Data Modeling

Gustavo Mello

2022-03-27

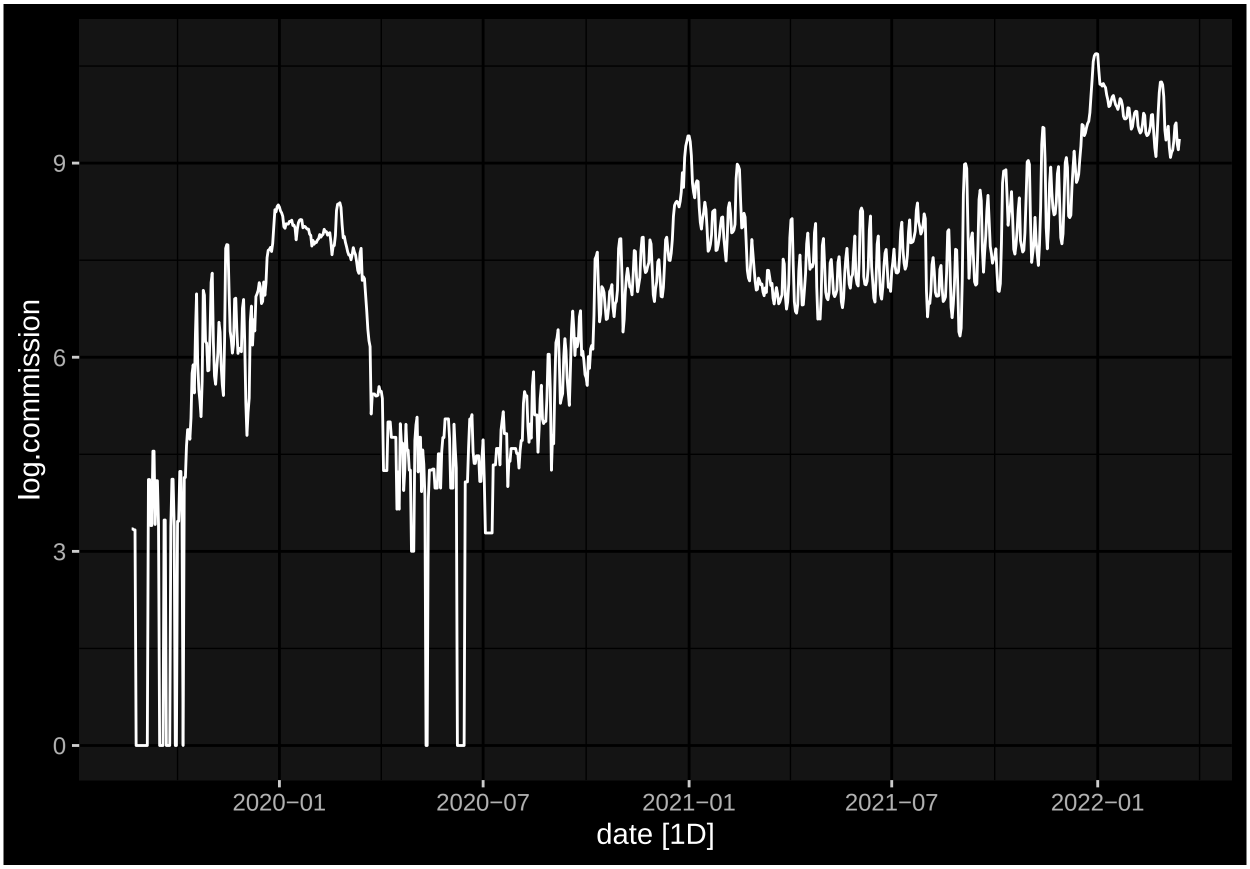
Data loading and preparing

```
cwd <- getwd()
daily.revenue.listings <- read_csv(paste0(cwd, "/../data/output/daily_revenue_listings.csv"))
```

```
## Rows: 289021 Columns: 26
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (6): listing, Localização, Categoria, Hotel, Status, Tipo
## dbl  (17): last_offered_price, occupancy, revenue, blocked, reservation_adva...
## date  (3): date, creation_date, Data.Inicial.do.contrato
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

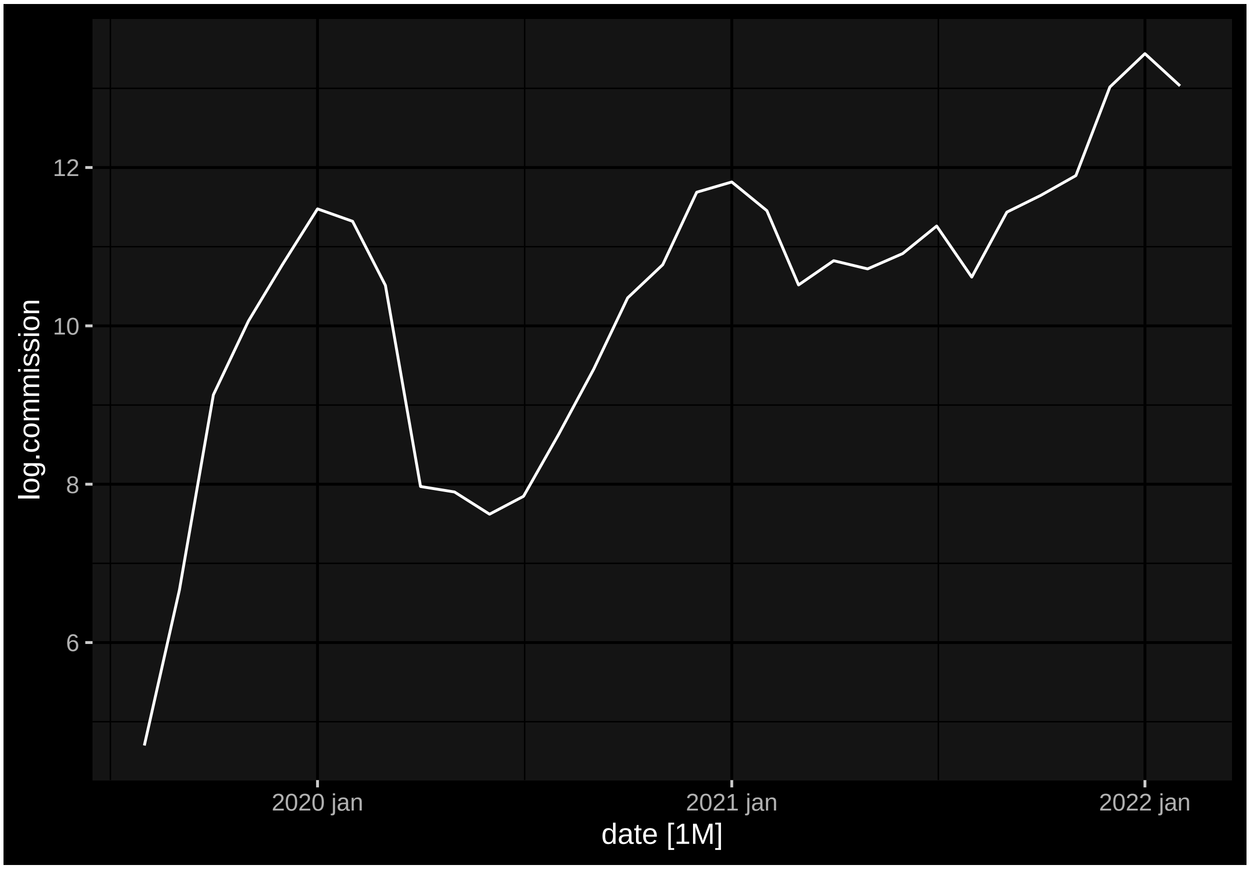
```
daily.commissions <- daily.revenue.listings %>%
  select(date, commission) %>%
  group_by(date) %>%
  summarise(commission=sum(commission)) %>%
  as_tsibble(index=date) %>%
  filter(date<="2022-03-15") %>%
  mutate(log.commission=log(commission + 1)) %>%
  select(-c(commission))

