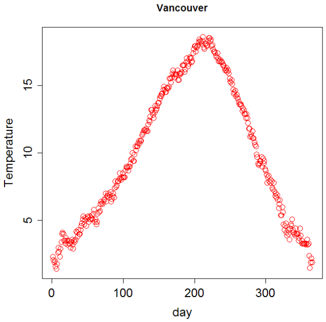
From Data To Functions

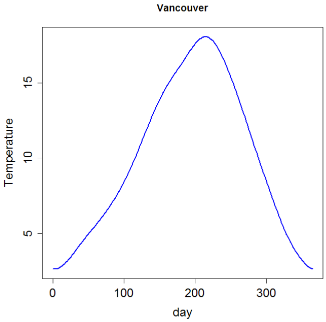
How do we go from

data



to

functions?



Fundamental Model

$$y_i = f(t_i) + \epsilon_i$$

- Data: $y_1, y_2, \ldots, y_n$

Fundamental Model

$$y_i = f(t_i) + \epsilon_i$$

- Data: $y_1, y_2, \ldots, y_n$
- Measurement errors / Noises: $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$

Fundamental Model

$$y_i = f(t_i) + \epsilon_i$$

- Data: $y_1, y_2, \ldots, y_n$
- Measurement errors / Noises: $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$
- We assume $\epsilon_i \sim \mathrm{Normal}(0, \sigma^2)$ and ϵ_i are independent.

Fundamental Model

$$y_i = f(t_i) + \epsilon_i$$

- Data: $y_1, y_2, \ldots, y_n$
- Measurement errors / Noises: $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$
- We assume $\epsilon_i \sim \mathrm{Normal}(0, \sigma^2)$ and ϵ_i are independent.
- We do not have the parametric form of $f(t)$.

Fundamental Model

$$y_i = f(t_i) + \epsilon_i$$

- Data: $y_1, y_2, \ldots, y_n$
- Measurement errors / Noises: $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$
- We assume $\epsilon_i \sim \mathrm{Normal}(0, \sigma^2)$ and ϵ_i are independent.
- We do not have the parametric form of $f(t)$.
- Question: How estimate $f(t)$ from the noisy and discrete data?

Basis Expansions

$$f(t) = \sum_{j=1}^{K} c_j \phi_j(t)$$

- $\phi_1(t), \ldots, \phi_J(t)$ are called basis functions

Basis Expansions

$$f(t) = \sum_{j=1}^{K} c_j \phi_j(t)$$

- $\phi_1(t), \dots, \phi_J(t)$ are called basis functions
- $c_1, \dots, c_J$ are called coefficients to basis functions

Basis Expansions

$$f(t) = \sum_{j=1}^{K} c_j \phi_j(t)$$

- $\phi_1(t), \ldots, \phi_J(t)$ are called basis functions
- $c_1, \ldots, c_J$ are called coefficients to basis functions
- Question 1: How to decide basis functions?

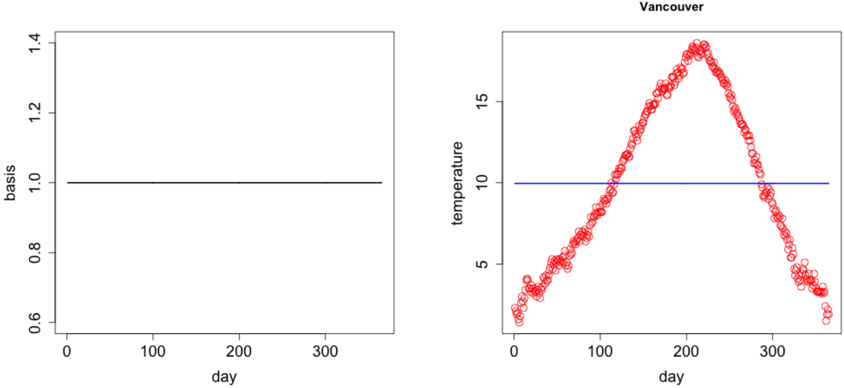
Basis Expansions

$$f(t) = \sum_{j=1}^{K} c_j \phi_j(t)$$

- $\phi_1(t), \ldots, \phi_J(t)$ are called basis functions
- $c_1, \ldots, c_J$ are called coefficients to basis functions
- Question 1: How to decide basis functions?
- Question 2: How to decide coefficients to basis functions

