ABI, mapping = aes(sample = ptau)) +
  geom_qq() +
  geom_qq_line() +
  facet_wrap(~ struct, scales = "free")
```




```
# Boxplot
ggplot(data = ABI, mapping = aes(x = ptau)) +
  geom_boxplot() +
  theme(axis.ticks.y = element_blank(),
        axis.text.y = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor.y = element_blank()) +
  facet_wrap(~ struct, scales = "free")
```



Okay, so as you can see, splitting up the distributions based on the `struct` variable seems to have made some more sense of the data. Essentially all of the higher values were from the Hippocampus tissue, while the large peak(s) near 1000 pg per mg were largely in the Parietal Cortex tissue samples. We can't visually assess normality on the histograms like before as the `stat_function` command doesn't behave well with `facet_wrap` . Based on just eyeballing the plots, it does appear that we still don't have normal distributions for this variable in each tissue, though we can confirm this using mathematical tests.

Mathematical Normality tests

First, let's test this variable across both tissues:

```
stat.desc.clean(dataset = ABI, variable = ptau)
```

```
## # A tibble: 1 × 6
##   skewness skew.2SE kurtosis kurt.2SE normtest.W    normtest.p
##      <dbl>    <dbl>    <dbl>    <dbl>      <dbl>         <dbl>
## 1     1.03     2.68    0.235    0.306      0.891 0.00000000206
```

Clearly, there is significant non-normality present in the data across both tissues, which we can see from the following points:

1. A skewness value of 1.03 is evidence for positive skew
2. A skew.2SE value of 2.68 is very strong evidence for positive skew, as this is far more than 1
3. A kurtosis value of 0.235 indicates weak evidence of kurtosis
4. A kurt.2SE value of 0.306 further confirms a lack of kurtosis

5. The normtest.p value of 2.06e-9 shows that the Shapiro-Wilk test was very significant and thus the variable is significantly non-normal

Now we wish to split this up based on the `struct` variable. We can do this using the `stat.desc.clean` function by simply adding the grouping variable (`struct` in this case) as a third argument.

