Patronage Forecasting

Lachlan Moody 27809951

2025-02-21

Libraries

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(fpp3)
```

```
## -- Attaching packages -------------------------------------------- fpp3 0.4.0 --
```

```
## v lubridate   1.8.0     v feasts      0.2.2
## v tsibble     1.1.1     v fable       0.3.1
## v tsibbledata 0.4.0
```

```
## -- Conflicts ------------------------------------------------- fpp3_conflicts --
## x lubridate::date()    masks base::date()
## x dplyr::filter()      masks stats::filter()
## x tsibble::intersect() masks base::intersect()
## x tsibble::interval()  masks lubridate::interval()
## x dplyr::lag()         masks stats::lag()
## x tsibble::setdiff()   masks base::setdiff()
## x tsibble::union()     masks base::union()
```

```
library(readxl)
library(janitor)
```

```
##
## Attaching package: 'janitor'
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(lubridate)
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

Data

```
## Historic Population

erp_victoria <- read_xlsx("Data/ERP - Victoria.xlsx", skip = 2) %>%
  filter(is.na(Number) == F) %>%
  clean_names()  %>%
  mutate(date = dmy(paste("01-07", year_ending_june_30, sep = "-"))) %>%
  select(date,
         erp_victoria = number)


erp_melb <- read_xlsx("Data/ERP - Greater Melbourne.xlsx", skip = 2) %>%
  filter(is.na(Number) == F) %>%
  clean_names()  %>%
  mutate(date = dmy(paste("01-07", year_ending_june_30, sep = "-"))) %>%
  select(date,
         erp_greater_melbourne = number)
```

```
## Forecasted Population

abs_victoria <- read_xlsx("Data/Projected population, components of change and summary statistics, 2022 (base) to 2071.xlsx", sheet = "Vic.", skip = 48, n_max = 1) %>%
  select(-`Total growth`:-`145339`) %>%
  pivot_longer(everything(), names_to = "discard", values_to = "erp_victoria") %>%
  mutate(year = seq(from = 2024, to = 2071)) %>%
  mutate(date = dmy(paste("01-07", year, sep = "-"))) %>%
  select(date,
         erp_victoria) %>%
  filter(date <= "2029-07-01")

abs_melb <- read_xlsx("Data/Projected population, components of change and summary statistics, 2022 (base) to 2071.xlsx", sheet = "Greater Melbourne", skip = 48, n_max = 1) %>%
  select(-`Total growth`:-`129347`) %>%
  pivot_longer(everything(), names_to = "discard", values_to = "erp_greater_melbourne") %>%
  mutate(year = seq(from = 2024, to = 2071)) %>%
  mutate(date = dmy(paste("01-07", year, sep = "-"))) %>%
  select(date,
         erp_greater_melbourne) %>%
  filter(date <= "2029-07-01")
```

```
# Combined Population

population_combined <- bind_rows(erp_victoria, abs_victoria) %>%
  left_join(
    bind_rows(erp_melb,
              abs_melb),
    by = "date"
  )
```

```
# Historic Jobs

historic_jobs <- read_csv("Data/employment-by-block-by-clue-industry.csv") %>%
  clean_names() %>%
  filter(clue_small_area %in% c("Melbourne (CBD)", "City of Melbourne (total)")) %>%
  group_by(census_year, clue_small_area) %>%
  summarise(jobs = sum(total_jobs_in_block, na.rm = TRUE)) %>%
  pivot_wider(names_from = clue_small_area, values_from = jobs) %>%
  clean_names()  %>%
  ungroup() %>%
  mutate(date = dmy(paste("01-07", census_year, sep = "-"))) %>%
  select(date,
         city_of_melbourne_jobs = city_of_melbourne_total,
         melbourne_cbd_jobs = melbourne_cbd) %>%
  filter(date != "2023-07-01")
```

```
## Rows: 13000 Columns: 24
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): CLUE small area
## dbl (23): Census year, Block ID, Accommodation, Admin and Support Services, ...
##
## 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.
## `summarise()` has grouped output by 'census_year'. You can override using the `.groups` ar
gument.
```

```
# Forecasted Jobs

forecasted_jobs <- read_csv("Data/city-of-melbourne-jobs-forecasts-by-small-area-2020-2040.cs
v") %>%
  clean_names() %>%
  filter(geography %in% c("Melbourne (CBD)", "City of Melbourne"),
         industry_space_use == "Total Jobs",
         category == "Jobs by industry") %>%
  select(geography, year, value) %>%
  pivot_wider(names_from = geography, values_from = value) %>%
  clean_names() %>%
  mutate(date = dmy(paste("01-07", year, sep = "-"))) %>%
  select(date,
         city_of_melbourne_jobs = city_of_melbourne,
         melbourne_cbd_jobs = melbourne_cbd) %>%
  filter(date <= "2029-07-01")
```

```
## Rows: 8505 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): Geography, Category, Industry Space Use
## dbl (2): Year, Value
##
## 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.
```

```
# Combined Jobs

jobs_combined <- bind_rows(historic_jobs, forecasted_jobs)
```

```
touch_ons_monthly <- read_csv("Data/monthly_touch_ons.csv") %>%
  mutate(date = ym(`Calendar Mth`)) %>%
  select(date, touch_ons = `Myki TouchOn`)
```

```
## Rows: 134 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Calendar Mth
## dbl (1): Myki TouchOn
##
## 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.
```

```
## WHF Data

wfh_pre_cov <- 5
wfh_cov <- 0
wfh_2022 <- 2
wfh_2023 <- 2.5
wfh_2024 <- 3
wfh_growth_rate_slow <- 0.025
wfh_growth_rate_med <- 0.05
wfh_growth_rate_fast <- 0.10

wfh_data <- tibble(date = seq.Date(from = as_date("2014-07-01"), to = as_date("2029-07-01"),
by = "1 year")) %>%
  mutate(wfh_base_rate = case_when(date <= "2019-07-01" ~ wfh_pre_cov,
                                   date <= "2021-07-01" ~ wfh_cov,
                                   date == "2022-07-01" ~ wfh_2022,
                                   date == "2023-07-01" ~ wfh_2023,
                                   TRUE ~ wfh_2024),
         slow_growth_rate = if_else(date <= "2024-07-01",
                                    1,
                                    1 + wfh_growth_rate_slow),
         wfh_slow_growth = wfh_base_rate * cumprod(slow_growth_rate),
         medium_growth_rate = if_else(date <= "2024-07-01",
                                    1,
                                    1 + wfh_growth_rate_med),
         wfh_medium_growth = wfh_base_rate * cumprod(medium_growth_rate),
         high_growth_rate = if_else(date <= "2024-07-01",
                                    1,
                                    1 + wfh_growth_rate_fast),
         wfh_high_growth = wfh_base_rate * cumprod(high_growth_rate)) %>%
  select(date, wfh_slow_growth, wfh_medium_growth, wfh_high_growth)
```