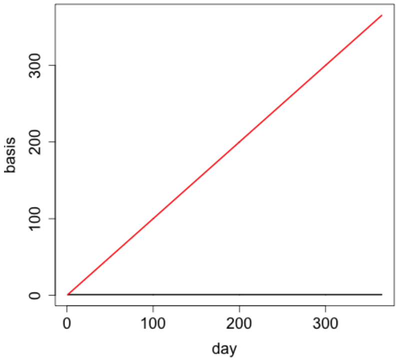
The Monomial Basis

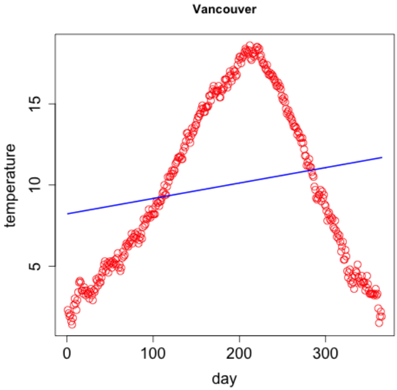
$$\Phi(t) = (1)$$



The Monomial Basis

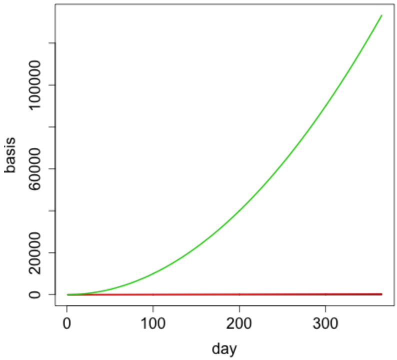
$$\Phi(t) = (1, t)$$

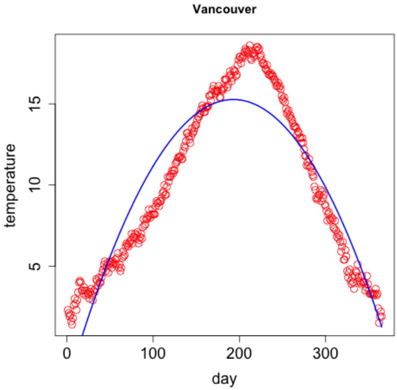




The Monomial Basis

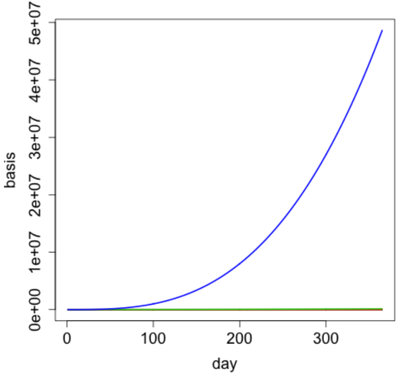
$$\Phi(t) = (1, t, t^2)$$





The Monomial Basis

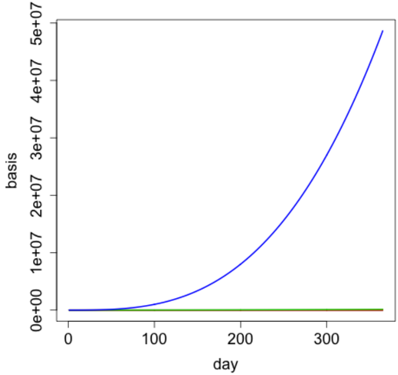
$$\Phi(t) = (1, t, t^2, t^3)$$




```
+       yhat = X[,1:i]%*%solve(t(X[,1:i])%*%X[,1:i])%*%(t(X[,1:i])%*%y)
+
+       png(name2)
+       plot(1:365,y,col=2,cex=1.5,xlab='day',ylab='temperature',cex.lab=1.5,
+               main='Vancouver',cex.axis=1.5)
+       lines(1:365,yhat,lwd=2,col=4)
+       dev.off()
+ }
Error in solve.default(t(X[, 1:i]) %*% X[, 1:i]) :
  system is computationally singular: reciprocal condition number = 1.75329e-16
>
```

Problems with the Monomial Basis

Numerically difficult for more than four basis functions!

