MONASH University

MONASH BUSINESS SCHOOL

# Feature-based time series analysis

Rob J Hyndman

21 June 2018

# Outline

1. Time series feature spaces
2. Irish smart metre data
3. Quantiles conditional on time of week
4. Finding typical and unusual households
5. Visualization via embedding
6. Features and limitations

# M3 competition

ELSEVIER

International Journal of Forecasting 16 (2000) 451–476

international journal of forecasting

www.elsevier.com/locate/ijforecast

## The M3-Competition: results, conclusions and implications

Spyros Makridakis, Michèle Hibon*

*INSEAD, Boulevard de Constance, 77305 Fontainebleau, France*

**Abstract**

This paper describes the M3-Competition, the latest of the M-Competitions. It explains the reasons for conducting the competition and summarizes its results and conclusions. In addition, the paper compares such results/conclusions with those of the previous two M-Competitions as well as with those of other major empirical studies. Finally, the implications of these results and conclusions are considered, their consequences for both the theory and practice of forecasting are explored and directions for future research are contemplated. 

*Keywords:* Comparative methods — time series: univariate; Forecasting competitions; M-Competition; Forecasting methods, Forecasting accuracy

# M3 competition

ELSEVIER

International Journal of Forecasting 16 (2000) 451–476

international journal of forecasting

www.elsevier.com/locate/ijforecast


## etition: results, conclusions ons

Spyros Makridakis, Michèle Hibon

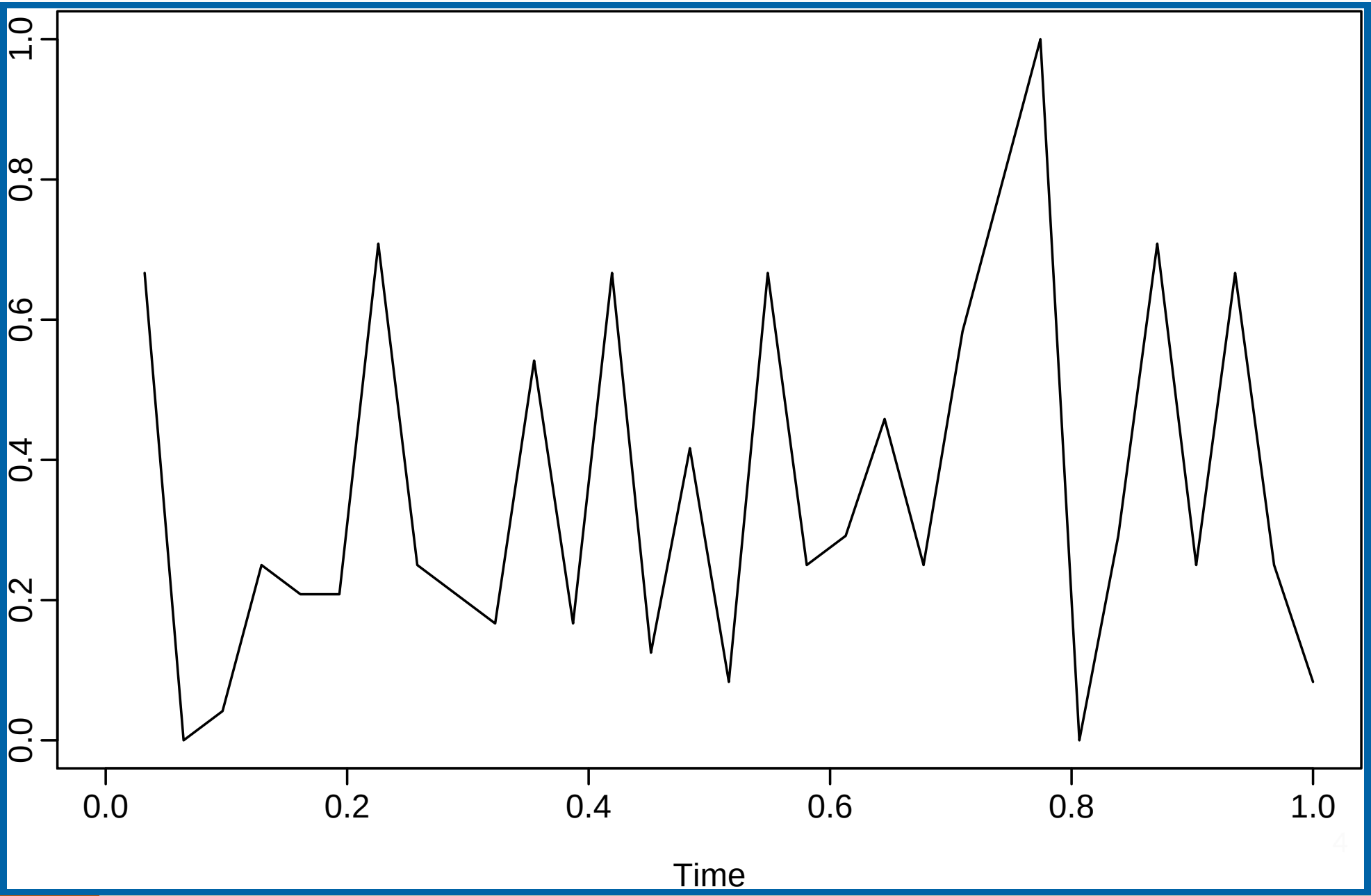
*EAD, Boulevard de Constance, 77305 Fontainebleau*

**Abstr**

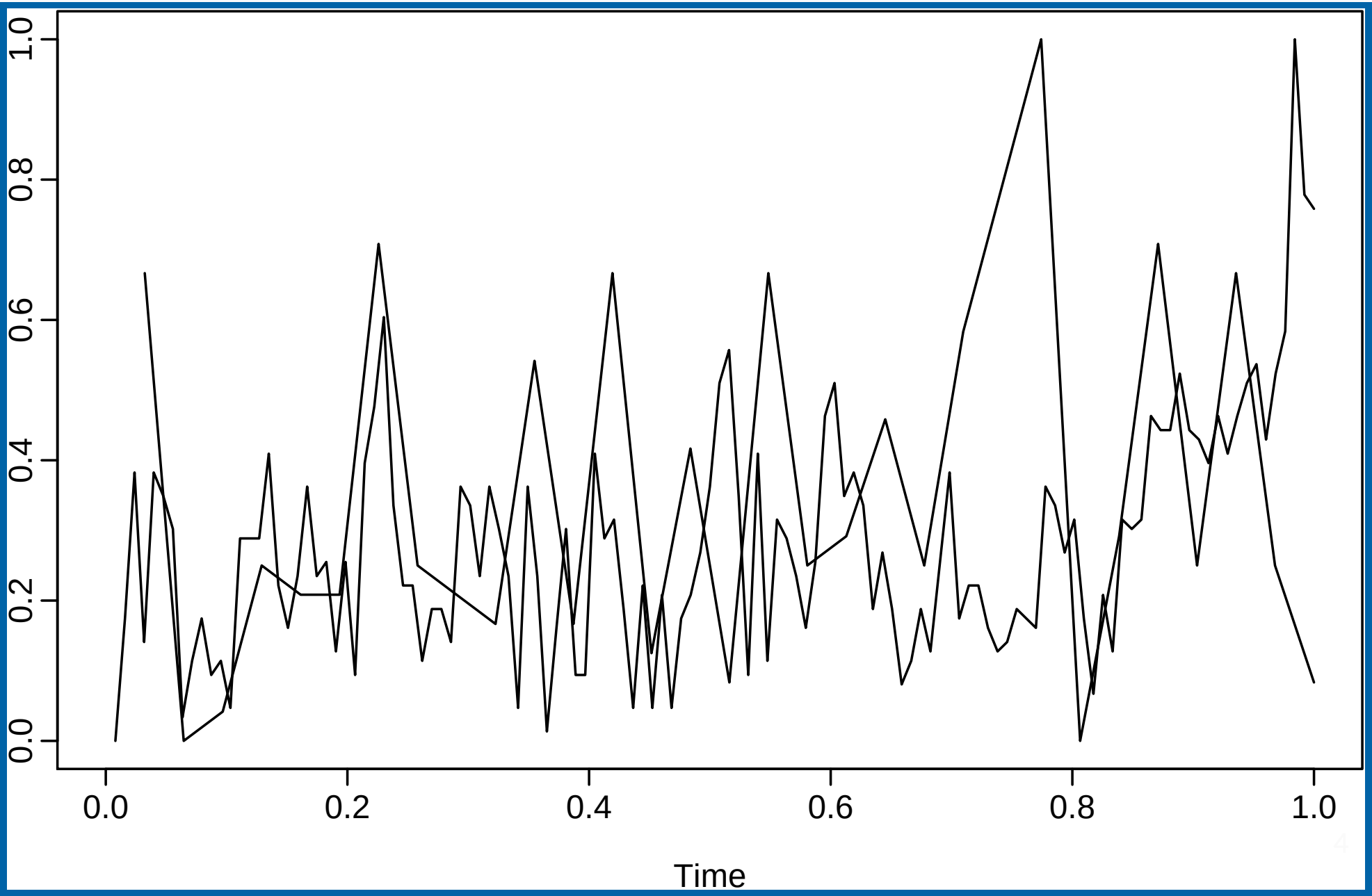
This paper describes the M3-Competition, the latest of the M-Competitions. It explains the reasons for conducting the competition and summarizes its results and conclusions. In addition, the paper compares such results/conclusions with those of the previous two M-Competitions as well as with those of other major empirical studies. Finally, the implications of these results and conclusions are considered, their consequences for both the theory and practice of forecasting are explored and directions for future research are contemplated. 

*Keywords:* Comparative methods — time series: univariate; Forecasting competitions; M-Competition; Forecasting methods, Forecasting accuracy

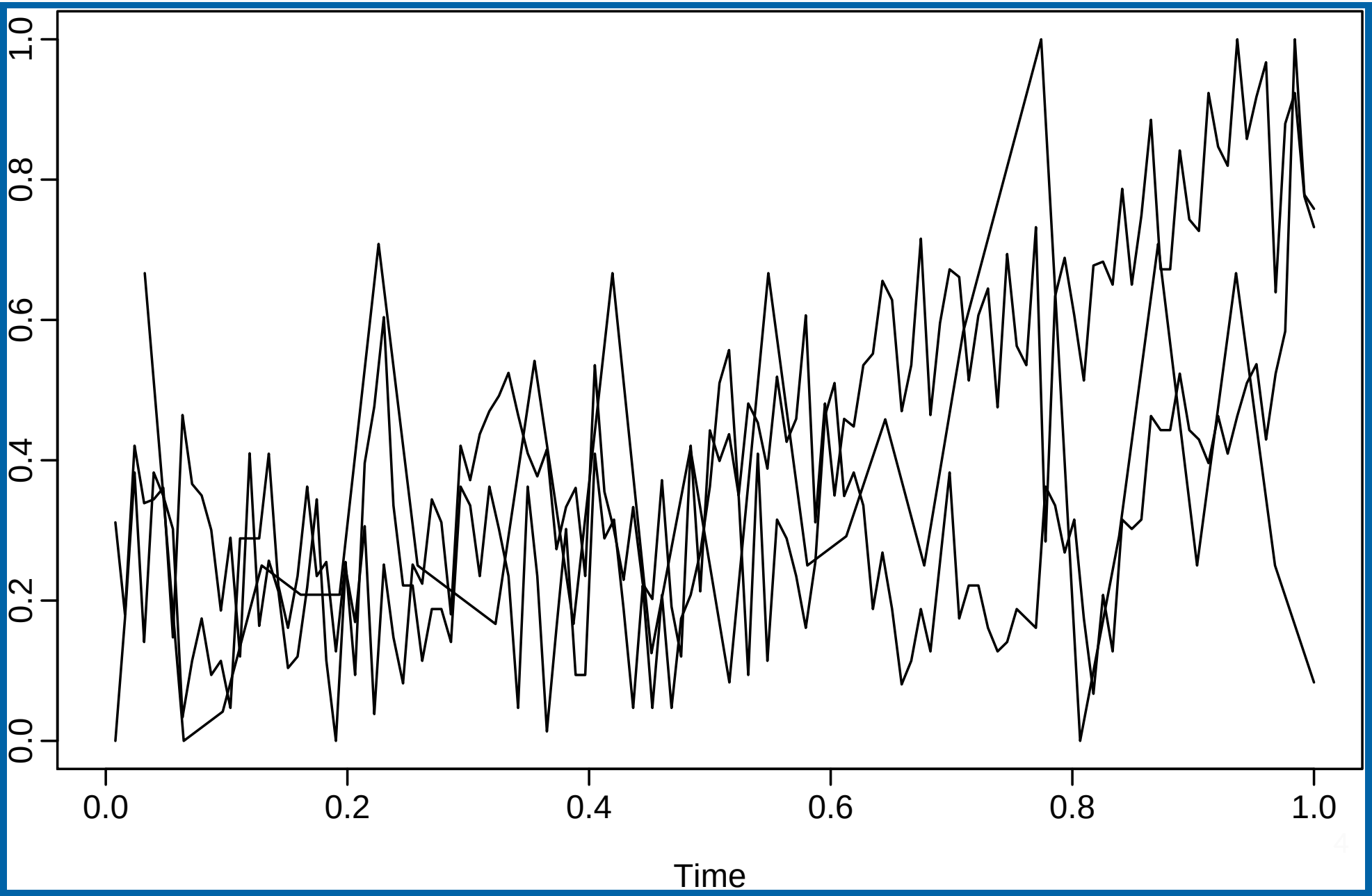
# How to plot lots of time series?



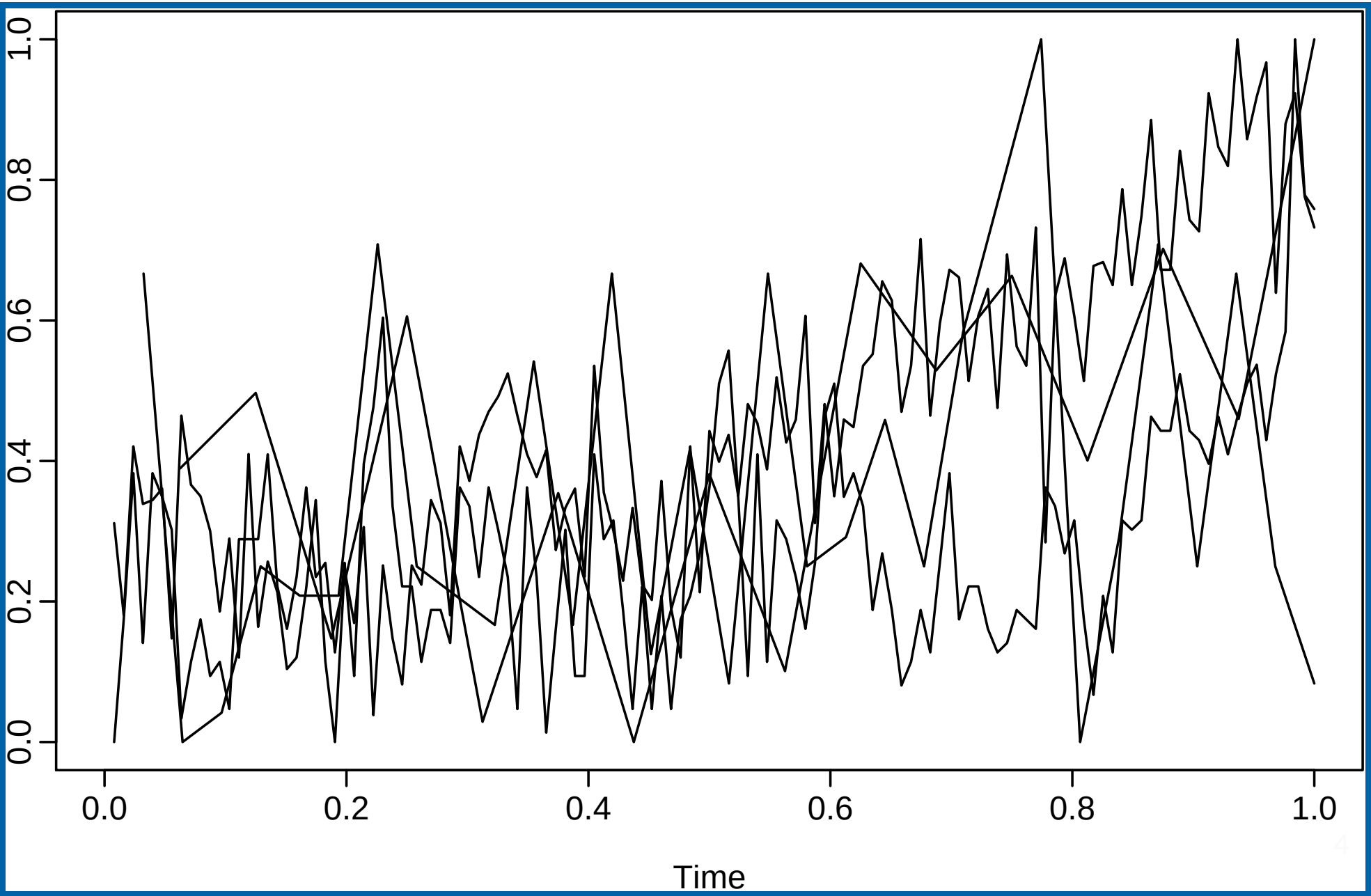
# How to plot lots of time series?



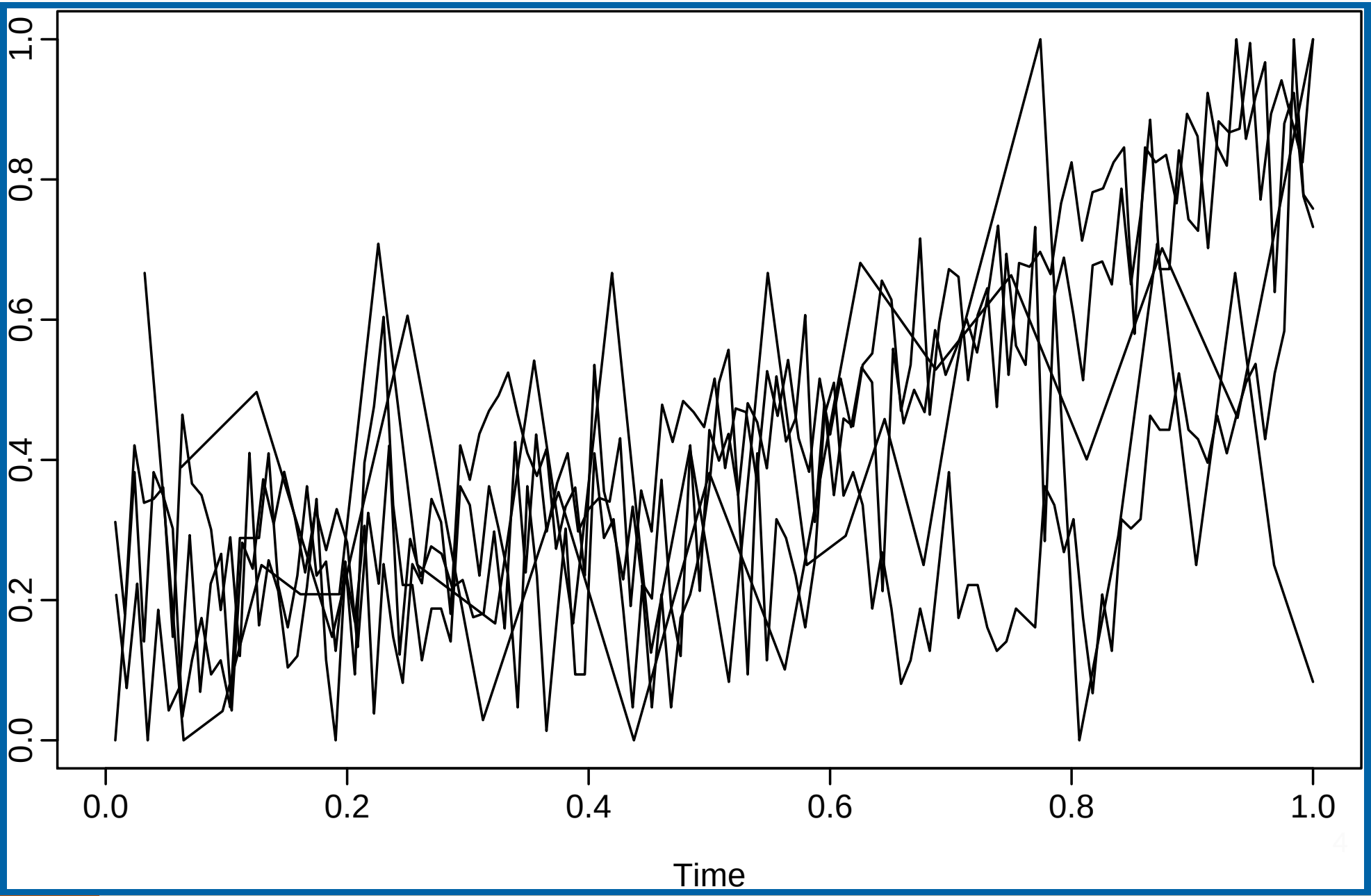
# How to plot lots of time series?



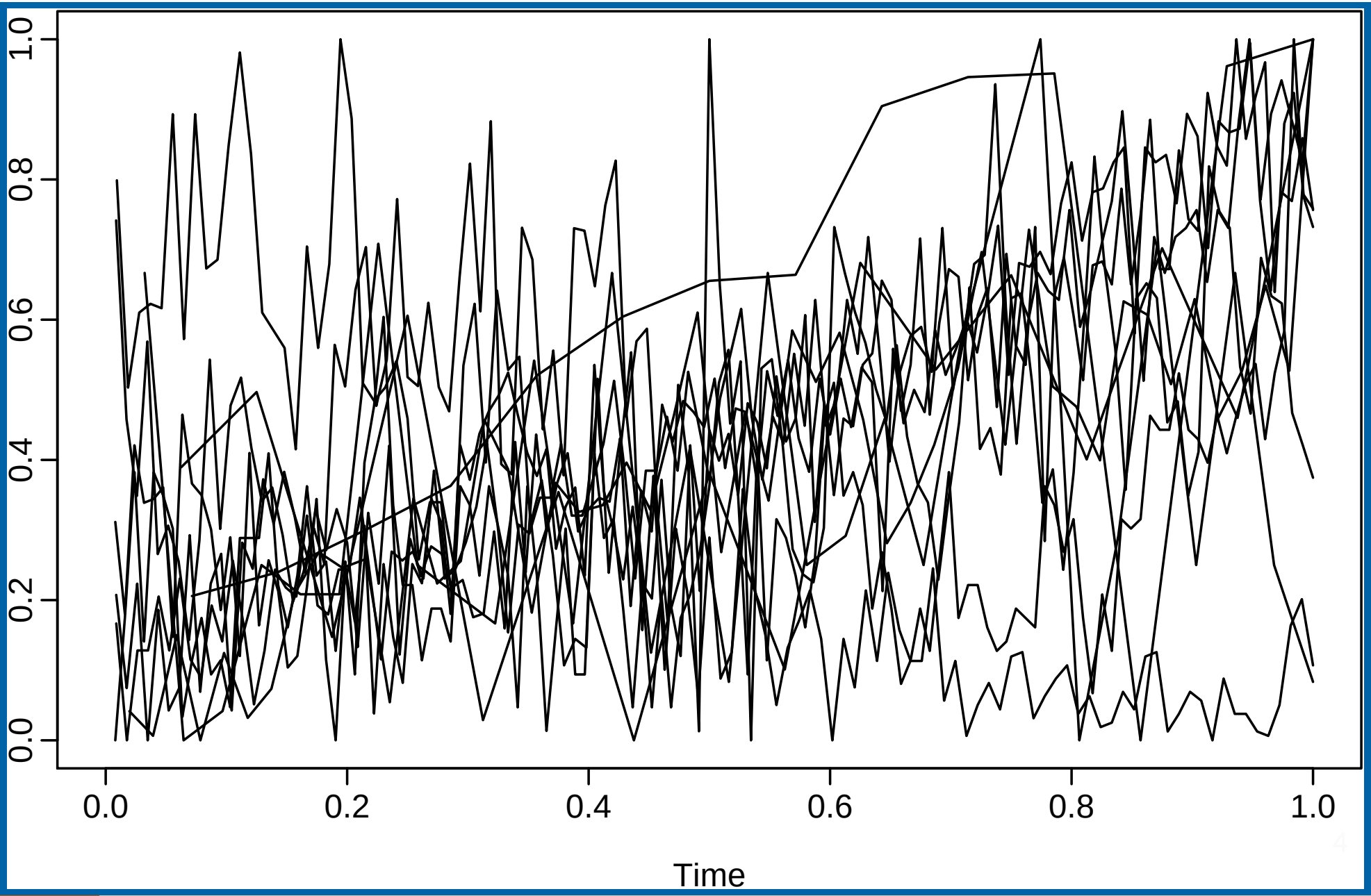
# How to plot lots of time series?



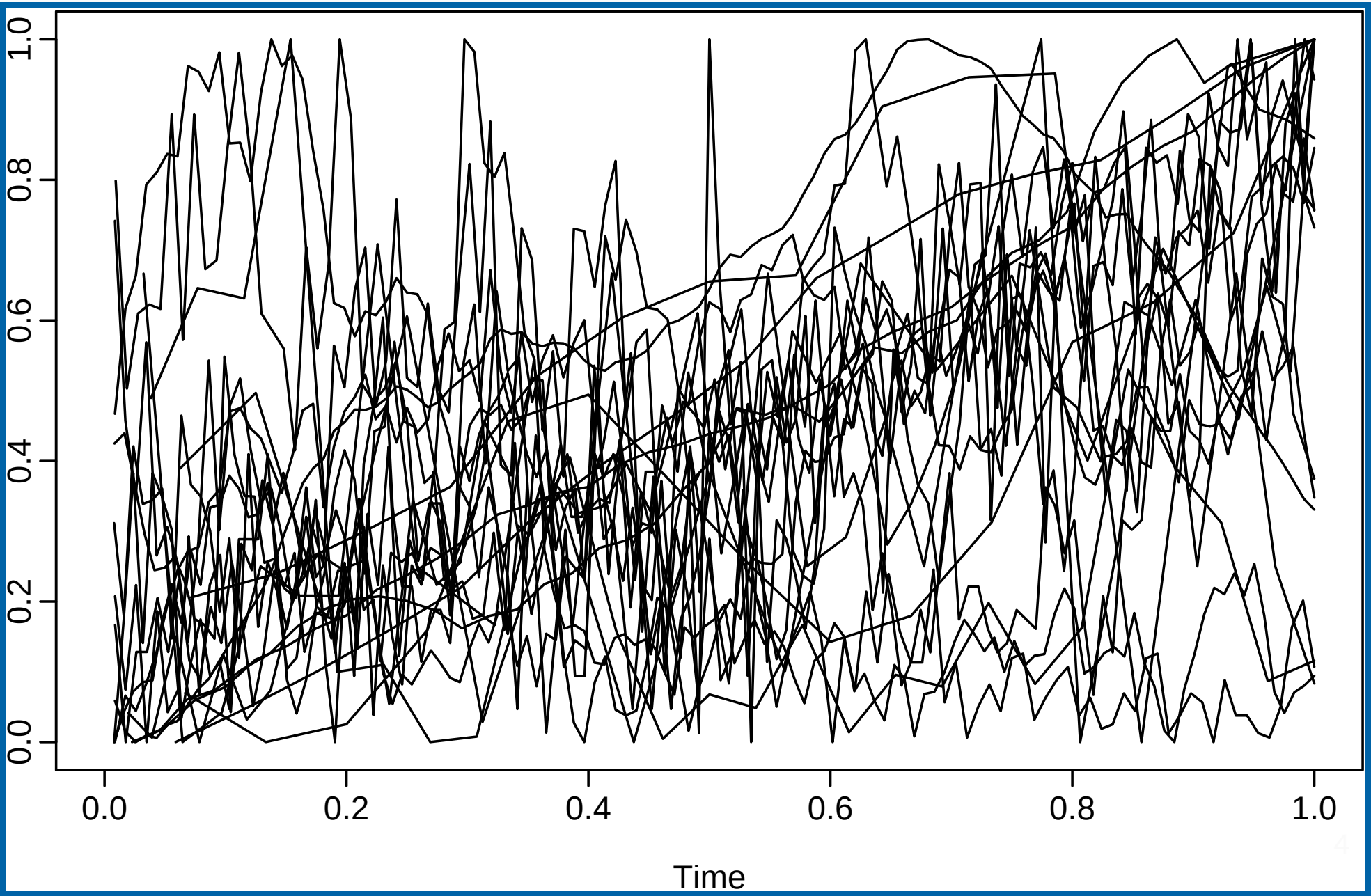
# How to plot lots of time series?



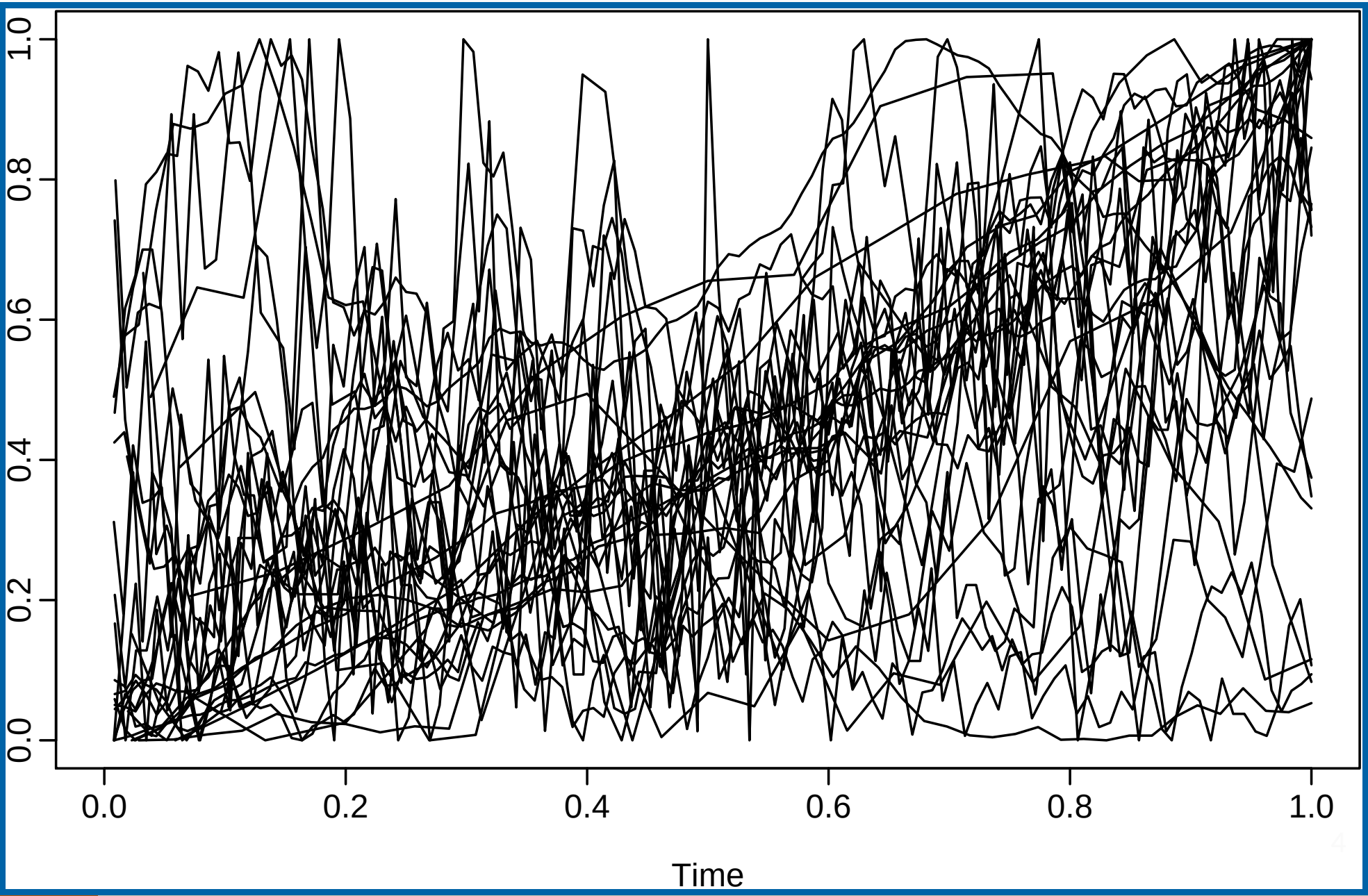
# How to plot lots of time series?