Forecast Prep

```
## Lets start with annual for now, can interpolate months later if required

touch_ons_for_forecast <- touch_ons_monthly %>%
  mutate(year = year(date)) %>%
  group_by(year) %>%
  summarise(touch_ons = sum(touch_ons)) %>%
  filter(year != 2025) %>%
  ungroup() %>%
  mutate(date = dmy(paste("01-07", year, sep = "-"))) %>%
  select(date, touch_ons)


forecast_data <- touch_ons_for_forecast %>%
  left_join(population_combined) %>%
  left_join(jobs_combined) %>%
  left_join(wfh_data)
```

```
## Joining, by = "date"
## Joining, by = "date"
## Joining, by = "date"
```

```
forecast_data %>%
  filter(date <= "2019-07-01") %>%
  select(-date) %>%
  cor()
```

```
## Warning in cor(.): the standard deviation is zero
##                        touch_ons erp_victoria erp_greater_melbourne
## touch_ons              1.0000000    0.9255571             0.9315344
## erp_victoria           0.9255571    1.0000000             0.9998373
## erp_greater_melbourne  0.9315344    0.9998373             1.0000000
## city_of_melbourne_jobs 0.7331661    0.9308269             0.9242525
## melbourne_cbd_jobs     0.6914782    0.8950610             0.8895301
## wfh_slow_growth               NA           NA                    NA
## wfh_medium_growth             NA           NA                    NA
## wfh_high_growth               NA           NA                    NA
##                        city_of_melbourne_jobs melbourne_cbd_jobs
## touch_ons                           0.7331661          0.6914782
## erp_victoria                        0.9308269          0.8950610
## erp_greater_melbourne               0.9242525          0.8895301
## city_of_melbourne_jobs              1.0000000          0.9478659
## melbourne_cbd_jobs                  0.9478659          1.0000000
## wfh_slow_growth                            NA                 NA
## wfh_medium_growth                          NA                 NA
## wfh_high_growth                            NA                 NA
##                        wfh_slow_growth wfh_medium_growth wfh_high_growth
## touch_ons                           NA                NA              NA
## erp_victoria                        NA                NA              NA
## erp_greater_melbourne               NA                NA              NA
## city_of_melbourne_jobs              NA                NA              NA
## melbourne_cbd_jobs                  NA                NA              NA
## wfh_slow_growth                      1                NA              NA
## wfh_medium_growth                   NA                 1              NA
## wfh_high_growth                     NA                NA               1
```

```
forecast_data %>%
  filter(date >= "2022-07-01") %>%
  select(-date) %>%
  cor()
```

```
##                       touch_ons erp_victoria erp_greater_melbourne
## touch_ons             1.0000000    0.9996651             0.9986353
## erp_victoria          0.9996651    1.0000000             0.9996523
## erp_greater_melbourne 0.9986353    0.9996523             1.0000000
## city_of_melbourne_jobs 0.9585578   0.9508644             0.9423704
## melbourne_cbd_jobs    0.9857654    0.9810846             0.9756394
## wfh_slow_growth       0.9813645    0.9760634             0.9699896
## wfh_medium_growth     0.9813645    0.9760634             0.9699896
## wfh_high_growth       0.9813645    0.9760634             0.9699896
##                       city_of_melbourne_jobs melbourne_cbd_jobs
## touch_ons                          0.9585578          0.9857654
## erp_victoria                       0.9508644          0.9810846
## erp_greater_melbourne              0.9423704          0.9756394
## city_of_melbourne_jobs             1.0000000          0.9928121
## melbourne_cbd_jobs                 0.9928121          1.0000000
## wfh_slow_growth                    0.9954393          0.9997016
## wfh_medium_growth                  0.9954393          0.9997016
## wfh_high_growth                    0.9954393          0.9997016
##                       wfh_slow_growth wfh_medium_growth wfh_high_growth
## touch_ons                   0.9813645         0.9813645       0.9813645
## erp_victoria                0.9760634         0.9760634       0.9760634
## erp_greater_melbourne       0.9699896         0.9699896       0.9699896
## city_of_melbourne_jobs      0.9954393         0.9954393       0.9954393
## melbourne_cbd_jobs          0.9997016         0.9997016       0.9997016
## wfh_slow_growth             1.0000000         1.0000000       1.0000000
## wfh_medium_growth           1.0000000         1.0000000       1.0000000
## wfh_high_growth             1.0000000         1.0000000       1.0000000
```

```
forecast_data %>%
  filter(date >= "2019-07-01") %>%
  select(-date) %>%
  cor()
```

```
##                        touch_ons erp_victoria erp_greater_melbourne
## touch_ons              1.0000000    0.2206047             0.2916083
## erp_victoria           0.2206047    1.0000000             0.9902487
## erp_greater_melbourne  0.2916083    0.9902487             1.0000000
## city_of_melbourne_jobs 0.8200029    0.6635303             0.6858116
## melbourne_cbd_jobs     0.8608595    0.6147971             0.6402535
## wfh_slow_growth        0.9921233    0.1801622             0.2436030
## wfh_medium_growth      0.9921233    0.1801622             0.2436030
## wfh_high_growth        0.9921233    0.1801622             0.2436030
##                        city_of_melbourne_jobs melbourne_cbd_jobs
## touch_ons                           0.8200029          0.8608595
## erp_victoria                        0.6635303          0.6147971
## erp_greater_melbourne               0.6858116          0.6402535
## city_of_melbourne_jobs              1.0000000          0.9943570
## melbourne_cbd_jobs                  0.9943570          1.0000000
## wfh_slow_growth                     0.7947805          0.8377939
## wfh_medium_growth                   0.7947805          0.8377939
## wfh_high_growth                     0.7947805          0.8377939
##                        wfh_slow_growth wfh_medium_growth wfh_high_growth
## touch_ons                    0.9921233         0.9921233       0.9921233
## erp_victoria                 0.1801622         0.1801622       0.1801622
## erp_greater_melbourne        0.2436030         0.2436030       0.2436030
## city_of_melbourne_jobs       0.7947805         0.7947805       0.7947805
## melbourne_cbd_jobs           0.8377939         0.8377939       0.8377939
## wfh_slow_growth              1.0000000         1.0000000       1.0000000
## wfh_medium_growth            1.0000000         1.0000000       1.0000000
## wfh_high_growth              1.0000000         1.0000000       1.0000000
```

