Analysis for Cell Culture Data

Marieke Jones, PhD

2022-11-02

Contents

Set up

Load packages we need for analysis

```
library(tidyverse)
library(readxl)
library(lme4)
library(lmerTest)
library(emmeans)
library(zoo)
library(ggeffects)
library(splines)
```

Figure 1D

MAPT+/+ (4 independent experiments)

Is the Soma-AIS gap different between vehicle and xcTauOs

Load Data

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/1D - MAPT+-+ Soma-AIS Gap/", filename), skip = 1) %>%
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Gap")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/1D - MAPT+-+ Soma-AIS Gap/")

Fig1D <- map(my_files, ~readclean(.))
Fig1D <- bind_rows(Fig1D)

Fig1D
```

```
## # A tibble: 298 x 4
##    `Distance_(microns)` filename                                  Neuron   Gap
##    <lgl>                <chr>                                     <chr>  <dbl>
##  1 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~ 18.0
##  2 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~  7.86
##  3 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~ 25.8
##  4 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~  7.63
##  5 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~  6.97
##  6 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~ 29.4
##  7 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~ 16.2
##  8 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~  8.29
##  9 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~  2.31
## 10 NA                   (1D) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV 1~ Neuro~ 20.9
## # ... with 288 more rows
```

Create new variables for Mouse and Group

```
Fig1D <- Fig1D %>%
  select(-`Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\+ ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig1D
```

```
## # A tibble: 298 x 3
##    Mouse     Neuron      Gap
##    <chr>     <chr>     <dbl>
##  1 Vehicle 1 Neuron 1  18.0
##  2 Vehicle 1 Neuron 2   7.86
##  3 Vehicle 1 Neuron 3  25.8
##  4 Vehicle 1 Neuron 4   7.63
##  5 Vehicle 1 Neuron 5   6.97
##  6 Vehicle 1 Neuron 6  29.4
##  7 Vehicle 1 Neuron 7  16.2
##  8 Vehicle 1 Neuron 8   8.29
##  9 Vehicle 1 Neuron 9   2.31
## 10 Vehicle 1 Neuron 10 20.9
## # ... with 288 more rows
```

Create treatment group variable

```
Fig1D <- Fig1D %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Gap)
```

Create Experiment variable

Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

```
Fig1D <- Fig1D %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D"))

Fig1D
```

```
## # A tibble: 298 x 5
##    Group   Mouse     Neuron      Gap Experiment
##    <chr>   <chr>     <chr>     <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1  18.0  A
##  2 Vehicle Vehicle 1 Neuron 2   7.86 A
##  3 Vehicle Vehicle 1 Neuron 3  25.8  A
##  4 Vehicle Vehicle 1 Neuron 4   7.63 A
##  5 Vehicle Vehicle 1 Neuron 5   6.97 A
##  6 Vehicle Vehicle 1 Neuron 6  29.4  A
##  7 Vehicle Vehicle 1 Neuron 7  16.2  A
```

```
##  8 Vehicle Vehicle 1 Neuron 8   8.29 A
##  9 Vehicle Vehicle 1 Neuron 9   2.31 A
## 10 Vehicle Vehicle 1 Neuron 10 20.9  A
## # ... with 288 more rows
```

Basic data checks

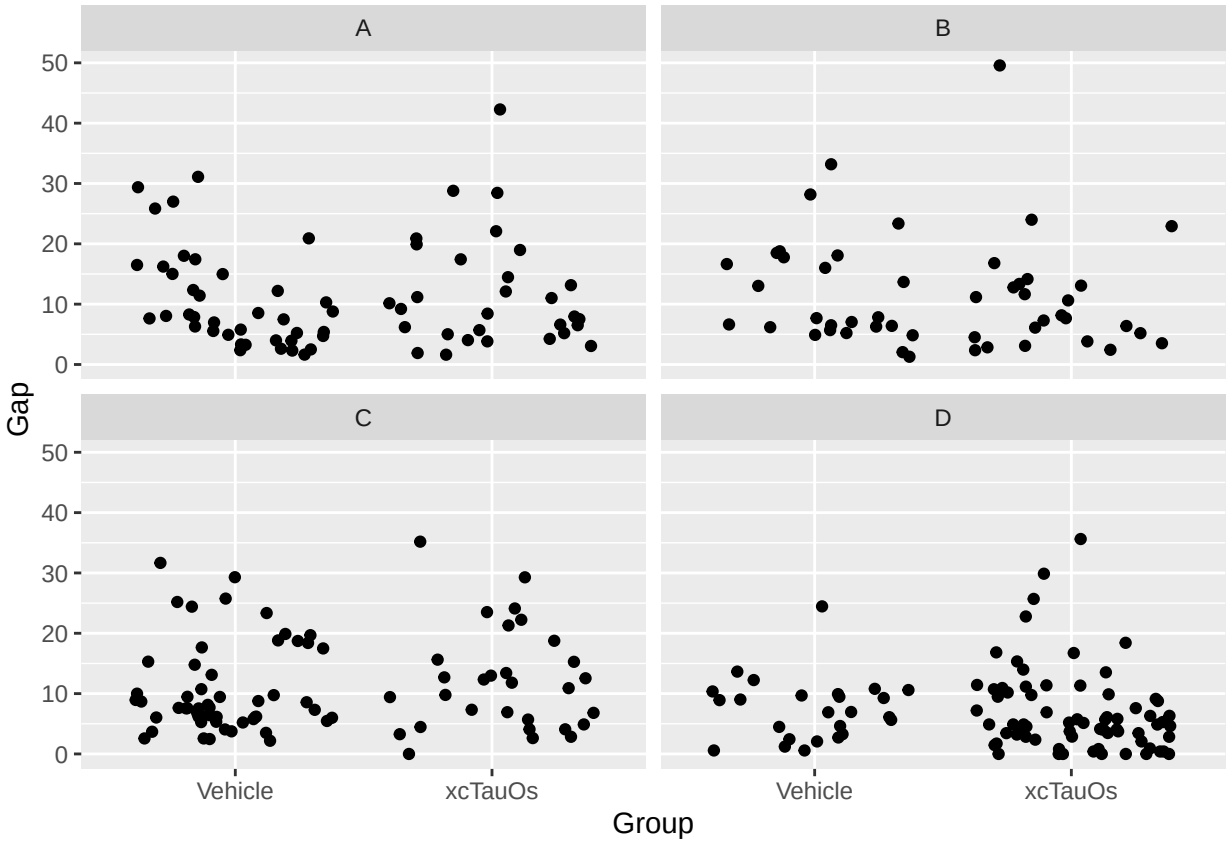
How many neurons per experiment & treatment

```
Fig1D %>%
  count(Mouse)
```

```
## # A tibble: 8 x 2
##   Mouse         n
##   <chr>     <int>
## 1 Vehicle 1    39
## 2 Vehicle 2    25
## 3 Vehicle 3    54
## 4 Vehicle 4    25
## 5 xcTauOs 1    30
## 6 xcTauOs 2    24
## 7 xcTauOs 3    30
## 8 xcTauOs 4    71
```

Exploratory plot

```
Fig1D %>%
  ggplot(aes(Group, Gap)) +
  geom_jitter() +
  facet_wrap(~Experiment)
```



Model

Take into account which mouse the cells came from. Embryos from one pregnant mouse were homogenized then divided into Vehicle and xcTauOs, so our random effects are experiment and Mouse

```
Fig1Dmod <- lmer(Gap ~ Group + (1|Experiment) + (1|Mouse), data = Fig1D)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(Fig1Dmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Gap ~ Group + (1 | Experiment) + (1 | Mouse)
##    Data: Fig1D
##
## REML criterion at convergence: 2078.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.4254 -0.6767 -0.3375  0.4276  4.8519
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
```

```
##  Mouse      (Intercept)  0.000   0.000
##  Experiment (Intercept)  3.927   1.982
##  Residual               62.328   7.895
## Number of obs: 298, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)    9.9918     1.1990   4.3976   8.333 0.000743 ***
## GroupxcTauOs   0.3127     0.9568 294.6617   0.327 0.744038
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.405
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

xcTauOs is 0.3127 higher (SE = 0.96)

No difference in Gap

Because the fit was singular, we should check that the estimates are stable

```
Fig1Dmod_check <- lmer(Gap ~ Group + (1|Experiment), data = Fig1D)
summary(Fig1Dmod_check)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Gap ~ Group + (1 | Experiment)
##    Data: Fig1D
##
## REML criterion at convergence: 2078.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.4254 -0.6767 -0.3375  0.4276  4.8519
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Experiment (Intercept)  3.927   1.982
##  Residual               62.328   7.895
## Number of obs: 298, groups:  Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)    9.9918     1.1990   4.3976   8.333 0.000743 ***
## GroupxcTauOs   0.3127     0.9568 294.6617   0.327 0.744038
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.405
```

Estimates are stable, use the Fig1Dmod despite the singular fit

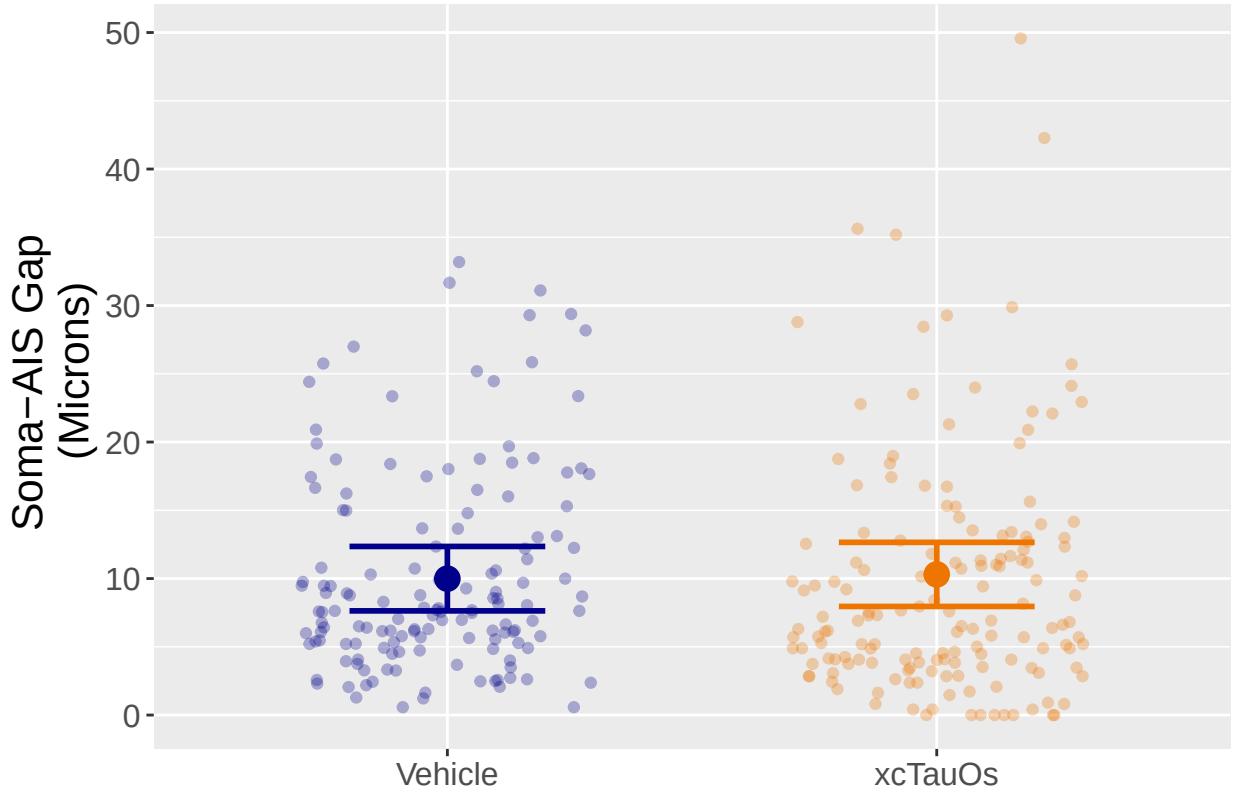
Plot

Jitter with 95% CI (like human data) - vehicle = blue - tau = orange

```
# return dataset of predicted mean +/- 95%CI
pred_gap <- ggeffect(Fig1Dmod, terms = "Group") %>%
  as_tibble() %>%
  rename(Group = x)

pred_gap
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      9.99      1.20     7.63      12.4 1
## 2 xcTauOs     10.3       1.19     7.96      12.7 1
```

```
gapplot <- Fig1D %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Gap,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_gap,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_gap,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Soma-AIS Gap \n (Microns)", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

gapplot
```




```
ggsave(gapplot, filename = "Figures/cell_1D.png", width = 6, height = 4)
```

Figure 1E

MAPT+/+ (4 independent experiments)

Is the AIS concentration different between vehicle and xcTauOs

Load Data

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/1E - MAPT+-+ AIS Concentration/", filename), skip = 1) %>%
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Concentration")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/1E - MAPT+-+ AIS Concentration/")

Fig1E <- map(my_files, ~readclean(.))
Fig1E <- bind_rows(Fig1E)

Fig1E
```

```
## # A tibble: 128,601 x 4
##    `Distance_(microns)` filename                              Neuron Concentration
##                   <dbl> <chr>                                <chr>          <dbl>
##  1                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~         130.
##  2                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~         116.
##  3                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~          49.5
##  4                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~          84.3
##  5                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~          95.5
##  6                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~         103.
##  7                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~          71.8
##  8                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~         122.
##  9                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~          47.7
## 10                    0 (1E) MAPT+_+ Vehicle 1 =  6.2.21 W~ Neuro~          88.9
## # ... with 128,591 more rows
```

Create new variables for Mouse and Group

```
Fig1E <- Fig1E %>%
  rename(Dist = `Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\+ ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig1E
```

```
## # A tibble: 128,601 x 4
##     Dist Mouse     Neuron    Concentration
##    <dbl> <chr>     <chr>             <dbl>
##  1     0 Vehicle 1 Neuron 1          130.
##  2     0 Vehicle 1 Neuron 2          116.
##  3     0 Vehicle 1 Neuron 3           49.5
##  4     0 Vehicle 1 Neuron 4           84.3
##  5     0 Vehicle 1 Neuron 5           95.5
##  6     0 Vehicle 1 Neuron 6          103.
##  7     0 Vehicle 1 Neuron 7           71.8
##  8     0 Vehicle 1 Neuron 8          122.
##  9     0 Vehicle 1 Neuron 9           47.7
## 10     0 Vehicle 1 Neuron 10          88.9
## # ... with 128,591 more rows
```

Create treatment group variable

```
Fig1E <- Fig1E %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Dist, Concentration)
```

Create Experiment variable

Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

```
Fig1E <- Fig1E %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D"))

Fig1E
```

```
## # A tibble: 128,601 x 6
##    Group   Mouse     Neuron     Dist Concentration Experiment
##    <chr>   <chr>     <chr>     <dbl>         <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1      0         130.  A
##  2 Vehicle Vehicle 1 Neuron 2      0         116.  A
##  3 Vehicle Vehicle 1 Neuron 3      0          49.5 A
##  4 Vehicle Vehicle 1 Neuron 4      0          84.3 A
##  5 Vehicle Vehicle 1 Neuron 5      0          95.5 A
##  6 Vehicle Vehicle 1 Neuron 6      0         103.  A
##  7 Vehicle Vehicle 1 Neuron 7      0          71.8 A
##  8 Vehicle Vehicle 1 Neuron 8      0         122.  A
##  9 Vehicle Vehicle 1 Neuron 9      0          47.7 A
## 10 Vehicle Vehicle 1 Neuron 10     0          88.9 A
## # ... with 128,591 more rows
```

Clean dataset

Drop observations where Concentration is missing because the neuron wasn't that long

```
Fig1E <- Fig1E %>%
  arrange(Group, Mouse, Neuron, Dist) %>%
  drop_na(Concentration)
```

Basic data checks

How many neurons per experiment

```
Fig1E %>%
  count(Mouse)
```

```
## # A tibble: 8 x 2
##   Mouse         n
##   <chr>     <int>
## 1 Vehicle 1  4870
## 2 Vehicle 2  4160
## 3 Vehicle 3  6043
## 4 Vehicle 4  3816
## 5 xcTauOs 1  2678
## 6 xcTauOs 2  2883
## 7 xcTauOs 3  2944
## 8 xcTauOs 4  8318
```

Exploratory plot

- Make rolling average plot

A rolling average across 3-distance observations works nicely to show the trend.

I used the `rollapply()` function rather than the more standard `rollmean()` function because `rollmean()` has no way to remove NAs.

```
Fig1E <- Fig1E %>%
  group_by(Mouse, Neuron) %>%
  mutate(roll_Conc = rollapply(Concentration, 3, mean, na.rm = TRUE, fill = NA)) %>%
  ungroup()

Fig1E
```

```
## # A tibble: 35,712 x 7
##    Group   Mouse     Neuron    Dist Concentration Experiment roll_Conc
##    <chr>   <chr>     <chr>    <dbl>         <dbl> <chr>          <dbl>
##  1 Vehicle Vehicle 1 Neuron 1 0              130. A               NA
##  2 Vehicle Vehicle 1 Neuron 1 0.135          143. A              144.
##  3 Vehicle Vehicle 1 Neuron 1 0.271          159. A              160.
##  4 Vehicle Vehicle 1 Neuron 1 0.406          177. A              178.
##  5 Vehicle Vehicle 1 Neuron 1 0.542          197. A              196.
##  6 Vehicle Vehicle 1 Neuron 1 0.677          213. A              212.
##  7 Vehicle Vehicle 1 Neuron 1 0.813          226. A              224.
##  8 Vehicle Vehicle 1 Neuron 1 0.948          234. A              233.
##  9 Vehicle Vehicle 1 Neuron 1 1.08           239. A              238.
## 10 Vehicle Vehicle 1 Neuron 1 1.22           240. A              240.
## # ... with 35,702 more rows
```

One line per mouse, rolling average averaged over all neurons at a given dist.

The rolling average is the average of a 3-dist chunk.

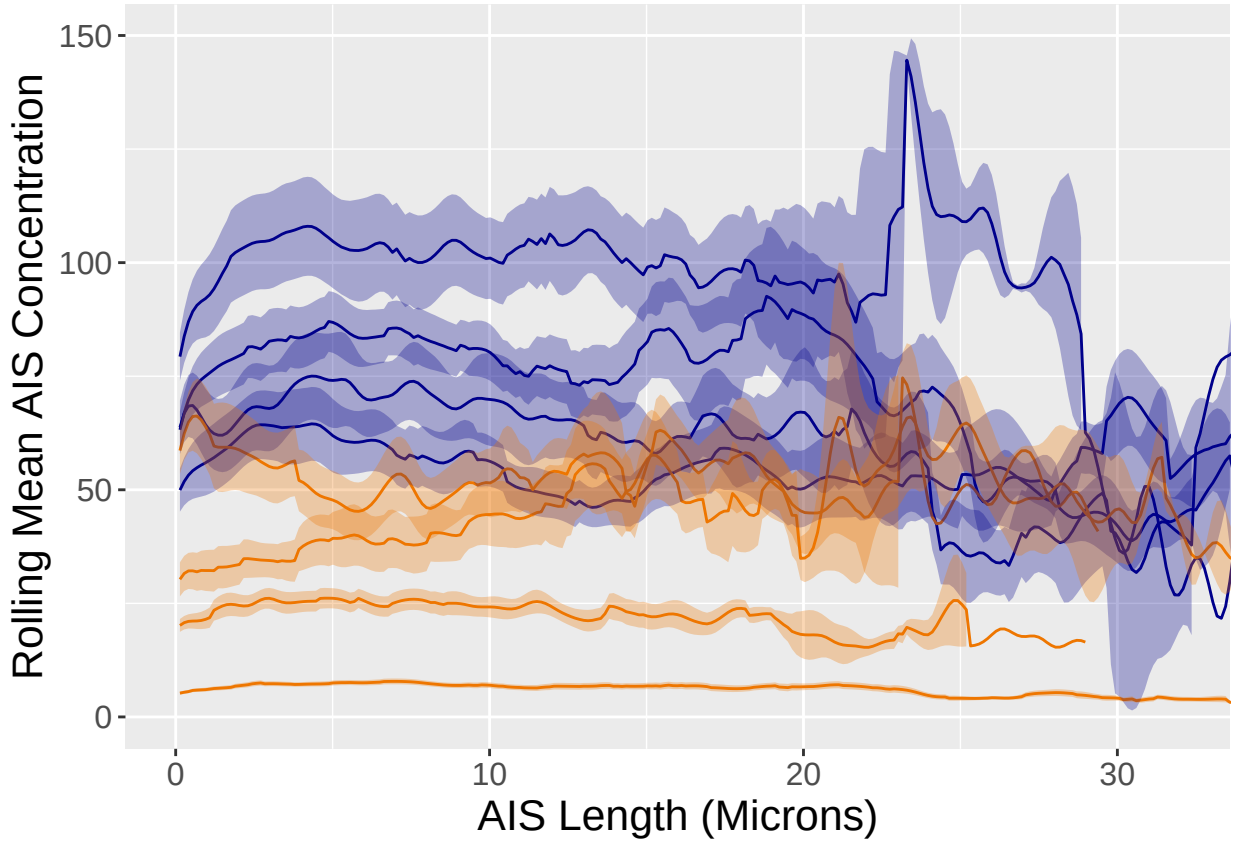
```
rollavgdat <- Fig1E %>%
  group_by(Group, Mouse, Dist) %>%
  summarize(Mean_Conc = mean(roll_Conc, na.rm = TRUE),
            sd_Conc = sd(roll_Conc, na.rm = TRUE),
            n_neurons = n(),
            se_Conc = sd_Conc/sqrt(n_neurons))
```

```
## `summarise()` has grouped output by 'Group', 'Mouse'. You can override using
## the `.groups` argument.
```

```
lineplot <- rollavgdat %>%
  ggplot(aes(Dist, Mean_Conc)) +
  geom_line(aes(group = Mouse, color = Group)) +
  geom_ribbon(aes(ymin = Mean_Conc-se_Conc,
                  ymax = Mean_Conc+se_Conc,
                  group = Mouse, fill = Group), alpha = .3) +
  scale_fill_manual(values = c("darkblue", "darkorange2")) +
  scale_color_manual(values = c("darkblue", "darkorange2")) +
```