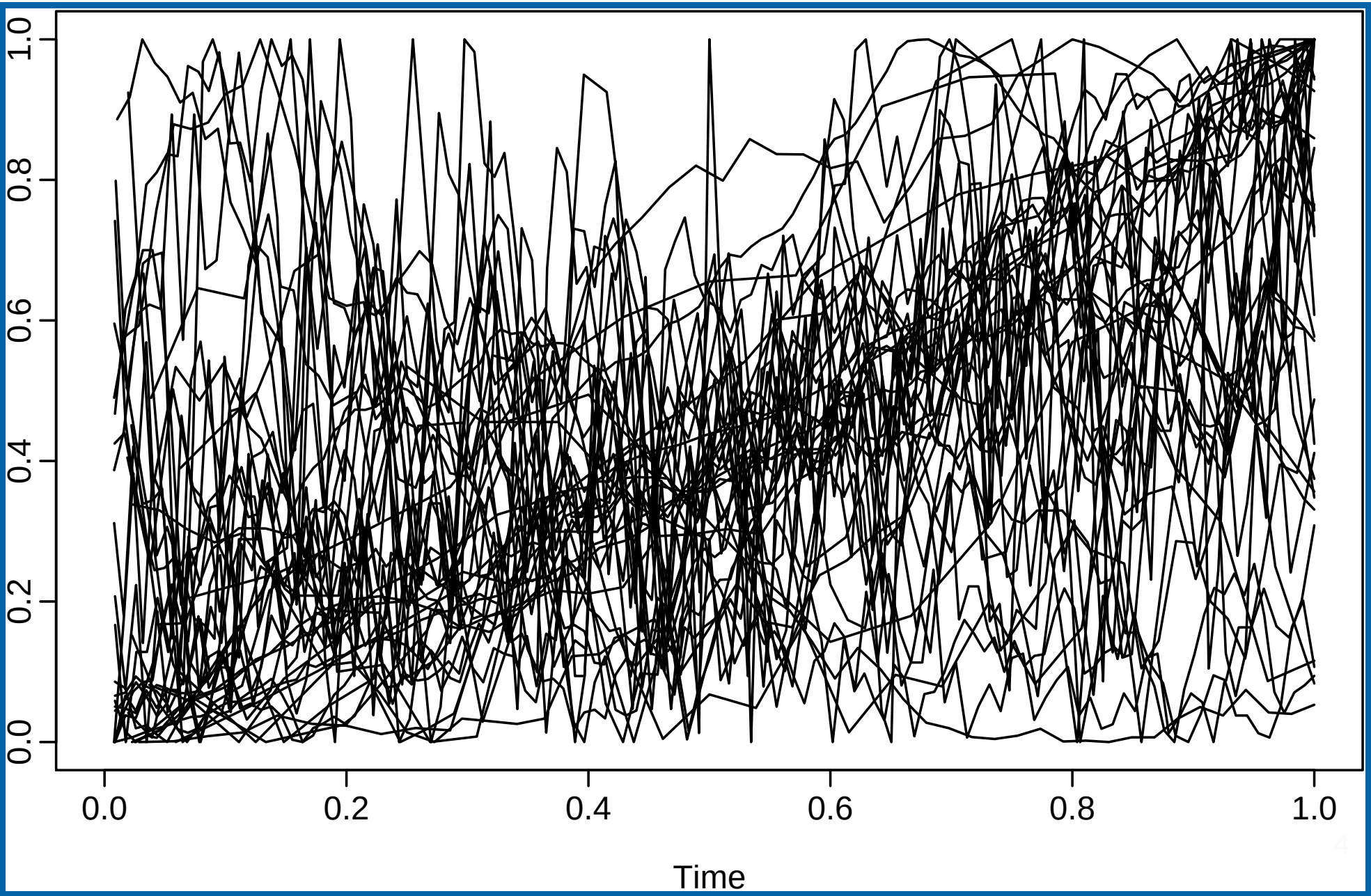
# How to plot lots of time series?



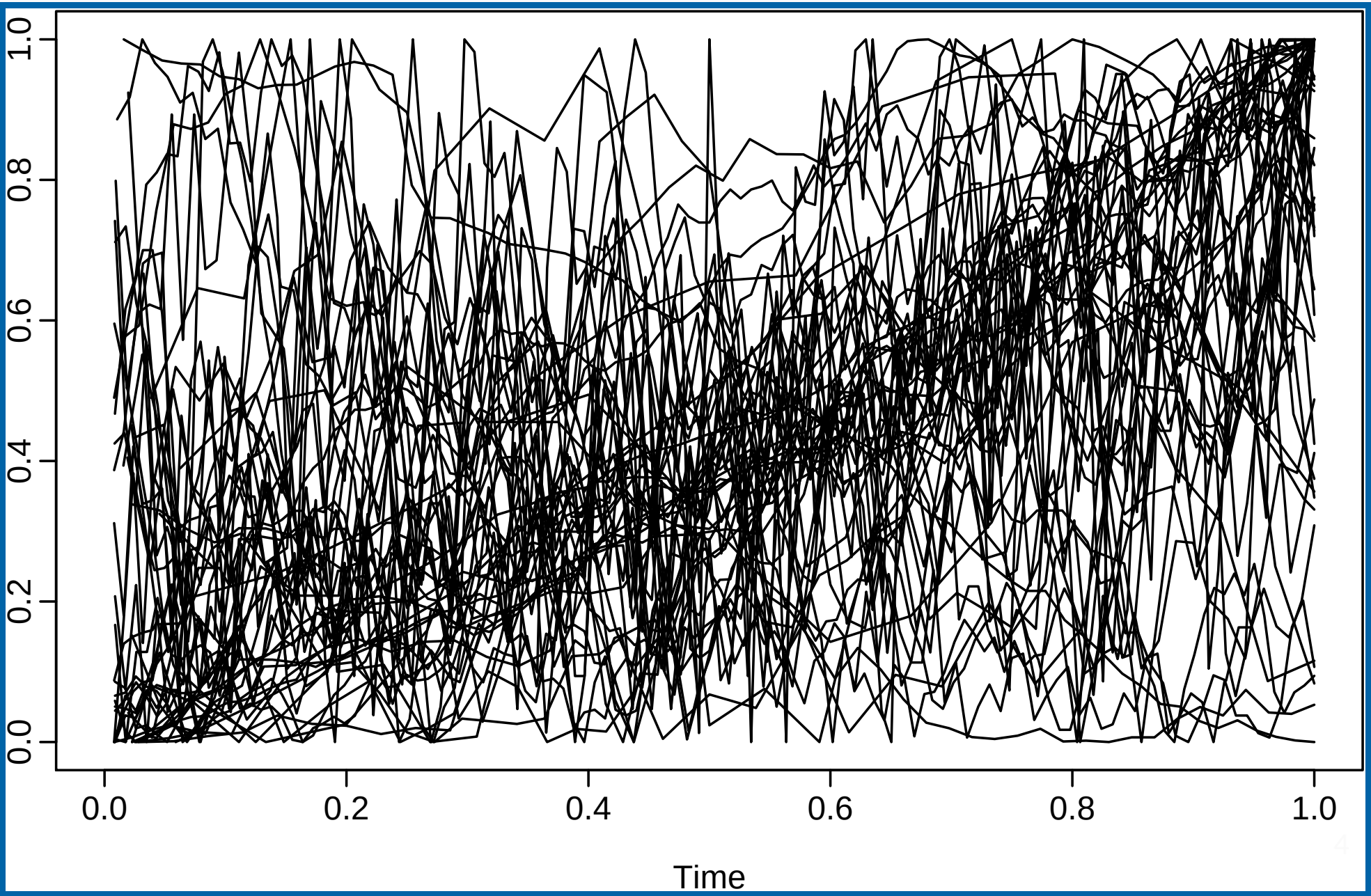
# How to plot lots of time series?



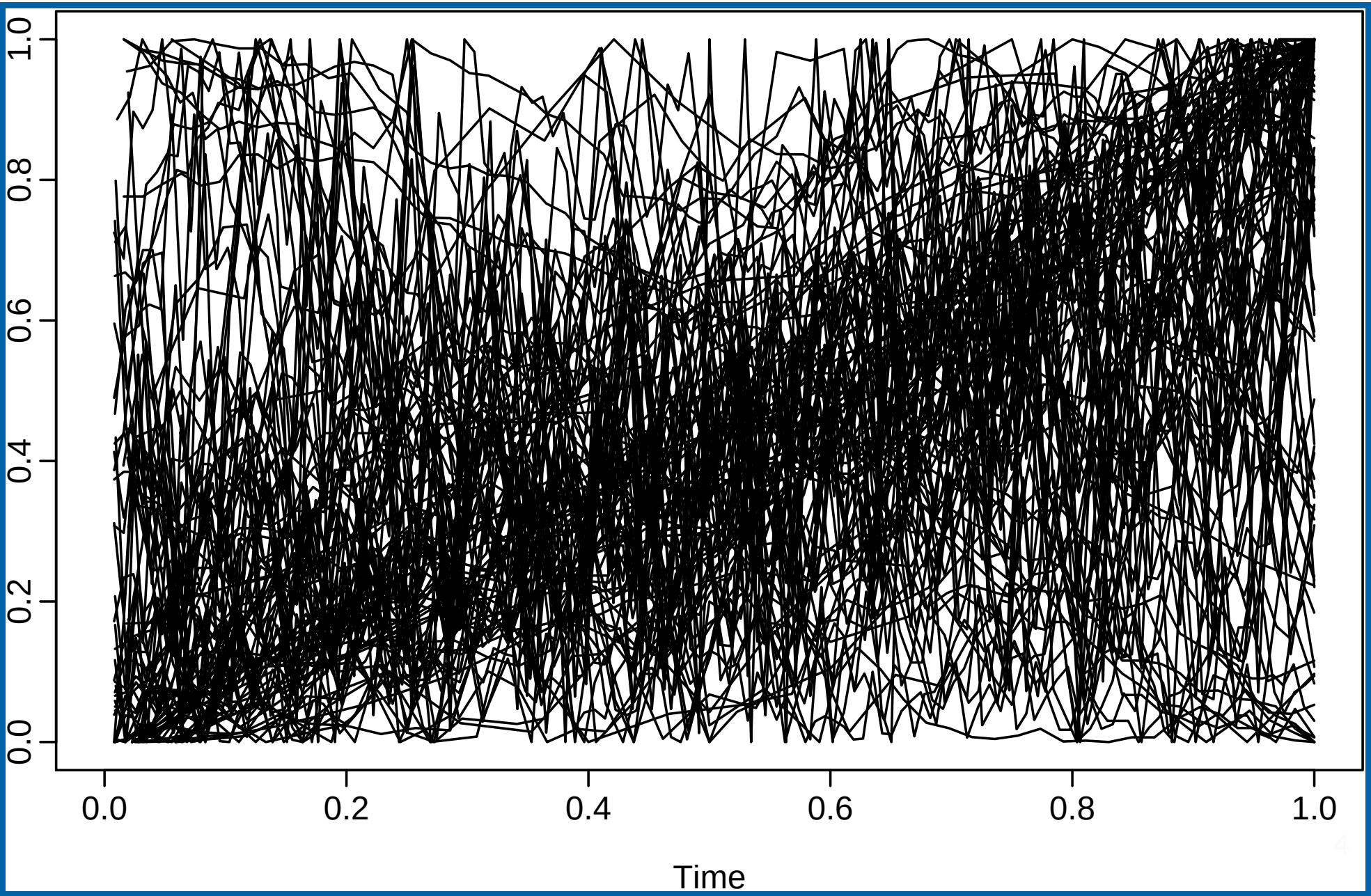
# How to plot lots of time series?



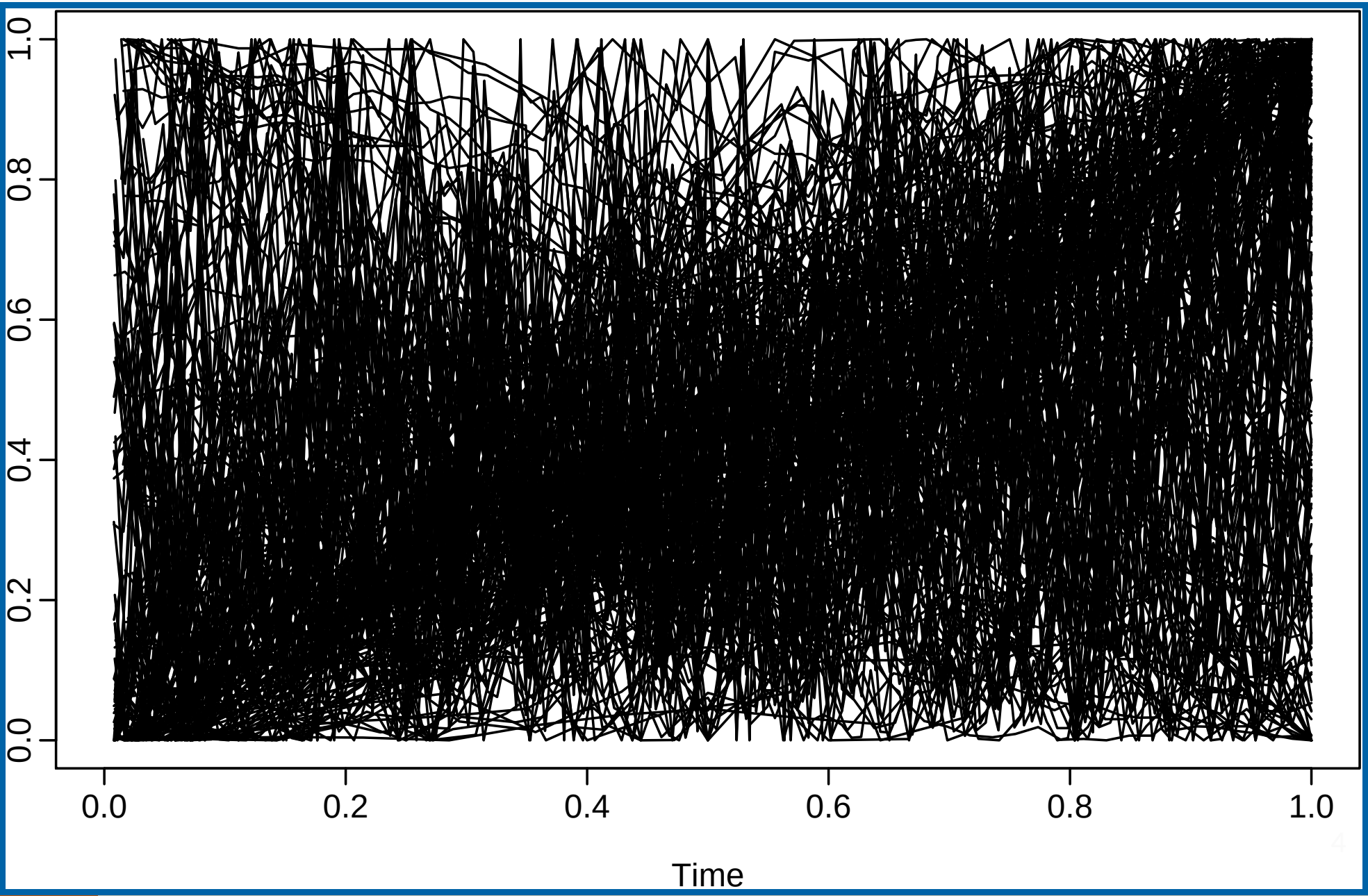
# How to plot lots of time series?



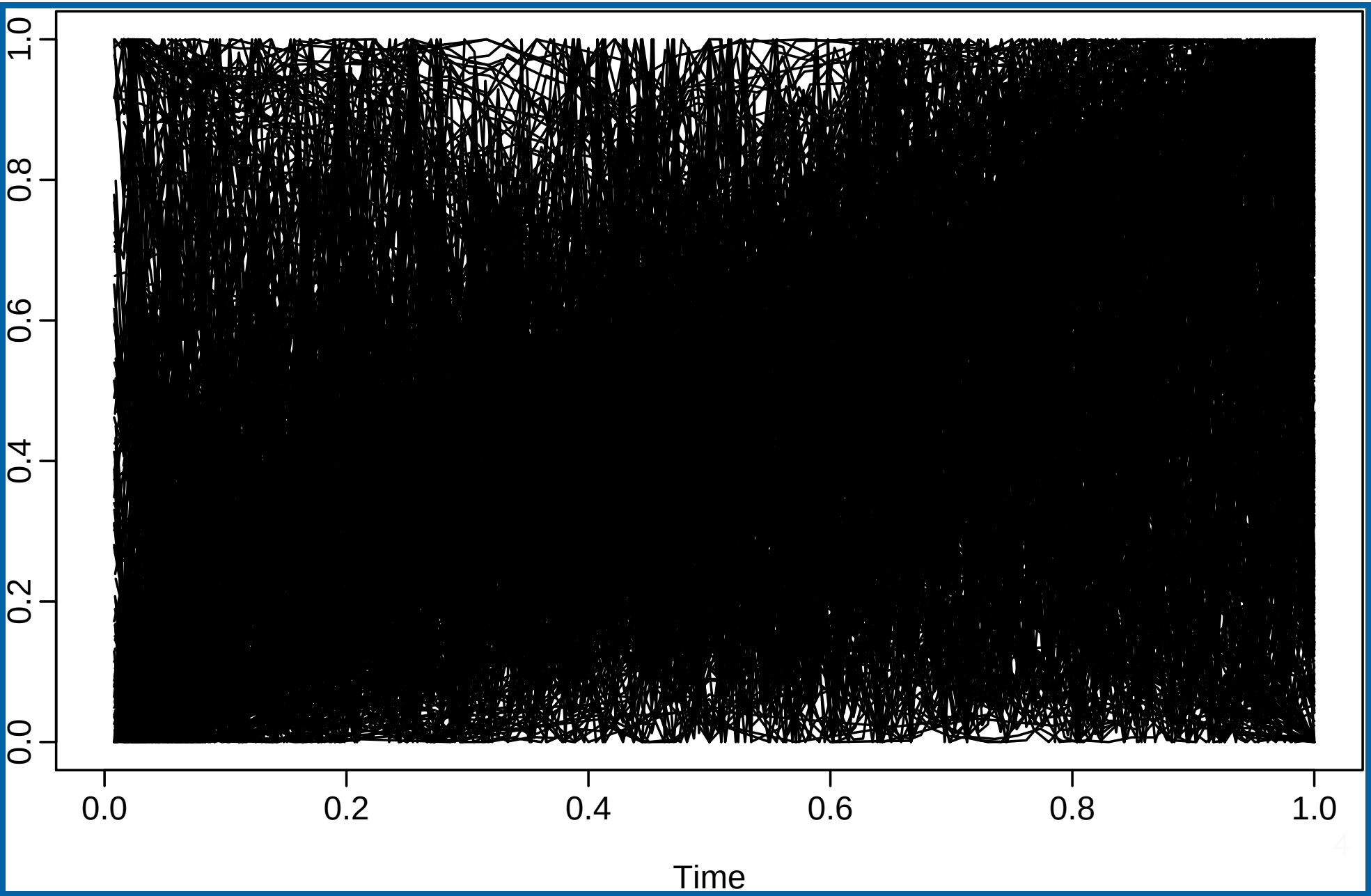
# How to plot lots of time series?



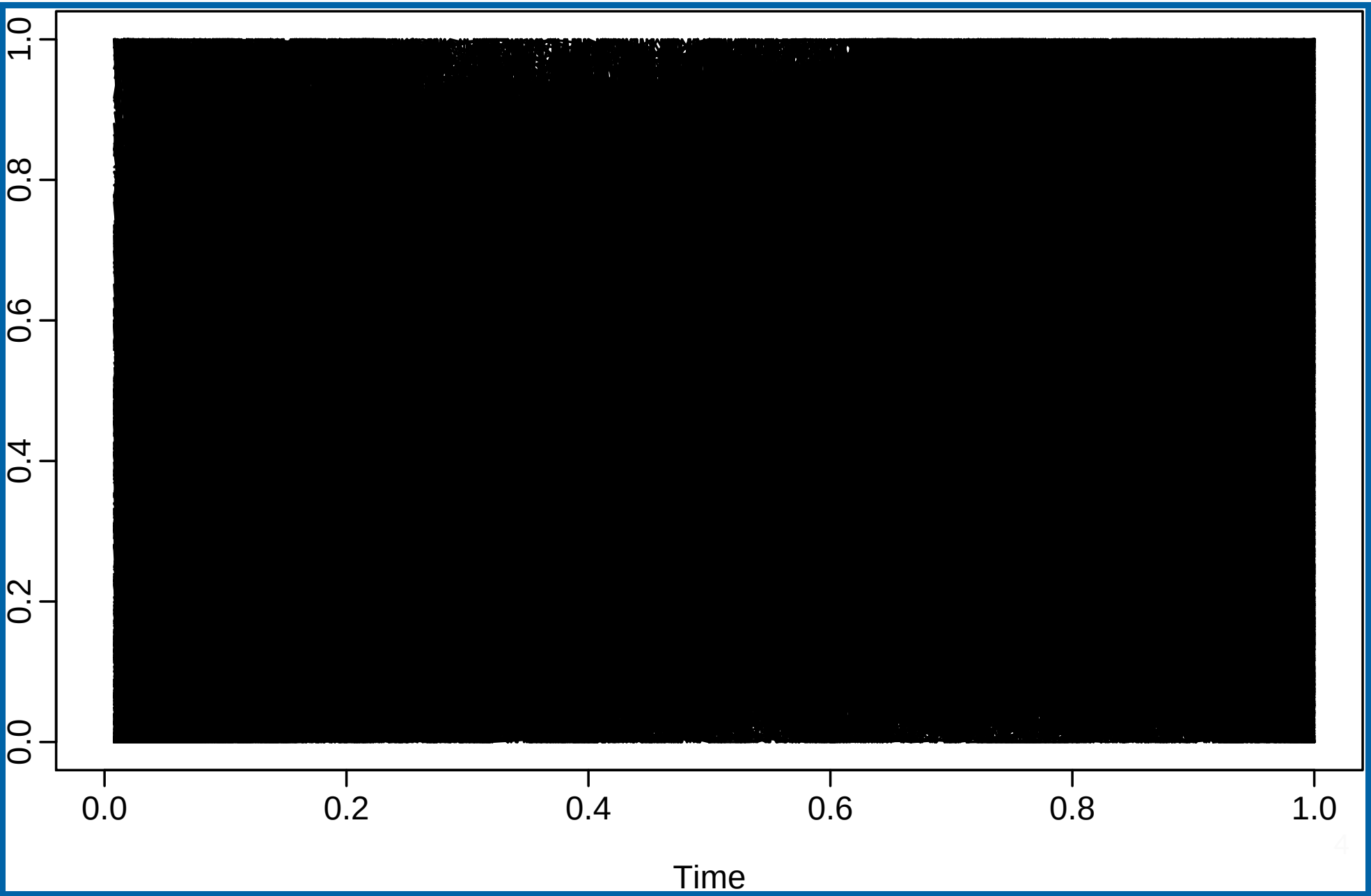
# How to plot lots of time series?



# How to plot lots of time series?



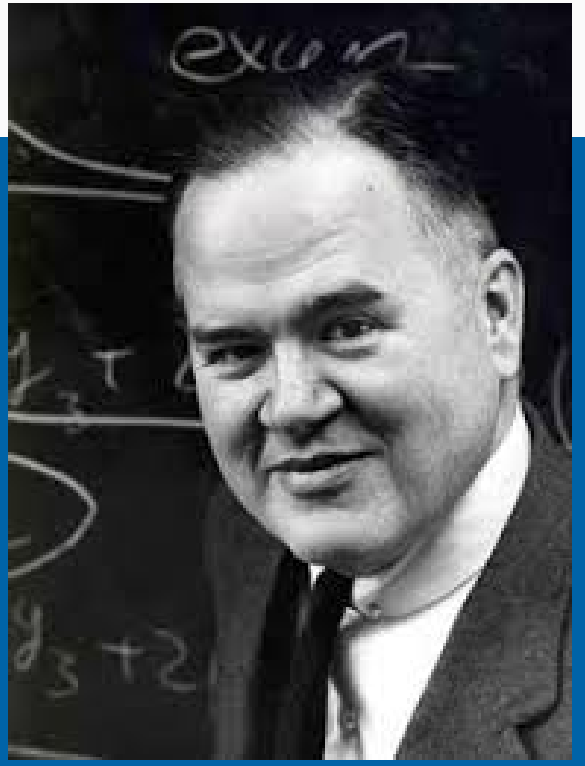
# How to plot lots of time series?



# Key idea

**Cognostics**

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

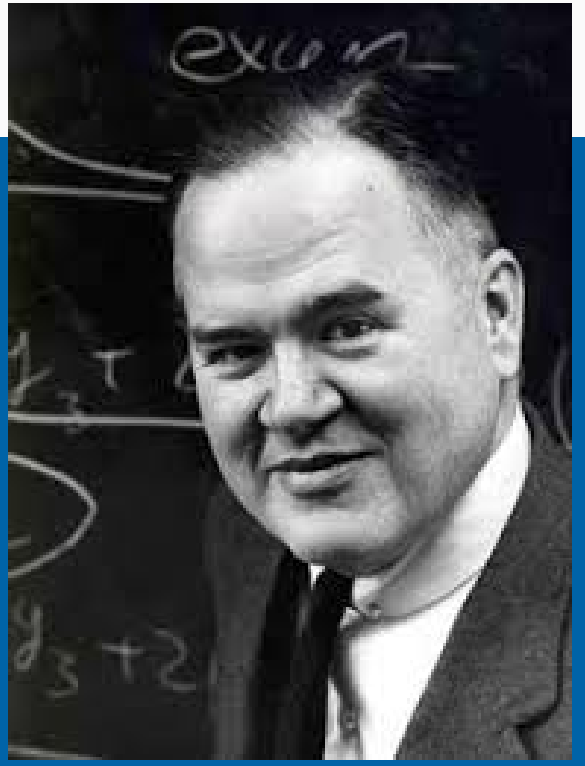


*John W Tukey*

# Key idea

**Cognostics**

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



*John W Tukey*

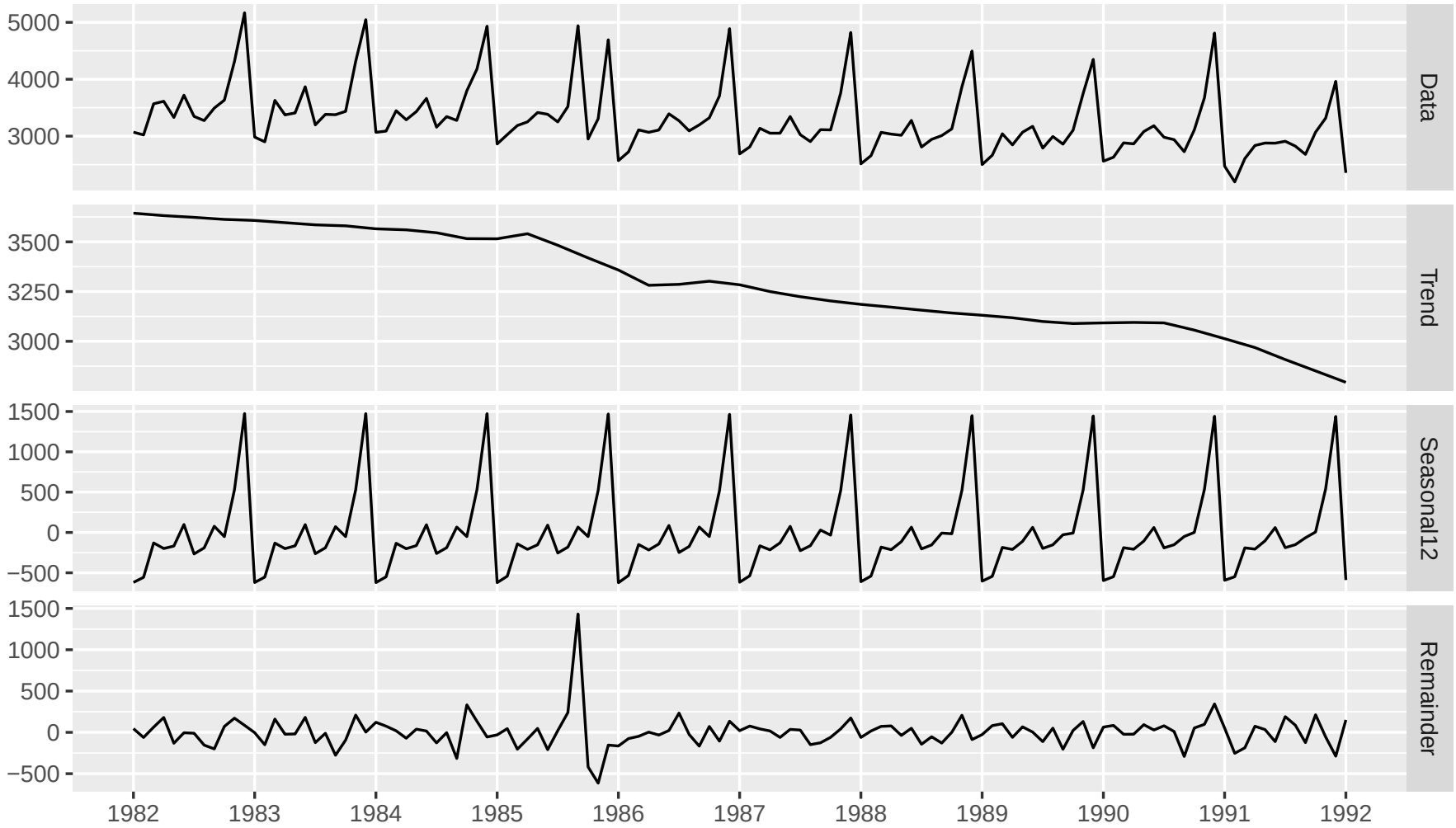
## Examples for time series

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

Called “features” in the machine learning literature.