autoplot(daily.commissions)
```

```
## Plot variable not specified, automatically selected `.vars = log.commission`
```



```
monthly.commissions <- daily.revenue.listings %>%
  select(date, commission) %>%
  mutate(date=yearmonth(date)) %>%
  group_by(date) %>%
  summarise(commission=sum(commission)) %>%
  as_tsibble(index=date) %>%
  filter_index(~"2022-02") %>%
  mutate(log.commission=log(commission + 1)) %>%
  select(-c(commission))

autoplot(monthly.commissions)
```

```
## Plot variable not specified, automatically selected `.vars = log.commission`
```



Model 1: Monthly Data

Exponential Smoothing (ETS method)

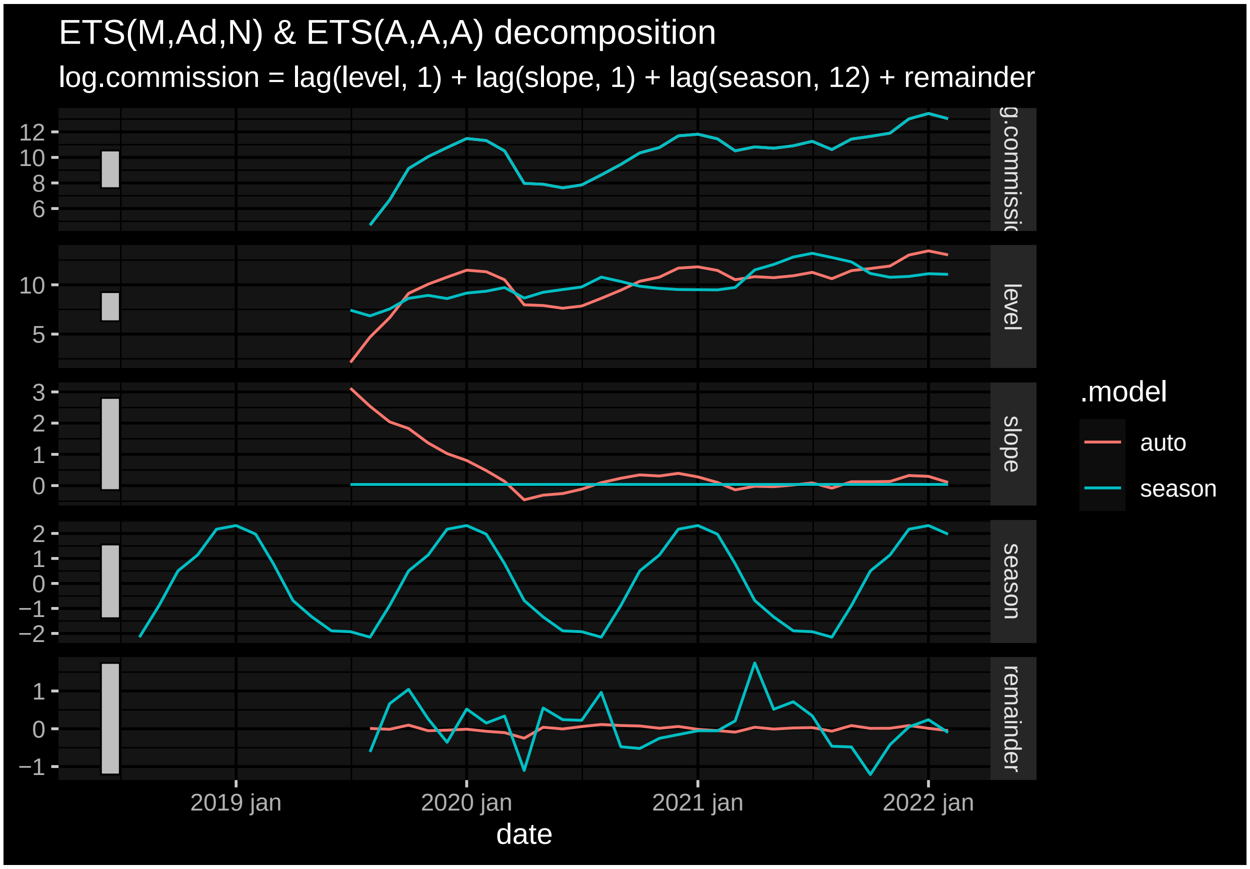
```
fit <- monthly.commissions %>%
  model(auto=ETS(log.commission),
        season=ETS(log.commission ~ trend("A") + season("A")))

