Tidy Tuesday 12/21

Emi

2022-12-21

```
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(ggplot2)
library(dplyr)
library(zoo)
```

```
##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

Weather Forecast

The data includes 16 months of forecasts and observations from 167 cities, as well as a separate data.frame of information about those cities and some other American cities.

Loading the dataset

```
tuesdata <- tidytuesdayR::tt_load('2022-12-20')
```

```
## --- Compiling #TidyTuesday Information for 2022-12-20 ----
```

```
## --- There are 3 files available ---

## --- Starting Download ---

##
##  Downloading file 1 of 3: ‘weather_forecasts.csv‘
##  Downloading file 2 of 3: ‘cities.csv‘
##  Downloading file 3 of 3: ‘outlook_meanings.csv‘

## --- Download complete ---
```

```
tuesdata <- tidytuesdayR::tt_load(2022, week = 51)
```

```
## --- Compiling #TidyTuesday Information for 2022-12-20 ----

## --- There are 3 files available ---

## --- Starting Download ---

##
##  Downloading file 1 of 3: ‘weather_forecasts.csv‘
##  Downloading file 2 of 3: ‘cities.csv‘
##  Downloading file 3 of 3: ‘outlook_meanings.csv‘

## --- Download complete ---
```

```
weather_forecasts <- tuesdata$weather_forecasts
cities <- tuesdata$cities
outlook_meanings <- tuesdata$outlook_meanings
```

EDA

```
glimpse(weather_forecasts)
```

```
## Rows: 651,968
## Columns: 10
## $ date                   <date> 2021-01-30, 2021-01-30, 2021-01-30, 2021-01-30,~
## $ city                   <chr> "ABILENE", "ABILENE", "ABILENE", "ABILENE", "ABI~
## $ state                  <chr> "TX", "TX", "TX", "TX", "TX", "TX", "TX", "TX", ~
## $ high_or_low            <chr> "high", "high", "high", "high", "low", "low", "l~
## $ forecast_hours_before <dbl> 48, 36, 24, 12, 48, 36, 24, 12, 48, 36, 24, 12, ~
## $ observed_temp          <dbl> 70, 70, 70, 70, 42, 42, 42, 42, 29, 29, 29, 29, ~
## $ forecast_temp          <dbl> NA, NA, NA, 70, NA, NA, 39, 38, NA, NA, NA, 30, ~
## $ observed_precip        <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, ~
## $ forecast_outlook       <chr> NA, NA, NA, "DUST", NA, NA, "DUST", "SUNNY", NA,~
## $ possible_error         <chr> "none", "none", "none", "none", "none", "none", ~
```