# An STL decomposition: N2096

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



# Candidate features

## STL decomposition

$$Y_t = S_t + T_t + R_t$$

# Candidate features

## STL decomposition

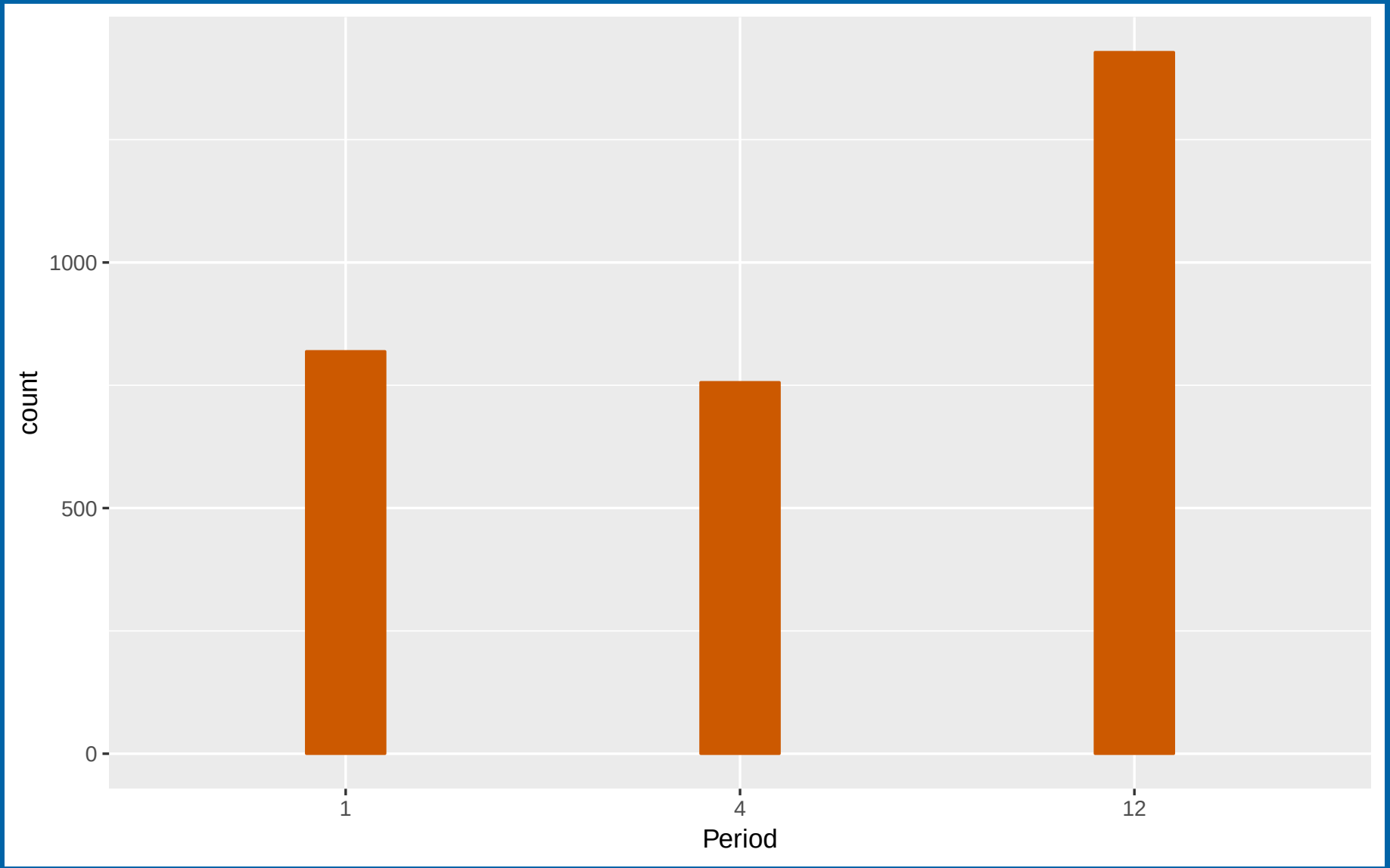
$$Y_t = S_t + T_t + R_t$$

- Seasonal period
- Autocorrelations of data $(Y_1, \dots, Y_T)$
- Autocorrelations of data $(R_1, \dots, R_T)$
- Strength of seasonality: $\max\left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - T_t)}\right)$
- Strength of trend: $\max\left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(Y_t - S_t)}\right)$
- Spectral entropy: $H = -\int_{-\pi}^{\pi} f_y(\lambda) \log f_y(\lambda) d\lambda$, where $f_y(\lambda)$ is spectral density of $Y_t$.\newline Low values of $H$ suggest a time series that is easier to forecast (more signal).
- Optimal Box-Cox transformation of data

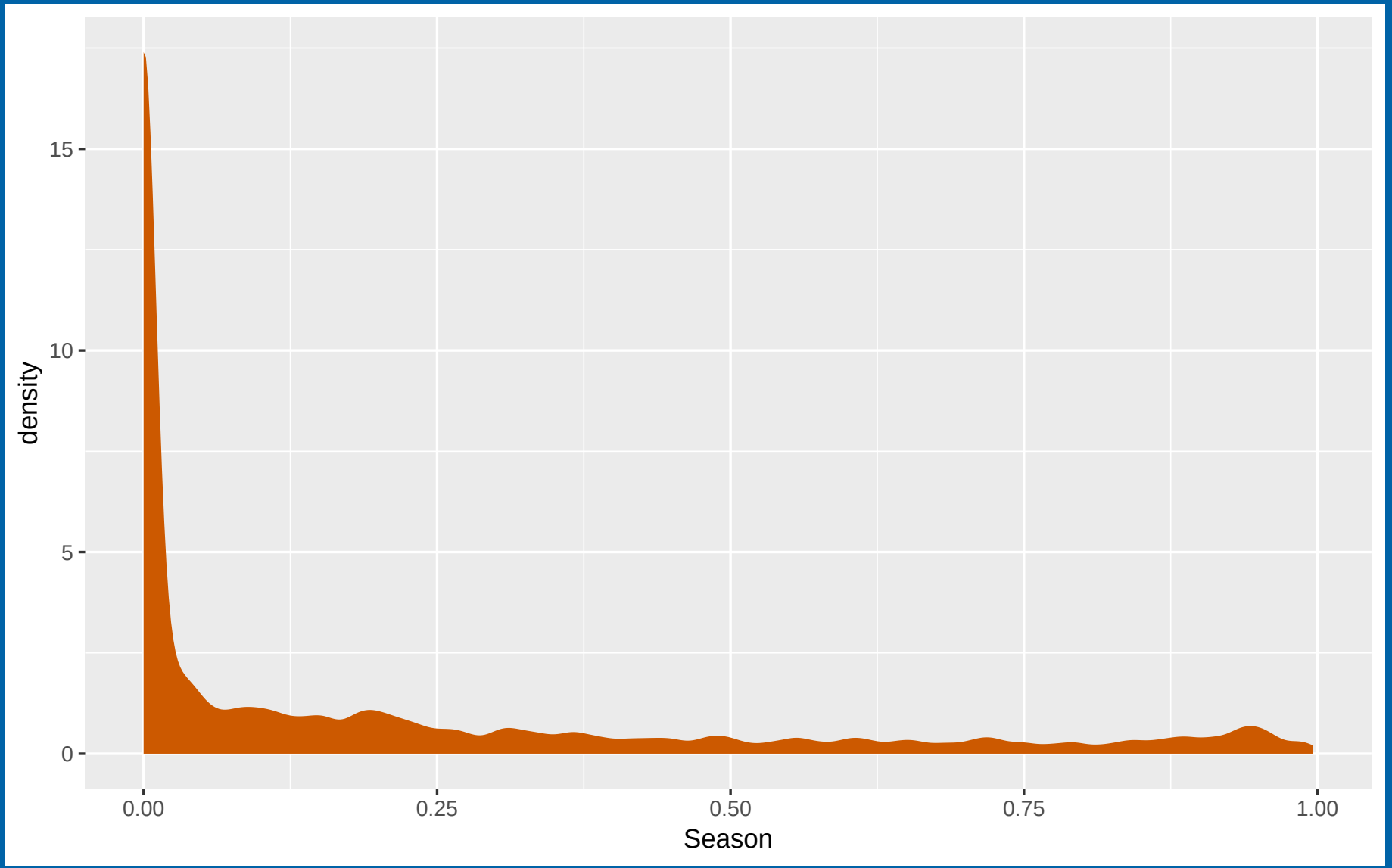
# tsfeatures package

```
library(tsfeatures)
lambda_stl <- function(x,...) {
  lambda <- forecast::BoxCox.lambda(x,
    lower=0, upper=1, method='loglik')
  y <- forecast::BoxCox(x, lambda)
  c(stl_features(y,s.window='periodic', robust=TRUE, ...),
    lambda=lambda)
}
M3Features <- bind_cols(
  tsfeatures(M3data, c("frequency", "entropy")),
  tsfeatures(M3data, "lambda_stl", scale=FALSE)) %>%
  select(frequency, entropy, trend, seasonal_strength,
    e_acf1, lambda) %>%
  replace_na(list(seasonal_strength=0)) %>%
  rename(
    Frequency = frequency,
    Entropy = entropy,
    Trend = trend,
    Season = seasonal_strength,
    ACF1 = e_acf1,
    Lambda = lambda) %>%
  mutate(Period = as.factor(Frequency))
```

