ADS 506 - Time Series

Fall 2023 - Week 6 OH

Dave Hurst

University of San Diego
SHILEY-MARCOS SCHOOL OF ENGINEERING

Start Recording!

University of San Diego
SHILEY-MARCOS SCHOOL OF ENGINEERING

Agenda

- Assignment 6.1 Hints
- Tidyverts Framework Review
- Final Project & Peer Notes
- Chapter 9 - Neural Networks Review (Erin)

Assignment 6.1

- 9.2 asks for a forecast of 2 months, while
  9.3 is asking for 12 months
- Express your forecasts in numbers and plots
- 5 Pt Bonus
  IF you use the updated template
  AND you use only the provided libraries
  (i.e `fable::NNETAR` rather than `forecast::nnetar()`)

fable::NNETAR() Example

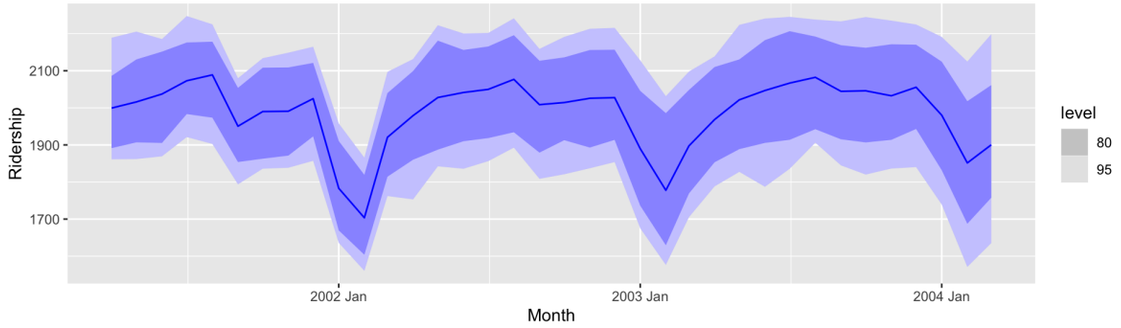
```
library(tidyverse)
library(fpp3)
library(readxl)

ridership <- read_excel(here::here("data", "Amtrak data.xls"),
    col_types = c("date", "numeric"))|>
  mutate(Month = yearmonth(Month)) |>
  as_tsibble(index = Month)

ridership_trn <- ridership |>
  filter_index(. ~ "2001 Mar")

ridership_val <- ridership |>
    anti_join(ridership_trn, by = 'Month')

ridership_nn_fit <- ridership_trn |>
  model(
    nnetar = NNETAR(Ridership)
  )
```

.model	.type	RMSE	MAE
nnetar	Training	81.1524	65.25672
nnetar	Test	109.4281	87.69099

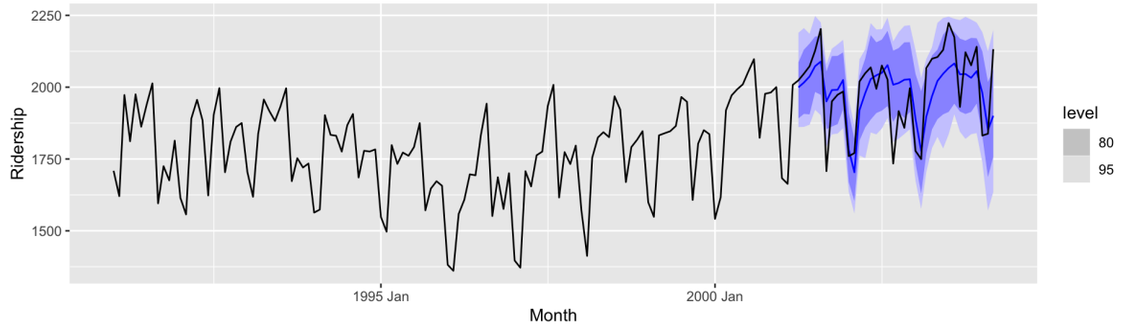
University of San Diego
SHILEY-MARCOS SCHOOL OF ENGINEERING