```r
table(weather_forecasts$date)[1]
```

```
## 2021-01-30
##       1336
```

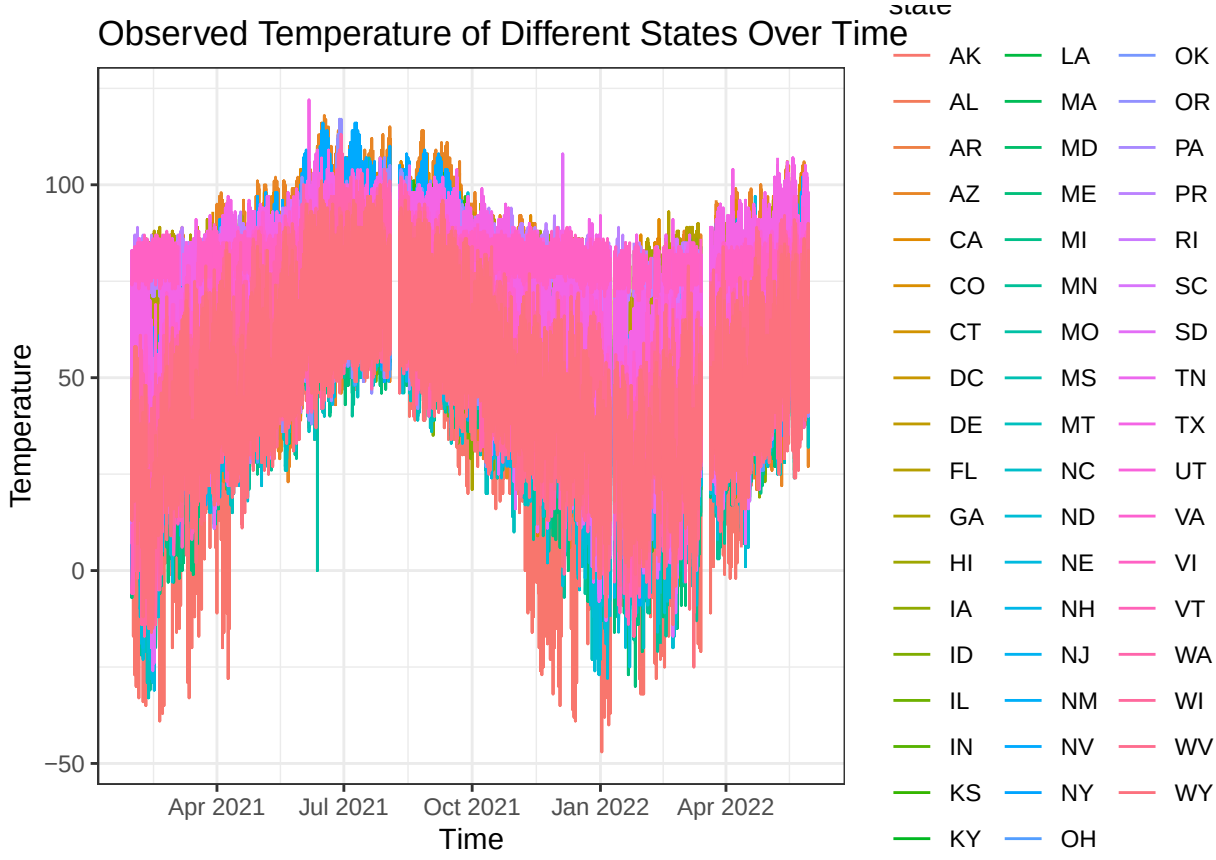
```r
tail(table(weather_forecasts$date), n = 1)
```

```
##
## 2022-06-01
##       1336
```

weather_forecasts.csv: Weather recorded from 1/30/21 to 6/1/22

```r
weather_forecasts %>%
  ggplot(aes(x = date, y = observed_temp, color = state)) +
    geom_line() +
    labs(title = "Observed Temperature of Different States Over Time", x = "Time", y = "Temperature") +
    theme_bw()
```

