# Adventures in Bayesian Structural Time Series

*Part 3: Analyzing SST Data*

Andrew Bates, Josh Gloyd, Tyler Tucker

# Outline

- SST Data

# Outline

- SST Data
- Use `bsts` for

# Outline

- SST Data
- Use `bsts` for
    - Fit

# Outline

- SST Data
- Use `bsts` for
    - Fit
        - local level

# Outline

- SST Data
- Use bsts for
  - Fit
    - local level
    - local linear trend model

# Outline

- SST Data
- Use `bsts` for
  - Fit
    - local level
    - local linear trend model
    - local trend with seasonality

# Outline

- SST Data
- Use `bsts` for
  - Fit
    - local level
    - local linear trend model
    - local trend with seasonality
  - Posterior distribution

# Outline

- SST Data
- Use `bsts` for
  - Fit
    - local level
    - local linear trend model
    - local trend with seasonality
  - Posterior distribution
  - Forecast

# Outline

- SST Data
- Use `bsts` for
    - Fit
        - local level
        - local linear trend model
        - local trend with seasonality
    - Posterior distribution
    - Forecast
    - Model Comparison

# SST Data

- Sea Surface Temperature near Gibraltar

# SST Data

- Sea Surface Temperature near Gibraltar
- Aggregated every 12 days

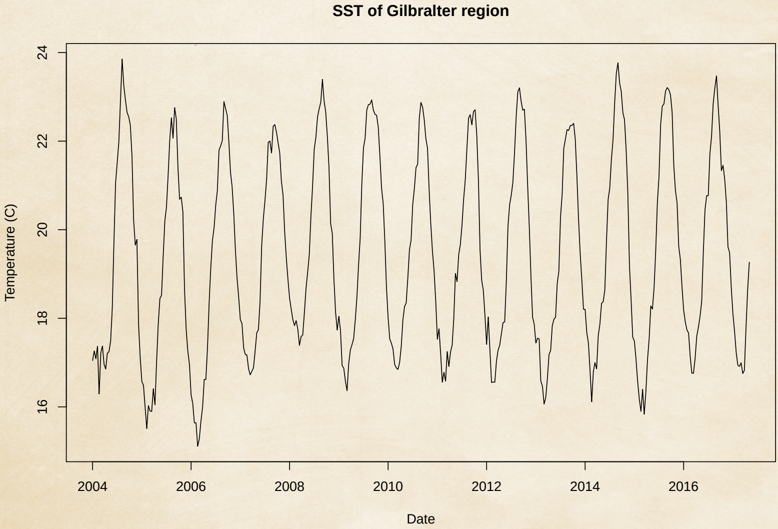
# SST Data

- Sea Surface Temperature near Gibraltar
- Aggregated every 12 days
- January 2004 to November 2017

# SST Data

- Sea Surface Temperature near Gibraltar
- Aggregated every 12 days
- January 2004 to November 2017
- Obtained from Argovis

# SST Data


SST of Gilbralter region
Temperature (C)
24
22
20
18
16
2004
2006
2008
2010
2012
2014
2016
Date

# Setup

```r
library(readr)
library(bsts)
# bsts also loads BoomSpikeSlab, Boom, MASS, zoo, xts


gilbralter <- read_csv("data/gilbraltersimple.csv")
gilt <- ts(gilbralter$tempMean, start=c(2004,1,13),
           end=c(2017, 11, 25), frequency=30)
plot(gilt, main='SST of Gilbralter region',
     xlab='Date',
     ylab='Temperature (C)')
```

# Model Fitting

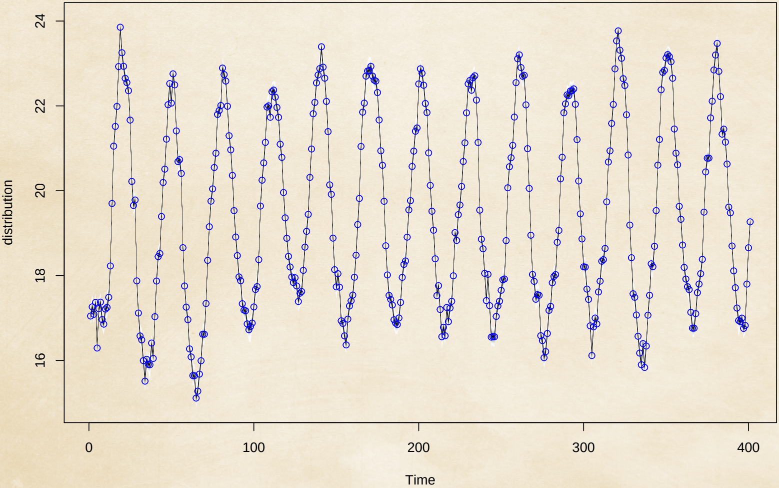
## Local Level Model

$$
\begin{aligned}
y_t &= \mu_t + \varepsilon_t & \varepsilon_t &\sim N(0, \sigma^2_\varepsilon) \\
\mu_{t+1} &= \mu_t + \xi_t & \xi_t &\sim N(0, \sigma^2_\xi)
\end{aligned}
$$

```
ll_ss <- list()
ll_ss <- AddLocalLevel(state.specification = ll_ss,
                       y = gilt)
ll_fit <- bsts(gilt, state.specification = ll_ss,
               niter = 1e3)
```