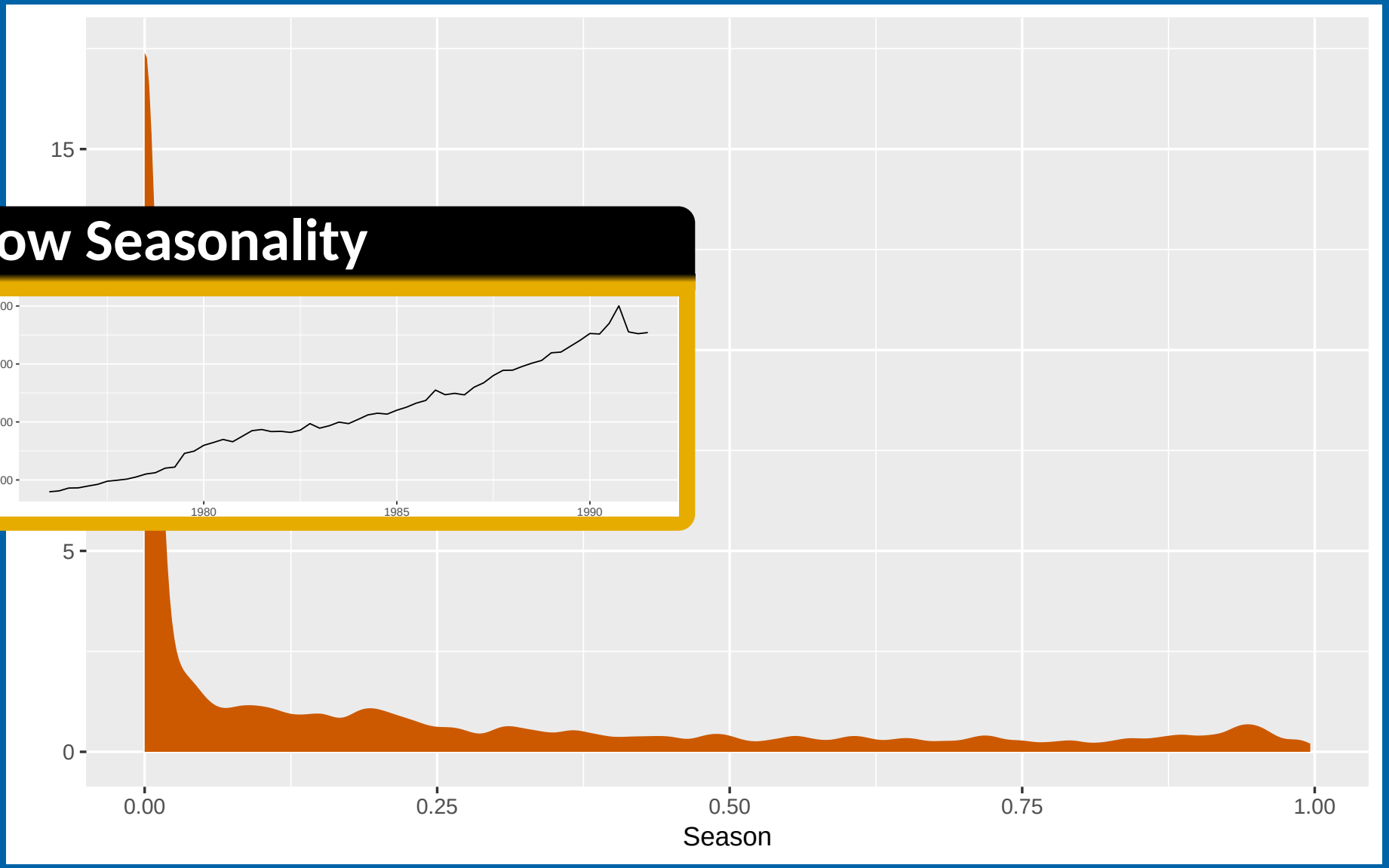
# Distribution of Period for M3



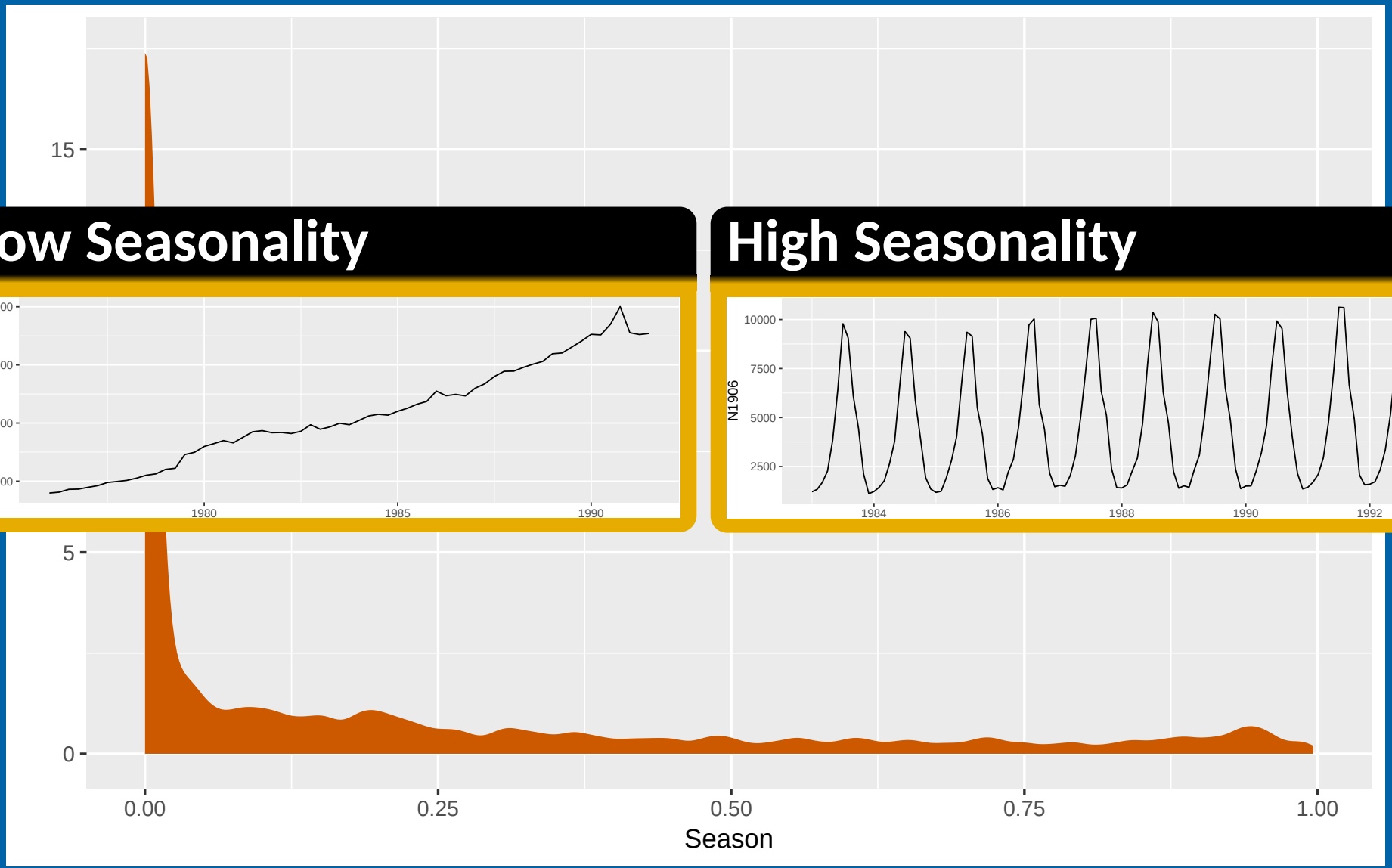
# Distribution of Seasonality for M3



# Distribution of Seasonality for M3



# Distribution of Seasonality for M3

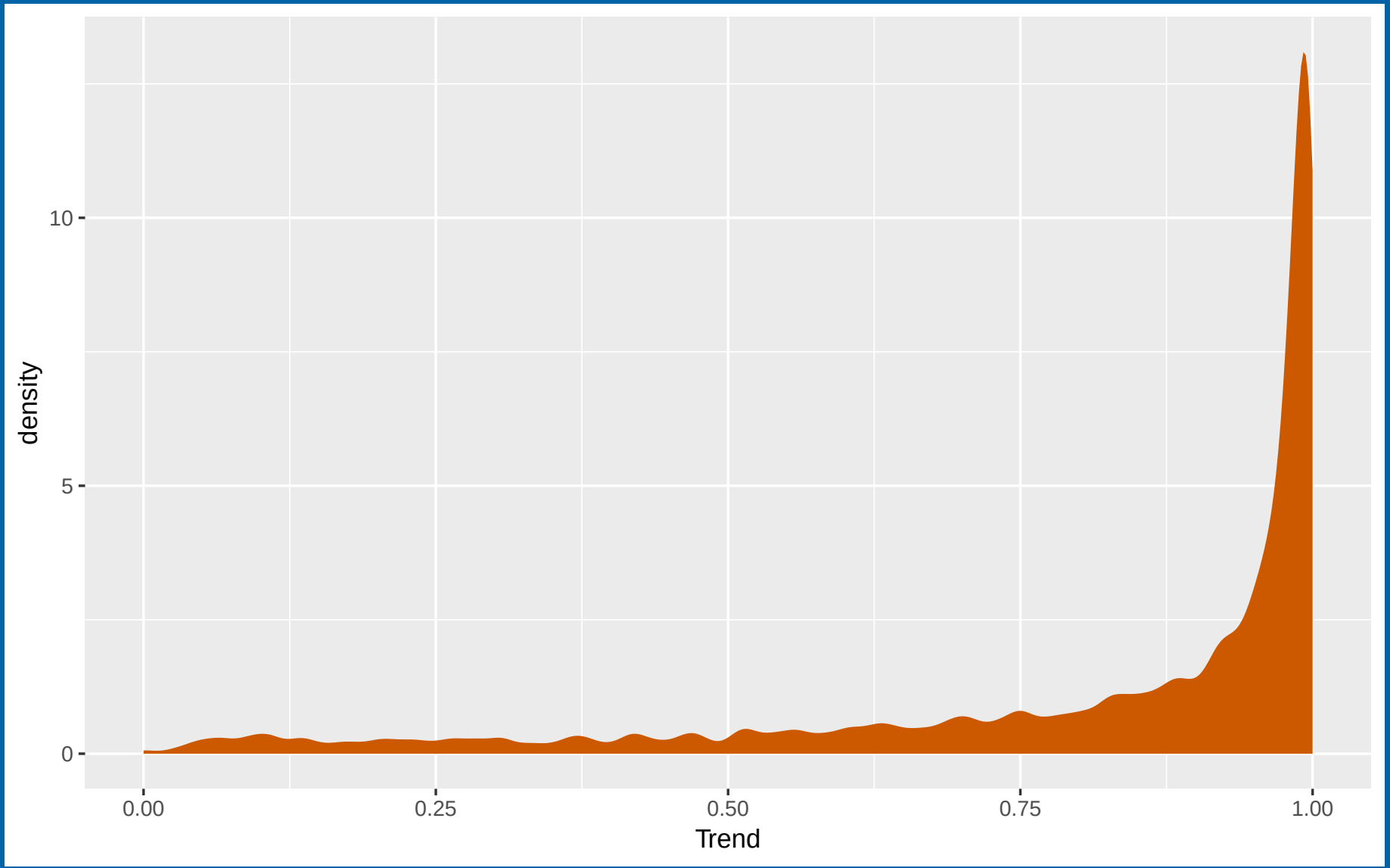



## Low Seasonality

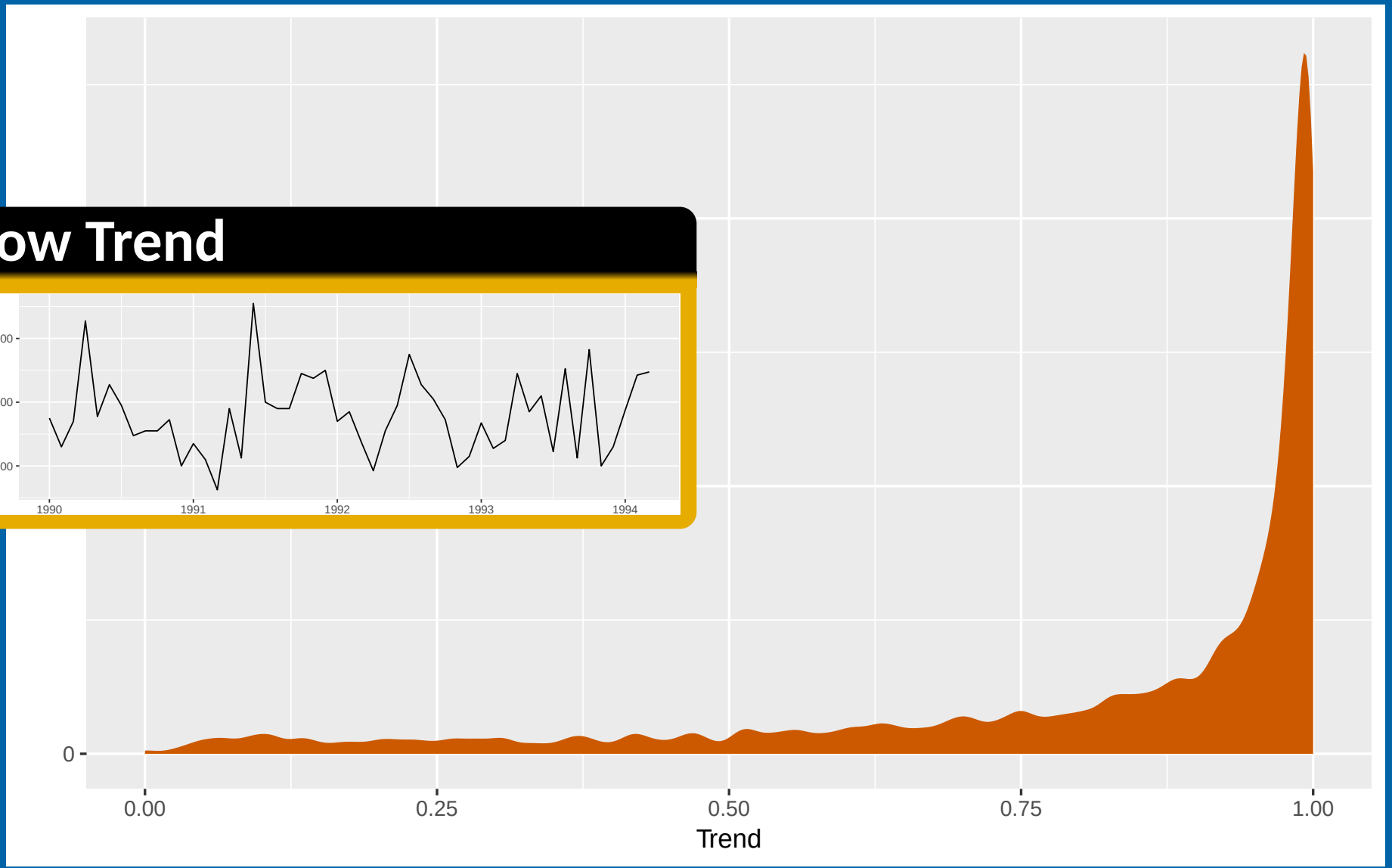



## High Seasonality

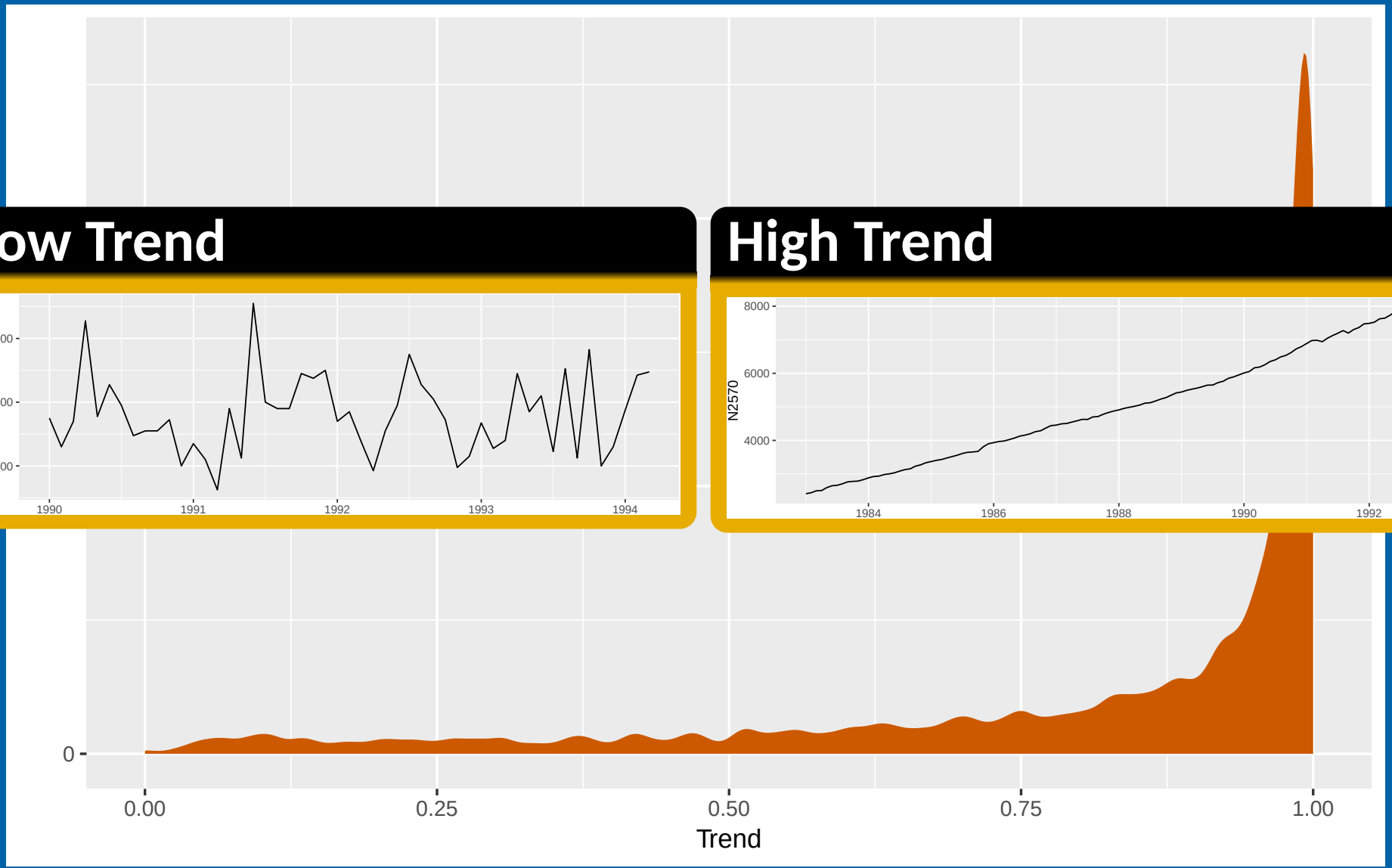
# Distribution of Trend for M3



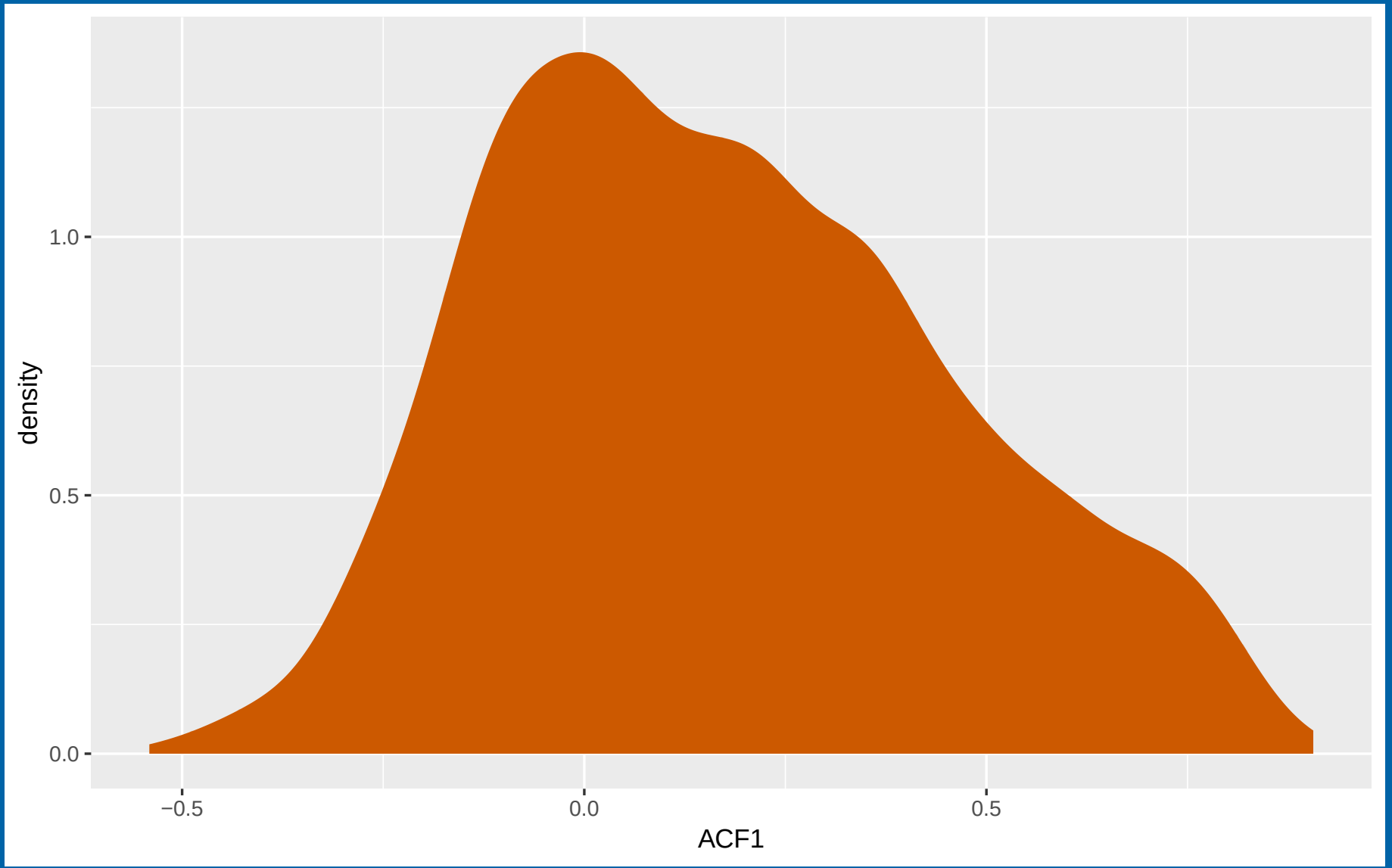
# Distribution of Trend for M3



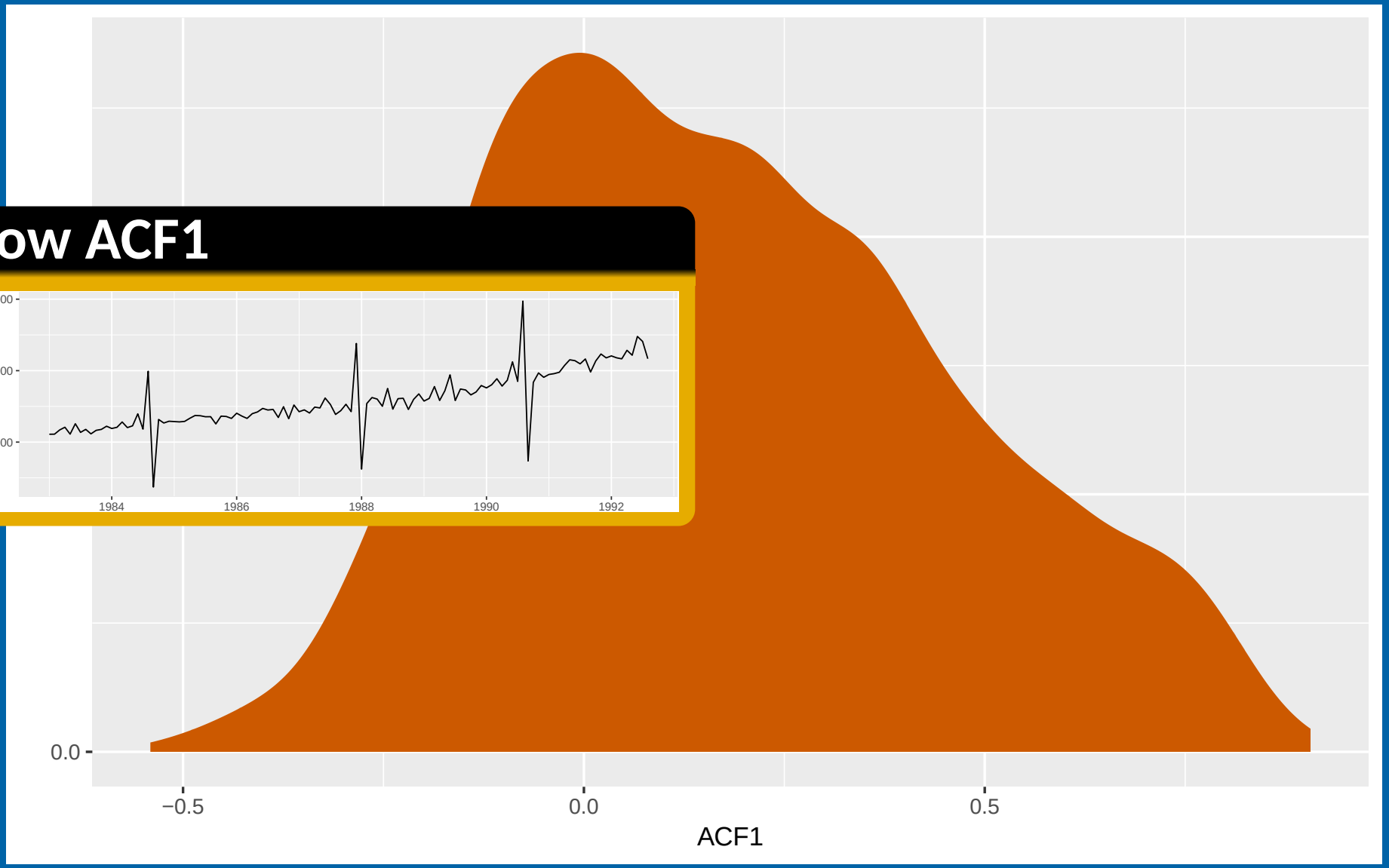
# Distribution of Trend for M3



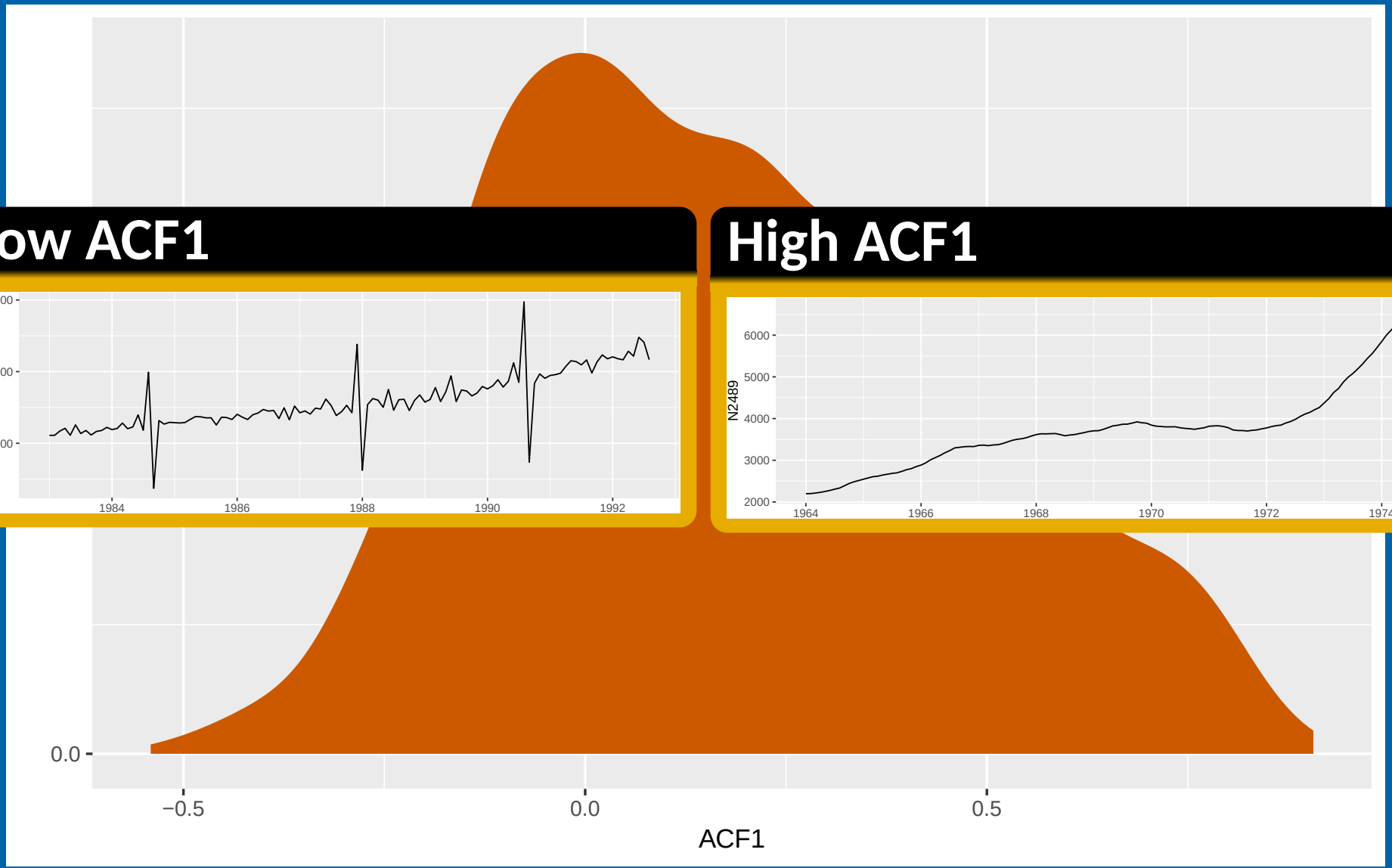
# Distribution of Residual ACF1 for M3



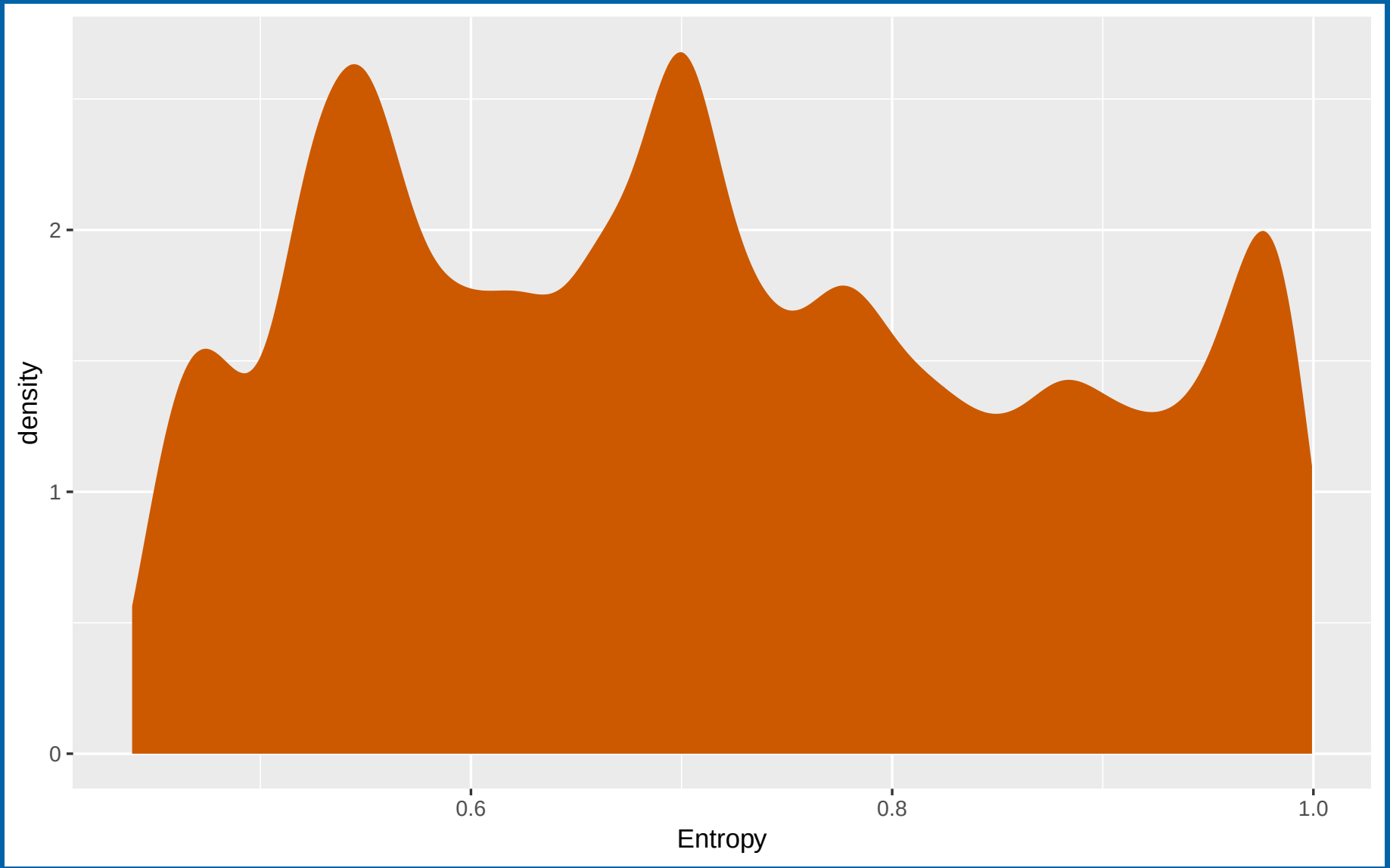
# Distribution of Residual ACF1 for M3



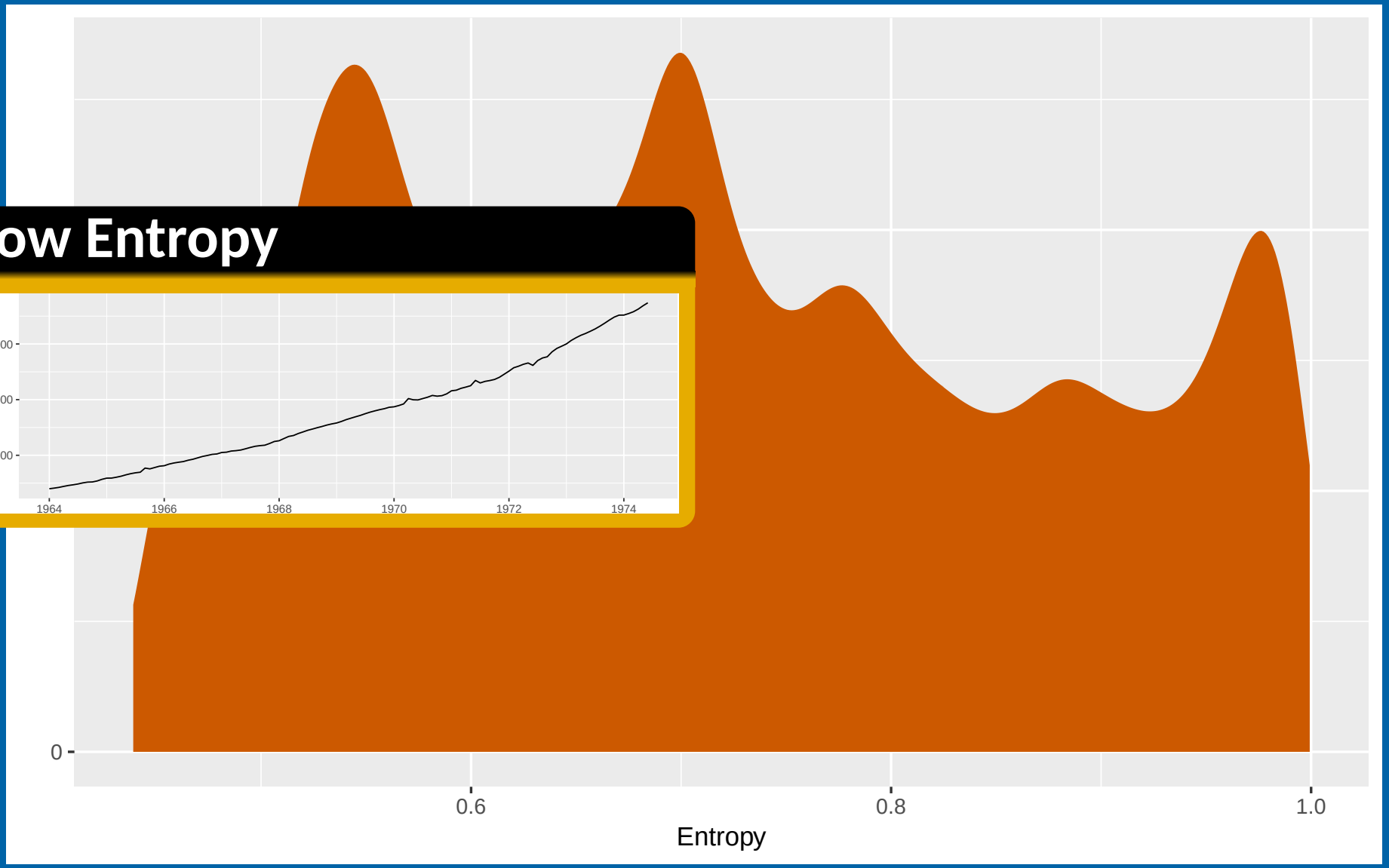
# Distribution of Residual ACF1 for M3



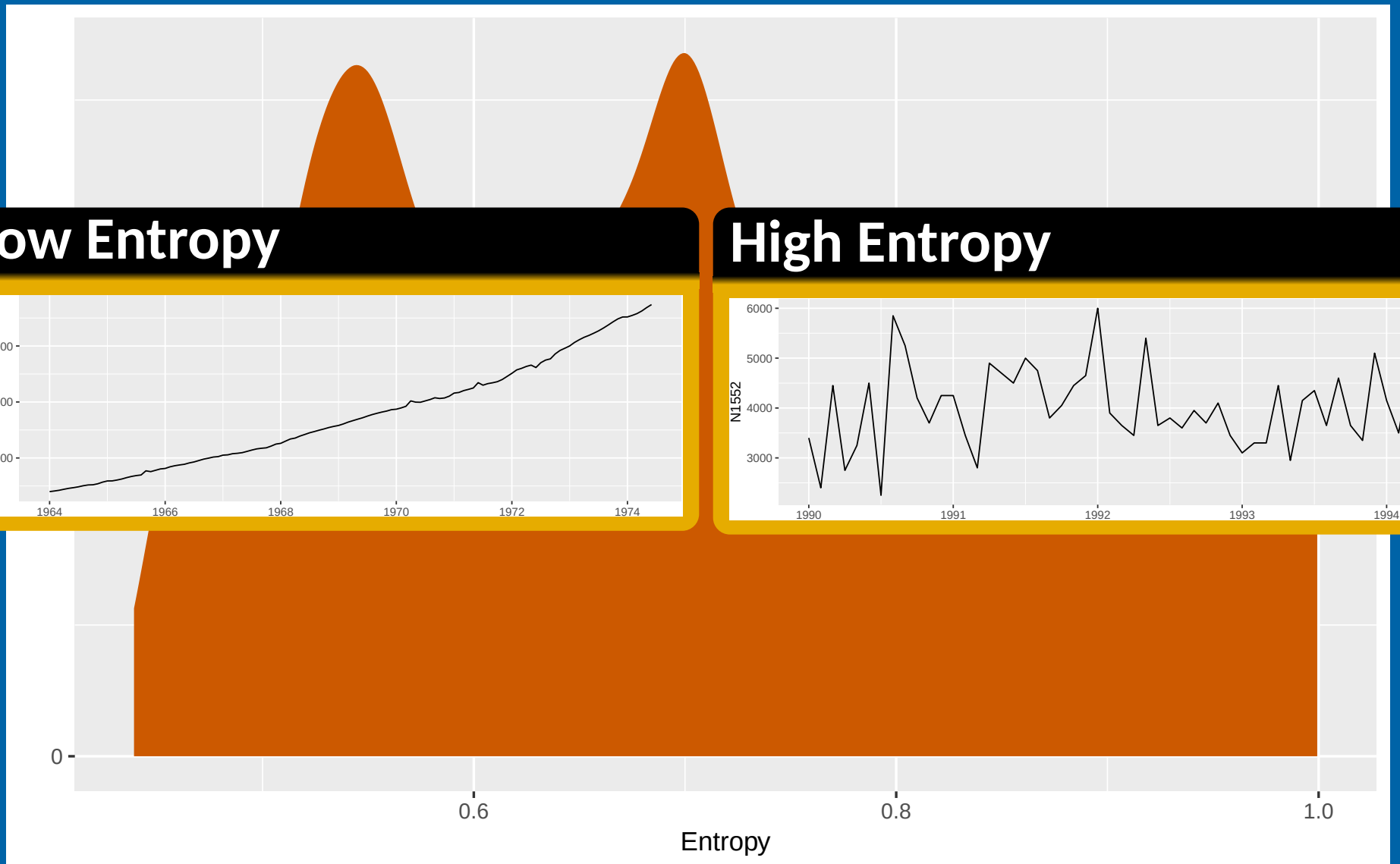
# Distribution of Spectral Entropy for M3



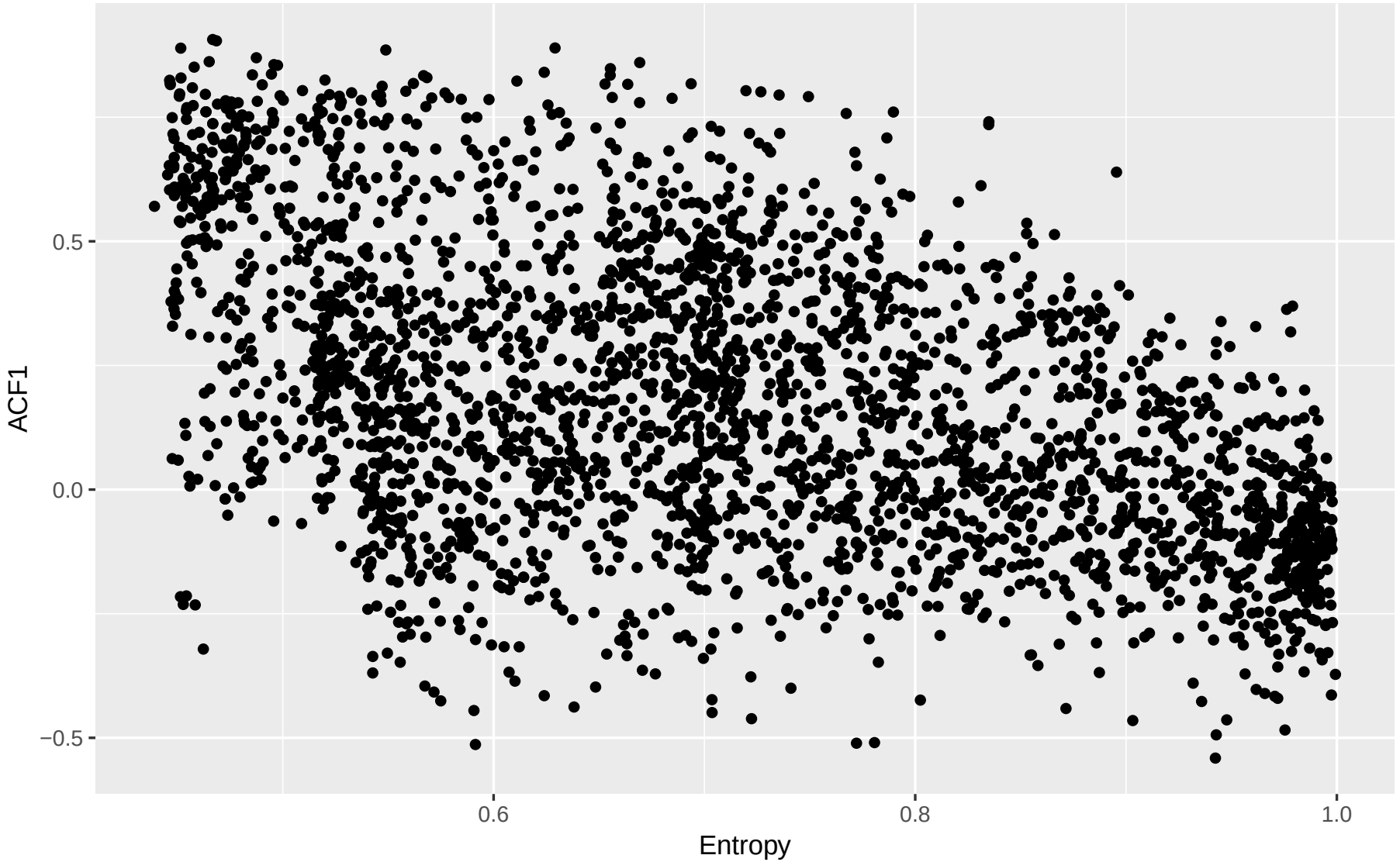
# Distribution of Spectral Entropy for M3



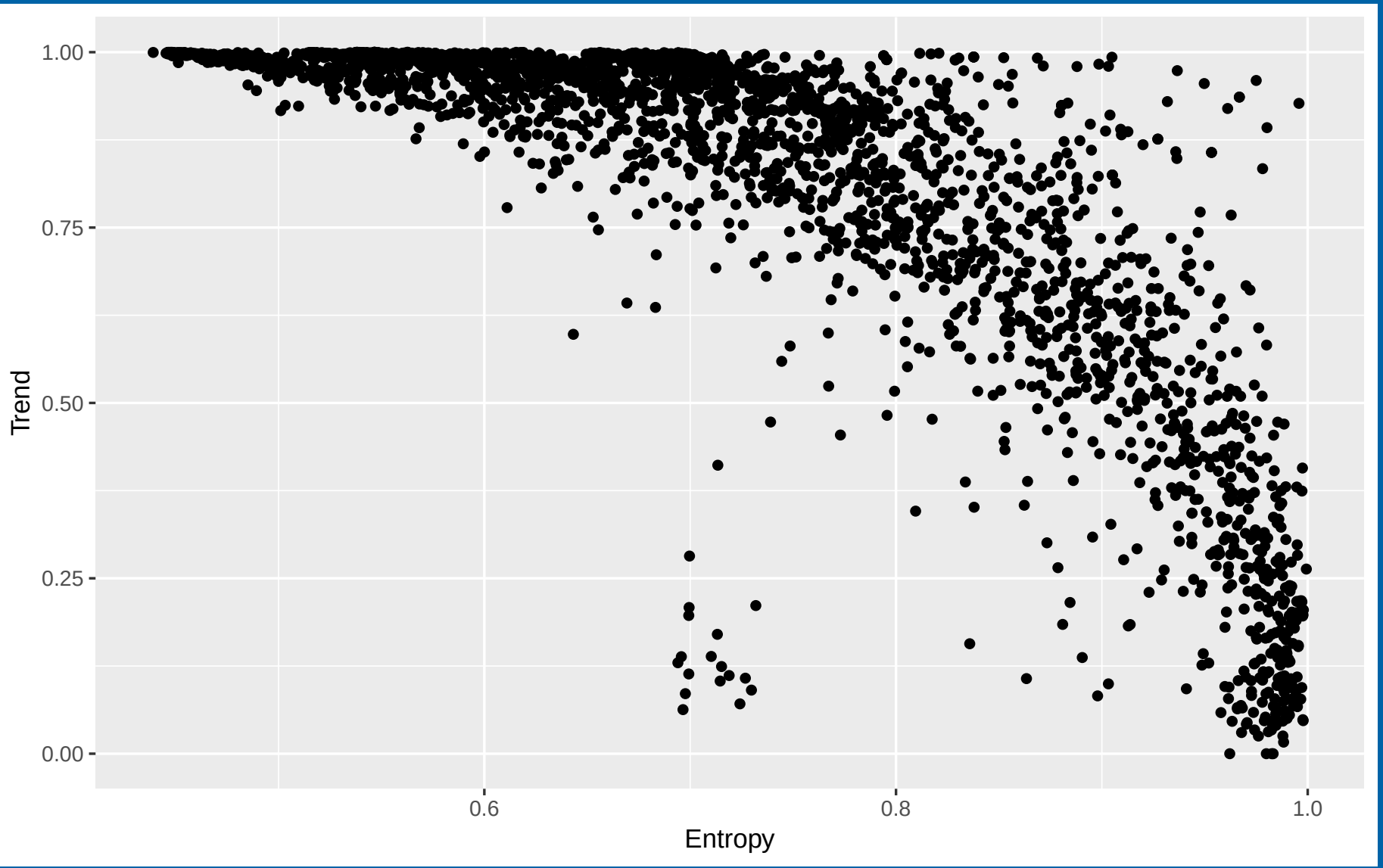
# Distribution of Spectral Entropy for M3



