



Adventures in Bayesian Structural Time Series

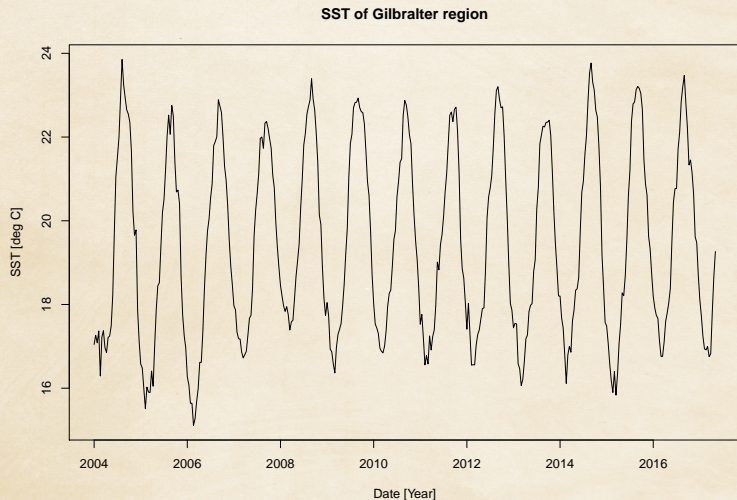
Part 3: Analyzing SST Data

Andrew Bates, Josh Gloyd, Tyler Tucker





- ⊠ SST data.
- ⊠ Model fitting of SST data, whose components include: local level, local linear trend, and seasonal trend.
- ⊠ Generating a posterior distribution of our model time series
- ⊠ Some heuristic comparisons between the different models.





- ⊠ SST data come from Argo floats
- ⊠ Aggregated every 12 days
- ⊠ January 2004 to November 2017
- ⊠ Obtained from www.Argovis.com
- ⊠ Learn more about Argovis by watching a tutorial at <https://www.youtube.com/watch?v=I1NJ0owuTHM&t=0s>



Setup

```
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 [Year]',
     ylab='SST [deg C]')
```



Local Level Model

$$y_t = \mu_t + \varepsilon_t \qquad \varepsilon_t \sim N(0, \sigma_\varepsilon^2)$$

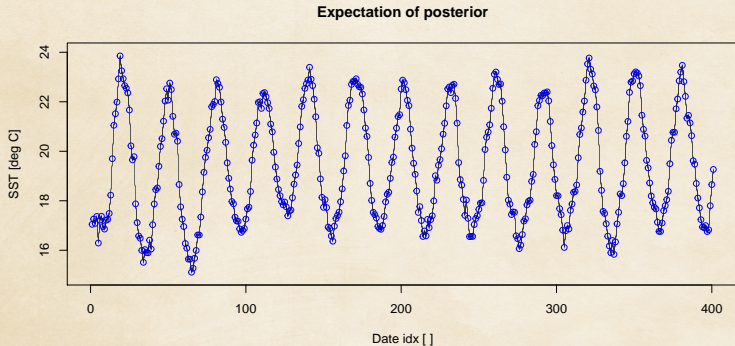
$$\mu_{t+1} = \mu_t + \xi_t \qquad \xi_t \sim N(0, \sigma_\xi^2)$$

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



Model Plotting

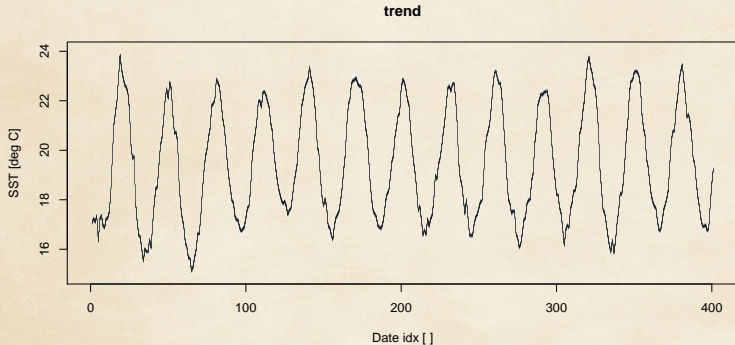
```
plot(ll_fit, main='Expectation of posterior',  
     xlab='Date idx [ ]',  
     ylab='SST [deg C]')
```





Model Plotting

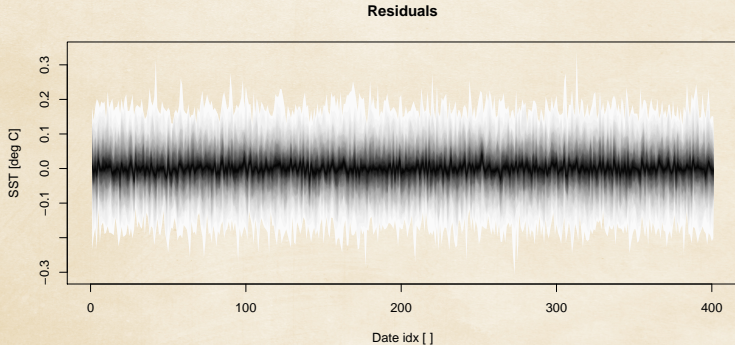
```
plot(ll_fit, 'components',  
     xlab='Date idx [ ]',  
     ylab='SST [deg C]')
```





