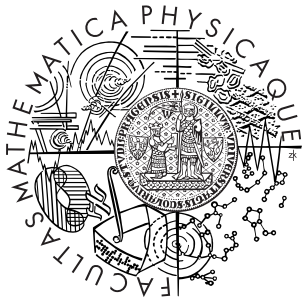
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Barbora Lebdušková

Vývoj dynamického modelu pro odhad radonové zátěže budov

Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce: RNDr. Ing. Marek Brabec, PhD., Státní zdravotní ústav
Studijní program: Matematika, matematická statistika

Ráda bych zde poděkovala panu RNDr. Ing. Brabcovi, PhD. za trpělivý přístup a poskytnutí dat a dále svojí rodině za podporu, které se mi od ní dostalo.

Prohlašuji, že jsem svou diplomovou práci napsala samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 10.12.2009 Barbora Lebdušková

Obsah

Název práce: Vývoj dynamického modelu pro odhad radonové zátěže budov
Autor: Barbora Lebdušková
Katedra (ústav): Katedra pravděpodobnosti a matematické statistiky
Vedoucí diplomové práce: RNDr. Ing. Marek Brabec, PhD.
e-mail vedoucího: mbrabec@cs.cas.szu

Abstrakt: V předložené práci je popsána metoda odhadu funkcionálních dat na základě diskrétních pozorování. Jedná se o aproximaci pomocí báze, přičemž se zde pracuje s periodickou Fourierovou bází a neperiodickým B-splajnem. Další část práce se věnuje concurrent modelu pro funkcionální data. Je zde odvozen tvar intervalů spolehllivosti regresní funkce a na simulovaných datech analyzována citlivost modelu. Jedna kapitola je věnována funkcím programovacího jazyka R, které umožňují výpočet výše popsaných postupů. V poslední části je tento model aplikován na reálná data obsahující měření koncentrace radonu v testovacím objektu.

Klíčová slova: Fourierova báze, B-splajn, concurrent model

Title: Dynamic model for estimation of radon concentration in buildings
Author: Barbora Lebdušková
Department: Department of probability and mathematical statistics
Supervisor: RNDr. Ing. Marek Brabec, PhD.
Supervisor's e-mail address: mbrabec@cs.cas.szu

Abstract: In the present work the method for estimation of functional data from discrete values is described. The basis aproximation is used and types of functions for basis construction are Fourier functions and b-spline. Next part of the work attends to the concurrent model for functional data. Here is also described construction of confidence interval for the regression function. One chapter is focused on application of these techniques in language R. In the last part of the work the the series of radon concentration measurements in experimental building is analysed.

Keywords: Fourier basis, B-spline, concurrent model

Kapitola 1

Úvod

Cílem této práce je vytvořit model, který popisuje vztah mezi náhodnými veličinami v průběhu času. Klasický regresní model popisuje tuto závislost diskrétním způsobem a nebere v úvahu, že se může v čase měnit. Model popsaný v této práci takovou dynamiku umožňuje, a to takovým způsobem, že modeluje proměnnou v čase t pomocí vysvětlujících proměnných v témže čase. Běžně se tento model označuje anglickým termínem *concurrent*, český ekvivalent je *souběžný*. Dále se budeme držet anglického označení. Odvodila jsem pro tento model odhad regresní funkce a jejích intervalů spolehlivosti a na simulovaných datech zkoumala citlivost tohoto modelu.

Aby bylo možné takový model zkonstruovat, je zapotřebí mít tzv. funkcionální data. To znamená, že jednotlivá pozorování jsou spojité funkce. Tomu, jak takový formát dat získat z diskrétních měření, jsem věnovala úvodní část práce. Jedná se o aproximaci naměřených hodnot pomocí lineární kombinace několika vybraných funkcí. Tyto skupiny funkcí se nazývají báze a existuje jich celá řada. Vybrala jsem dva nejrozšířejnější typy. Fourierova báze reprezentuje skupinu periodických bází a B-splajn neperiodické báze. Na simulovaných datech hodnotím jejich vlastnosti. Pro Fourierovu bázi jsem odvodila tvar penalizační matice, která se využívá při hledání koeficientů lineární kombinace.

