

Figures

Data

##Load Data

```
study_data <- read_csv("study_data2.csv")
diseasemodel_data <- read_csv("diseasemodel_data2.csv")
outcome_data <- read_csv("outcome_data2.csv")
```

My Data

```
study_data <- study_data %>%
  filter(Year >= 2018 & Year <= 2023, SystematicSearchName ==
         "2020-2023 Update", InvestigatorName == "Laurel Renton",
         StudyId != "6d537dfe-3d19-4dbf-b9af-bf9fad52bb4b") # this study was falsely included

outcome_data <- outcome_data %>%
  filter(Year >= 2018 & Year <= 2023, SystematicSearchName ==
         "2020-2023 Update", InvestigatorName == "Laurel Renton",
         StudyId != "6d537dfe-3d19-4dbf-b9af-bf9fad52bb4b")

diseasemodel_data <- diseasemodel_data %>%
  filter(Year >= 2018 & Year <= 2023, SystematicSearchName ==
         "2020-2023 Update", InvestigatorName == "Laurel Renton",
         StudyId != "6d537dfe-3d19-4dbf-b9af-bf9fad52bb4b")
```

Reporting Data

```
full_data_figures <- read_csv("full_data.csv") %>%
  filter(AnnotatorIdStr == "00374321-95a7-4a2a-a12c-af4c0f9e599b",
         StudyId != "6d537dfe-3d19-4dbf-b9af-bf9fad52bb4b")
```

Number of Studies Included

```
no_of_studies <- study_data %>%
  filter(QuestionId == "335bd414-ce8b-4127-a4ff-b02ec66a3b81",
         Answer == "Yes")
```

24 studies included after the screening process.

Transgenic APP Models of AD

Data Wrangling

```
diseasemodel_comments <- read_csv("comments.csv") %>%
  filter(InvestigatorName == "Laurel Renton") %>%
  select(StudyId, Answer, Comments)
# adds comments for the models that weren't listed in the
# SyRF project

model_data <- diseasemodel_data %>%
  left_join(diseasemodel_comments) %>%
  filter(QuestionId == "8255f1cf-1db1-4756-b77c-003df2860ed9") %>%
  select(StudyId, Answer, Comments) %>%
  unique() %>%
  mutate(Answer = if_else(!is.na(Comments), Comments, Answer),
         Answer = str_replace_all(Answer, "AppNL-G-F", "APP NL-G-F Knock-in"),
         Answer = str_replace_all(Answer, "NL-P-F", "APP NL-P-F Knock-in"),
         Count = 1) %>%
  select(Answer, Count) %>%
  group_by(Answer) %>%
  summarise(Count = sum(Count)) %>%
  filter(Answer != "PDGF-APP(WT) (line I5)")
```

Plotting Donut Plot

```
# Donut plot
hsize <- 4

color_palette <- c("#D1E7F6", "#75B6E5", "#77CEEF", "#1CADE4",
                  "#2683C6", "#7AE0E5", "#42BA97", "#146266", "#A1C8C5", "#D3F5F6",
                  "#BECAD4", "#597287", "#0E5772", "#134263", "#6E9281", "#59A79E")

donut_plot <- model_data %>%
  mutate(x = hsize) %>%
  ggplot(aes(x = hsize, y = Count, fill = Answer)) + geom_col(color = "black") +
  geom_text(aes(label = Count), position = position_stack(vjust = 0.5),
            size = 10, color = "white") + coord_polar(theta = "y") +
  scale_fill_manual(values = color_palette) + xlim(c(0.2, hsize +
0.5)) + theme(panel.background = element_rect(fill = "white"),
              panel.grid = element_blank(), axis.title = element_blank(),
              axis.ticks = element_blank(), axis.text = element_blank(),
              legend.text = element_text(size = 12)) + guides(fill = guide_legend(title = "AD APP Model"))
donut_plot
```

