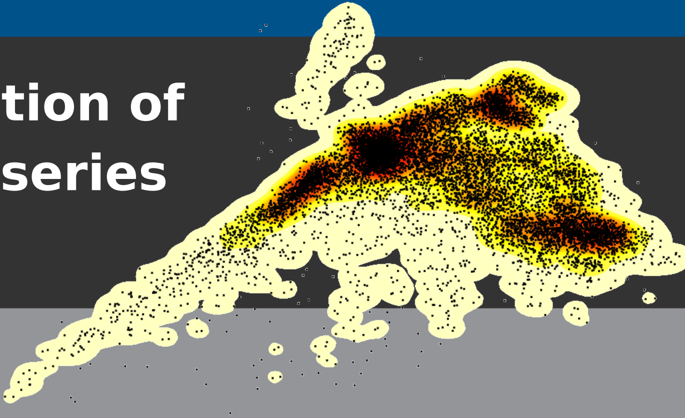
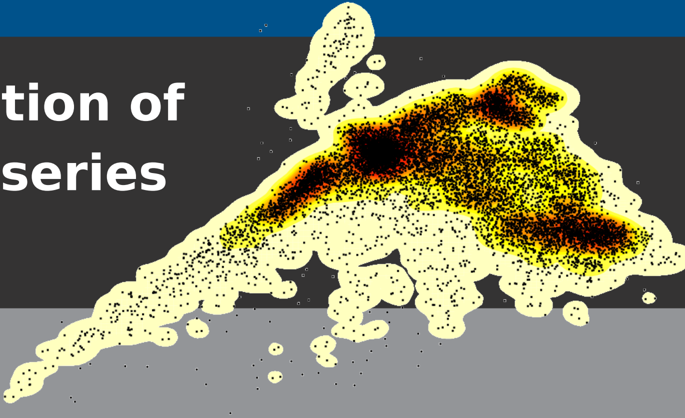
MONASH BUSINESS SCHOOL

Visualisation of big time series data



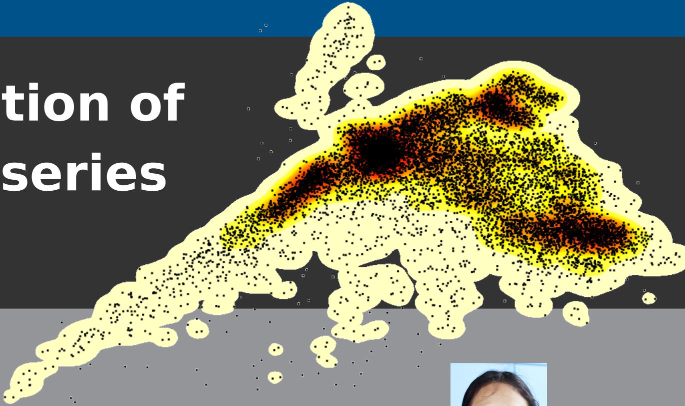
Rob J Hyndman

Visualisation of big time series data



Rob J Hyndman

with Earo Wang, Nikolay Laptev
Yanfei Kang, Kate Smith-Miles



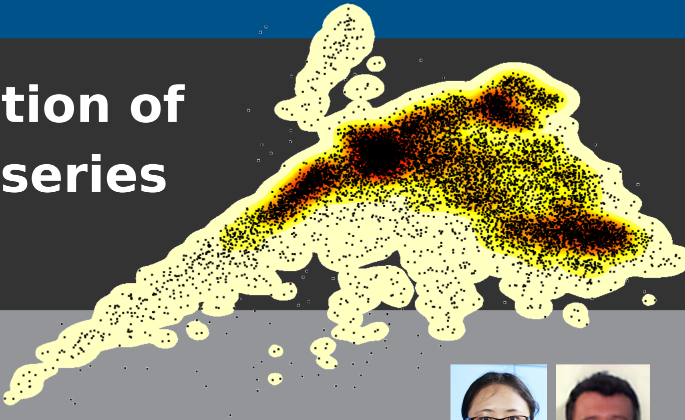
Visualisation of big time series data

Rob J Hyndman

with Earo Wang, Nikolay Laptev
Yanfei Kang, Kate Smith-Miles

MONASH BUSINESS SCHOOL

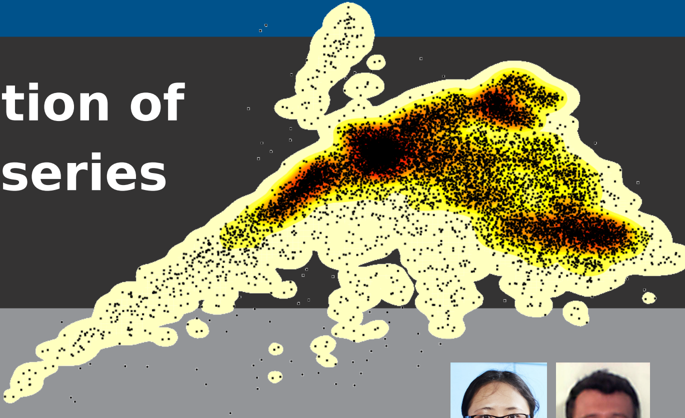
Visualisation of big time series data



Rob J Hyndman

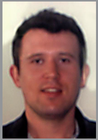
with Earo Wang, Nikolay Laptev
Yanfei Kang, Kate Smith-Miles

Visualisation of big time series data



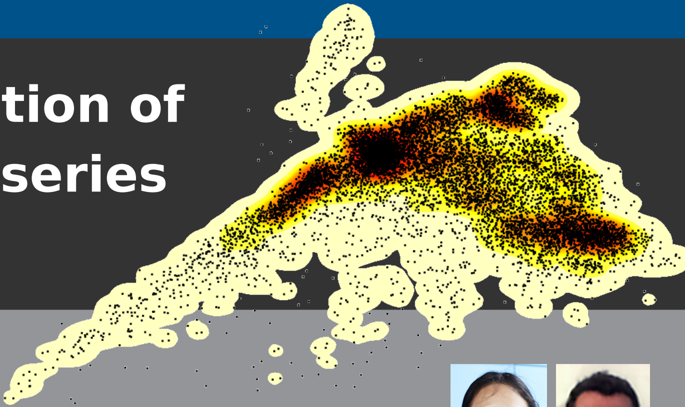
Rob J Hyndman

with Earo Wang, Nikolay Laptev
Yanfei Kang, Kate Smith-Miles



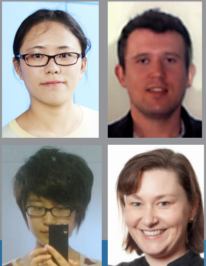
MONASH BUSINESS SCHOOL

Visualisation of big time series data

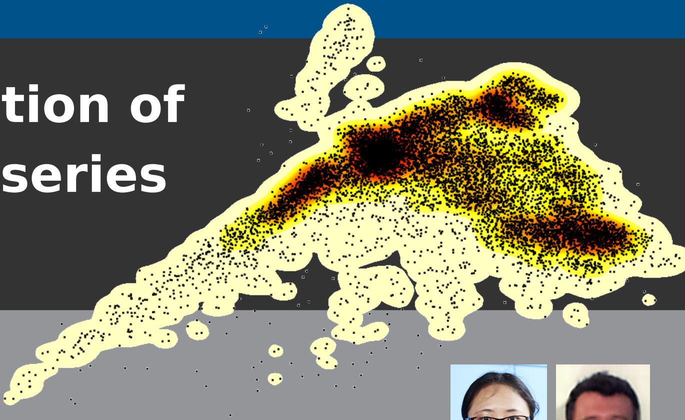


Rob J Hyndman

with Earo Wang, Nikolay Laptev
Yanfei Kang, Kate Smith-Miles

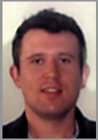


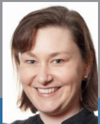
Visualisation of big time series data



Rob J Hyndman

with Earo Wang, Nikolay Laptev
Yanfei Kang, Kate Smith-Miles

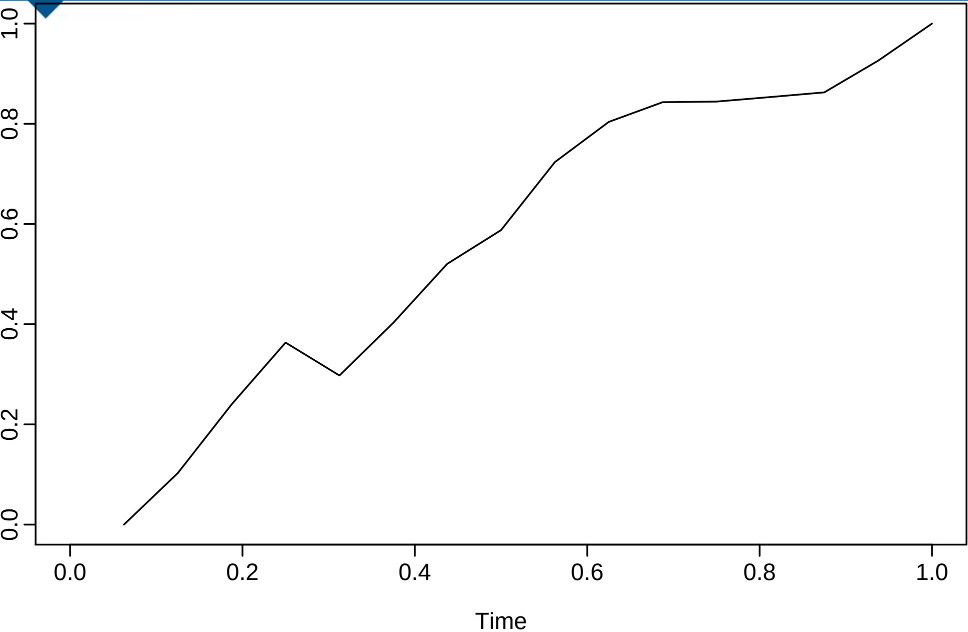





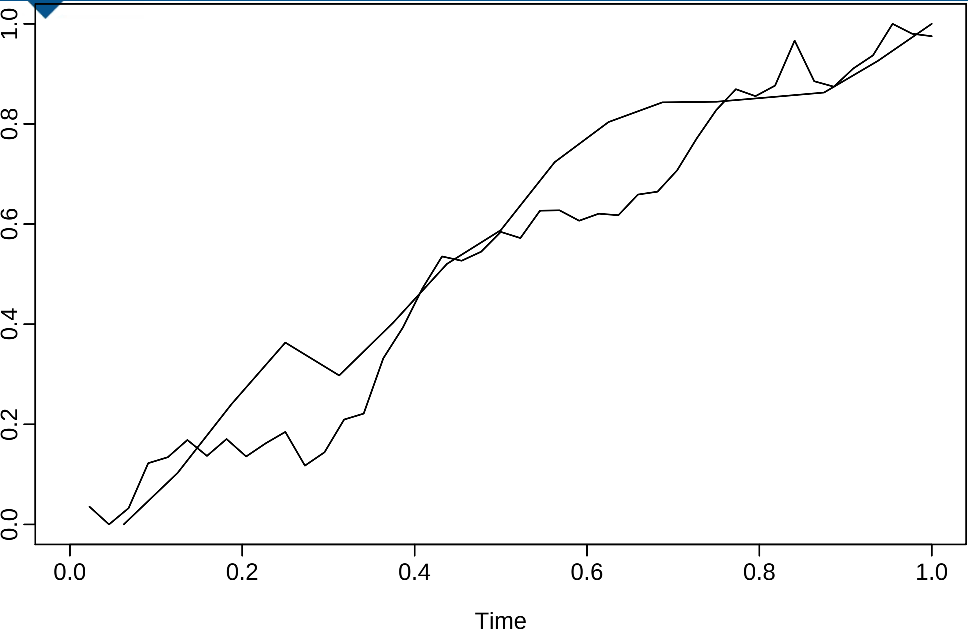
Outline

1. **The problem**
2. Australian tourism demand
3. M3 competition data
4. Yahoo web traffic
5. What next?

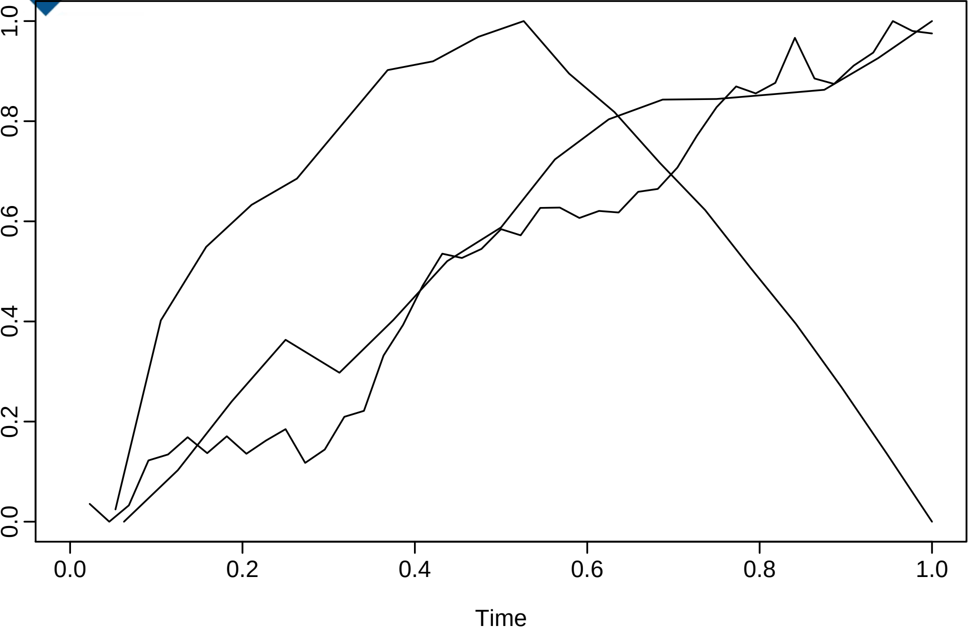
How to plot lots of time series?



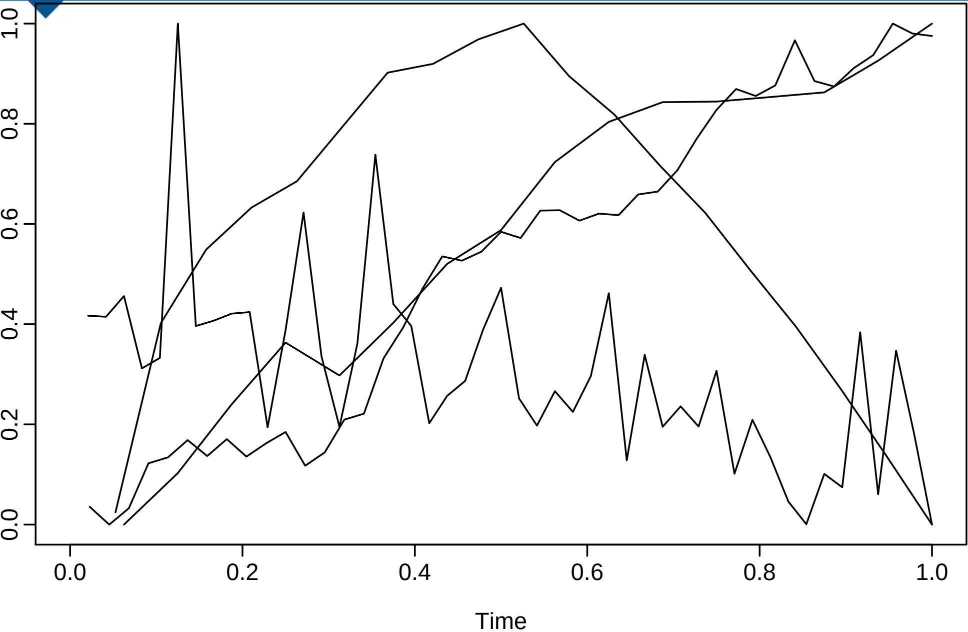
How to plot lots of time series?



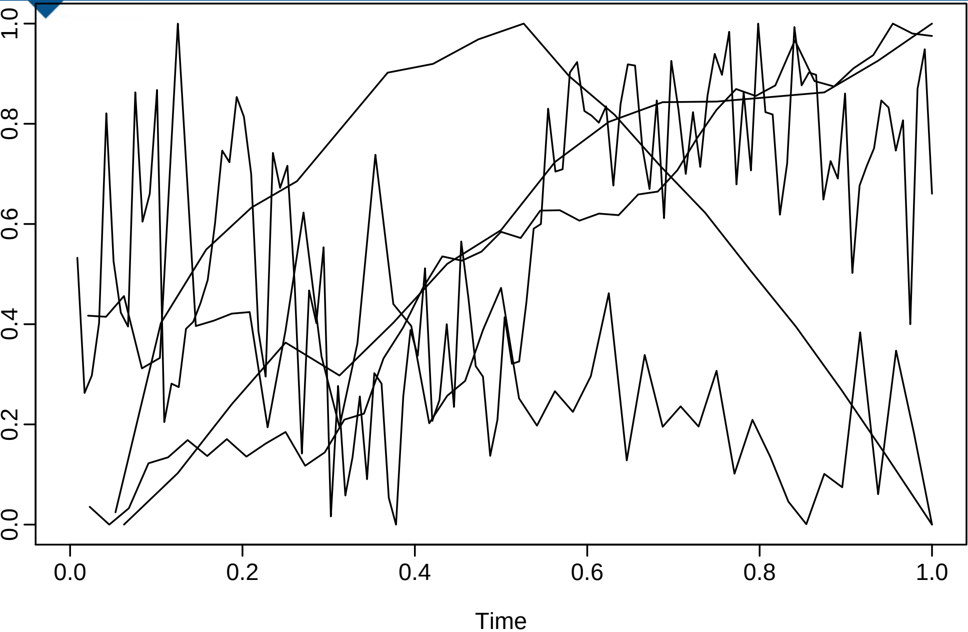
How to plot lots of time series?



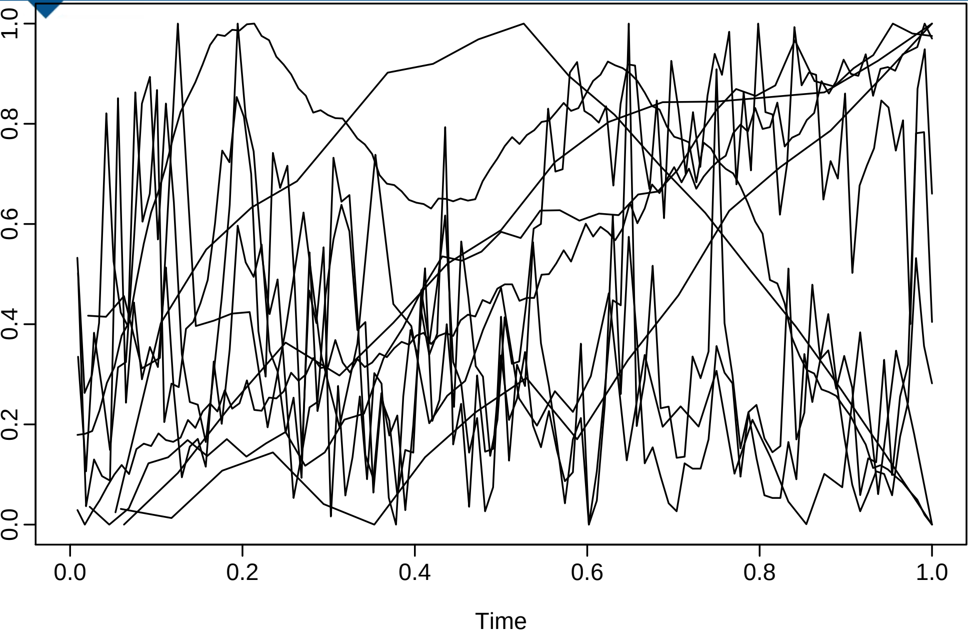
How to plot lots of time series?