V závěru jsem model aplikovala na reálná data, která popisují koncentraci radonu v testovacím objektu.

Všechny výpočty jsem prováděla v softwaru R 2.9.0, který je volně dostupný. Přepisy jednotlivých programů jsou na přiloženém CD. Základní možnosti knihovny `fda` jsem popsala v kapitole (4).

Kapitola 2

Funkcionální data

V reálném životě se často setkáváme s jevy, jejichž průběh se dá popsat spojitou funkcí v určitém časovém intervalu. Může se jednat například o měření teploty v průběhu roku apod. Data, kde jednotlivá pozorování nejsou skaláry, ale spojité reálné funkce, budeme označovat jako *funkcionální data*. Budeme uvažovat sadu funkcionálních dat

$$y_j(t), j = 1, \ldots, q, \quad t \in T, \tag{2.1}$$

kde T je časový interval. Ve skutečnosti však nepozorujeme hodnoty y_j souvisle v celém intervalu T, ale pouze v bodech $t_1, t_2, \ldots, t_n$, které náleží do intervalu $T \subset \mathbb{R}$. Je tedy nutné odhadnout hodnoty y_j pro celý interval T. V následujících odstavcích zmíníme základní metody získání odhadů $\widehat{y_j}$. Pro jednoduchost budeme uvažovat $j = 1$ a $y_1(t)$ označíme jako $y(t)$.

2.1 Odhad funkcionálních dat na základě diskrétních pozorování

Uvažujme funkcionální bázi $\{\phi_l(t)\}_{l=1}^{L}$, kde $t \in T \subset \mathbb{R}$, takovou, že

$$y(t_i) = \sum_{l=1}^{L} \phi_l(t_i)\, c_l + \epsilon(t_i), \tag{2.2}$$

kde $\boldsymbol{c} = (c_1, \ldots, c_L)$ je vektor koeficientů, $\epsilon(t)$ je náhodná chyba a body $(t_1, \ldots, t_n) \in T$ jsou časové okamžiky, ve kterých byly pozorovány diskrétní hodnoty náhodné veličiny y. Znamená to, že hodnoty $\epsilon(t_i)$ musí být nezávislé a stejně rozdělené. Báze můžeme podle charakteru funkcí ϕ_l rozdělit na periodické a neperiodické. Počet funkcí báze

budeme dále označovat jako dimenzi báze. Dimenze báze ovlivňuje, jak hodně se budou lišit hodnoty odhadu $\widehat{y}(t) = \sum_{l=1}^{L} \phi_l(t) c_l$ od naměřených hodnot $y(t_i)$.

Pokud bychom kvalitu odhadu posuzovali pouze podle míry odlišnosti od originálních dat, odhadli bychom hodnoty $\boldsymbol{c}$ pomocí minimalizace součtu čtverců (SSE),

$$SSE(\boldsymbol{c}) = \sum_{i=1}^{n} \left[y(t_i) - \sum_{l=1}^{L} \phi_l(t_i) c_l \right]^2. \tag{2.3}$$

Dále budeme používat označení $\boldsymbol{y}(t) = (y(t_1), \ldots, y(t_n))'$ a $\boldsymbol{\Phi}$ nechť je matice, jejíž sloupce tvoří funkce $\phi_l, l = 1, \ldots, L$, v bodech $t_1, \ldots, t_n$. Výraz (2.3) je možné zapsat jako

$$\begin{aligned} SSE(\boldsymbol{c}) &= (\boldsymbol{y} - \boldsymbol{\Phi c})'(\boldsymbol{y} - \boldsymbol{\Phi c}) = \\ &= \boldsymbol{y'y} - \boldsymbol{y'\Phi c} - \boldsymbol{c'\Phi y} + \boldsymbol{c'\Phi'\Phi c}. \end{aligned} \tag{2.4}$$

Tento výraz je minimální pro $\widehat{\boldsymbol{c}} = (\boldsymbol{\Phi'\Phi})^{-1} \boldsymbol{\Phi' y}$.