Forecast Modelling

```
forecast_ts <- forecast_data %>%
  mutate(year = year(date)) %>%
  as_tsibble(index = year)

## Lets try standard arima, vic pop, greater melb pop, com jobs, cbd jobs, wfh


models <- forecast_ts %>%
  model(
    stepwise = ARIMA(touch_ons),
    search = ARIMA(touch_ons, stepwise = FALSE, approximation = FALSE),
    everything = TSLM(touch_ons ~ erp_victoria + erp_greater_melbourne + city_of_melbourne_jo
bs + melbourne_cbd_jobs + wfh_slow_growth),
    vic_com = TSLM(touch_ons ~ erp_victoria + city_of_melbourne_jobs + wfh_slow_growth),
    melb_cbd = TSLM(touch_ons ~ erp_greater_melbourne + melbourne_cbd_jobs + wfh_slow_growt
h),
    vic_pop = TSLM(touch_ons ~ erp_victoria + wfh_slow_growth),
    melb_pop = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth),
    com_job = TSLM(touch_ons ~ city_of_melbourne_jobs + wfh_slow_growth),
    cbd_job = TSLM(touch_ons ~ melbourne_cbd_jobs + wfh_slow_growth)
  )


models %>%
  report() %>%
  arrange(AICc)
```

```
## Warning in report.mdl_df(.): 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: 9 x 17
##   .model      sigma2 log_lik   AIC  AICc   BIC ar_roots  ma_roots  r_squared
##   <chr>        <dbl>   <dbl> <dbl> <dbl> <dbl> <list>    <list>        <dbl>
## 1 melb_pop   4.95e13   -187.  351.  358.  353. <NULL>    <NULL>        0.987
## 2 vic_pop    5.40e13   -188.  352.  359.  354. <NULL>    <NULL>        0.986
## 3 com_job    7.05e13   -189.  355.  362.  357. <NULL>    <NULL>        0.981
## 4 cbd_job    7.61e13   -190.  356.  363.  358. <NULL>    <NULL>        0.980
## 5 melb_cbd   4.62e13   -186.  351.  363.  353. <NULL>    <NULL>        0.989
## 6 vic_com    5.39e13   -187.  353.  365.  355. <NULL>    <NULL>        0.987
## 7 everything 4.21e13   -184.  350.  388.  353. <NULL>    <NULL>        0.993
## 8 stepwise   3.01e15   -211.  426.  428.  427. <cpl [0]> <cpl [0]>    NA
## 9 search     3.01e15   -211.  426.  428.  427. <cpl [0]> <cpl [0]>    NA
## # ... with 8 more variables: adj_r_squared <dbl>, statistic <dbl>,
## #   p_value <dbl>, df <int>, CV <dbl>, deviance <dbl>, df.residual <int>,
## #   rank <int>
```

```
## Melb and vic pop seem to be the best models - lets double check by looking at just data up
until 2019
```

```
models <- forecast_ts %>%
  filter(year <= 2019) %>%
  model(
    stepwise = ARIMA(touch_ons),
    search = ARIMA(touch_ons, stepwise = FALSE, approximation = FALSE),
    everything = TSLM(touch_ons ~ erp_victoria + erp_greater_melbourne + city_of_melbourne_jo
bs + melbourne_cbd_jobs + wfh_slow_growth),
    vic_com = TSLM(touch_ons ~ erp_victoria + city_of_melbourne_jobs + wfh_slow_growth),
    melb_cbd = TSLM(touch_ons ~ erp_greater_melbourne + melbourne_cbd_jobs + wfh_slow_growt
h),
    vic_pop = TSLM(touch_ons ~ erp_victoria + wfh_slow_growth),
    melb_pop = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth),
    com_job = TSLM(touch_ons ~ city_of_melbourne_jobs + wfh_slow_growth),
    cbd_job = TSLM(touch_ons ~ melbourne_cbd_jobs + wfh_slow_growth)
  )


models %>%
  report() %>%
  arrange(AICc)
```

```
## Warning in report.mdl_df(.): 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: 9 x 17
##   .model      sigma2 log_lik   AIC  AICc   BIC ar_roots  ma_roots  r_squared
##   <chr>        <dbl>   <dbl> <dbl> <dbl> <dbl> <list>    <list>        <dbl>
## 1 everything 8.22e12   -92.4  180.  95.7  178. <NULL>    <NULL>        0.987
## 2 melb_pop   2.11e13   -99.3  188. 200.   187. <NULL>    <NULL>        0.868
## 3 vic_pop    2.29e13   -99.6  188. 200.   188. <NULL>    <NULL>        0.857
## 4 com_job    7.39e13  -103.   195. 207.   195. <NULL>    <NULL>        0.538
## 5 cbd_job    8.34e13  -103.   196. 208.   195. <NULL>    <NULL>        0.478
## 6 vic_com    4.25e12   -93.7  178. 218.   177. <NULL>    <NULL>        0.980
## 7 stepwise   1.28e14  -105.   215. 219.   214. <cpl [0]> <cpl [0]>    NA
## 8 search     1.28e14  -105.   215. 219.   214. <cpl [0]> <cpl [0]>    NA
## 9 melb_cbd   8.97e12   -95.9  183. 223.   182. <NULL>    <NULL>        0.958
## # ... with 8 more variables: adj_r_squared <dbl>, statistic <dbl>,
## #   p_value <dbl>, df <int>, CV <dbl>, deviance <dbl>, df.residual <int>,
## #   rank <int>
```

```
## Bit odd that the model with everything did the best here but melb and vic pop are still be
st of the rest, lets try one more with trend and seaonal components
```

```
models <- forecast_ts %>%
  model(
    vic_pop = TSLM(touch_ons ~ erp_victoria + wfh_slow_growth),
    melb_pop = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth),
    vic_pop_w_trend = TSLM(touch_ons ~ erp_victoria + wfh_slow_growth + trend()),
    melb_pop_w_trend = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth + trend()),
    vic_pop_w_season = TSLM(touch_ons ~ erp_victoria + wfh_slow_growth + season()),
    melb_pop_w_season = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth + season()),
    vic_pop_w_both = TSLM(touch_ons ~ erp_victoria + wfh_slow_growth+ trend() + season()),
    melb_pop_w_both = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth + trend() + se
ason())

  )
```

```
## Warning: 1 error encountered for vic_pop_w_season
## [1] contrasts can be applied only to factors with 2 or more levels
```

```
## Warning: 1 error encountered for melb_pop_w_season
## [1] contrasts can be applied only to factors with 2 or more levels
```

```
## Warning: 1 error encountered for vic_pop_w_both
## [1] contrasts can be applied only to factors with 2 or more levels
```

```
## Warning: 1 error encountered for melb_pop_w_both
## [1] contrasts can be applied only to factors with 2 or more levels
```

