

# lab3\_slides\_supp

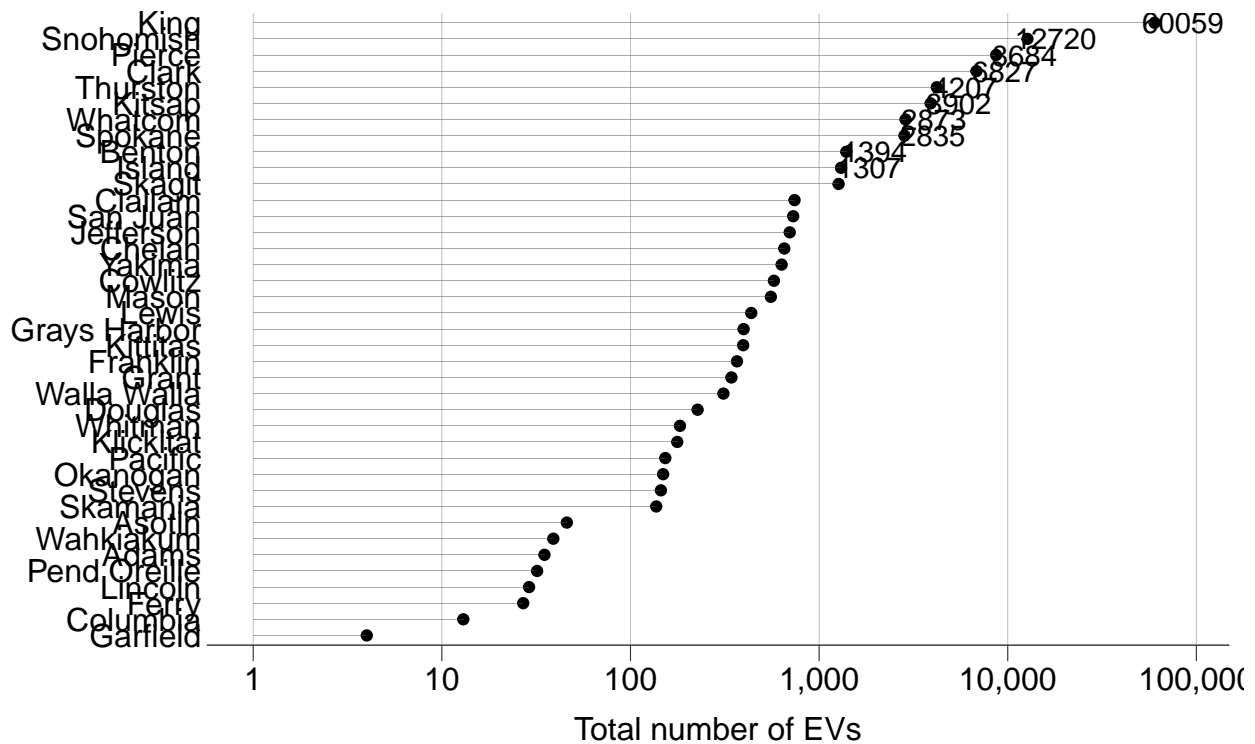
Ramses Llobet

2026-01-20

Which county has most EVs?

```
# Count number of EVs by county
ev_by_county <-
  ev_data |>
  count(county) |>
  mutate(county = fct_reorder(county, n)) |>
  arrange(desc(county))

# Visualize
ggplot(ev_by_county,
       aes(y = county, x = n)) +
  theme_cavis_vgrid +
  geom_point() +
  geom_col(width = 0.05,
          alpha = 0.5) +
  scale_x_continuous(trans = "log10",
                    breaks = 10 ^ (0:5),
                    labels = scales::label_comma(),) +
  geom_text(data = slice_max(ev_by_county, n, n = 10),
          aes(label = n),
          nudge_x = 0.15) +
  labs(y = NULL, x = "Total number of EVs")
```