Study Quality Protocols

```
quality_data <- full_data_figures %>%
  select(StudyId, `Sample Size` = "Is.a.sample.size.calculation.reported.",
        `Blinding` = "Does.the.paper.report.that.experimenters.were.blind.to.experimental.groups.during.o",
        `Welfare Comittee Approval` = "Does.the.paper.report.that.experiments.were.approved.by.an.animal",
        `Conflict of Interest` = "Is.there.a.conflicts.of.interest.statement.",
        `Exclusion Criteria` = "Are.the.reasons.for.the.exclusion.of.data.points.reported.") %>%
  unique() %>%
  select(-StudyId) %>%
  pivot_longer(cols = everything(), names_to = "study_quality_items",
               values_to = "Value") %>%
  group_by(study_quality_items, Value) %>%
  summarize(Count = n()) %>%
  pivot_wider(names_from = Value, values_from = Count) %>%
  mutate(overall = 23, percentage_of_studies = `TRUE`/overall *
        100, across(where(is.numeric), ~round(., 2)))

quality_data_plot <- quality_data %>%
  ggplot(aes(x = reorder(study_quality_items, -percentage_of_studies),
               y = percentage_of_studies)) + geom_bar(stat = "identity",
  fill = "#174E77") + geom_text(aes(label = percentage_of_studies),
  position = position_dodge(width = 0.9), hjust = 1.25, color = "white",
  size = 3.5) + theme(plot.title = element_text(hjust = 0.5)) +
  ggtitle("Studies (%) Reporting Different Study Quality Items") +
  labs(x = "Study Quality Items", y = "Included (%)") + coord_flip() +
  scale_y_continuous(labels = abs, limits = c(0, 100))

quality_data_plot
```

Electrophysiology reporting quality

```
## Applies to all experiments
ephys_data_1 <- full_data_figures %>%
  filter(Outcome.category == "Electrophysiology") %>%
  select(StudyId, Outcome = "OutcomeLabel", Age = "What.age..weeks..were.animals.when.EPhys.outcomes.",
        `Brain Pathway` = "Brain.pathway.recorded", `Type of Recording Chamber` = "What.type.of.recordi")
  unique() %>%
  select(-c(StudyId, Outcome, Age)) %>%
  mutate(`Brain Pathway` = "Reported", `Type of Recording Chamber` = case_when(`Type of Recording Chamber`
    "Not reported" ~ "NotReported", TRUE ~ "Reported")) %>%
  pivot_longer(cols = c(`Brain Pathway`, `Type of Recording Chamber`),
               names_to = "variable") %>%
  group_by(variable, value) %>%
  summarise(count = n()) %>%
  pivot_wider(names_from = value, values_from = count) %>%
  na_replace(0) %>%
  mutate(Overall = Reported + NotReported)
```

```

## Applies to only experiments looking at LTP and LTD
ephys_data_2 <- full_data_figures %>%
  filter(Outcome.category == "Electrophysiology", OutcomeLabel %in%
    c("LTP", "LTD")) %>%
  select(StudyId, Outcome = "OutcomeLabel", `Stimulation Type (LTP)` = "What.type.of.stimulation.was.",
    `Stimulation Type (LTD)` = "What.type.of.stimulation.was.used.to.induce.LTD.",
    Age = "What.age..weeks..were.animals.when.EPhys.outcomes.were.assessed.",
    `Total Stimulations` = "Total.number.of.stimulations",
    `% of Maximal Response of Baseline Recording` = "Ephys..percentage....of.maximal.response.from")
  unique() %>%
  select(-StudyId, -Age) %>%
  unite("Stimulation Type", c("Stimulation Type (LTP)", "Stimulation Type (LTD)")) %>%
  mutate(`LTP/LTD Stimulation Type**` = case_when(`Stimulation Type` ==
    c("Other (please leave a comment)_", "_Other (please leave a comment)") ~
    "NotReported", TRUE ~ "Reported"), `Total Stimulations**` = case_when(`Total Stimulations` ==
    "NR" ~ "NotReported", TRUE ~ "Reported"), `% of Maximal Response of Baseline Recording**` = case_when(
    "NR" ~ "NotReported", TRUE ~ "Reported")) %>%
  select(-c(Outcome, `Stimulation Type`)) %>%
  pivot_longer(cols = c(`LTP/LTD Stimulation Type**`, `Total Stimulations**`,
    `% of Maximal Response of Baseline Recording**`), names_to = "variable") %>%
  group_by(variable, value) %>%
  summarise(count = n()) %>%
  pivot_wider(names_from = value, values_from = count) %>%
  mutate(Overall = Reported + NotReported)

ephys_data <- ephys_data_1 %>%
  full_join(ephys_data_2) %>%
  mutate(percentage = Reported/Overall, across(where(is.numeric),
    ~round(., 2))) %>%
  select(-c(NotReported, Reported, Overall))

ephys_table <- ephys_data %>%
  gt() %>%
  fmt_percent(columns = percentage, decimals = 1) %>%
  tab_header(title = md("***Reporting Quality of Electrophysiology Measures***")) %>%
  cols_label(percentage = "Experiments Reported*") %>%
  tab_source_note(source_note = "* 58 experiments across 24 studies assessing synaptic plasticity/ tr",
  tab_source_note(source_note = "** across 28 experiments assessing LTP/LTD") %>%
  tab_style(locations = cells_column_labels(columns = everything()),
    style = list(cell_borders(sides = "bottom", weight = px(3)),
    cell_text(weight = "bold"))) %>%
  cols_width(percentage ~ "70%")

ephys_table