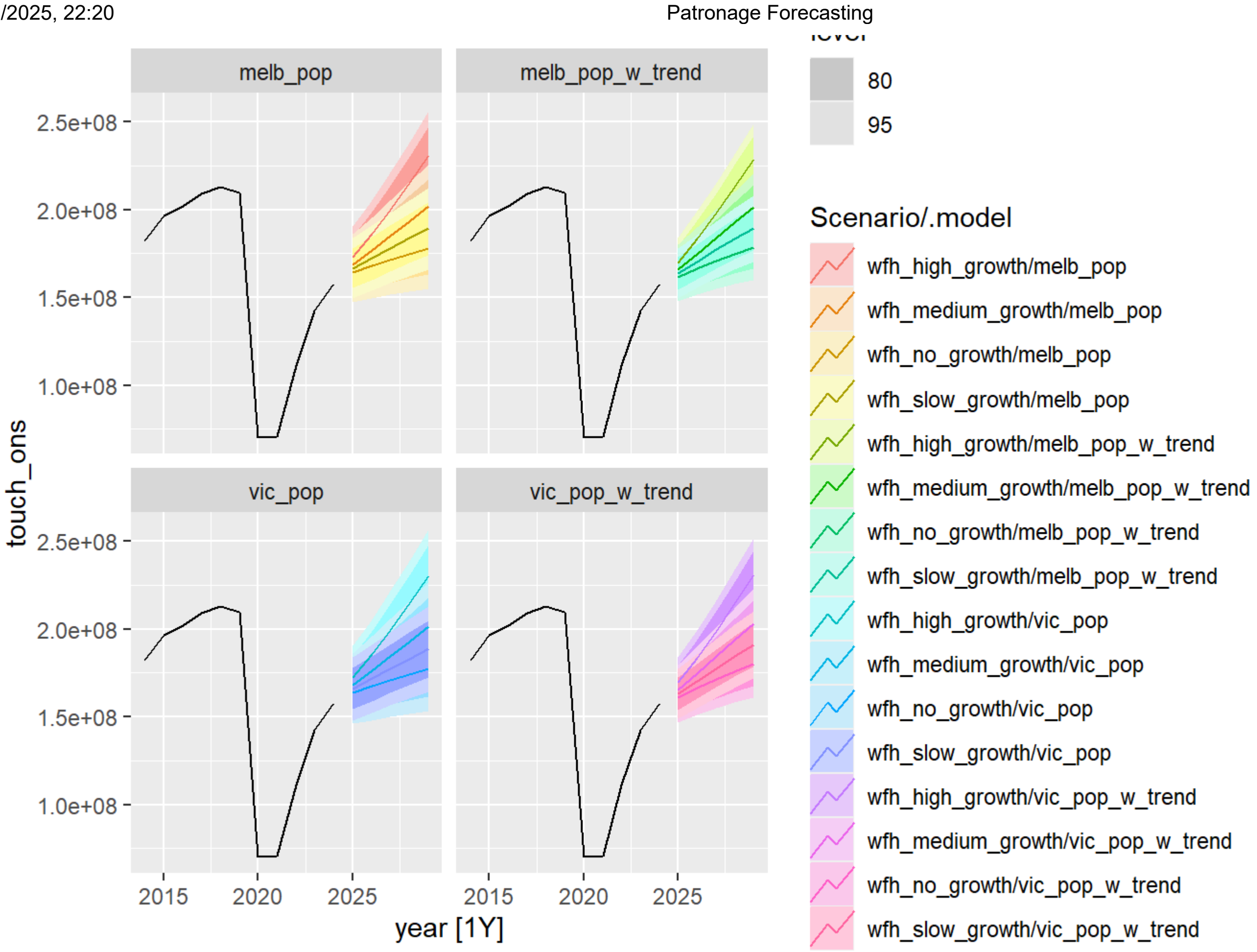
```
models %>%
  report() %>%
  arrange(AICc)
```

```
## Warning in report.mdl_df(.): 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.
```