Tímto přístupem se reguluje pouze chyba odhadu, která se zmenšuje s rostoucí dimenzí báze. V případě použití dostatečně vysokého počtu bazických funkcí, dosáhneme vždy odhadu jehož SSE bude nulová. Další nevýhodou takto konstruovaných odhadů je nestabilita derivací. Cílem je však najít odhad, který na jedné straně dostatečně dobře kopíruje původní data, ale zároveň nemá příliš vysoký rozptyl. Toho docílíme buď použitím báze s menší dimenzí, nebo když místo střední čtvercové chyby (3.14) použijeme jako minimalizační kritérium penalizovanou čtvercovou chybu

$$\begin{aligned} PSSE_\lambda(\boldsymbol{c}) &= \sum_{i=1}^{n} \left[y(t_i) - \sum_{l=1}^{L} \phi_l(t_i) c_l \right]^2 + \lambda PEN(\widehat{\boldsymbol{y}}) = \\ &= (\boldsymbol{y} - \boldsymbol{\Phi c})'(\boldsymbol{y} - \boldsymbol{\Phi c}) + \lambda PEN(\widehat{\boldsymbol{y}}). \end{aligned} \tag{2.5}$$

V tomto případě je dobré použít co největší bázi. Jako vhodné penalizační kritérium je možné vzít například

$$\begin{aligned} PEN_m(\widehat{\boldsymbol{y}}) &= \int [D^m \widehat{y}(t)]^2 dt = \\ &= \int_T (D^m \boldsymbol{c'\phi}(t))^2 dt = \\ &= \int_T (D^m \boldsymbol{c'\phi}(t)) (D^m \boldsymbol{\phi}'(t) \boldsymbol{c}') dt = \\ &= \boldsymbol{c'} \left[\int_T (D^m \boldsymbol{\phi}(t)) (D^m \boldsymbol{\phi}'(t)) dt \right] \boldsymbol{c} = \\ &\overset{ozn.}{=} \boldsymbol{c'Rc}, \end{aligned} \tag{2.6}$$

kde D^m značí m-tou derivaci a $\boldsymbol{\phi}$ je vektor funkcí $\phi_l(t)$, pro $l = 1, \ldots, L$. (Pokud chceme hladkou derivaci odhadu $\widehat{y}$ řádu r, musí platit $m = r + 2$.) Nejběžnější volbou je $m = 2$, protože v případě, kdy má křivka vysoký rozptyl, je hodnota $\left[D^2\widehat{y}(t)\right]^2$ vysoká. Potom se $\widehat{\boldsymbol{c}}$, které minimalizuje penalizovaný součet čtverců ($PSSE$ viz (2.5)), dá vyjádřit jako

$$\widehat{\boldsymbol{c}} = \left(\boldsymbol{\Phi}'\boldsymbol{\Phi} + \lambda \boldsymbol{R}\right)^{-1} \boldsymbol{\Phi}'\boldsymbol{y} \tag{2.7}$$

a odhad $\widehat{\boldsymbol{y}}$ lze zapsat jako

$$\widehat{\boldsymbol{y}} = \boldsymbol{\Phi}\left(\boldsymbol{\Phi}'\boldsymbol{\Phi} + \lambda \boldsymbol{R}\right)^{-1} \boldsymbol{\Phi}'\boldsymbol{y} = \boldsymbol{S}_{\lambda,\Phi}\boldsymbol{y}. \tag{2.8}$$

Poměr mezi přesností a rozptylem odhadu je určen velikostí parametru λ. Pokud je parametr $\lambda = 0$, nedochází k penalizaci a výsledný odhad je příliš variabilní. V opačném případě (λ je vysoká) se odhad blíží lineární funkci. Je tedy velmi důležité nastavit tento parametr co nejlépe. V případě, že předpokládáme nezávislost $y(t_i)$, můžeme použít metodu *cross-validace* nebo *zobecněné cross-validace*. První metoda je zde popsána na základě informací z knihy [4] a popis druhé je čerpán z knihy [11].