```
  labs(y = "Rolling Mean AIS Concentration", x = "AIS Length (Microns)") +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size =  12)) +
  coord_cartesian(xlim =c(0, 32))

lineplot
```

```
## Warning: Removed 16 row(s) containing missing values (geom_path).
```




```
ggsave(lineplot, filename = "Figures/cell_1E_distVintensity.png", width =  6, height = 4)
```

```
## Warning: Removed 16 row(s) containing missing values (geom_path).
```

Model splines

Splines are a way to fit a non-linear curve to data to understand how the relationship between Distance and Concentration changes for xcTauOs v. Vehicle.

Try basic natural splines model

```
splinemod <- lmer(Concentration ~ ns(Dist, df = 5) + Group + (1|Mouse) + (1|Experiment), data = Fig1E)
```

Now try with interaction term

```
splinemod2 <- lmer(Concentration ~ ns(Dist, df = 5) * Group + (1|Mouse) + (1|Experiment), data = Fig1E)
```

See if there is a difference in model fit between splinemod and splinemod2

```
anova(splinemod2, splinemod)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: Fig1E
## Models:
## splinemod: Concentration ~ ns(Dist, df = 5) + Group + (1 | Mouse) + (1 | Experiment)
## splinemod2: Concentration ~ ns(Dist, df = 5) * Group + (1 | Mouse) + (1 | Experiment)
##            npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## splinemod    10 354740 354825 -177360   354720
## splinemod2   15 354577 354704 -177274   354547 172.49  5  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

splinemod2 fits better than splinemod, so use that

Interpretation of splinemod2, use emmeans to find average difference

```
pairs(emmeans(splinemod2, specs = "Group"))
```

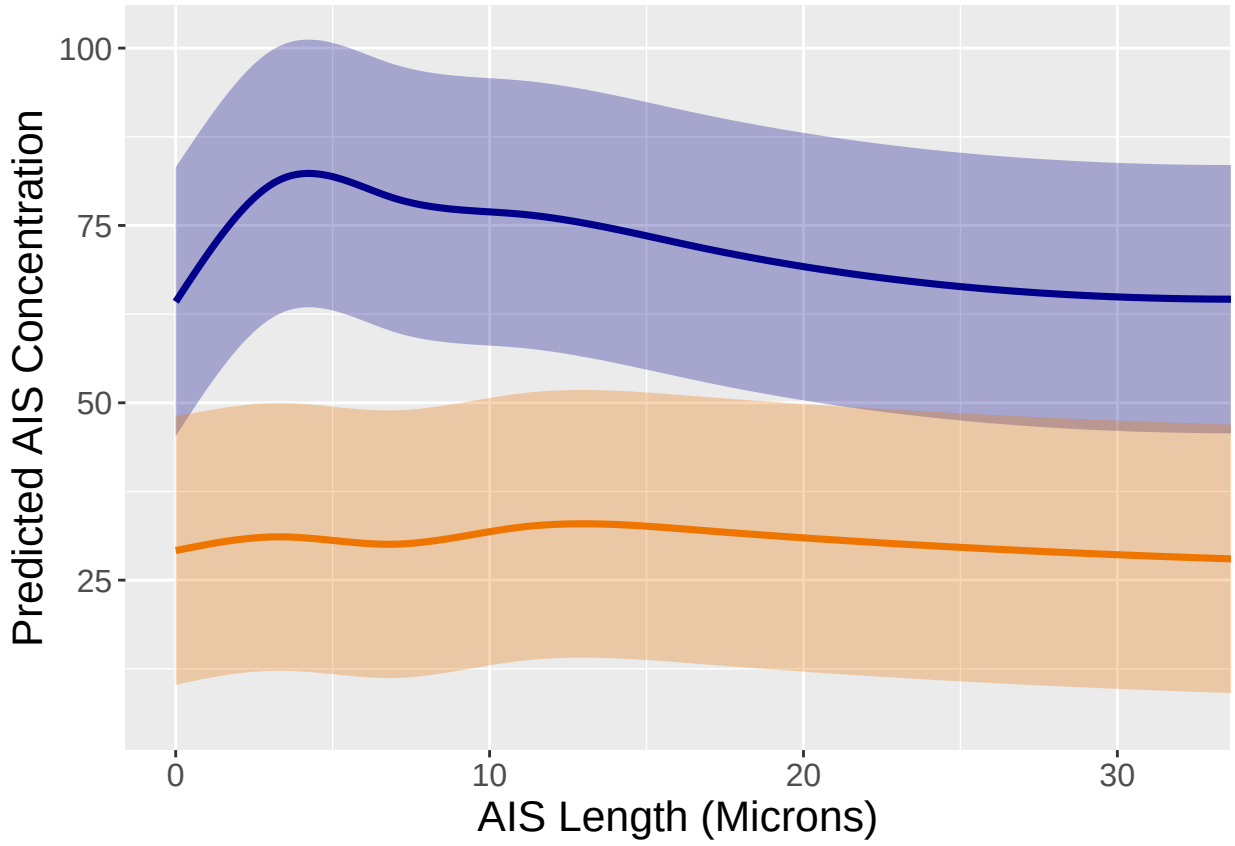
```
##  contrast          estimate   SE  df z.ratio p.value
##  Vehicle - xcTauOs     44.4 10.6 Inf   4.200  <.0001
##
## Degrees-of-freedom method: asymptotic
```

On average xcTauOs has AIS concentration 44.4 lower than vehicle ($p < 0.0001$)

Plot splines

```
splinesplot <- ggpredict(splinemod2, terms = c("Dist [all]", "Group")) %>%
  as_tibble() %>%
  rename(Dist = x,
         Group = group) %>%
  ggplot(aes(Dist, predicted, color = Group)) +
  #facet_wrap(~Group, nrow = 2) +
  geom_ribbon(aes(ymin = conf.low,
                  ymax = conf.high,
                  fill = Group),
              alpha = .3,
              color = NA) +
  geom_line(lwd = 1.25) +
  scale_color_manual(values = c("darkblue", "darkorange2")) +
  scale_fill_manual(values = c("darkblue", "darkorange2")) +
  labs(x = "AIS Length (Microns)",
       y = "Predicted AIS Concentration", color = "") +
```

```
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12)) +
  coord_cartesian(xlim = c(0,32))

splinesplot
```




```
ggsave(splinesplot, filename = "Figures/cell_1E_splines.png", width = 6, height = 4)
```

Model mean concentration

Create average intensity dataset. For each neuron, what is the average intensity across the whole distance that was measured

```
avgint <- Fig1E %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarize(Mean_Conc = mean(Concentration),
            n = n()) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

```
avgint
```

```
## # A tibble: 297 x 6
##    Group   Experiment Mouse     Neuron    Mean_Conc     n
##    <chr>   <chr>      <chr>     <chr>         <dbl> <int>
##  1 Vehicle A          Vehicle 1 Neuron 1      191.    124
##  2 Vehicle A          Vehicle 1 Neuron 10     109.    122
##  3 Vehicle A          Vehicle 1 Neuron 11     148.    143
##  4 Vehicle A          Vehicle 1 Neuron 12      90.6    14
##  5 Vehicle A          Vehicle 1 Neuron 13     120.    136
##  6 Vehicle A          Vehicle 1 Neuron 14     113.    115
##  7 Vehicle A          Vehicle 1 Neuron 15      71.1   142
##  8 Vehicle A          Vehicle 1 Neuron 16      52.9    79
##  9 Vehicle A          Vehicle 1 Neuron 17     198.    138
## 10 Vehicle A          Vehicle 1 Neuron 18      69.4   107
## # ... with 287 more rows
```

Model the mean concentration by Group

```
avgintmodB <- lmer(Mean_Conc ~ Group + (1|Mouse) + (1|Experiment), data = avgint)

summary(avgintmodB)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Mean_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: avgint
##
## REML criterion at convergence: 2896.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7114 -0.4953 -0.0701  0.3149  3.9829
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept) 258.18   16.068
##  Experiment (Intercept)  56.58    7.522
##  Residual               988.21   31.436
## Number of obs: 297, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    74.889      9.290   5.846   8.062 0.000222 ***
## GroupxcTauOs  -45.724     12.008   2.857  -3.808 0.034657 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.647
```

xcTauOs has average concentration 45.7 lower than Vehicle (p = 0.034)

Plot Mean Concentration

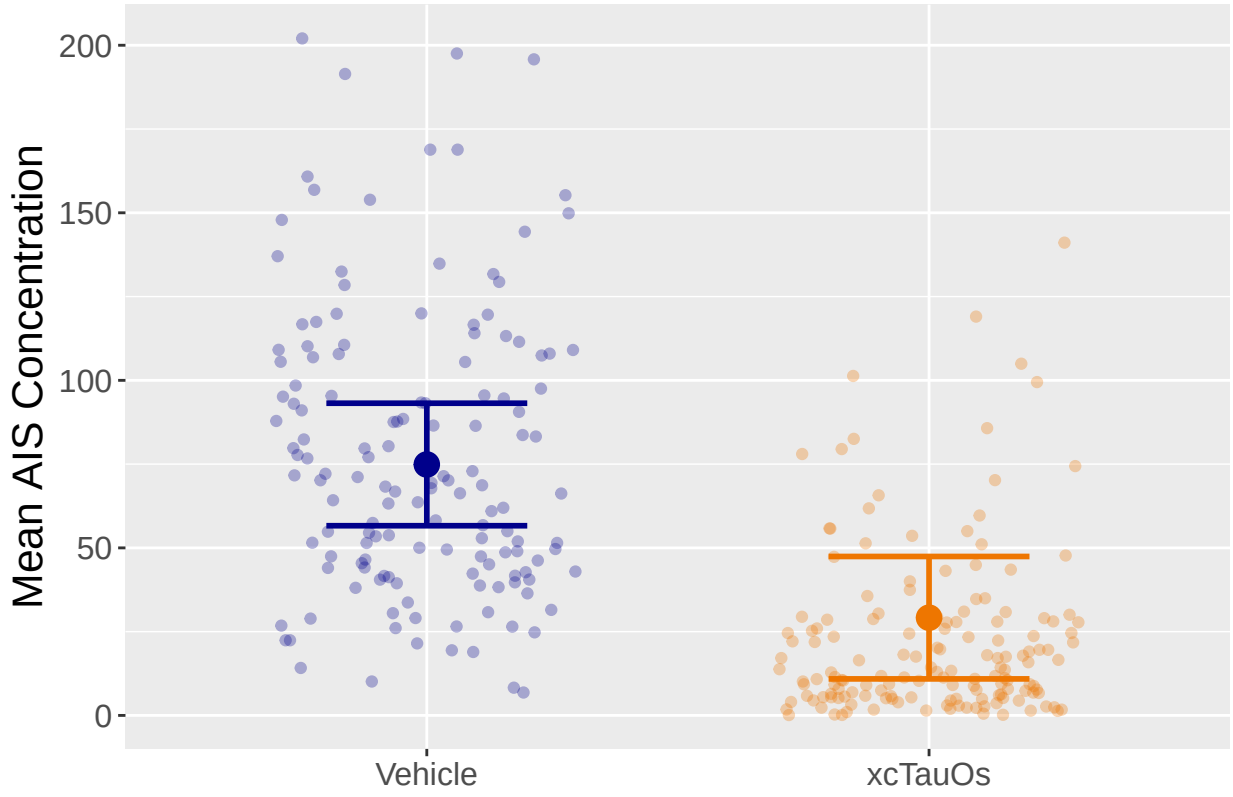
Each dot is an average of all of the concentration values for each neuron.

On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```
# return dataset of predicted mean +/- 95%CI
pred_mean_avgintmod <- ggeffect(avgintmodB, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)

pred_mean_avgintmod
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      74.9      9.29     56.6      93.2 1
## 2 xcTauOs      29.2      9.28     10.9      47.4 1
```

```
meanint_supp_plot <- avgint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Mean_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_mean_avgintmod,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_mean_avgintmod,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Mean AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

meanint_supp_plot
```




```
ggsave(meanint_supp_plot, filename = "Figures/cell_1E_meanconc.png", width = 6, height = 4)
```

Model max concentration

Create max concentration variable

Define the maximum concentration based on the rolling average concentration for each neuron since that worked better for the human data.

There is one Neuron (Neuron 30 for Mouse T3) that only has one measurement at Dist = 0, therefore, no rolling average concentration and no Max concentration. Remove.

```
maxint <- Fig1E %>%
  filter(!(Mouse == "xcTauOs 3" & Neuron == "Neuron 30")) %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarise(Max_Conc = max(roll_Conc, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

Check that each neuron only has one maximum

```
maxint %>%
  count(Mouse, Neuron) %>%
  arrange(-n)
```

```
## # A tibble: 296 x 5
## # Groups:   Group, Experiment, Mouse [8]
##    Group   Experiment Mouse     Neuron        n
##    <chr>   <chr>      <chr>     <chr>     <int>
##  1 Vehicle A          Vehicle 1 Neuron 1      1
##  2 Vehicle A          Vehicle 1 Neuron 10     1
##  3 Vehicle A          Vehicle 1 Neuron 11     1
##  4 Vehicle A          Vehicle 1 Neuron 12     1
##  5 Vehicle A          Vehicle 1 Neuron 13     1
##  6 Vehicle A          Vehicle 1 Neuron 14     1
##  7 Vehicle A          Vehicle 1 Neuron 15     1
##  8 Vehicle A          Vehicle 1 Neuron 16     1
##  9 Vehicle A          Vehicle 1 Neuron 17     1
## 10 Vehicle A          Vehicle 1 Neuron 18     1
## # ... with 286 more rows
```

Model the max concentration by Group

```
maxmod <- lmer(Max_Conc ~ Group + (1|Mouse) + (1|Experiment), data = maxint)

summary(maxmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Max_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: maxint
##
## REML criterion at convergence: 3050.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.7941 -0.5227 -0.0509  0.4043  3.2144
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)  621.9   24.94
##  Experiment (Intercept)  300.4   17.33
##  Residual                1710.8   41.36
## Number of obs: 296, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   114.545     15.613   5.465   7.336   0.0005 ***
## GroupxcTauOs  -66.490     18.366   2.897  -3.620   0.0384 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.588
```

xcTauOs has maximum concentration 66.5 lower than Vehicle (p = 0.038)

Plot Max Concentration

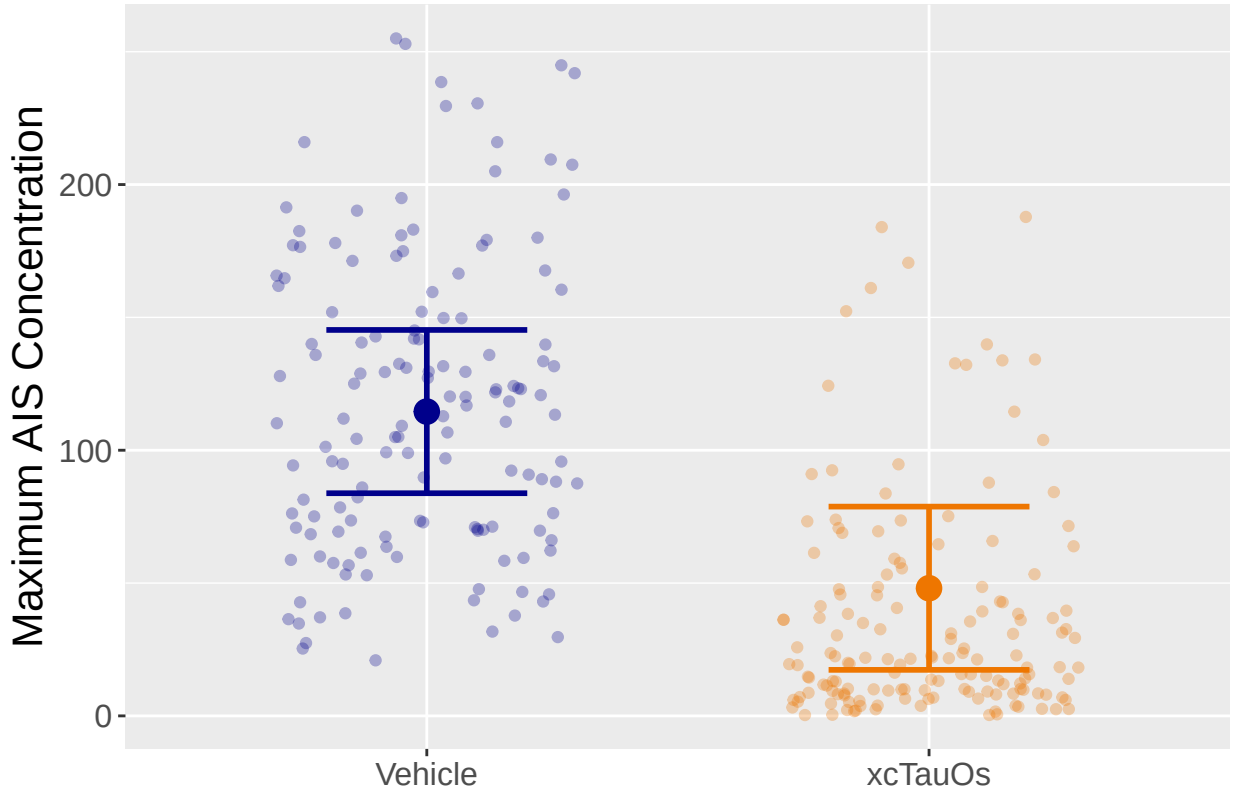
Each dot is the maximum concentration values for each neuron.

On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```
# return dataset of predicted mean +/- 95%CI
pred_max <- ggeffect(maxmod, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)

pred_max
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle     115.       15.6     83.8     145.  1
## 2 xcTauOs      48.1      15.6     17.3      78.8 1
```

```
maxplot <- maxint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Max_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_max,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_max,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Maximum AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

maxplot
```




```
ggsave(maxplot, filename = "Figures/cell_1E_maxconc.png", width = 6, height = 4)
```

Model Min Concentration

Create min concentration

Define the minimum concentration based on the rolling average concentration for each neuron since that worked better for the human data.

There is one Neuron (Neuron 30 for Mouse T3) that only has one measurement at Dist = 0, therefore, no rolling average concentration and no Max concentration. Remove.

```
minint <- Fig1E %>%
  filter(!(Mouse == "xcTauOs 3" & Neuron == "Neuron 30")) %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarise(Min_Conc = min(roll_Conc, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

Check that each neuron only has one maximum

```
minint %>%
  count(Mouse, Neuron) %>%
  arrange(-n)
```

```
## # A tibble: 296 x 5
## # Groups:   Group, Experiment, Mouse [8]
##    Group   Experiment Mouse     Neuron        n
##    <chr>   <chr>      <chr>     <chr>     <int>
##  1 Vehicle A          Vehicle 1 Neuron 1      1
##  2 Vehicle A          Vehicle 1 Neuron 10     1
##  3 Vehicle A          Vehicle 1 Neuron 11     1
##  4 Vehicle A          Vehicle 1 Neuron 12     1
##  5 Vehicle A          Vehicle 1 Neuron 13     1
##  6 Vehicle A          Vehicle 1 Neuron 14     1
##  7 Vehicle A          Vehicle 1 Neuron 15     1
##  8 Vehicle A          Vehicle 1 Neuron 16     1
##  9 Vehicle A          Vehicle 1 Neuron 17     1
## 10 Vehicle A          Vehicle 1 Neuron 18     1
## # ... with 286 more rows
```

Model the min concentration by Group

```
minmod <- lmer(Min_Conc ~ Group + (1|Mouse) + (1|Experiment), data = minint)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(minmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Min_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: minint
##
## REML criterion at convergence: 2695.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0932 -0.5683 -0.0874  0.2612  3.7956
##
## Random effects:
##  Groups     Name        Variance  Std.Dev.
##  Mouse      (Intercept) 7.581e+01 8.707e+00
##  Experiment (Intercept) 8.040e-11 8.967e-06
##  Residual               5.233e+02 2.288e+01
## Number of obs: 296, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    41.436      4.790   6.104   8.650  0.00012 ***
## GroupxcTauOs  -25.677      6.773   6.091  -3.791  0.00881 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.707
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

xcTauOs is 25.677 (SE = 6.77) less than vehicle (p = 0.009)

Check the minmod

Because we received a note that the fit was singular, we should make sure that the estimates are stable. We will check the estimates by allowing the model to estimate the coefficients assuming we had 296 observations across the 4 females.

```
minmod_check <- lmer(Min_Conc ~ Group + (1|Experiment), data = minint)

summary(minmod_check)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Min_Conc ~ Group + (1 | Experiment)
##    Data: minint
##
## REML criterion at convergence: 2703
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0245 -0.5604 -0.1487  0.2481  3.7842
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Experiment (Intercept)  43.1     6.565
##  Residual               546.3    23.372
## Number of obs: 296, groups:  Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    42.203      3.844   4.157  10.978 0.000317 ***
## GroupxcTauOs  -27.282      2.847 293.551  -9.584  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.374
```

xcTauOs is 27.28 less than vehicle (SE = 2.85)

These estimates are close, so use the minmod

Plot Min Concentration

Each dot is the minimum concentration value for each neuron.