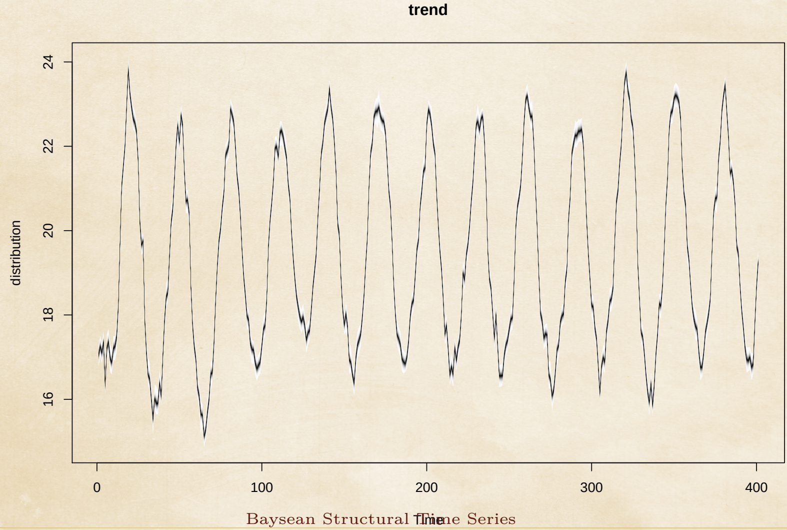
# Model Plotting

```
plot(ll_fit)
```



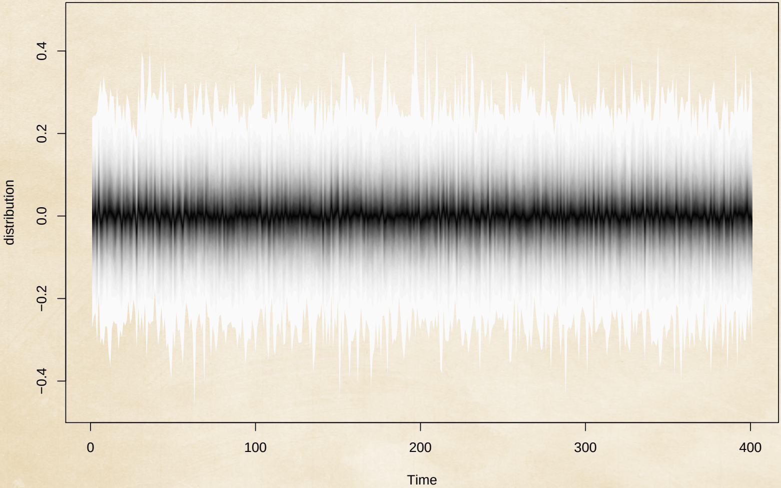
# Model Plotting

```r
plot(ll_fit, 'components')
```



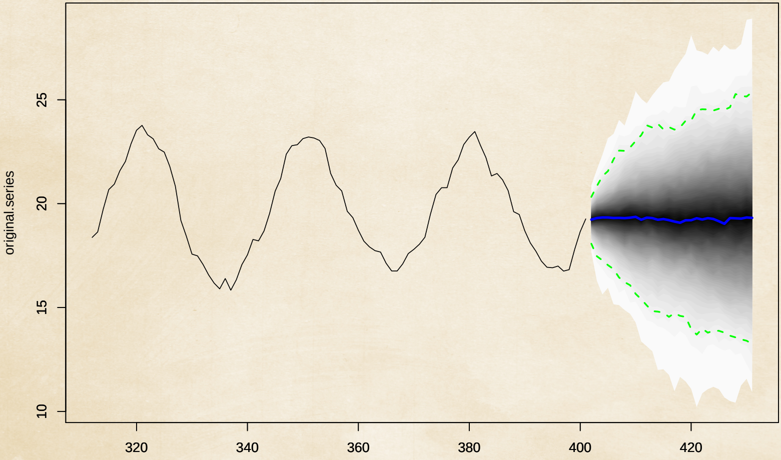
# Model Plotting

```
plot(ll_fit, 'residuals')
```



# Forecasting

```r
ll_pred <- predict(ll_fit, horizon = 30)
plot(ll_pred, plot.original = 90)
```


original.series
Time