```
## Warning: Removed 2672 row(s) containing missing values (geom_path).
```

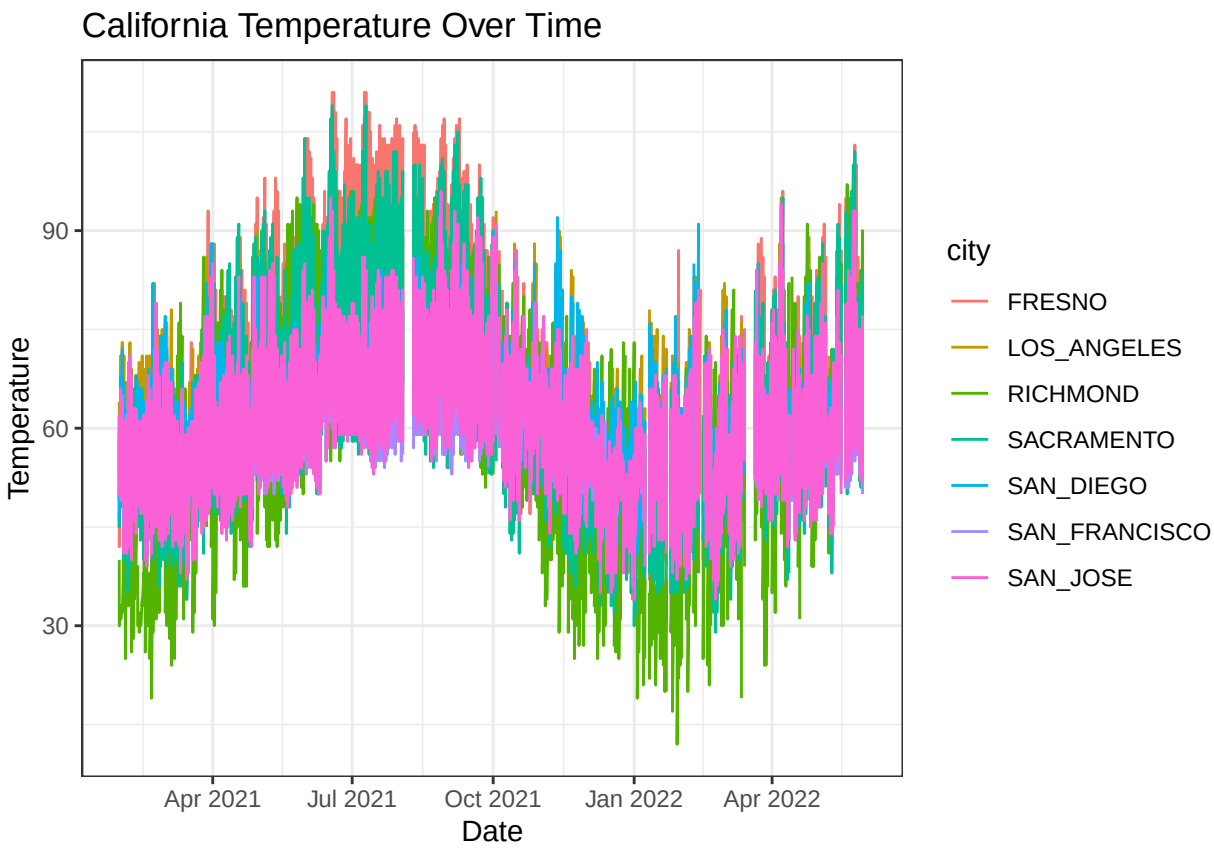



```r
# Get the percentage difference between predicted and observed temperature
error = (weather_forecasts$observed_temp - weather_forecasts$forecast_temp)^2
forecasts_df <- cbind(weather_forecasts, error)
head(forecasts_df)
```

```
##         date    city state high_or_low forecast_hours_before observed_temp
## 1 2021-01-30 ABILENE    TX        high                     48            70
## 2 2021-01-30 ABILENE    TX        high                     36            70
## 3 2021-01-30 ABILENE    TX        high                     24            70
## 4 2021-01-30 ABILENE    TX        high                     12            70
## 5 2021-01-30 ABILENE    TX         low                     48            42
## 6 2021-01-30 ABILENE    TX         low                     36            42
##   forecast_temp observed_precip forecast_outlook possible_error error
## 1            NA               0             <NA>           none    NA
## 2            NA               0             <NA>           none    NA
## 3            NA               0             <NA>           none    NA
## 4            70               0             DUST           none     0
## 5            NA               0             <NA>           none    NA
## 6            NA               0             <NA>           none    NA
```

```
forecasts_df[forecasts_df$state == "CA",] %>%
  ggplot(aes(x = date, y = observed_temp, color = city)) +
  geom_line() +
  labs(title = "California Temperature Over Time", x = "Date", y = "Temperature") +
  theme_bw()
```