Digression: Forecast plots

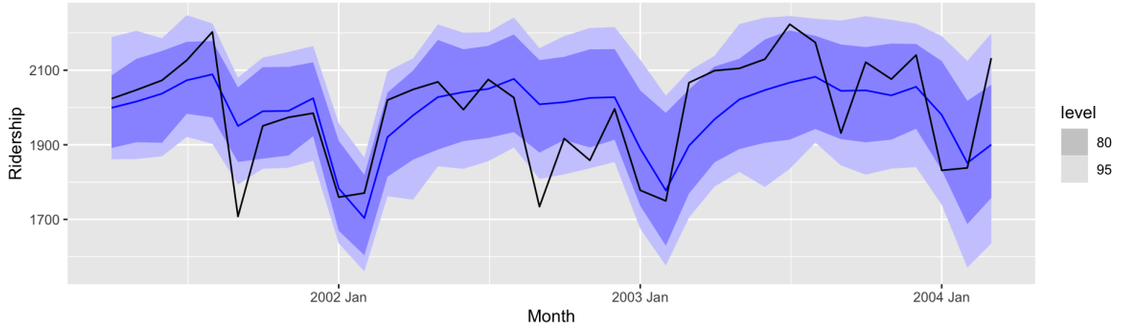
```
ridership_nn_fc |>
    autoplot()
```



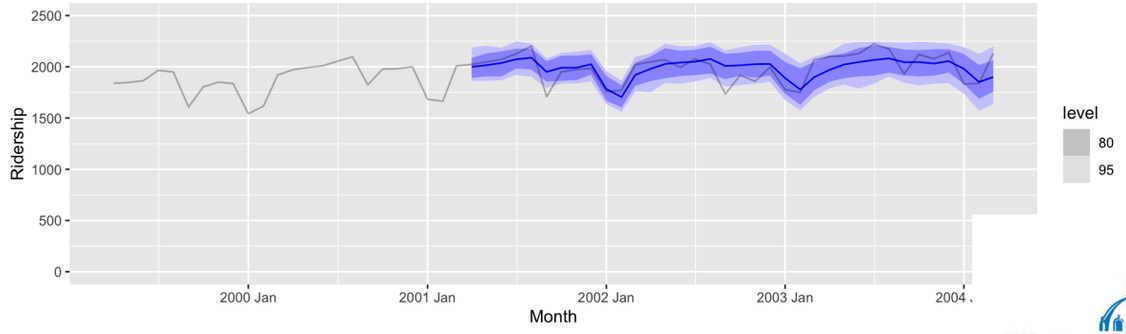

```
ridership_nn_fc |>
    autoplot(ridership)
```




```
ridership_nn_fc |>
    autoplot(ridership_val)
```