Model Plotting

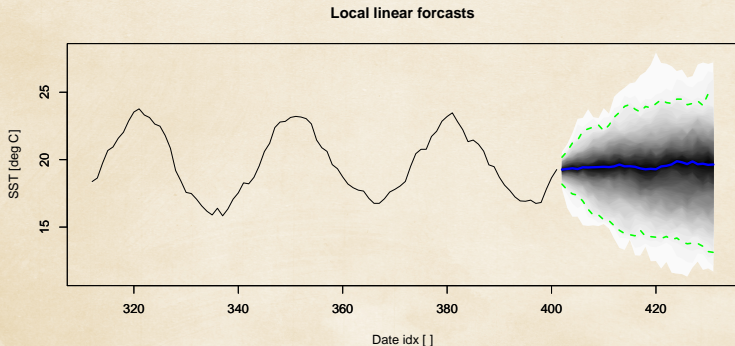
```
plot(ll_fit, 'residuals', main='Residuals',  
     xlab='Date idx [ ]',  
     ylab='SST [deg C]')
```





Forecasting

```
ll_pred <- predict(ll_fit, horizon = 30)
plot(ll_pred, plot.original = 90,
     main='Local linear forecasts',
     xlab='Date idx [ ]',
     ylab='SST [deg C]')
```





Local Linear Trend Model

$$y_t = \mu_t + \varepsilon_t \qquad \varepsilon_t \sim N(0, \sigma_\varepsilon^2)$$

$$\mu_{t+1} = \mu_t + \nu_t + \xi_t \qquad \xi_t \sim N(0, \sigma_\xi^2)$$

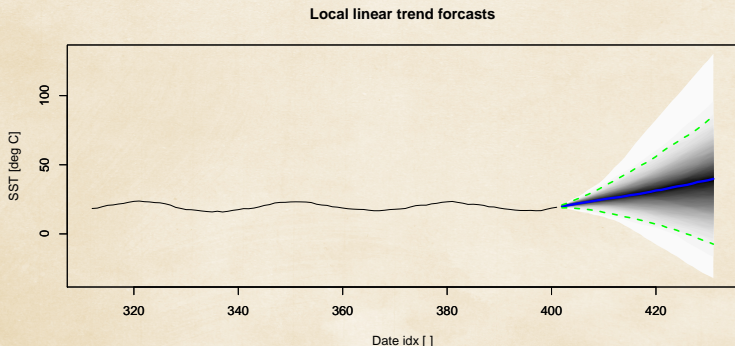
$$\nu_{t+1} = \nu_t + \zeta_t \qquad \zeta_t \sim N(0, \sigma_\zeta^2)$$

```
llt_ss <- list()
llt_ss <- AddLocalLinearTrend(
  state.specification = llt_ss, y = gilt)
llt_fit <- bsts(gilt, state.specification = llt_ss,
  niter = 1e3)
```



Forecasts

```
l1t_pred <- predict(l1t_fit, horizon = 30)
plot(l1t_pred, plot.original = 90,
     main='Local linear trend forecasts',
     xlab='Date idx [ ]',
     ylab='SST [deg C]')
```





Local Trend With Seasonality

$$y_t = \mu_t + \tau_t + \varepsilon_t$$

$$\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$$

$$\tau_t = - \sum_{s=1}^{S-1} \tau_{t-s} + \omega_t$$

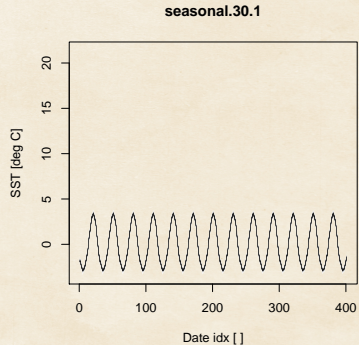
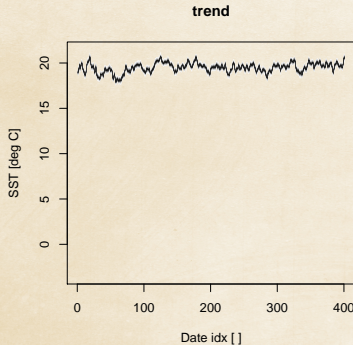
$$\omega_t \sim N(0, \sigma_\omega^2)$$

```
lts_ss <- list()
lts_ss <- AddLocalLinearTrend(lts_ss, y = gilt)
lts_ss <- AddSeasonal(lts_ss, gilt, nseasons = 30)
lts_fit <- bsts(gilt, state.specification = lts_ss,
               niter = 1e3)
```



Components

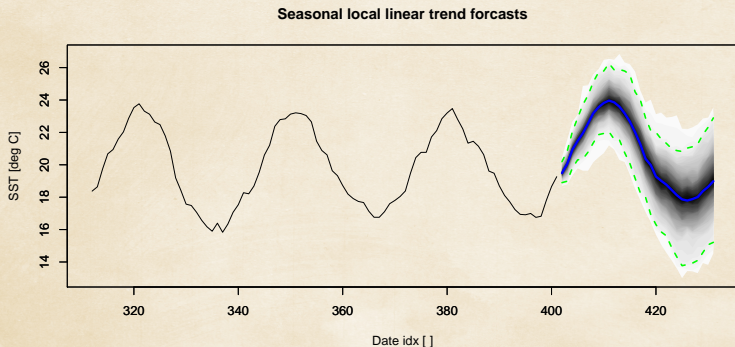
```
plot(lts_fit, 'components',  
     xlab='Date idx [ ]',  
     ylab='SST [deg C]')
```





Forecasts

```
lts_pred <- predict(lts_fit, horizon = 30)
plot(lts_pred, plot.original = 90,
     main='Seasonal local linear trend forecasts',
     xlab='Date idx [ ]',
     ylab='SST [deg C]')
```





Model Comparison

```
CompareBstsModels(lwd = 4, model.list = list(  
  level = ll_fit, trend = llt_fit, season = lts_fit),  
  colors = c("forestgreen", "firebrick", "blue4"),  
  xlab='Date idx [ ]')
```