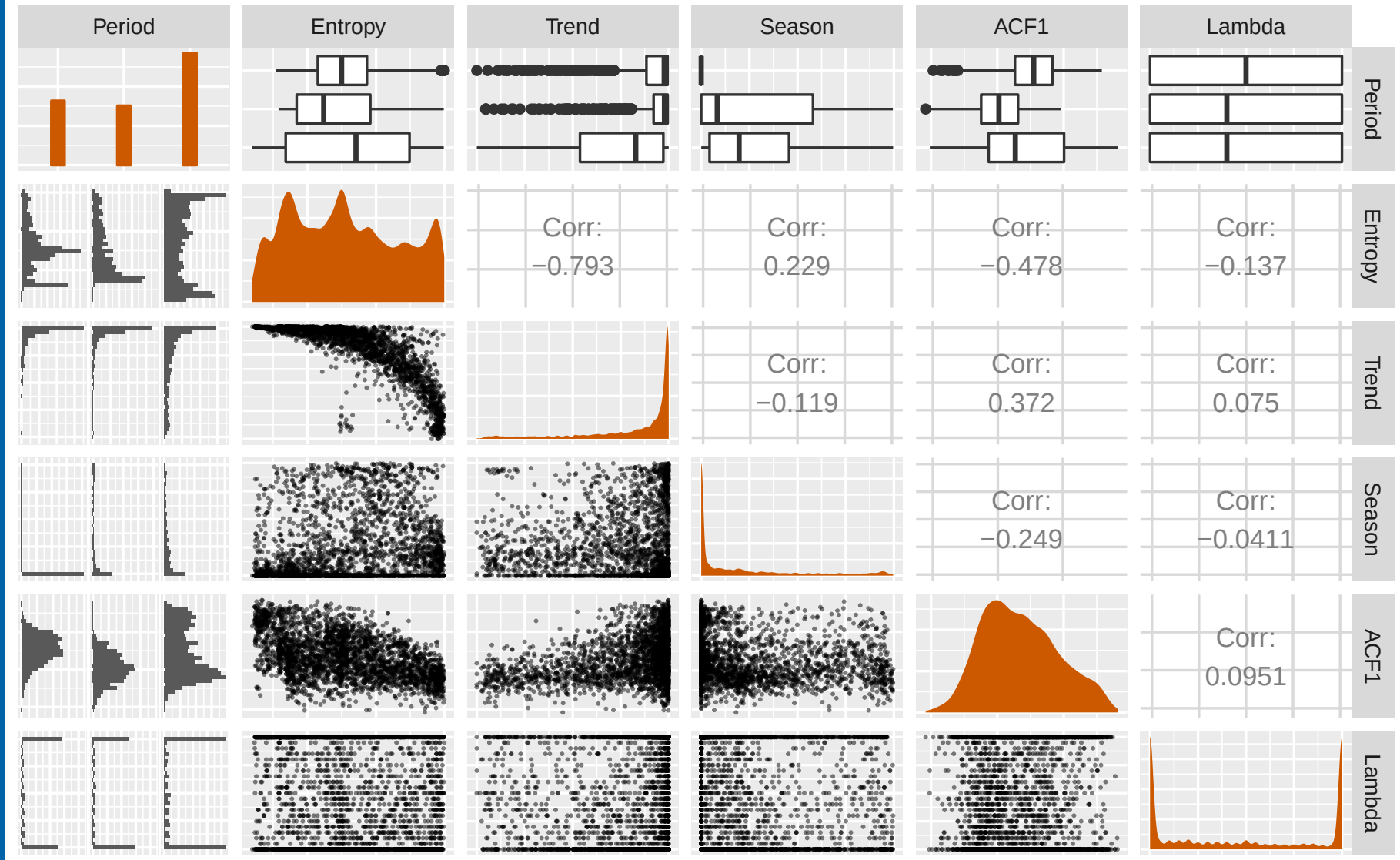
# Feature distributions



# Feature distributions

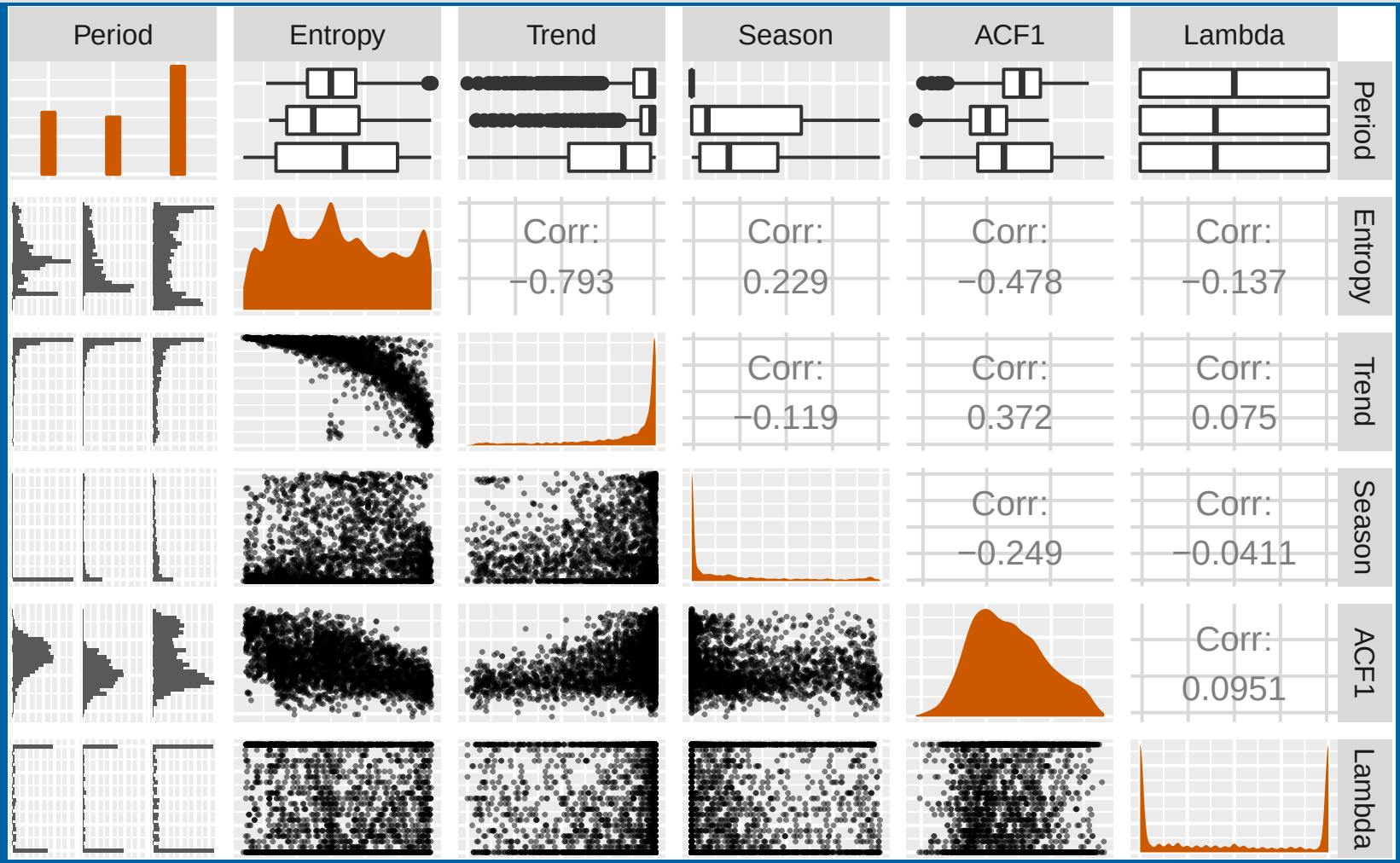


# Feature distributions

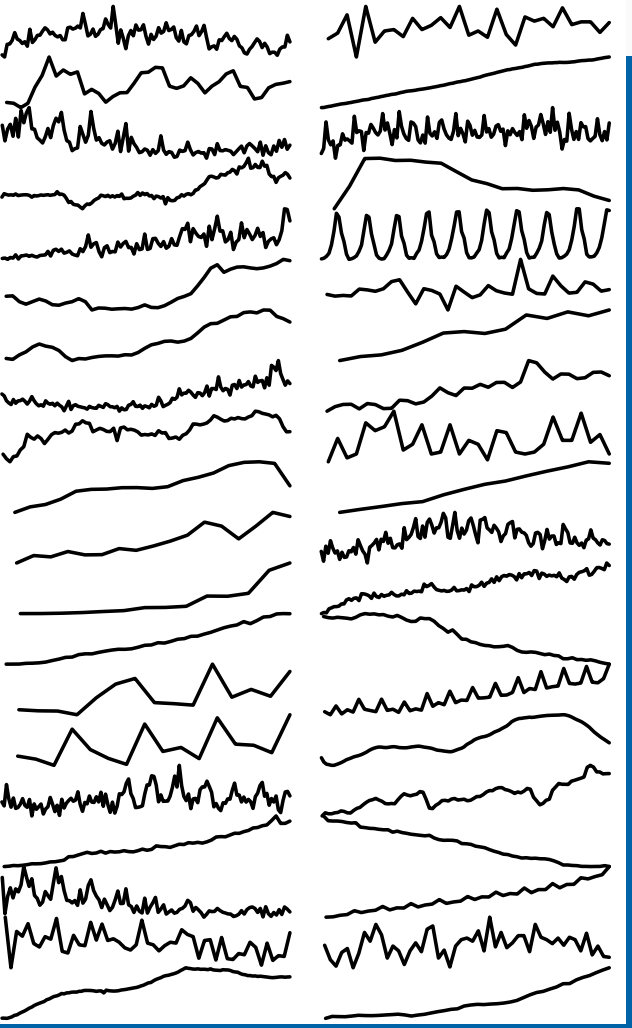


# Feature distributions

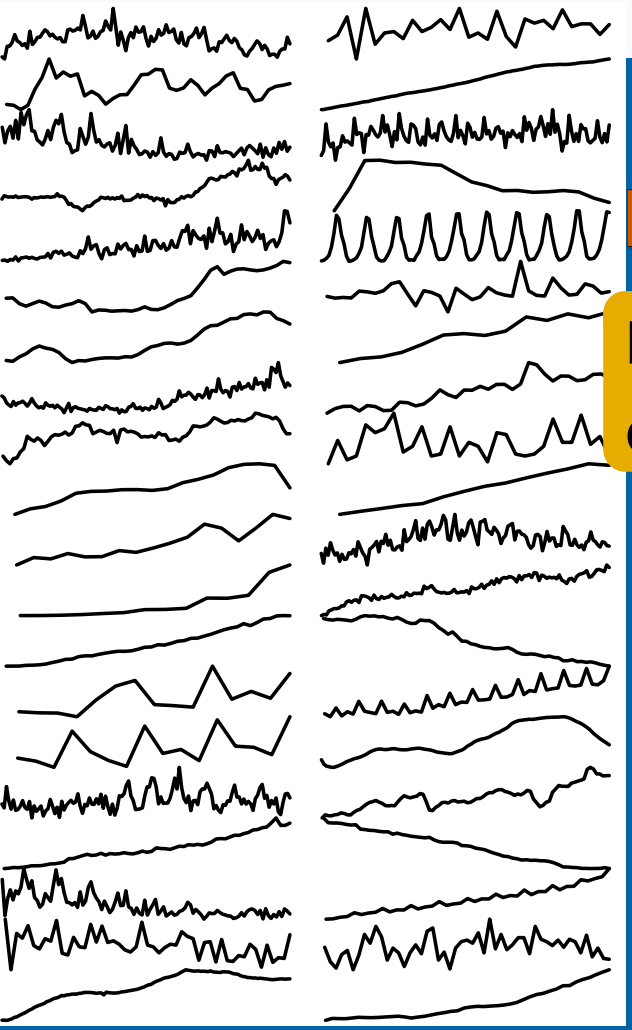
```
M3Features %>%
  select(Period, Entropy, Trend, Season, ACF1, Lambda) %>%
  GGally::ggpairs()
```



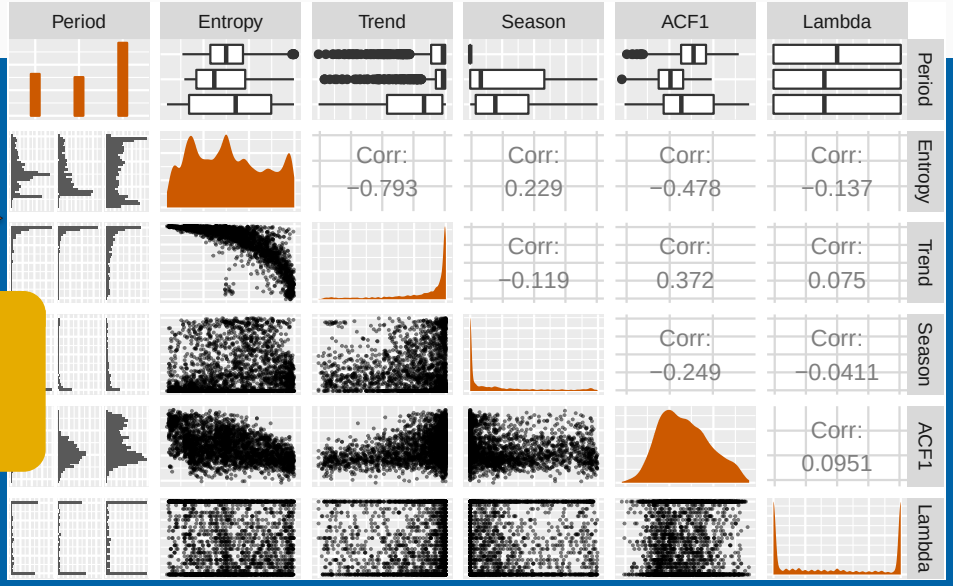
# Dimension reduction for time series



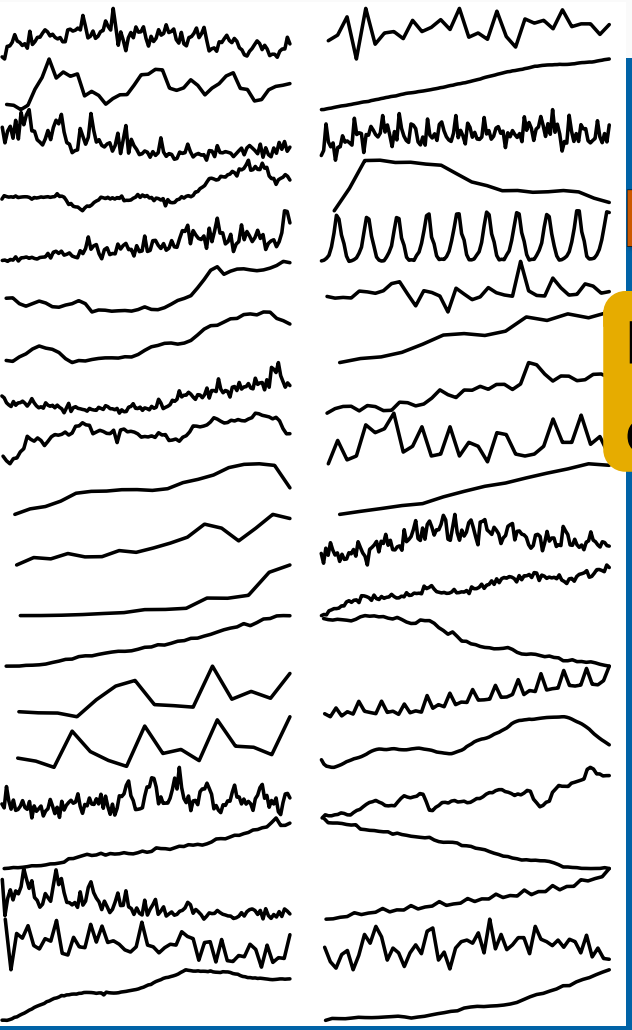
# Dimension reduction for time series



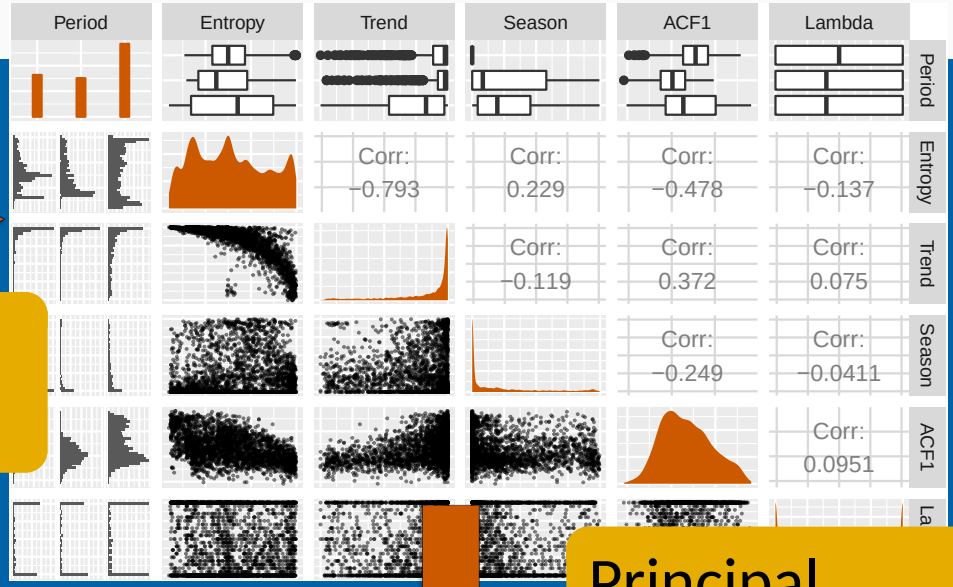
Feature calculation



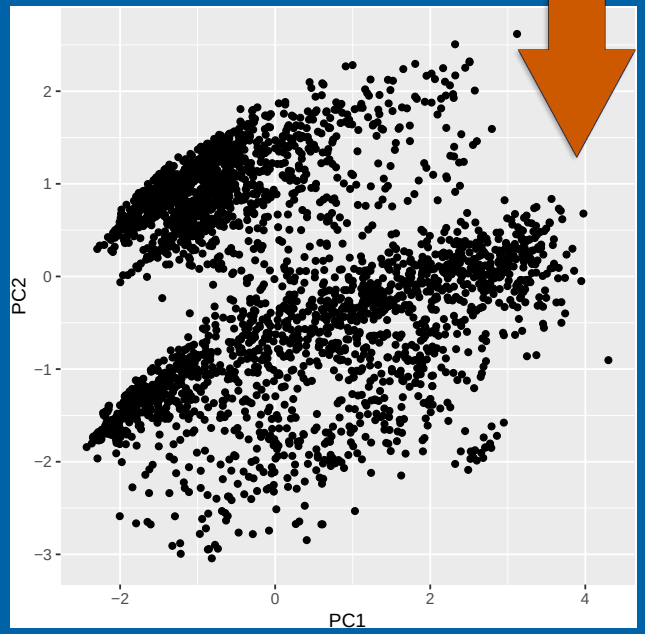
# Dimension reduction for time series



Feature calculation

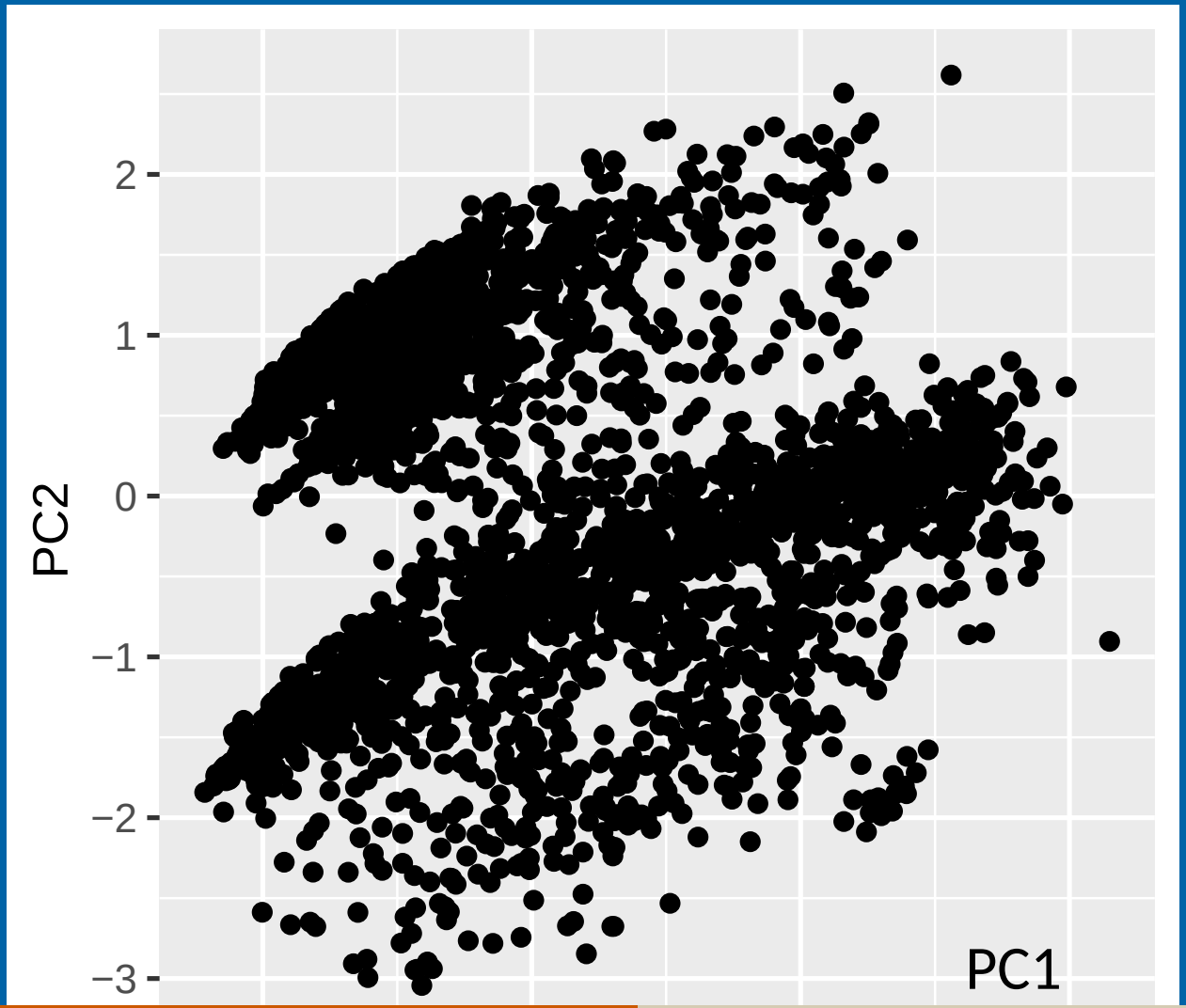



Principal component decomposition



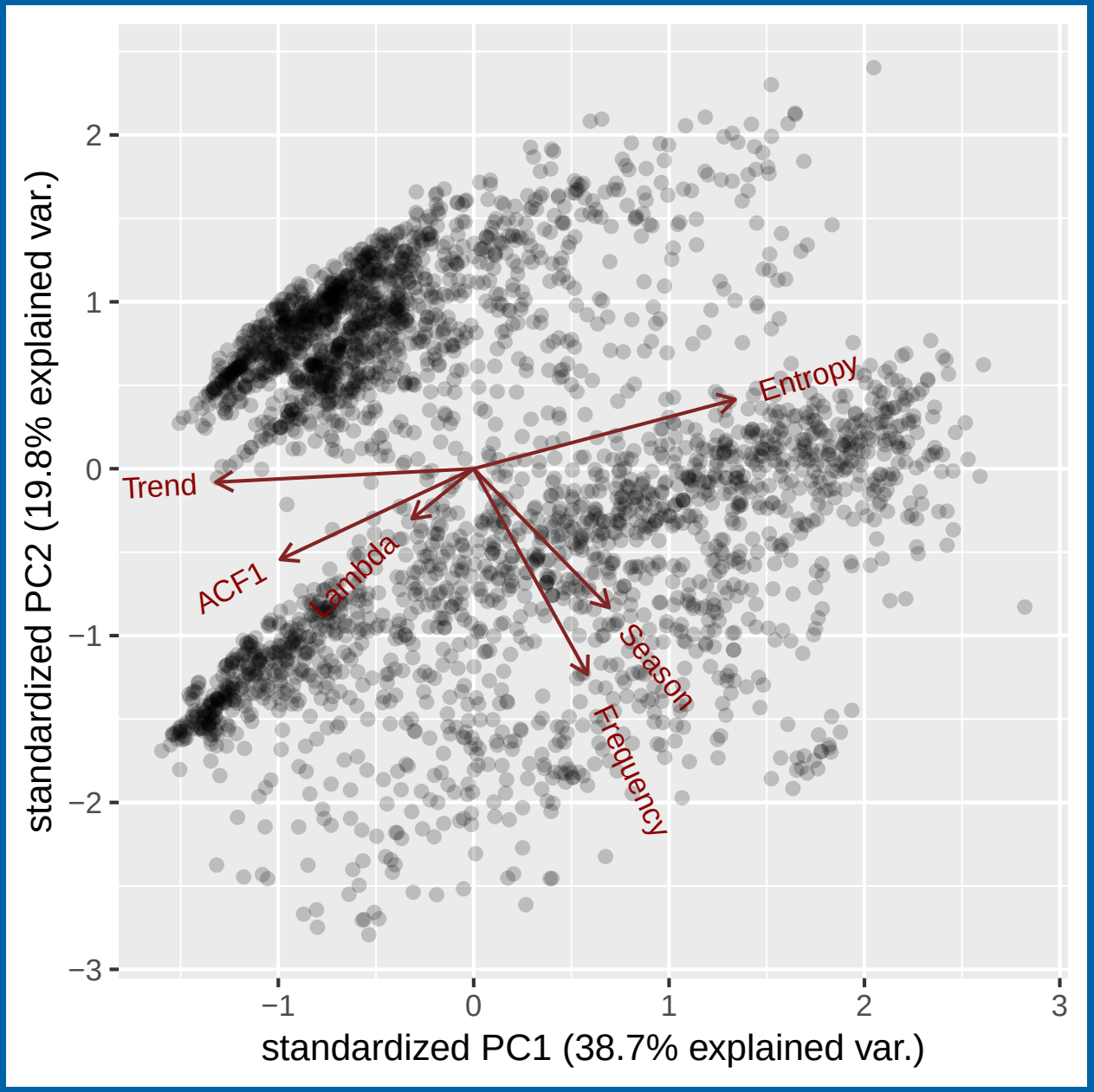
# M3 feature space

```
prcomp(select(M3Features, -Period), scale=TRUE)$x %>%
  ggplot(aes(x=PC1, y=PC2))
```

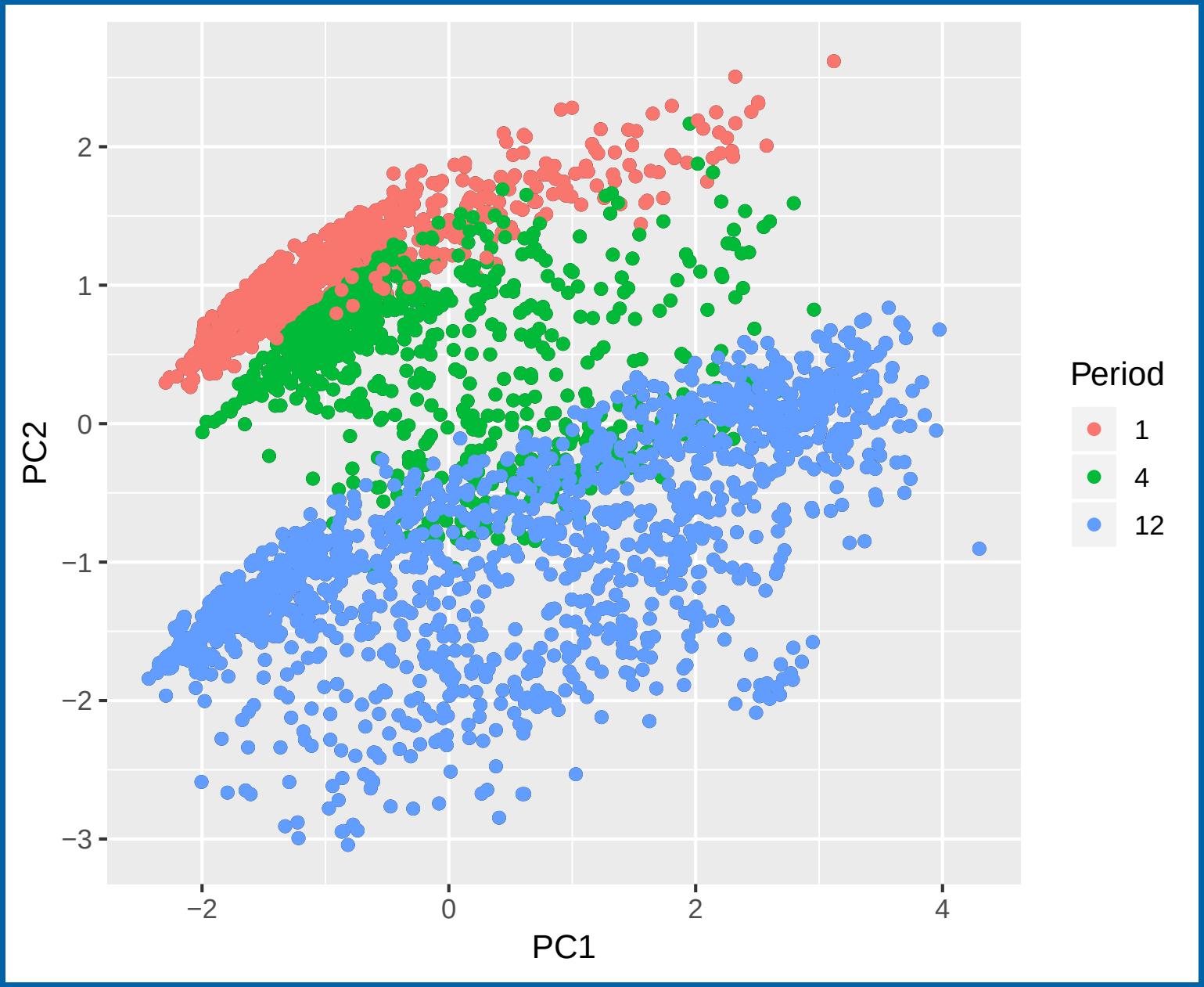



First two PCs explain 58.5% of the variance.