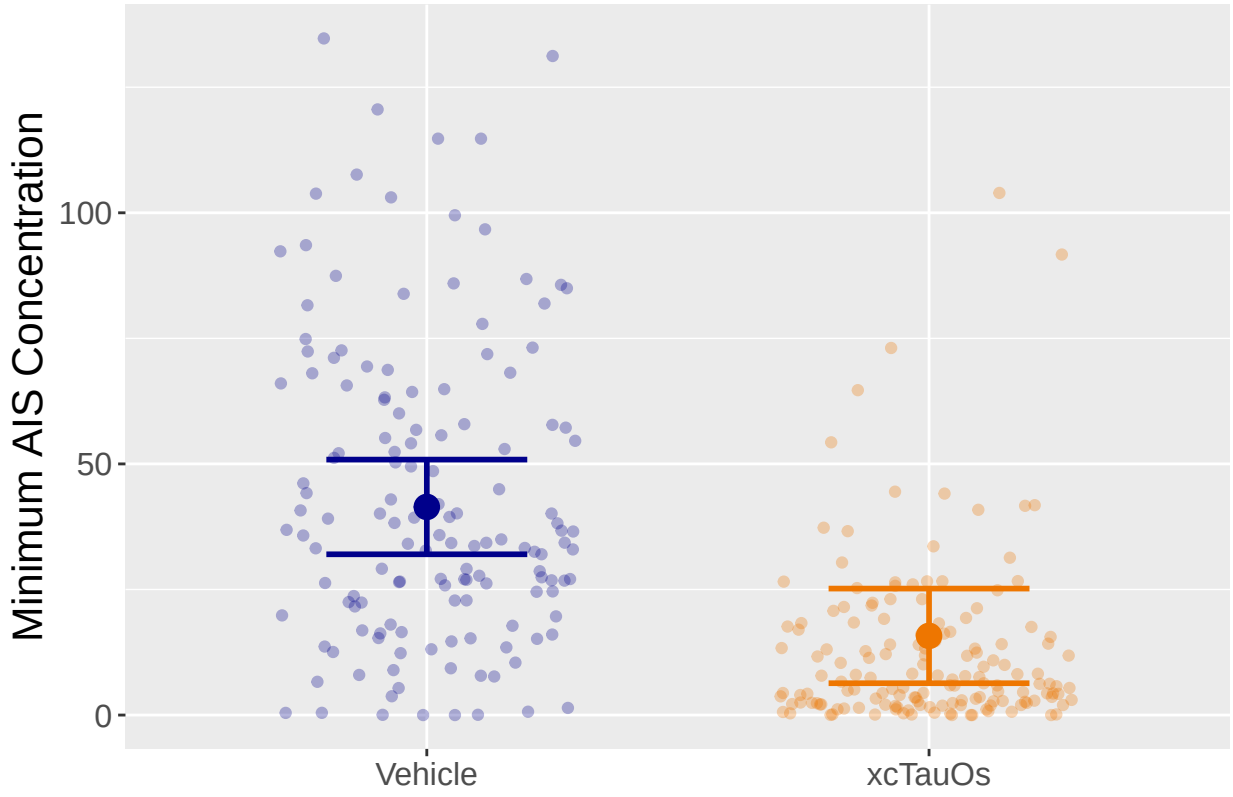
On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```
# return dataset of predicted mean +/- 95%CI
pred_min <- ggeffect(minmod, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)
```

```
pred_min
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      41.4      4.79    32.0       50.9 1
## 2 xcTauOs      15.8      4.79     6.34      25.2 1
```

```
minplot <- minint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Min_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_min,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_min,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Minimum AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

minplot
```




```
ggsave(minplot, filename = "Figures/cell_1E_minconc.png", width = 6, height = 4)
```

Fig 1F

MAPT+/+ (4 independent experiments) Is the AIS Length different between treatments?

Load Data

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/1F- MAPT+-+ AIS Length/", filename), skip = 1) %>%
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Length")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/1F- MAPT+-+ AIS Length/")

Fig1F <- map(my_files, ~readclean(.))
```

```
Fig1F <- bind_rows(Fig1F)

Fig1F
```

```
## # A tibble: 298 x 4
##    `Distance_(microns)` filename                                    Neuron Length
##    <lgl>                <chr>                                       <chr>   <dbl>
##  1 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   16.7
##  2 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   21.1
##  3 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   22.8
##  4 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   18.7
##  5 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   13.8
##  6 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   21.4
##  7 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   15.2
##  8 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   12.2
##  9 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   21.1
## 10 NA                   (1F) MAPT+_+ Vehicle 1 =  6.2.21 WTs DIV ~ Neuro~   16.4
## # ... with 288 more rows
```

Create new variables for Mouse and Group

```
Fig1F <- Fig1F %>%
  select(-`Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\+ ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig1F
```

```
## # A tibble: 298 x 3
##    Mouse     Neuron    Length
##    <chr>     <chr>      <dbl>
##  1 Vehicle 1 Neuron 1    16.7
##  2 Vehicle 1 Neuron 2    21.1
##  3 Vehicle 1 Neuron 3    22.8
##  4 Vehicle 1 Neuron 4    18.7
##  5 Vehicle 1 Neuron 5    13.8
##  6 Vehicle 1 Neuron 6    21.4
##  7 Vehicle 1 Neuron 7    15.2
##  8 Vehicle 1 Neuron 8    12.2
##  9 Vehicle 1 Neuron 9    21.1
## 10 Vehicle 1 Neuron 10   16.4
## # ... with 288 more rows
```

Create treatment group variable

```
Fig1F <- Fig1F %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Length)
```

Create Experiment variable

Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

```
Fig1F <- Fig1F %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D"))

Fig1F
```

```
## # A tibble: 298 x 5
##    Group   Mouse     Neuron    Length Experiment
##    <chr>   <chr>     <chr>      <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1    16.7 A
##  2 Vehicle Vehicle 1 Neuron 2    21.1 A
##  3 Vehicle Vehicle 1 Neuron 3    22.8 A
##  4 Vehicle Vehicle 1 Neuron 4    18.7 A
##  5 Vehicle Vehicle 1 Neuron 5    13.8 A
##  6 Vehicle Vehicle 1 Neuron 6    21.4 A
##  7 Vehicle Vehicle 1 Neuron 7    15.2 A
##  8 Vehicle Vehicle 1 Neuron 8    12.2 A
##  9 Vehicle Vehicle 1 Neuron 9    21.1 A
## 10 Vehicle Vehicle 1 Neuron 10   16.4 A
## # ... with 288 more rows
```

Basic data checks

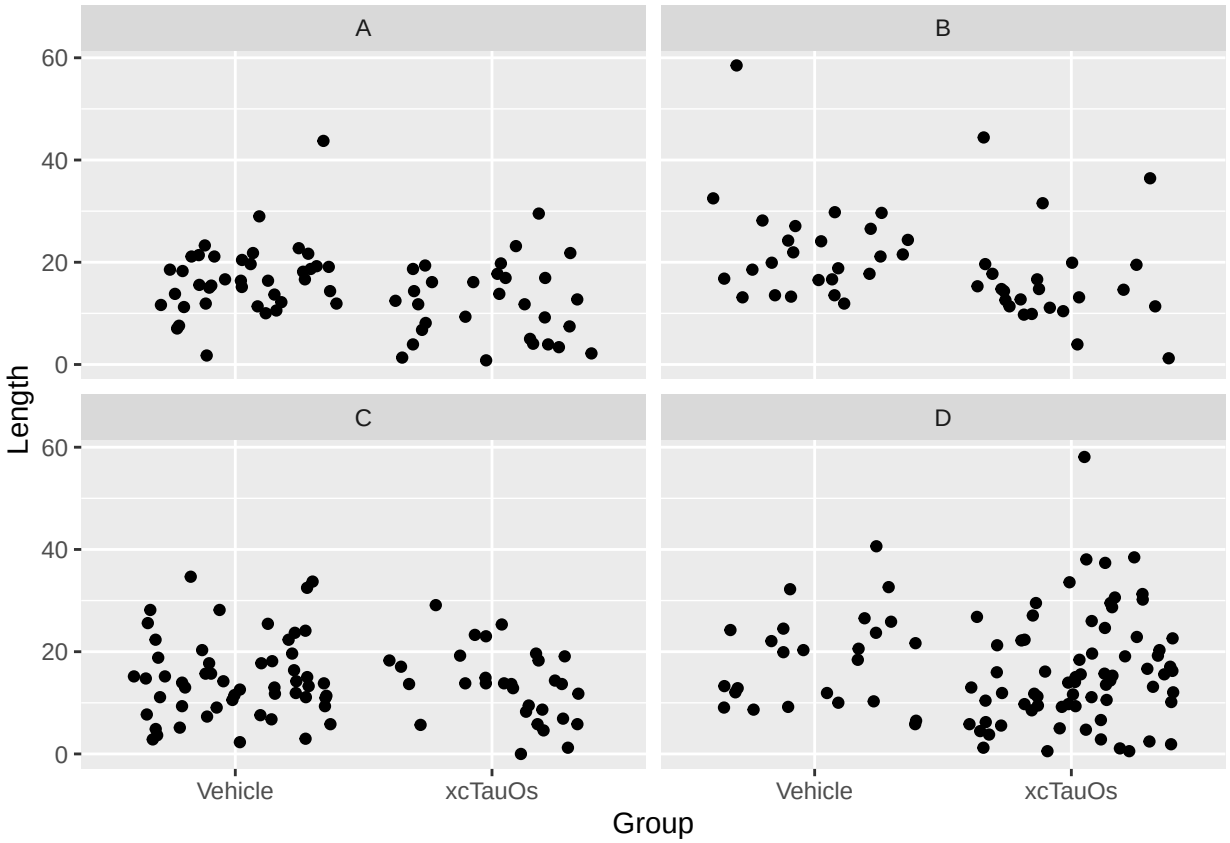
How many neurons per experiment & treatment

```
Fig1F %>%
  count(Mouse)
```

```
## # A tibble: 8 x 2
##   Mouse         n
##   <chr>     <int>
## 1 Vehicle 1    39
## 2 Vehicle 2    25
## 3 Vehicle 3    54
## 4 Vehicle 4    25
## 5 xcTauOs 1    30
## 6 xcTauOs 2    24
## 7 xcTauOs 3    30
## 8 xcTauOs 4    71
```

Exploratory Plot

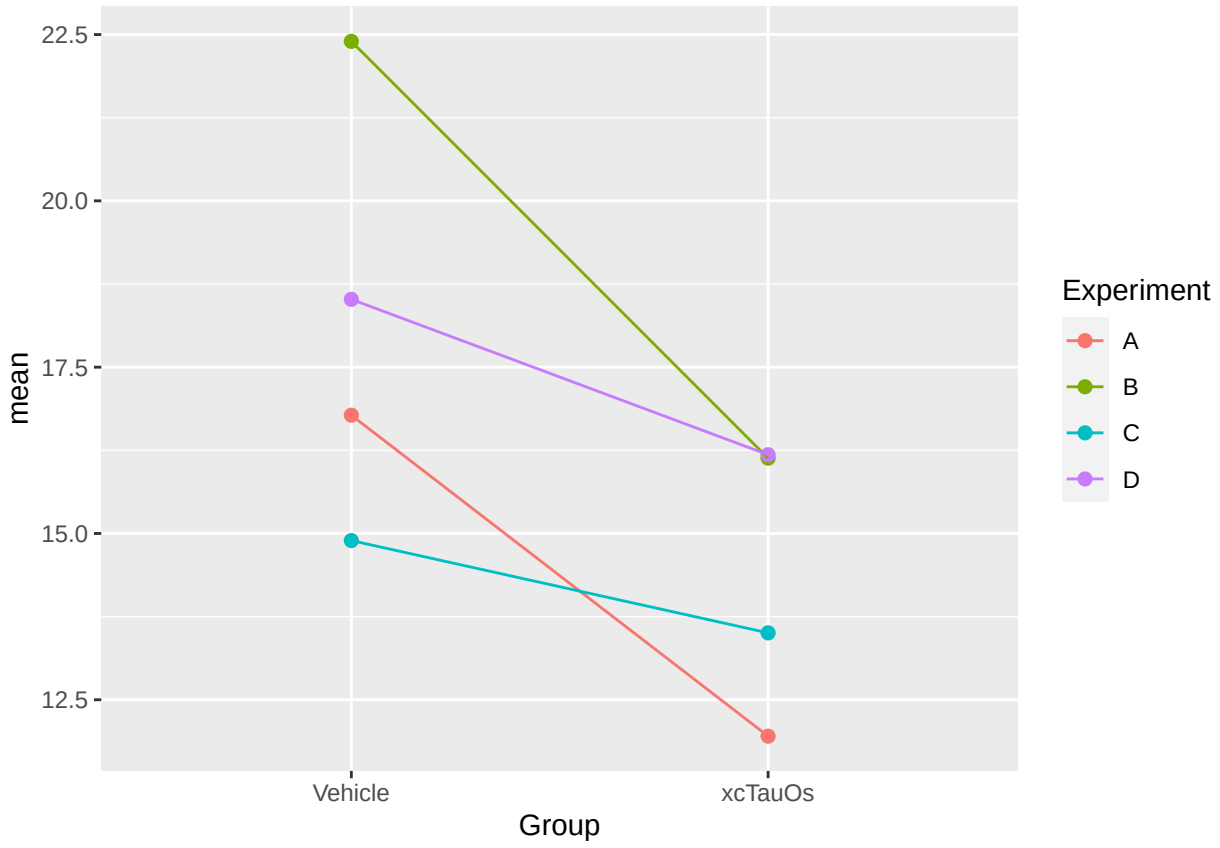
```
Fig1F %>%
  ggplot(aes(Group, Length)) +
  geom_jitter(height = 0) +
  facet_wrap(~Experiment)
```



Calculate the average for each Mouse and then plot those - this is the method that Merci used in the paper

```
Fig1F %>%
  group_by(Experiment, Mouse, Group) %>%
  summarize(mean = mean(Length)) %>%
  ggplot(aes(Group, mean, color = Experiment)) +
  geom_point(size = 2) +
  geom_line(aes(group = Experiment))
```

```
## `summarise()` has grouped output by 'Experiment', 'Mouse'. You can override
## using the `.groups` argument.
```



The mean is being pulled by a high outlier in Experiment A and B Vehicle groups, so I am not sure the mean is a good indicator of the actual situation

Model

```
Fig1Fmod <- lmer(Length ~ Group + (1|Mouse) + (1|Experiment), data = Fig1F)
summary(Fig1Fmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Length ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: Fig1F
##
## REML criterion at convergence: 2144.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.7744 -0.6655 -0.0986  0.5108  4.7931
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)  0.1149  0.3389
##  Experiment (Intercept)  5.1044  2.2593
##  Residual               77.9536  8.8291
```

```
## Number of obs: 298, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)    17.904      1.370  4.005  13.070 0.000196 ***
## GroupxcTauOs   -3.284      1.098  2.541  -2.992 0.071503 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.406
```

No difference in Length (p = 0.07)

Plot

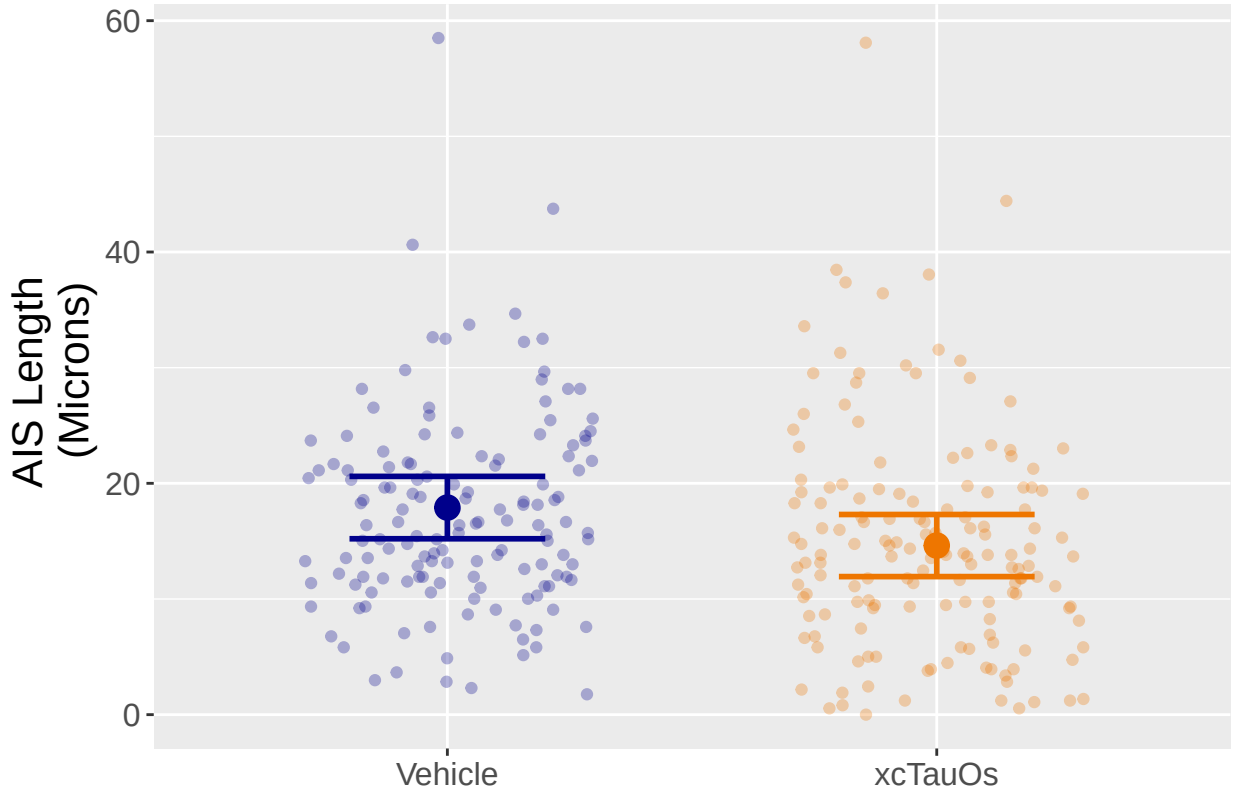
Jitter with 95% CI (like human data) - vehicle = blue - tau = orange

```
# return dataset of predicted mean +/- 95%CI
pred_length <- ggeffect(Fig1Fmod, terms = "Group") %>%
  as_tibble() %>%
  rename(Group = x)

pred_length
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      17.9      1.37     15.2      20.6 1
## 2 xcTauOs      14.6      1.36     11.9      17.3 1
```

```
lengthplot <- Fig1F %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Length,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_length,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_length,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "AIS Length \n (Microns)", x = "") +
```

```
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

lengthplot
```




```
ggsave(lengthplot, filename = "Figures/cell_1F.png", width = 6, height = 4)
```

Fig 2D

MAPT-/- (5 independent experiments)

Is the Soma-AIS gap different between vehicle and xcTauOs

Load Data

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/2D - MAPT--- Soma-AIS Gap/", filename), skip = 1) %>%
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
```

```
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Gap")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/2D - MAPT--- Soma-AIS Gap/")

Fig2D <- map(my_files, ~readclean(.))
Fig2D <- bind_rows(Fig2D)

Fig2D
```

```
## # A tibble: 368 x 4
##    `Distance_(microns)` filename                                    Neuron   Gap
##    <lgl>                <chr>                                       <chr>  <dbl>
##  1 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  3.74
##  2 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  6.55
##  3 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  4.54
##  4 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  4.47
##  5 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  5.86
##  6 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  6.15
##  7 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  8.29
##  8 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  5.34
##  9 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  5.74
## 10 NA                   (2D) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV ~ Neuro~  2.60
## # ... with 358 more rows
```

Create new variables for Mouse and Group

```
Fig2D <- Fig2D %>%
  select(-`Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\- ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig2D
```

```
## # A tibble: 368 x 3
##    Mouse     Neuron      Gap
##    <chr>     <chr>     <dbl>
##  1 Vehicle 1 Neuron 1   3.74
##  2 Vehicle 1 Neuron 2   6.55
##  3 Vehicle 1 Neuron 3   4.54
##  4 Vehicle 1 Neuron 4   4.47
##  5 Vehicle 1 Neuron 5   5.86
##  6 Vehicle 1 Neuron 6   6.15
##  7 Vehicle 1 Neuron 7   8.29
##  8 Vehicle 1 Neuron 8   5.34
##  9 Vehicle 1 Neuron 9   5.74
## 10 Vehicle 1 Neuron 10  2.60
## # ... with 358 more rows
```

Create treatment group variable

```
Fig2D <- Fig2D %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Gap)
```

Create Experiment variable

Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

```
Fig2D <- Fig2D %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D",
                                str_detect(Mouse,"5") ~ "E"))

Fig2D
```

```
## # A tibble: 368 x 5
##    Group   Mouse     Neuron      Gap Experiment
##    <chr>   <chr>     <chr>     <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1   3.74 A
##  2 Vehicle Vehicle 1 Neuron 2   6.55 A
##  3 Vehicle Vehicle 1 Neuron 3   4.54 A
##  4 Vehicle Vehicle 1 Neuron 4   4.47 A
##  5 Vehicle Vehicle 1 Neuron 5   5.86 A
##  6 Vehicle Vehicle 1 Neuron 6   6.15 A
##  7 Vehicle Vehicle 1 Neuron 7   8.29 A
##  8 Vehicle Vehicle 1 Neuron 8   5.34 A
##  9 Vehicle Vehicle 1 Neuron 9   5.74 A
## 10 Vehicle Vehicle 1 Neuron 10  2.60 A
## # ... with 358 more rows
```

Basic data checks

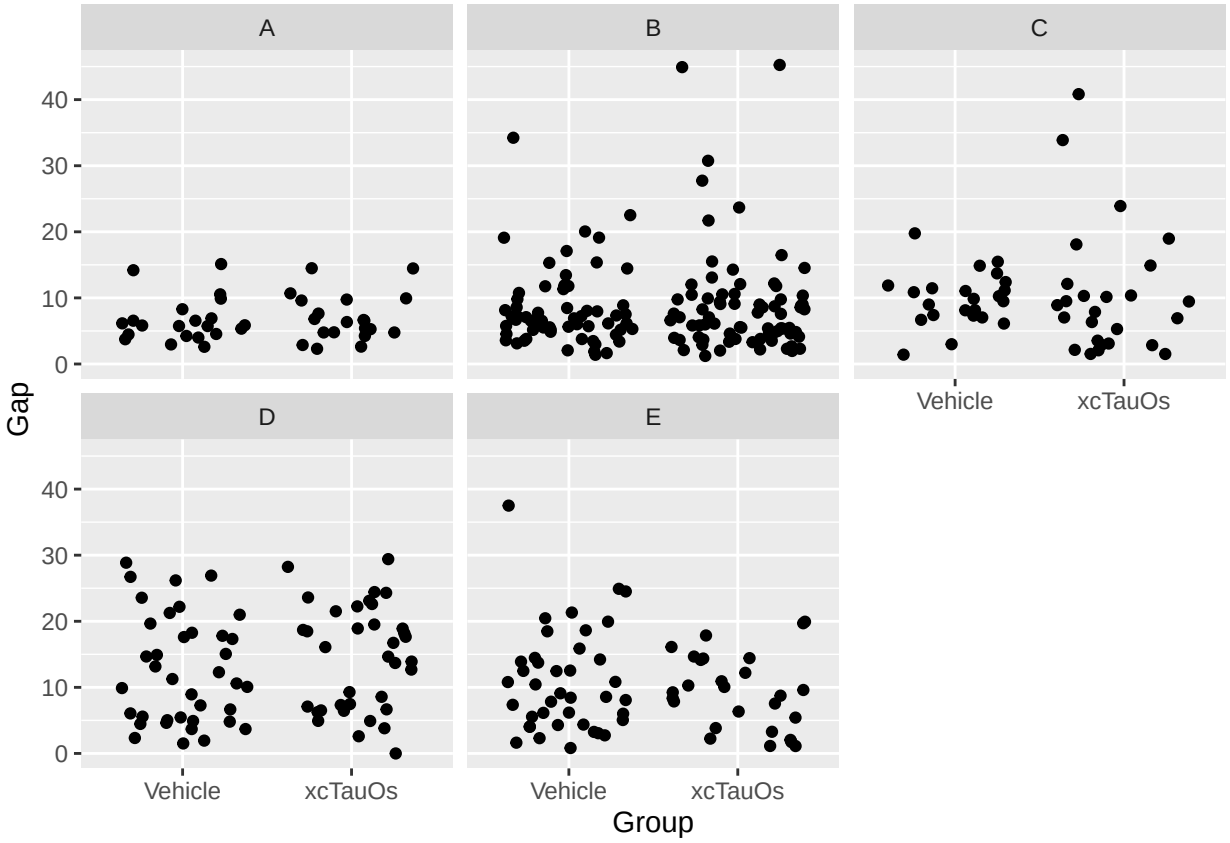
How many neurons per experiment & treatment