report(fit)
```

```
## Warning in report.mdl_df(fit): Model reporting is only supported for individual
## models, so a glance will be shown. To see the report for a specific model, use
## `select()` and `filter()` to identify a single model.
```

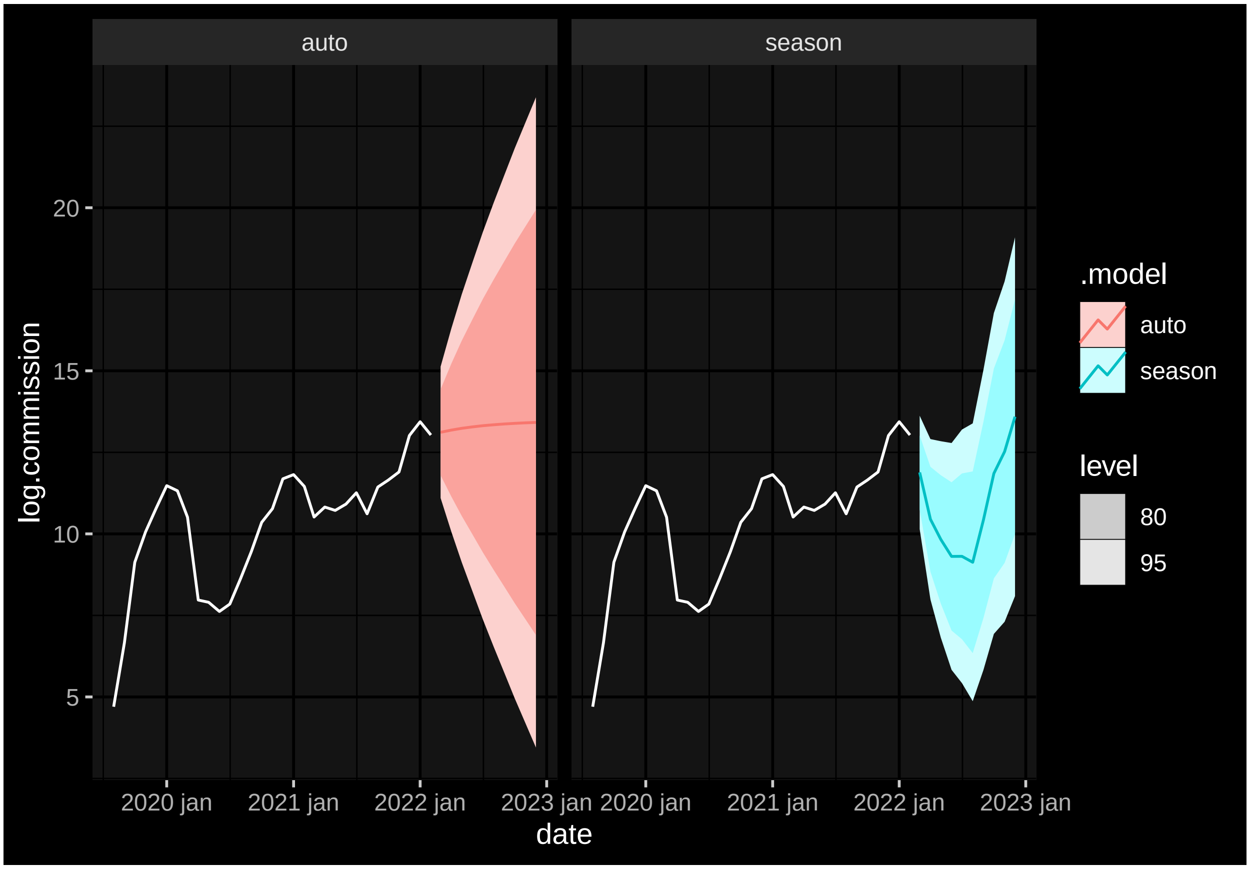
```
## # A tibble: 2 x 9
##   .model  sigma2 log_lik   AIC  AICc   BIC   MSE  AMSE    MAE
##   <chr>    <dbl>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 auto   0.00614   -43.3  98.5  102.  107. 0.561 2.08  0.0533
## 2 season 0.787     -38.3 111.   158.  135. 0.381 0.993 0.485
```

```
components(fit) %>%
  autoplot()
```

```
## Warning: Removed 13 row(s) containing missing values (geom_path).
```



```
fit %>%
  forecast(h = 10) %>%
  autoplot(monthly.commissions) +
  facet_wrap(vars(.model))
```

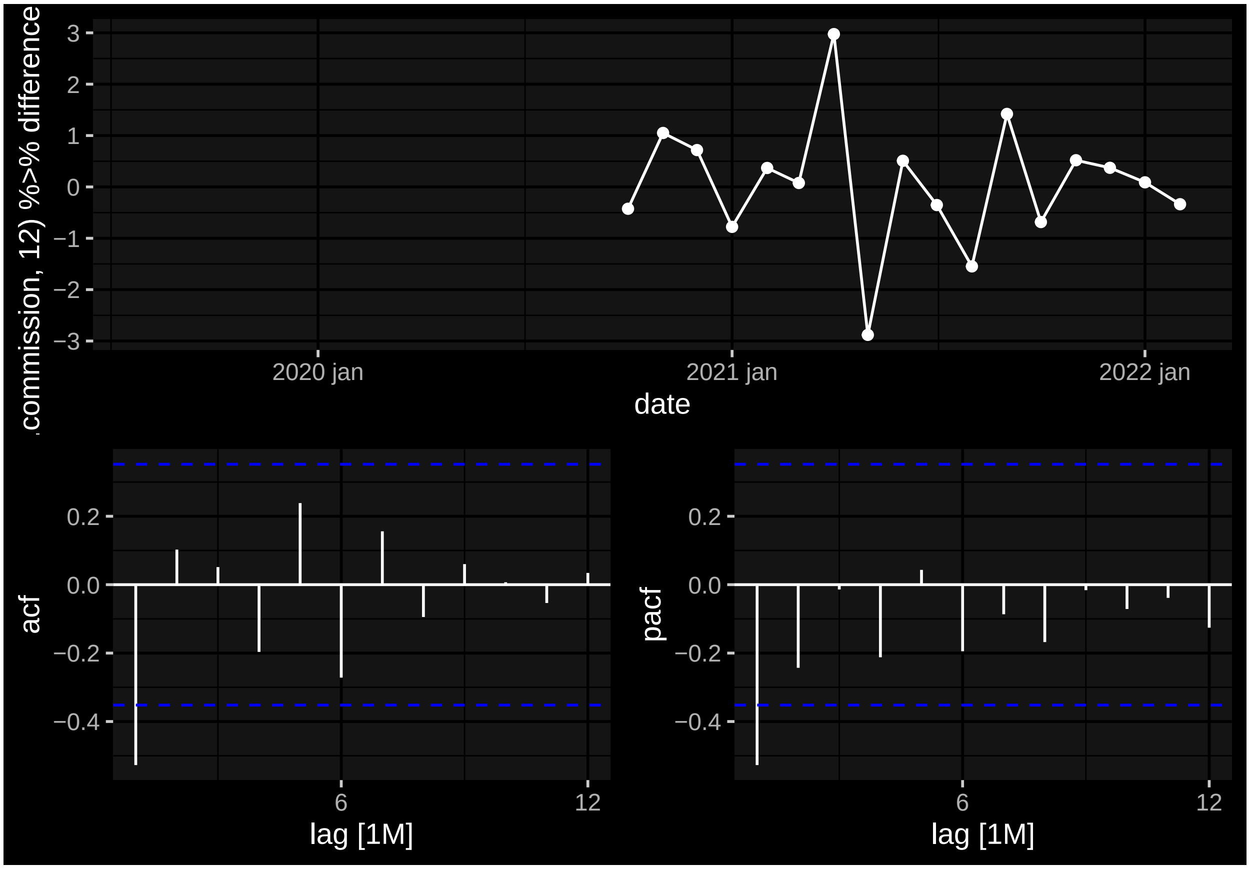


ARIMA model