```
stat.desc.clean(dataset = ABI, variable = ptau, struct)
```

```
## # A tibble: 2 × 7
## # Groups:   struct [2]
##   struct skewness skew.2SE kurtosis kurt.2SE normtest.W normtest.p
##   <fct>     <dbl>    <dbl>    <dbl>    <dbl>      <dbl>      <dbl>
## 1 HIP       0.412    0.757   -0.845   -0.785      0.957 0.0102
## 2 PCx       1.35     2.50     1.65     1.55       0.876 0.00000132
```

We can see from the statistics above that there was stronger evidence of positive skew of the `ptau` variable in the Parietal Cortex tissue compared to the Hippocampus tissue. This is indicated by the `skewness` and `skew.2SE` values both being over 1 for `PCx` but not `HIP` . The `ptau` distribution in the Parietal Cortex tissue also seemed to be leptokurtic as indicated by the `kurtosis` and `kurt.2SE` values both being over 1. This can be seen visually in the histogram above by looking at the sharp peak. In comparison, the `ptau` variable is more platykurtic in the `HIP` tissue, as shown by the negative `kurtosis` and `kurt.2SE` values, along with the flatter distribution in the histogram. The `normtest.p` value was significant (< 0.05) for both `HIP` and `PCx` , indicating that there was evidence for non-normality in both tissues, though the evidence was much stronger for the Parietal Cortex.

Is the ptau variable normally distributed?