```
Fig2D %>%
  count(Mouse)
```

```
## # A tibble: 10 x 2
##    Mouse         n
##    <chr>     <int>
##  1 Vehicle 1    21
##  2 Vehicle 2    63
##  3 Vehicle 3    23
##  4 Vehicle 4    38
##  5 Vehicle 5    40
##  6 xcTauOs 1    20
##  7 xcTauOs 2    74
##  8 xcTauOs 3    26
##  9 xcTauOs 4    36
## 10 xcTauOs 5    27
```

Exploratory plot

```
Fig2D %>%
  ggplot(aes(Group, Gap)) +
  geom_jitter() +
  facet_wrap(~Experiment)
```



Model

```
Fig2Dmod <- lmer(Gap ~ Group + (1|Mouse) + (1|Experiment), data = Fig2D)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(Fig2Dmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Gap ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: Fig2D
##
## REML criterion at convergence: 2490.8
```

```
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8772 -0.6559 -0.1898  0.4257  5.1108
##
## Random effects:
##  Groups     Name        Variance  Std.Dev.
##  Mouse      (Intercept) 6.022e-14 2.454e-07
##  Experiment (Intercept) 5.072e+00 2.252e+00
##  Residual               5.024e+01 7.088e+00
## Number of obs: 368, groups:  Mouse, 10; Experiment, 5
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)    9.6746     1.1426   4.8424   8.467 0.000441 ***
## GroupxcTauOs   0.5232     0.7423 362.7351   0.705 0.481350
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.318
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

xcTauOs is 0.5232 higher (SE = 0.7423) (p = 0.481)

No difference in Gap

Check the model by removing the random effect of Mouse

```
Fig2Dmod_check <- lmer(Gap ~ Group + (1|Experiment), data = Fig2D)
summary(Fig2Dmod_check)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Gap ~ Group + (1 | Experiment)
##    Data: Fig2D
##
## REML criterion at convergence: 2490.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8772 -0.6559 -0.1898  0.4257  5.1108
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Experiment (Intercept)  5.072   2.252
##  Residual               50.239   7.088
## Number of obs: 368, groups:  Experiment, 5
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)    9.6746     1.1426   4.8424   8.467 0.000441 ***
```

```
## GroupxcTauOs   0.5232     0.7423 362.7351   0.705 0.481349
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.318
```

Same estimates. Use the Fig2Dmod anyway

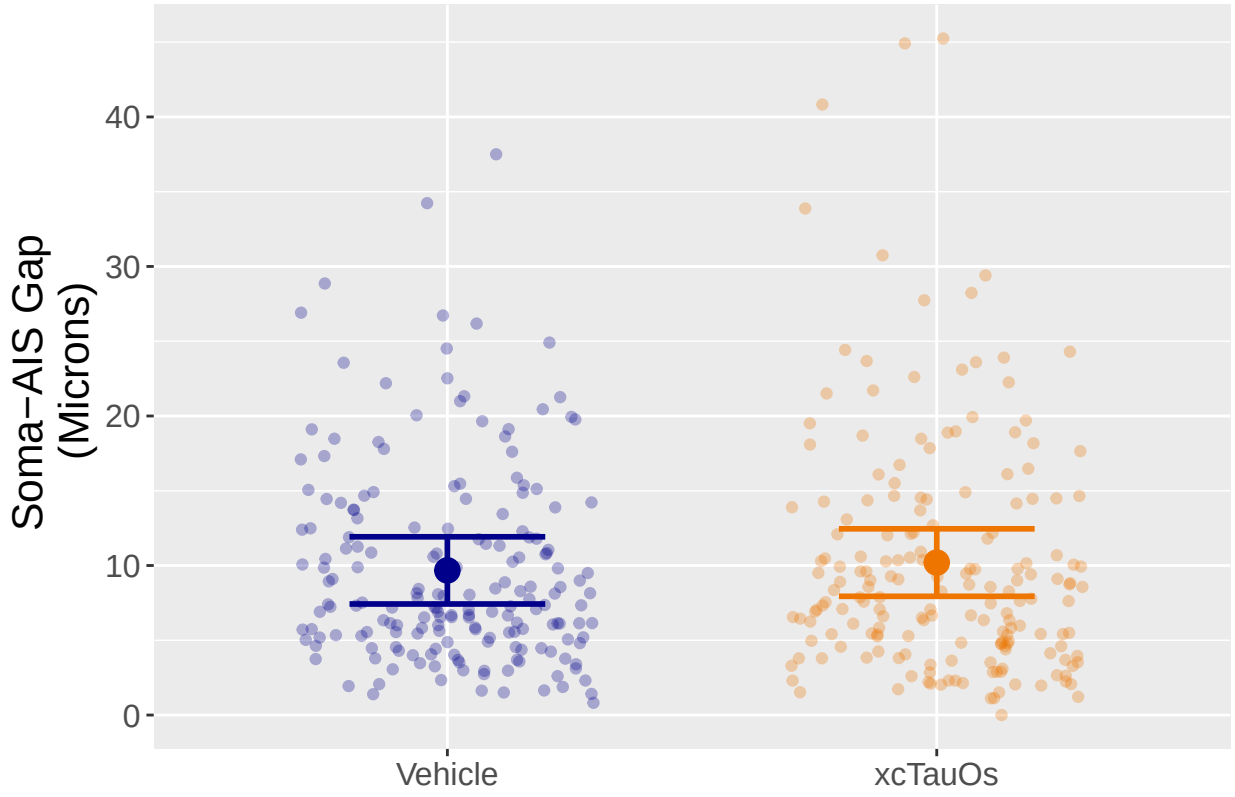
Plot

Jitter with 95% CI (like human data) - vehicle = blue - tau = orange

```
# return dataset of predicted mean +/- 95%CI
pred_gap <- ggeffect(Fig2Dmod, terms = "Group") %>%
  as_tibble() %>%
  rename(Group = x)

pred_gap
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      9.67      1.14     7.43      11.9 1
## 2 xcTauOs     10.2       1.15     7.94      12.5 1
```

```
gapplot <- Fig2D %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Gap,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_gap,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_gap,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Soma-AIS Gap \n (Microns)", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

gapplot
```




```
ggsave(gapplot, filename = "Figures/cell_2D.png", width = 6, height = 4)
```

Fig 2E

MAPT-/- (5 independent experiments)

Is the AIS concentration different between vehicle and xcTauOs

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/2E - MAPT--- AIS Concentration/", filename), skip = 1) %>%
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Concentration")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/2E - MAPT--- AIS Concentration/")

Fig2E <- map(my_files, ~readclean(.))
```

```
## New names:
```

```
## * `` -> `...80`
## * `` -> `...81`
## * `` -> `...82`
```

```
Fig2E <- bind_rows(Fig2E)
```

```
## New names:
## * `...80` -> `...2`
## * `...81` -> `...3`
## * `...82` -> `...4`
```

```
Fig2E
```

```
## # A tibble: 193,976 x 7
##    `Distance_(microns)` filename          Neuron Concentration ...2  ...3  ...4
##                   <dbl> <chr>             <chr>          <dbl> <chr> <chr> <chr>
##  1                    0 (2E) MAPT-_- Veh~ Neuro~         16.8  <NA>  <NA>  <NA>
##  2                    0 (2E) MAPT-_- Veh~ Neuro~          5.09 <NA>  <NA>  <NA>
##  3                    0 (2E) MAPT-_- Veh~ Neuro~          6.14 <NA>  <NA>  <NA>
##  4                    0 (2E) MAPT-_- Veh~ Neuro~         12.3  <NA>  <NA>  <NA>
##  5                    0 (2E) MAPT-_- Veh~ Neuro~          8.20 <NA>  <NA>  <NA>
##  6                    0 (2E) MAPT-_- Veh~ Neuro~          8.56 <NA>  <NA>  <NA>
##  7                    0 (2E) MAPT-_- Veh~ Neuro~         18.7  <NA>  <NA>  <NA>
##  8                    0 (2E) MAPT-_- Veh~ Neuro~          8.20 <NA>  <NA>  <NA>
##  9                    0 (2E) MAPT-_- Veh~ Neuro~          6.49 <NA>  <NA>  <NA>
## 10                    0 (2E) MAPT-_- Veh~ Neuro~         25.3  <NA>  <NA>  <NA>
## # ... with 193,966 more rows
```

```
# drop empty columns
Fig2E <- Fig2E %>% select(-contains("..."))
```

Create new variables for Mouse and Group

```
Fig2E <- Fig2E %>%
  rename(Dist = `Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\- ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig2E
```

```
## # A tibble: 193,976 x 4
##     Dist Mouse     Neuron    Concentration
##    <dbl> <chr>     <chr>             <dbl>
##  1     0 Vehicle 1 Neuron 1          16.8
##  2     0 Vehicle 1 Neuron 2           5.09
##  3     0 Vehicle 1 Neuron 3           6.14
##  4     0 Vehicle 1 Neuron 4          12.3
##  5     0 Vehicle 1 Neuron 5           8.20
##  6     0 Vehicle 1 Neuron 6           8.56
##  7     0 Vehicle 1 Neuron 7          18.7
##  8     0 Vehicle 1 Neuron 8           8.20
```

```
##  9     0 Vehicle 1 Neuron 9           6.49
## 10     0 Vehicle 1 Neuron 10         25.3
## # ... with 193,966 more rows
```

Create treatment group variable

```
Fig2E <- Fig2E %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Dist, Concentration)
```

Create Experiment variable

Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

```
Fig2E <- Fig2E %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D",
                                str_detect(Mouse,"5") ~ "E"))

Fig2E
```

```
## # A tibble: 193,976 x 6
##    Group   Mouse     Neuron     Dist Concentration Experiment
##    <chr>   <chr>     <chr>     <dbl>         <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1      0         16.8  A
##  2 Vehicle Vehicle 1 Neuron 2      0          5.09 A
##  3 Vehicle Vehicle 1 Neuron 3      0          6.14 A
##  4 Vehicle Vehicle 1 Neuron 4      0         12.3  A
##  5 Vehicle Vehicle 1 Neuron 5      0          8.20 A
##  6 Vehicle Vehicle 1 Neuron 6      0          8.56 A
##  7 Vehicle Vehicle 1 Neuron 7      0         18.7  A
##  8 Vehicle Vehicle 1 Neuron 8      0          8.20 A
##  9 Vehicle Vehicle 1 Neuron 9      0          6.49 A
## 10 Vehicle Vehicle 1 Neuron 10     0         25.3  A
## # ... with 193,966 more rows
```

Clean dataset

Drop observations where Concentration is missing because the neuron wasn't that long

```
Fig2E <- Fig2E %>%
  arrange(Group, Experiment, Mouse, Neuron, Dist) %>%
  drop_na(Concentration)
```

Basic data checks

How many neurons per experiment

```
Fig2E %>%
  count(Mouse)
```

```
## # A tibble: 10 x 2
##    Mouse         n
##    <chr>     <int>
##  1 Vehicle 1  4815
##  2 Vehicle 2  8139
##  3 Vehicle 3  3241
##  4 Vehicle 4  3997
##  5 Vehicle 5  8667
##  6 xcTauOs 1  4931
##  7 xcTauOs 2  9058
##  8 xcTauOs 3  3176
##  9 xcTauOs 4  4474
## 10 xcTauOs 5  5464
```

Exploratory plot

- Make rolling average plot

A rolling average across 3-distance observations works nicely to show the trend.

I used the `rollapply()` function rather than the more standard `rollmean()` function because `rollmean()` has no way to remove NAs.

```
Fig2E <- Fig2E %>%
  group_by(Mouse, Neuron) %>%
  mutate(roll_Conc = rollapply(Concentration, 3, mean, na.rm = TRUE, fill = NA)) %>%
  ungroup()

Fig2E
```

```
## # A tibble: 55,962 x 7
##    Group   Mouse     Neuron    Dist Concentration Experiment roll_Conc
##    <chr>   <chr>     <chr>    <dbl>         <dbl> <chr>          <dbl>
##  1 Vehicle Vehicle 1 Neuron 1 0              16.8 A               NA
##  2 Vehicle Vehicle 1 Neuron 1 0.135          16.7 A               16.8
##  3 Vehicle Vehicle 1 Neuron 1 0.271          17.0 A               17.1
##  4 Vehicle Vehicle 1 Neuron 1 0.406          17.4 A               17.5
##  5 Vehicle Vehicle 1 Neuron 1 0.542          18.1 A               18.0
##  6 Vehicle Vehicle 1 Neuron 1 0.677          18.4 A               18.4
##  7 Vehicle Vehicle 1 Neuron 1 0.813          18.6 A               18.7
##  8 Vehicle Vehicle 1 Neuron 1 0.948          18.9 A               18.9
##  9 Vehicle Vehicle 1 Neuron 1 1.08           19.1 A               19.1
## 10 Vehicle Vehicle 1 Neuron 1 1.22           19.3 A               19.3
## # ... with 55,952 more rows
```

One line per mouse, rolling average averaged over all neurons at a given dist.

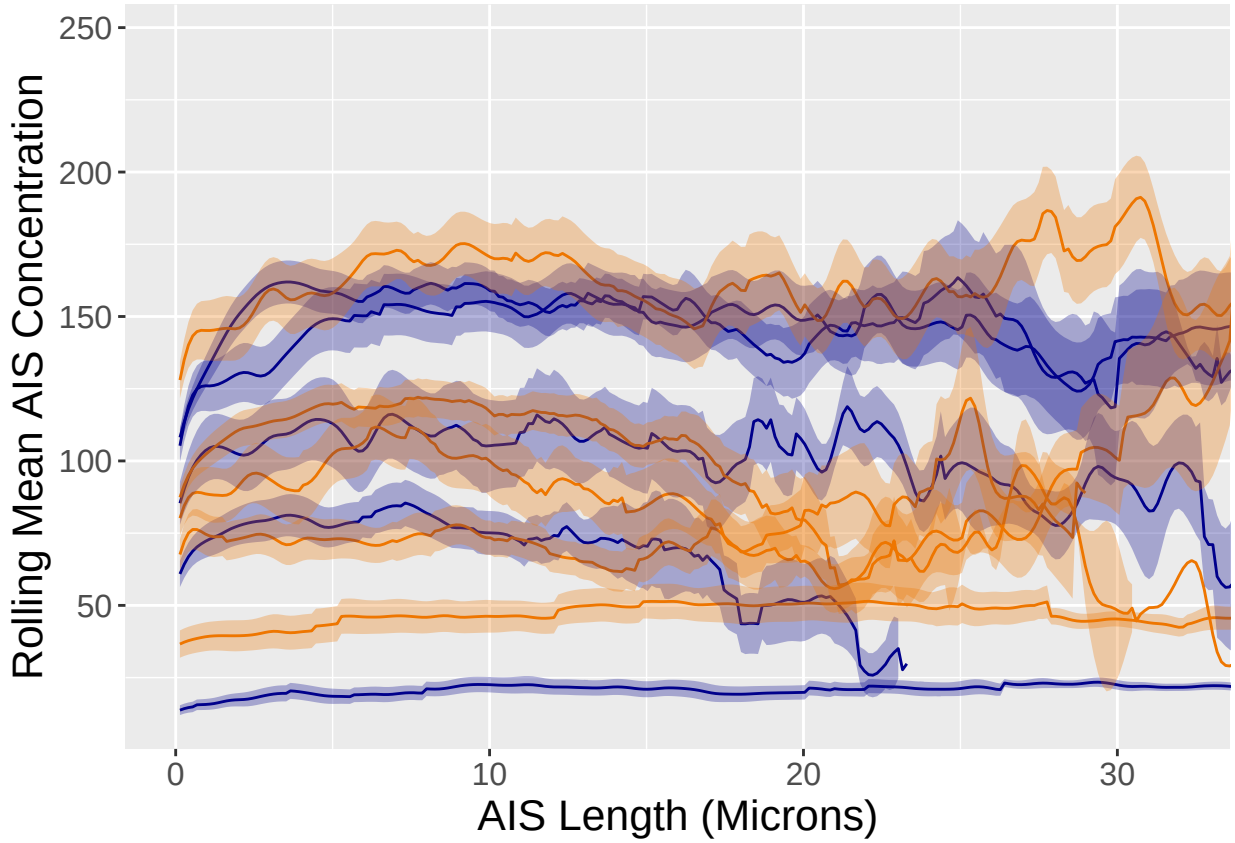
The rolling average is the average of a 3-dist chunk.

```
rollavgdat <- Fig2E %>%
  group_by(Group, Mouse, Dist) %>%
  summarize(Mean_Conc = mean(roll_Conc, na.rm = TRUE),
            sd_Conc = sd(roll_Conc, na.rm = TRUE),
            n_neurons = n(),
            se_Conc = sd_Conc/sqrt(n_neurons))
```

```
## `summarise()` has grouped output by 'Group', 'Mouse'. You can override using
## the `.groups` argument.
```

```
lineplot <- rollavgdat %>%
  ggplot(aes(Dist, Mean_Conc)) +
  geom_line(aes(group = Mouse, color = Group)) +
  geom_ribbon(aes(ymin = Mean_Conc-se_Conc,
                  ymax = Mean_Conc+se_Conc,
                  group = Mouse, fill = Group), alpha = .3) +
  scale_fill_manual(values = c("darkblue", "darkorange2")) +
  scale_color_manual(values = c("darkblue", "darkorange2")) +
  labs(y = "Rolling Mean AIS Concentration", x = "AIS Length (Microns)") +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size =  12)) +
  coord_cartesian(xlim = c(0,32))

lineplot
```

```
## Warning: Removed 20 row(s) containing missing values (geom_path).
```




```
ggsave(lineplot, filename = "Figures/cell_2E_distVintensity.png", width =  6, height = 4)
```

```
## Warning: Removed 20 row(s) containing missing values (geom_path).
```

Model splines

Splines are a way to fit a non-linear curve to data to understand how the relationship between Distance and Concentration changes for xcTauOs v. Vehicle.

Try basic natural splines model

```
splinemod <- lmer(Concentration ~ ns(Dist, df = 5) + Group + (1|Mouse) + (1|Experiment), data = Fig2E)
```

Now try with interaction term

```
splinemod2 <- lmer(Concentration ~ ns(Dist, df = 5) * Group + (1|Mouse) + (1| Experiment), data = Fig2E
```

See if there is a difference in model fit between splinemod and splinemod2

```
anova(splinemod2, splinemod)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: Fig2E
## Models:
## splinemod: Concentration ~ ns(Dist, df = 5) + Group + (1 | Mouse) + (1 | Experiment)
## splinemod2: Concentration ~ ns(Dist, df = 5) * Group + (1 | Mouse) + (1 | Experiment)
##            npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## splinemod    10 594748 594837 -297364   594728
## splinemod2   15 594328 594462 -297149   594298 429.62  5  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

splinemod2 fits better than splinemod, so use that

Interpretation of splinemod2, use emmeans to find average difference

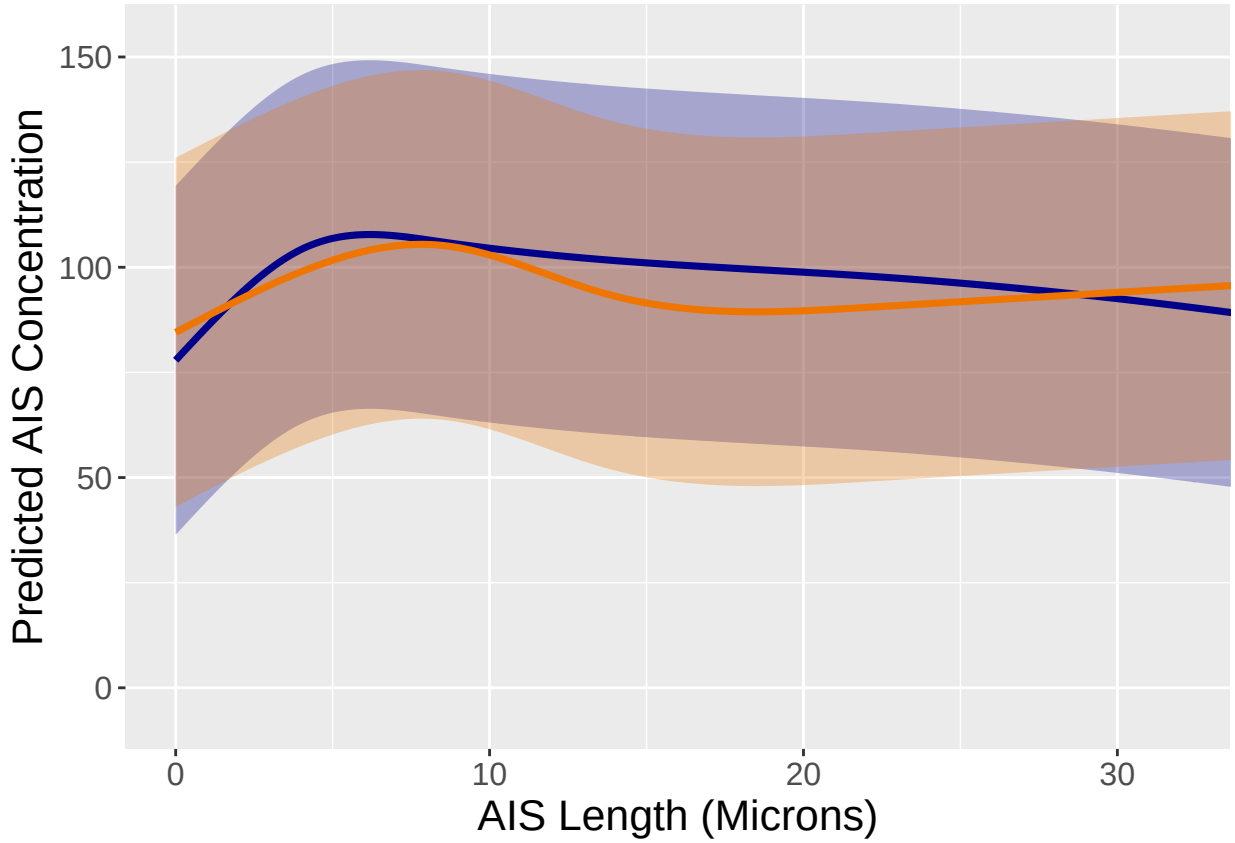
```
pairs(emmeans(splinemod2, specs = "Group"))
```

```
##  contrast          estimate   SE  df z.ratio p.value
##  Vehicle - xcTauOs     8.54 11.3 Inf   0.753  0.4514
##
## Degrees-of-freedom method: asymptotic
```