```
monthly.commissions %>%
  gg_tsdisplay(difference(log.commission, 12) %>% difference () %>% difference(), plot_type='partial')
```

```
## Warning: Removed 14 row(s) containing missing values (geom_path).
```

```
## Warning: Removed 14 rows containing missing values (geom_point).
```



```
fit <- monthly.commissions %>%
  model(manual   = ARIMA(log.commission ~ pdq(0, 2, 0) + PDQ(0, 1, 0)),
        stepwise = ARIMA(log.commission),
        search   = ARIMA(log.commission, stepwise=FALSE),
        auto     = ARIMA(log.commission, stepwise=FALSE, approx=FALSE))
```

```
## Warning: Having 3 or more differencing operations is not recommended. Please
## consider reducing the total number of differences.

## Warning: It looks like you're trying to fully specify your ARIMA model but have not said if a constan
## You can include a constant using `ARIMA(y~1)` to the formula or exclude it by adding `ARIMA(y~0)`.

## Warning: 1 error encountered for manual
## [1] There are no ARIMA models to choose from after imposing the `order_constraint`, please consider
```

```
report(fit %>% select(auto))
```

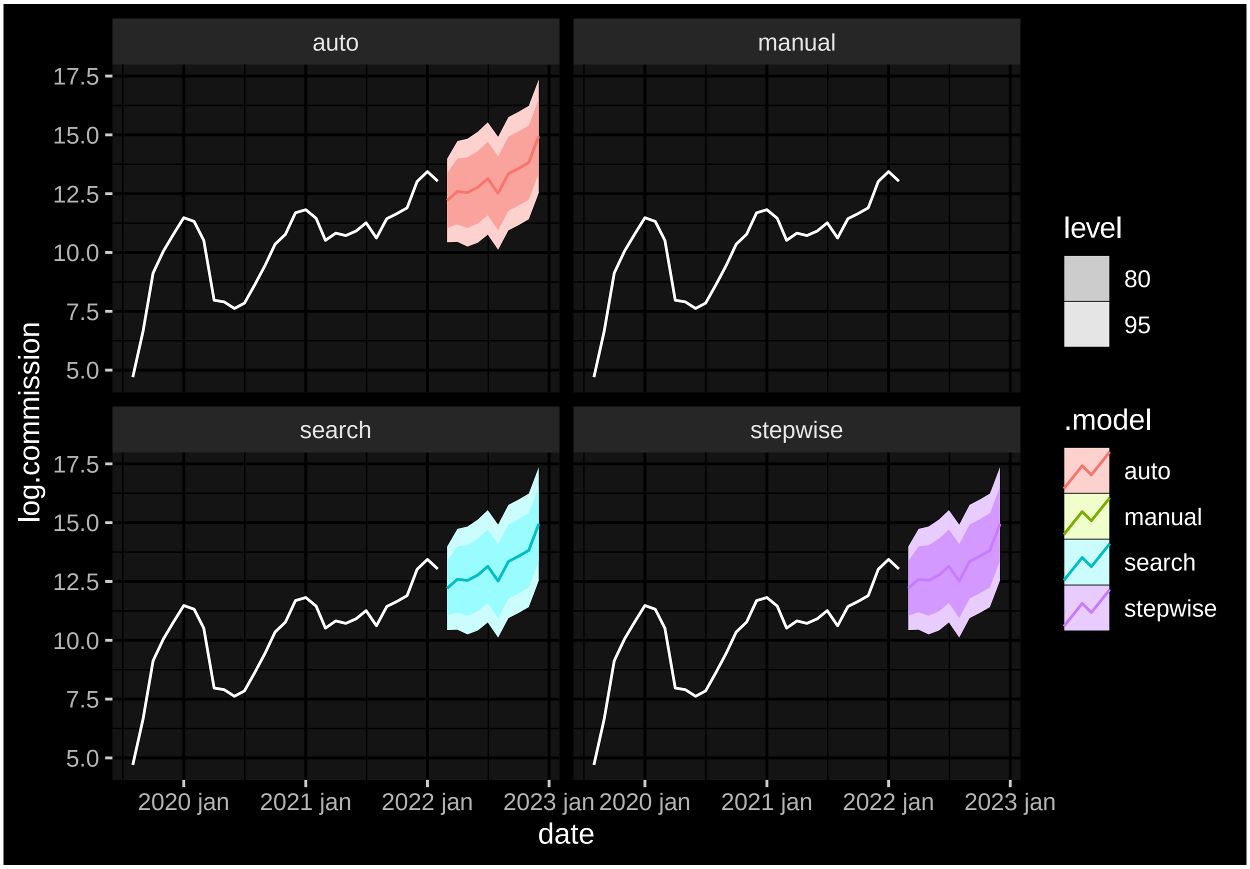
```
## Series: log.commission
## Model: ARIMA(1,0,0)(0,1,0)[12] w/ drift
##
## Coefficients:
##          ar1  constant
##       0.6780    0.6232
## s.e.  0.1787    0.1819
##
## sigma^2 estimated as 0.8187:  log likelihood=-24.31
## AIC=54.62   AICc=56.22   BIC=57.45
```

```
fit %>%
  forecast(h=10) %>%
  autoplot(monthly.commissions) +
  facet_wrap(vars(.model))
```

```
## Warning in max(ids, na.rm = TRUE): nenhum argumento não faltante para max;
## retornando -Inf
```

```
## Warning in max(ids, na.rm = TRUE): nenhum argumento não faltante para max;
## retornando -Inf
```