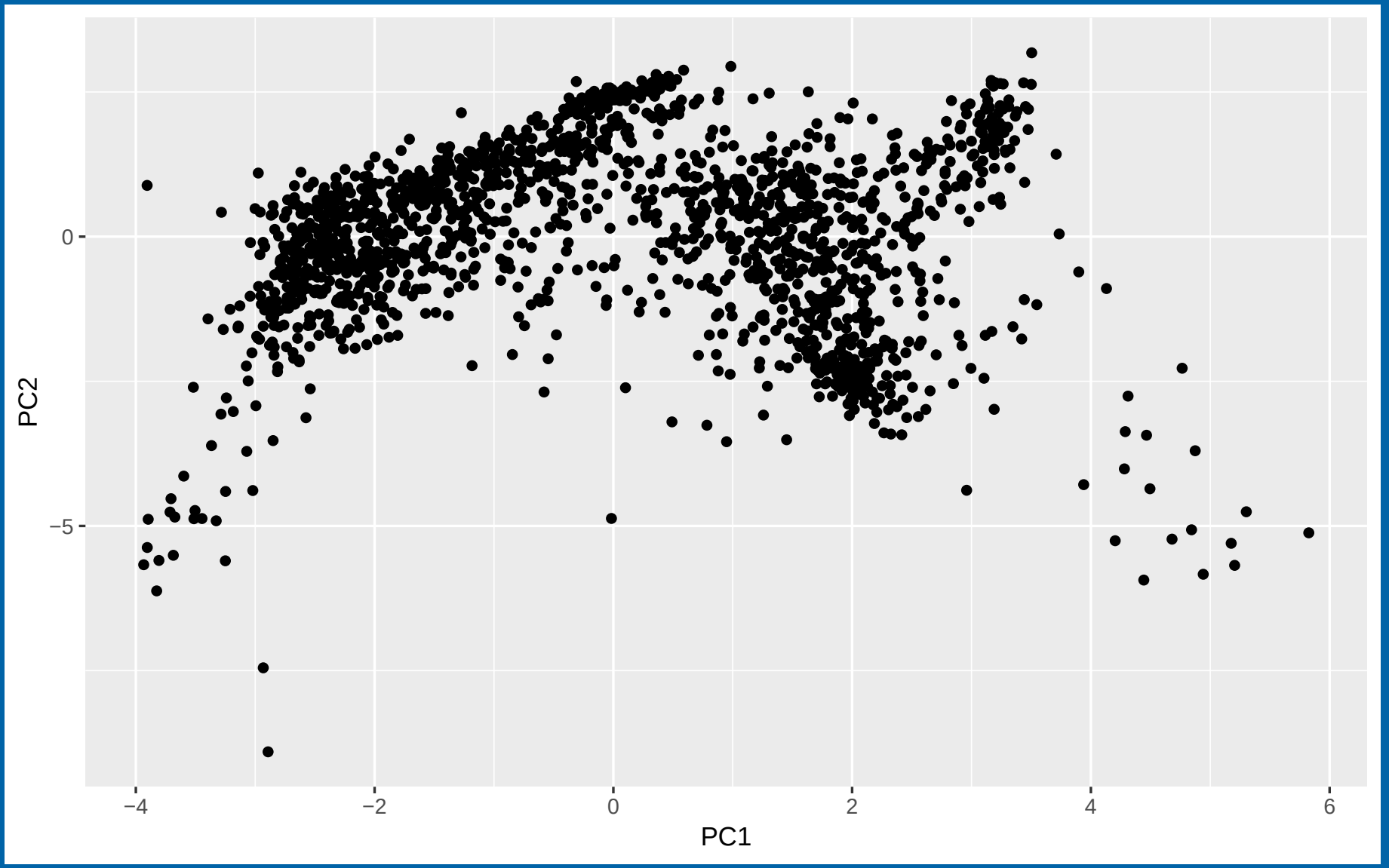
# M3 feature space



# M3 feature space



# Hyndman, Wang and Laptev (ICDM 2015)



# Outline

1. Time series feature spaces
2. Irish smart metre data
3. Quantiles conditional on time of week
4. Finding typical and unusual households
5. Visualization via embedding
6. Features and limitations

# Irish smart metre data

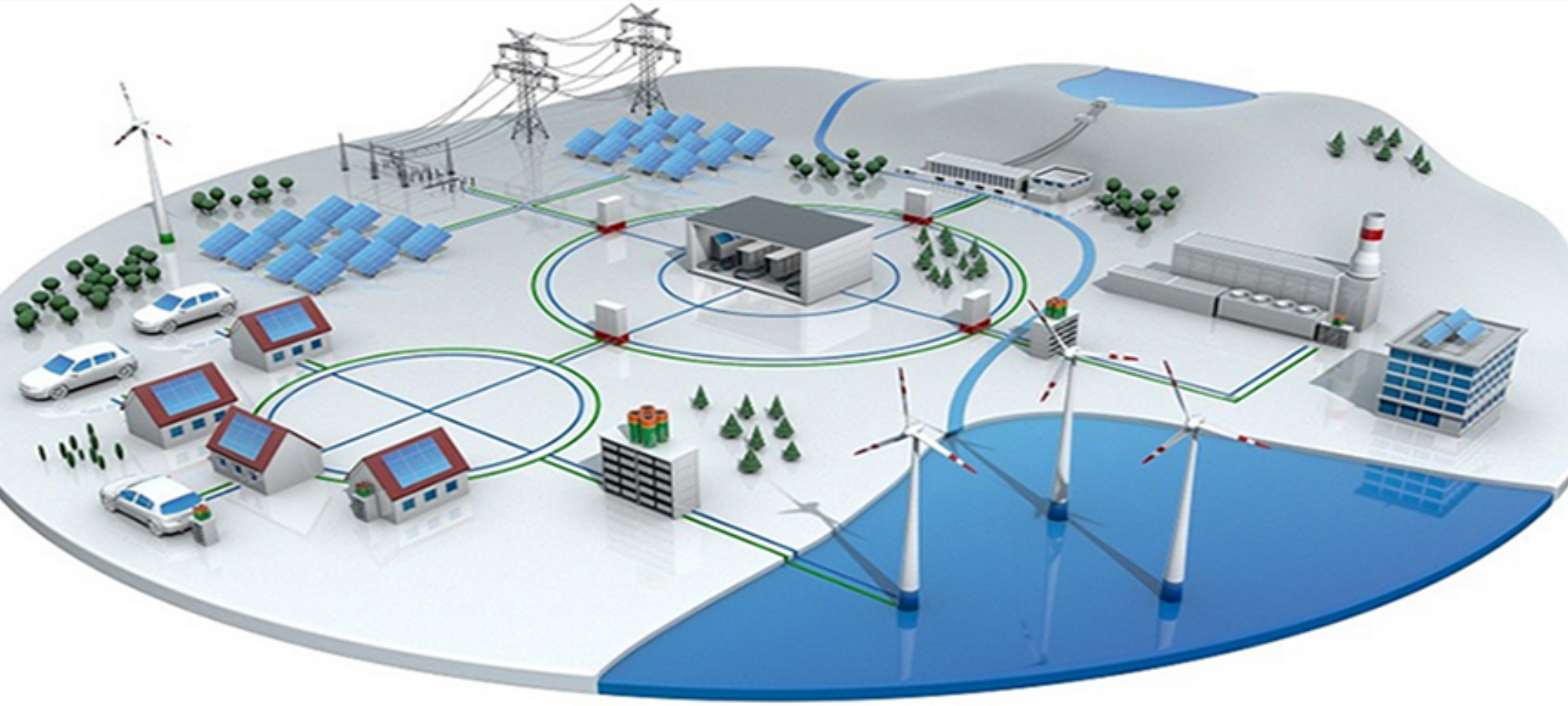
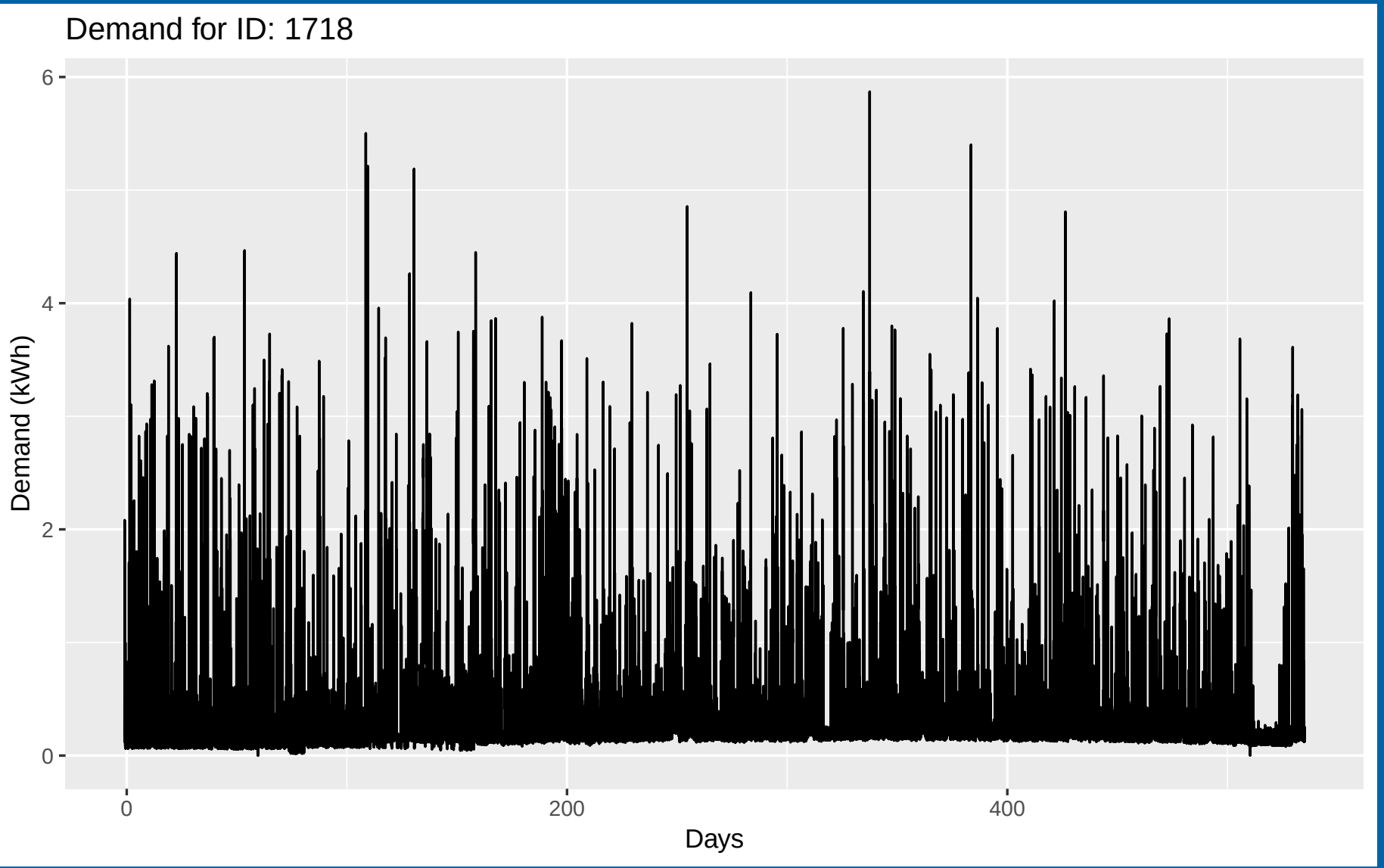


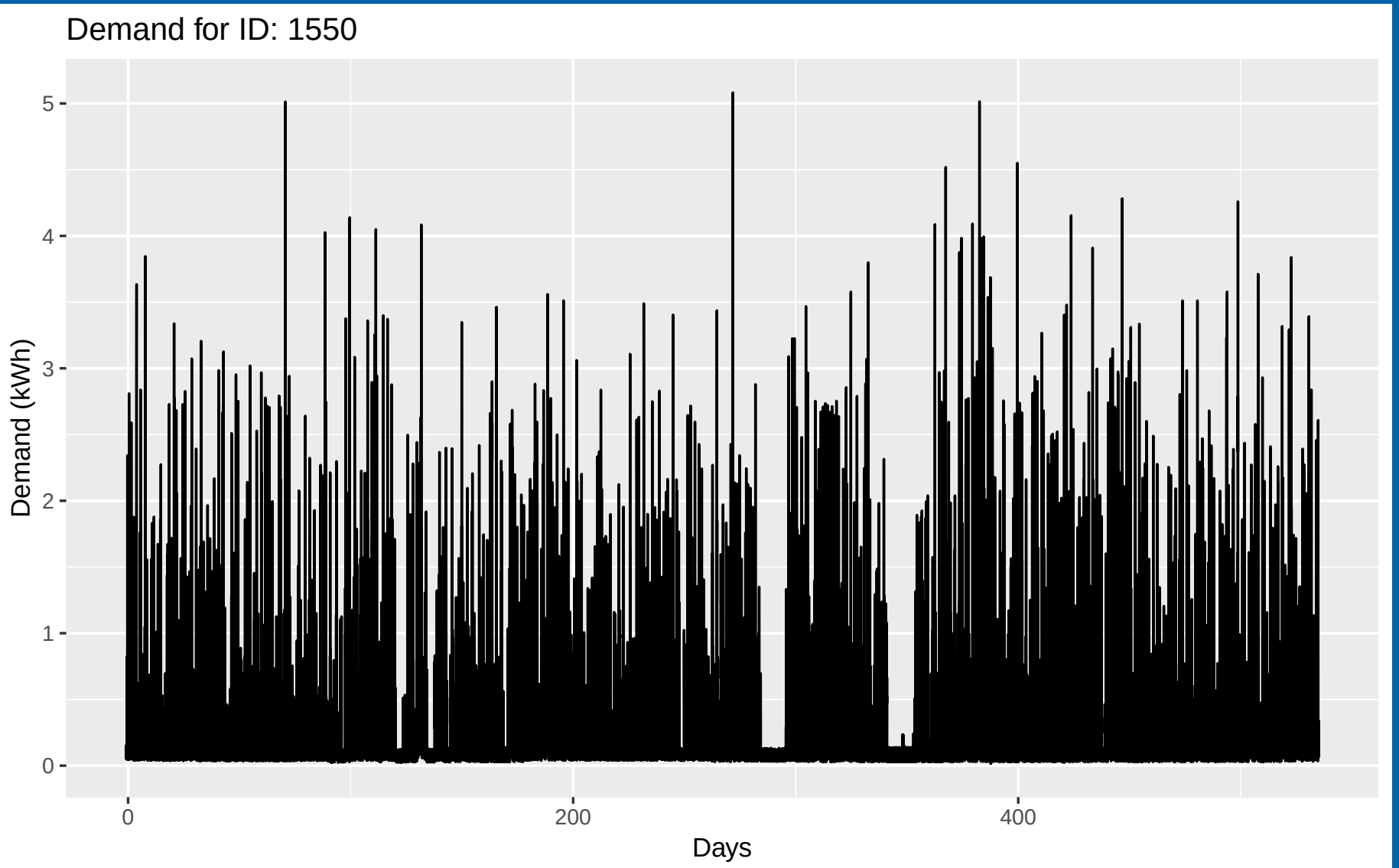
Figure: http://solutions.3m.com

- 500 households from smart metering trial
- Electricity consumption at 30-minute intervals between 14 July 2009 and 31 December 2010
- Heating/cooling energy usage excluded

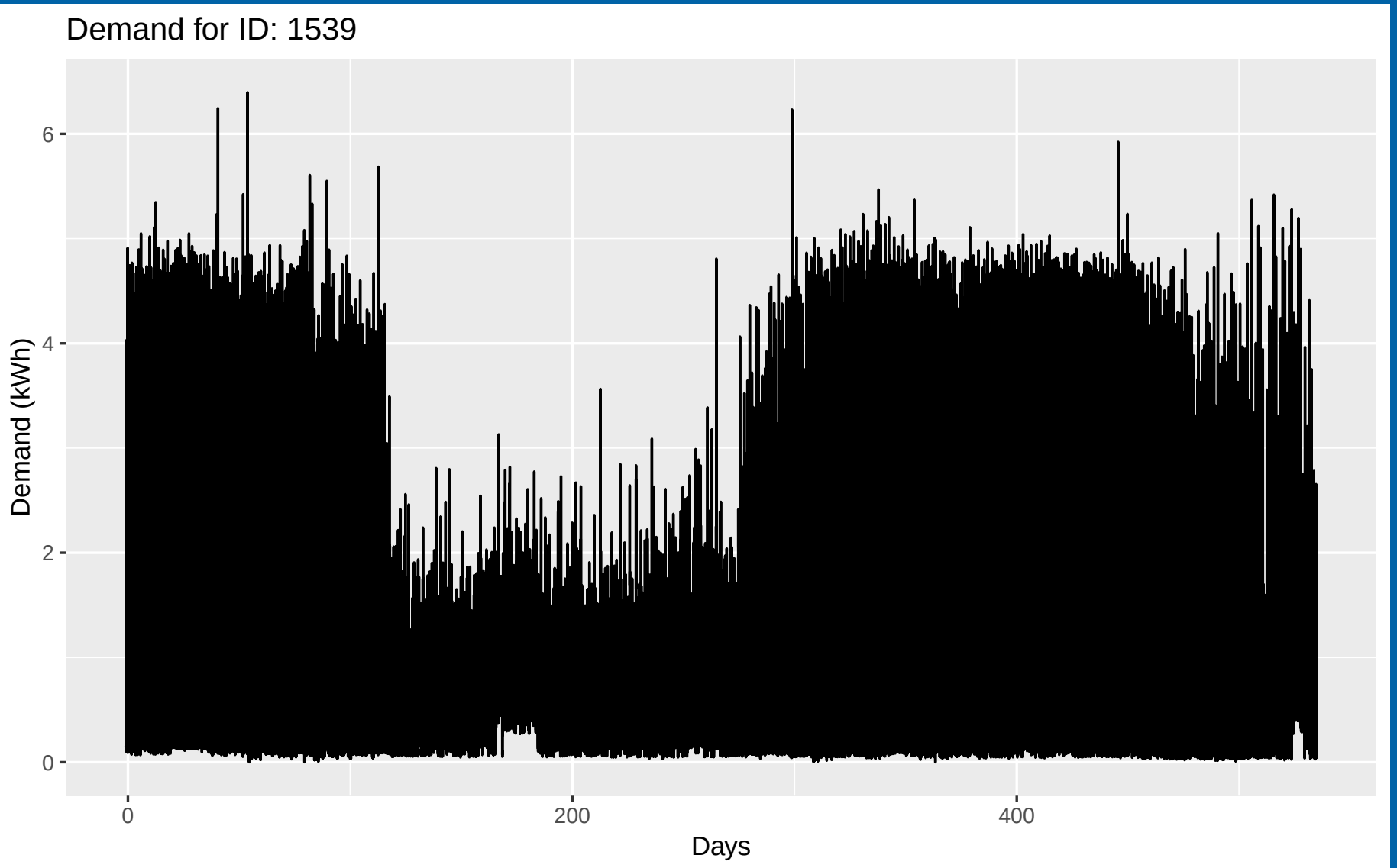
# Irish smart metre data



# Irish smart metre data



# Irish smart metre data

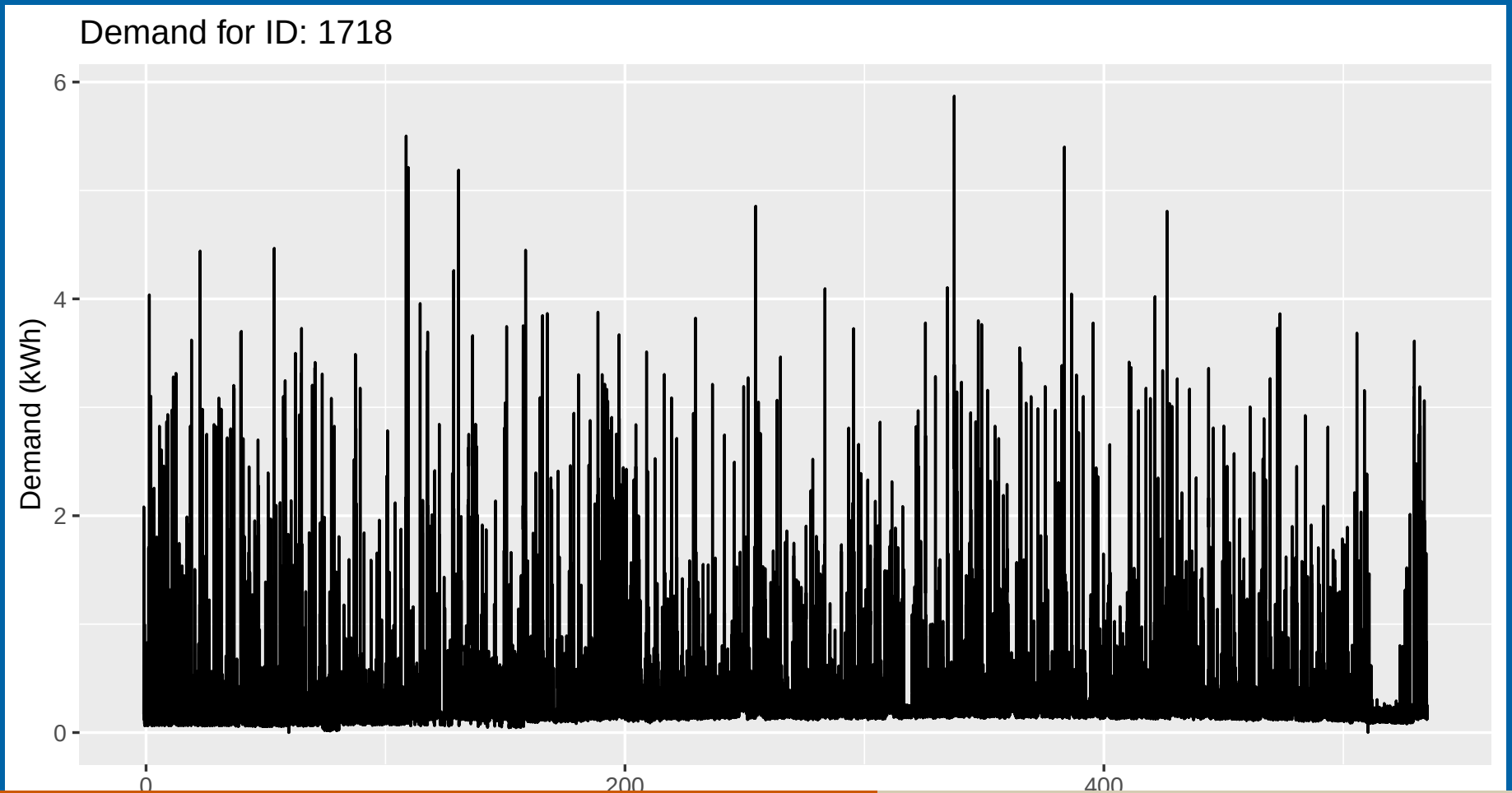


# Outline

1. Time series feature spaces
2. Irish smart metre data
3. Quantiles conditional on time of week
4. Finding typical and unusual households
5. Visualization via embedding
6. Features and limitations

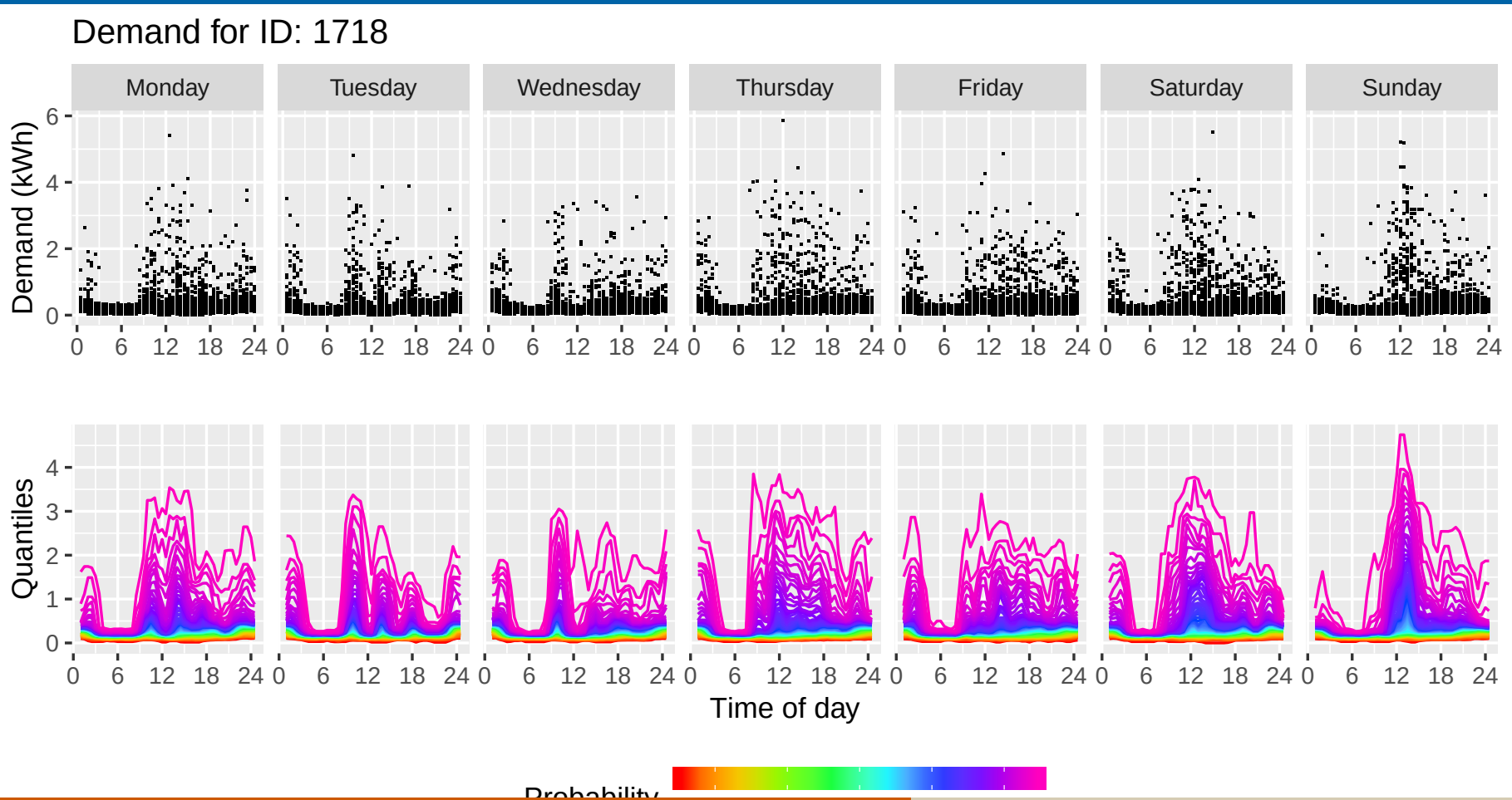
# Quantiles conditional on time of week

- Compute sample quantiles at $p = 0.01, 0.02, \ldots, 0.99$ for each household and each half-hour of the week.
- 336 probability distributions per household.



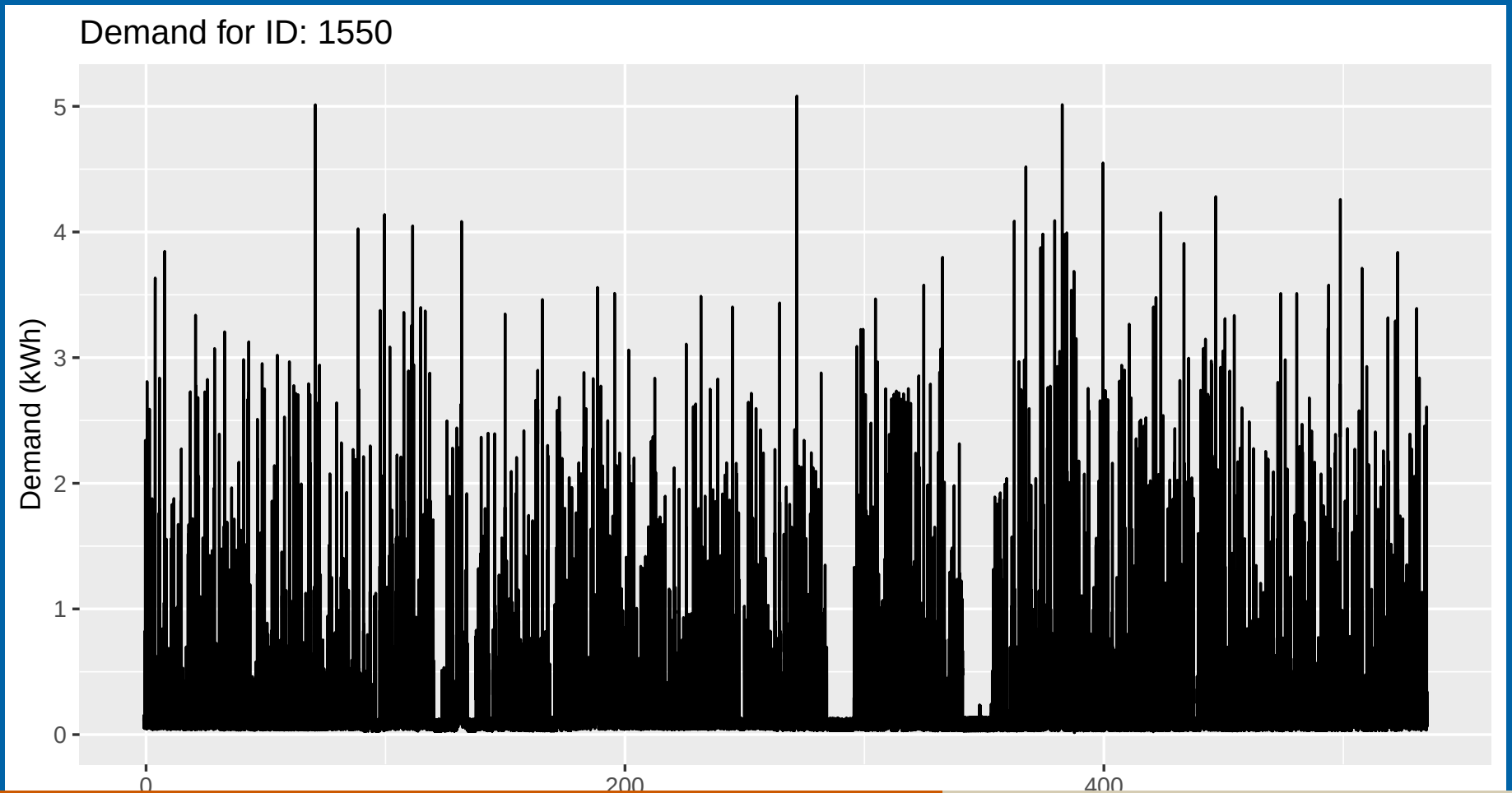
# Quantiles conditional on time of week

- Compute sample quantiles at $p = 0.01, 0.02, \ldots, 0.99$ for each household and each half-hour of the week.
- 336 probability distributions per household.