```

Animal Husbandry

```

animal_data <- full_data_figures %>%
  select(StudyId, lightcycle = "Animal.husbandry..light.cycle..h..of.animal.facilities",
    animalspercentage = "Animal.husbandry..number.of.animals.per.cage",

```

```

    environmental = "Animal.husbandry..is.any.environmental.enrichment.reported.") %>%
unique() %>%
select(-StudyId) %>%
mutate(lightcycle = case_when(lightcycle == "NR" ~ "NotReported",
    TRUE ~ "Reported"), animalspercentage = case_when(animalspercentage ==
    "NR" ~ "NotReported", TRUE ~ "Reported"), environmental = case_when(environmental ==
    "FALSE" ~ "NotReported", TRUE ~ "Reported")) %>%
pivot_longer(cols = c(lightcycle, animalspercentage, environmental),
    names_to = "variable") %>%
group_by(variable, value) %>%
summarise(count = n()) %>%
pivot_wider(names_from = value, values_from = count) %>%
mutate(Overall = Reported + NotReported, percentage = Reported/Overall *
    100, across(where(is.numeric), ~round(., 2))) %>%
select(-c(NotReported, Reported, Overall))

```

Slice Protocol Measures

```

slice_data <- full_data_figures %>%
select(StudyId, `Ca concentration in slicing solution` = "Ephys..calcium.concentration..mM..of.slicing.solution",
    `Ca concentration in recording solution` = "Ephys..calcium.concentration..mM..of.recording.solution",
    `Mg concentration in slicing solution` = "Ephys..magnesium.concentration..mM..of.slicing.solution",
    `Mg concentration in recording solution` = "Ephys..magnesium.concentration..mM..of.ACSF.used.for.recording",
    `Plane slice taken from` = "Which.plane.was.the.slice.taken.from.",
    `Temperature slice left to recover at` = "Ephys..temperature...C..slices.left.to.recover.at",
    `Slice recovery time` = "How.long..minutes..were.brain.slices.left.to.recover.after.slicing.",
    `Kynurenic acid used` = "Ephys..was.kynurenic.acid.used.during.the.brain.dissection.",
    `Anaesthetised prior to sacrifice` = "Ephys..were.animals.anaesthetised.prior.to.decapitation.",
    `Sacrifice time of day` = "Ephys..what.time.of.day.were.animals.sacrificed.") %>%
unique() %>%
select(-StudyId) %>%
mutate(`Ca concentration in slicing solution` = case_when(`Ca concentration in slicing solution` ==
    "NR" ~ "NotReported", TRUE ~ "Reported"), `Ca concentration in recording solution` = case_when(
    "NR" ~ "NotReported", TRUE ~ "Reported"), `Mg concentration in slicing solution` = case_when(`Mg
    "NR" ~ "NotReported", TRUE ~ "Reported"), `Mg concentration in recording solution` = case_when(
    "NR" ~ "NotReported", TRUE ~ "Reported"), `Plane slice taken from` = case_when(`Plane slice taken
    "Not reported" ~ "NotReported", TRUE ~ "Reported"), `Temperature slice left to recover at` = ca
    "NR" ~ "NotReported", TRUE ~ "Reported"), `Slice recovery time` = case_when(`Slice recovery time
    "NR" ~ "NotReported", TRUE ~ "Reported"), `Kynurenic acid used` = case_when(`Kynurenic acid use
    "No" ~ "NotReported", TRUE ~ "Reported"), `Anaesthetised prior to sacrifice` = case_when(`Anaes
    "Not reported" ~ "NotReported", TRUE ~ "Reported"), `Sacrifice time of day` = case_when(`Sacrif
    "Not reported" ~ "NotReported", TRUE ~ "Reported"), ) %>%
pivot_longer(cols = c(`Ca concentration in slicing solution`,
    `Ca concentration in recording solution`, `Mg concentration in slicing solution`,
    `Mg concentration in recording solution`, `Plane slice taken from`,
    `Temperature slice left to recover at`, `Slice recovery time`,
    `Kynurenic acid used`, `Anaesthetised prior to sacrifice`,
    `Sacrifice time of day`), names_to = "variable") %>%
group_by(variable, value) %>%
summarise(count = n()) %>%
pivot_wider(names_from = value, values_from = count) %>%