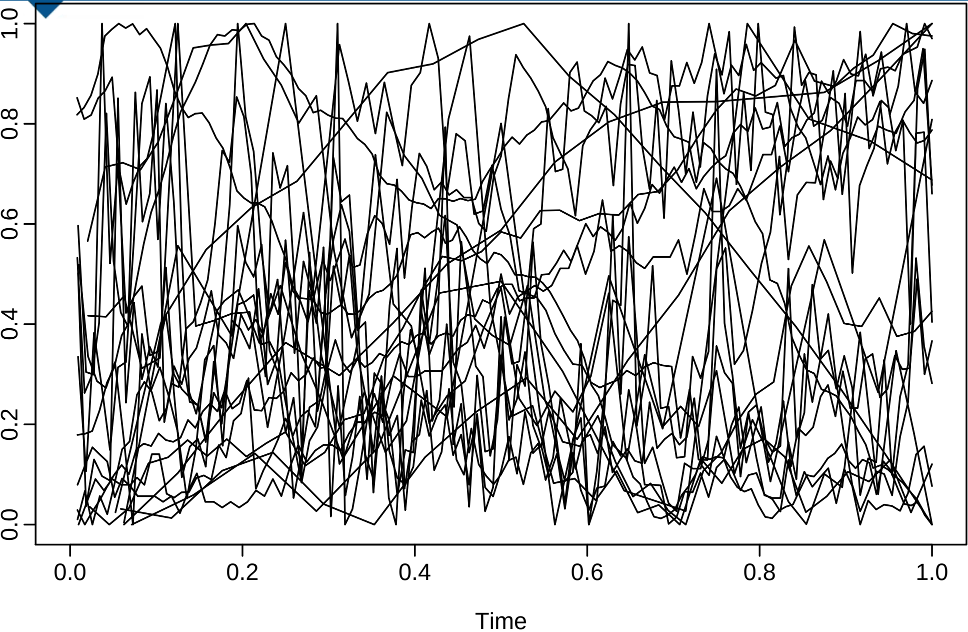
How to plot lots of time series?



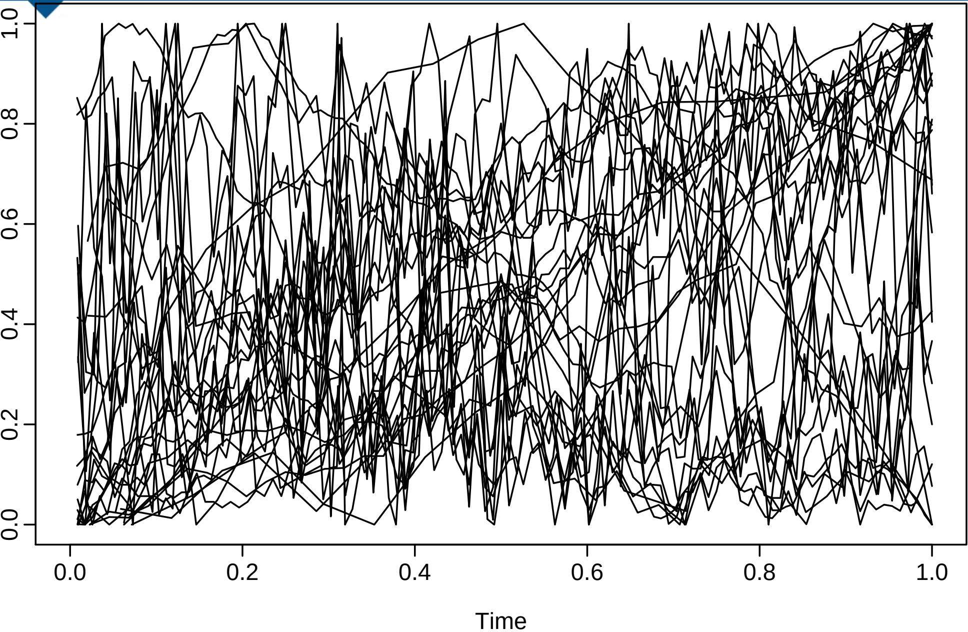
How to plot lots of time series?



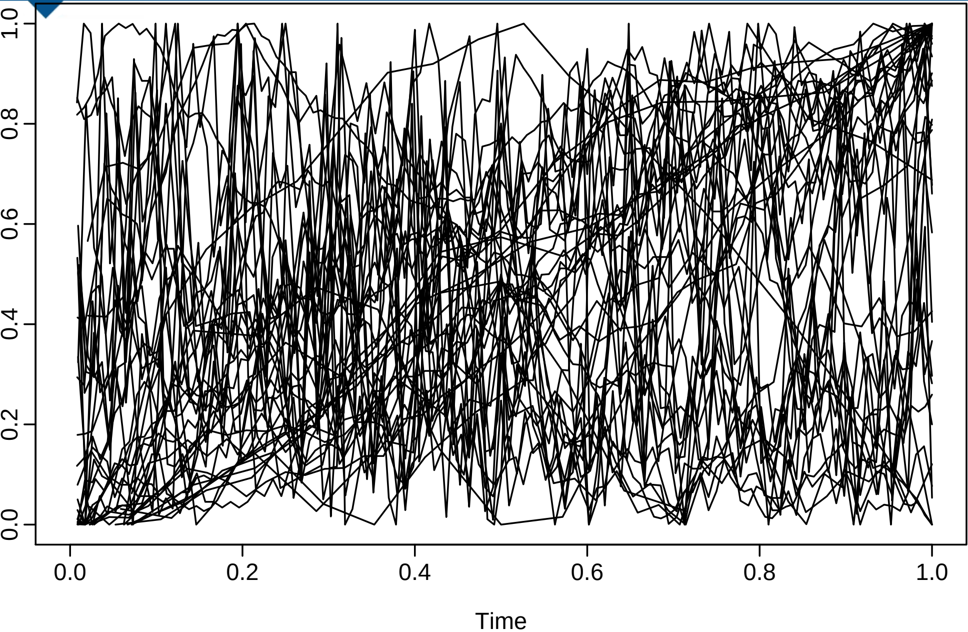
How to plot lots of time series?



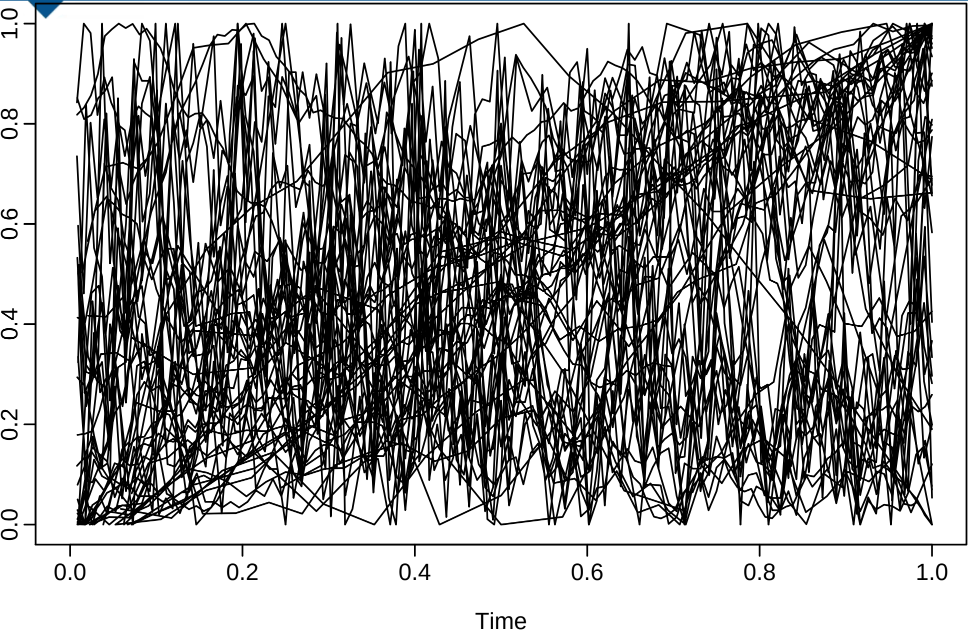
How to plot lots of time series?



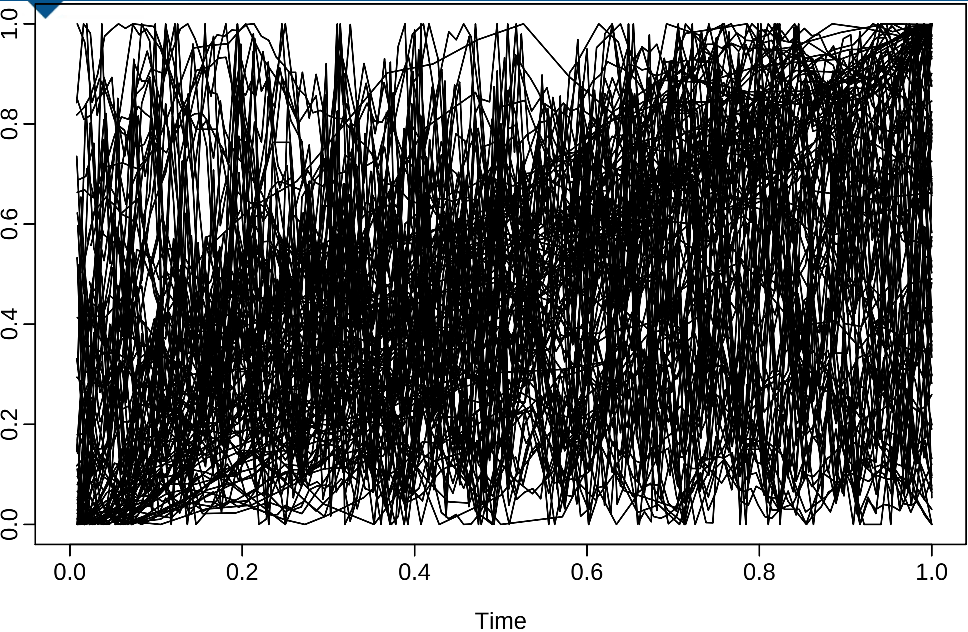
How to plot lots of time series?



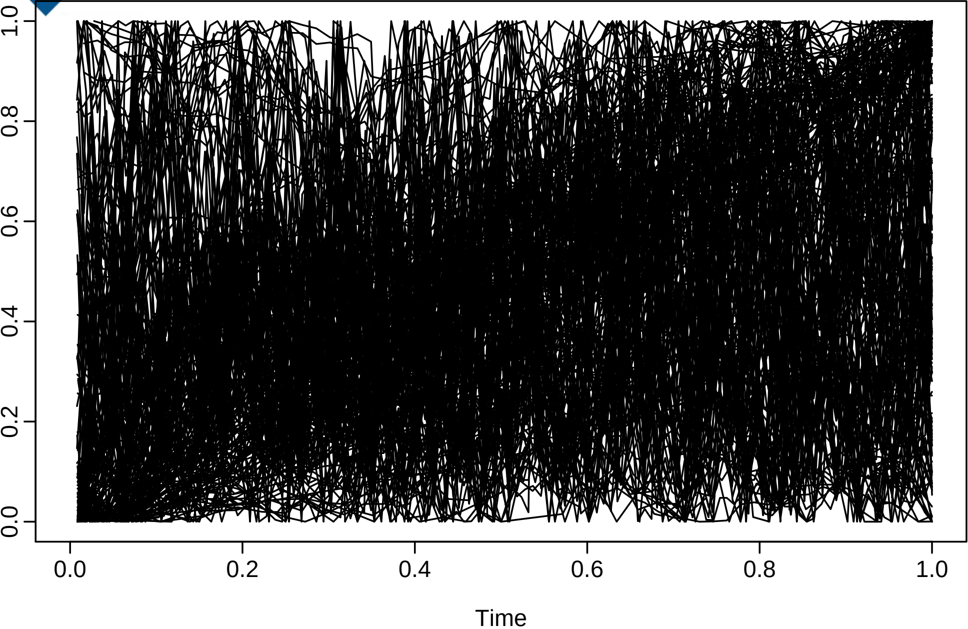
How to plot lots of time series?



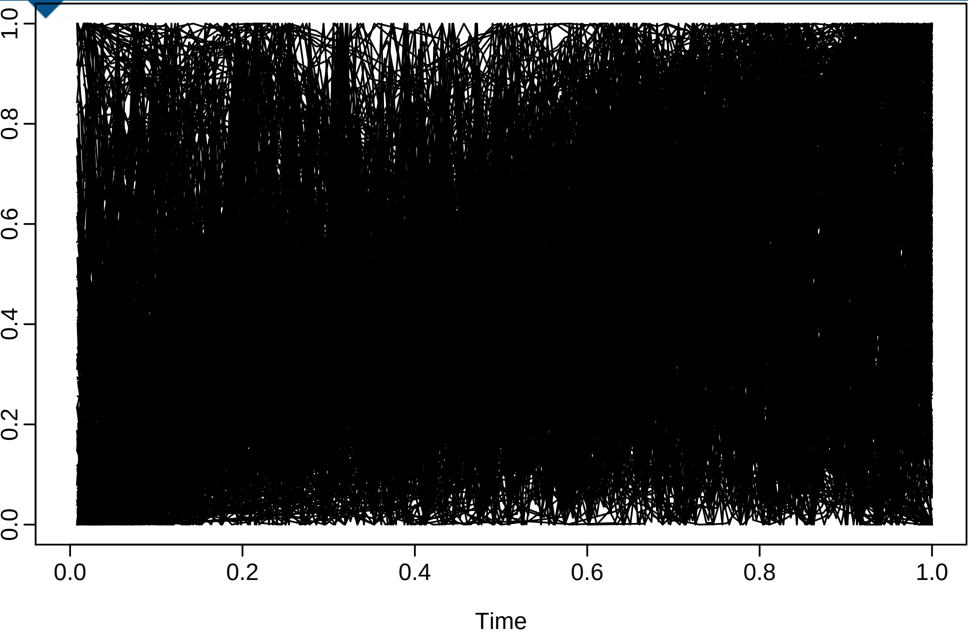
How to plot lots of time series?



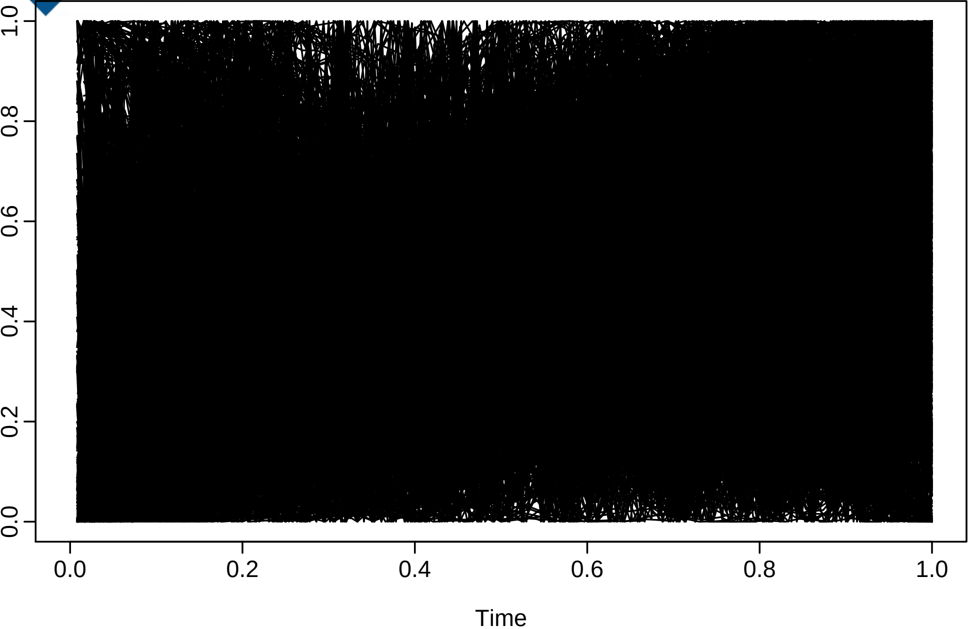
How to plot lots of time series?



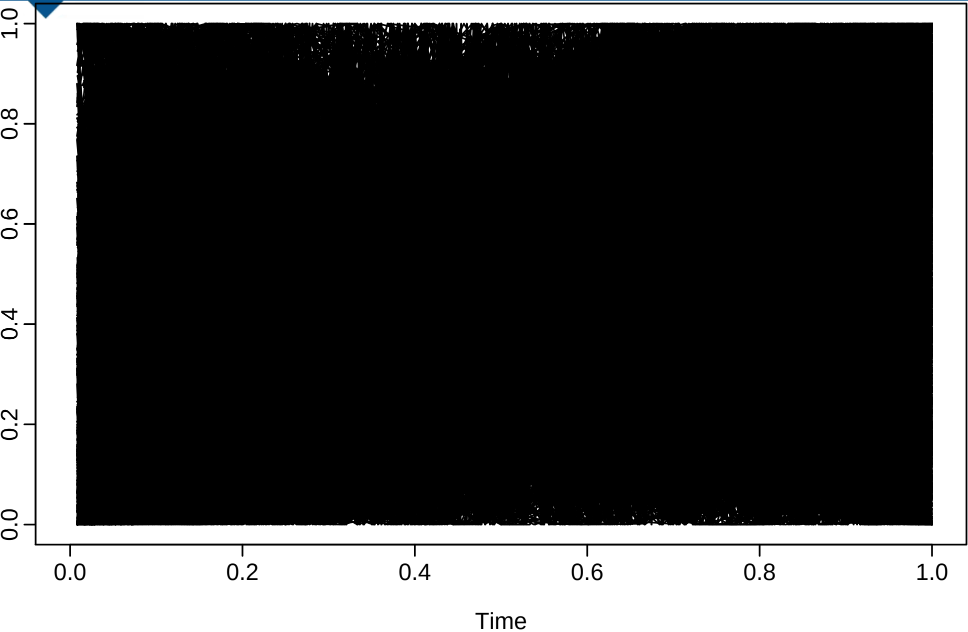
How to plot lots of time series?



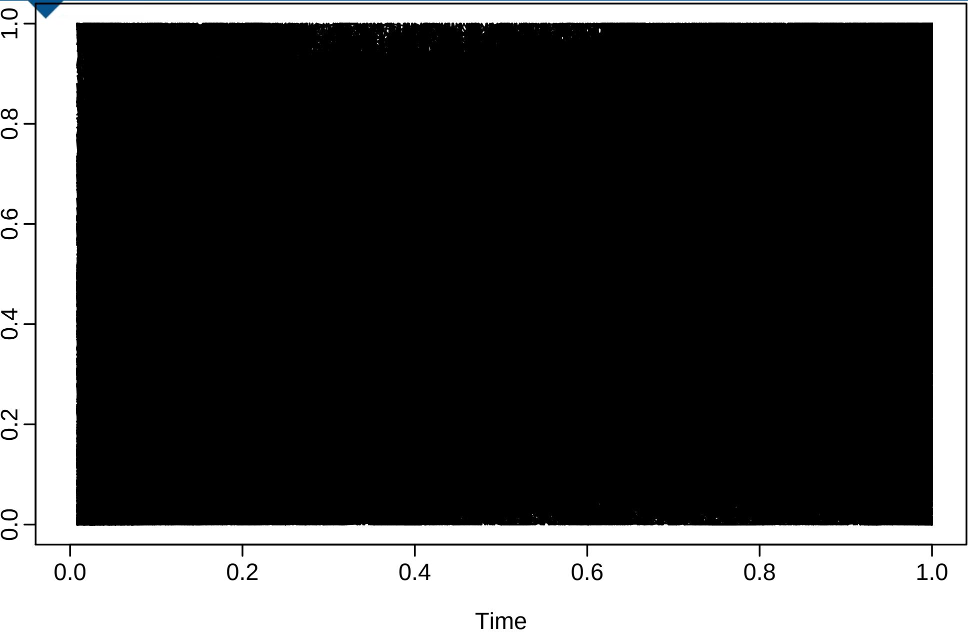
How to plot lots of time series?