Cross-validace (CV):

- Na základě datových bodů $y(t_1), \ldots, y(t_n)$ určíme pro $j = 1, \ldots, n$ odhady $\widehat{\boldsymbol{y}}_{-j}$, které jsou založené vždy na všech datech kromě $y(t_j)$. To znamená, že pro výpočet takového odhadu použijeme datové body $y(t_1), \ldots, y(t_{j-1}), y(t_{j+1}), \ldots, y(t_n)$.
- Spočítáme

$$CV(\lambda) = \frac{1}{n}\sum_{j=1}^{n}\left[y(t_j) - \widehat{y}(t_j)_{-j}\right]^2. \tag{2.9}$$

- Optimální λ minimalizuje výraz (2.9).

Zobecněná cross-validace (GCV):

- Spočítáme

$$GCV(\lambda) = \left[\frac{n}{n - tr\left(\boldsymbol{S}_{\lambda\Phi}\right)}\right]\left[\frac{SSE}{n - tr\left(\boldsymbol{S}_{\lambda\Phi}\right)}\right], \tag{2.10}$$

  kde $tr(\cdot)$ značí stopu matice.
- Optimální λ minimalizuje výraz (2.10).

Metoda GCV byla poprvé popsána v článku [2] a původně sloužila jako aproximace metody CV. Postupem času se ukázalo, že v praxi má GCV často lepší vlastnosti než CV.

Výše uvedená automatická optimalizace parametru λ nemusí vždy dávat dobré výsledky. Velmi často může nastat situace, že data nejsou nezávislá. Potom obě metody (GCV i CV) dávají odhad, který příliš kopíruje originální data a pro další využití je nevhodný. V takovém případě je možné použít metody, které byly vyvinuty právě pro závislé časové řady. Touto problematikou se zabývá článek [3]. Jiným možným řešením je zvolit parametr λ bez použití automatických metod pouze na základě vlastního úsudku.

Pro hodnocení kvality modelu se používá normalizovaný součet čtverců ($RMSE$), který je definovaný takto

$$RMSE = \sqrt{\frac{1}{n}\sum_{i-1}^{n}\left(\widehat{y}\left(t_i\right) - y\left(t_i\right)\right)^2} = \sqrt{\frac{1}{n}\left(\boldsymbol{y} - \boldsymbol{\Phi c}\right)'\left(\boldsymbol{y} - \boldsymbol{\Phi c}\right)}. \tag{2.11}$$

2.2 Typ báze

Jak jsme již uvedli dříve, báze můžeme rozdělit na periodické a neperiodické. V této kapitole se seznámíme se zástupci obou těchto skupin. Periodické báze bude reprezentovat Fourierova báze a z neperiodických bází uvedeme B-splajn. Možností, jak volit bázi, je však daleko více. Kromě již zmíněných dvou je možné použít například jádrové funkce. Dále existují speciální báze pro vyhlazování monotonních funkcí apod. Podrobnější popis této problematiky je možné najít v [11].

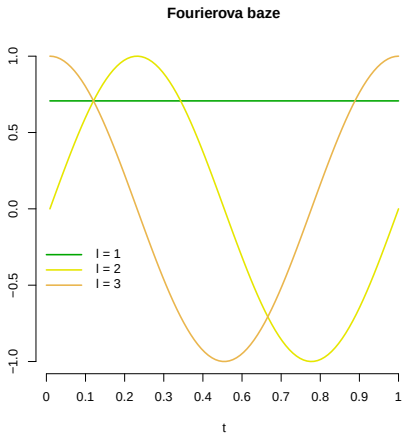
Volba báze je velmi důležitá a vždy musíme zvažovat dvě důležitá kritéria. Prvním je výpočetní náročnost a druhým schopnost báze aproximovat původní hodnoty. S rychlým vývojem výpočetní techniky se první kritérium sice dostává mírně do pozadí, ale stále není zanedbatelné. Oba zde uvedené typy báze dobře splňují tyto požadavky.