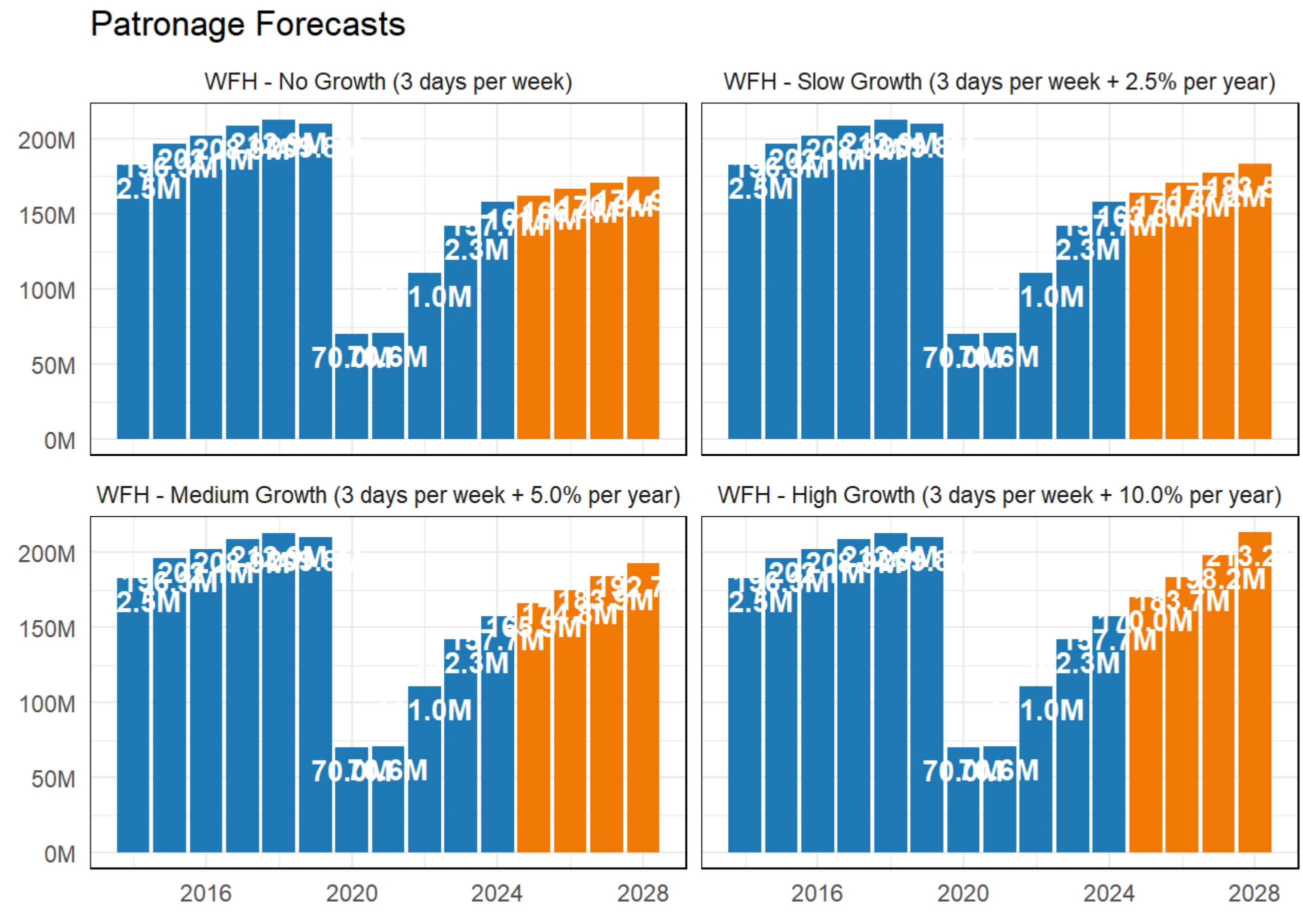
```
## # A tibble: 4 x 15
##   .model   r_squared adj_r_squared  sigma2 statistic p_value    df log_lik   AIC
##   <chr>        <dbl>         <dbl>   <dbl>     <dbl>   <dbl> <int>   <dbl> <dbl>
## 1 melb_pop     0.987         0.984 4.95e13      300. 3.01e-8     3   -187.  351.
## 2 vic_pop      0.986         0.982 5.40e13      275. 4.25e-8     3   -188.  352.
## 3 melb_po~     0.992         0.989 3.23e13      308. 8.52e-8     4   -184.  347.
## 4 vic_pop~     0.992         0.989 3.33e13      298. 9.55e-8     4   -184.  348.
## # ... with 6 more variables: AICc <dbl>, BIC <dbl>, CV <dbl>, deviance <dbl>,
## #   df.residual <int>, rank <int>
```

```
## No real change - existing predictors likely already covering trend changes but can still i
nclude to compare results
```

```
# Model creation

candidate_models <- forecast_ts %>%
  model(
    vic_pop = TSLM(touch_ons ~ erp_victoria + wfh_slow_growth),
    melb_pop = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth),
    vic_pop_w_trend = TSLM(touch_ons ~ erp_victoria + wfh_slow_growth + trend()),
    melb_pop_w_trend = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth + trend())
    )



future_scenarios <- scenarios(

  wfh_no_growth = new_data(forecast_ts, 5) %>%
    mutate(erp_victoria = filter(population_combined, year(date) >= 2025)$erp_victoria,
    erp_greater_melbourne = filter(population_combined, year(date) >= 2025)$erp_greater_melbo
urne,
    wfh_slow_growth = 3),

  wfh_slow_growth = new_data(forecast_ts, 5) %>%
    mutate(erp_victoria = filter(population_combined, year(date) >= 2025)$erp_victoria,
    erp_greater_melbourne = filter(population_combined, year(date) >= 2025)$erp_greater_melbo
urne,
    wfh_slow_growth = filter(wfh_data, year(date) >= 2025)$wfh_slow_growth),

  wfh_medium_growth = new_data(forecast_ts, 5) %>%
    mutate(erp_victoria = filter(population_combined, year(date) >= 2025)$erp_victoria,
    erp_greater_melbourne = filter(population_combined, year(date) >= 2025)$erp_greater_melbo
urne,
    wfh_slow_growth = filter(wfh_data, year(date) >= 2025)$wfh_medium_growth),

  wfh_high_growth = new_data(forecast_ts, 5) %>%
    mutate(erp_victoria = filter(population_combined, year(date) >= 2025)$erp_victoria,
    erp_greater_melbourne = filter(population_combined, year(date) >= 2025)$erp_greater_melbo
urne,
    wfh_slow_growth = filter(wfh_data, year(date) >= 2025)$wfh_high_growth),

  names_to = "Scenario"

)


scenario_forecast <- forecast(candidate_models,
                              new_data = future_scenarios)



forecast_ts %>%
  autoplot(touch_ons) +
  autolayer(scenario_forecast) +
  facet_wrap(~.model)
```