On average xcTauOs has AIS concentration 8.54 lower than vehicle (p = 0.451)

Plot splines

```
splinesplot <- ggpredict(splinemod2, terms = c("Dist [all]", "Group")) %>%
  as_tibble() %>%
  rename(Dist = x,
         Group = group) %>%
  ggplot(aes(Dist, predicted, color = Group)) +
  #facet_wrap(~Group, nrow = 2) +
  geom_ribbon(aes(ymin = conf.low,
                  ymax = conf.high,
                  fill = Group),
              alpha = .3,
              color = NA) +
  geom_line(lwd = 1.25) +
  scale_color_manual(values = c("darkblue", "darkorange2")) +
  scale_fill_manual(values = c("darkblue", "darkorange2")) +
  labs(x = "AIS Length (Microns)",
       y = "Predicted AIS Concentration", color = "") +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12)) +
  coord_cartesian(xlim = c(0,32))

splinesplot
```




```
ggsave(splinesplot, filename = "Figures/cell_2E_splines.png", width = 6, height = 4)
```

Model mean concentration

Create average intensity dataset. For each neuron, what is the average intensity across the whole distance that was measured

```
avgint <- Fig2E %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarize(Mean_Conc = mean(Concentration),
            n = n()) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

```
avgint
```

```
## # A tibble: 368 x 6
##    Group   Experiment Mouse     Neuron    Mean_Conc     n
##    <chr>   <chr>      <chr>     <chr>         <dbl> <int>
##  1 Vehicle A          Vehicle 1 Neuron 1      17.2    150
##  2 Vehicle A          Vehicle 1 Neuron 10     29.2    255
```

```
##  3 Vehicle A          Vehicle 1 Neuron 11     14.7    156
##  4 Vehicle A          Vehicle 1 Neuron 12      3.13   150
##  5 Vehicle A          Vehicle 1 Neuron 13     24.3    507
##  6 Vehicle A          Vehicle 1 Neuron 14     25.4    260
##  7 Vehicle A          Vehicle 1 Neuron 15      1.82    28
##  8 Vehicle A          Vehicle 1 Neuron 16     29.3    303
##  9 Vehicle A          Vehicle 1 Neuron 17     24.1    466
## 10 Vehicle A          Vehicle 1 Neuron 18     24.4    377
## # ... with 358 more rows
```

Model the mean concentration by Group

```
avgintmodB <- lmer(Mean_Conc ~ Group + (1|Mouse) + (1|Experiment), data = avgint)

summary(avgintmodB)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Mean_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: avgint
##
## REML criterion at convergence: 3790
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.77502 -0.70472  0.02006  0.61630  2.68397
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)  340.4   18.45
##  Experiment (Intercept) 1865.9   43.20
##  Residual               1666.4   40.82
## Number of obs: 368, groups:  Mouse, 10; Experiment, 5
##
## Fixed effects:
##              Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)    95.055     21.253  4.726   4.472   0.0075 **
## GroupxcTauOs    1.503     12.547  4.212   0.120   0.9101
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.294
```

No difference

Plot Mean Concentration

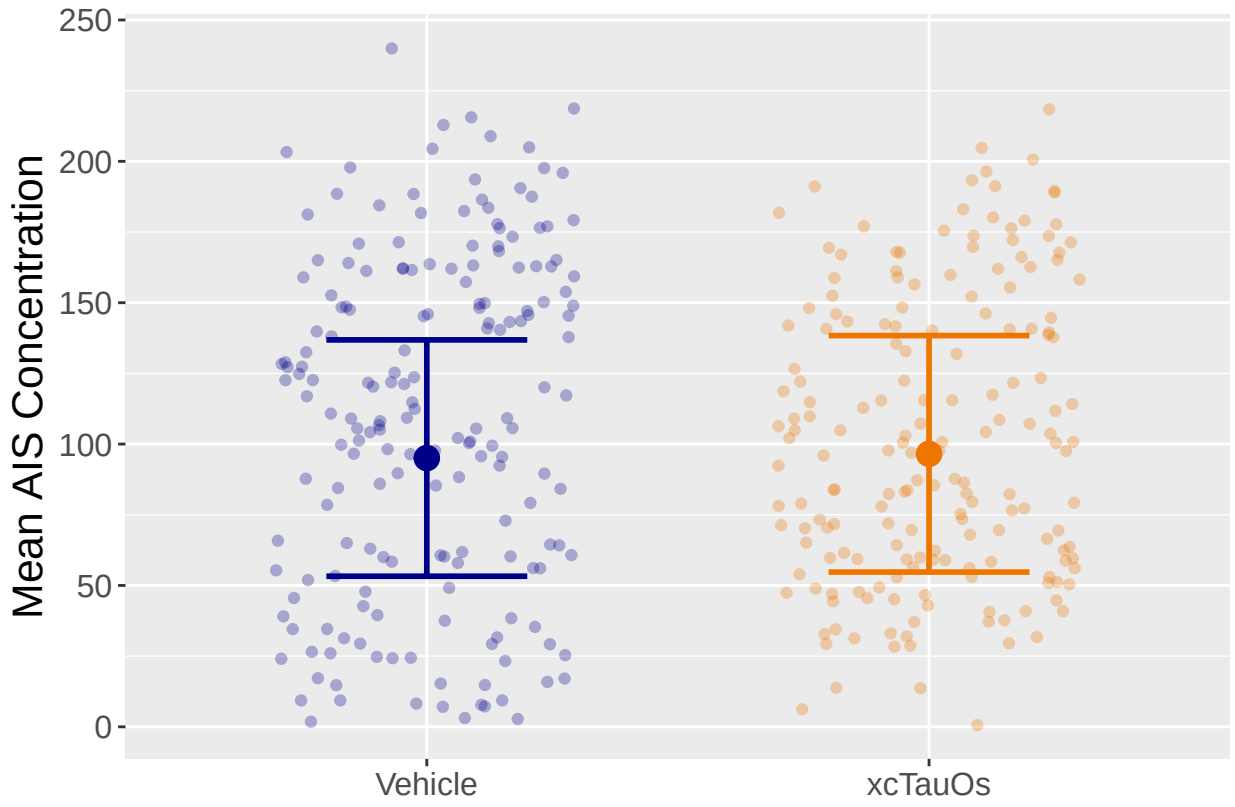
Each dot is an average of all of the concentration values for each neuron.

On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```r
# return dataset of predicted mean +/- 95%CI
pred_mean_avgintmod <- ggeffect(avgintmodB, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)

pred_mean_avgintmod
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      95.1      21.3     53.3      137. 1
## 2 xcTauOs      96.6      21.3     54.7      138. 1
```

```r
meanint_supp_plot <- avgint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Mean_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_mean_avgintmod,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_mean_avgintmod,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Mean AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

meanint_supp_plot
```




```
ggsave(meanint_supp_plot, filename = "Figures/cell_2E_meanconc.png", width = 6, height = 4)
```

Model max concentration

Create max concentration variable

Define the maximum concentration based on the rolling average concentration for each neuron since that worked better for the human data.

```
maxint <- Fig2E %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarise(Max_Conc = max(roll_Conc, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

Check that each neuron only has one maximum

```
maxint %>%
  count(Mouse, Neuron) %>%
  arrange(-n)
```

```
## # A tibble: 368 x 5
## # Groups:   Group, Experiment, Mouse [10]
```

```
##    Group   Experiment Mouse     Neuron        n
##    <chr>   <chr>      <chr>     <chr>     <int>
##  1 Vehicle A          Vehicle 1 Neuron 1      1
##  2 Vehicle A          Vehicle 1 Neuron 10     1
##  3 Vehicle A          Vehicle 1 Neuron 11     1
##  4 Vehicle A          Vehicle 1 Neuron 12     1
##  5 Vehicle A          Vehicle 1 Neuron 13     1
##  6 Vehicle A          Vehicle 1 Neuron 14     1
##  7 Vehicle A          Vehicle 1 Neuron 15     1
##  8 Vehicle A          Vehicle 1 Neuron 16     1
##  9 Vehicle A          Vehicle 1 Neuron 17     1
## 10 Vehicle A          Vehicle 1 Neuron 18     1
## # ... with 358 more rows
```

Model the max concentration by Group

```
maxmod <- lmer(Max_Conc ~ Group + (1|Mouse) + (1|Experiment), data = maxint)

summary(maxmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Max_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: maxint
##
## REML criterion at convergence: 3954
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9520 -0.7033  0.1152  0.6196  2.0189
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)  448.8   21.19
##  Experiment (Intercept) 3459.9   58.82
##  Residual               2609.0   51.08
## Number of obs: 368, groups:  Mouse, 10; Experiment, 5
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   132.895     28.250   4.541   4.704  0.00679 **
## GroupxcTauOs    3.962     14.586   4.202   0.272  0.79873
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.257
```

No difference

Plot Max Concentration

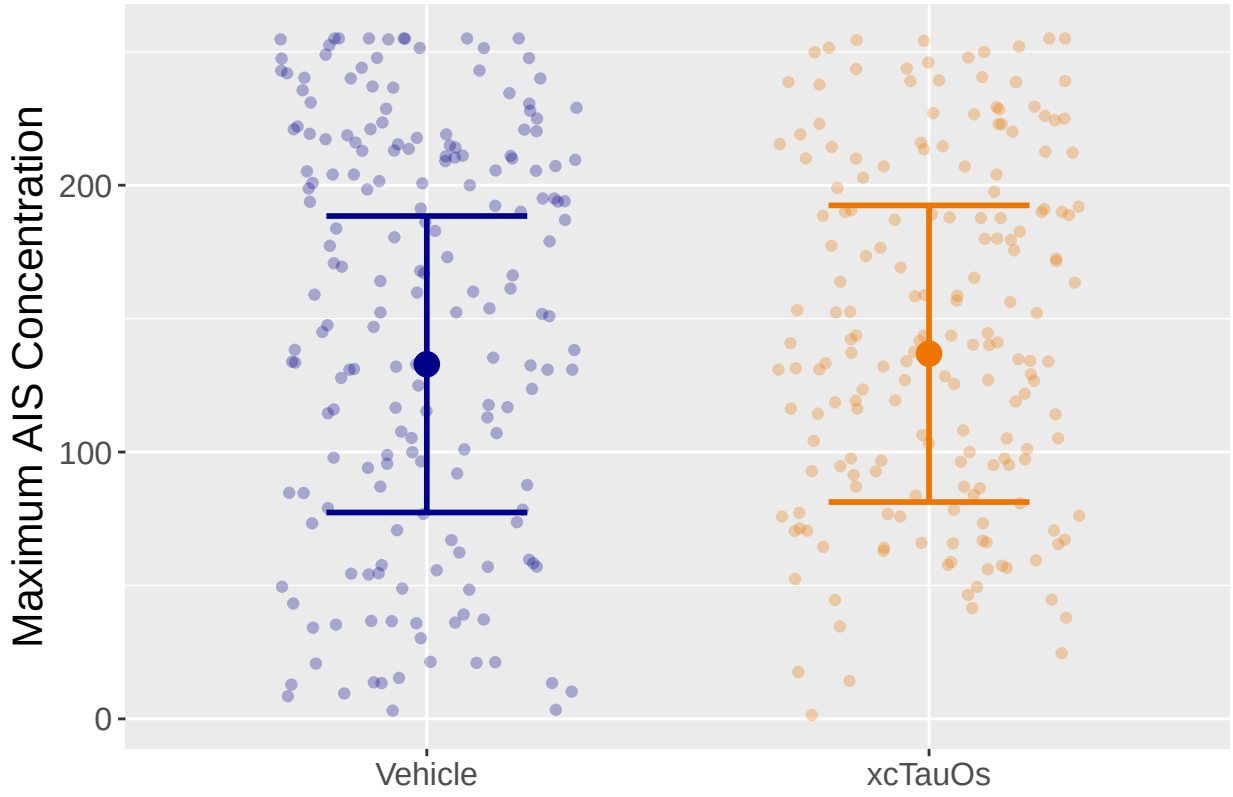
Each dot is the maximum concentration values for each neuron.

On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```
# return dataset of predicted mean +/- 95%CI
pred_max <- ggeffect(maxmod, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)

pred_max
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      133.      28.3     77.3      188. 1
## 2 xcTauOs      137.      28.3     81.3      192. 1
```

```
maxplot <- maxint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Max_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_max,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_max,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Maximum AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

maxplot
```




```
ggsave(maxplot, filename = "Figures/cell_2E_maxconc.png", width = 6, height = 4)
```

Model min concentration

Create min concentration variable

Define the minimum concentration based on the rolling average concentration for each neuron since that worked better for the human data.

```
minint <- Fig2E %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarise(Min_Conc = min(roll_Conc, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

Check that each neuron only has one maximum

```
minint %>%
  count(Mouse, Neuron) %>%
  arrange(-n)
```

```
## # A tibble: 368 x 5
## # Groups:   Group, Experiment, Mouse [10]
```

```
##    Group   Experiment Mouse     Neuron        n
##    <chr>   <chr>      <chr>     <chr>     <int>
##  1 Vehicle A          Vehicle 1 Neuron 1      1
##  2 Vehicle A          Vehicle 1 Neuron 10     1
##  3 Vehicle A          Vehicle 1 Neuron 11     1
##  4 Vehicle A          Vehicle 1 Neuron 12     1
##  5 Vehicle A          Vehicle 1 Neuron 13     1
##  6 Vehicle A          Vehicle 1 Neuron 14     1
##  7 Vehicle A          Vehicle 1 Neuron 15     1
##  8 Vehicle A          Vehicle 1 Neuron 16     1
##  9 Vehicle A          Vehicle 1 Neuron 17     1
## 10 Vehicle A          Vehicle 1 Neuron 18     1
## # ... with 358 more rows
```

Model the min concentration by Group

```
minmod <- lmer(Min_Conc ~ Group + (1|Mouse) + (1|Experiment), data = minint)

summary(minmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Min_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: minint
##
## REML criterion at convergence: 3589.7
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.2547 -0.7224 -0.0407  0.6337  3.2118
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)  96.61    9.829
##  Experiment (Intercept) 368.71   19.202
##  Residual               981.42   31.328
## Number of obs: 368, groups:  Mouse, 10; Experiment, 5
##
## Fixed effects:
##              Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)    52.718      9.958  5.093   5.294  0.00304 **
## GroupxcTauOs    1.296      7.140  4.266   0.182  0.86424
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.356
```

No difference

Plot Min Concentration

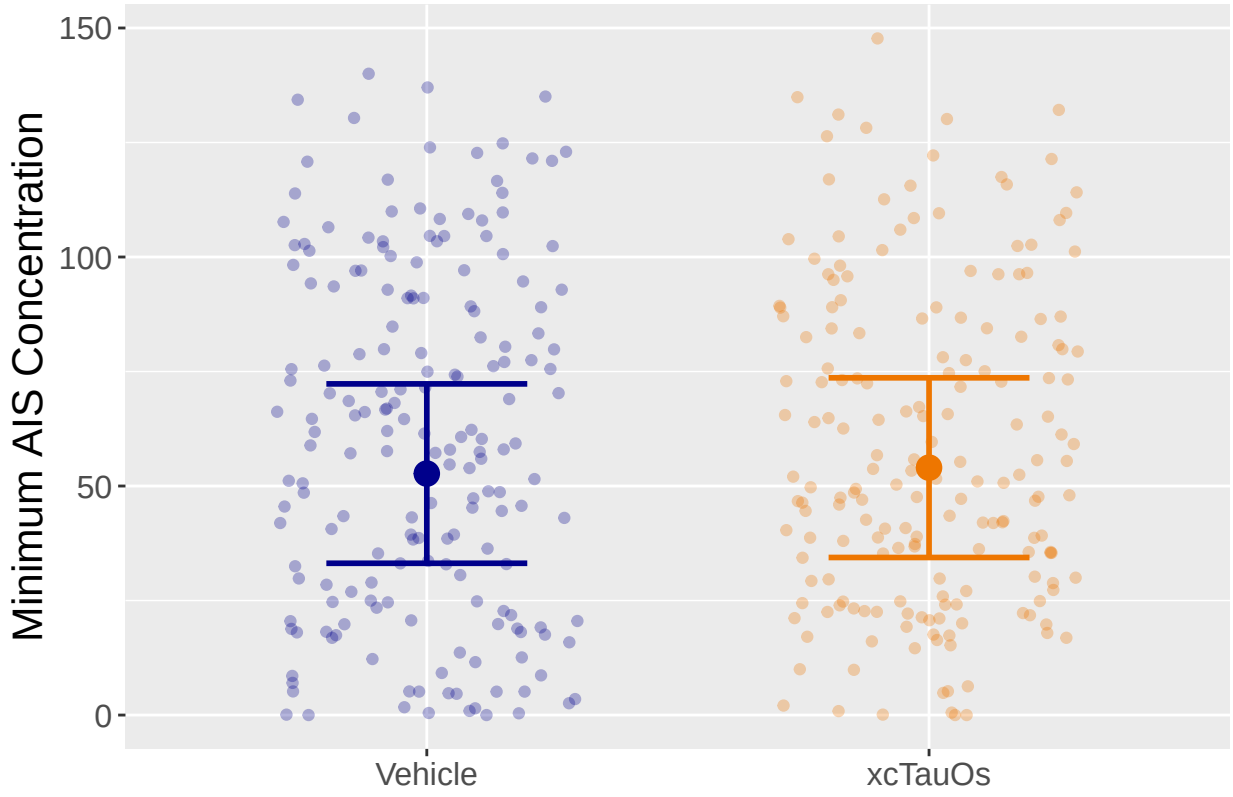
Each dot is the minimum concentration values for each neuron.

On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```
# return dataset of predicted mean +/- 95%CI
pred_min <- ggeffect(minmod, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)

pred_min
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      52.7      9.96     33.1      72.3 1
## 2 xcTauOs      54.0      9.97     34.4      73.6 1
```

```
minplot <- minint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Min_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_min,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_min,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Minimum AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

minplot
```




```
ggsave(minplot, filename = "Figures/cell_2E_minconc.png", width = 6, height = 4)
```

Fig 2F

MAPT-/- (5 independent experiments) Is the AIS Length different between treatments?