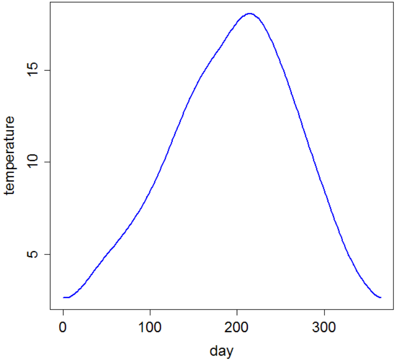


Larger terms over-run smaller ones; especially with unevenly-spaced observations.

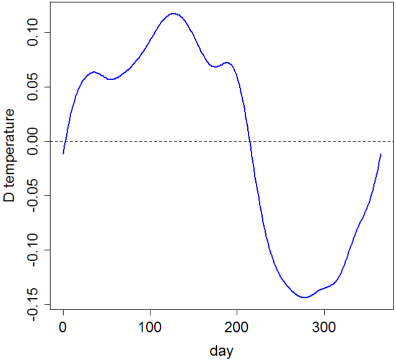
Problems with the Monomial Basis

We are often interested in *rates of change*

Function



Derivative

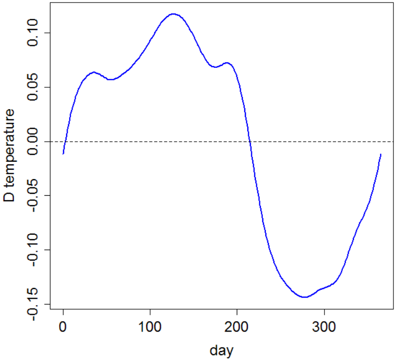


Problems with the Monomial Basis

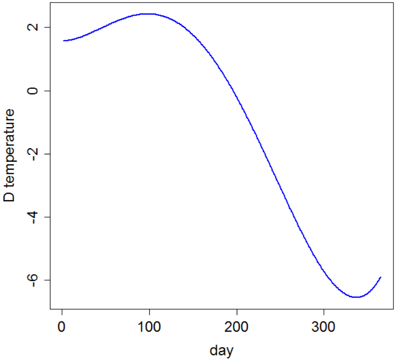
But monomial derivatives get simpler:

$$f(t) = \sum_{k=0}^{K} c_k t^k, \quad Df(t) = \sum_{k=1}^{K-1} c_k k t^{k-1}$$

Derivative

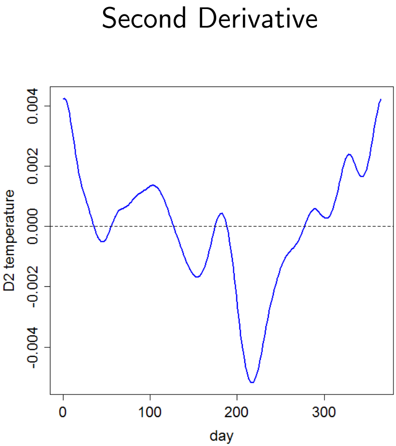


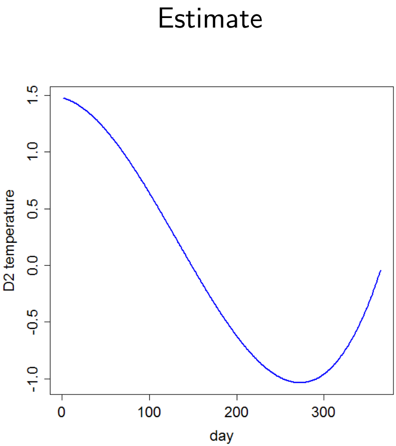
Estimate



Problems with the Monomial Basis

Whereas the opposite happens in most real-world data:





The Fourier Basis

- basis functions are sine and cosine functions of increasing frequency:

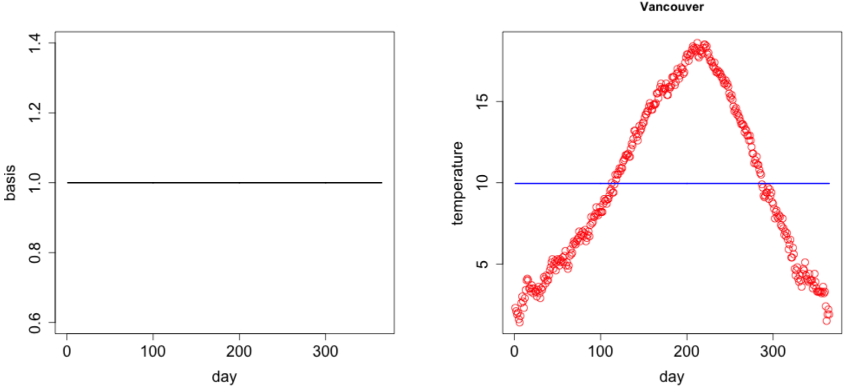
$$1, sin(\omega t), cos(\omega t), sin(2\omega t), cos(2\omega t), \ldots$$

$$sin(m\omega t), cos(m\omega t), \ldots$$

- constant ω defines the period of oscillation of the first sine/cosine pair. This is $\omega = 2\pi/P$ where P is the period.
- $K = 2M + 1$ where M is the largest number of oscillations required in a period of length P.