level
80
95
Scenario/.model
wfh_high_growth/melb_pop
wfh_medium_growth/melb_pop
wfh_no_growth/melb_pop
wfh_slow_growth/melb_pop
wfh_high_growth/melb_pop_w_trend
wfh_medium_growth/melb_pop_w_trend
wfh_no_growth/melb_pop_w_trend
wfh_slow_growth/melb_pop_w_trend
wfh_high_growth/vic_pop
wfh_medium_growth/vic_pop
wfh_no_growth/vic_pop
wfh_slow_growth/vic_pop
wfh_high_growth/vic_pop_w_trend
wfh_medium_growth/vic_pop_w_trend
wfh_no_growth/vic_pop_w_trend
wfh_slow_growth/vic_pop_w_trend
melb_pop
melb_pop_w_trend
vic_pop
vic_pop_w_trend
touch_ons
2.5e+08
2.0e+08
1.5e+08
1.0e+08
2015
2020
2025
year [1Y]

```
histoic_data <- touch_ons_for_forecast %>%
  mutate(year = year(date)) %>%
  select(year, touch_ons)

histoic_data_for_plot <- histoic_data %>%
  bind_rows(replicate(3, histoic_data, simplify = FALSE)) %>%
  mutate(Scenario = c(rep("wfh_no_growth", 11),
                      rep("wfh_slow_growth", 11),
                      rep("wfh_medium_growth", 11),
                      rep("wfh_high_growth", 11)))

melb_pop_w_trend_forecasts <- scenario_forecast %>%
  filter(.model == "melb_pop_w_trend") %>%
  as_tibble() %>%
  select(Scenario, year, touch_ons = .mean)



plot_data <- histoic_data_for_plot %>%
  bind_rows(melb_pop_w_trend_forecasts) %>%
  mutate(scenario = case_when(
    Scenario == "wfh_no_growth" ~ "WFH - No Growth (3 days per week)",
    Scenario == "wfh_slow_growth" ~ paste("WFH - Slow Growth (3 days per week +", percent(wfh
_growth_rate_slow, accuracy = .1), "per year)"),
    Scenario == "wfh_medium_growth" ~ paste("WFH - Medium Growth (3 days per week +", percent
(wfh_growth_rate_med, accuracy = .1), "per year)"),
    Scenario == "wfh_high_growth" ~ paste("WFH - High Growth (3 days per week +", percent(wfh
_growth_rate_fast, accuracy = .1), "per year)"),
  )) %>%
  mutate(scenario = factor(scenario,
                           levels = c(
                             "WFH - No Growth (3 days per week)",
                             paste("WFH - Slow Growth (3 days per week +", percent(wfh_growth
_rate_slow, accuracy = .1), "per year)"),
                             paste("WFH - Medium Growth (3 days per week +", percent(wfh_grow
th_rate_med, accuracy = .1), "per year)"),
                             paste("WFH - High Growth (3 days per week +", percent(wfh_growth
_rate_fast, accuracy = .1), "per year)")
                             )
                           )
         ) %>%
  mutate(period = if_else(year <= 2024, "Historic", "Forecast")) %>%
  filter(year != 2029)




plot <- plot_data %>%
  ggplot(aes(x = year, y = touch_ons, fill = period)) +
  geom_col() +
  geom_text(aes(label = comma(touch_ons, suffix = "M", scale = 0.000001, accuracy = .1)), vju
st = 1.5, colour = "white", fontface = "bold") +
  facet_wrap(~scenario) +
  scale_fill_manual(values = c("Historic" = "#227BB4",
                               "Forecast" = "#F07B05")) +
  theme_minimal() +
  theme(legend.position = "none",
        panel.border = element_rect(fill = NA)) +
  labs(x = "", y = "", title = "Patronage Forecasts") +
  scale_y_continuous(labels = comma_format(suffix = "M", scale = 0.000001))


plot
```