2.2.1 Fourierova báze

Tato báze je odvozena od Fourierových řad. V knize [11] je definována následovně:

Definice 2.2.1 *Fourierova báze je definována jako* $\{\phi_l(t)\}_{l=1}^{L}$. *Jednotlivé* $\phi_l(t)$ *mají tvar*

$$\begin{aligned} \phi_1(t) &= 1, \\ \phi_{2r}(t) &= \sin r\omega t, \\ \phi_{2r+1}(t) &= \cos r\omega t, \end{aligned}$$



Obrázek 2.1: První tři funkce Fourierovy báze.

kde r je celé číslo a ω parametr, který určuje délku periody $2\pi/\omega$.

Zobrazení ϕ_l pro $l = 1, 2, 3$ je na obrázku (2.1). Tato báze je periodická a je vhodná hlavně pro data, která jsou stabilní v čase, popřípadě mají periodický charakter.

Budeme předpokládat, že počet funkcí L v bázi je lichý (tzn. obsahuje ke každé složce $\sin(\cdot)$ příslušnou složku $\cos(\cdot)$), tj. $L = 2r + 1$, r je kladné celé číslo, $\boldsymbol{P}$ je $(2r+1)$-diagonální matice s diagonálou $(1, 1, 1, 2, 2, \ldots, r, r)$ a $t \in T = (t_a, t_b)$. Potom pro všechna kladná celá čísla s platí:

$$\begin{aligned}
\boldsymbol{\phi}(t) &= (1, \sin\omega t, \cos\omega t, \ldots, \sin r\omega t, \cos r\omega t)', \\
D^{4s-3}\boldsymbol{\phi}(t) &= \omega^{4s-3}\boldsymbol{P}^{4s-3}(0, \cos\omega t, -\sin\omega t, \ldots, \cos r\omega t, -\sin r\omega t)', \\
D^{4s-2}\boldsymbol{\phi}(t) &= -\omega^{4s-2}\boldsymbol{P}^{4s-2}(0, \sin\omega t, \cos\omega t, \ldots, \sin r\omega t, \cos r\omega t)', \\
D^{4s-1}\boldsymbol{\phi}(t) &= -\omega^{2}\boldsymbol{P}^{2}D^{4s-3}\boldsymbol{\phi}(t), \\
D^{4s}\boldsymbol{\phi}(t) &= -\omega^{2}\boldsymbol{P}^{2}D^{4s-2}\boldsymbol{\phi}(t),
\end{aligned}$$

$$\boldsymbol{\Phi}(t) = \begin{pmatrix}
1 & \sin\omega t_1 & \cos\omega t_1 & \ldots & \sin\omega r t_1 & \cos\omega r t_1 \\
1 & \sin\omega t_2 & \cos\omega t_2 & \ldots & \sin\omega r t_2 & \cos\omega r t_2 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
1 & \sin\omega t_n & \cos\omega t_n & \ldots & \sin\omega r t_n & \cos\omega r t_n
\end{pmatrix}, \tag{2.12}$$

$$\boldsymbol{R}_{4s-3} = \omega^{8s-6}\boldsymbol{P}^{8s-6}\begin{pmatrix} 0 & 0 & \dots & 0 \\ 0 & \left[\frac{1}{4\omega}\sin 2\omega t + \frac{t}{2}\right]_{t_a}^{t_b} & \dots & \left[-\frac{\cos(r+1)\omega t}{2\omega(r+1)} - \frac{\cos(r-1)\omega t}{2\omega(r-1)}\right]_{t_a}^{t_b} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \left[-\frac{\cos(r+1)\omega t}{2\omega(r+1)} - \frac{\cos(r-1)\omega t}{2\omega(r-1)}\right]_{t_a}^{t_b} & \dots & \left[-\frac{1}{4\omega r}\sin 2\omega r t + \frac{t}{2}\right]_{t_a}^{t_b} \end{pmatrix},$$