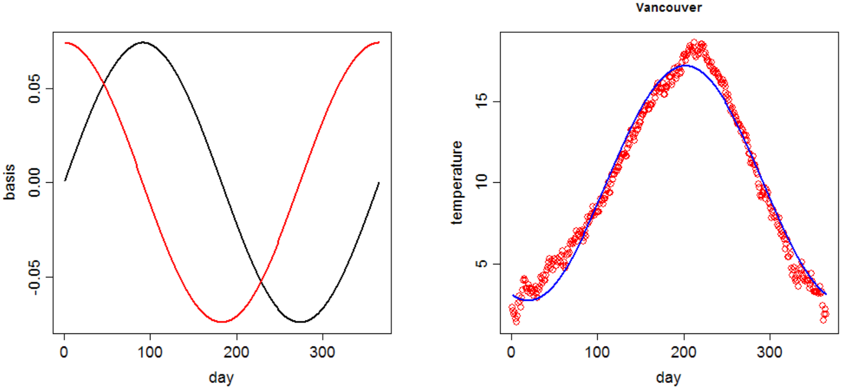
The Fourier Basis

$$\Phi(t) = (1)$$



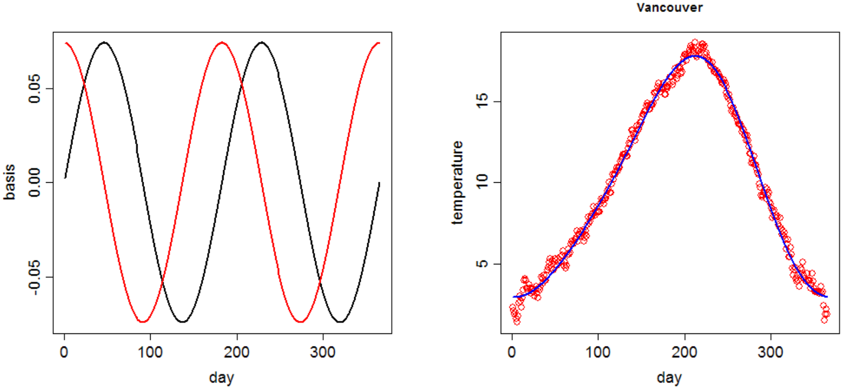
The Fourier Basis

$$\Phi(t) = (1, sin(\omega t), cos(\omega t))$$



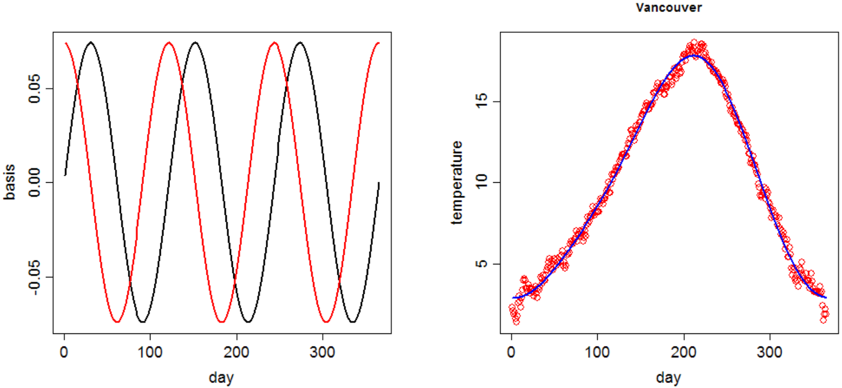
The Fourier Basis

$$\Phi(t) = (1, sin(\omega t), cos(\omega t), sin(2\omega t), cos(2\omega t))$$



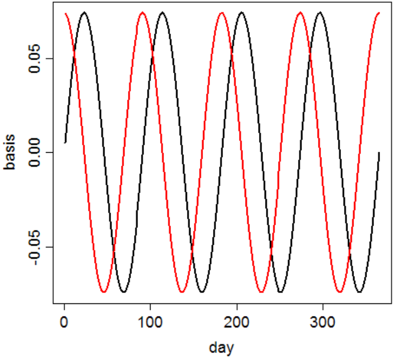
The Fourier Basis

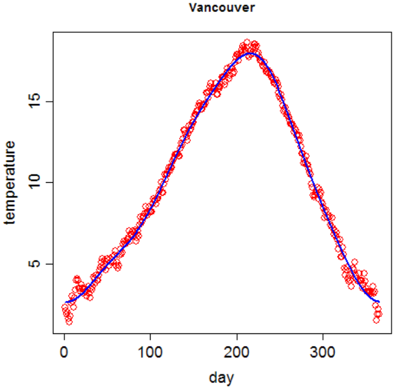
$$\Phi(t) = (1, sin(\omega t), cos(\omega t), sin(2\omega t), cos(2\omega t), sin(3\omega t), cos(3\omega t))$$



The Fourier Basis

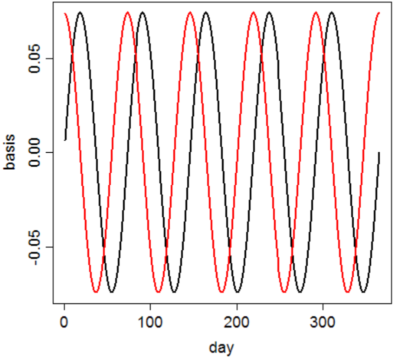
$$\Phi(t) = (1, sin(\omega t), cos(\omega t), \ldots, sin(4\omega t), cos(4\omega t))$$

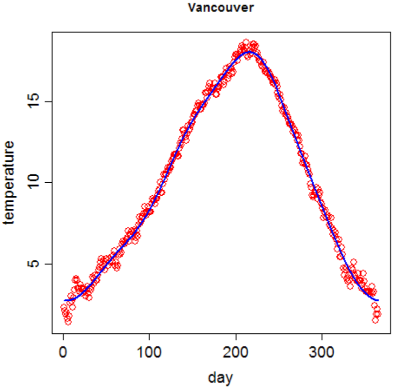




The Fourier Basis

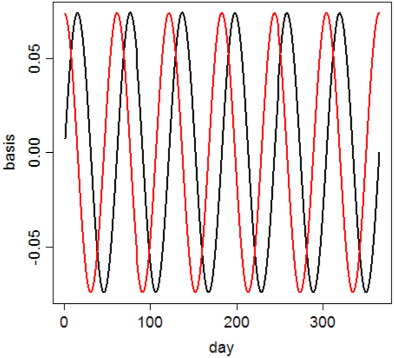
$$\Phi(t) = (1, sin(\omega t), cos(\omega t), \ldots, sin(5\omega t), cos(5\omega t))$$