```
## Warning: Removed 10 row(s) containing missing values (geom_path).
```



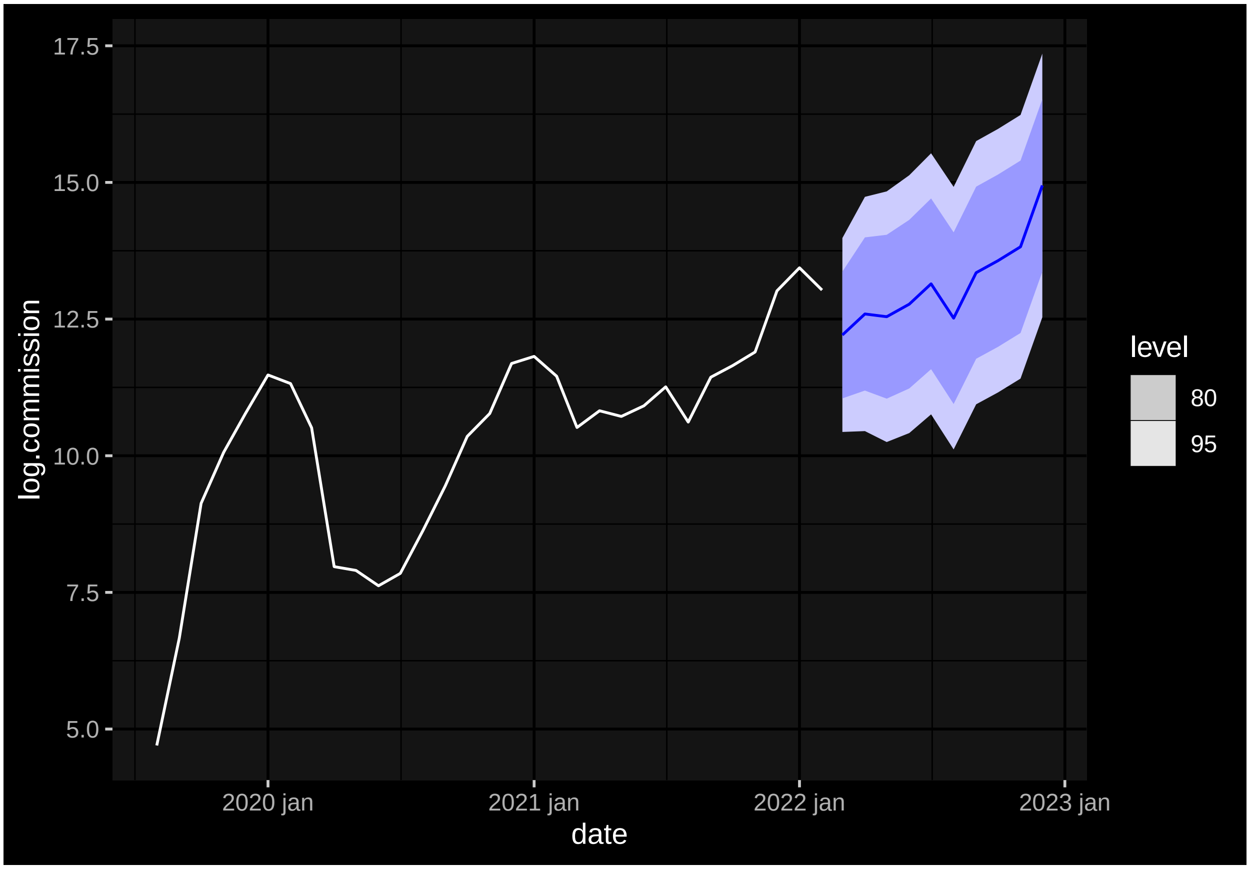
Cross-validation to select best fit

```
monthly.commissions %>%
  filter_index(~ "2022-01") %>%
  stretch_tsibble(.init = 10) %>%
  model(
    ETS(log.commission),
    ARIMA(log.commission)
  ) %>%
  forecast(h = 1) %>%
  accuracy(monthly.commissions) %>%
  select(.model, RMSE:MAPE)
```

```
## # A tibble: 2 x 5
##   .model                  RMSE   MAE   MPE  MAPE
##   <chr>                  <dbl> <dbl> <dbl> <dbl>
## 1 ARIMA(log.commission) 0.542 0.400 2.13   3.75
## 2 ETS(log.commission)   0.622 0.529 0.934  4.94
```

Arima predictions in log(commissions)

```
monthly.commissions.forecast <- monthly.commissions %>%
  model(ARIMA(log.commission)) %>%
  forecast(h=10)

monthly.commissions.forecast %>%
  autoplot(monthly.commissions)
```



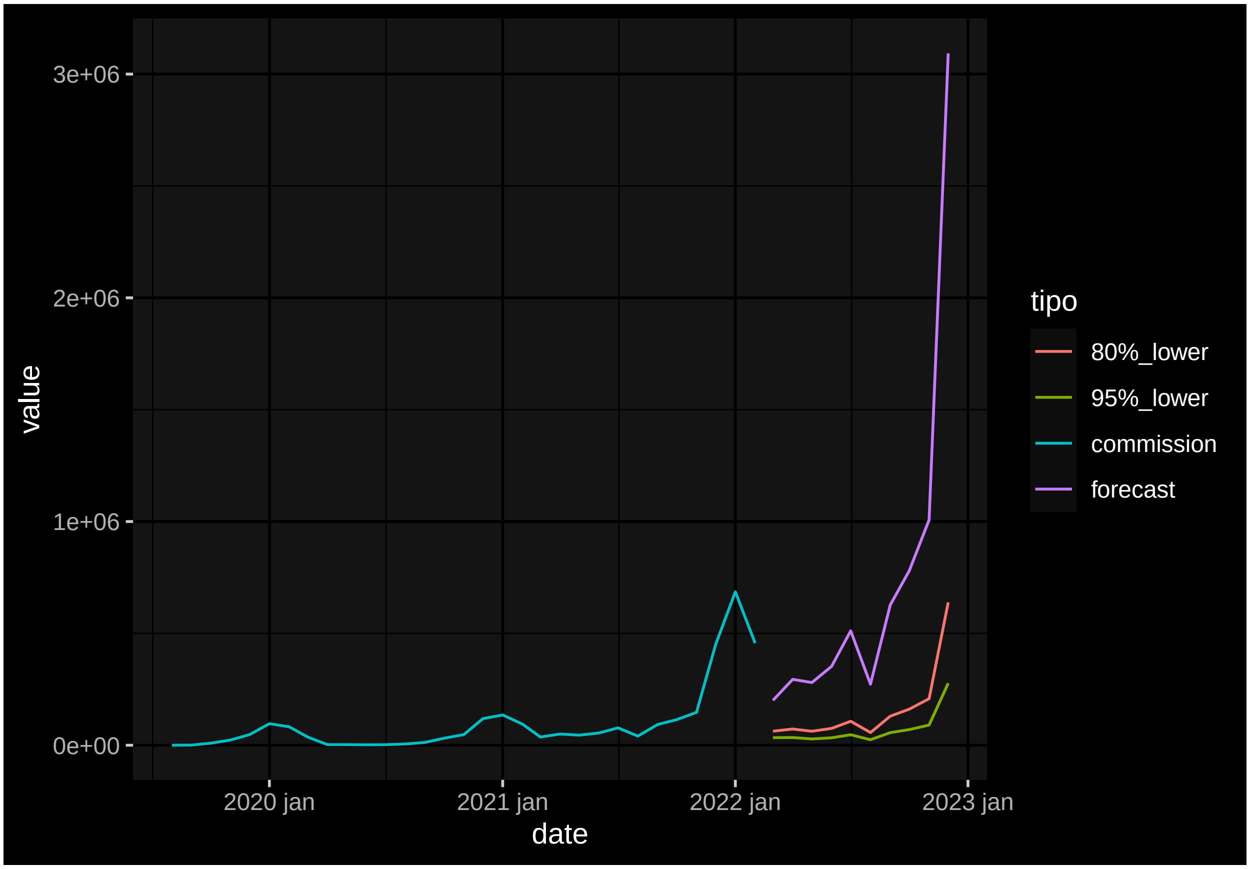
```
monthly.commissions.forecasted <- monthly.commissions.forecast %>%
  hilo() %>%
  unpack_hilo(c("80%", "95%")) %>%
  mutate(across(-(.model:log.commission), ~exp(.))) %>%
  select(-c(.model, log.commission)) %>%
  rename(forecast=.mean)