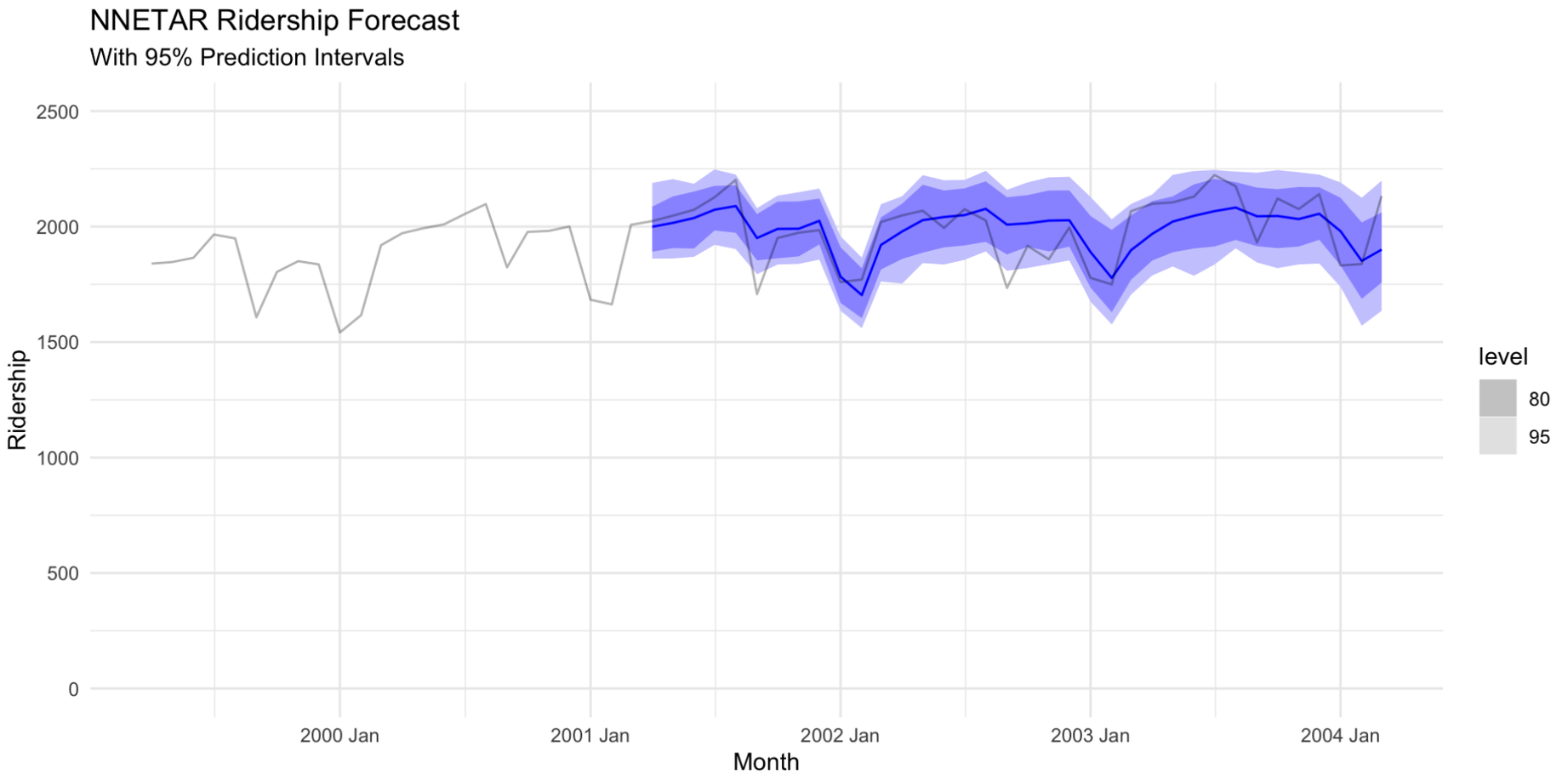



```
ridership_nn_fc |>
    autoplot() +
    autolayer(ridership |> tail(60), alpha
    ylim(0, 2500)
```



University of San Diego
SHILEY-MARCOS SCHOOL OF ENGINEERING

```r
ridership_nn_fc |>
    autoplot() +
    autolayer(ridership |> tail(60), alpha = 0.3, .vars = Ridership) +
    ylim(0, 2500) +
    theme_minimal() +
    labs(title = "NNETAR Ridership Forecast",
         subtitle = "With 95% Prediction Intervals",
         y = "Ridership",
         x = "Month")
```



tidyverts Framework Review

`tidyverts` strengths

- Ability to rapidly experiment with different models
- Built in capabilities for multiple time series
- Easier cross validation / rolling origin tools
- Latest implementations of forecasting methods

Weaknesses

- Lack of coverage
- incompatibility with other packages

tidyverts = fpp3

- `fpp3` is a framework for time series analysis and forecasting

fpp3 = tsibble + fable + feasts + fabletools

Objects:

- tsibble
- mable
- fable

Learn more at

- https://tidyverts.org/
- https://otexts.com/fpp3/

```
library(fpp3)
library(readxl)

ridership_nn_fit <- ridership_trn |>
  model(
    nnetar = NNETAR(Ridership)
  )

ridership_nn_fc <- ridership_nn_fit |>
  forecast(h = 36, times = 100)

accuracy(ridership_nn_fc, ridership_val)
```

University of San Diego
SHILEY-MARCOS SCHOOL OF ENGINEERING

PTFS Forecasting Methods

Recap of methods from PTFS

- mean
- naive.fixed <- naive(train.ts, h = fixed.nValid)
- holts.winter <- ets(train.ts, model = “MAA”)