```
## Warning: Removed 112 row(s) containing missing values (geom_path).
```



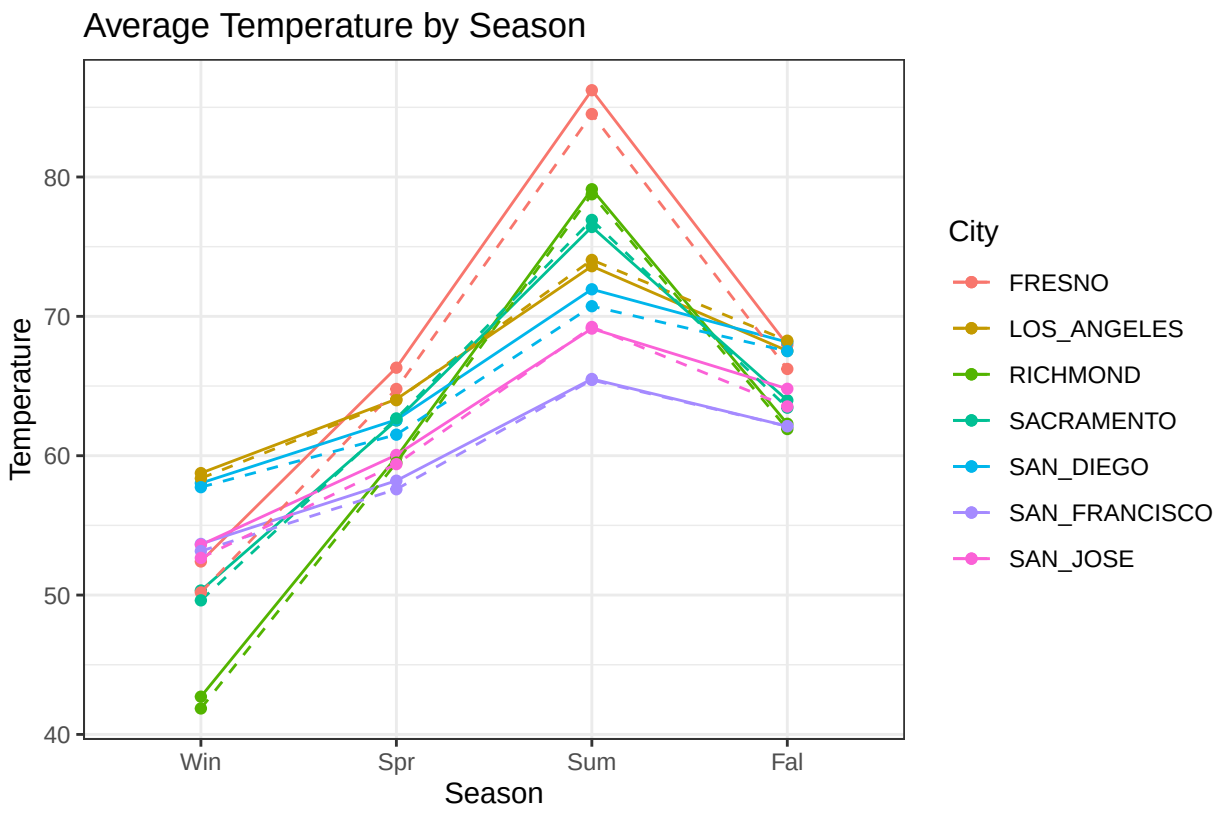

```
# Section the time into four seasons:
yq <- as.yearqtr(as.yearmon(forecasts_df$date, "%m/%d/%y") + 1/12)
```

```
forecasts_df$season <- factor(format(yq, "%q"), levels = 1:4,
                              labels = c("Win","Spr","Sum","Fal"))
```

```
# Get the average for every season and city
forecasts_df <- forecasts_df %>%
                    group_by(city, season) %>%
                    mutate(avg_season = mean(observed_temp, na.rm = TRUE))
```

```
forecasts_df %>%
  group_by(city, state, season) %>%
  summarize(avg_ob_sea = mean(observed_temp, na.rm = TRUE), avg_pred_sea = mean(forecast_temp, na.rm = T
  filter(state == "CA") %>%
  ggplot(aes(x = season, group = city, color = city)) +
    geom_line(aes(y = avg_ob_sea))+
    geom_point(aes(y = avg_ob_sea)) +
    geom_line(aes(y = avg_pred_sea), linetype = "dashed") +
    geom_point(aes(y = avg_pred_sea)) +
    scale_linetype_manual("Avg Temperature",values=c("Observed"=2,"Predicted"=1)) +
    theme_bw() +
    labs(title = "Average Temperature by Season", x = "Season", y = "Temperature", color = "City", capti
```

```
## `summarise()` has grouped output by 'city', 'state'. You can override using the
## `.groups` argument.
```



line represents predicted temperature and solid line represents observed temperature ""