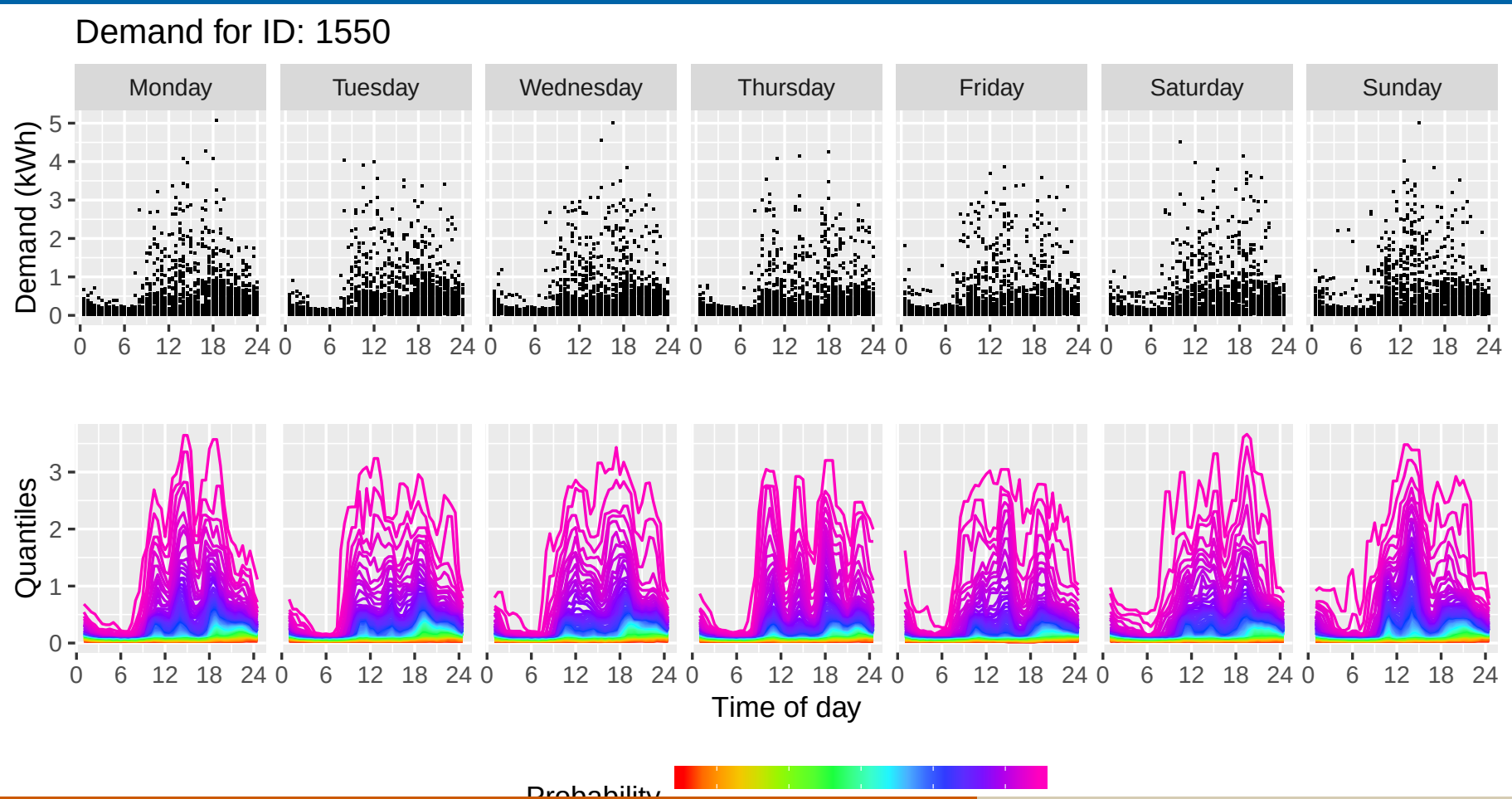
# Quantiles conditional on time of week

- Compute sample quantiles at $p = 0.01, 0.02, \ldots, 0.99$ for each household and each half-hour of the week.
- 336 probability distributions per household.



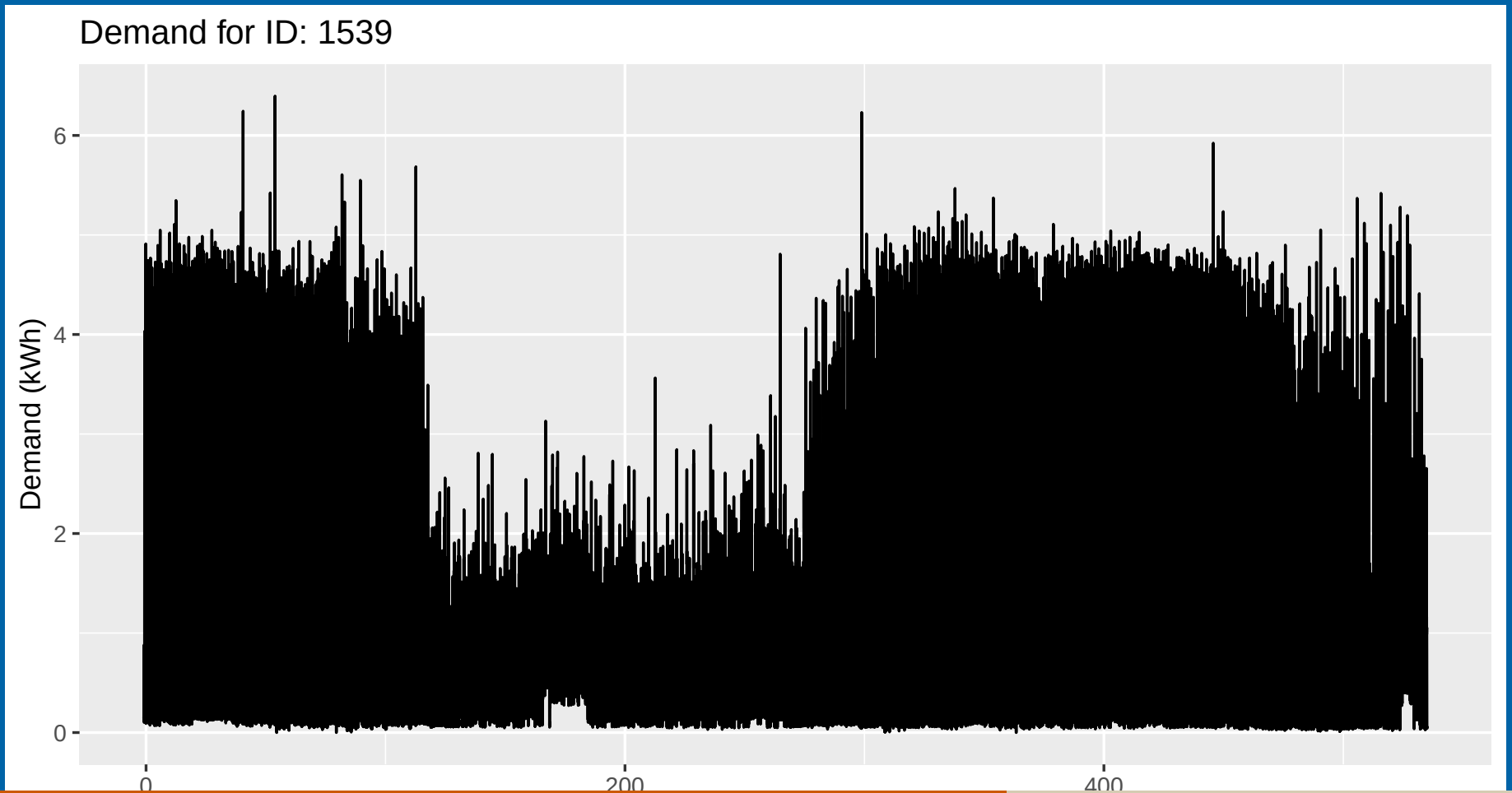
# Quantiles conditional on time of week

- Compute sample quantiles at $p = 0.01, 0.02, \ldots, 0.99$ for each household and each half-hour of the week.
- 336 probability distributions per household.



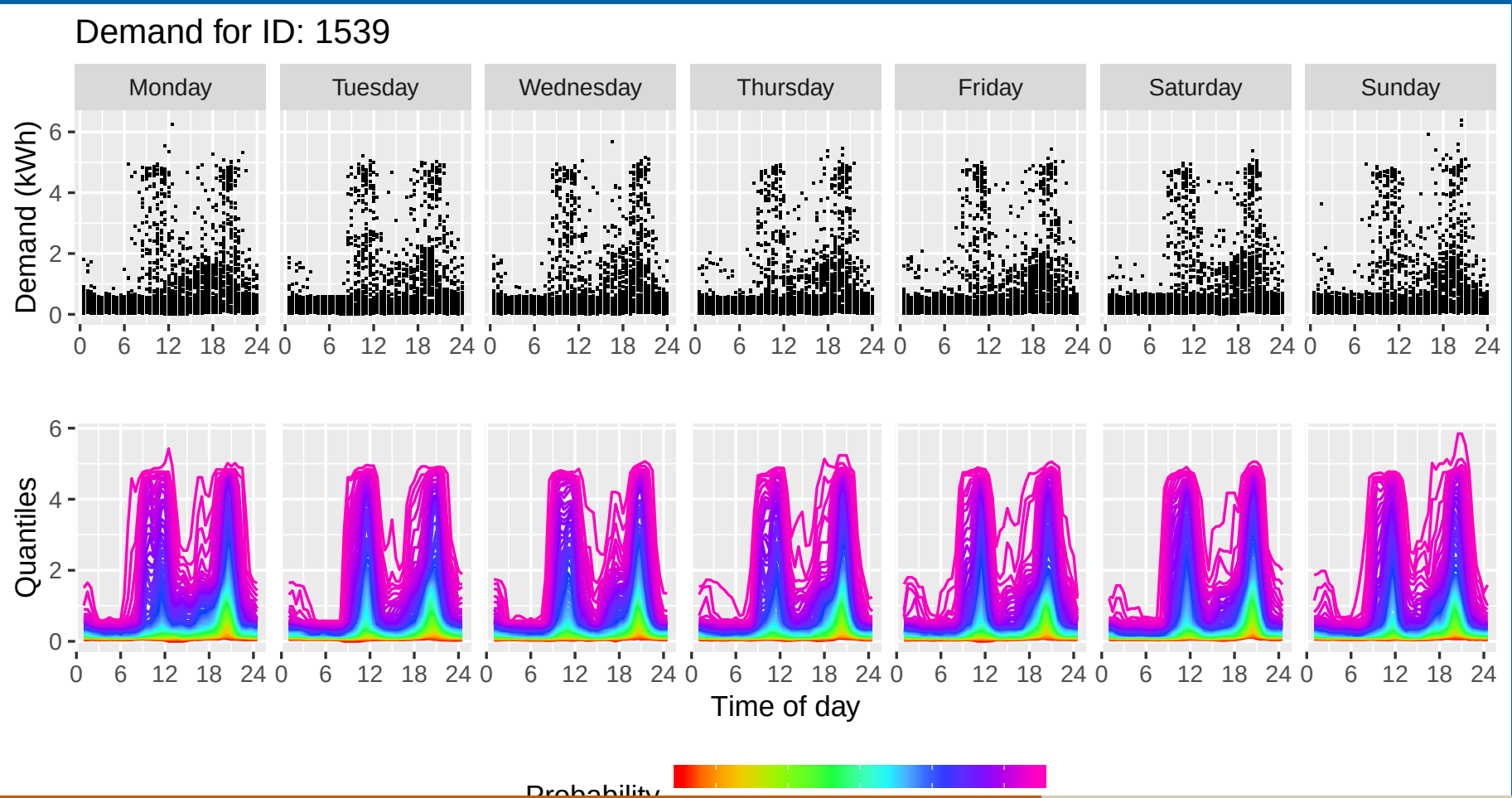
# Quantiles conditional on time of week

- Compute sample quantiles at $p = 0.01, 0.02, \ldots, 0.99$ for each household and each half-hour of the week.
- 336 probability distributions per household.

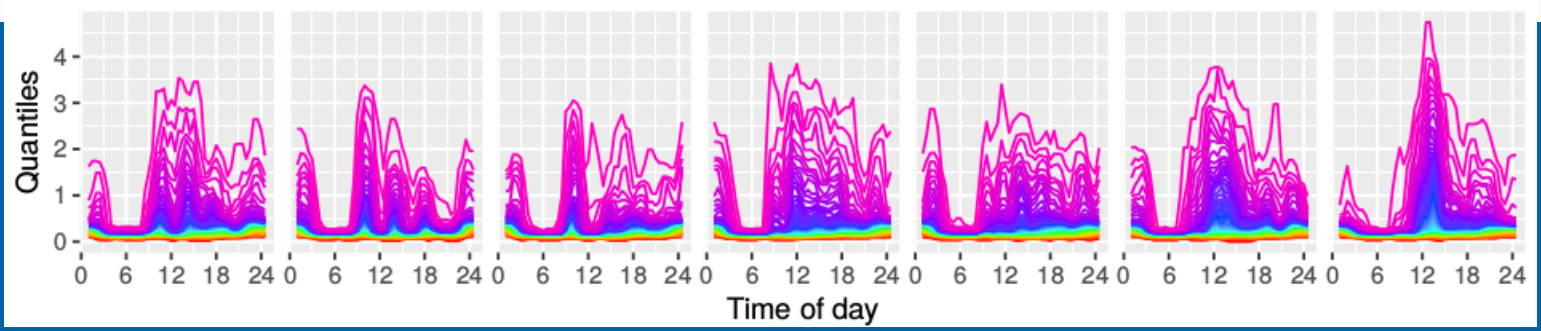


# Quantiles conditional on time of week

- Compute sample quantiles at $p = 0.01, 0.02, \ldots, 0.99$ for each household and each half-hour of the week.
- 336 probability distributions per household.

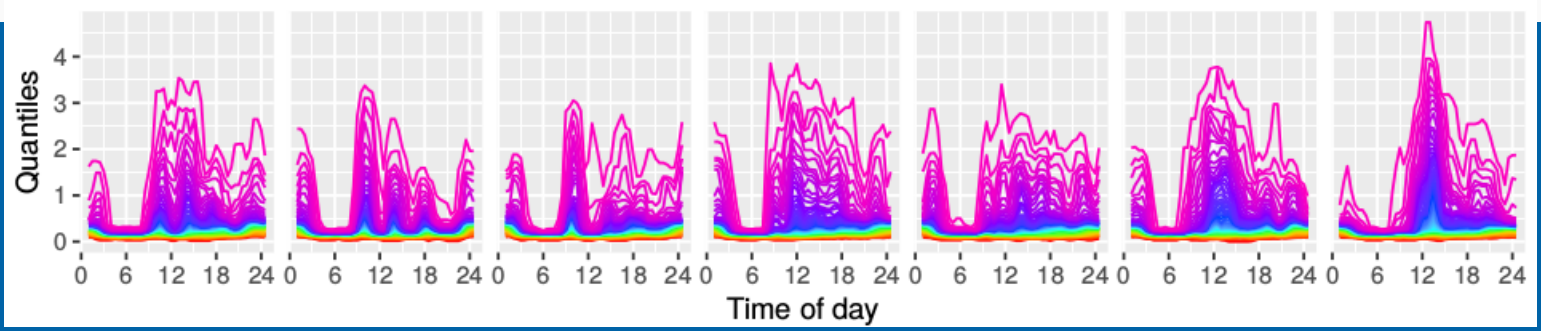


# Quantiles conditional on time of week



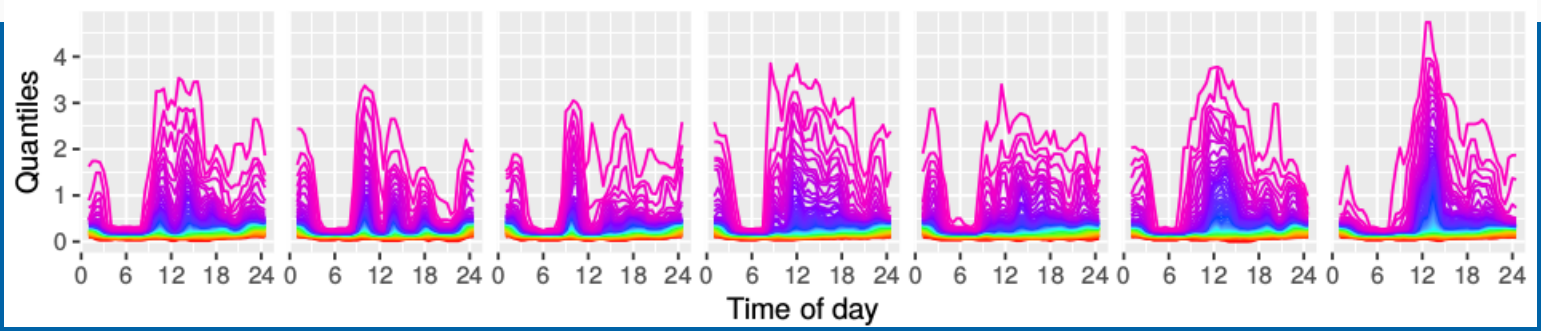

- Sample quantiles better than kernel density estimate:

# Quantiles conditional on time of week



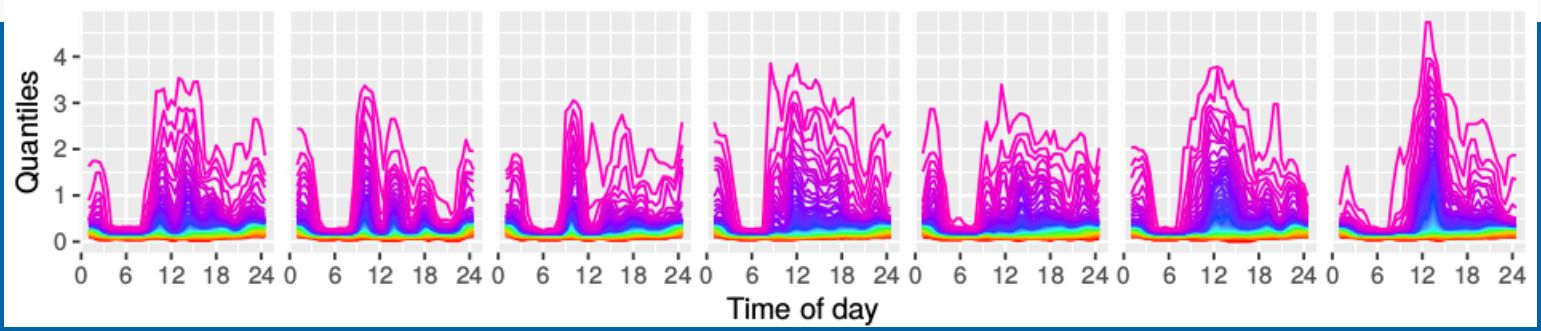

- Sample quantiles better than kernel density estimate:
  - presence of zeros

# Quantiles conditional on time of week



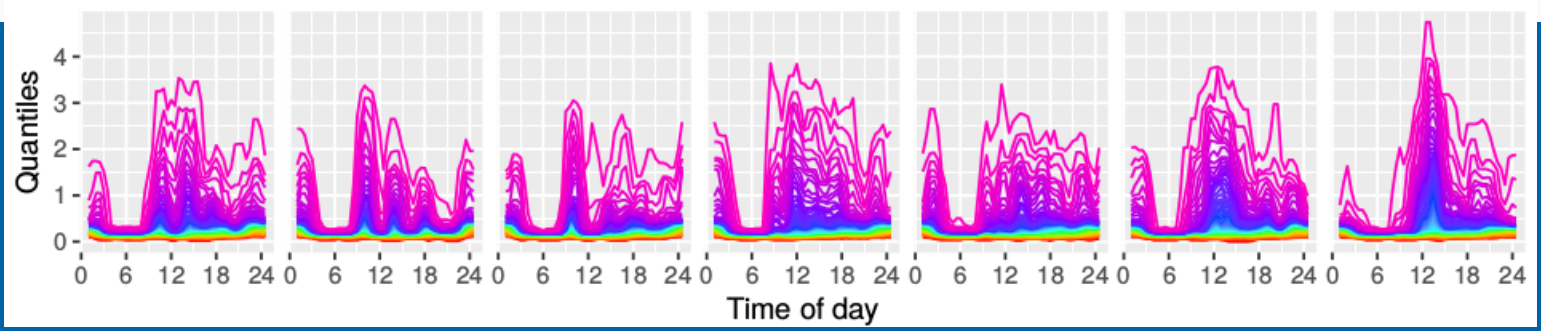

- Sample quantiles better than kernel density estimate:
  - presence of zeros
  - non-negative support

# Quantiles conditional on time of week



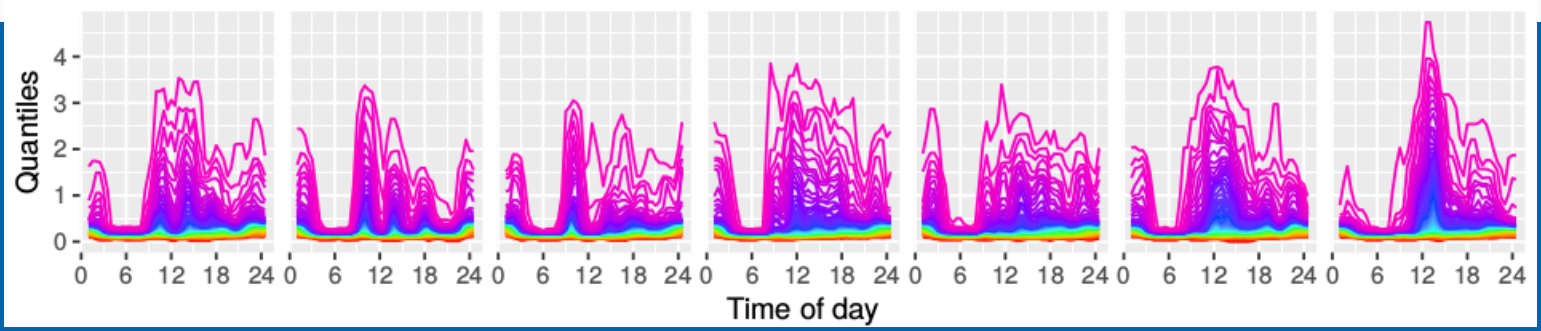

- Sample quantiles better than kernel density estimate:
  - presence of zeros
  - non-negative support
  - high skewness

# Quantiles conditional on time of week



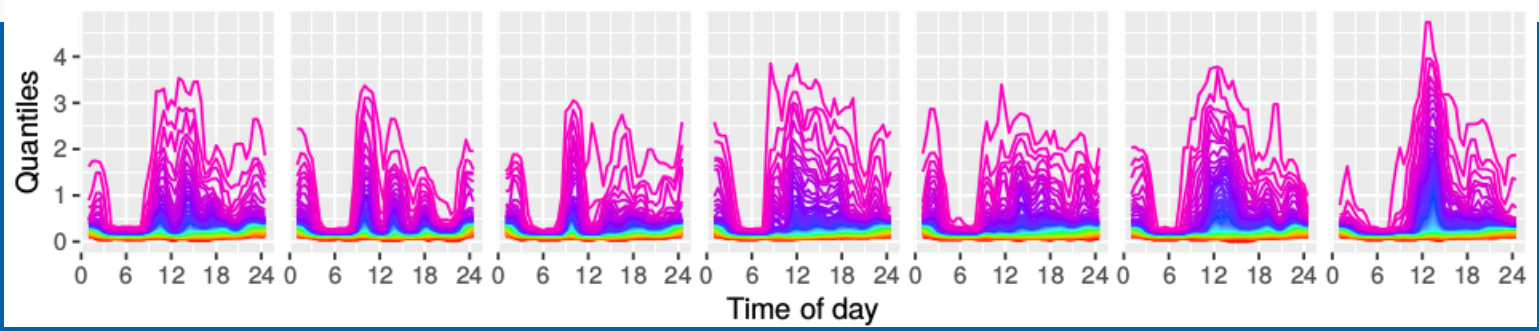

- Sample quantiles better than kernel density estimate:
  - presence of zeros
  - non-negative support
  - high skewness
- Avoids missing data issues and variation in series length

# Quantiles conditional on time of week



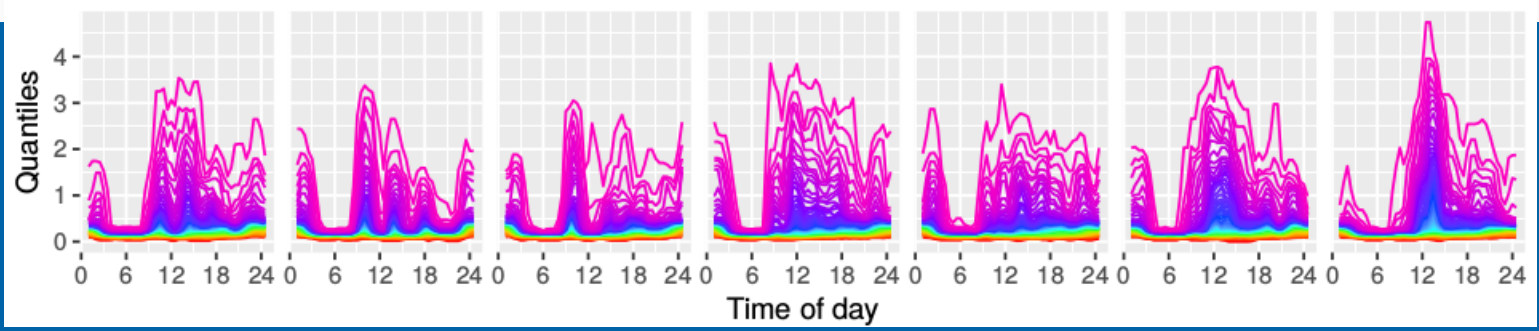

- Sample quantiles better than kernel density estimate:
  - presence of zeros
  - non-negative support
  - high skewness
- Avoids missing data issues and variation in series length
- Avoids timing of household events, holidays, etc.

# Quantiles conditional on time of week



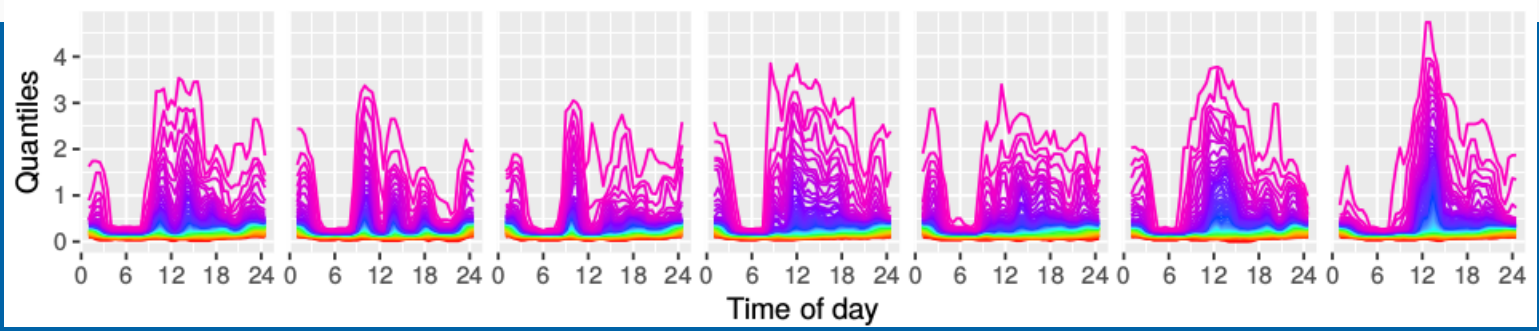

- Sample quantiles better than kernel density estimate:
  - presence of zeros
  - non-negative support
  - high skewness
- Avoids missing data issues and variation in series length
- Avoids timing of household events, holidays, etc.
- Allows clustering of households based on probabilistic behaviour rather than coincident behaviour.

# Quantiles conditional on time of week




- Sample quantiles better than kernel density estimate:
  - presence of zeros
  - non-negative support
  - high skewness
- Avoids missing data issues and variation in series length
- Avoids timing of household events, holidays, etc.
- Allows clustering of households based on probabilistic behaviour rather than coincident behaviour.
- Allows identification of anomalous households.

# Quantiles conditional on time of week

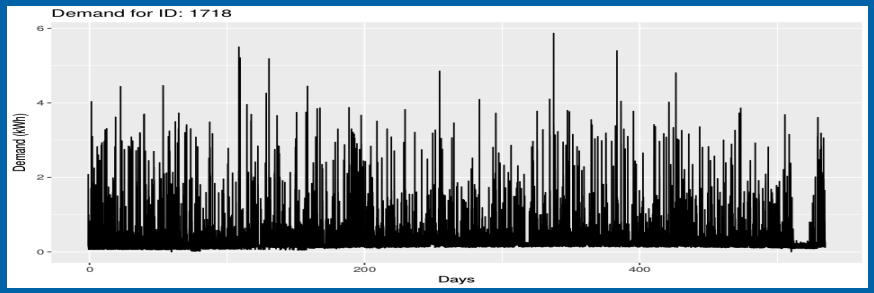


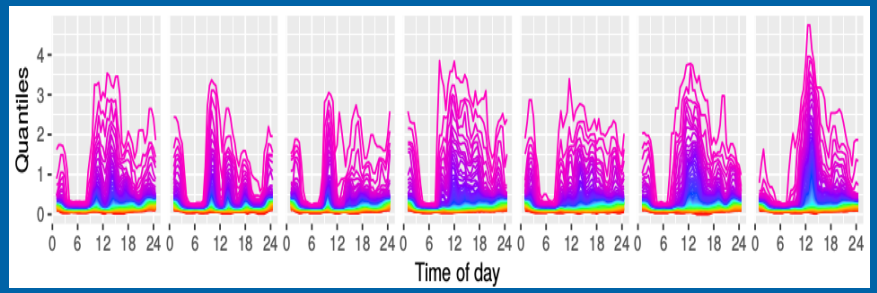

- Sample quantiles better than kernel density estimate:
  - presence of zeros
  - non-negative support
  - high skewness
- Avoids missing data issues and variation in series length
- Avoids timing of household events, holidays, etc.
- Allows clustering of households based on probabilistic behaviour rather than coincident behaviour.
- Allows identification of anomalous households.
- Allows estimation of typical household behaviour.

# Outline

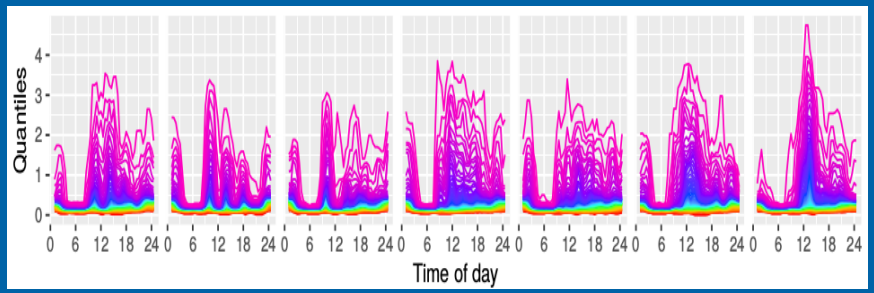
1. Time series feature spaces
2. Irish smart metre data
3. Quantiles conditional on time of week
4. Finding typical and unusual households
5. Visualization via embedding
6. Features and limitations

# Pairwise distances

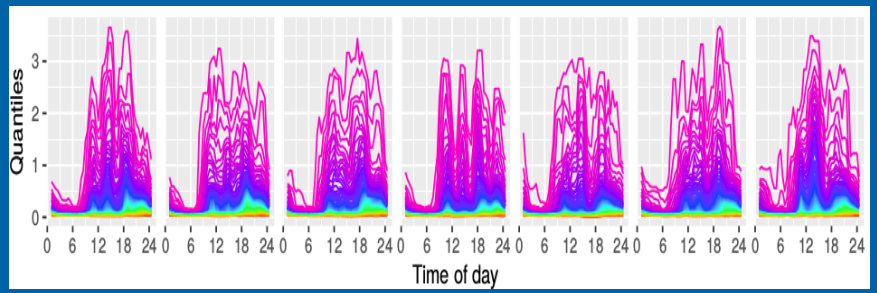



- The time series of $535 \times 48$ observations per household is mapped to a set of $7 \times 48 \times 99$ quantiles giving a bivariate surface for each household.
- Can we compute pairwise distances between all households?