No, there is clear evidence for non-normality for the `ptau` variable overall based on the graphical tests. The mathematical tests confirm this non-normality overall. When looking within each individual tissue however, there is evidence that this is largely being driven by the Parietal Cortex tissue, with the Hippocampus tissue exhibiting a closer to normal distribution of `ptau` . Regardless, the `ptau` variable appears to be non-normal when observed across both tissues or within each tissue.

Transformations to Meet Normality

If your data is not normally distributed, one option is to transform your variable. There are several options for transforming your data, but the two transformation methods we recommend are:

1. Square-root transformation
2. Log transformation

Both of these transformations are useful for shifting positively skewed data toward normality, with the log transformation being a more influential method compared to the square-root transformation.

Let's try transforming the `ptau` variable, then we can re-test normality of this variable using the following two functions:

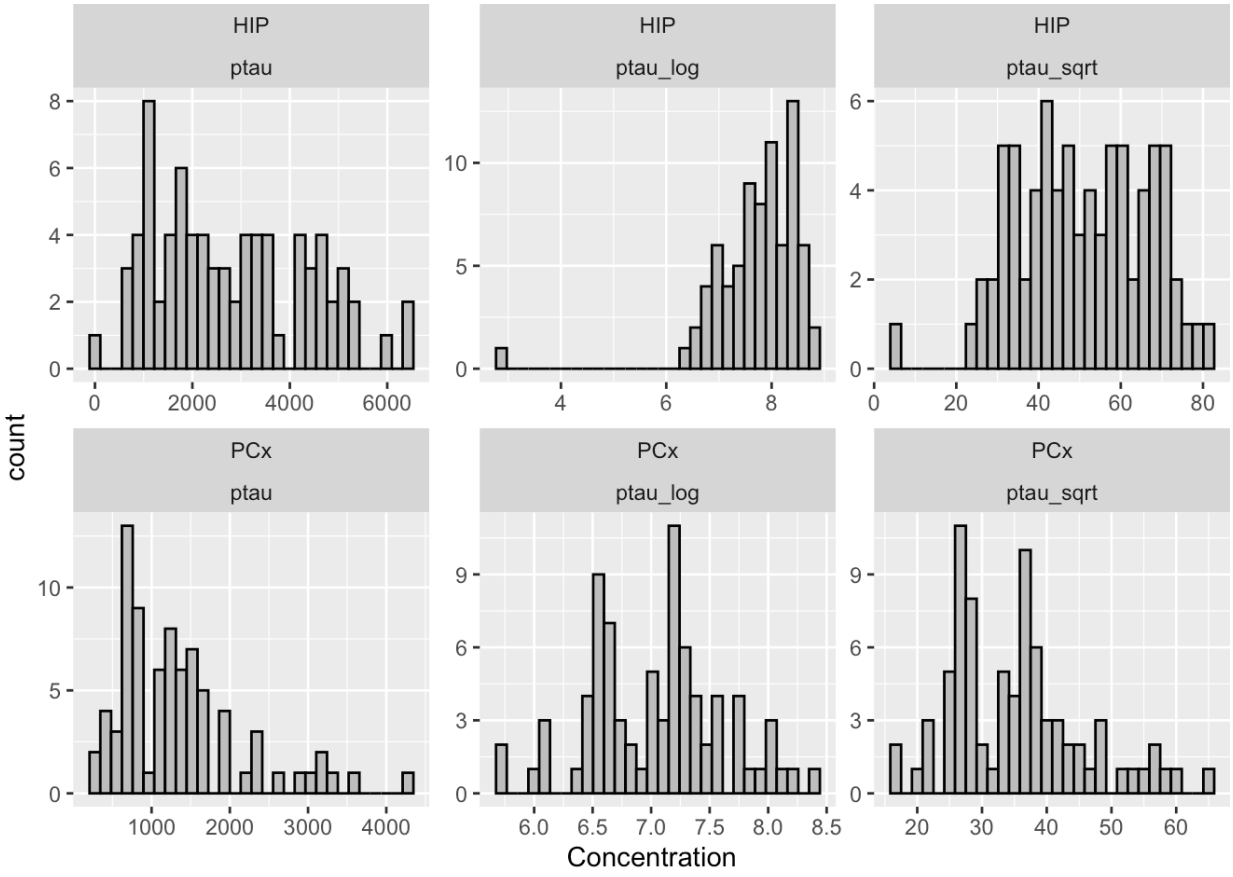
- `sqrt` : transform your data by taking its square root
- `log` : transform your data using the logarithmic function (defaults to natural log, change with `base` argument) - be careful when transforming certain numbers this way (such as 0)