$$\boldsymbol{R}_{4s-2} = \omega^{8s-4}\boldsymbol{P}^{8s-4}\begin{pmatrix} 0 & 0 & \dots & 0 \\ 0 & \left[-\frac{1}{4\omega}\sin 2\omega t + \frac{t}{2}\right]_{t_a}^{t_b} & \dots & \left[-\frac{\cos(r+1)\omega t}{2\omega(r+1)} - \frac{\cos(1-r)\omega t}{2\omega(1-r)}\right]_{t_a}^{t_b} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \left[-\frac{\cos(r+1)\omega t}{2\omega(r+1)} - \frac{\cos(1-r)\omega t}{2\omega(1-r)}\right]_{t_a}^{t_b} & \dots & \left[\frac{1}{4\omega r}\sin 2\omega r t + \frac{t}{2}\right]_{t_a}^{t_b} \end{pmatrix},$$

$$\begin{aligned} \boldsymbol{R}_{4s-1} &= -\omega^4\boldsymbol{P}^4\boldsymbol{R}_{4s-3}, \\ \boldsymbol{R}_{4s} &= -\omega^4\boldsymbol{P}^4\boldsymbol{R}_{4s-2}, \end{aligned}$$

kde $[f(x)]_a^b = f(b) - f(a)$.

2.2.2 Příklad 1

Tento příklad ilustruje nevhodnost Fourierovy báze pro neperiodická data. Vygenerujeme řadu periodických a neperiodických dat a porovnáme, jak fungují Fourierovy odhady.

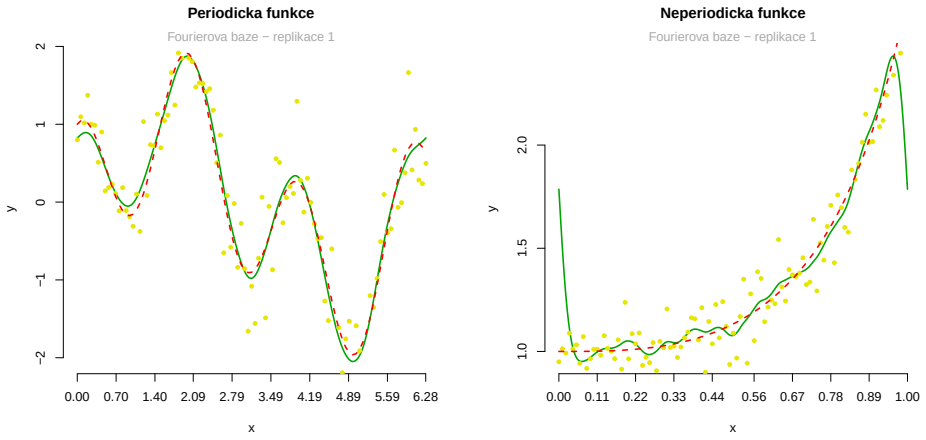
Periodická funkce má tvar

$$f_1(t) = \sin(t) + \cos(\pi t), \quad t \in (0, 2\pi).$$

Jako pozorovaná data byly použity hodnoty $f_1(t_i) + \epsilon_1(t_i)$ v bodech $t_i = \frac{2\pi i}{100}, \quad i = 0, \dots, 100$, přičemž $\epsilon_1 \sim N(0,\ 0, 4^2)$. Pro vyhlazení byla použita Fourierova báze skládající se ze 101 funkcí a pro penalizaci PEN_2 (viz 2.6). Parametr λ byl spočten pomocí metody zobecněné cross-validace a jeho hodnota je $0,0089$.

Neperiodická data odpovídají modelu

$$f_2(s) = e^{s^3} + \epsilon_2(s), \quad s \in (0, 1).$$



Obrázek 2.2: Vhodnost Fourierovy báze pro periodická a neperiodická data - žlutě jsou vyznačena simulovaná data, červeně hodnoty funkce $f_1(t)$ (popř. $f_2(t)$) a zeleně výsledný odhad.