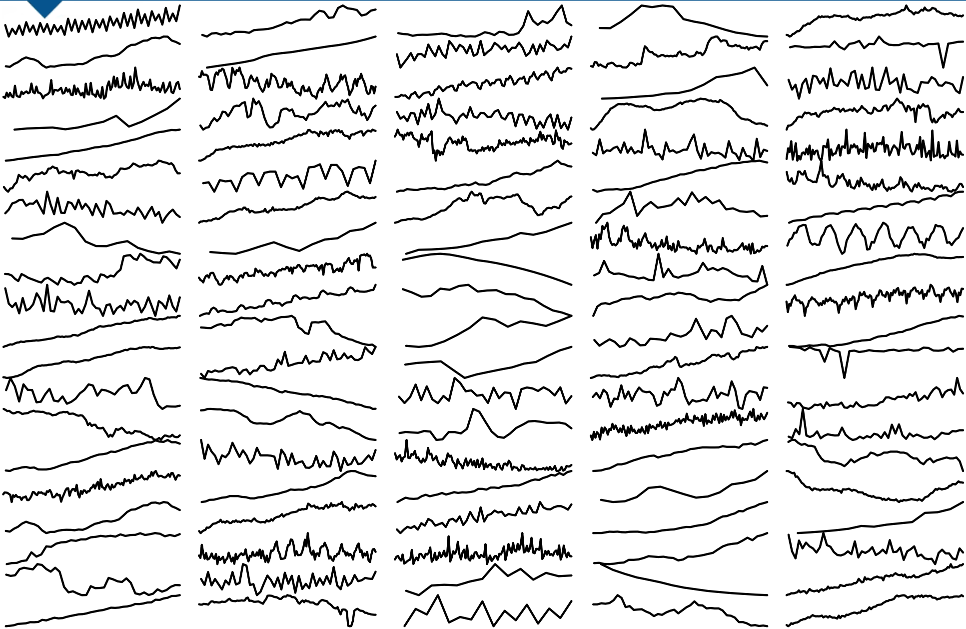
How to plot lots of time series?



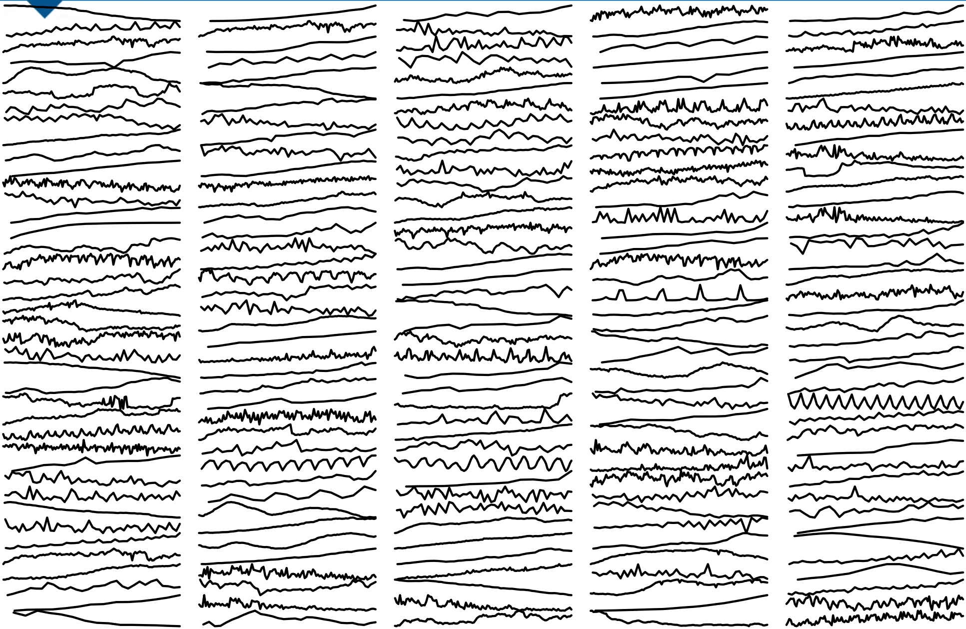
How to plot lots of time series?



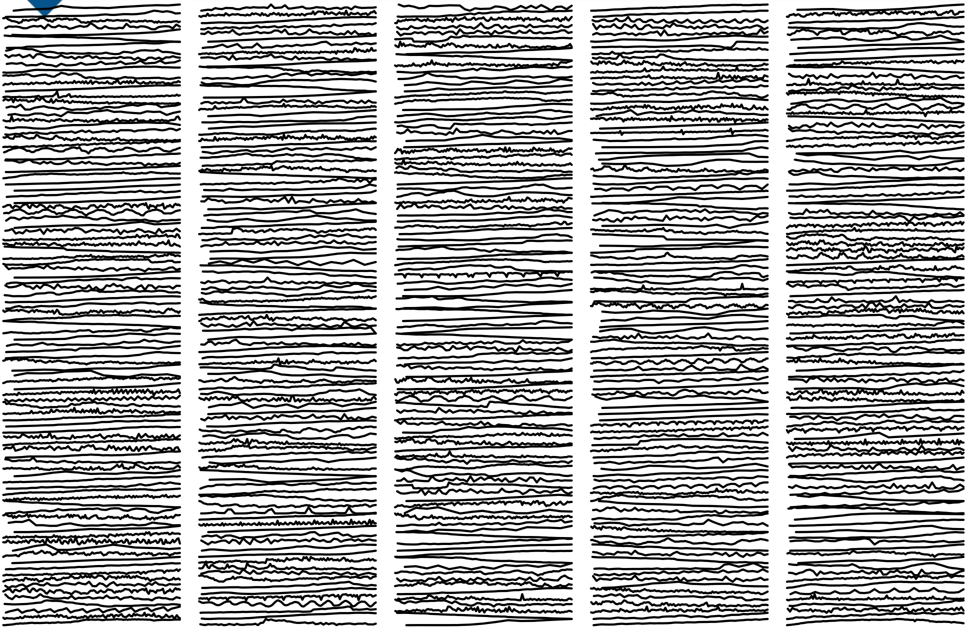
How to plot lots of time series?



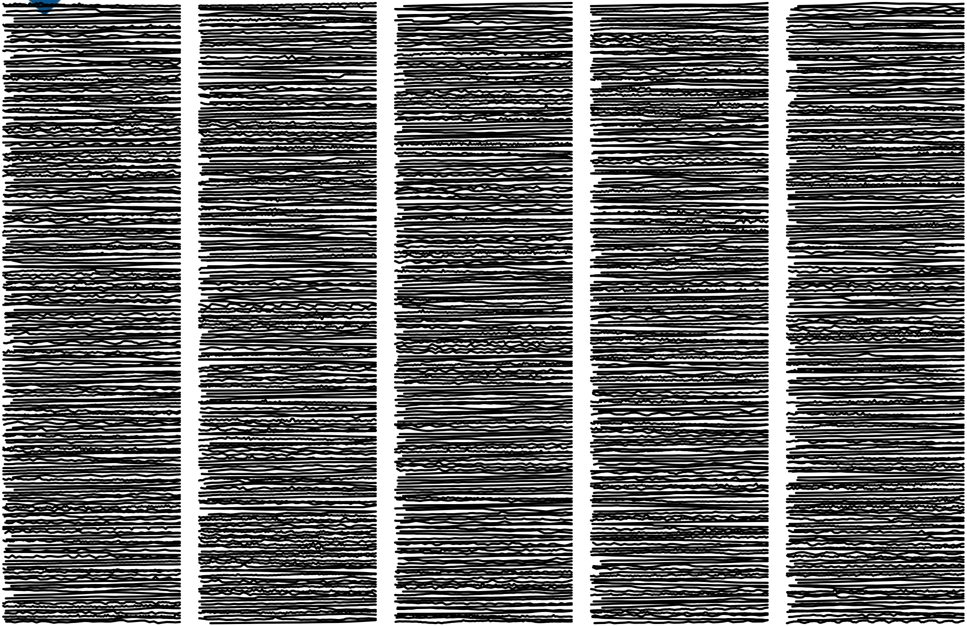
How to plot lots of time series?



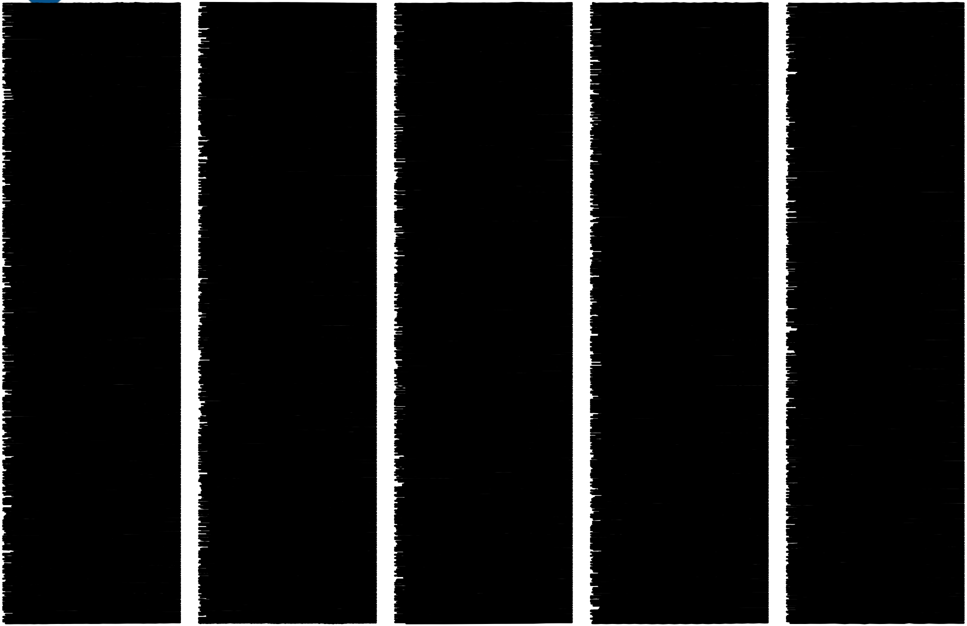
How to plot lots of time series?



How to plot lots of time series?



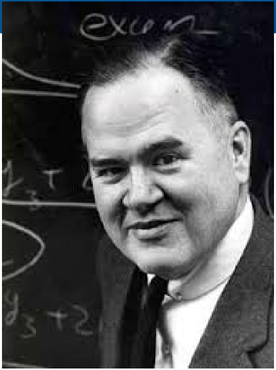
How to plot lots of time series?



Key idea

Cognostics

Computer-produced diagnostics (Tukey and Tukey, 1985).



John W Tukey

Examples for time series

- lag correlation
- size and direction of trend
- strength of seasonality
- timing of peak seasonality
- spectral entropy

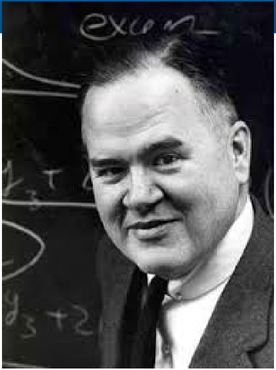
Key idea

Cognostics

Computer-produced diagnostics (Tukey and Tukey, 1985).

Examples for time series

- lag correlation
- size and direction of trend
- strength of seasonality
- timing of peak seasonality
- spectral entropy

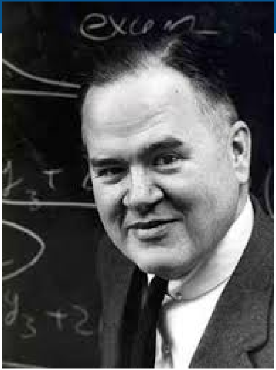


John W Tukey

Key idea

Cognostics

Computer-produced diagnostics (Tukey and Tukey, 1985).



John W Tukey

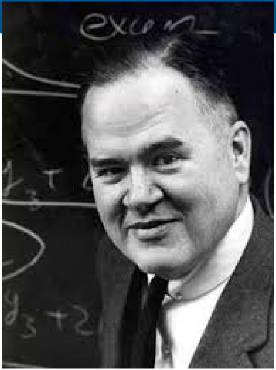
Examples for time series

- lag correlation
- size and direction of trend
- strength of seasonality
- timing of peak seasonality
- spectral entropy

Key idea

Cognostics

Computer-produced diagnostics (Tukey and Tukey, 1985).



John W Tukey

Examples for time series

- lag correlation
- size and direction of trend
- strength of seasonality
- timing of peak seasonality
- spectral entropy

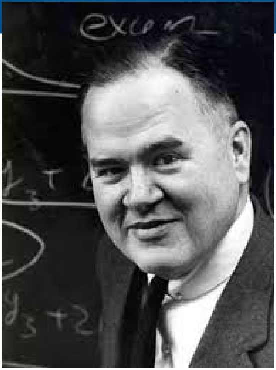
Key idea

Cognostics

Computer-produced diagnostics (Tukey and Tukey, 1985).

Examples for time series

- lag correlation
- size and direction of trend
- strength of seasonality
- timing of peak seasonality
- spectral entropy

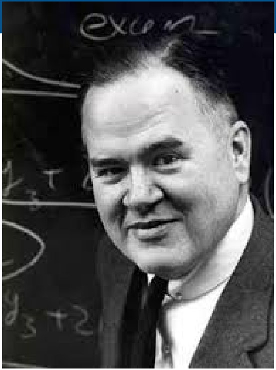


John W Tukey

Key idea

Cognostics

Computer-produced diagnostics (Tukey and Tukey, 1985).



John W Tukey

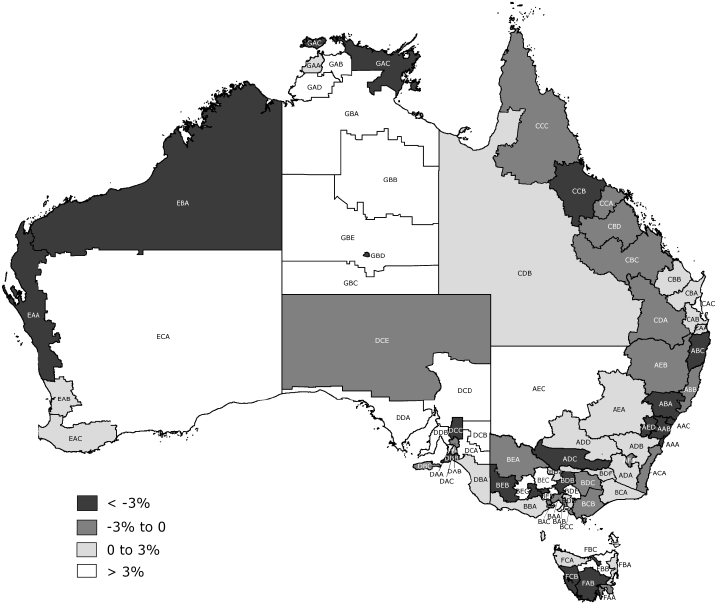
Examples for time series

- lag correlation
- size and direction of trend
- strength of seasonality
- timing of peak seasonality
- spectral entropy

Called "features" or "characteristics" in the machine learning literature.

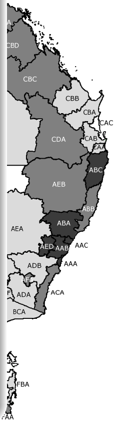
Outline

Australian tourism demand

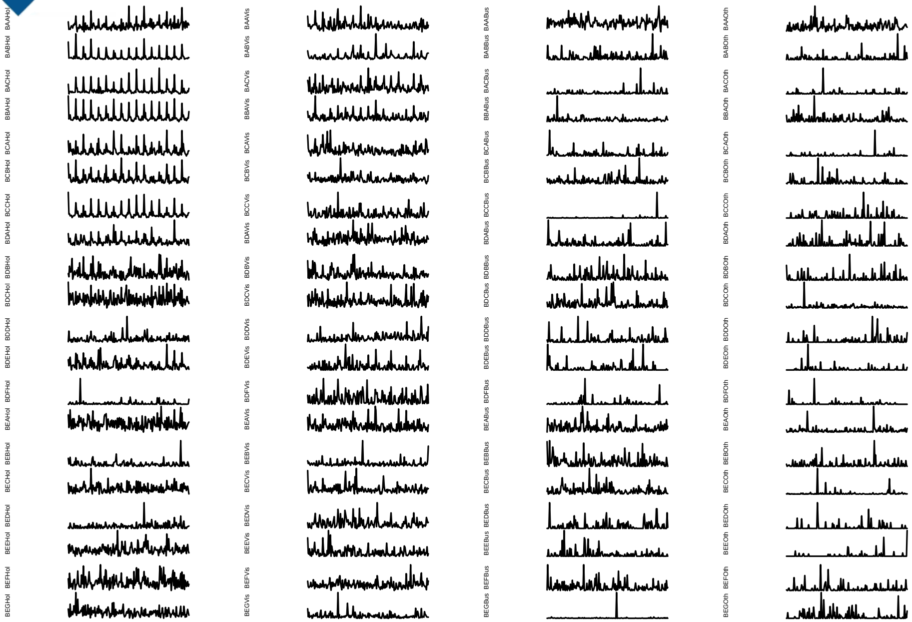


Australian tourism demand

- Quarterly data on visitor night from 1998:Q1 – 2013:Q4
- From: *National Visitor Survey*, based on annual interviews of 120,000 Australians aged 15+, collected by Tourism Research Australia.
- Split by 7 states, 27 zones and 76 regions (a geographical hierarchy)
- Also split by purpose of travel
  - Holiday
  - Visiting friends and relatives (VFR)
  - Business
  - Other
- 304 disaggregated series



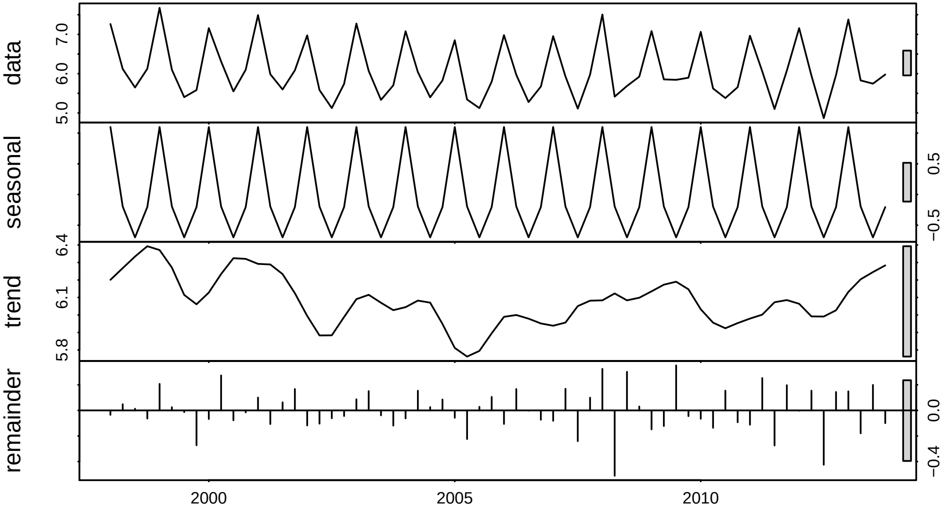
Domestic tourism demand: Victoria



An STL decomposition

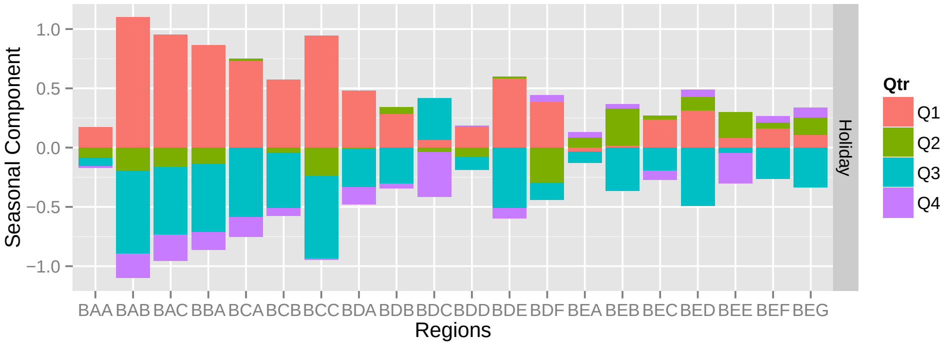
Tourism demand for holidays in Peninsula