  …
- ridership.lm <- tslm(train.ts ~ trend + I(trend^2))
- tslm.trend <- tslm(train.ts ~ trend)
- train.season <- tslm(train.ts ~ season)
- train.res.ar1 <- Arima(xx$residuals, order = c(1,0,0)) ## on RESIDUALS

  …
- ridership.nnetar <- nnetar(train.ts, repeats = 20, p = 11, P = 1, size = 7)

Equivalent methods in tidyverts

- mean = MEAN(Ridership)
- naive.fixed = NAIVE(Ridership)
- holts_winter = ETS(Ridership ~ error(‘M’) + trend(‘A’) + season(‘A’))
- tslm = TSLM(Ridership ~ trend() + I(trend()^2))
- tslm_trend = TSLM(Ridership ~ trend())
- tslm_season = TSLM(Ridership ~ season())
- ar1 = ARIMA(Ridership ~ 1 + pdq(1, 0, 0)). ## Directly vs on residuals
- auto_arima = ARIMA(Ridership)
- nnetar = NNETAR(Ridership)

tidyverts modeling

```
library(fpp3)
library(readxl)

ridership_nn_fit <- ridership_trn |>
  model(
    nnetar = NNETAR(Ridership)
  )
```

Equivalent methods in tidyverts

- mean = MEAN(Ridership)
- naive.fixed = NAIVE(Ridership)
- holts_winter = ETS(Ridership ~ error('M') + trend('A') + season('A'))
- tslm = TSLM(Ridership ~ trend() + I(trend()^2))
- tslm_trend = TSLM(Ridership ~ trend())
- tslm_season = TSLM(Ridership ~ season())
- ar1 = ARIMA(Ridership ~ 1 + pdq(1, 0, 0)). ## Directly vs on residuals
- auto_arima = ARIMA(Ridership)
- nnetar = NNETAR(Ridership)

tidyverts modeling

```r
library(fpp3)
library(readxl)

ridership_nn_fit <- ridership_trn |>
  model(
      mean = MEAN(Ridership),
      naive.fixed = NAIVE(Ridership),
      holts_winter = ETS(
          Ridership ~ error('M') + trend('A') + season('A')),
      tslm = TSLM(Ridership ~ trend() + I(trend()^2)),
      tslm_trend = TSLM(Ridership ~ trend()),
      tslm_season = TSLM(Ridership ~ season()),
      ar1 = ARIMA(Ridership ~ 1 + pdq(1, 0, 0)),
      auto_arima = ARIMA(Ridership),
      nnetar = NNETAR(Ridership)
  )
```