To visualize the effects of these transformations, I will modify the `ABI` dataframe such that we add two new columns representing the square root transformed and the log transformed `ptau` data. We then use the `select` function to select the specific columns of interest by name. Finally, we use pivot_longer to change the dataframe into the long format, enabling easier plotting.

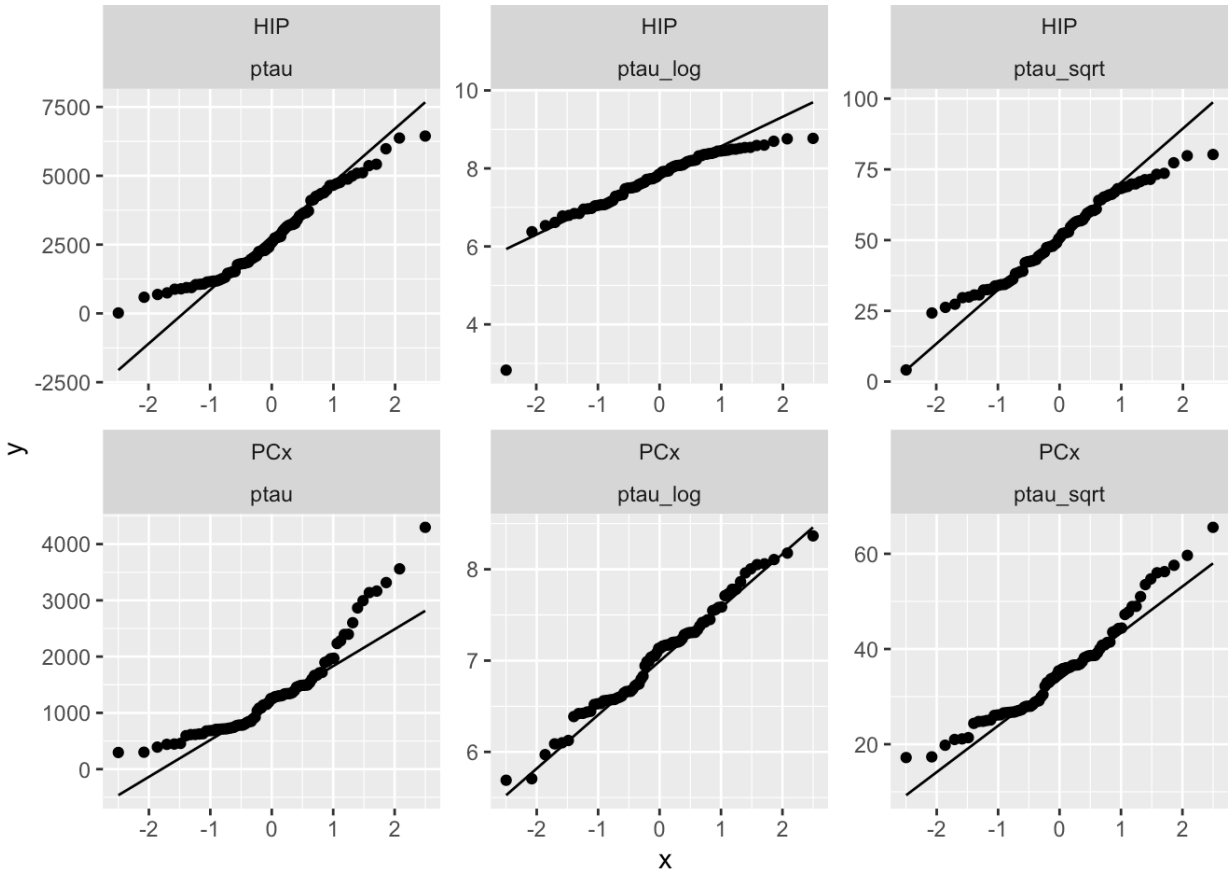
```
ABI2 <- ABI %>%
  mutate(ptau_sqrt = sqrt(ptau),
         ptau_log = log(ptau)) %>%
  select(ID, struct, ptau, ptau_sqrt, ptau_log) %>%
  pivot_longer(cols = ptau:ptau_log,
               names_to = "Measurement",
               values_to = "Concentration")
```

Once you transform your data, you can then check for normality using the standard graphical and mathematical tests. Here I have modified the code slightly for easier visualization, but the functions don't fundamentally differ. The only new thing for you all is that I have faceted based on a combination of the `struct` variable and the `Measurement` variable, specified by using the `*` between them. I also provided both grouping variables for the `stat.desc.clean` function.

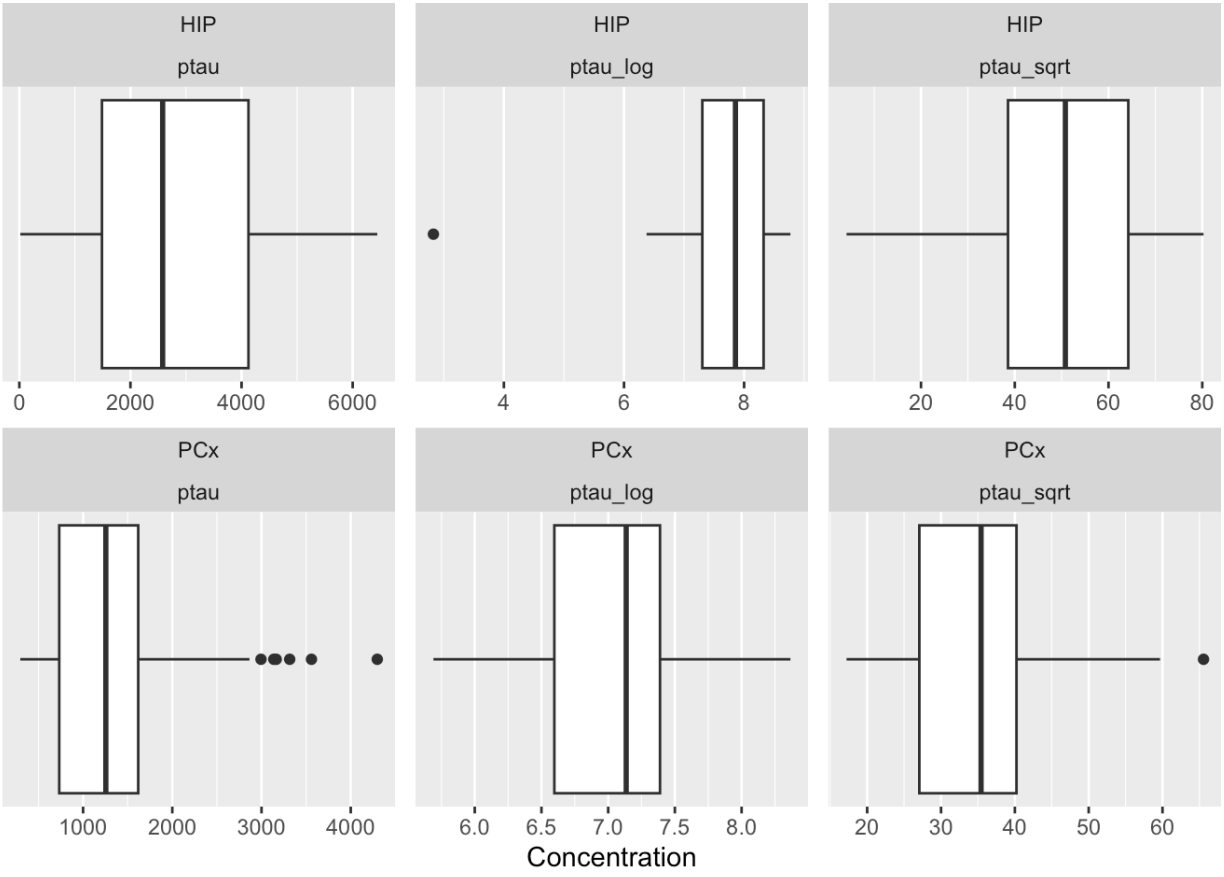
```
# Histogram
ggplot(data = ABI2, mapping = aes(x = Concentration)) +
  geom_histogram(bins = 30, fill = 'gray', color = 'black') +
  facet_wrap(~ struct * Measurement, scales = "free")
```