$Y_t = S_t + T_t + R_t$ $\quad$ S_t is periodic with mean 0

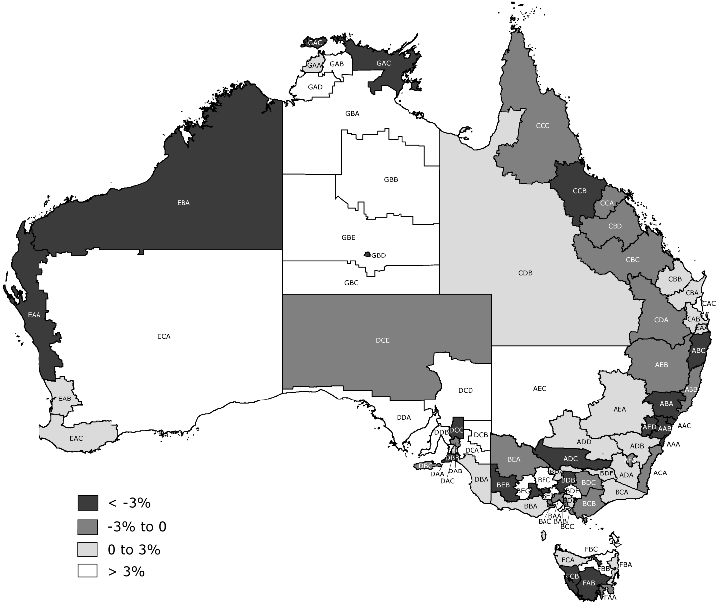


Seasonal stacked bar chart

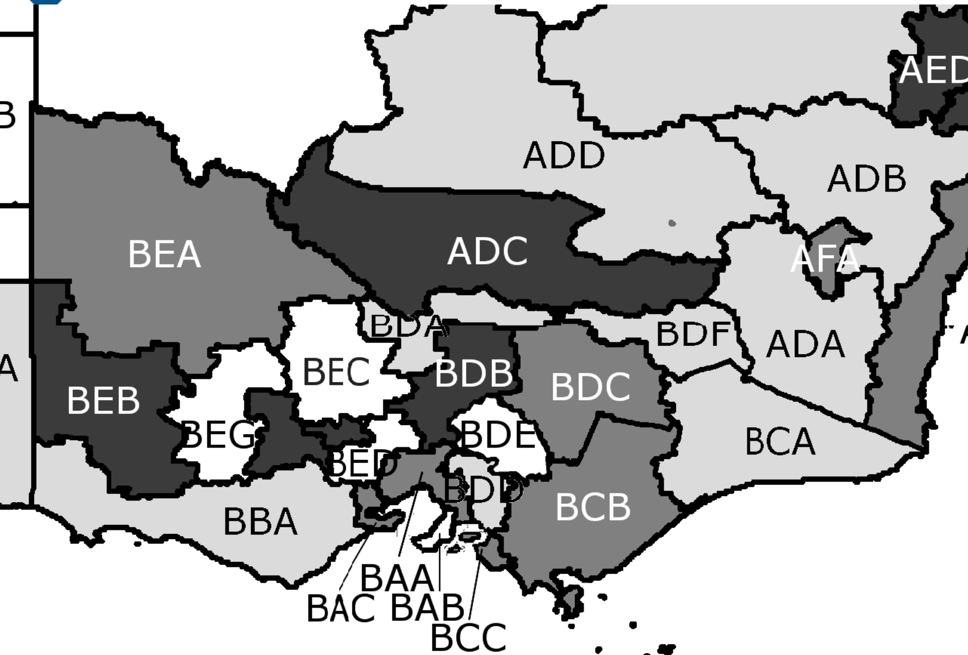
- Place positive values above the origin while negative values below the origin
- Map the bar length to the magnitude
- Encode quarters by colours



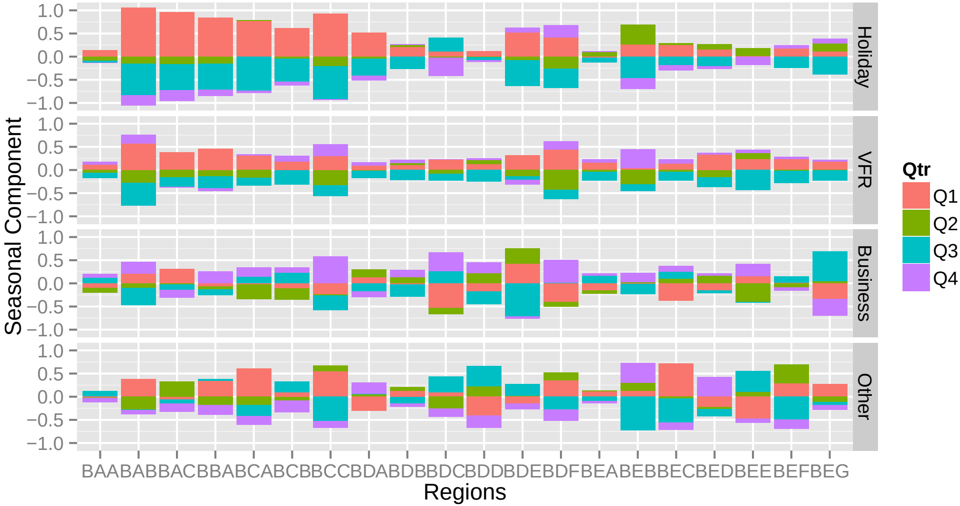
Seasonal stacked bar chart: VIC



Seasonal stacked bar chart: VIC



Seasonal stacked bar chart: VIC



Trend analysis

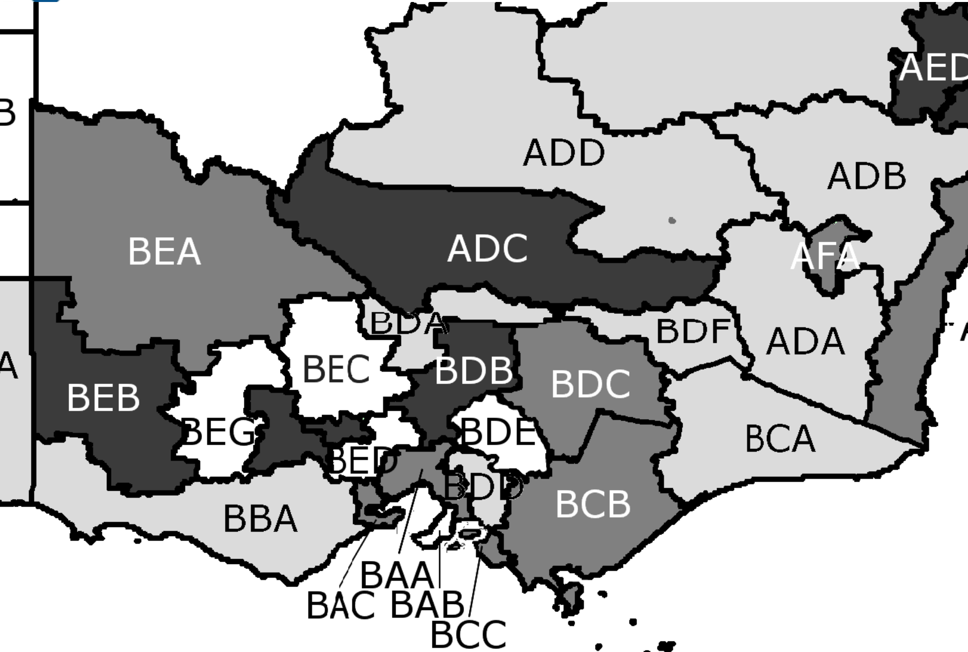
- **Linearity:** the long-term direction and strength of trend.
- **Curvature:** the “changing direction” of trend.
- Estimate by regression:

$$T_t = \hat{\beta}_0 + \hat{\beta}_1\phi_1(t) + \hat{\beta}_2\phi_2(t) + e_t$$

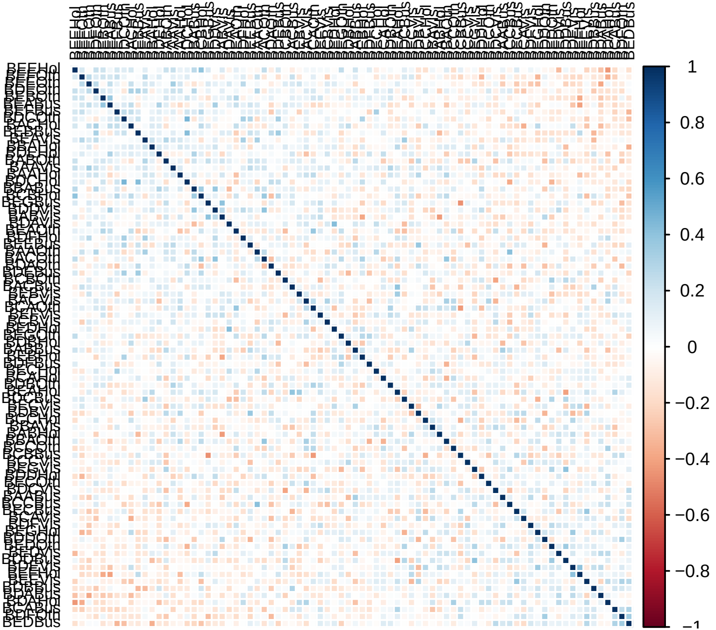
  where $\phi_k(t)$ is a kth-degree **orthogonal polynomial** in time t.
- To separate the linearity ($\hat{\beta}_1$) and curvature ($\hat{\beta}_2$).

Trend analysis

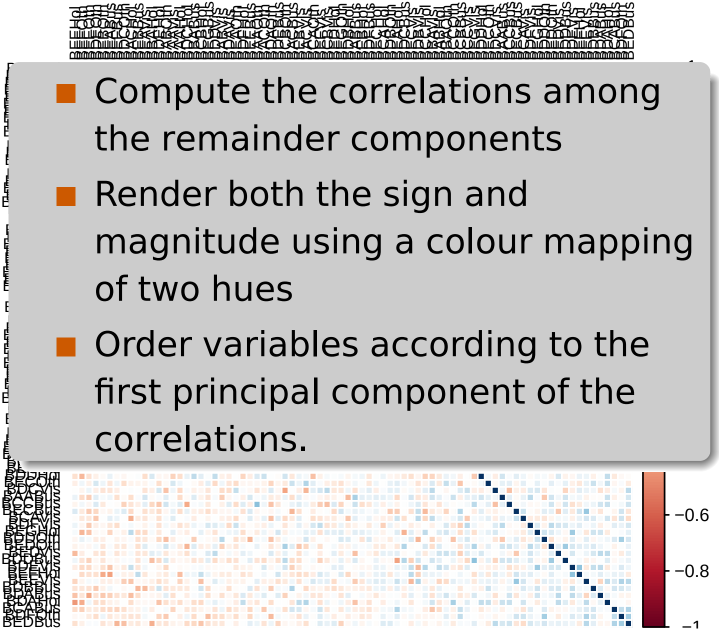
Trend analysis



Corrgram of remainder

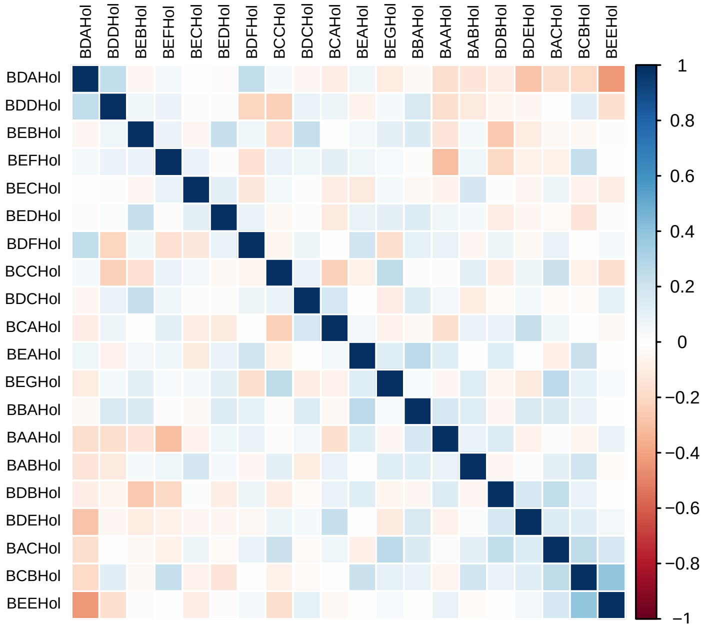


Corrgram of remainder

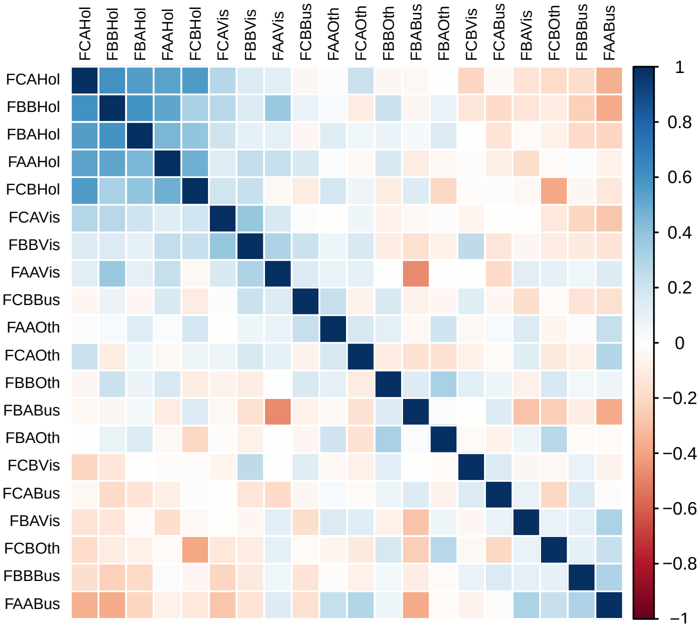


- Compute the correlations among the remainder components
- Render both the sign and magnitude using a colour mapping of two hues
- Order variables according to the first principal component of the correlations.

Corrgram of remainder



Corrgram of remainder: TAS

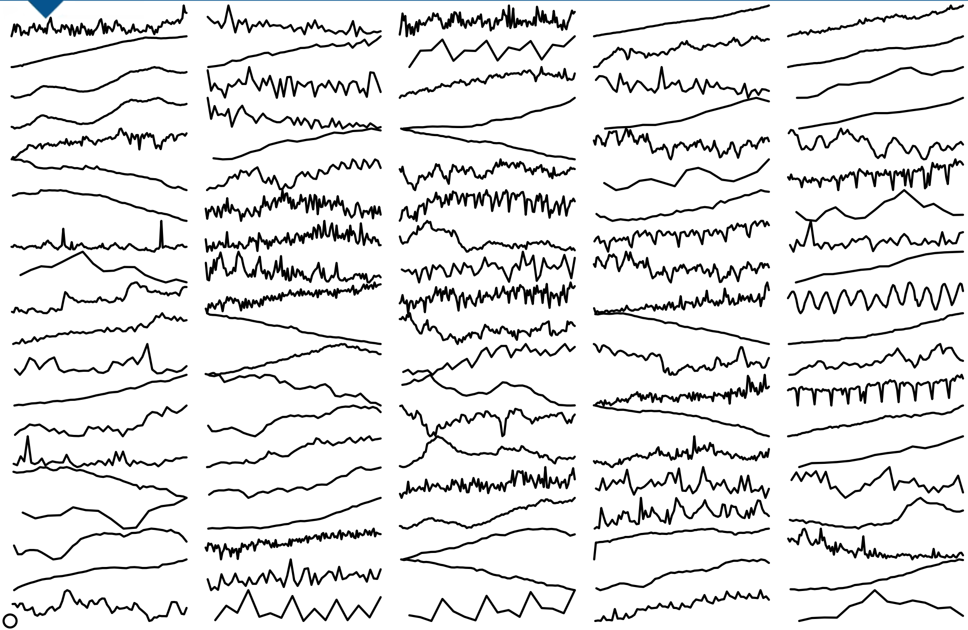


Outline

M3 forecasting competition

- 3003 series
- All data from business, demography, finance and economics.
- Series length between 14 and 126.
- Either non-seasonal, monthly or quarterly.
- All time series positive.

M3 forecasting competition



Candidate features

STL decomposition

$$Y_t = S_t + T_t + R_t$$

Candidate features

STL decomposition

$$Y_t = S_t + T_t + R_t$$

- Seasonal period
- Strength of seasonality: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - T_t)}$
- Strength of trend: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - S_t)}$
- Spectral entropy: $H = -\int_{-\pi}^{\pi} f_y(\lambda) \log f_y(\lambda) d\lambda$, where $f_y(\lambda)$ is spectral density of Y_t. Low values of H suggest a time series that is easier to forecast (more signal).
- Autocorrelations: $r_1, r_2, r_3, \dots$
- Optimal Box-Cox transformation parameter λ

Candidate features

STL decomposition

$$Y_t = S_t + T_t + R_t$$

- Seasonal period
- Strength of seasonality: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - T_t)}$
- Strength of trend: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - S_t)}$
- Spectral entropy: $H = -\int_{-\pi}^{\pi} f_y(\lambda) \log f_y(\lambda) d\lambda$, where $f_y(\lambda)$ is spectral density of Y_t. Low values of H suggest a time series that is easier to forecast (more signal).
- Autocorrelations: $r_1, r_2, r_3, \ldots$
- Optimal Box-Cox transformation parameter λ

Candidate features

STL decomposition

$$Y_t = S_t + T_t + R_t$$

- Seasonal period
- Strength of seasonality: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - T_t)}$
- Strength of trend: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - S_t)}$
- Spectral entropy: $H = -\int_{-\pi}^{\pi} f_y(\lambda) \log f_y(\lambda) d\lambda$, where $f_y(\lambda)$ is spectral density of Y_t. Low values of H suggest a time series that is easier to forecast (more signal).
- Autocorrelations: $r_1, r_2, r_3, \ldots$
- Optimal Box-Cox transformation parameter λ

Candidate features

STL decomposition

$$Y_t = S_t + T_t + R_t$$

- Seasonal period
- Strength of seasonality: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - T_t)}$
- Strength of trend: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - S_t)}$
- Spectral entropy: $H = -\int_{-\pi}^{\pi} f_y(\lambda) \log f_y(\lambda) d\lambda$, where $f_y(\lambda)$ is spectral density of Y_t.
  Low values of H suggest a time series that is easier to forecast (more signal).
- Autocorrelations: $r_1, r_2, r_3, \dots$
- Optimal Box-Cox transformation parameter λ

Candidate features

STL decomposition

$$Y_t = S_t + T_t + R_t$$

- Seasonal period
- Strength of seasonality: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - T_t)}$
- Strength of trend: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - S_t)}$
- Spectral entropy: $H = -\int_{-\pi}^{\pi} f_y(\lambda) \log f_y(\lambda) d\lambda$, where $f_y(\lambda)$ is spectral density of Y_t.
  Low values of H suggest a time series that is easier to forecast (more signal).
- Autocorrelations: $r_1, r_2, r_3, \ldots$
- Optimal Box-Cox transformation parameter λ