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/2F - MAPT--- AIS Length/", filename), skip = 1) %>%
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Length")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/2F - MAPT--- AIS Length/")

Fig2F <- map(my_files, ~readclean(.))
Fig2F <- bind_rows(Fig2F)

Fig2F
```

```
## # A tibble: 368 x 4
##    `Distance_(microns)` filename                                    Neuron Length
##    <lgl>                <chr>                                       <chr>   <dbl>
##  1 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~  20.0
##  2 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~   7.85
##  3 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~   5.55
##  4 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~  34.9
##  5 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~  26.3
##  6 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~   7.85
##  7 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~  47.0
##  8 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~  53.8
##  9 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~  21.9
## 10 NA                   (2F) MAPT-_- Vehicle 1 = 2.18.22 TKOs DIV~ Neuro~  34.3
## # ... with 358 more rows
```

Create new variables for Mouse and Group

```
Fig2F <- Fig2F %>%
  select(-`Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\- ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig2F
```

```
## # A tibble: 368 x 3
##    Mouse     Neuron    Length
##    <chr>     <chr>      <dbl>
##  1 Vehicle 1 Neuron 1   20.0
##  2 Vehicle 1 Neuron 2    7.85
##  3 Vehicle 1 Neuron 3    5.55
##  4 Vehicle 1 Neuron 4   34.9
##  5 Vehicle 1 Neuron 5   26.3
##  6 Vehicle 1 Neuron 6    7.85
##  7 Vehicle 1 Neuron 7   47.0
##  8 Vehicle 1 Neuron 8   53.8
##  9 Vehicle 1 Neuron 9   21.9
## 10 Vehicle 1 Neuron 10  34.3
## # ... with 358 more rows
```

Create treatment group variable

```
Fig2F <- Fig2F %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Length)
```

Create Experiment variable

Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

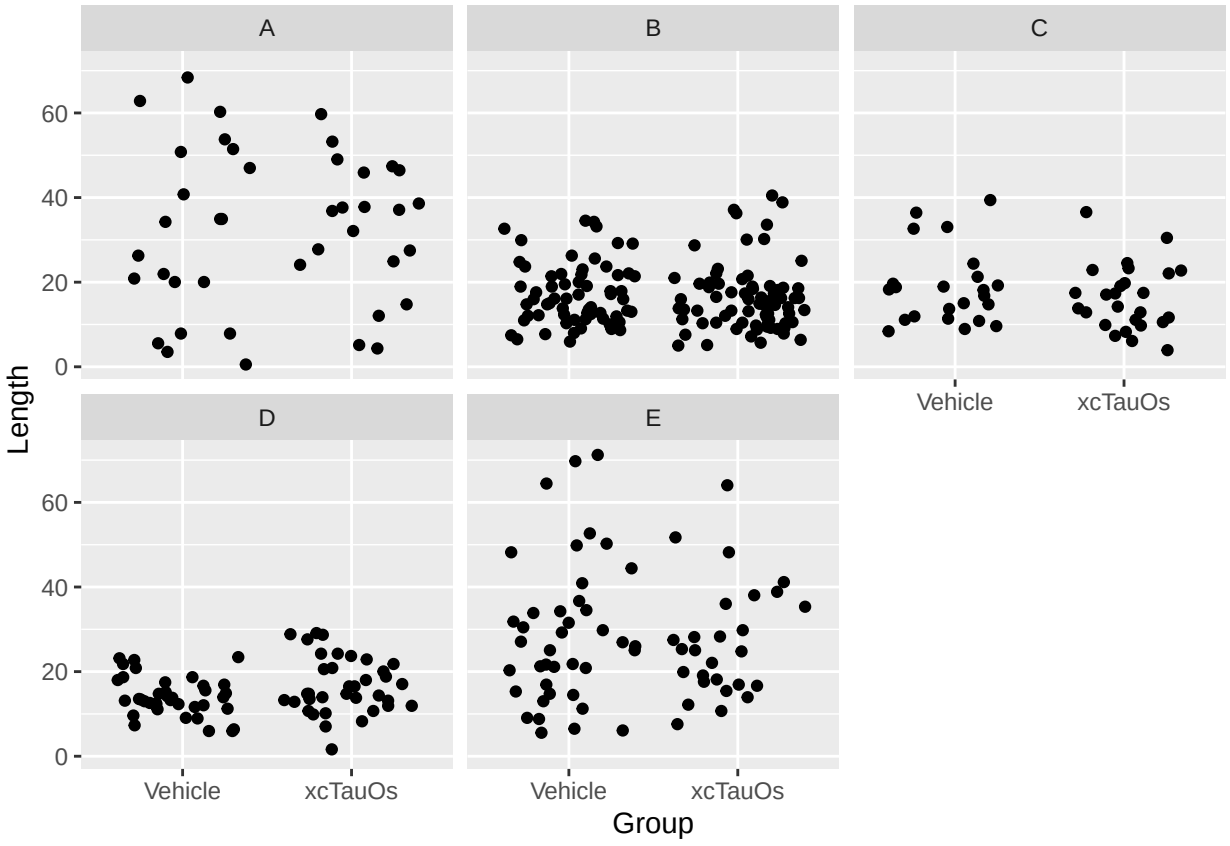
```
Fig2F <- Fig2F %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
```

```
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D",
                                str_detect(Mouse,"5") ~ "E"))

Fig2F
```

```
## # A tibble: 368 x 5
##    Group   Mouse     Neuron    Length Experiment
##    <chr>   <chr>     <chr>      <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1   20.0  A
##  2 Vehicle Vehicle 1 Neuron 2    7.85 A
##  3 Vehicle Vehicle 1 Neuron 3    5.55 A
##  4 Vehicle Vehicle 1 Neuron 4   34.9  A
##  5 Vehicle Vehicle 1 Neuron 5   26.3  A
##  6 Vehicle Vehicle 1 Neuron 6    7.85 A
##  7 Vehicle Vehicle 1 Neuron 7   47.0  A
##  8 Vehicle Vehicle 1 Neuron 8   53.8  A
##  9 Vehicle Vehicle 1 Neuron 9   21.9  A
## 10 Vehicle Vehicle 1 Neuron 10  34.3  A
## # ... with 358 more rows
```

Exploratory data analysis

```
Fig2F %>%
  ggplot(aes(Group, Length)) +
  geom_jitter(height = 0) +
  facet_wrap(~Experiment)
```



Model

```
Fig2Fmod <- lmer(Length ~ Group + (1|Mouse) + (1|Experiment), data = Fig2F)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(Fig2Fmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Length ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: Fig2F
##
## REML criterion at convergence: 2811.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.9140 -0.5816 -0.1168  0.4568  3.9388
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)   0.00    0.000
##  Experiment (Intercept)  58.48    7.647
```

```
##  Residual               118.85   10.902
## Number of obs: 368, groups:  Mouse, 10; Experiment, 5
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)   22.2464     3.5200   4.1690   6.320  0.00279 **
## GroupxcTauOs  -0.4512     1.1422 362.1476  -0.395  0.69302
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.159
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

No difference in Length

Mouse does not matter at all, so don't worry about singular fit message.

Plot

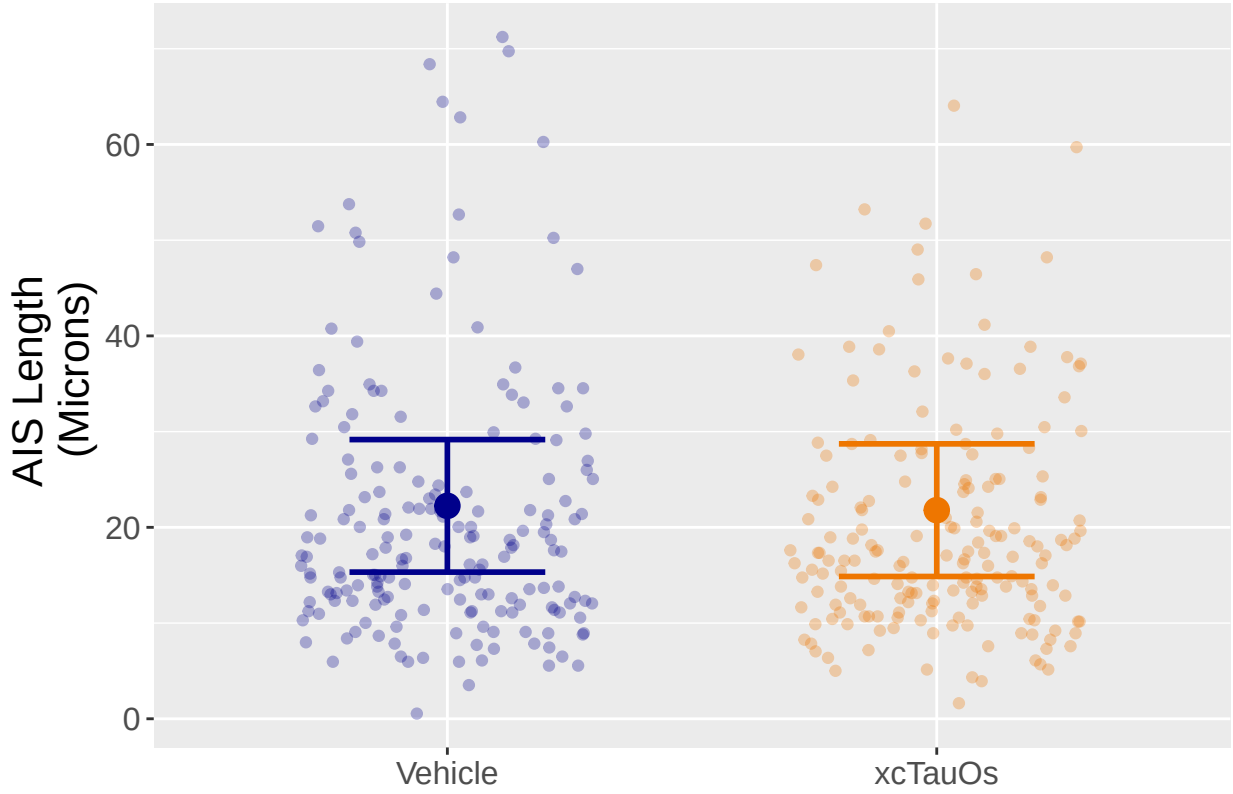
Jitter with 95% CI (like human data) - vehicle = blue - tau = orange

```
# return dataset of predicted mean +/- 95%CI
pred_length <- ggeffect(Fig2Fmod, terms = "Group") %>%
  as_tibble() %>%
  rename(Group = x)

pred_length
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      22.2      3.52     15.3      29.2 1
## 2 xcTauOs      21.8      3.52     14.9      28.7 1
```

```
lengthplot <- Fig2F %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Length,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_length,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_length,
                aes(x = Group,
                    ymin = conf.low,
```

```
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "AIS Length \n (Microns)", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

lengthplot
```




```
ggsave(lengthplot, filename = "Figures/cell_2F.png", width = 6, height = 4)
```

Fig 3D

Tau lentivirus and MAPT-/- (4 independent experiments)

Is the Soma-AIS gap different between vehicle and xcTauOs

Load Data

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/3D - Tau lentivirus + MAPT--- Soma-AIS Gap/", filename), skip = 
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Gap")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/3D - Tau lentivirus + MAPT--- Soma-AIS Gap/")

Fig3D <- map(my_files, ~readclean(.))
Fig3D <- bind_rows(Fig3D)

Fig3D
```

```
## # A tibble: 466 x 4
##    `Distance_(microns)` filename                                  Neuron    Gap
##    <lgl>                <chr>                                     <chr>   <dbl>
##  1 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~ 18.0
##  2 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~ 13.4
##  3 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~  2.96
##  4 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~  9.21
##  5 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~ 10.4
##  6 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~  2.19
##  7 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~  3.66
##  8 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~  7.81
##  9 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~  4.32
## 10 NA                   (3D) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~  0.908
## # ... with 456 more rows
```

Create new variables for Mouse and Group

```
Fig3D <- Fig3D %>%
  select(-`Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\- ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig3D
```

```
## # A tibble: 466 x 3
##    Mouse     Neuron       Gap
##    <chr>     <chr>      <dbl>
##  1 Vehicle 1 Neuron 1  18.0
##  2 Vehicle 1 Neuron 2  13.4
##  3 Vehicle 1 Neuron 3   2.96
##  4 Vehicle 1 Neuron 4   9.21
##  5 Vehicle 1 Neuron 5  10.4
##  6 Vehicle 1 Neuron 6   2.19
##  7 Vehicle 1 Neuron 7   3.66
##  8 Vehicle 1 Neuron 8   7.81
##  9 Vehicle 1 Neuron 9   4.32
```

```
## 10 Vehicle 1 Neuron 10  0.908
## # ... with 456 more rows
```

Create treatment group variable

```
Fig3D <- Fig3D %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Gap)
```

Create Experiment variable

Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

```
Fig3D <- Fig3D %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D"))

Fig3D
```

```
## # A tibble: 466 x 5
##    Group   Mouse     Neuron       Gap Experiment
##    <chr>   <chr>     <chr>      <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1  18.0   A
##  2 Vehicle Vehicle 1 Neuron 2  13.4   A
##  3 Vehicle Vehicle 1 Neuron 3   2.96  A
##  4 Vehicle Vehicle 1 Neuron 4   9.21  A
##  5 Vehicle Vehicle 1 Neuron 5  10.4   A
##  6 Vehicle Vehicle 1 Neuron 6   2.19  A
##  7 Vehicle Vehicle 1 Neuron 7   3.66  A
##  8 Vehicle Vehicle 1 Neuron 8   7.81  A
##  9 Vehicle Vehicle 1 Neuron 9   4.32  A
## 10 Vehicle Vehicle 1 Neuron 10  0.908 A
## # ... with 456 more rows
```

Basic data checks

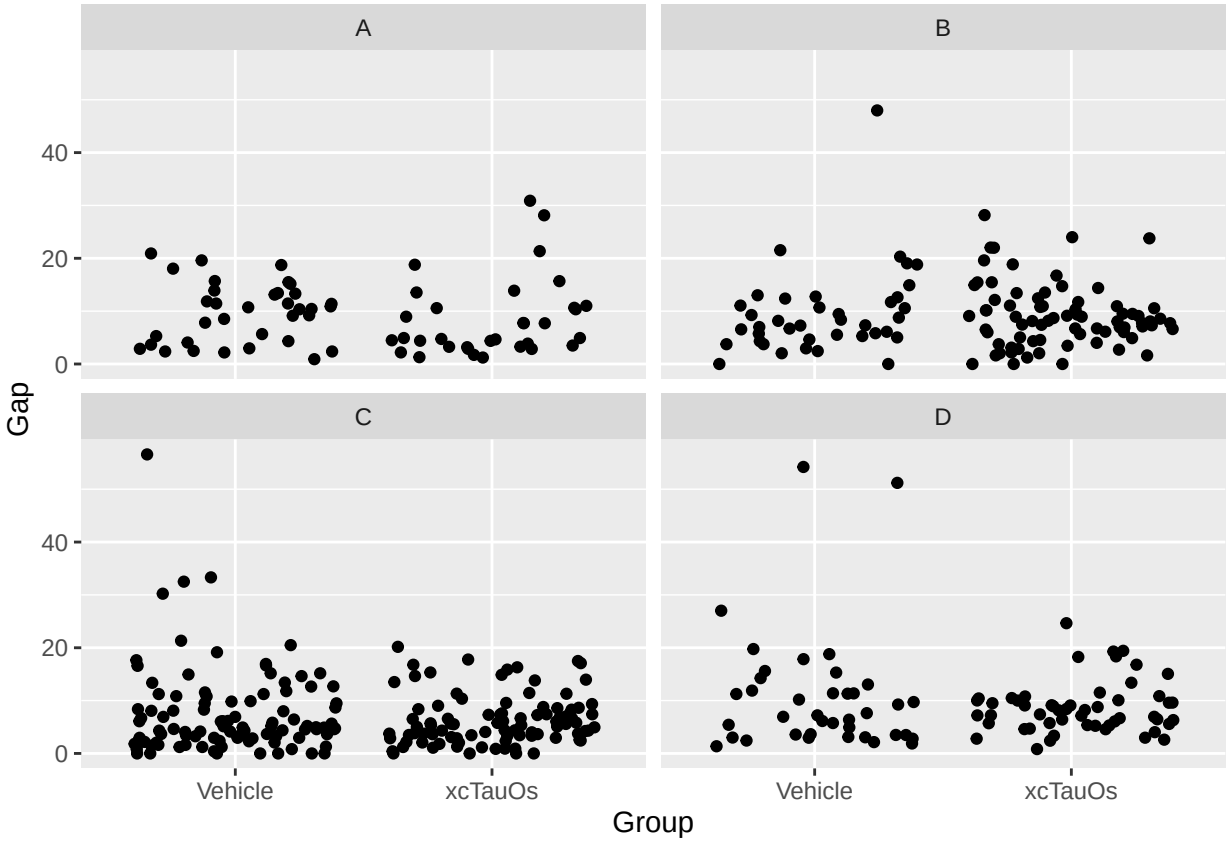
How many neurons per experiment

```
Fig3D %>%
  count(Mouse)
```

```
## # A tibble: 8 x 2
##   Mouse            n
##   <chr>        <int>
## 1 "Vehicle 1"     35
## 2 "Vehicle 2 "    39
## 3 "Vehicle 3"     99
## 4 "Vehicle 4"     39
## 5 "xcTauOs 1"     33
## 6 "xcTauOs 2 "    75
## 7 "xcTauOs 3"     93
## 8 "xcTauOs 4"     53
```

Exploratory plot

```
Fig3D %>%
  ggplot(aes(Group, Gap)) +
  geom_jitter() +
  facet_wrap(~Experiment)
```



Model

```
Fig3Dmod <- lmer(Gap ~ Group + (1|Mouse) + (1|Experiment), data = Fig3D)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(Fig3Dmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Gap ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: Fig3D
##
## REML criterion at convergence: 3151.7
```

```
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.3857 -0.6317 -0.2104  0.3106  6.8433
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)  0.000   0.000
##  Experiment (Intercept)  1.252   1.119
##  Residual               50.537   7.109
## Number of obs: 466, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)    9.3332     0.7540   5.8974   12.38 1.93e-05 ***
## GroupxcTauOs  -1.2069     0.6667 463.9513   -1.81   0.0709 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.487
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

No difference in Gap (p = 0.071)

Mouse removed 0 variance, so don't worry about singular fit message

Plot

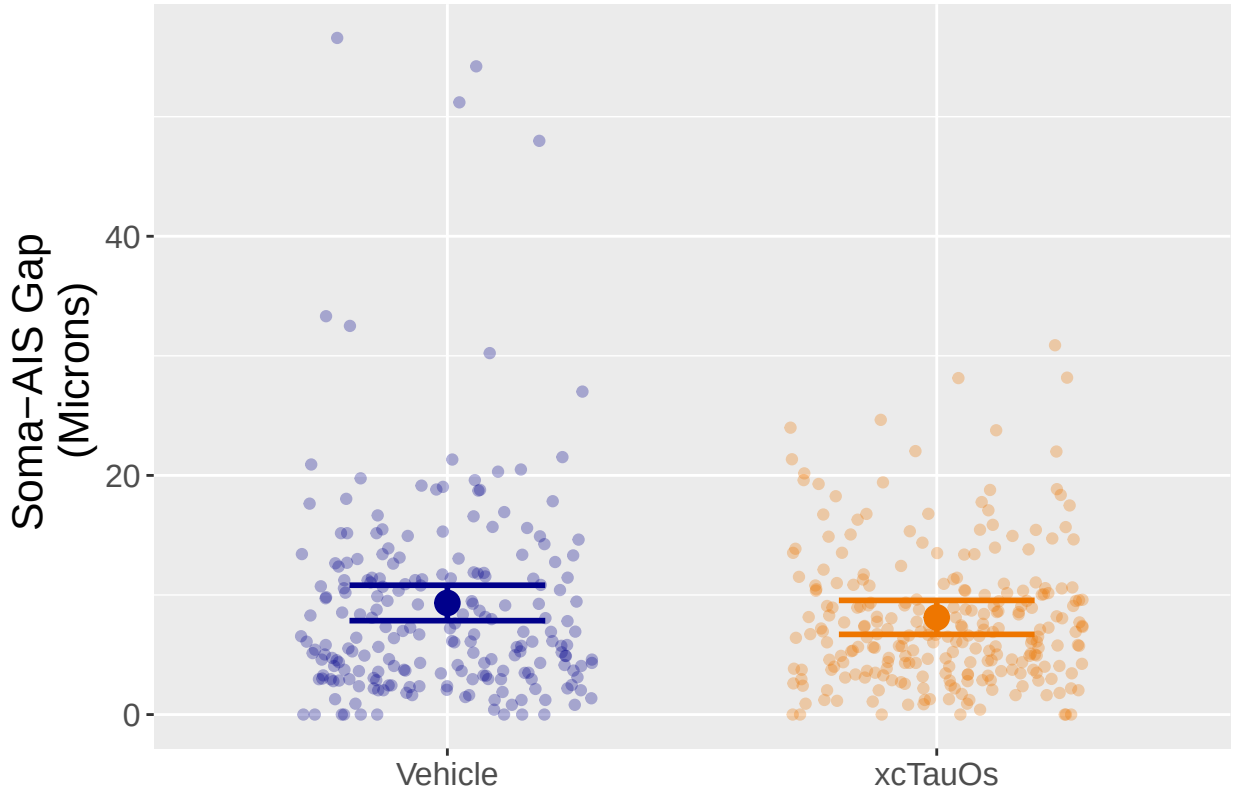
Jitter with 95% CI (like human data) - vehicle = blue - tau = orange

```
# return dataset of predicted mean +/- 95%CI
pred_gap <- ggeffect(Fig3Dmod, terms = "Group") %>%
  as_tibble() %>%
  rename(Group = x)

pred_gap
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      9.33     0.754     7.85     10.8  1
## 2 xcTauOs      8.13     0.723     6.70      9.55 1
```

```
gapplot <- Fig3D %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Gap,
                  color = Group),
              alpha = .3,
```

```
              width = .3) +
  geom_point(data = pred_gap,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_gap,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Soma-AIS Gap \n (Microns)", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

gapplot
```



```
ggsave(gapplot, filename = "Figures/cell_3D.png", width = 6, height = 4)
```

Fig 3E

Lentivirus + MAPT-/- (4 independent experiments)

Is the AIS concentration different between vehicle and xcTauOs

Load Data

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/3E - Tau lentivirus + MAPT--- AIS Concentration/", filename), sk
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Concentration")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/3E - Tau lentivirus + MAPT--- AIS Concentration/")

Fig3E <- map(my_files, ~readclean(.))
Fig3E <- bind_rows(Fig3E)

Fig3E
```

```
## # A tibble: 197,584 x 4
##    `Distance_(microns)` filename                              Neuron Concentration
##                   <dbl> <chr>                                <chr>          <dbl>
##  1                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          46.9
##  2                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          36.3
##  3                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          52.5
##  4                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          30.7
##  5                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          24.8
##  6                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          81.6
##  7                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          58.4
##  8                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          61.6
##  9                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~          70.7
## 10                    0 (3E) Tau lentivirus + MAPT-_- Vehi~ Neuro~         102.
## # ... with 197,574 more rows
```

Create new variables for Mouse and Group

```
Fig3E <- Fig3E %>%
  rename(Dist = `Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\- ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig3E
```

```
## # A tibble: 197,584 x 4
##     Dist Mouse     Neuron    Concentration
##    <dbl> <chr>     <chr>             <dbl>
##  1     0 Vehicle 1 Neuron 1           46.9
##  2     0 Vehicle 1 Neuron 2           36.3
##  3     0 Vehicle 1 Neuron 3           52.5
##  4     0 Vehicle 1 Neuron 4           30.7
##  5     0 Vehicle 1 Neuron 5           24.8
##  6     0 Vehicle 1 Neuron 6           81.6
##  7     0 Vehicle 1 Neuron 7           58.4
##  8     0 Vehicle 1 Neuron 8           61.6
##  9     0 Vehicle 1 Neuron 9           70.7
## 10     0 Vehicle 1 Neuron 10         102.
## # ... with 197,574 more rows
```

Create treatment group variable

```
Fig3E <- Fig3E %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Dist, Concentration)
```

Create Experiment variable

Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

```
Fig3E <- Fig3E %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D"))

Fig3E
```

```
## # A tibble: 197,584 x 6
##    Group   Mouse     Neuron     Dist Concentration Experiment
##    <chr>   <chr>     <chr>     <dbl>         <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1      0          46.9 A
##  2 Vehicle Vehicle 1 Neuron 2      0          36.3 A
##  3 Vehicle Vehicle 1 Neuron 3      0          52.5 A
##  4 Vehicle Vehicle 1 Neuron 4      0          30.7 A
##  5 Vehicle Vehicle 1 Neuron 5      0          24.8 A
##  6 Vehicle Vehicle 1 Neuron 6      0          81.6 A
##  7 Vehicle Vehicle 1 Neuron 7      0          58.4 A
##  8 Vehicle Vehicle 1 Neuron 8      0          61.6 A
##  9 Vehicle Vehicle 1 Neuron 9      0          70.7 A
## 10 Vehicle Vehicle 1 Neuron 10     0         102.  A
## # ... with 197,574 more rows
```

Clean dataset

Drop observations where Concentration is missing because the neuron wasn't that long

```
Fig3E <- Fig3E %>%
  arrange(Group, Experiment, Mouse, Neuron, Dist) %>%
  drop_na(Concentration)
```

Basic data checks

How many neurons per experiment

```
Fig3E %>%
  count(Mouse)
```

```
## # A tibble: 8 x 2
##   Mouse            n
##   <chr>        <int>
## 1 "Vehicle 1"   5150
## 2 "Vehicle 2 "  6038
## 3 "Vehicle 3"  12369
## 4 "Vehicle 4"   5995
## 5 "xcTauOs 1"   4016
## 6 "xcTauOs 2 "  7722
## 7 "xcTauOs 3"   7341
## 8 "xcTauOs 4"   6649
```

Exploratory plot

- Make rolling average plot

A rolling average across 3-distance observations works nicely to show the trend.

I used the `rollapply()` function rather than the more standard `rollmean()` function because `rollmean()` has no way to remove NAs.