```
# Q-Q plot
ggplot(data = ABI2, mapping = aes(sample = Concentration)) +
  geom_qq() +
  geom_qq_line() +
  facet_wrap(~ struct * Measurement, scales = "free")
```




```
# Boxplot
ggplot(data = ABI2, mapping = aes(x = Concentration)) +
  geom_boxplot() +
  theme(axis.ticks.y = element_blank(),
        axis.text.y = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor.y = element_blank()) +
  facet_wrap(~ struct * Measurement, scales = "free")
```




```
stat.desc.clean(dataset = ABI2, variable = Concentration, struct, Measurement)
```

```
## # A tibble: 6 × 8
## # Groups:   struct, Measurement [6]
##   struct Measurement  skewness  skew.2SE kurtosis kurt.2SE normtest.W normtest.p
##   <fct>  <chr>           <dbl>     <dbl>    <dbl>    <dbl>      <dbl>      <dbl>
## 1 HIP    ptau          0.412     0.757     -0.845  -0.785       0.957    1.02e-2
## 2 HIP    ptau_log     -2.71     -4.98       13.6    12.6         0.788    3.11e-9
## 3 HIP    ptau_sqrt    -0.193    -0.354     -0.464  -0.431       0.977    1.64e-1
## 4 PCx    ptau          1.35      2.50       1.65    1.55        0.876    1.32e-6
## 5 PCx    ptau_log     -0.000203 -0.000378  -0.365  -0.343       0.985    4.68e-1
## 6 PCx    ptau_sqrt     0.692     1.29       0.100   0.0944      0.954    6.21e-3
```

You can see that the two transformations have similar effects, though the log transformation more dramatically pushes the distribution to the right (i.e. corrects positive skew) compared to the square root transformation. Note that in the PCx tissue, the log transformation dramatically improved the normality of the `ptau` variable.

Homogeneity of Variance

When comparing a variable between groups, the groups should have roughly equal variance. This essentially means the spread of the data in different groups should be roughly the same. In order to test if variances in different groups are the same, we will use Levene's test. For this, we will use the `leveneTest` function from the `car` package.

The `leveneTest` function uses another type of R object called a formula. A formula is a useful way to tell a function you want to see how a variable varies "with respect to" another variable (or variables), with the following form:

```
dependent/outcome variable ~ independent/predictor variable 1 + independent/predictor variable 2 + ...
```

Think of this as “dependent variable as a function of (~) independent variable 1 and independent variable 2”. The `leveneTest` function will have the following form:

```
leveneTest(y = formula, data = df)
```

You are testing the variance of the dependent variable as a function of the independent variable. When using the Levene Test, the dependent variable should be numeric while the independent variable(s) is/are your grouping variable(s).

The output of the `leveneTest` function is a small table with the grouping variable, the degrees of freedom, the F statistic, and the p value. If the p-value is < 0.05, the variances within each group are significantly different from each other.

Now let’s test if the variance in `ptau` is the same as a function of `struct`:

```
leveneTest(ptau ~ struct, data = ABI)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value    Pr(>F)
## group   1  38.221 5.293e-09 ***
##       156
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Is the variance of `ptau` similar between each structure?

No, the p-value is less than 0.05, the variance between structures is significantly different (F(1,156) = 38.221, p = 5.293e-09). Thus, the spread of data between the groups is dissimilar.

Practice

1. Go ahead and test the normality assumption for `mip_1a` the following two ways. Try out the two above transformation methods to see how they influence your data:
    a. In the full dataset
    b. In each tissue individually. Hint: you may need to remove some data points for the log transformation to work properly.