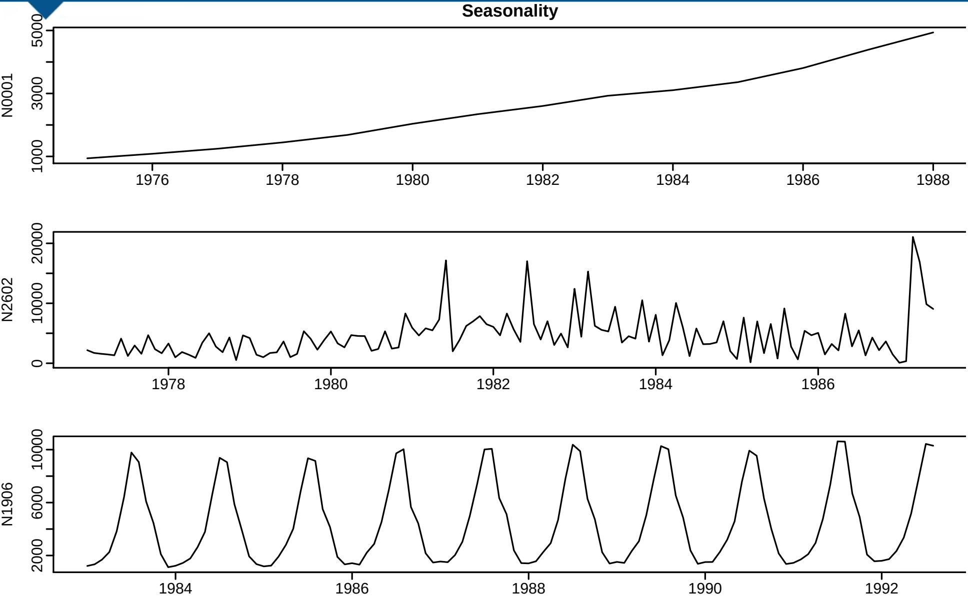
Candidate features

STL decomposition

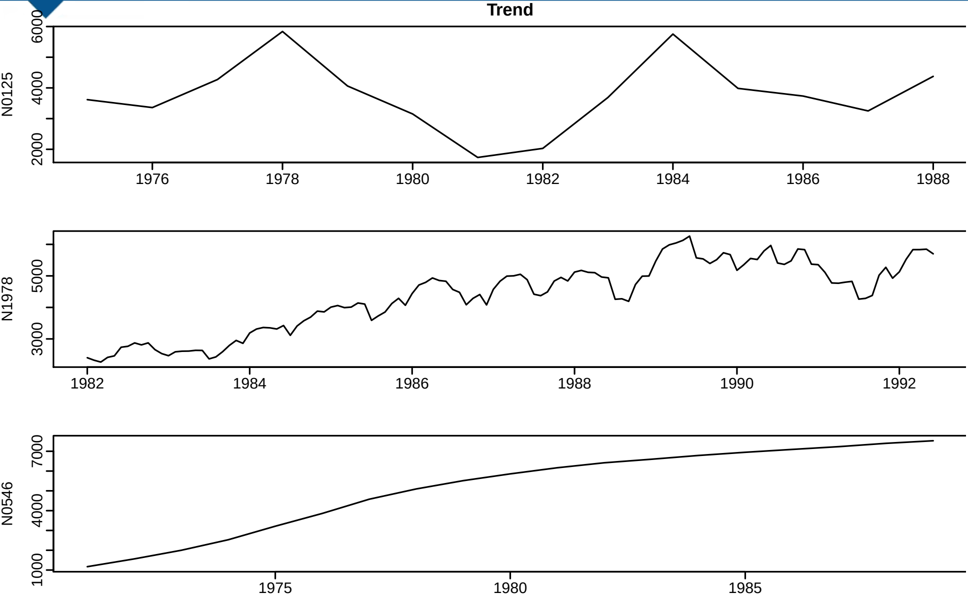
$$Y_t = S_t + T_t + R_t$$

- Seasonal period
- Strength of seasonality: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - T_t)}$
- Strength of trend: $1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - S_t)}$
- Spectral entropy: $H = -\int_{-\pi}^{\pi} f_y(\lambda) \log f_y(\lambda) d\lambda$, where $f_y(\lambda)$ is spectral density of Y_t.
  Low values of H suggest a time series that is easier to forecast (more signal).
- Autocorrelations: $r_1, r_2, r_3, \dots$
- Optimal Box-Cox transformation parameter λ

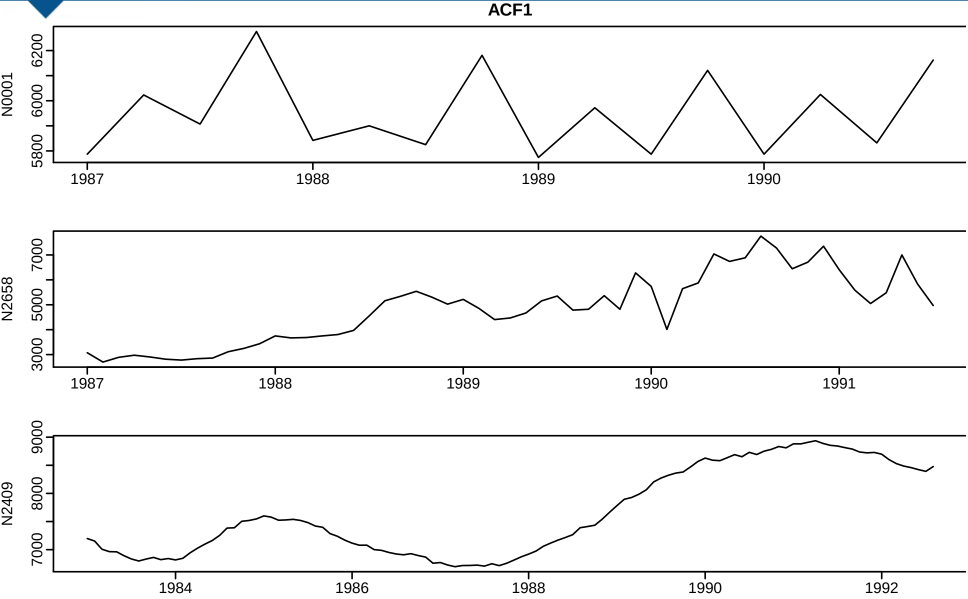
Candidate features



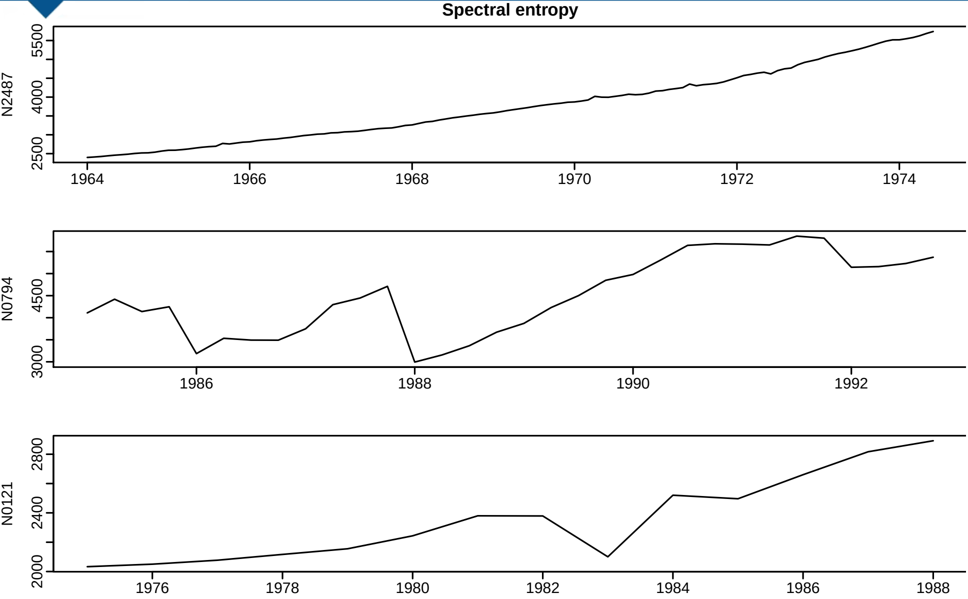
Candidate features



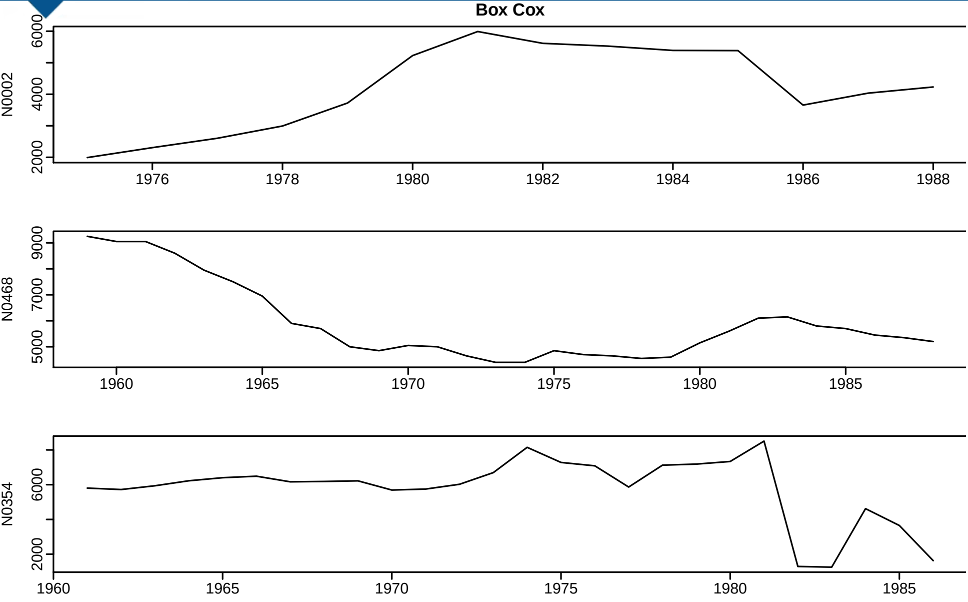
Candidate features



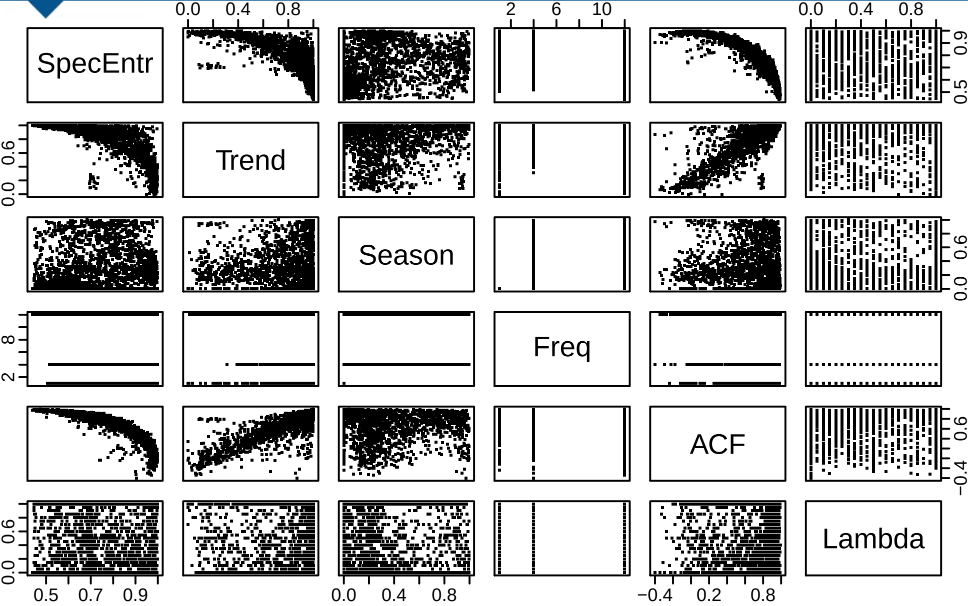
Candidate features



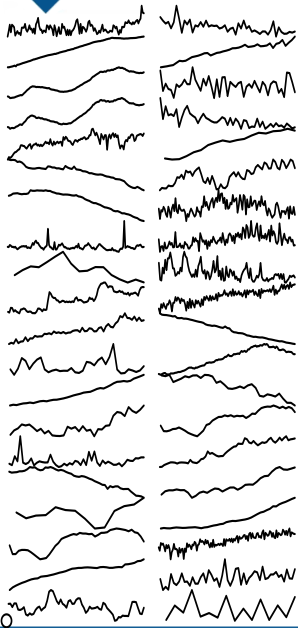
Candidate features



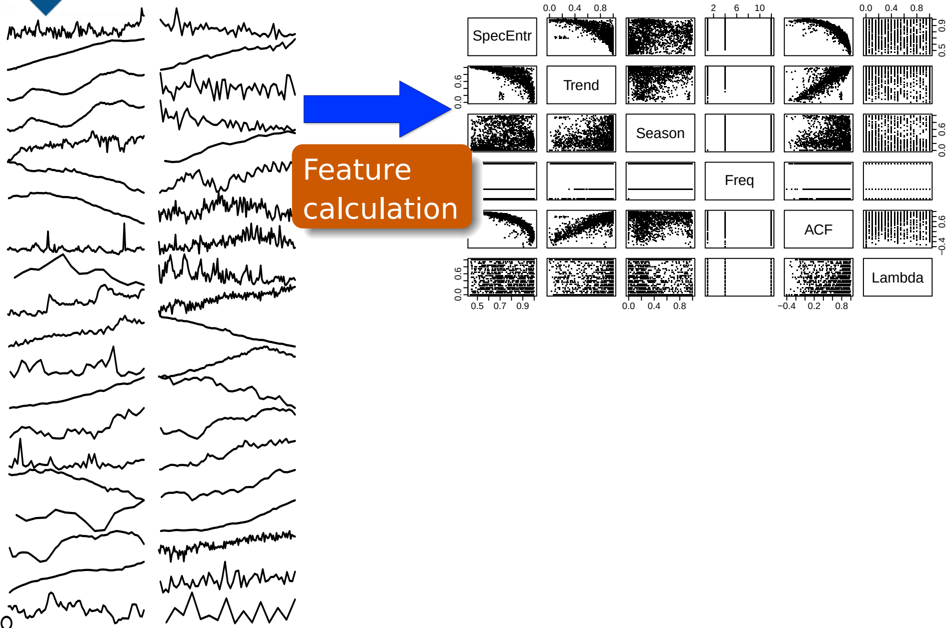
Candidate features



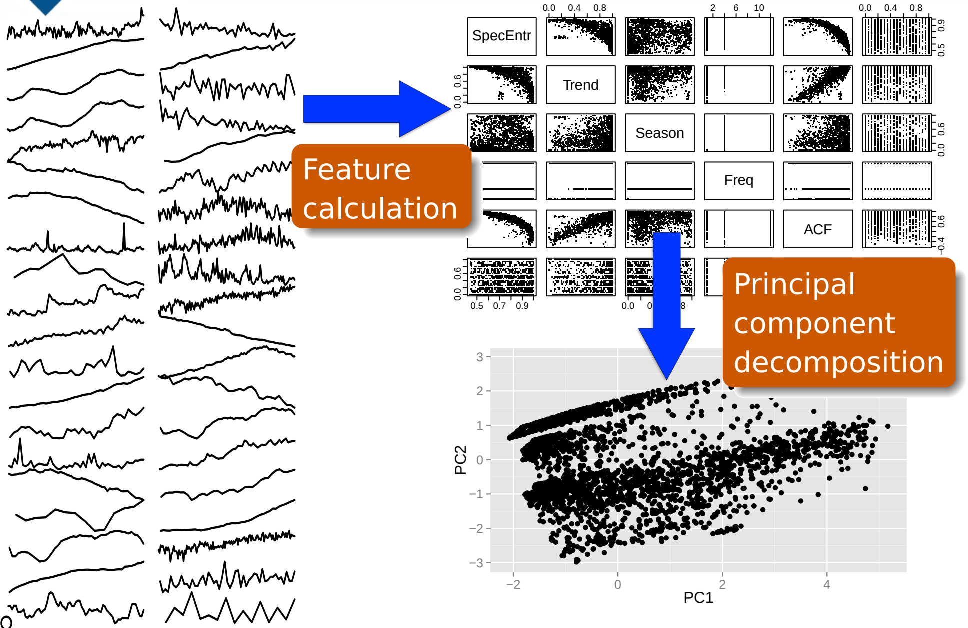
Dimension reduction for time series



Dimension reduction for time series

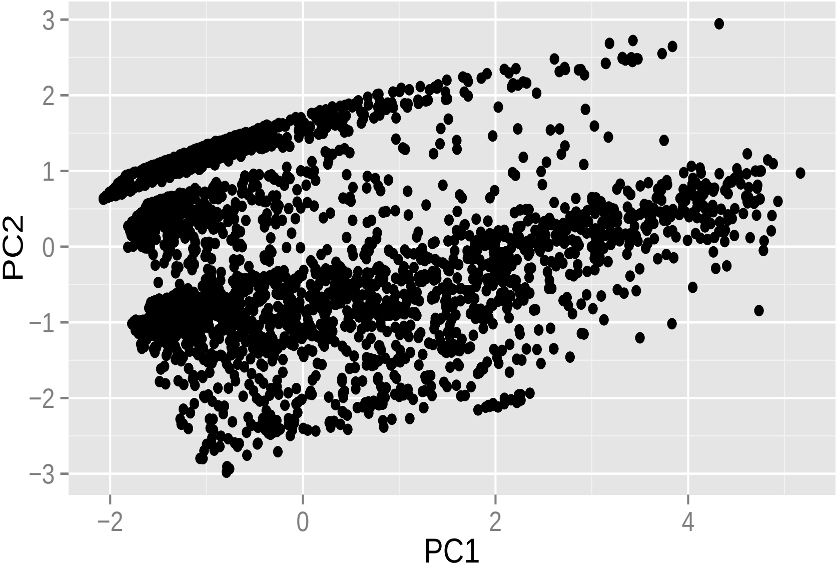


Dimension reduction for time series

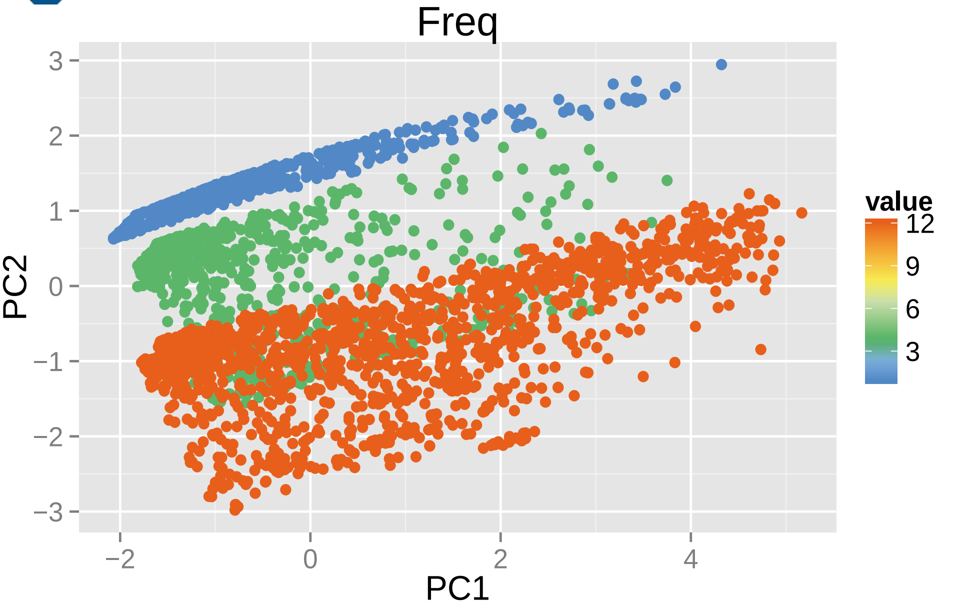


Feature space of M3 data

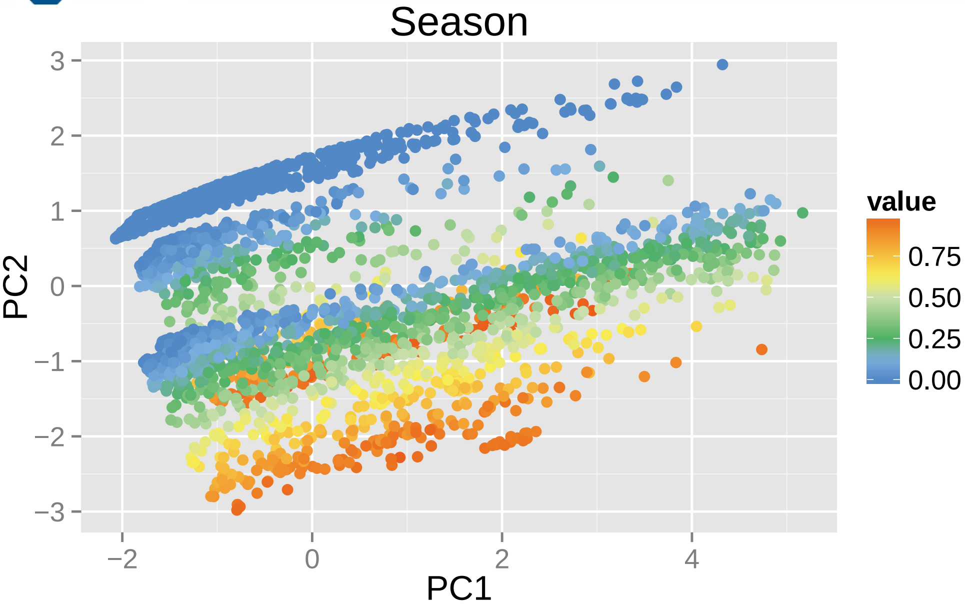
First two PCs explain 68% of variation.