```
Fig3E <- Fig3E %>%
  group_by(Mouse, Neuron) %>%
  mutate(roll_Conc = rollapply(Concentration, 3, mean, na.rm = TRUE, fill = NA)) %>%
  ungroup()

Fig3E
```

```
## # A tibble: 55,280 x 7
##    Group   Mouse     Neuron    Dist Concentration Experiment roll_Conc
##    <chr>   <chr>     <chr>    <dbl>         <dbl> <chr>          <dbl>
##  1 Vehicle Vehicle 1 Neuron 1 0              46.9 A               NA
##  2 Vehicle Vehicle 1 Neuron 1 0.135          48.9 A               48.5
##  3 Vehicle Vehicle 1 Neuron 1 0.271          49.9 A               50.0
##  4 Vehicle Vehicle 1 Neuron 1 0.406          51.3 A               51.1
##  5 Vehicle Vehicle 1 Neuron 1 0.542          52.1 A               52.1
##  6 Vehicle Vehicle 1 Neuron 1 0.677          52.9 A               53.0
##  7 Vehicle Vehicle 1 Neuron 1 0.813          54.1 A               54.2
##  8 Vehicle Vehicle 1 Neuron 1 0.948          55.6 A               55.6
##  9 Vehicle Vehicle 1 Neuron 1 1.08           57.1 A               57.3
## 10 Vehicle Vehicle 1 Neuron 1 1.22           59.1 A               59.4
## # ... with 55,270 more rows
```

One line per mouse, rolling average averaged over all neurons at a given dist.

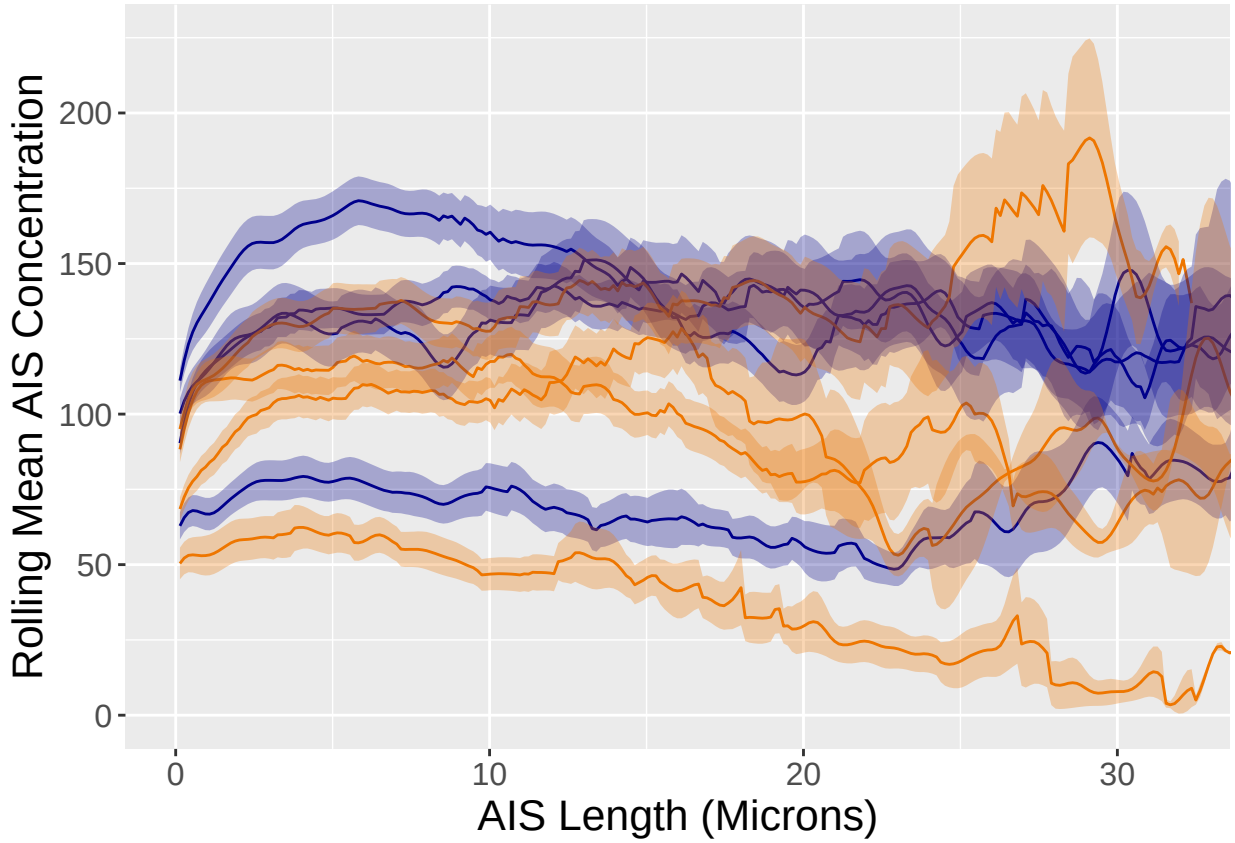
The rolling average is the average of a 3-dist chunk.

```
rollavgdat <- Fig3E %>%
  group_by(Group, Experiment, Mouse, Dist) %>%
  summarize(Mean_Conc = mean(roll_Conc, na.rm = TRUE),
            sd_Conc = sd(roll_Conc, na.rm = TRUE),
            n_neurons = n(),
            se_Conc = sd_Conc/sqrt(n_neurons))
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

```
lineplot <- rollavgdat %>%
  ggplot(aes(Dist, Mean_Conc)) +
  geom_line(aes(group = Mouse, color = Group)) +
  geom_ribbon(aes(ymin = Mean_Conc-se_Conc,
                  ymax = Mean_Conc+se_Conc,
                  group = Mouse,
                  fill = Group), alpha = .3) +
  scale_fill_manual(values = c("darkblue", "darkorange2")) +
  scale_color_manual(values = c("darkblue", "darkorange2")) +
  labs(y = "Rolling Mean AIS Concentration", x = "AIS Length (Microns)") +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size =  12)) +
  coord_cartesian(xlim = c(0,32))

lineplot
```

```
## Warning: Removed 16 row(s) containing missing values (geom_path).
```




```
ggsave(lineplot, filename = "Figures/cell_3E_distVintensity.png", width =  6, height = 4)
```

```
## Warning: Removed 16 row(s) containing missing values (geom_path).
```

Model splines

Splines are a way to fit a non-linear curve to data to understand how the relationship between Distance and Concentration changes for xcTauOs v. Vehicle.

Try basic natural splines model

```
splinemod <- lmer(Concentration ~ ns(Dist, df = 5) + Group + (1|Mouse) + (1|Experiment), data = Fig3E)
```

Now try with interaction term

```
splinemod2 <- lmer(Concentration ~ ns(Dist, df = 5) * Group + (1|Mouse) + (1|Experiment), data = Fig3E)
```

See if there is a difference in model fit between splinemod and splinemod2

```
anova(splinemod2, splinemod)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: Fig3E
## Models:
## splinemod: Concentration ~ ns(Dist, df = 5) + Group + (1 | Mouse) + (1 | Experiment)
## splinemod2: Concentration ~ ns(Dist, df = 5) * Group + (1 | Mouse) + (1 | Experiment)
##            npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## splinemod    10 601649 601738 -300815   601629
## splinemod2   15 601600 601734 -300785   601570 59.516  5   1.53e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

splinemod2 fits better than splinemod, so use that

Interpretation of splinemod2, use emmeans to find average difference

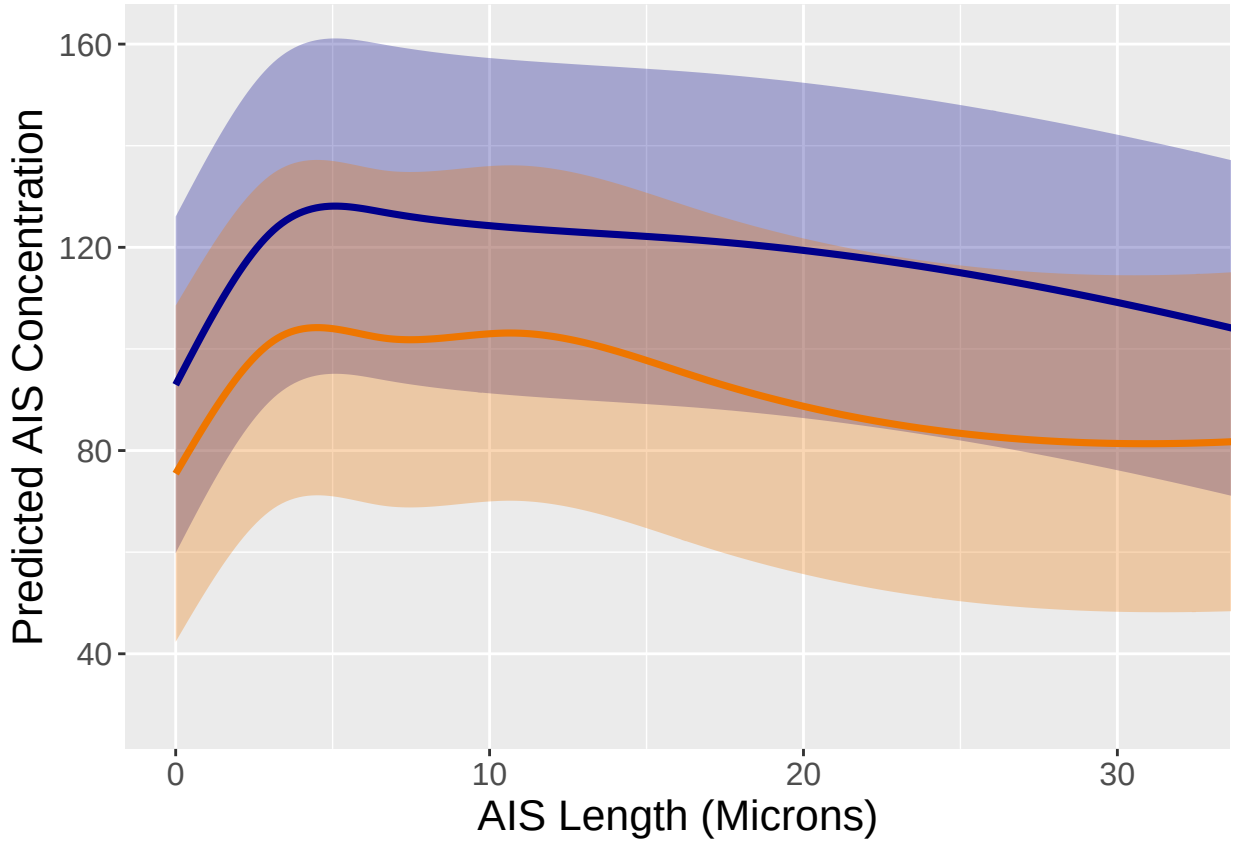
```
pairs(emmeans(splinemod2, specs = "Group"))
```

```
##  contrast          estimate   SE  df z.ratio p.value
##  Vehicle - xcTauOs     20.8 8.07 Inf   2.576  0.0100
##
## Degrees-of-freedom method: asymptotic
```

On average xcTauOs has AIS concentration 20.8 lower than vehicle (p = 0.010)

Plot splines

```
splinesplot <- ggpredict(splinemod2, terms = c("Dist [all]", "Group")) %>%
  as_tibble() %>%
  rename(Dist = x,
         Group = group) %>%
  ggplot(aes(Dist, predicted, color = Group)) +
  #facet_wrap(~Group, nrow = 2) +
  geom_ribbon(aes(ymin = conf.low,
                  ymax = conf.high,
                  fill = Group),
              alpha = .3,
              color = NA) +
  geom_line(lwd = 1.25) +
  scale_color_manual(values = c("darkblue", "darkorange2")) +
  scale_fill_manual(values = c("darkblue", "darkorange2")) +
  labs(x = "AIS Length (Microns)",
       y = "Predicted AIS Concentration", color = "") +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12)) +
  coord_cartesian(xlim = c(0,32))

splinesplot
```




```
ggsave(splinesplot, filename = "Figures/cell_3E_splines.png", width = 6, height = 4)
```

Model mean concentration

Create average intensity dataset. For each neuron, what is the average intensity across the whole distance that was measured

```
avgint <- Fig3E %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarize(Mean_Conc = mean(Concentration),
            n = n()) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

```
avgint
```

```
## # A tibble: 466 x 6
##    Group   Experiment Mouse     Neuron    Mean_Conc     n
##    <chr>   <chr>      <chr>     <chr>         <dbl> <int>
##  1 Vehicle A          Vehicle 1 Neuron 1       64.4   120
##  2 Vehicle A          Vehicle 1 Neuron 10     125.    108
```

```
##  3 Vehicle A          Vehicle 1 Neuron 11      40.3   189
##  4 Vehicle A          Vehicle 1 Neuron 12     138.    132
##  5 Vehicle A          Vehicle 1 Neuron 13     136.     99
##  6 Vehicle A          Vehicle 1 Neuron 14      59.2   163
##  7 Vehicle A          Vehicle 1 Neuron 15      88.8   162
##  8 Vehicle A          Vehicle 1 Neuron 16      68.9    90
##  9 Vehicle A          Vehicle 1 Neuron 17      79.3    56
## 10 Vehicle A          Vehicle 1 Neuron 18      19.1    74
## # ... with 456 more rows
```

Model the mean concentration by Group

```
avgintmodB <- lmer(Mean_Conc ~ Group + (1|Mouse) + (1|Experiment), data = avgint)

summary(avgintmodB)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Mean_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: avgint
##
## REML criterion at convergence: 4929.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.22877 -0.72774 -0.01416  0.72958  2.48332
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)   84.8    9.209
##  Experiment (Intercept)  831.9   28.842
##  Residual               2279.1   47.740
## Number of obs: 466, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   116.197     15.546   3.408   7.475  0.00315 **
## GroupxcTauOs  -22.310      8.043   2.829  -2.774  0.07410 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.268
```

No difference (p = 0.074)

Plot Mean Concentration

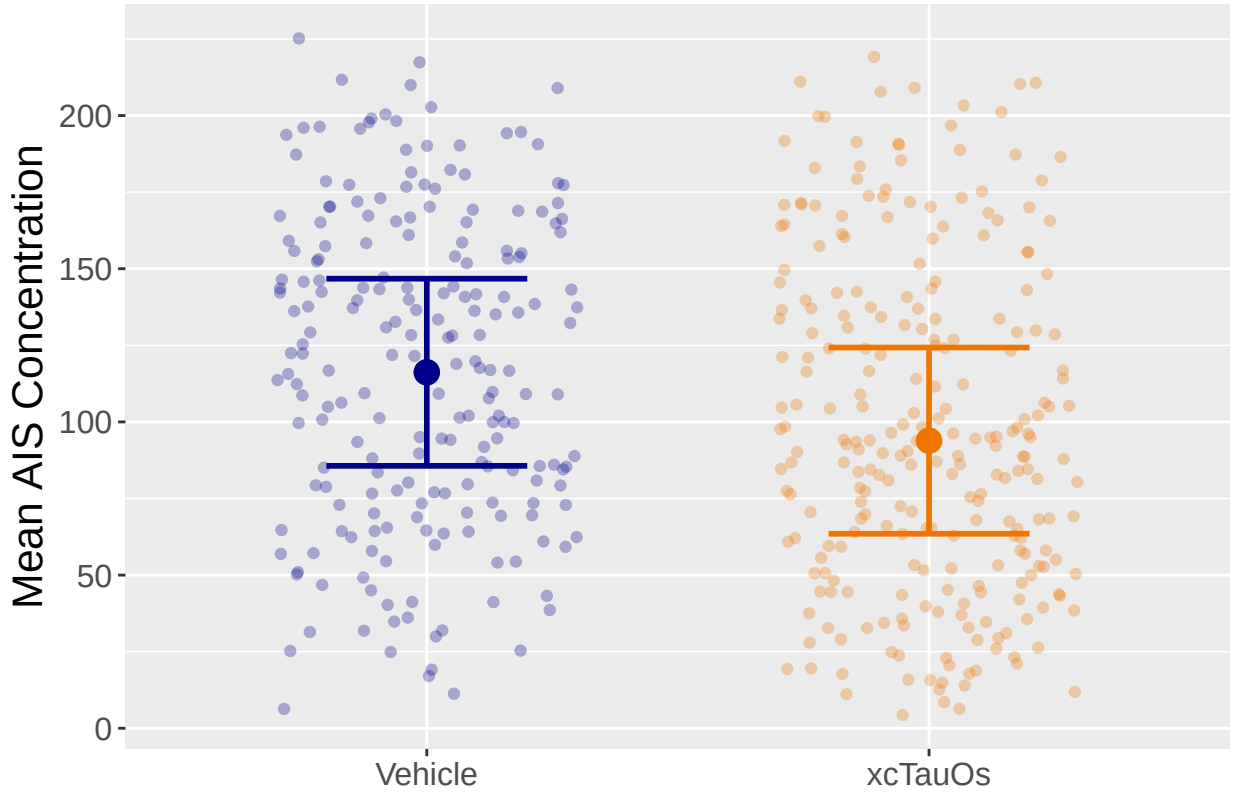
Each dot is an average of all of the concentration values for each neuron.

On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```r
# return dataset of predicted mean +/- 95%CI
pred_mean_avgintmod <- ggeffect(avgintmodB, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)

pred_mean_avgintmod
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle     116.       15.5     85.6      147. 1
## 2 xcTauOs      93.9      15.5     63.5      124. 1
```

```r
meanint_supp_plot <- avgint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Mean_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_mean_avgintmod,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_mean_avgintmod,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Mean AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

meanint_supp_plot
```




```
ggsave(meanint_supp_plot, filename = "Figures/cell_3E_meanconc.png", width = 6, height = 4)
```

Model max concentration

Create max concentration variable

Define the maximum concentration based on the rolling average concentration for each neuron since that worked better for the human data.