$\leftarrow$ ? $\rightarrow$
Distance



# Jensen-Shannon distances

## Kullback-Leibler divergence between two densities

$$D(p,q) = \int_{\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$$

# Jensen-Shannon distances

## Kullback-Leibler divergence between two densities

$$D(p,q) = \int_{\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$$

Not symmetric: $D(p,q) \neq D(q,p)$

# Jensen-Shannon distances

## Kullback-Leibler divergence between two densities

$$D(p,q) = \int_{\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$$

Not symmetric: $D(p,q) \neq D(q,p)$

## Jensen-Shannon distance between two densities

$$\mathrm{JS}(p,q) = [D(p,r) + D(q,r)]/2 \qquad \text{where } r = (p+q)/2$$

# Jensen-Shannon distances

## Kullback-Leibler divergence between two densities

$$D(p,q) = \int_{\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$$

Not symmetric: $D(p,q) \neq D(q,p)$

## Jensen-Shannon distance between two densities

$$\mathrm{JS}(p,q) = [D(p,r) + D(q,r)]/2 \qquad \text{where } r = (p+q)/2$$

## Distance between two households

$$\Delta_{ij} = \sum_{t=1}^{7\times 48} \mathrm{JS}(p_t, q_t)$$

# Kernel matrix and density ranking

## Similarity between two households

$$w_{ij} = \exp(-\Delta_{ij}^2/h^2).$$

# Kernel matrix and density ranking

## Similarity between two households

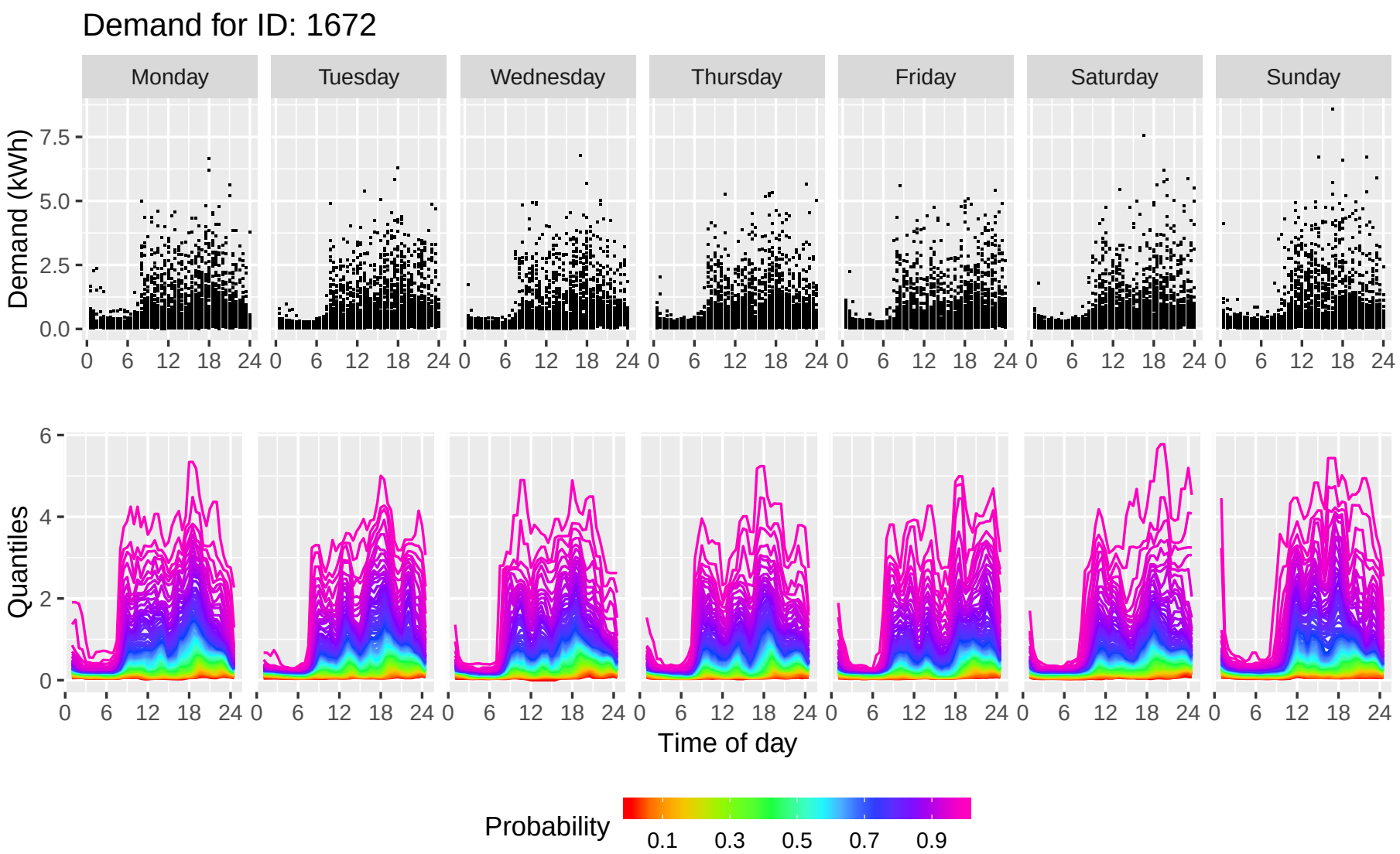
$$w_{ij} = \exp(-\Delta_{ij}^2/h^2).$$

Row sums of the kernel matrix gives a scaled kernel density estimate of households:

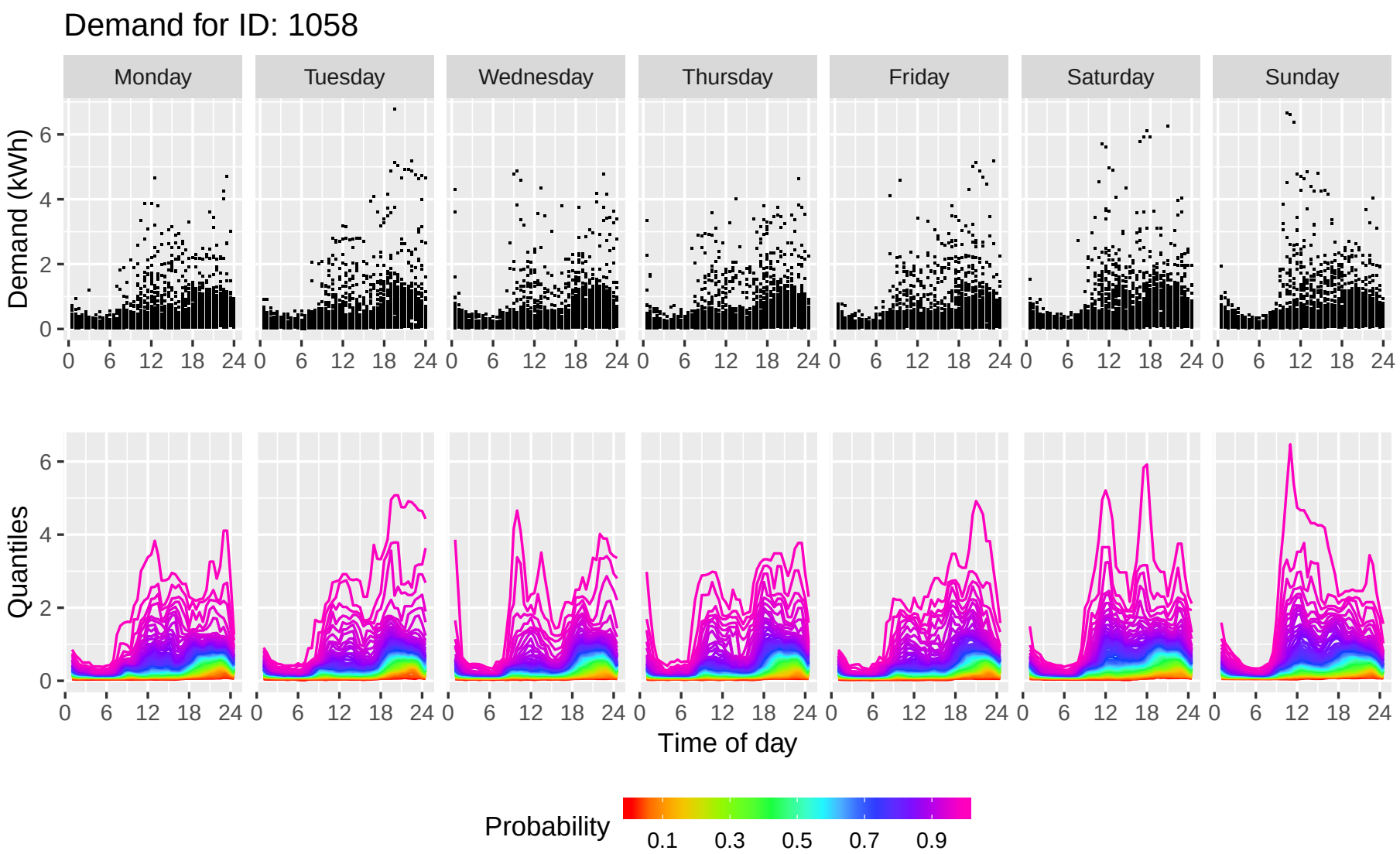
$$\hat{f}_i = \sum_{j=1}^{n} w_{ij}$$

- $h$ is bandwidth in Gaussian kernel.
- Households can be ranked by density values.

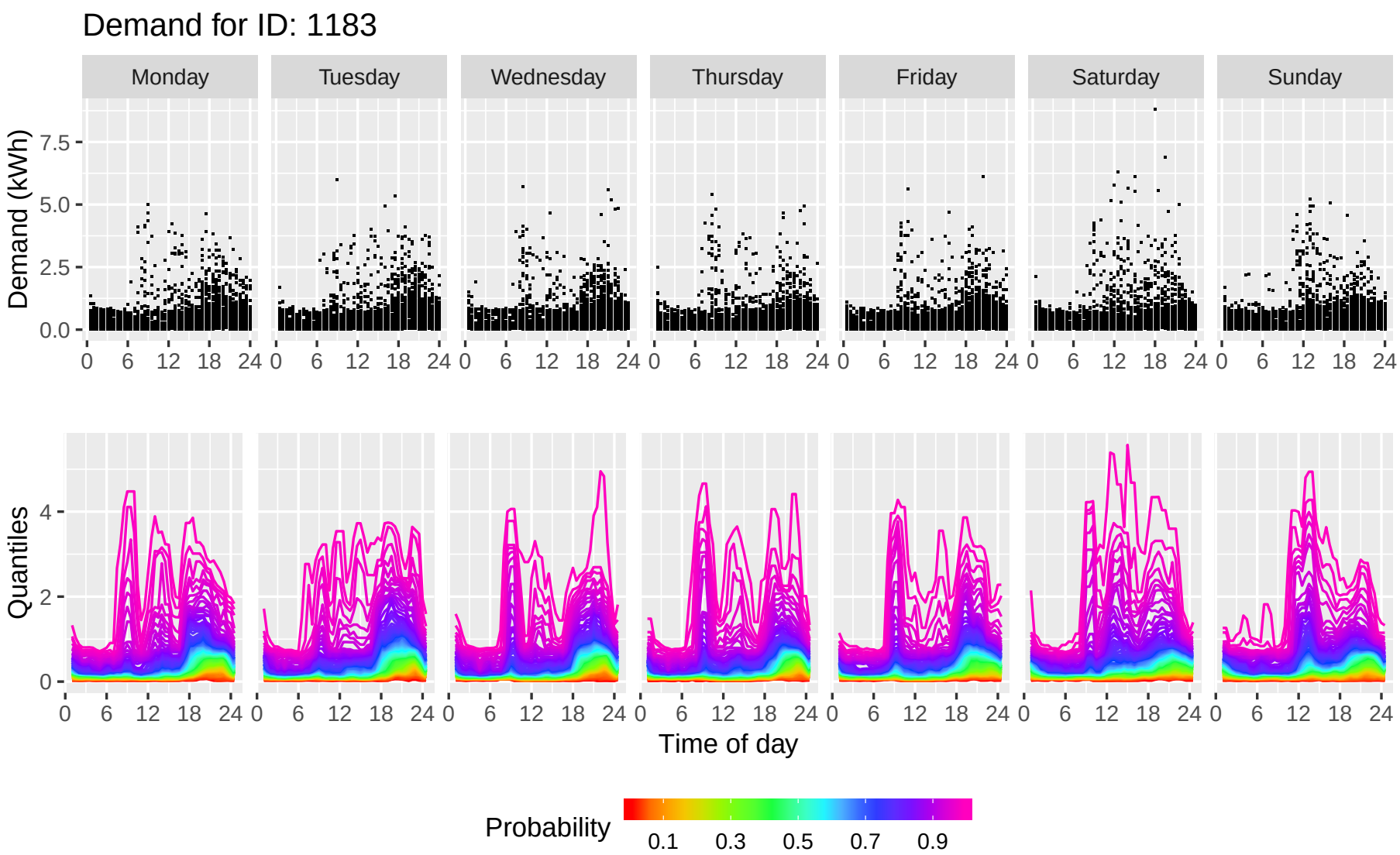
# Typical households



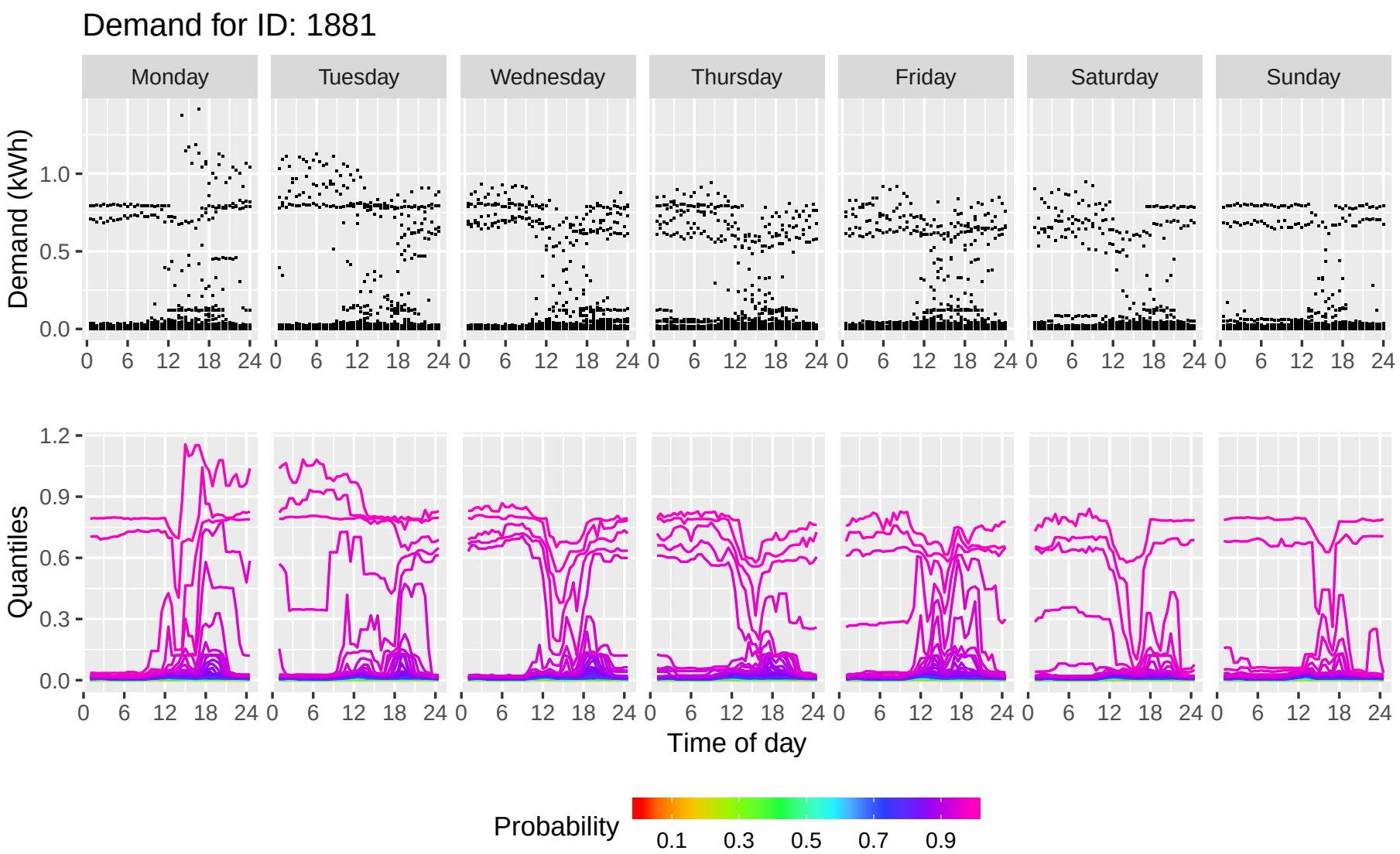
# Typical households



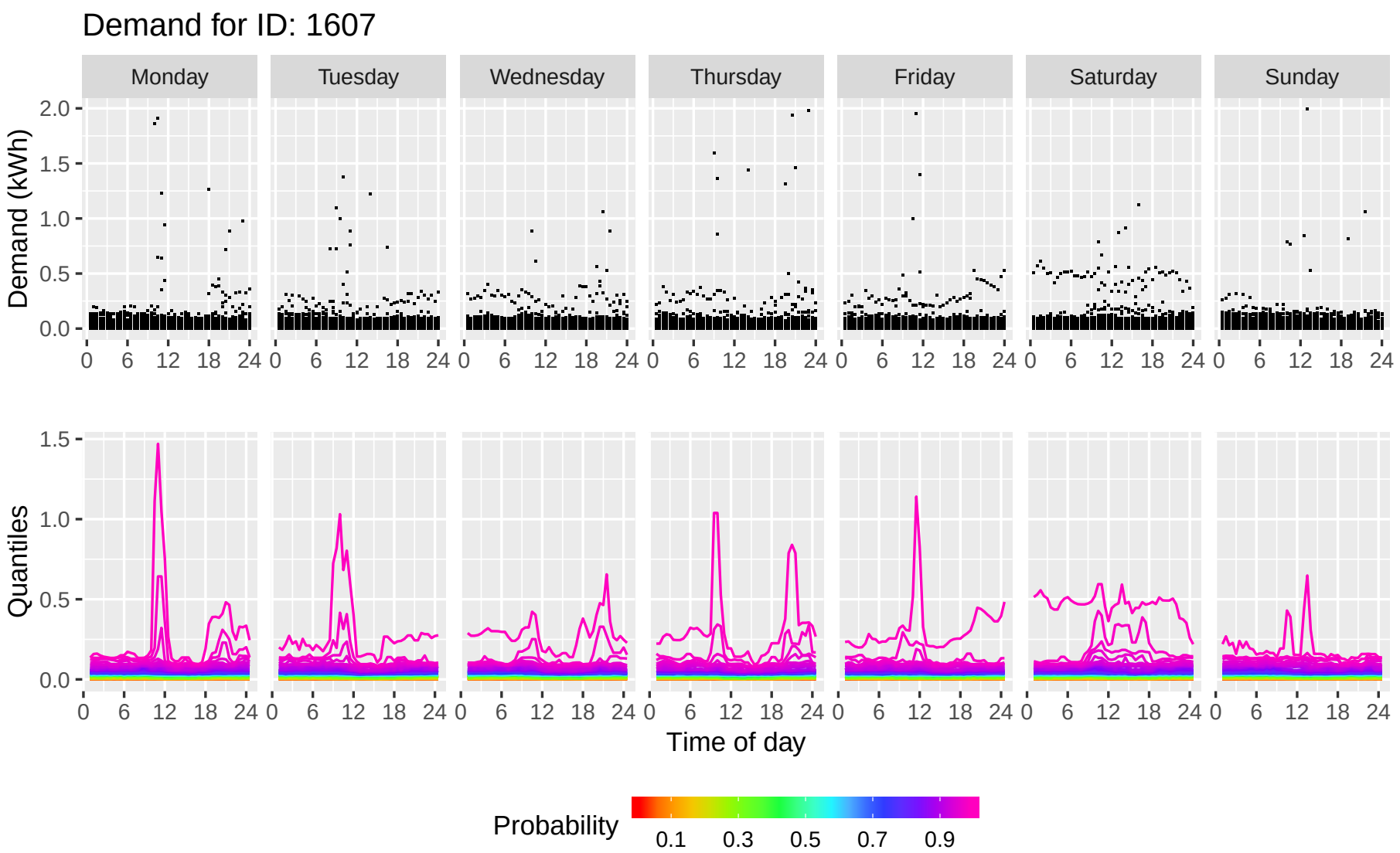
# Typical households



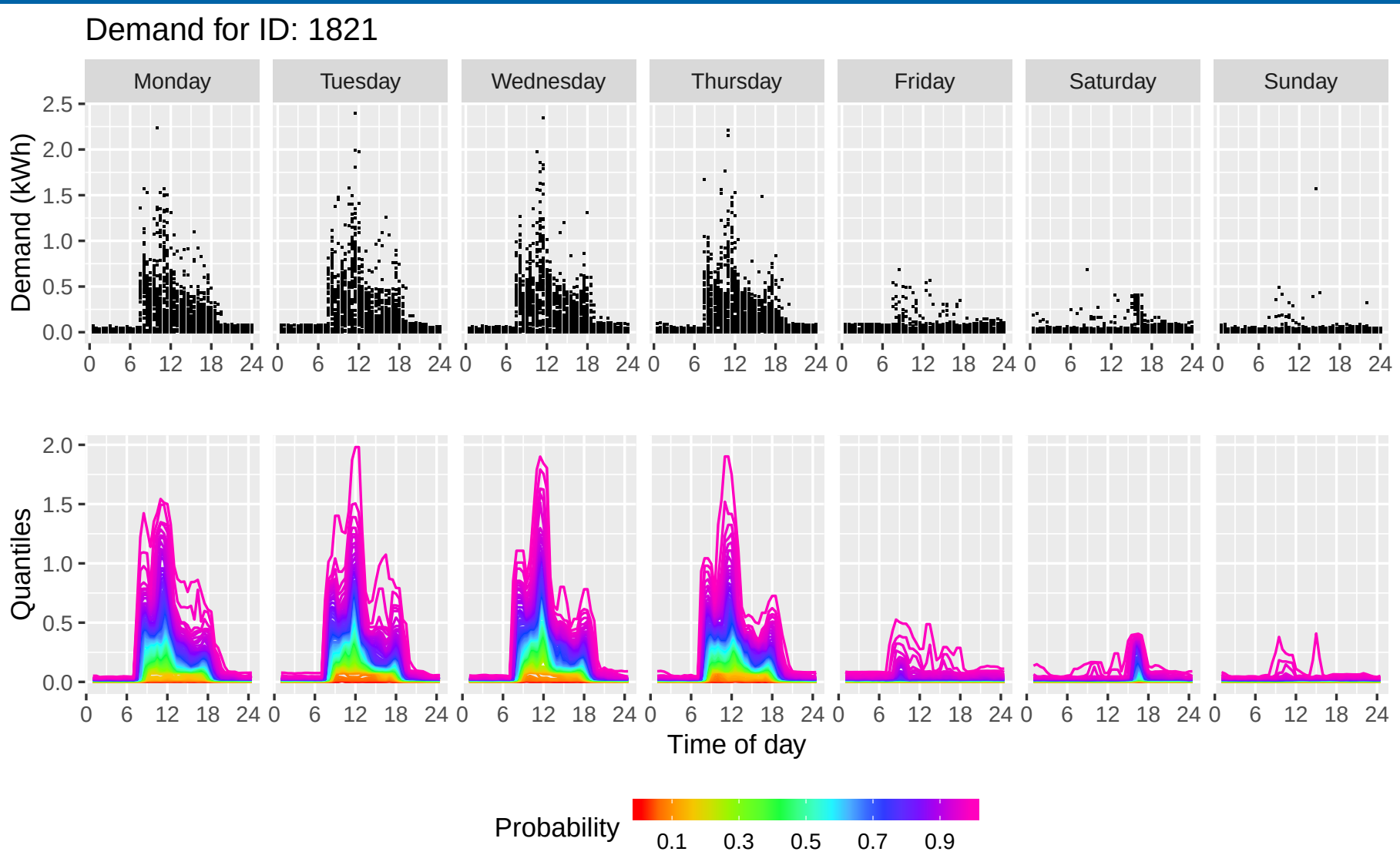
# Anomalous households



# Anomalous households



# Anomalous households



# Outline

1. Time series feature spaces
2. Irish smart metre data
3. Quantiles conditional on time of week
4. Finding typical and unusual households
5. Visualization via embedding
6. Features and limitations

# Laplacian eigenmaps

- **Idea:** Embed conditional densities in a 2d space where the distances are preserved “as far as possible”.

# Laplacian eigenmaps

- **Idea:** Embed conditional densities in a 2d space where the distances are preserved "as far as possible".
- Let $\boldsymbol{W} = [w_{ij}]$ where $w_{ij} = \exp(-\Delta_{ij}^2/h^2)$.

  $\boldsymbol{D} = \mathrm{diag}(\hat{f}_i)$ where $\hat{f}_i = \sum_{j=1}^{n} w_{ij}$

  $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ (the Laplacian matrix).

# Laplacian eigenmaps

- **Idea:** Embed conditional densities in a 2d space where the distances are preserved "as far as possible".
- Let $\boldsymbol{W} = [w_{ij}]$ where $w_{ij} = \exp(-\Delta_{ij}^2/h^2)$.

  $\boldsymbol{D} = \mathrm{diag}(\hat{f}_i)$ where $\hat{f}_i = \sum_{j=1}^{n} w_{ij}$

  $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ (the Laplacian matrix).
- Solve generalized eigenvector problem: $\boldsymbol{Le} = \lambda \boldsymbol{De}$.

# Laplacian eigenmaps

- **Idea:** Embed conditional densities in a 2d space where the distances are preserved "as far as possible".
- Let $\boldsymbol{W} = [w_{ij}]$ where $w_{ij} = \exp(-\Delta_{ij}^2/h^2)$.

  $\boldsymbol{D} = \mathrm{diag}(\hat{f}_i)$ where $\hat{f}_i = \sum_{j=1}^{n} w_{ij}$

  $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ (the Laplacian matrix).
- Solve generalized eigenvector problem: $\boldsymbol{Le} = \lambda \boldsymbol{De}$.
- Let $\boldsymbol{e}_k$ be eigenvector corresponding to $k$th *smallest* eigenvalue.

# Laplacian eigenmaps

- **Idea:** Embed conditional densities in a 2d space where the distances are preserved "as far as possible".
- Let $\boldsymbol{W} = [w_{ij}]$ where $w_{ij} = \exp(-\Delta_{ij}^2/h^2)$.

  $\boldsymbol{D} = \mathrm{diag}(\hat{f}_i)$ where $\hat{f}_i = \sum_{j=1}^{n} w_{ij}$

  $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ (the Laplacian matrix).
- Solve generalized eigenvector problem: $\boldsymbol{Le} = \lambda \boldsymbol{De}$.
- Let $\boldsymbol{e}_k$ be eigenvector corresponding to $k$th *smallest* eigenvalue.
- Then $\boldsymbol{e}_2$ and $\boldsymbol{e}_3$ create an embedding of households in 2d space.

# Key property of Laplacian embedding

Let $y_i = (e_{2,i}, e_{3,i})$ be the embedded point corresponding to household $i$.

Then the Laplacian eigenmap minimizes

$$\sum_{ij} w_{ij}(y_i - y_j)^2 = \boldsymbol{y}'\boldsymbol{L}\boldsymbol{y} \qquad \text{such that} \quad \boldsymbol{y}'\boldsymbol{D}\boldsymbol{y} = 1.$$

# Key property of Laplacian embedding

Let $y_i = (e_{2,i}, e_{3,i})$ be the embedded point corresponding to household $i$.

Then the Laplacian eigenmap minimizes

$$\sum_{ij} w_{ij}(y_i - y_j)^2 = \boldsymbol{y}'\boldsymbol{L}\boldsymbol{y} \qquad \text{such that} \quad \boldsymbol{y}'\boldsymbol{D}\boldsymbol{y} = 1.$$

- the most similar points are as close as possible.

# Key property of Laplacian embedding

Let $y_i = (e_{2,i}, e_{3,i})$ be the embedded point corresponding to household $i$.

Then the Laplacian eigenmap minimizes

$$\sum_{ij} w_{ij}(y_i - y_j)^2 = \boldsymbol{y}'\boldsymbol{L}\boldsymbol{y} \qquad \text{such that} \quad \boldsymbol{y}'\boldsymbol{D}\boldsymbol{y} = 1.$$

- the most similar points are as close as possible.
- First eigenvalue is 0 due to translation invariance.

# Key property of Laplacian embedding

Let $y_i = (e_{2,i}, e_{3,i})$ be the embedded point corresponding to household $i$.

Then the Laplacian eigenmap minimizes

$$\sum_{ij} w_{ij}(y_i - y_j)^2 = \boldsymbol{y}'\boldsymbol{L}\boldsymbol{y} \qquad \text{such that} \quad \boldsymbol{y}'\boldsymbol{D}\boldsymbol{y} = 1.$$

- the most similar points are as close as possible.
- First eigenvalue is 0 due to translation invariance.
- Equivalent to optimal embedding using Laplace-Beltrami operator on manifolds.

# Outliers computed in embedded space:

# Outline

1. Time series feature spaces
2. Irish smart metre data
3. Quantiles conditional on time of week
4. Finding typical and unusual households
5. Visualization via embedding
6. Features and limitations

# Features and limitations

## Features of approach

- Converting time series to quantile surfaces conditional on time of week.
- Using pairwise distances between households
- Using kernel matrices for density ranking, embedding and clustering

# Features and limitations

## Features of approach

- Converting time series to quantile surfaces conditional on time of week.
- Using pairwise distances between households
- Using kernel matrices for density ranking, embedding and clustering

### Unresolved issues

- Need to select the bandwidth *h* in constructing the similarity matrix.
- Two different uses of bandwidth: density-ranking, embedding. Different bandwidth in each case?
- The use of pairwise distances makes it hard to scale this algorithm.