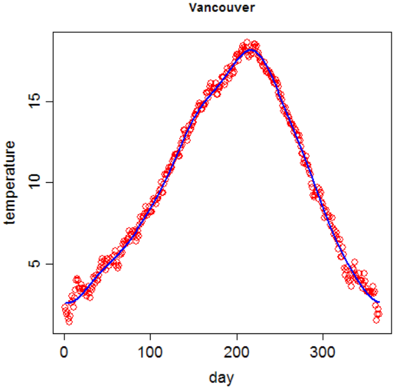




The Fourier Basis

$$\Phi(t) = (1, sin(\omega t), cos(\omega t), \ldots, sin(6\omega t), cos(6\omega t))$$





Advantages of Fourier Bases

- Only alternative to monomial bases until the middle of the 20th century
- Excellent computational properites, especially if the observations are equally spaced.
- Natural for describing periodic data, such as the annual weather cycle

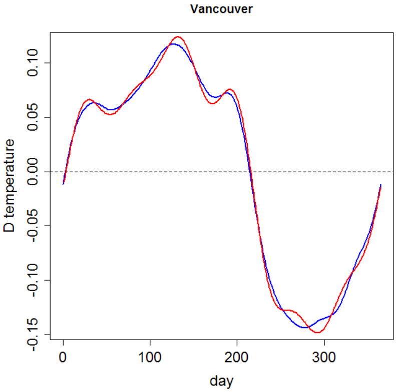
BUT many functions are not periodic; this can be a problem if the data are, for example, growth curves.
Fourier basis is still the first choice in many fields, such as signal analysis, even when the data are not periodic.

Fourier Derivatives

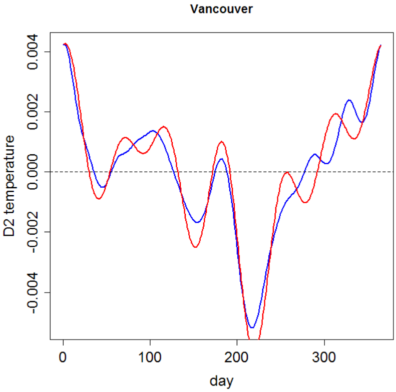
$$Dsin(\omega t) = \omega cos(\omega t), \;\; Dcos(\omega t) = -\omega sin(\omega t)$$

So derivatives retain complexity, easy to compute

D



D^2

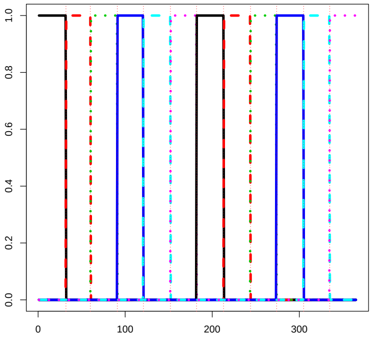


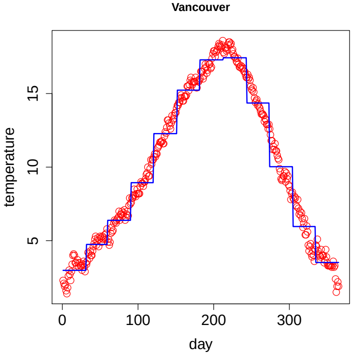
Splines

- Splines are polynomial segments joined end-to-end
- Segments are constrained to be smooth at the join
- The points at which the segments join are called *knots*
- The order m (order $=$ degree$+1$) of the polynomial segments and
- the location of the knots define the system.
- **Bsplines** are a particularly useful means of incorporating the constraints.