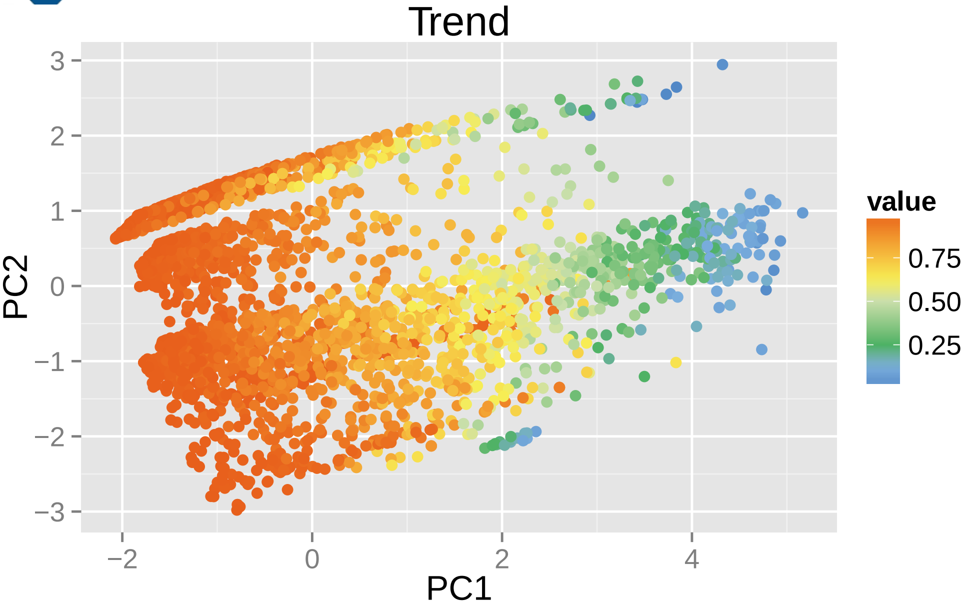
Feature space of M3 data



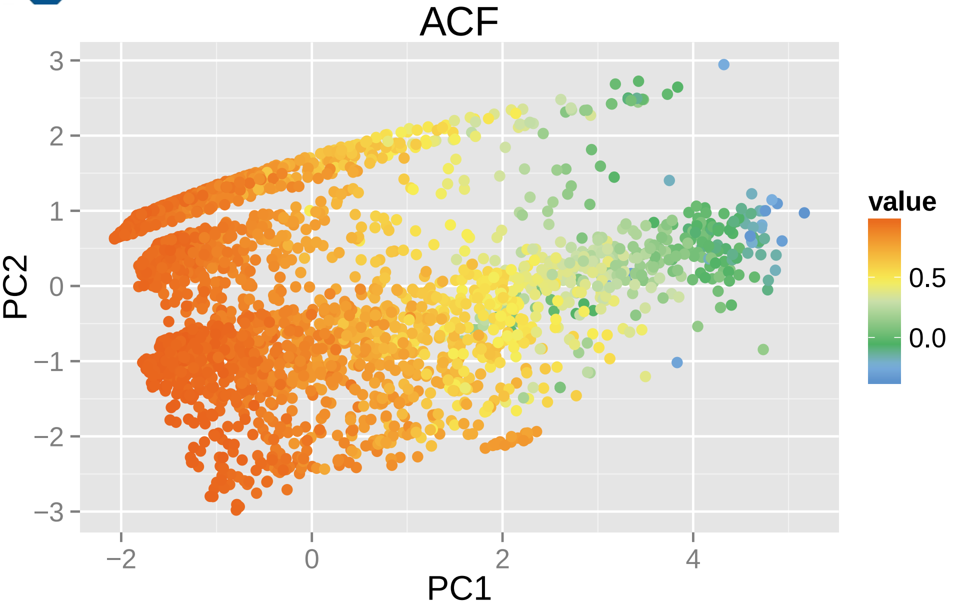
Feature space of M3 data



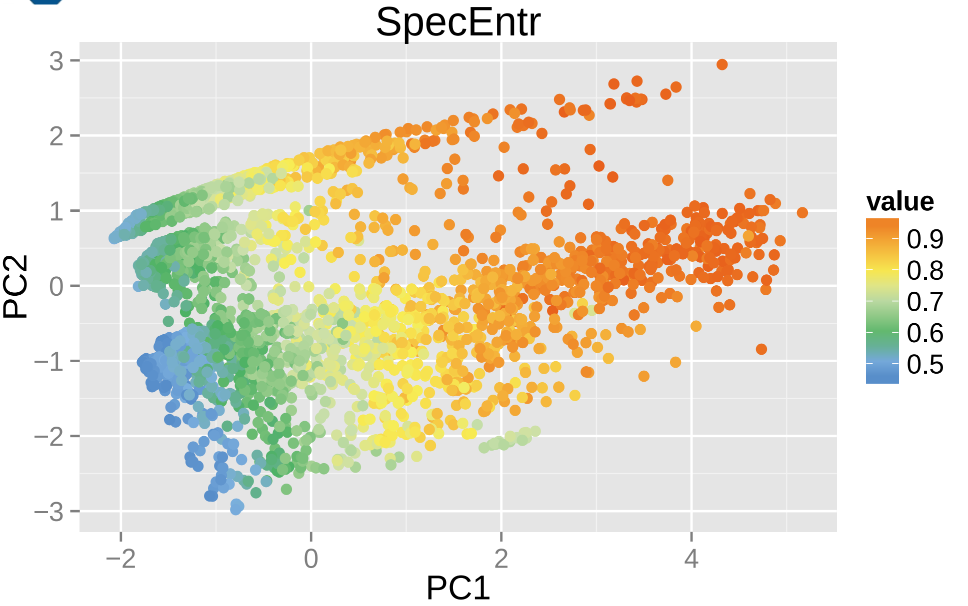
Feature space of M3 data



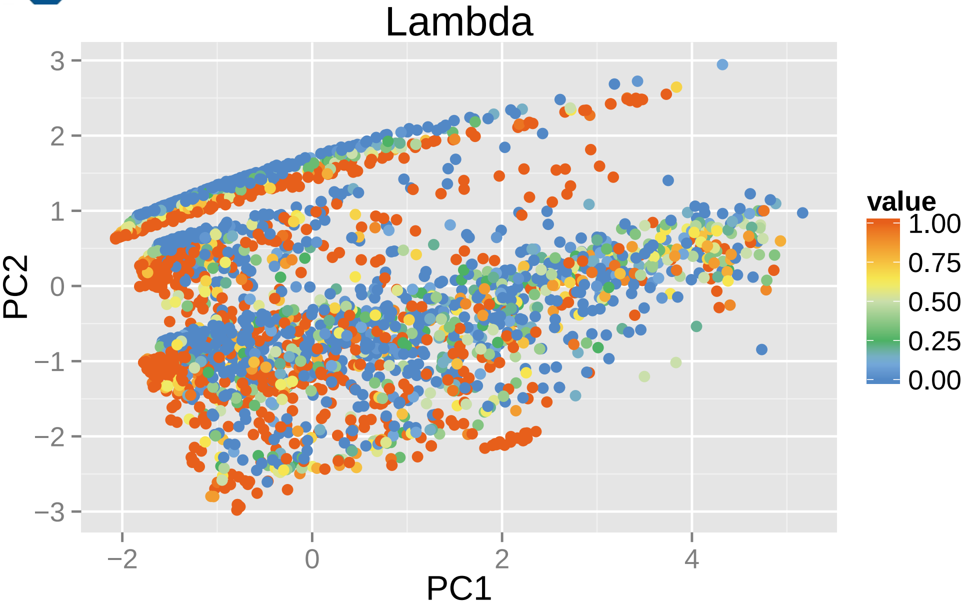
Feature space of M3 data



Feature space of M3 data



Feature space of M3 data



Predictability

Three general forecasting methods:

Theta method	Best overall in 2000 M3 competition
ETS	Exponential smoothing state space models
STL-AR	AR model applied to seasonally adjusted series from STL, and seasonal component forecast using the seasonal naive method.

Compute minimum MASE from all three methods

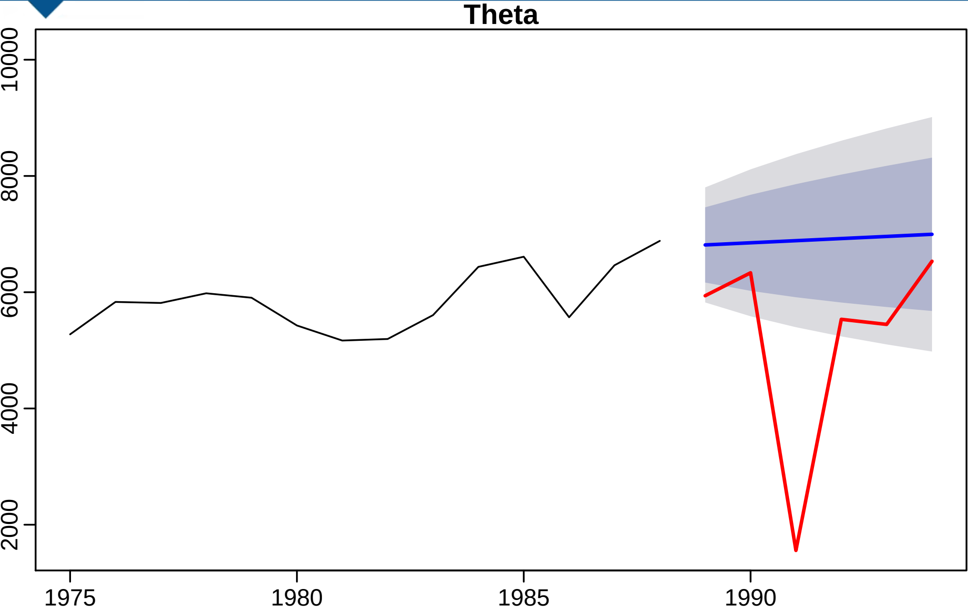
Predictability

Three general forecasting methods:

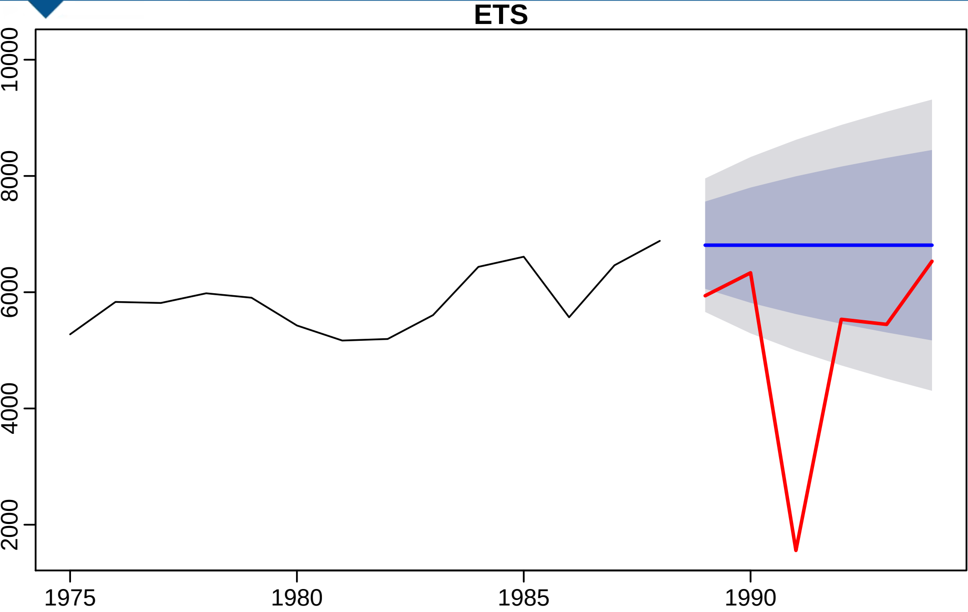
Theta method	Best overall in 2000 M3 competition
ETS	Exponential smoothing state space models
STL-AR	AR model applied to seasonally adjusted series from STL, and seasonal component forecast using the seasonal naive method.

Compute minimum MASE from all three methods

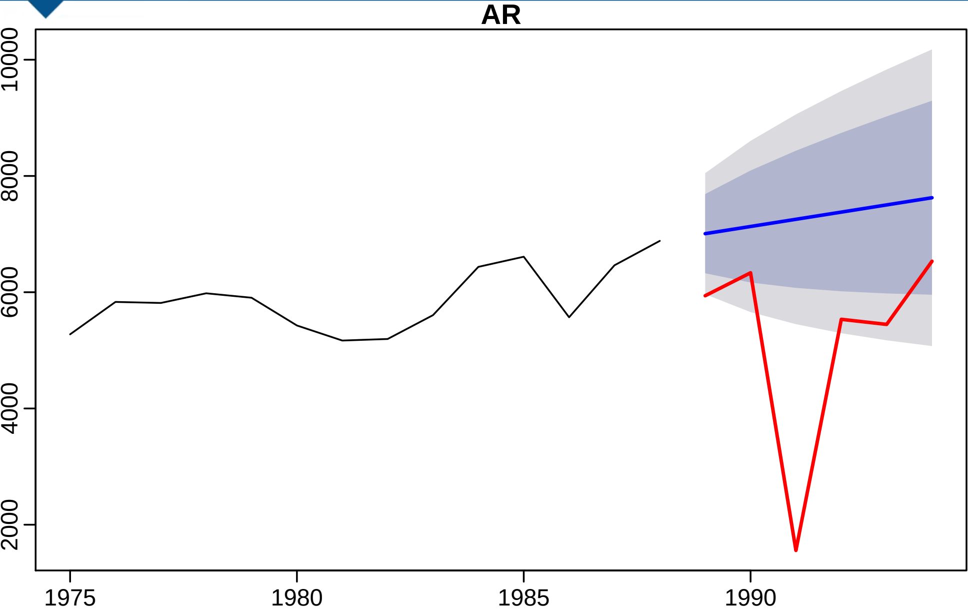
Predictability



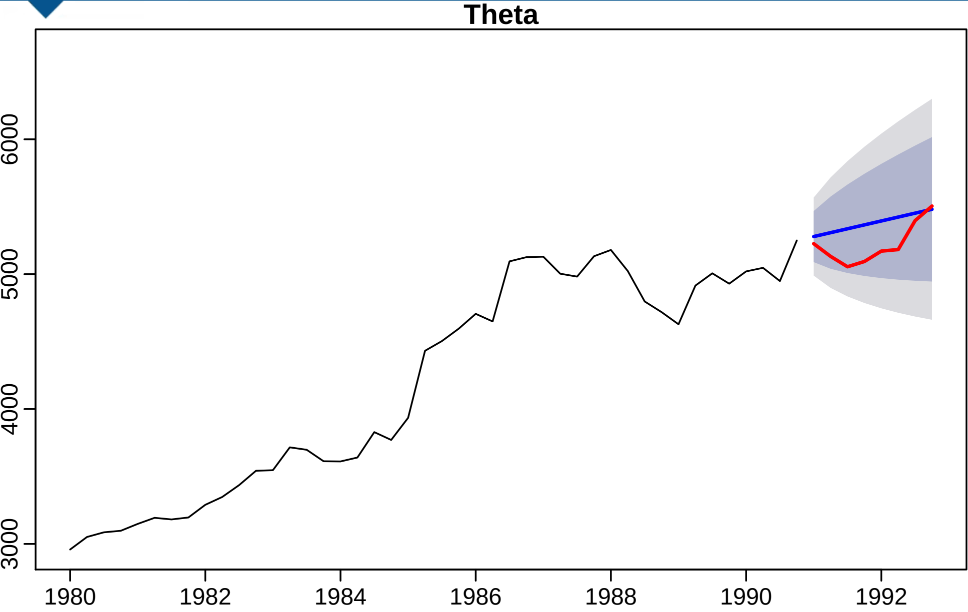
Predictability



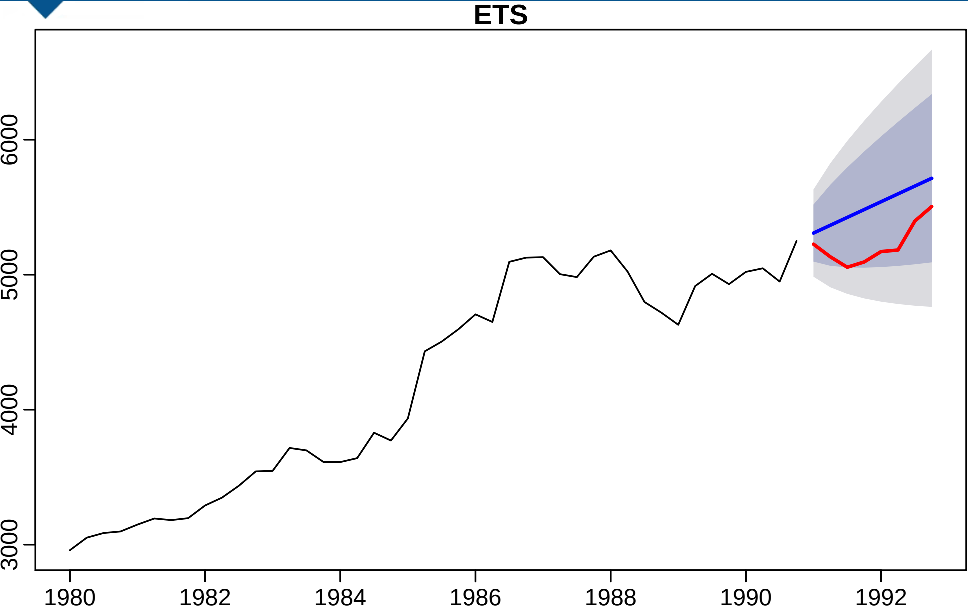
Predictability



Predictability



Predictability



Predictability

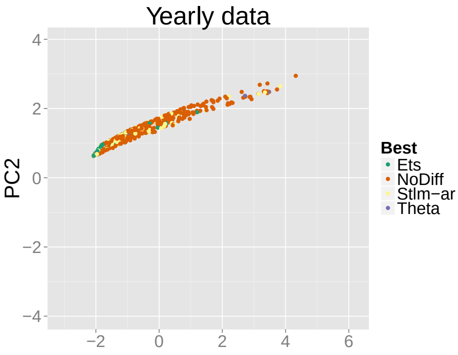
Predictability

Medium
MASE values

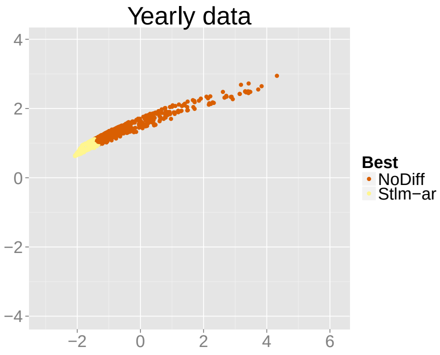
Predictability

High
MASE values

Predictability

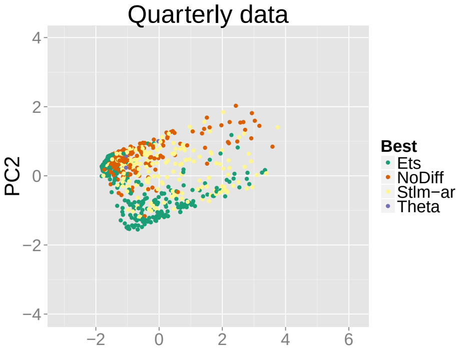


Actual

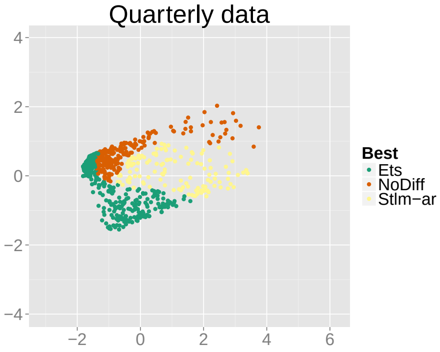


SVM prediction

Predictability

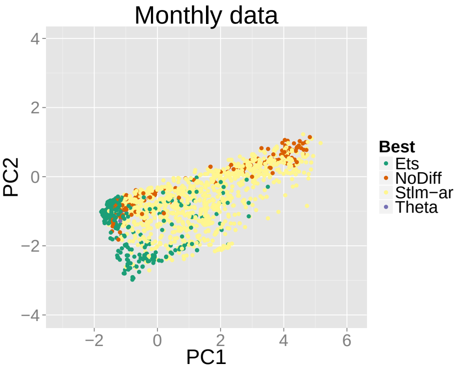


Actual

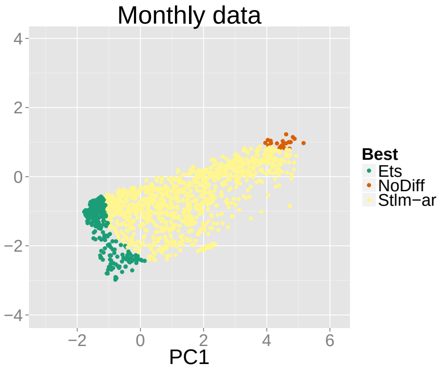


SVM prediction

Predictability



Actual



SVM prediction

Generating new time series

We can use the feature space to:

- Generate new time series with similar features to existing series
- Generate new time series where there are "holes" in the feature space.