### Trend of number of EVs across county

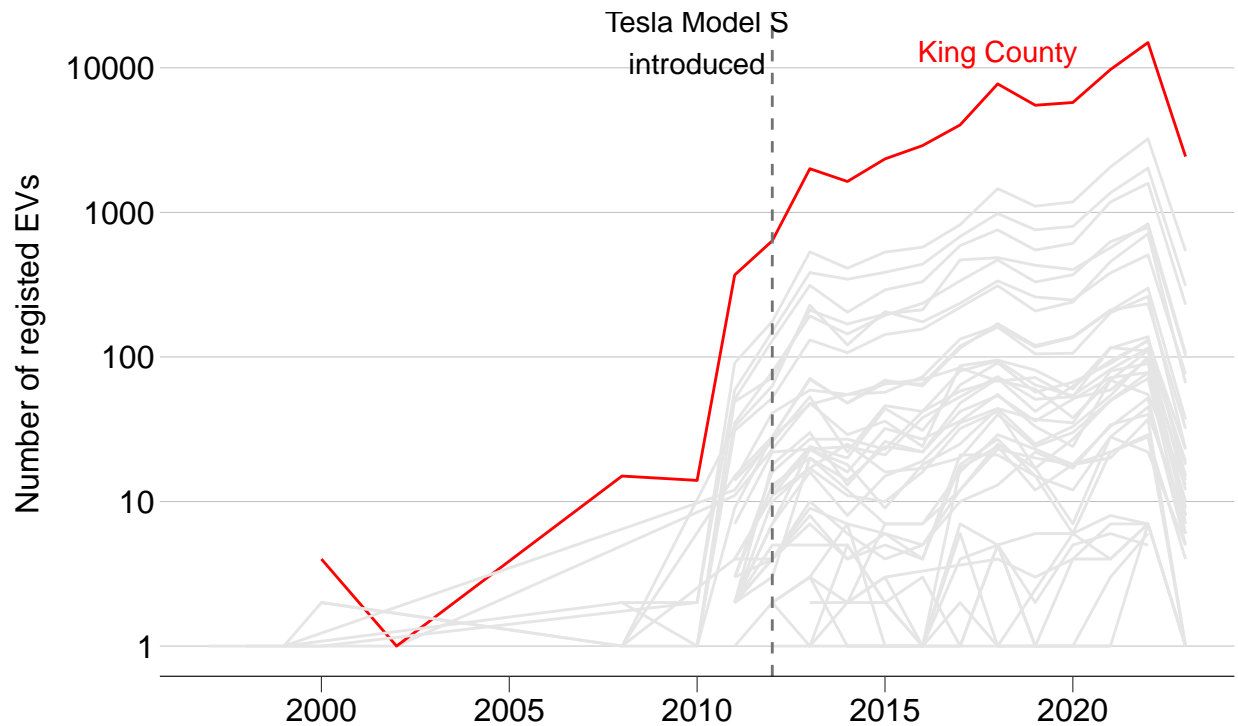
```
# Count number of EVs by county-year
ev_by_countyYear <-
  ev_data |>
  group_by(model_year, county) |>
  count()

# Visualize
ggplot(ev_by_countyYear,
  aes(x = model_year,
    y = n,
    group = county,
    color = county == "King")) +
  theme_cavis_hgrid +
  geom_line() +
  scale_y_continuous(trans = "log10") +
  scale_color_manual(values = c("grey90", "red")) +
  guides(color = "none") +
  annotate("text", x = 2018, y = 13000,
    label = "King County",
    color = "red") +
  geom_vline(xintercept = 2012,
    linetype = 2,
    color = "grey45") +
```

```

annotate("text", x = 2010, y = 15000,
        label = "Tesla Model S\nintroduced",
        color = "black") +
labs(y = "Number of registered EVs", x = NULL)

```



## Relationship between price and range

```

# Find unique models
ev_models <-
  ev_data |>
  distinct(make, model, model_year, ev_type, electric_range, retail_price) |>
  filter(retail_price > 0 & electric_range > 0)

# Pick colors
colors <- brewer.pal(n = 5, name = "Set1")
blue <- colors[2]
orange <- colors[5]

# Visualize
ggplot(ev_models,
       aes(x = electric_range, y = retail_price,
           color = ev_type, fill = ev_type)) +
  theme_cavis_hgrid +
  geom_point(alpha = 0.75) +
  geom_smooth(method = "lm",

```

```

    alpha = 0.15, size = 0.45) +
  scale_x_continuous(trans = "log2") +
  scale_y_continuous(trans = "log10",
    labels = label_dollar()) +
  scale_color_manual(values = c(blue, orange)) +
  scale_fill_manual(values = c(blue, orange)) +
  geom_text_repel(
    aes(label = paste(make, model, model_year)),
    show.legend = FALSE,
    size = 3, alpha = 0.75
  ) +
  theme(legend.position = "top") +
  labs(y = "Retail price", x = "Electric range (mile)")