```
maxint <- Fig3E %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarise(Max_Conc = max(roll_Conc, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

Check that each neuron only has one maximum

```
maxint %>%
  count(Mouse, Neuron) %>%
  arrange(-n)
```

```
## # A tibble: 466 x 5
## # Groups:   Group, Experiment, Mouse [8]
```

```
##    Group   Experiment Mouse     Neuron        n
##    <chr>   <chr>      <chr>     <chr>     <int>
##  1 Vehicle A          Vehicle 1 Neuron 1      1
##  2 Vehicle A          Vehicle 1 Neuron 10     1
##  3 Vehicle A          Vehicle 1 Neuron 11     1
##  4 Vehicle A          Vehicle 1 Neuron 12     1
##  5 Vehicle A          Vehicle 1 Neuron 13     1
##  6 Vehicle A          Vehicle 1 Neuron 14     1
##  7 Vehicle A          Vehicle 1 Neuron 15     1
##  8 Vehicle A          Vehicle 1 Neuron 16     1
##  9 Vehicle A          Vehicle 1 Neuron 17     1
## 10 Vehicle A          Vehicle 1 Neuron 18     1
## # ... with 456 more rows
```

Model the max concentration by Group

```
maxmod <- lmer(Max_Conc ~ Group + (1|Mouse) + (1|Experiment), data = maxint)

summary(maxmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Max_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: maxint
##
## REML criterion at convergence: 5140.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.32481 -0.75001  0.02246  0.78083  2.16413
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)   40.48   6.363
##  Experiment (Intercept) 1175.12  34.280
##  Residual               3606.15  60.051
## Number of obs: 466, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   160.774     17.978   3.245   8.943  0.00215 **
## GroupxcTauOs  -28.753      7.357   2.273  -3.908  0.04831 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.218
```

xcTauOs is 28.75 lower ($p = 0.048$)

Plot Max Concentration

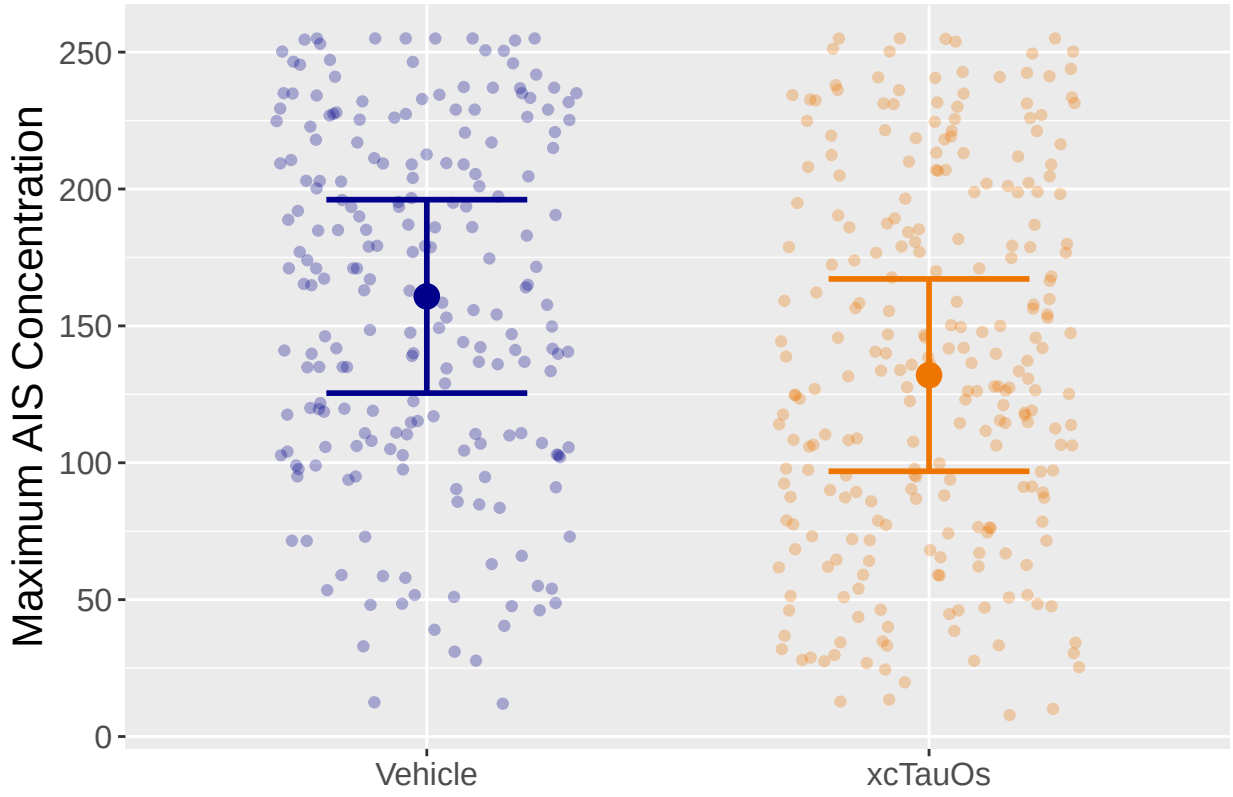
Each dot is the maximum concentration values for each neuron.

On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```
# return dataset of predicted mean +/- 95%CI
pred_max <- ggeffect(maxmod, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)

pred_max
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      161.      18.0    125.       196. 1
## 2 xcTauOs      132.      17.9     96.9      167. 1
```

```
maxplot <- maxint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Max_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_max,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_max,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Maximum AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

maxplot
```




```
ggsave(maxplot, filename = "Figures/cell_3E_maxconc.png", width = 6, height = 4)
```

Model min concentration

Create min concentration variable

Define the minimum concentration based on the rolling average concentration for each neuron since that worked better for the human data.

```
minint <- Fig3E %>%
  group_by(Group, Experiment, Mouse, Neuron) %>%
  summarise(Min_Conc = min(roll_Conc, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'Group', 'Experiment', 'Mouse'. You can
## override using the `.groups` argument.
```

Check that each neuron only has one maximum

```
minint %>%
  count(Mouse, Neuron) %>%
  arrange(-n)
```

```
## # A tibble: 466 x 5
## # Groups:   Group, Experiment, Mouse [8]
```

```
##    Group   Experiment Mouse     Neuron        n
##    <chr>   <chr>      <chr>     <chr>     <int>
##  1 Vehicle A          Vehicle 1 Neuron 1      1
##  2 Vehicle A          Vehicle 1 Neuron 10     1
##  3 Vehicle A          Vehicle 1 Neuron 11     1
##  4 Vehicle A          Vehicle 1 Neuron 12     1
##  5 Vehicle A          Vehicle 1 Neuron 13     1
##  6 Vehicle A          Vehicle 1 Neuron 14     1
##  7 Vehicle A          Vehicle 1 Neuron 15     1
##  8 Vehicle A          Vehicle 1 Neuron 16     1
##  9 Vehicle A          Vehicle 1 Neuron 17     1
## 10 Vehicle A          Vehicle 1 Neuron 18     1
## # ... with 456 more rows
```

Model the min concentration by Group

```
minmod <- lmer(Min_Conc ~ Group + (1|Mouse) + (1|Experiment), data = minint)

summary(minmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Min_Conc ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: minint
##
## REML criterion at convergence: 4586
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.34999 -0.73577 -0.05332  0.75866  2.59268
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  Mouse      (Intercept)   27.05   5.201
##  Experiment (Intercept)  245.34  15.663
##  Residual               1092.05  33.046
## Number of obs: 466, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)    62.418      8.605  3.479   7.254  0.00323 **
## GroupxcTauOs   -6.588      4.906  2.541  -1.343  0.28663
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.298
```

No difference

Plot Min Concentration

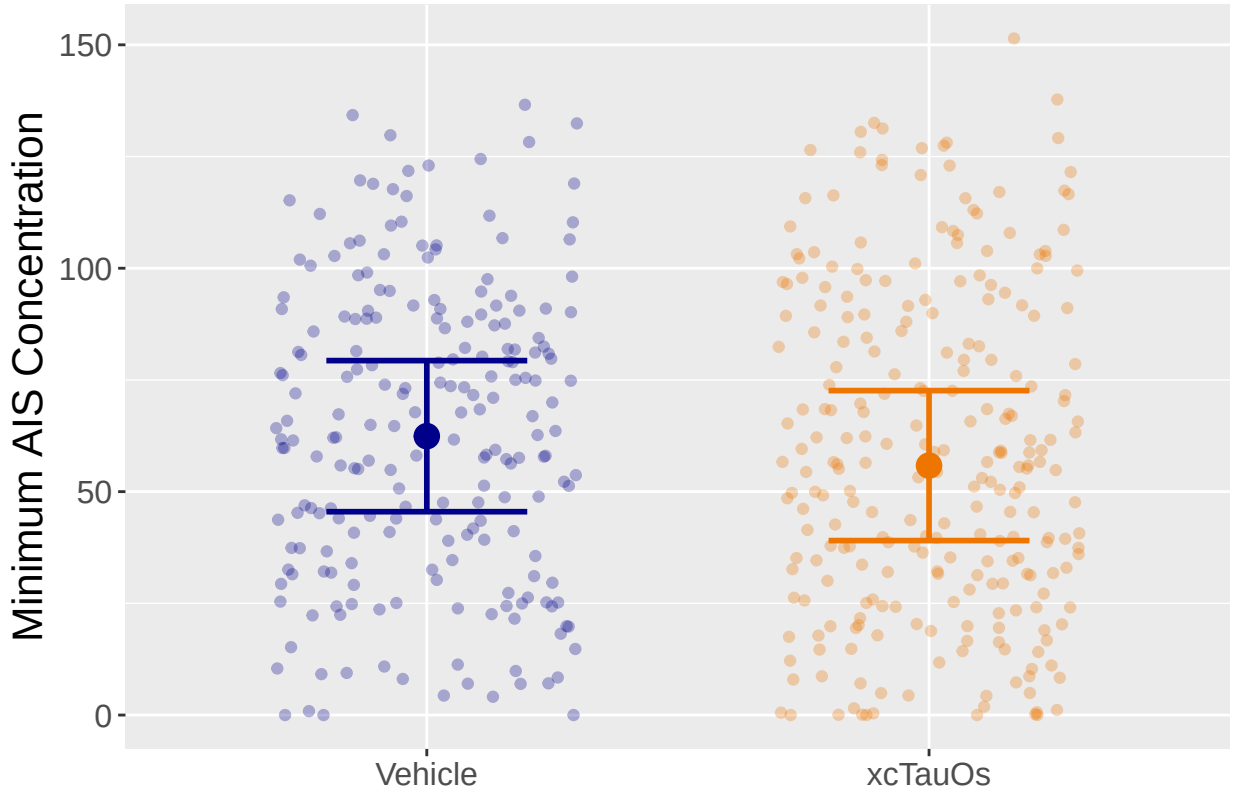
Each dot is the minimum concentration values for each neuron.

On top of dots are the predicted means and 95% confidence interval from the linear mixed model.

```
# return dataset of predicted mean +/- 95%CI
pred_min <- ggeffect(minmod, terms = c("Group")) %>%
  as_tibble() %>%
  rename(Group = x)

pred_min
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      62.4      8.60     45.5      79.3 1
## 2 xcTauOs      55.8      8.54     39.0      72.6 1
```

```
minplot <- minint %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Min_Conc,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_min,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_min,
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "Minimum AIS Concentration", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

minplot
```




```
ggsave(minplot, filename = "Figures/cell_3E_minconc.png", width = 6, height = 4)
```

Fig 3F

Lentivirus + MAPT-/- (4 independent experiments) Is the AIS Length different between treatments?

Load Data

Write a function that will read the file and clean it

```
readclean <- function(filename){
  dat <- read_excel(paste0("Data/Mouse/3F - Tau lentivirus + MAPT--- AIS Length/", filename), skip = 1)
  dplyr::select(-avg, -std, -count, - `std rror`) %>%
  mutate(filename = filename) %>%
  pivot_longer(cols = contains("Neur"), names_to = "Neuron", values_to = "Length")
}
```

Create a list of all of the datafiles and then iterate through them to read them all in.

```
my_files <- list.files(path = "Data/Mouse/3F - Tau lentivirus + MAPT--- AIS Length/")

Fig3F <- map(my_files, ~readclean(.))
```

```
Fig3F <- bind_rows(Fig3F)

Fig3F
```

```
## # A tibble: 466 x 4
##    `Distance_(microns)` filename                                    Neuron Length
##    <lgl>                <chr>                                       <chr>   <dbl>
##  1 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   16.1
##  2 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   16.8
##  3 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   14.6
##  4 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   21.1
##  5 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   13.5
##  6 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   33.6
##  7 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   21.5
##  8 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   20.0
##  9 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   13.3
## 10 NA                   (3F) Tau lentivirus + MAPT-_- Vehicle 1 =~ Neuro~   14.5
## # ... with 456 more rows
```

Create new variables for Mouse and Group

```
Fig3F <- Fig3F %>%
  select(-`Distance_(microns)`) %>%
  separate(filename, into = c(NA, "Mouse"), sep = "\\- ") %>%
  separate(Mouse, into = c("Mouse", NA), sep = " = ")

Fig3F
```

```
## # A tibble: 466 x 3
##    Mouse     Neuron    Length
##    <chr>     <chr>      <dbl>
##  1 Vehicle 1 Neuron 1    16.1
##  2 Vehicle 1 Neuron 2    16.8
##  3 Vehicle 1 Neuron 3    14.6
##  4 Vehicle 1 Neuron 4    21.1
##  5 Vehicle 1 Neuron 5    13.5
##  6 Vehicle 1 Neuron 6    33.6
##  7 Vehicle 1 Neuron 7    21.5
##  8 Vehicle 1 Neuron 8    20.0
##  9 Vehicle 1 Neuron 9    13.3
## 10 Vehicle 1 Neuron 10   14.5
## # ... with 456 more rows
```

Create treatment group variable

```
Fig3F <- Fig3F %>%
  mutate(Group = if_else(str_detect(Mouse,"V"), "Vehicle", "xcTauOs")) %>%
  select(Group, Mouse, Neuron, Length)
```

Create Experiment variable

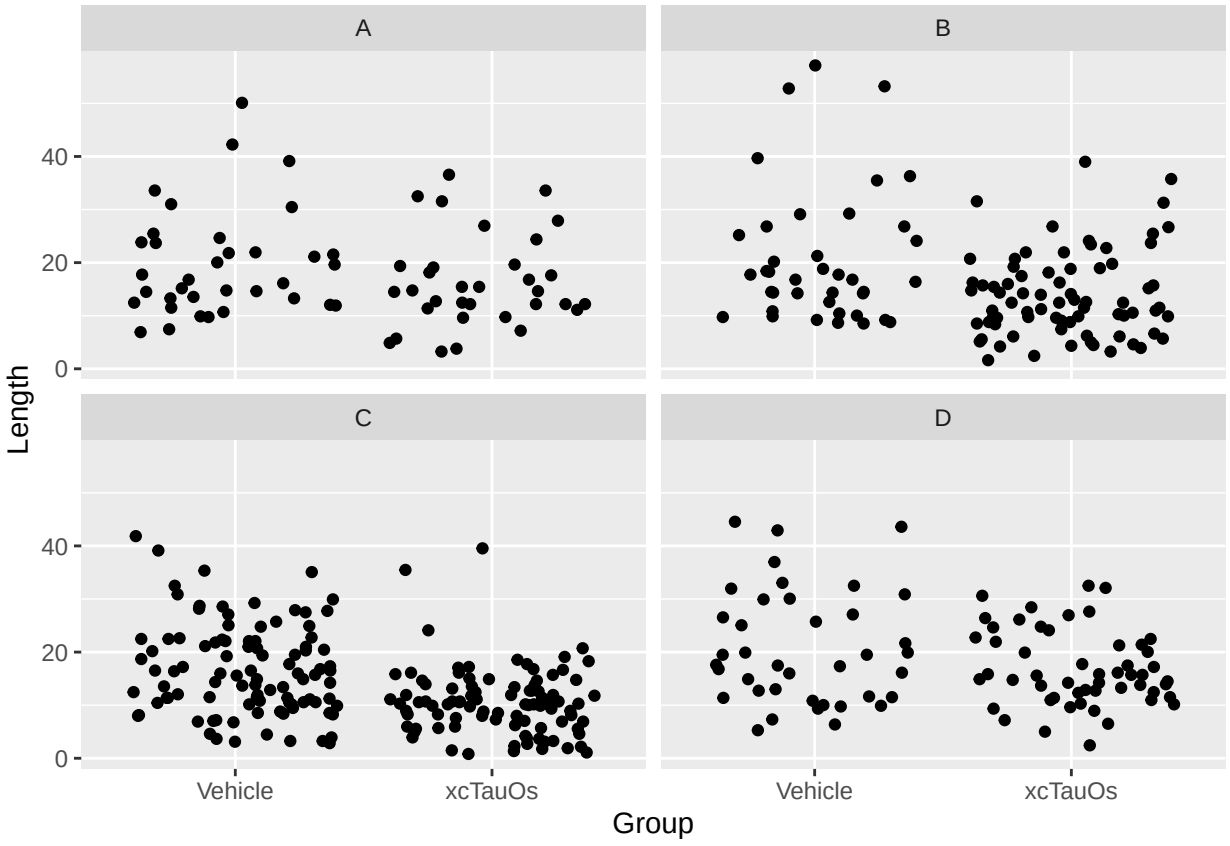
Cells from embryos from one pregnant female were divided into vehicle and xcTauOs so we need to know which pregnant female the cells came from.

```
Fig3F <- Fig3F %>%
  mutate(Experiment = case_when(str_detect(Mouse,"1") ~ "A",
                                str_detect(Mouse,"2") ~ "B",
                                str_detect(Mouse,"3") ~ "C",
                                str_detect(Mouse,"4") ~ "D"))

Fig3F
```

```
## # A tibble: 466 x 5
##    Group   Mouse     Neuron    Length Experiment
##    <chr>   <chr>     <chr>      <dbl> <chr>
##  1 Vehicle Vehicle 1 Neuron 1    16.1 A
##  2 Vehicle Vehicle 1 Neuron 2    16.8 A
##  3 Vehicle Vehicle 1 Neuron 3    14.6 A
##  4 Vehicle Vehicle 1 Neuron 4    21.1 A
##  5 Vehicle Vehicle 1 Neuron 5    13.5 A
##  6 Vehicle Vehicle 1 Neuron 6    33.6 A
##  7 Vehicle Vehicle 1 Neuron 7    21.5 A
##  8 Vehicle Vehicle 1 Neuron 8    20.0 A
##  9 Vehicle Vehicle 1 Neuron 9    13.3 A
## 10 Vehicle Vehicle 1 Neuron 10   14.5 A
## # ... with 456 more rows
```

Exploratory data analysis

```
Fig3F %>%
  ggplot(aes(Group, Length)) +
  geom_jitter(height = 0) +
  facet_wrap(~Experiment)
```



We can see the difference

Model

```
Fig3Fmod <- lmer(Length ~ Group + (1|Mouse) + (1|Experiment), data = Fig3F)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(Fig3Fmod)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Length ~ Group + (1 | Mouse) + (1 | Experiment)
##    Data: Fig3F
##
## REML criterion at convergence: 3340.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.8472 -0.6542 -0.1574  0.5197  4.2992
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
```

```
##  Mouse      (Intercept)  0.000   0.000
##  Experiment (Intercept)  4.824   2.196
##  Residual               75.517   8.690
## Number of obs: 466, groups:  Mouse, 8; Experiment, 4
##
## Fixed effects:
##              Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)   19.5855     1.2617   4.1770  15.523 7.49e-05 ***
## GroupxcTauOs  -5.4777     0.8162 462.9629  -6.711 5.67e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr)
## GroupxcTaOs -0.357
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

xcTauOs has AIS 5.48 microns shorter than Vehicle (p = 5.67e-11)

Mouse removes zero variance, so don't worry about the singular fit.

Plot

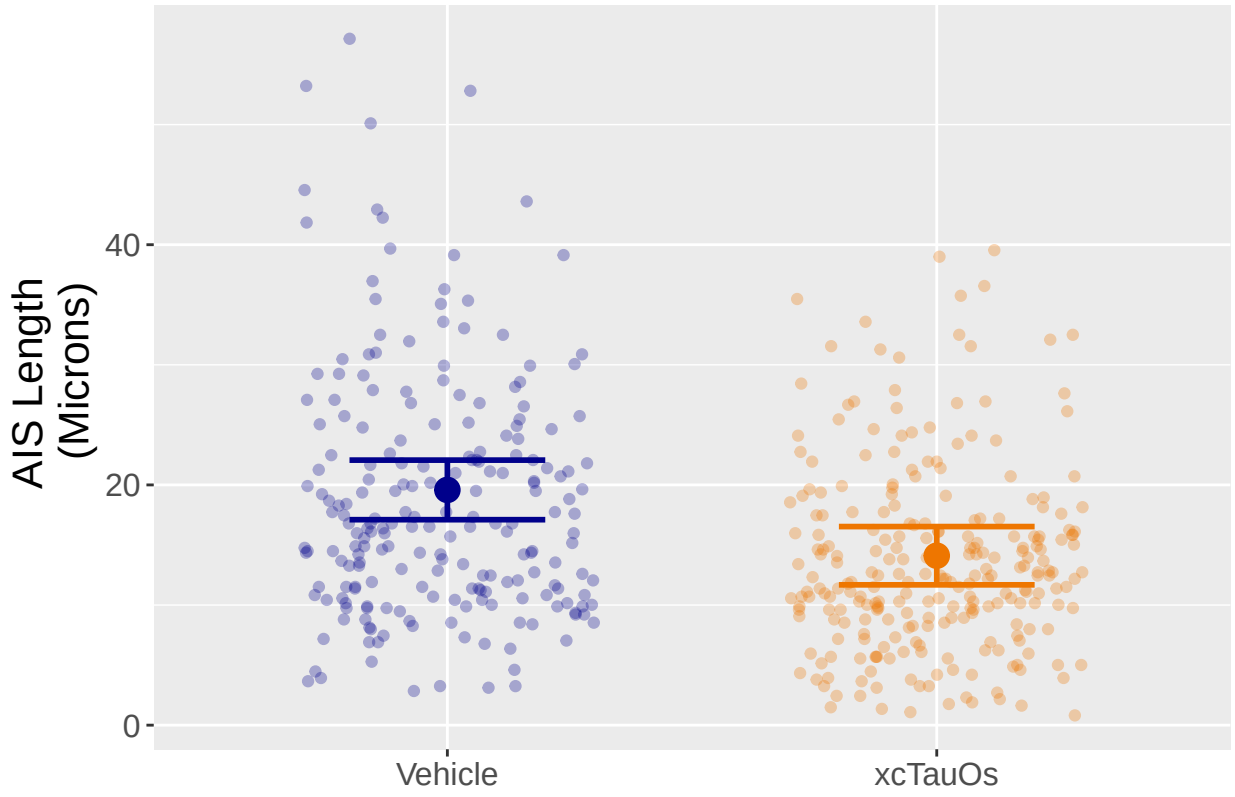
Jitter with 95% CI (like human data) - vehicle = blue - tau = orange

```
# return dataset of predicted mean +/- 95%CI
pred_length <- ggeffect(Fig3Fmod, terms = "Group") %>%
  as_tibble() %>%
  rename(Group = x)

pred_length
```

```
## # A tibble: 2 x 6
##   Group   predicted std.error conf.low conf.high group
##   <fct>       <dbl>     <dbl>    <dbl>     <dbl> <fct>
## 1 Vehicle      19.6      1.26     17.1      22.1 1
## 2 xcTauOs      14.1      1.23     11.7      16.5 1
```

```
lengthplot <- Fig3F %>%
  ggplot() +
  geom_jitter(aes(x = Group,
                  y = Length,
                  color = Group),
              alpha = .3,
              width = .3) +
  geom_point(data = pred_length,
                aes(x = Group,
                    y = predicted,
                    color = Group),
                size = 4) +
  geom_errorbar(data = pred_length,
```

```
                aes(x = Group,
                    ymin = conf.low,
                    ymax = conf.high,
                    color = Group),
                width = .4,
                lwd = 1) +
  labs(y = "AIS Length \n (Microns)", x = "") +
  scale_color_manual(values =  c("darkblue", "darkorange2")) +
  theme(legend.position = "none",
        axis.title = element_text(size = 16),
        axis.text = element_text(size = 12))

lengthplot
```




```
ggsave(lengthplot, filename = "Figures/cell_3F.png", width = 6, height = 4)
```