```

```

mutate(Overall = Reported + NotReported, percentage = Reported/Overall *
      100, across(where(is.numeric), ~round(., 2))) %>%
select(-c(NotReported, Reported, Overall))

slice_plot <- slice_data %>%
  ggplot(aes(x = reorder(variable, percentage), y = percentage)) +
  geom_bar(stat = "identity", fill = "#2683C6") + geom_text(aes(label = percentage),
    position = position_dodge(width = 0.9), hjust = 1.25, color = "white",
    size = 3.5) + theme(plot.title = element_text(hjust = 0.5)) +
  ggtitle("Studies(%) Reporting Different Slice Protocol Measures") +
  labs(x = "Protocol Measure", y = "Included (%)") + coord_flip() +
  scale_y_continuous(labels = abs, limits = c(0, 100))

slice_plot

```

Age and Sex Data

```

age_sex_data <- full_data_figures %>%
  filter(Outcome.category == "Electrophysiology") %>%
  select(StudyId, Outcome = "OutcomeLabel", Age = "What.age..weeks..were.animals.when.EPhys.outcomes.",
    Sex = "Sex.of.animals.in.cohort") %>%
  unique()

sex_data <- age_sex_data %>%
  count(Sex) %>%
  mutate(overall = sum(n), percentage = n/overall * 100, across(where(is.numeric),
    ~round(., 2))) %>%
  ggplot(aes(x = Sex, y = percentage)) + ylim(0, 100) + geom_bar(stat = "identity",
    fill = "#2683C6") + geom_text(aes(label = percentage), vjust = 1.5,
    color = "white") + theme(plot.title = element_text(hjust = 0.5)) +
  ggtitle("Sex of Transgenic Animals Used") + labs(x = "Sex",
    y = "(%) Experiments Using Sex")

sex_data

```

```

age_count <- age_sex_data %>%
  count(Age)

age_data <- age_sex_data %>%
  mutate(Age = as.numeric(Age)) %>%
  ggplot(aes(x = Age)) + geom_histogram(aes(y = ..density..),
    fill = "#174E77") + theme(plot.title = element_text(hjust = 0.5)) +
  ggtitle("Age of Transgenic Animals Used") + labs(x = "Age (Weeks)",
    y = "Density of Experiments Using Age")

age_data

```

Publication Location

```
countries_included <- full_data_figures %>%
  select(region = "What.is.the.country.of.origin.of.the.corresponding.author",
    StudyId) %>%
  unique() %>%
  mutate(`Number of Studies` = 1) %>%
  select(-StudyId) %>%
  group_by(region) %>%
  summarise(`Number of Studies` = sum(`Number of Studies`)) %>%
  mutate(region = if_else(region == "United Kingdom", "UK",
    if_else(region == "United States", "USA", if_else(region ==
      "Korea, South", "South Korea", region))))

mapdata <- map_data("world") %>%
  left_join(countries_included, by = "region")

map1 <- mapdata %>%
  ggplot(aes(x = long, y = lat, group = group)) + geom_polygon(aes(fill = `Number of Studies`))
map1
```

Year Published

```
year_data <- full_data_figures %>%
  select(StudyId, Year) %>%
  unique() %>%
  mutate(`Number of Studies` = 1) %>%
  select(-StudyId) %>%
  group_by(Year) %>%
  summarise(`Number of Studies` = sum(`Number of Studies`)) %>%
  add_row(Year = 2023, `Number of Studies` = 0)

year_plot <- year_data %>%
  ggplot(aes(x = Year, y = `Number of Studies`, group = 1)) +
  geom_line() + geom_point() + geom_smooth(method = "lm", se = FALSE,
  col = "#174E77")

year_plot
```