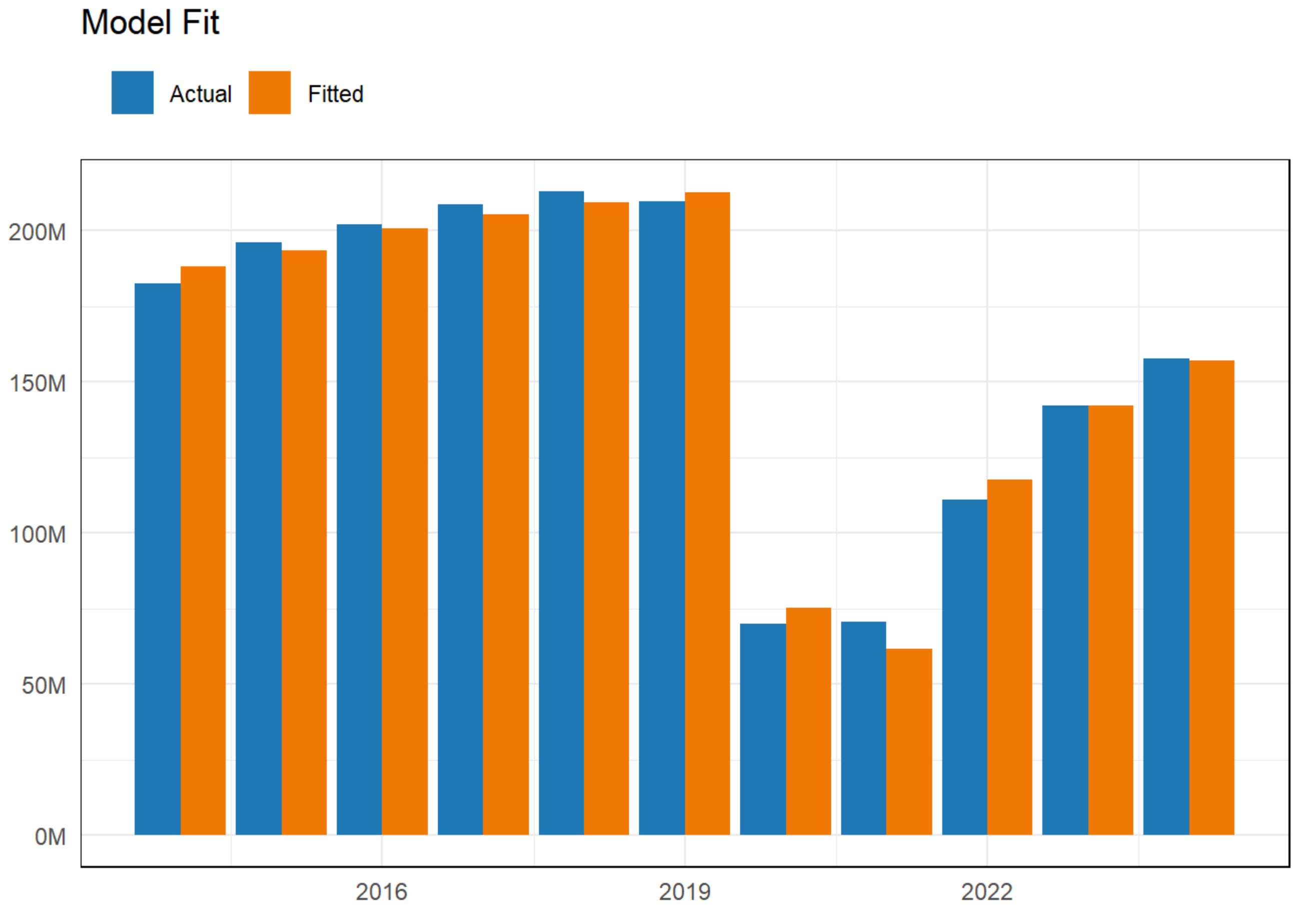
```
plot_data %>%
  filter(year >= 2024) %>%
  select(year, scenario, touch_ons) %>%
  group_by(scenario) %>%
  mutate(touch_ons_2024 = first(touch_ons),
         growth_from_2024 = (touch_ons - touch_ons_2024) / touch_ons,
         annual_growth = (touch_ons - lag(touch_ons)) / touch_ons) %>%
  ungroup() %>%
  select(-touch_ons_2024) %>%
  write_csv("forecast_outcomes.csv")
```

```
options(scipen = 999)

model_selected <- candidate_models %>%
  select(melb_pop_w_trend)

report(model_selected)
```

```
## Series: touch_ons
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6659764 -4071821   681798  3045740  8877937
##
## Coefficients:
##                           Estimate    Std. Error t value     Pr(>|t|)
## (Intercept)           -372177576.04  124209325.76  -2.996       0.0200 *
## erp_greater_melbourne         95.67         28.14   3.400       0.0114 *
## wfh_slow_growth         27510856.04    1225875.84  22.442 0.0000000883 ***
## trend()                 -5259174.92    2289174.61  -2.297       0.0552 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5680000 on 7 degrees of freedom
## Multiple R-squared: 0.9925,  Adjusted R-squared: 0.9893
## F-statistic: 308.4 on 3 and 7 DF, p-value: 0.000000085209
```

```
# For every 1 person increase in the ERP of Greater Melbourne, touch_ons increase by 95.67
# A one day increase in wfh increases touch-ons by ~27.5 million per year
# Independent of the above, weak trend showing touch-ons declining by ~5.25 mliion per year
```

```
augment(model_selected) %>%
  select(year, touch_ons, .fitted) %>%
  pivot_longer(touch_ons:.fitted) %>%