- Let $\{\mathrm{PC}_1, \mathrm{PC}_2, \ldots, \mathrm{PC}_n\}$ be a "population" of time series of specified length and period.
- Genetic algorithm uses a process of selection, crossover and mutation to evolve the population towards a target point T_i.
- Optimize: Fitness $(\mathrm{PC}_j) = -\sqrt{(|\mathrm{PC}_j - T_i|^2)}$.
- Initial population random with some series in neighbourhood of T_i.

Generating new time series

We can use the feature space to:

- Generate new time series with similar features to existing series
- Generate new time series where there are "holes" in the feature space.

- Let $\{\mathrm{PC}_1, \mathrm{PC}_2, \ldots, \mathrm{PC}_n\}$ be a "population" of time series of specified length and period.
- Genetic algorithm uses a process of selection, crossover and mutation to evolve the population towards a target point T_i.
- Optimize: Fitness $(\mathrm{PC}_j) = -\sqrt{(|\mathrm{PC}_j - T_i|^2)}$.
- Initial population random with some series in neighbourhood of T_i.

Generating new time series

We can use the feature space to:

- Generate new time series with similar features to existing series
- Generate new time series where there are "holes" in the feature space.

- Let $\{\mathrm{PC}_1, \mathrm{PC}_2, \ldots, \mathrm{PC}_n\}$ be a "population" of time series of specified length and period.
- Genetic algorithm uses a process of selection, crossover and mutation to evolve the population towards a target point T_i.
- Optimize: Fitness $(\mathrm{PC}_j) = -\sqrt{(|\mathrm{PC}_j - T_i|^2)}$.
- Initial population random with some series in neighbourhood of T_i.

Generating new time series

We can use the feature space to:

- Generate new time series with similar features to existing series
- Generate new time series where there are "holes" in the feature space.

- Let $\{\mathrm{PC}_1, \mathrm{PC}_2, \ldots, \mathrm{PC}_n\}$ be a "population" of time series of specified length and period.
- Genetic algorithm uses a process of selection, crossover and mutation to evolve the population towards a target point T_i.
- Optimize: Fitness $(\mathrm{PC}_j) = -\sqrt{(|\mathrm{PC}_j - T_i|^2)}$.
- Initial population random with some series in neighbourhood of T_i.

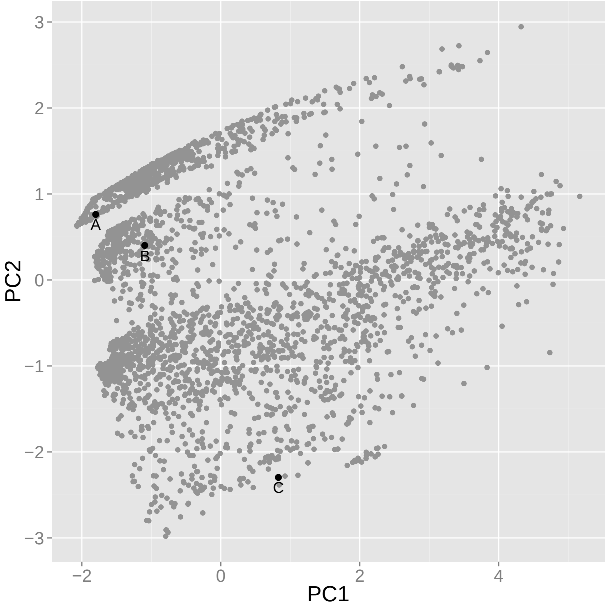
Generating new time series

We can use the feature space to:

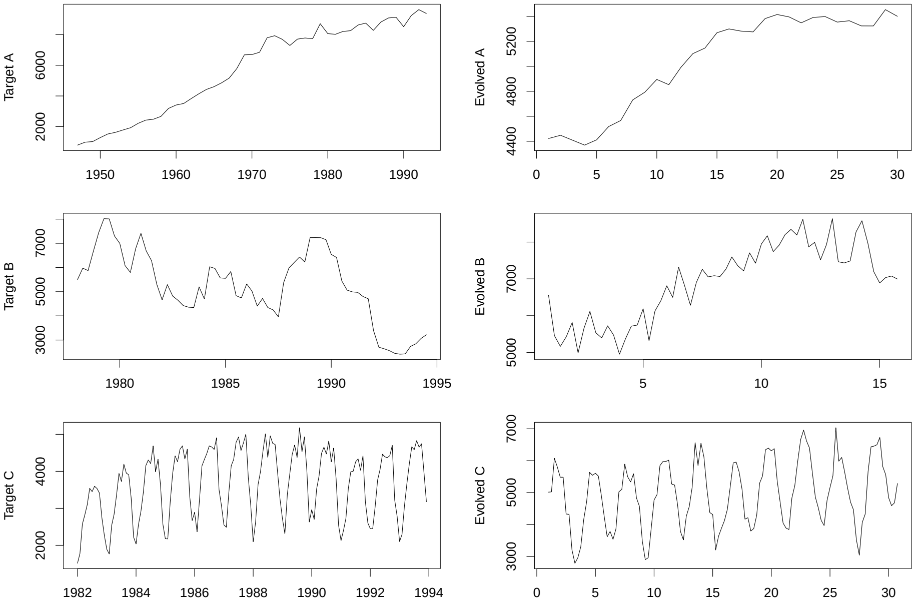
- Generate new time series with similar features to existing series
- Generate new time series where there are "holes" in the feature space.

- Let $\{\mathrm{PC}_1, \mathrm{PC}_2, \ldots, \mathrm{PC}_n\}$ be a "population" of time series of specified length and period.
- Genetic algorithm uses a process of selection, crossover and mutation to evolve the population towards a target point T_i.
- Optimize: Fitness $(\mathrm{PC}_j) = -\sqrt{(|\mathrm{PC}_j - T_i|^2)}$.
- Initial population random with some series in neighbourhood of T_i.

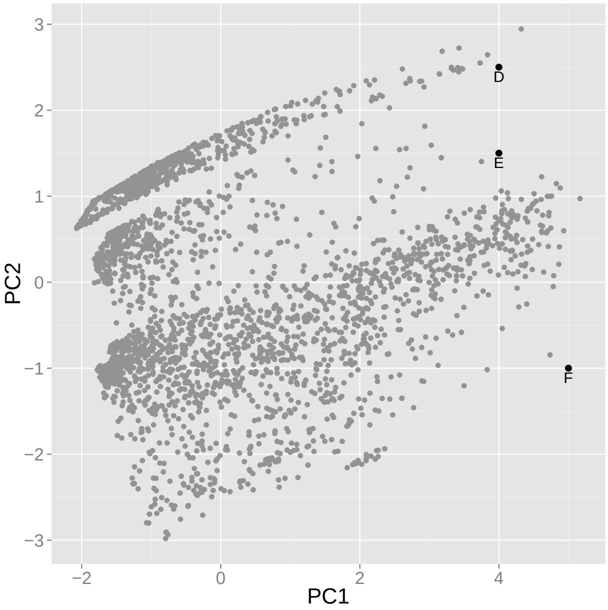
Evolving new time series



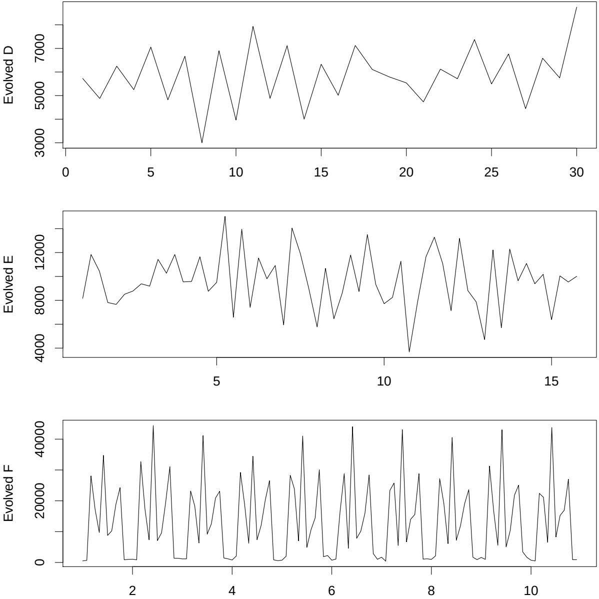
Evolving new time series



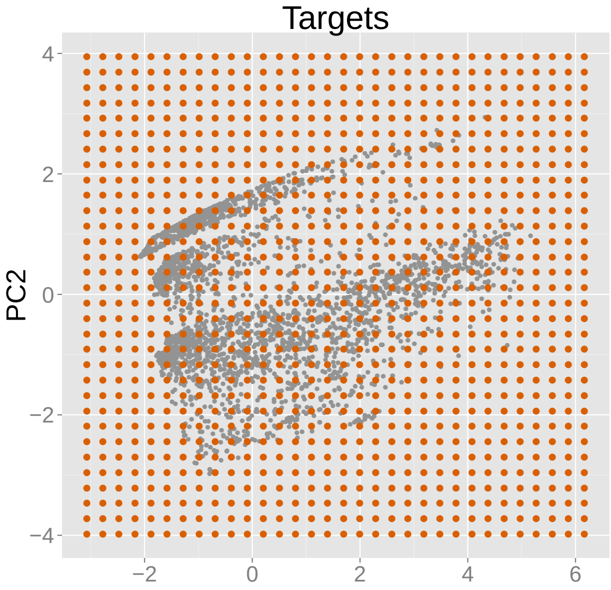
Evolving new time series



Evolving new time series

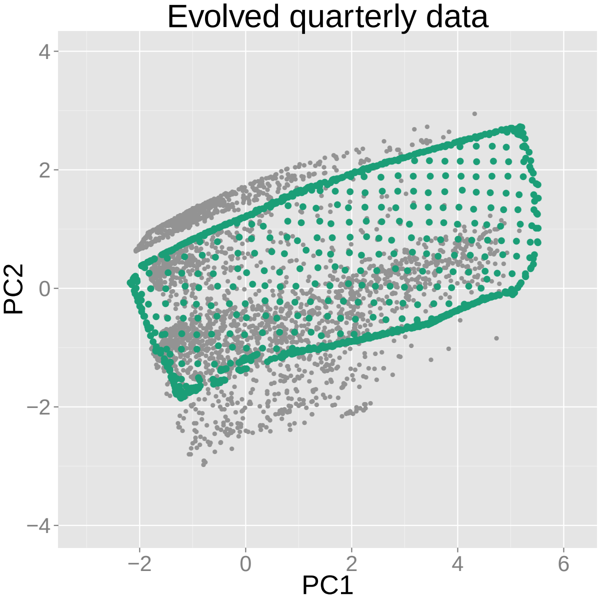


Evolving new time series

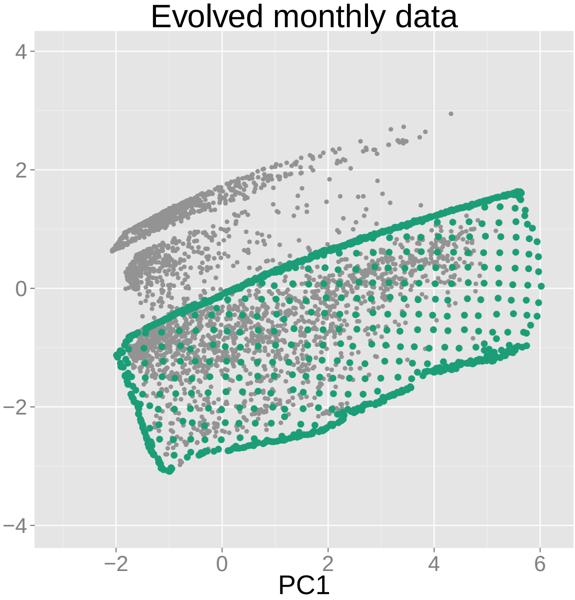


Evolving new time series

Evolving new time series



Evolving new time series



Questions raised

- Can SVM be used to create a forecast selection routine to give better forecasts?
- How much do M3 conclusions depend on the particular set of time series involved?
- Has the M3 data set biased forecast method development?
- What other features should we consider? What difference does it make?
- Is PCA the right approach? Perhaps we should use multidimensional scaling? Or something else?
- Should we use more than 2 PC dimensions?

Questions raised

- Can SVM be used to create a forecast selection routine to give better forecasts?
- How much do M3 conclusions depend on the particular set of time series involved?
- Has the M3 data set biased forecast method development?
- What other features should we consider? What difference does it make?
- Is PCA the right approach? Perhaps we should use multidimensional scaling? Or something else?
- Should we use more than 2 PC dimensions?

Questions raised

- Can SVM be used to create a forecast selection routine to give better forecasts?
- How much do M3 conclusions depend on the particular set of time series involved?
- Has the M3 data set biased forecast method development?
- What other features should we consider? What difference does it make?
- Is PCA the right approach? Perhaps we should use multidimensional scaling? Or something else?
- Should we use more than 2 PC dimensions?

Questions raised

- Can SVM be used to create a forecast selection routine to give better forecasts?
- How much do M3 conclusions depend on the particular set of time series involved?
- Has the M3 data set biased forecast method development?
- What other features should we consider? What difference does it make?
- Is PCA the right approach? Perhaps we should use multidimensional scaling? Or something else?
- Should we use more than 2 PC dimensions?

Questions raised

- Can SVM be used to create a forecast selection routine to give better forecasts?
- How much do M3 conclusions depend on the particular set of time series involved?
- Has the M3 data set biased forecast method development?
- What other features should we consider? What difference does it make?
- Is PCA the right approach? Perhaps we should use multidimensional scaling? Or something else?
- Should we use more than 2 PC dimensions?

Questions raised

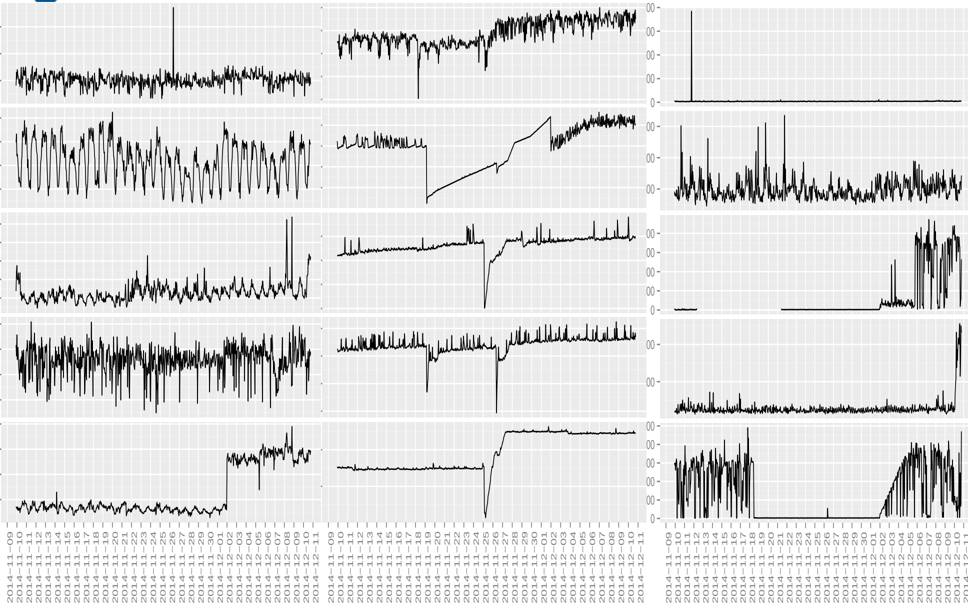
- Can SVM be used to create a forecast selection routine to give better forecasts?
- How much do M3 conclusions depend on the particular set of time series involved?
- Has the M3 data set biased forecast method development?
- What other features should we consider? What difference does it make?
- Is PCA the right approach? Perhaps we should use multidimensional scaling? Or something else?
- Should we use more than 2 PC dimensions?

Outline

Yahoo web-traffic

- Tens of thousands of time series collected at one-hour intervals over one month.
- Consisting of several server metrics (e.g. CPU usage and paging views) from many server farms globally.
- Aim: find unusual (anomalous) time series.