monthly.commissions.actual.and.forecasted <- monthly.commissions %>%
  mutate(commission=exp(log.commission)) %>%
  select(-c(log.commission)) %>%
  full_join(monthly.commissions.forecasted, by=c("date"="date"))
```

```
monthly.commissions.actual.and.forecasted
```

```
## # A tsibble: 41 x 7 [1M]
##        date commission forecast `80%_lower` `80%_upper` `95%_lower` `95%_upper`
##       <mth>      <dbl>    <dbl>       <dbl>       <dbl>       <dbl>       <dbl>
##  1 2019 ago       110.       NA          NA          NA          NA          NA
##  2 2019 set       783.       NA          NA          NA          NA          NA
##  3 2019 out      9216.       NA          NA          NA          NA          NA
##  4 2019 nov     23425.       NA          NA          NA          NA          NA
##  5 2019 dez     47736.       NA          NA          NA          NA          NA
##  6 2020 jan     96531.       NA          NA          NA          NA          NA
##  7 2020 fev     82504.       NA          NA          NA          NA          NA
##  8 2020 mar     36732.       NA          NA          NA          NA          NA
##  9 2020 abr      2896.       NA          NA          NA          NA          NA
## 10 2020 mai      2702.       NA          NA          NA          NA          NA
## # ... with 31 more rows
```

```
monthly.commissions.actual.and.forecasted %>%
  select(-contains("upper")) %>%
  pivot_longer(!date, names_to="tipo", values_to="value") %>%
  ggplot(aes(x=date, y=value, color=tipo)) +
  geom_line()
```

```
## Warning: Removed 103 row(s) containing missing values (geom_path).
```



```
revenue_by_year <- monthly.commissions.actual.and.forecasted %>%
  as_tibble() %>%
  select(date, commission, forecast, "80%_lower", "80%_upper") %>%
  rename(upper80="80%_upper", lower80="80%_lower") %>%
  mutate(year=year(date)) %>%
  group_by(year) %>%
  summarise(realized=sum(commission, na.rm=TRUE),
            forecasted_mean=sum(forecast, na.rm=TRUE),
            upper80=sum(upper80, na.rm=TRUE),
            lower80=sum(lower80, na.rm=TRUE))

revenue_by_year
```

```
## # A tibble: 4 x 5
##    year realized forecasted_mean   upper80  lower80
##   <dbl>    <dbl>           <dbl>     <dbl>    <dbl>
## 1  2019   81270.               0         0        0
## 2  2020  442459.               0         0        0
## 3  2021 1339324.               0         0        0
## 4  2022 1142239.        7419885. 35137965. 1575767.
```

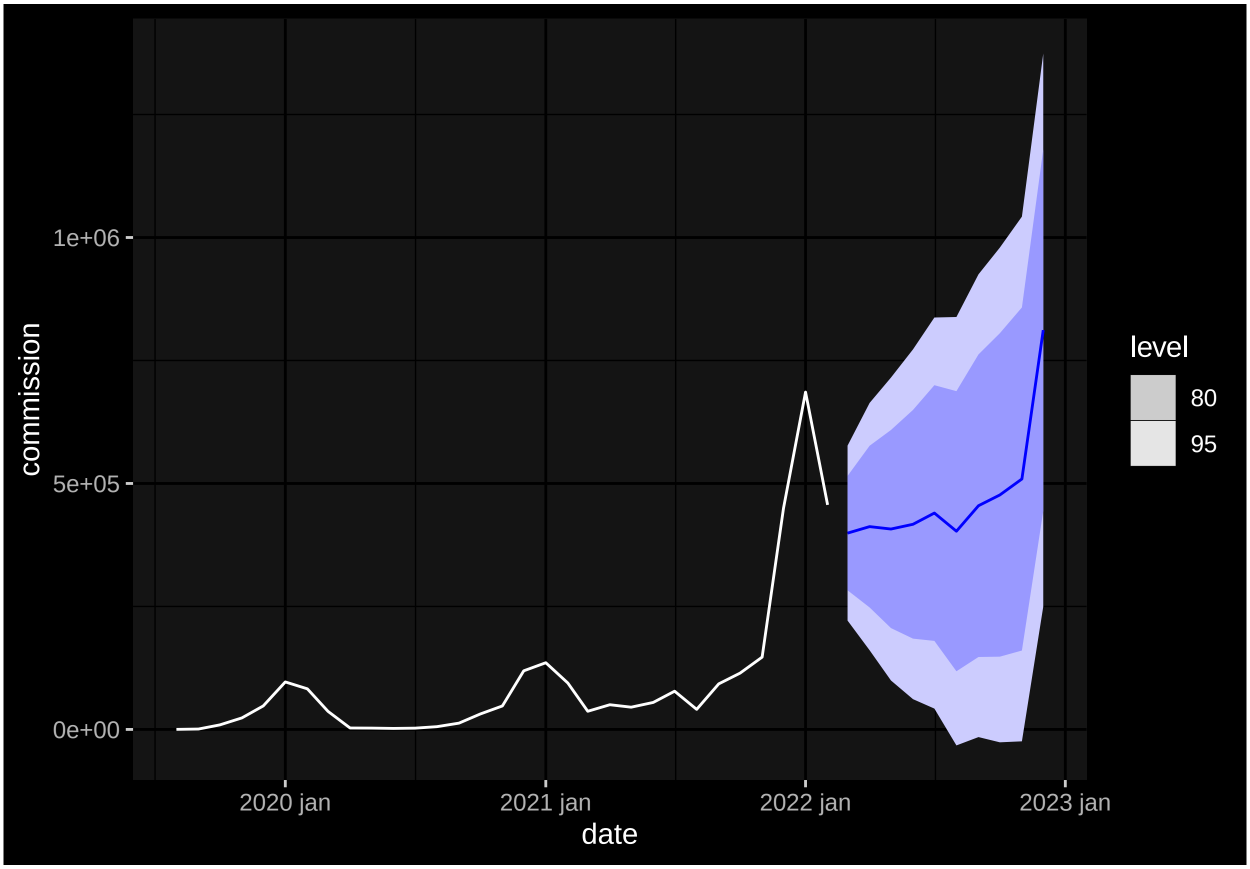
Log(commission) vs commission models comparison