Splines

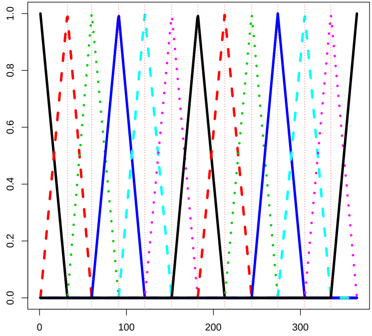
Vancouver temperature with knots at months.
Splines of order 1

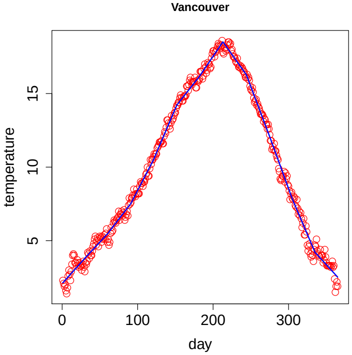




Splines

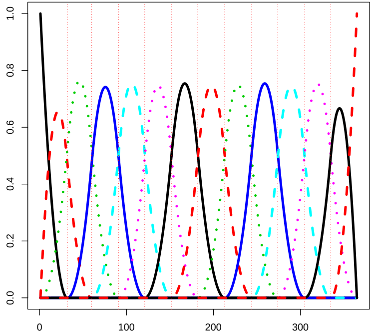
Vancouver temperature with knots at months.
Splines of order 2

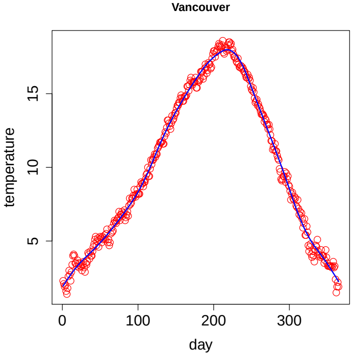




Splines

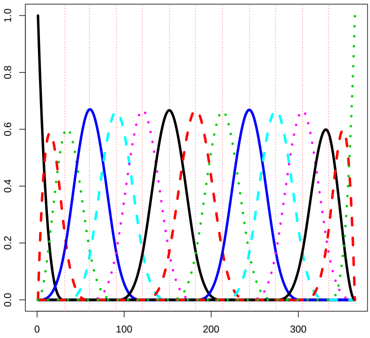
Vancouver temperature with knots at months.
Splines of order 3

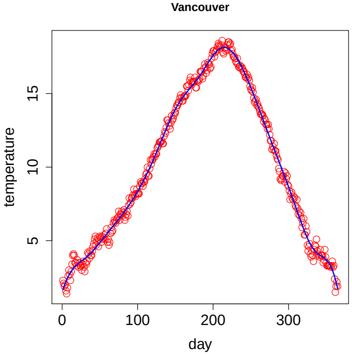




Splines

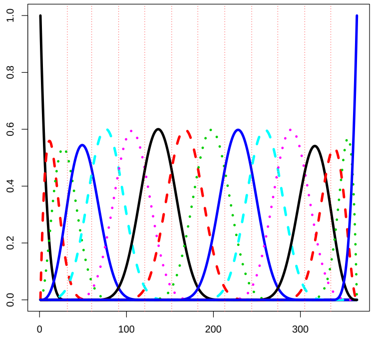
Vancouver temperature with knots at months.
Splines of order 4

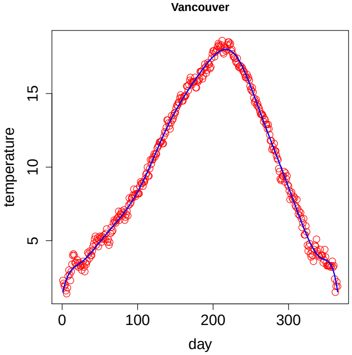




Splines

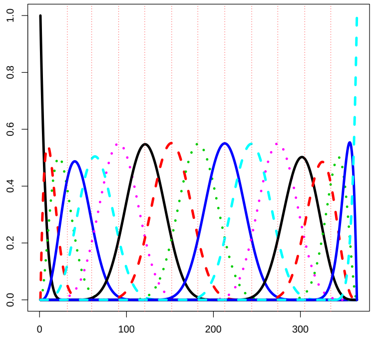
Vancouver temperature with knots at months.
Splines of order 5