Equivalent methods in tidyverts

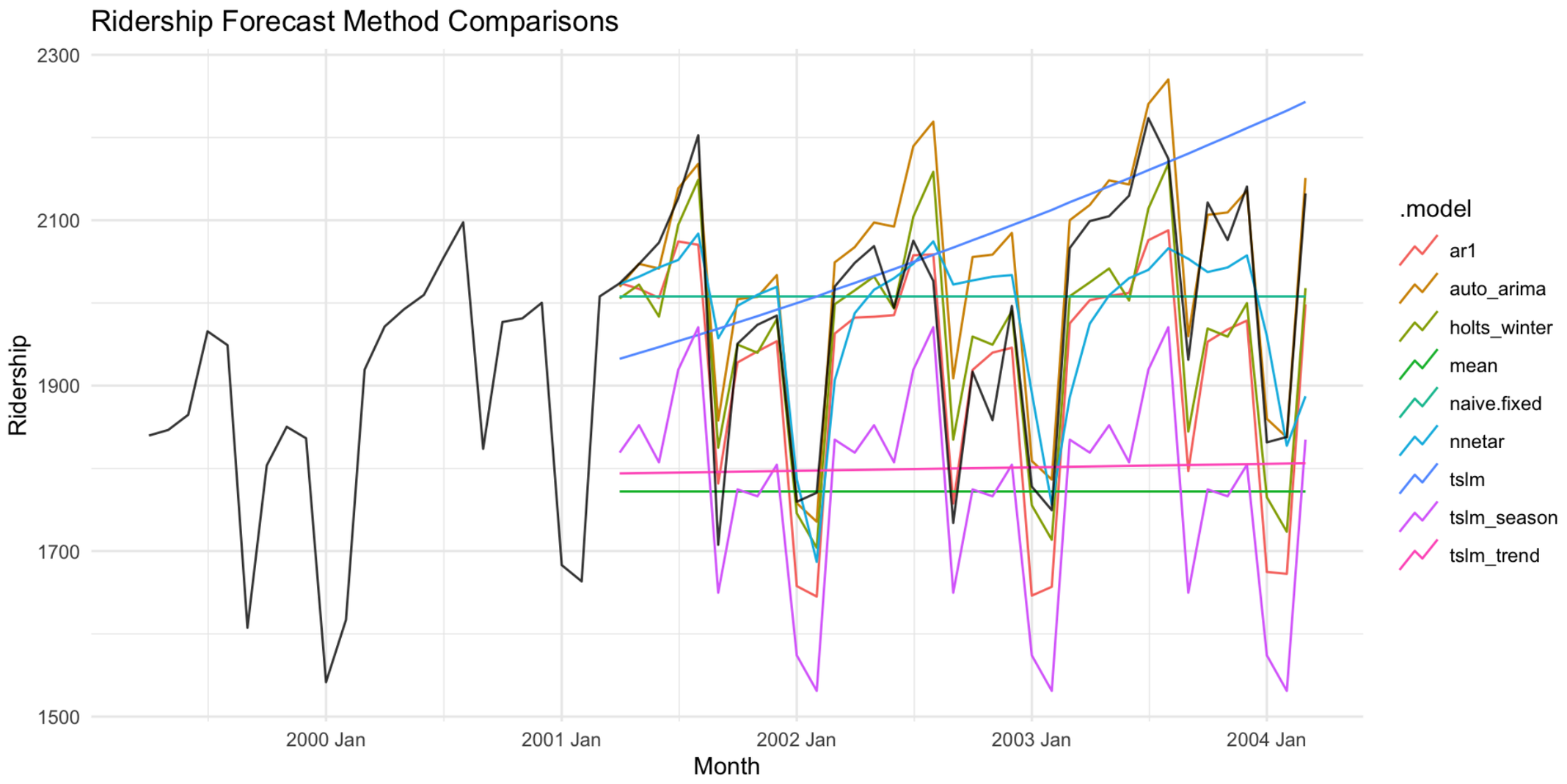
- mean = MEAN(Ridership)
- naive.fixed = NAIVE(Ridership)
- holts_winter = ETS(Ridership ~ error(‘M’) + trend(‘A’) + season(‘A’))
- tslm = TSLM(Ridership ~ trend() + I(trend()^2))
- tslm_trend = TSLM(Ridership ~ trend())
- tslm_season = TSLM(Ridership ~ season())
- ar1 = ARIMA(Ridership ~ 1 + pdq(1, 0, 0)). ## Directly vs on residuals
- auto_arima = ARIMA(Ridership)
- nnetar = NNETAR(Ridership)

tidyverts modeling

```r
library(fpp3)
library(readxl)

ridership_nn_fit <- ridership_trn |>
  model(
      mean = MEAN(Ridership),
      naive.fixed = NAIVE(Ridership),
      holts_winter = ETS(
          Ridership ~ error('M') + trend('A') + season('A')),
      tslm = TSLM(Ridership ~ trend() + I(trend()^2)),
      tslm_trend = TSLM(Ridership ~ trend()),
      tslm_season = TSLM(Ridership ~ season()),
      ar1 = ARIMA(Ridership ~ 1 + pdq(1, 0, 0)),
      auto_arima = ARIMA(Ridership),
      nnetar = NNETAR(Ridership)
  )
ridership_nn_fc <- ridership_nn_fit |>
  forecast(h = 36, times = 100)

accuracy(ridership_nn_fc, ridership_val) |>
    select(.model, .type, RMSE, MAPE, MAE) |>
    arrange(RMSE) |>
    gt::gt()
```