  mutate(name = if_else(name == "touch_ons", "Actual", "Fitted")) %>%
  ggplot(aes(x = year, y = value, fill = name)) +
  geom_col(position = "dodge") +
  scale_fill_manual(values = c("Actual" = "#227BB4",
                               "Fitted" = "#F07B05")) +
  theme_minimal() +
  theme(legend.position = "top",
        panel.border = element_rect(fill = NA),
        legend.justification = "left") +
  labs(x = "", y = "", title = "Model Fit", fill = "") +
  scale_y_continuous(labels = comma_format(suffix = "M", scale = 0.000001))
```



```
# Last try with arima erros

models_final <- forecast_ts %>%
  model(
    melb_pop_w_trend = TSLM(touch_ons ~ erp_greater_melbourne + wfh_slow_growth + trend()),
    melb_pop_w_arima = ARIMA(touch_ons ~ erp_greater_melbourne + wfh_slow_growth),
    melb_pop_w_arima_and_trend = ARIMA(touch_ons ~ erp_greater_melbourne + wfh_slow_growth +
trend()))


models_final %>%
  report() %>%
  arrange(AICc)
```

```
## Warning in report.mdl_df(.): 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: 3 x 17
##   .model  r_squared adj_r_squared  sigma2 statistic  p_value    df log_lik   AIC
##   <chr>       <dbl>         <dbl>   <dbl>     <dbl>    <dbl> <int>   <dbl> <dbl>
## 1 melb_p~     0.992         0.989 3.23e13      308.  8.52e-8     4   -184.  347.
## 2 melb_p~    NA            NA     6.38e13       NA  NA          NA   -189.  385.
## 3 melb_p~    NA            NA     6.44e13       NA  NA          NA   -189.  385.
## # ... with 8 more variables: AICc <dbl>, BIC <dbl>, CV <dbl>, deviance <dbl>,
## #   df.residual <int>, rank <int>, ar_roots <list>, ma_roots <list>
```

```
# Regular with trend seems the best
```