Yahoo web-traffic



Feature space

- **ACF1**: first order autocorrelation = Corr(Y_t, Y_{t-1})
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = $\text{Corr}(Y_t, Y_{t-1})$
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = $\text{Corr}(Y_t, Y_{t-1})$
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = Corr(Y_t, Y_{t-1})
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x)dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = Corr(Y_t, Y_{t-1})
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x)dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = Corr(Y_t, Y_{t-1})
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = Corr(Y_t, Y_{t-1})
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = $\mathrm{Corr}(Y_t, Y_{t-1})$
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = $\text{Corr}(Y_t, Y_{t-1})$
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x)dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = $\text{Corr}(Y_t, Y_{t-1})$
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = $\text{Corr}(Y_t, Y_{t-1})$
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

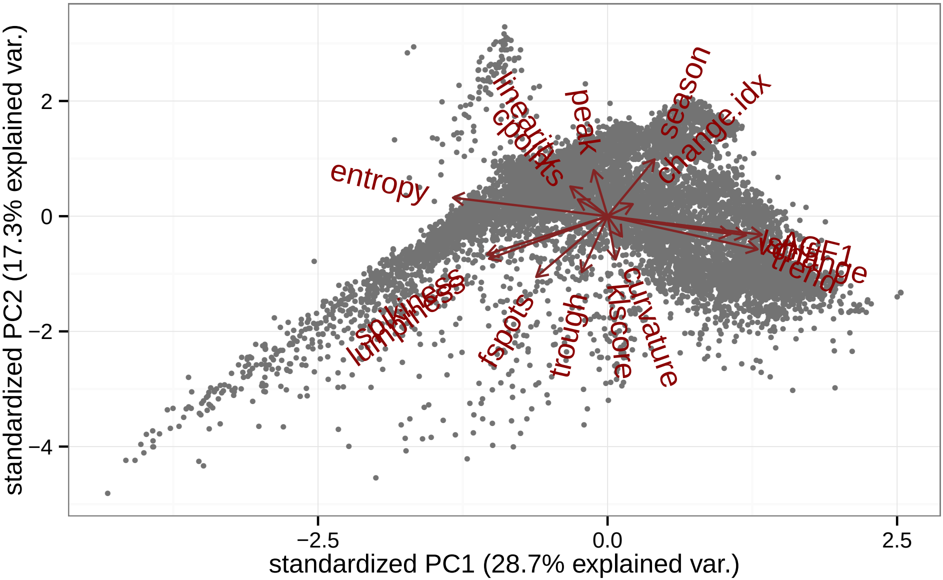
Feature space

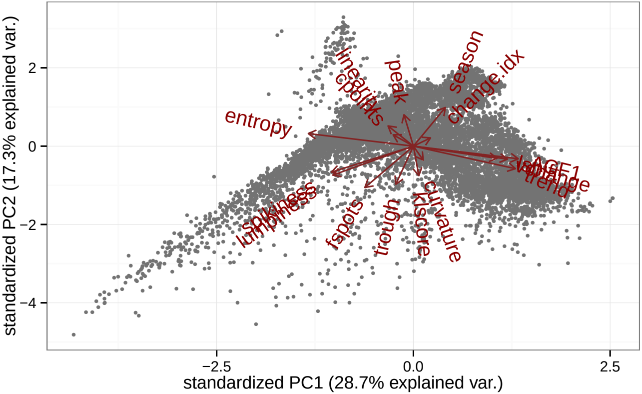
- **ACF1**: first order autocorrelation = $\mathrm{Corr}(Y_t, Y_{t-1})$
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Feature space

- **ACF1**: first order autocorrelation = $\mathrm{Corr}(Y_t, Y_{t-1})$
- Strength of **trend** and **seasonality** based on STL
- Trend **linearity** and **curvature**
- Size of seasonal **peak** and **trough**
- Spectral **entropy**
- **Lumpiness**: variance of block variances (block size 24).
- **Spikiness**: variances of leave-one-out variances of STL remainders.
- **Level shift**: Maximum difference in trimmed means of consecutive moving windows of size 24.
- **Variance change**: Max difference in variances of consecutive moving windows of size 24.
- **Flat spots**: Discretize sample space into 10 equal-sized intervals. Find max run length in any interval.
- Number of **crossing points** of mean line.
- **Kullback-Leibler score**: Maximum of $D_{KL}(P\|Q) = \int P(x) \ln P(x)/Q(x) dx$ where P and Q are estimated by kernel density estimators applied to consecutive windows of size 48.
- **Change index**: Time of maximum KL score

Principal component analysis

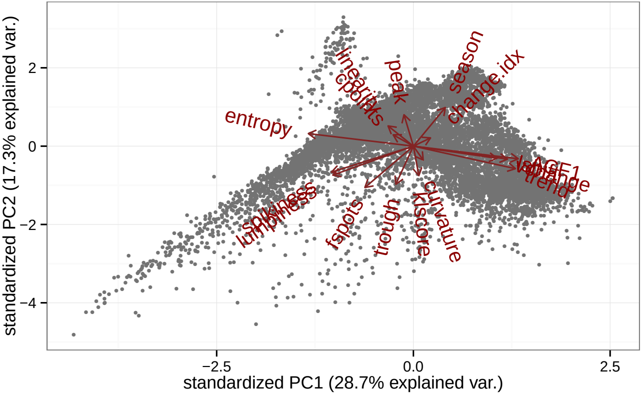




We need a measure of the “anomalousness” of a time series.

1. Rank points based on their local density.
2. Rank points based on whether they are within α-convex hulls of different radius.

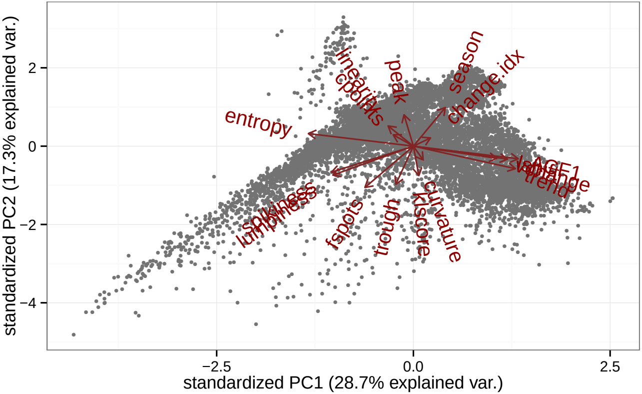
What is “anomalous”



We need a measure of the “anomalousness” of a time series.

1. Rank points based on their local density.
2. Rank points based on whether they are within α-convex hulls of different radius.

What is “anomalous”



We need a measure of the “anomalousness” of a time series.

1. Rank points based on their local density.
2. Rank points based on whether they are within α-convex hulls of different radius.

Bivariate kernel density

$$\hat{f}(\mathbf{x};\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x} - \mathbf{X}_i)$$

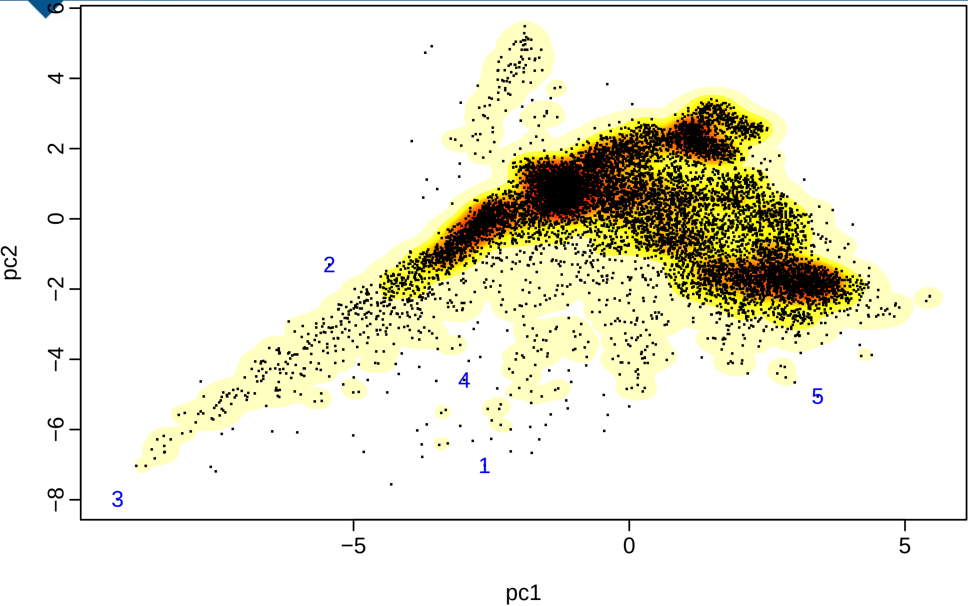
- $\mathbf{X}_i \in$ a bivariate random sample $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$
- $K_{\mathbf{H}}(\mathbf{x})$ is the standard normal kernel function
- $\mathbf{H}$ estimated by minimizing the sum of AMISE
- Rank points based on $\hat{f}$ values in 2d PCA space.

Bivariate kernel density

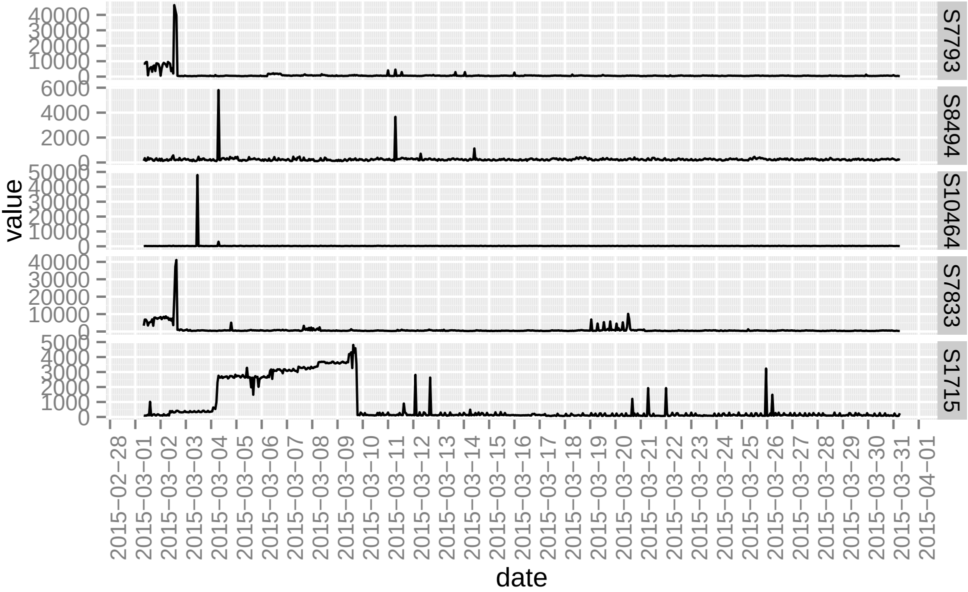
$$\hat{f}(\mathbf{x};\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x}-\mathbf{X}_i)$$

- $\mathbf{X}_i \in$ a bivariate random sample $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$
- $K_{\mathbf{H}}(\mathbf{x})$ is the standard normal kernel function
- $\mathbf{H}$ estimated by minimizing the sum of AMISE
- Rank points based on $\hat{f}$ values in 2d PCA space.

Bivariate density ranking

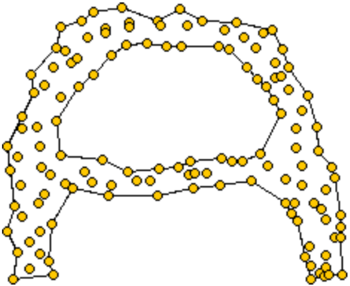


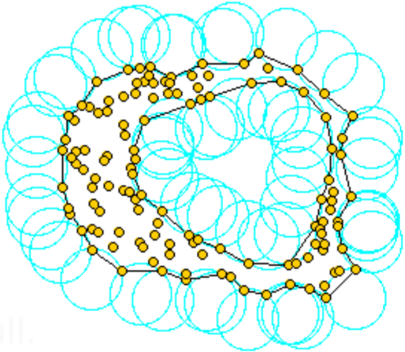
Bivariate density ranking



α-convex hulls

The space generated by point pairs that can be touched by an empty disc of radius α.

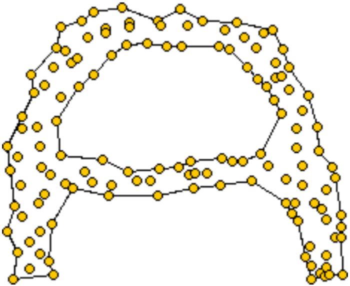


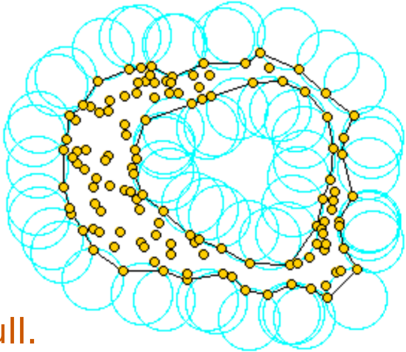


- $\alpha \to \infty$ gives a convex hull.
- Points can become isolated when α is small.
- We rank points based on the value of α when they become isolated.

α-convex hulls

The space generated by point pairs that can be touched by an empty disc of radius α.

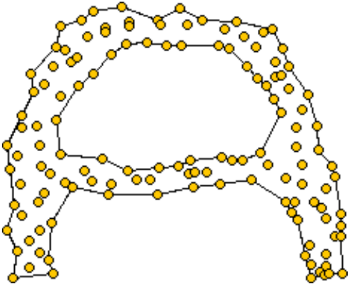


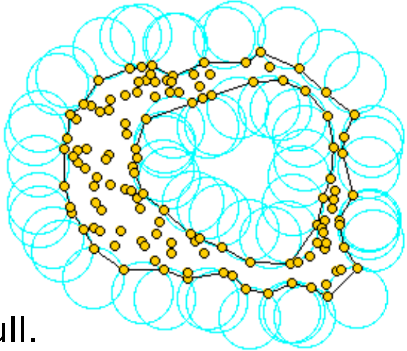


- $\alpha \to \infty$ gives a convex hull.
- Points can become isolated when α is small.
- We rank points based on the value of α when they become isolated.

α-convex hulls

The space generated by point pairs that can be touched by an empty disc of radius α.

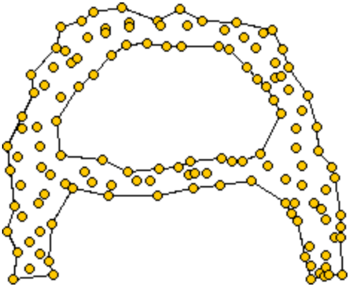


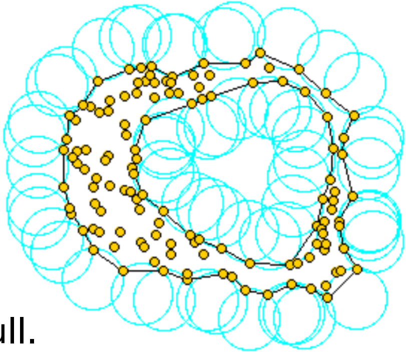


- $\alpha \to \infty$ gives a convex hull.
- Points can become isolated when α is small.
- We rank points based on the value of α when they become isolated.

α-convex hulls

The space generated by point pairs that can be touched by an empty disc of radius α.

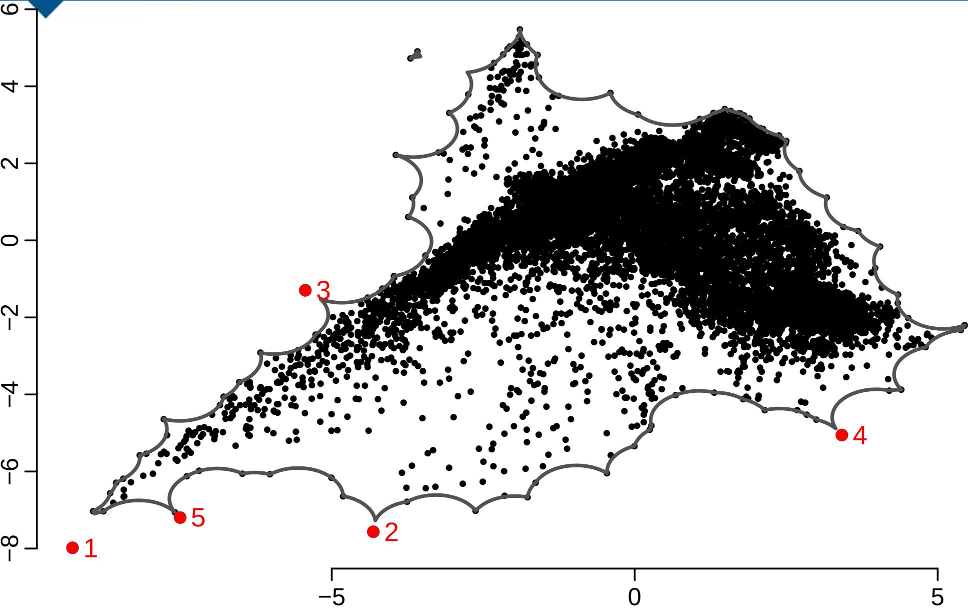




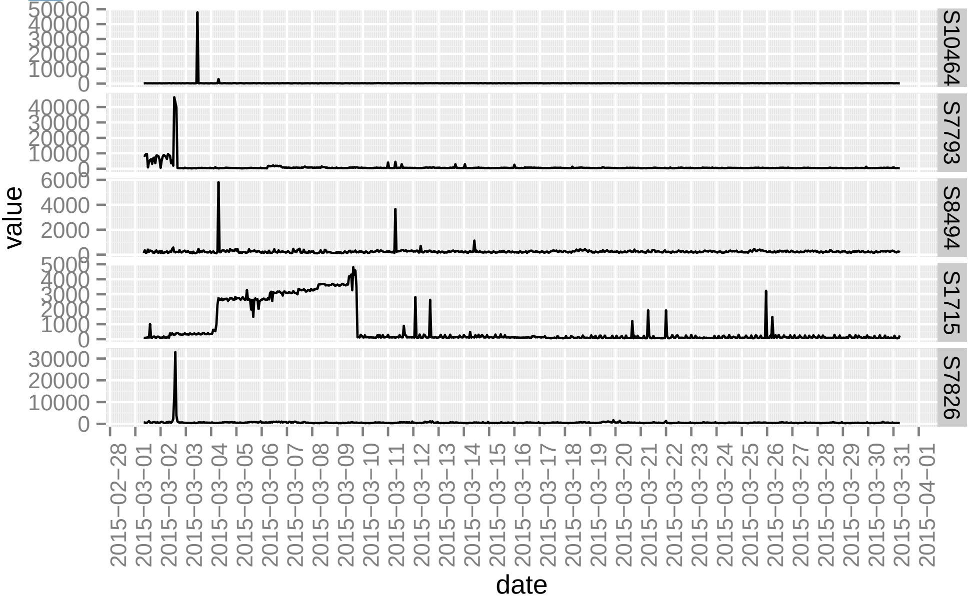
- $\alpha \to \infty$ gives a convex hull.
- Points can become isolated when α is small.
- We rank points based on the value of α when they become isolated.

α-convex hull

α-convex hull ranking

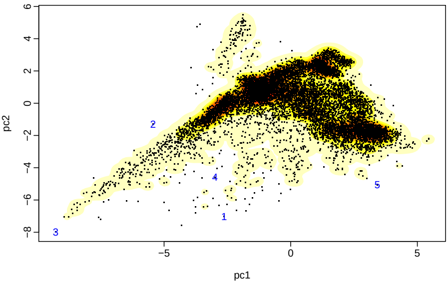


α-convex hull ranking

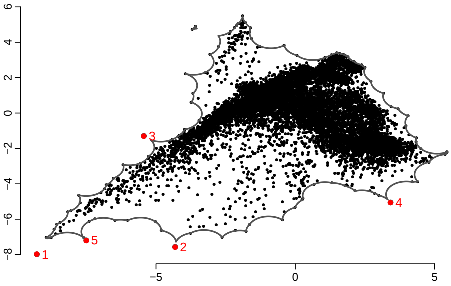


HDR versus α-convex hull

HDR boxplot

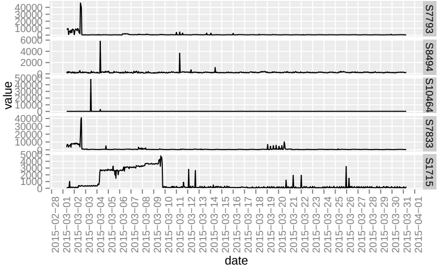


α-convex hull

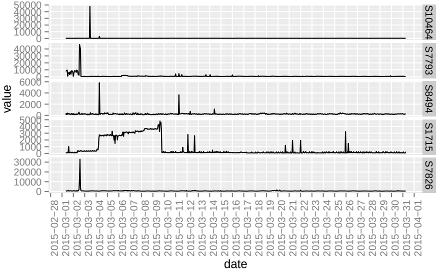


Top 5 anomalous time series

HDR



α-convex hull



Outline

What next?

- Develop a more comprehensive set of features that are reliable measures and fast to compute. e.g., for finance data.
- Consider other dimension reduction methods and more than 2 dimensions.
- Develop dynamic and interactive visualization tools.
- Make methods available in an R package.

What next?

- Develop a more comprehensive set of features that are reliable measures and fast to compute. e.g., for finance data.
- Consider other dimension reduction methods and more than 2 dimensions.
- Develop dynamic and interactive visualization tools.
- Make methods available in an R package.
  - Some of the methods are already part of the anomalous package for R on Github.

What next?

- Develop a more comprehensive set of features that are reliable measures and fast to compute. e.g., for finance data.
- Consider other dimension reduction methods and more than 2 dimensions.
- Develop dynamic and interactive visualization tools.
- Make methods available in an R package.
  - Some of the methods are already available in the **anomalous** package for R on github.

What next?

- Develop a more comprehensive set of features that are reliable measures and fast to compute. e.g., for finance data.
- Consider other dimension reduction methods and more than 2 dimensions.
- Develop dynamic and interactive visualization tools.
- Make methods available in an R package.
    - Some of the methods are already available in the **anomalous** package for R on github.

What next?

- Develop a more comprehensive set of features that are reliable measures and fast to compute. e.g., for finance data.
- Consider other dimension reduction methods and more than 2 dimensions.
- Develop dynamic and interactive visualization tools.
- Make methods available in an R package.
  - Some of the methods are already available in the **anomalous** package for R on github.

Further information

- Papers and R packages: **robjhyndman.com**
- Blog: **robjhyndman.com/hyndsight**
- Code: **github.com/robjhyndman**
- Email: **Rob.Hyndman@monash.edu**