.model	.type	RMSE	MAPE	MAE
auto_arima	Test	76.50042	2.739223	53.04340
holts_winter	Test	76.65107	3.122042	62.16977
ar1	Test	96.59242	4.169418	82.84539
nnetar	Test	115.20197	4.634819	91.17639
naive.fixed	Test	142.75509	6.021396	115.92339
tslm	Test	179.84942	7.075730	133.73831
tslm_season	Test	229.65092	10.864618	217.92668
tslm_trend	Test	239.48626	10.147732	209.43710
mean	Test	262.74025	11.065043	228.79690

Code



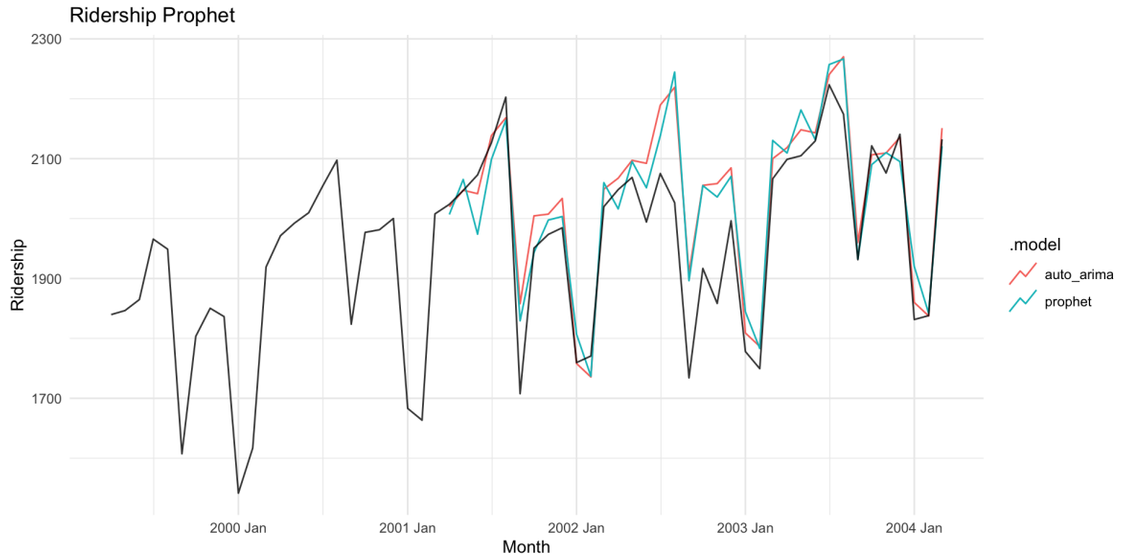
University of San Diego
SHILEY-MARCOS SCHOOL OF ENGINEERING

Experimenting with prophet

```r
library(fable.prophet)

prophet_fit <- ridership_trn |>
    model(
        auto_arima = ARIMA(Ridership),
        prophet = prophet(Ridership),
    )
prophet_fc <- prophet_fit |>
    forecast(h = 36, times = 100)

accuracy(prophet_fc, ridership_val) |>
    select(.model, .type, RMSE, MAPE, MAE)
    arrange(RMSE) |>
    gt::gt()
```

.model	.type	RMSE	MAPE	MAE
prophet	Test	76.35193	2.911116	56.60681
auto_arima	Test	76.50042	2.739223	53.04340

```r
prophet_fc |>
    autoplot(level = NULL) +
    autolayer(ridership |> tail(60), alpha
    theme_minimal() +
    labs(title = "Ridership Prophet",
         y = "Ridership",
         x = "Month")
```



Caveat: Prophet has several options that aren't explored here.

University of San Diego
SHILEY-MARCOS SCHOOL OF ENGINEERING

Final Project

- Presentation
  - DUE Saturday (Midnight)
  - 8-10 Minutes
  - All Students should present a portion of the project
  - Templates are fine, but avoid fluff
  - Audience is non-technical, but familiar with the data
  - Presentation should build technical credibility
- Report
  - Inline figures and graphs
  - Appendices for supporting figures and graphs

Chapter 9 - Review (Erin)

University of San Diego
SHILEY-MARCOS SCHOOL OF ENGINEERING