```
monthly.commissions.no.log <- daily.revenue.listings %>%
  select(date, commission) %>%
  mutate(date=yearmonth(date)) %>%
  group_by(date) %>%
  summarise(commission=sum(commission)) %>%
  as_tsibble(index=date) %>%
  filter_index(~"2022-02")

monthly.commissions.no.log %>%
  filter_index(~ "2022-01") %>%
  stretch_tsibble(.init = 10) %>%
  model(
    ETS(log(commission)),
    ARIMA(log(commission)),
    ETS(commission),
    ARIMA(commission)
  ) %>%
  forecast(h = 1) %>%
  accuracy(monthly.commissions.no.log) %>%
  select(.model, RMSE:MAPE)
```

```
## # A tibble: 4 x 5
##   .model                    RMSE     MAE    MPE  MAPE
##   <chr>                    <dbl>   <dbl>  <dbl> <dbl>
## 1 ARIMA(commission)       93014.  53875.  25.5   51.6
## 2 ARIMA(log(commission)) 126318.  74019. -32.4   51.5
## 3 ETS(commission)         99532.  55677.   8.13  47.6
## 4 ETS(log(commission))   194420. 103490. -65.3   83.5
```

ARIMA predictions

```r
monthly.commissions.forecast <- monthly.commissions.no.log %>%
  model(ARIMA(commission)) %>%
  forecast(h=10)

monthly.commissions.forecast %>%
  autoplot(monthly.commissions.no.log)
```

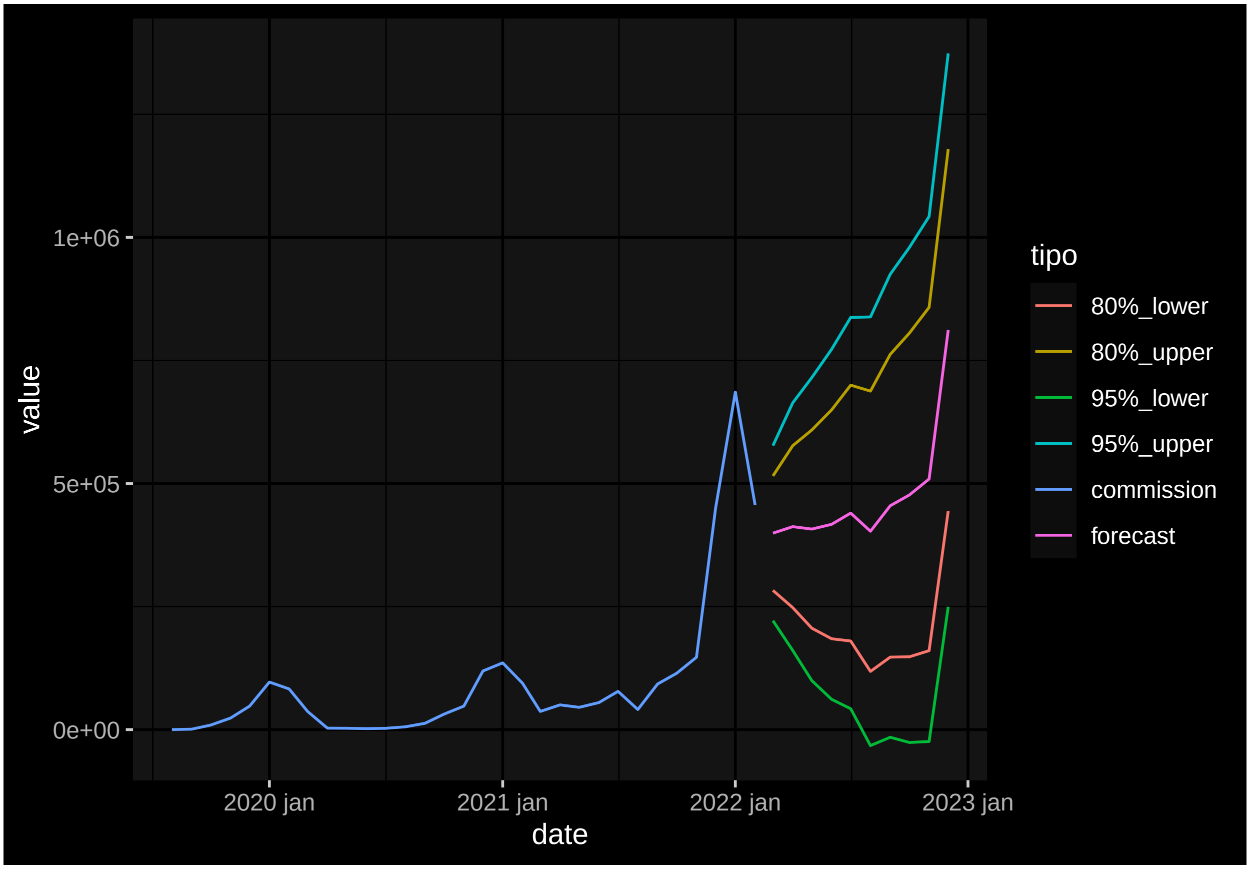