```

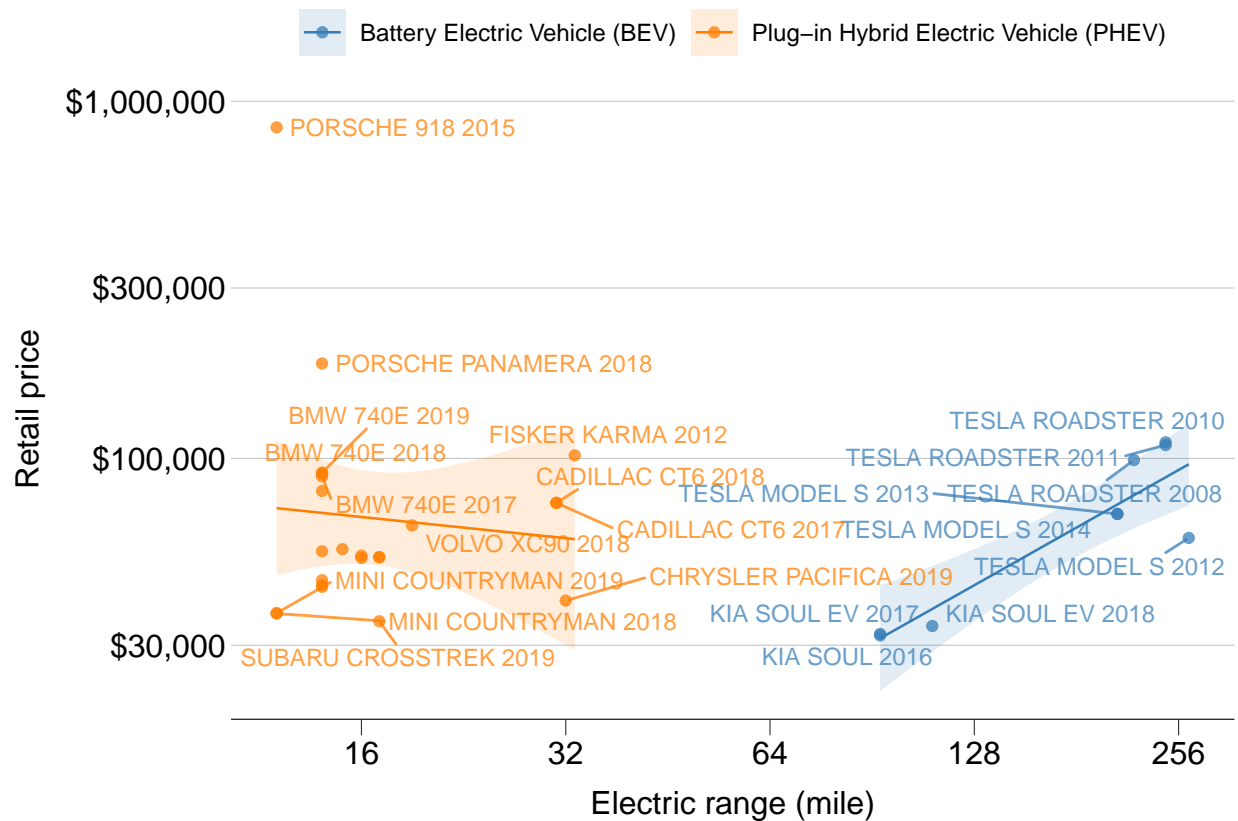
```

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

## `geom_smooth()` using formula = 'y ~ x'

## Warning: ggrepel: 10 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps