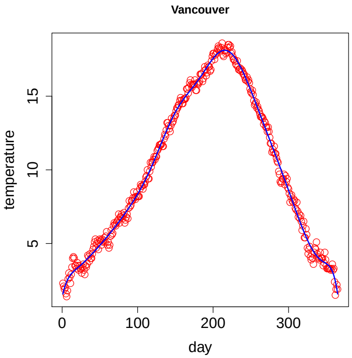




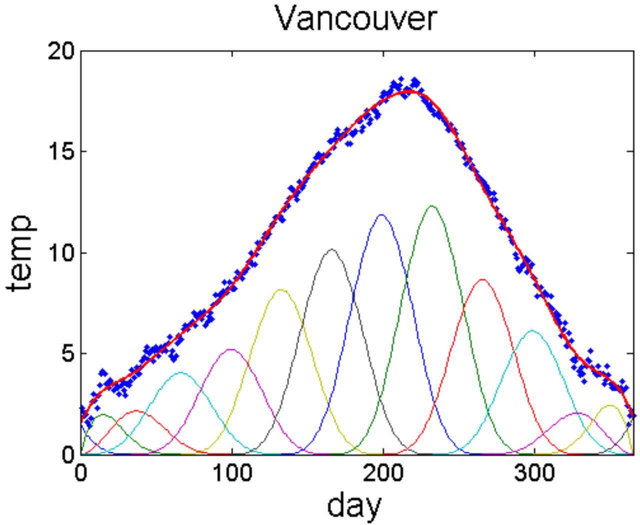
Splines

Vancouver temperature with knots at months.
Splines of order 6





An illustration of basis expansions for *local* basis functions



Properties of B-splines

- Number of basis functions:

$$order + number\ interior\ knots$$

- Derivatives up to $m-2$ are continuous.
- B-spline basis functions are positive over at most m adjacent intervals $\rightarrow$ fast computation for even thousands of basis functions.
- Sum of all B-splines in a basis is always 1; can fit any polynomial of order m.
- Most popular choice is order 4, implying continuous second derivatives. Second derivatives have straight-line segments.

Bsplines: Choosing Knots and Order

- The order of the spline should be at least $k + 2$ if you are interested in k derivatives.
- The order of the spline usually is $k + 4$ if you are interested in k derivatives.
- When determining the number of basis functions, we generally fix the order of the spline and change the number of knots.
- Knots are often equally spaced (a useful default)
- But there are two important rules:
  - Place more knots where you know there is strong curvature, and fewer where the function changes slowly.
  - Be sure there is at least one data point in every interval.
- Later, we'll discuss placing a knot at each point of observation.
- Co-incident knots reduce the number of continuous derivatives at each point. This can be useful (more later).

Other Bases

The `fda` library in `R` also allows the following bases:

- **Constant** $\phi(t) = 1$, the simplest of all.
- **Power** $t^{\lambda_1}, t^{\lambda_2}, t^{\lambda_3}, \ldots$, powers are distinct but not necessarily integers or positive.
- **Exponential** $e^{\lambda_1 t}, e^{\lambda_2 t}, e^{\lambda_3 t}, \ldots$

Other possible bases include

- **Wavelets** especially for sharp, local features
- **Empirical** we will investigate functional Principal Components
- **Designer** see our section on dynamic models: tailoring a basis to data (if you know something about the data) can be much more efficient.

Summary

1. Basis expansions: just like adding different independent variables in linear regression
2. Monomial basis: direct extension of adding interaction and quadratic terms. Poor numerics, bad for derivatives.
3. Fourier basis: classical, common in signal processing etc. Great for periodic functions. Must be infinitely differentiable.
4. B-spline basis: locally polynomial. Allows control of smoothness and accuracy. Local definition $\Rightarrow$ good numerics.
5. Other basis systems also exist.
6. What is best depends on the data.