```r
monthly.commissions.forecasted <- monthly.commissions.forecast %>%
  hilo() %>%
  unpack_hilo(c("80%", "95%")) %>%
  select(-c(.model, commission)) %>%
  rename(forecast=.mean)

monthly.commissions.actual.and.forecasted <- monthly.commissions.no.log %>%
  full_join(monthly.commissions.forecasted, by=c("date"="date"))

monthly.commissions.actual.and.forecasted
```

```
## # A tsibble: 41 x 7 [1M]
##        date commission forecast `80%_lower` `80%_upper` `95%_lower` `95%_upper`
##       <mth>      <dbl>    <dbl>       <dbl>       <dbl>       <dbl>       <dbl>
##  1 2019 ago       109.       NA          NA          NA          NA          NA
##  2 2019 set       782.       NA          NA          NA          NA          NA
##  3 2019 out      9215.       NA          NA          NA          NA          NA
##  4 2019 nov     23424.       NA          NA          NA          NA          NA
##  5 2019 dez     47735.       NA          NA          NA          NA          NA
##  6 2020 jan     96530.       NA          NA          NA          NA          NA
##  7 2020 fev     82503.       NA          NA          NA          NA          NA
##  8 2020 mar     36731.       NA          NA          NA          NA          NA
##  9 2020 abr      2895.       NA          NA          NA          NA          NA
## 10 2020 mai      2701.       NA          NA          NA          NA          NA
## # ... with 31 more rows
```

```
monthly.commissions.actual.and.forecasted %>%
  pivot_longer(!date, names_to="tipo", values_to="value") %>%
  ggplot(aes(x=date, y=value, color=tipo)) +
  geom_line()
```

```
## Warning: Removed 165 row(s) containing missing values (geom_path).
```

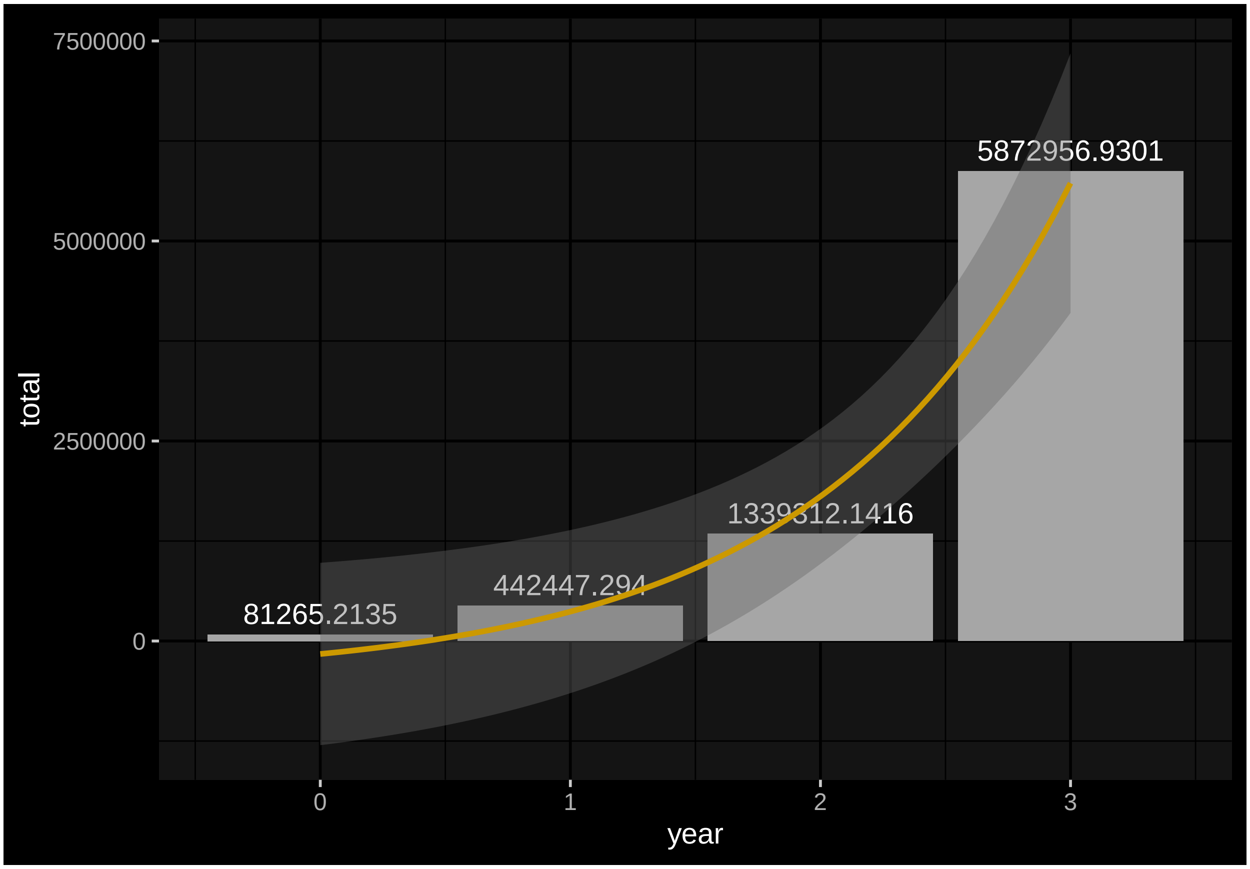


```
revenue_by_year <- monthly.commissions.actual.and.forecasted %>%
  as_tibble() %>%
  select(date, commission, forecast, "80%_lower", "80%_upper") %>%
  rename(upper80="80%_upper", lower80="80%_lower") %>%
  mutate(year=year(date)) %>%
  group_by(year) %>%
  summarise(realized=sum(commission, na.rm=TRUE),
            forecasted_mean=sum(forecast, na.rm=TRUE),
            upper80=sum(upper80, na.rm=TRUE),
            lower80=sum(lower80, na.rm=TRUE)) %>%
  mutate(total_lower=realized+lower80,
         total=realized+forecasted_mean,
         total_upper=realized+upper80)
```

```
revenue_by_year
```

```
## # A tibble: 4 x 8
##    year realized forecasted_mean  upper80 lower80 total_lower  total total_upper
##   <dbl>    <dbl>           <dbl>    <dbl>   <dbl>       <dbl>  <dbl>       <dbl>
## 1  2019   81265.               0        0  0            81265. 8.13e4      81265.
## 2  2020  442447.               0        0  0           442447. 4.42e5     442447.
## 3  2021 1339312.               0        0  0          1339312. 1.34e6    1339312.
## 4  2022 1142237.         4730720. 7342358.  2.12e6    3261319. 5.87e6    8484595.
```

```
revenue_by_year %>%
  mutate(year=year-2019) %>%
  ggplot(aes(x=year, y=total, label=total)) +
  geom_col() +
  geom_text(vjust=-0.5) +
  geom_smooth(method = "lm", formula = y ~ exp(x))
```


total
year