```



## Save a graph

```
width <- 12
ggsave("ev_bivariate.pdf", width = width, height = width/1.618)

## `geom_smooth()` using formula = 'y ~ x'

## Warning: ggrepel: 5 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

## Shares of makes across counties

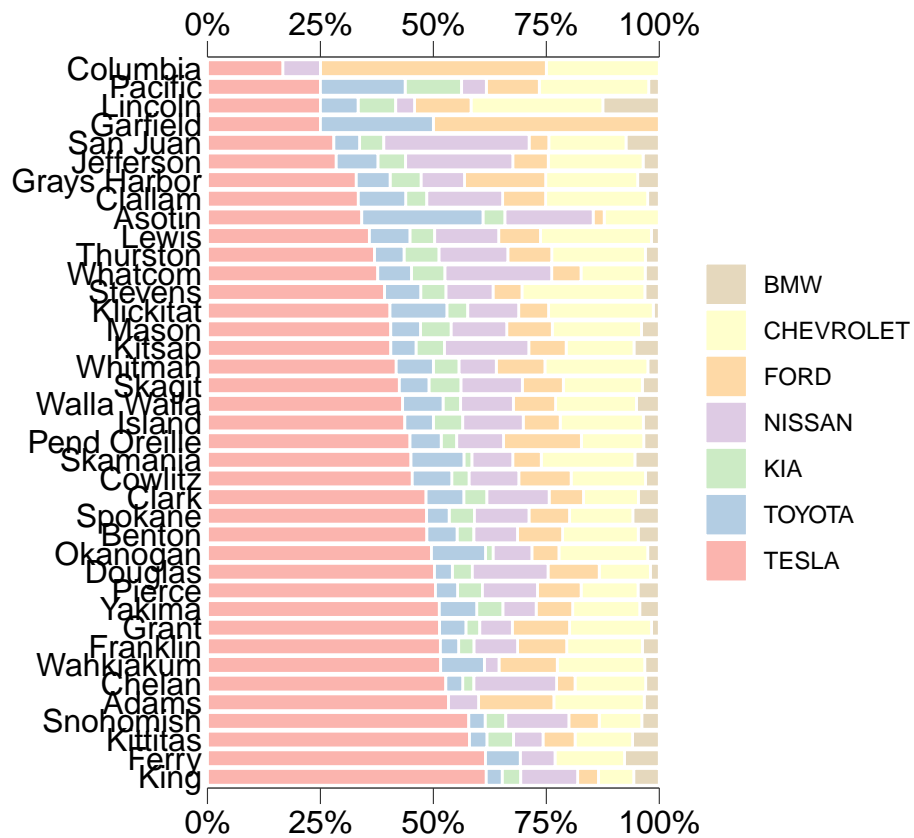
```
# Focus on the top 7 most popular makes
top7_make <-
  ev_data |>
  count(make) |>
  slice_max(order_by = n, n = 7)

# Shares of top-7 makes across county
make_share_byCounty <-
  ev_data |>
  group_by(county, make) |>
  count() |>
  filter(make %in% top7_make$make) |>
  group_by(county) |>
  mutate(county_sum = sum(n), prop = n / county_sum)

# Coerce `make` into factor and put TESLA as last level
make_share_byCounty <-
  make_share_byCounty |>
  mutate(make = as.factor(make),
         make = fct_relevel(make, "TESLA", after = Inf))

# Visualize
ggplot(make_share_byCounty,
       aes(x = prop,
           y = fct_reorder2(county, make, prop),
           fill = make)) +
  theme_cavis +
  geom_bar(
    position = "stack", stat = "identity",
    color = "white"
  ) +
  scale_fill_brewer(
    type = "qual",
    palette = 4,
    direction = -1
  ) +
  scale_x_continuous(
    expand = c(0, 0),
    labels = label_percent(),
    sec.axis = dup_axis()
  ) +
  theme(aspect.ratio = 1.618) +
```

```
labs(y = NULL, x = NULL)
```



## Num. of EVs and median income

```
# Compute num of EVs per hh
ev_by_income <-
  ev_data |>
  group_by(county) |>
  count() |>
  left_join(county_data, by = "county") |>
  mutate(n_EV_perhh = n / hh_num * 1000)

# Find outliers
outliners <-
  lm(n_EV_perhh ~ hh_income, ev_by_income) |>
  broom::augment(ev_by_income) |>
  slice_max(order_by = abs(.resid), n = 10) |>
  pull(county)

# Visualize
ggplot(ev_by_income,
  aes(x = hh_income, y = n_EV_perhh)) +
  theme_cavis_hgrid +
  geom_point(alpha = 0.75, color = blue) +
  scale_x_continuous(trans = "log10",
```

```

        labels = label_dollar()) +
scale_y_continuous(trans = "log2") +
geom_smooth(
  method = "lm",
  alpha = 0.2,
  color = blue,
  fill = blue
) +
labs(y = "N of EVs per 1,000 household",
     x = "Median household income") + #BREAK
geom_text_repel(data = filter(ev_by_income, county %in% outliers),
               aes(label = county))

```

## `geom\_smooth()` using formula = 'y ~ x'