Hodnoty byly počítány v bodech $s_i = \frac{i}{100}, \quad i = 0, \ldots, 100$ a $\epsilon_2 \sim N\left(0, (0,1)^2\right)$. Pro získání parametru λ používáme stejnou metodu i bázi jako v prvním případě. Pro tato data vyšel parametr $\lambda = 7,08 \cdot 10^{-6}$.

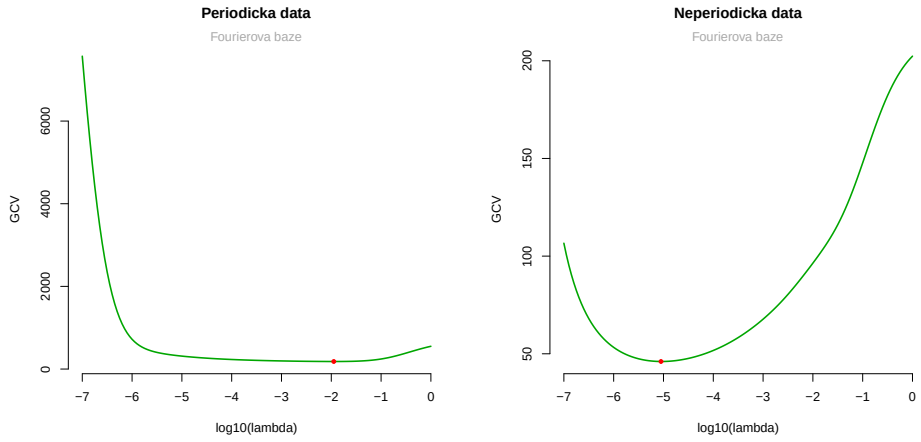
Výsledky jsou znázorněny na obrázku (2.2). V prvním případě proběhlo vyhlazení velmi dobře, ale ve druhém nastal problém s okraji definičního oboru a ani metoda zobecněné cross-validace nefunguje, jak bychom očekávali. Parametr λ je příliš malý a dochází k nedostatečnému vyhlazení. Příklad ukazuje nevhodnost Fourierovy báze pro neperiodická data.

Na obrázku 2.3 je průběh $GCV(\lambda)$. Je vidět, že pro periodická data není minimum funkce $GCV(\lambda)$ ostře vymezené. Znamená to, že informace o parametru λ není dostatečně určující, tj. výsledný odhad není příliš citlivý na změny tohoto parametru.

2.2.3 B-splajn

Splajny nabízejí větší variabilitu než Fourierova báze, ale jsou také složitější. Existuje velké množství různých typů splajnů. V této kapitole se podrobněji seznámíme s B-splajnem.

Opět nás bude zajímat odhad $\widehat{\boldsymbol{y}}(t)$, $t \in T$, na základě bodů $y(t_i)$. Budeme předpokládat, že T je interval s krajními body s_1 a s_m, a body $s_2, \ldots, s_{m-1}$ jsou vnitřní body (uzly) tohoto intervalu. Obecně je polynomiální splajn definován tak, že na subintervalech mezi libovolnými dvěma sousedními uzly je definován polynom stupně K a v



Obrázek 2.3: Závislost GCV na velikosti parametru λ.

každém uzlu mají sousední polynomy stejnou hodnotu (výsledná funkce je spojitá), a to i pro $K-1$ derivací. Pokud však chceme hladkou (nejen spojitou) p-tou derivaci, musí být stupeň splajnu alespoň $p+2$. Nejběžněji používaný je kubický B-splajn, tj. $K=3$, ale pro práci s derivacemi je zapotřebí zvolit vyšší stupeň splajnu. Podrobný popis splajnů je možné nalézt v knize [1], ze které pochází také následující definice.

Definice 2.2.2 *Nechť je* $\boldsymbol{s} = (s_1, \ldots, s_m)$ *neklesající posloupnost bodů z intervalu* T. *Pak jednotlivé B-splajny řádu* $k = 1, \ldots, K$ *definujeme pomocí rekurzivního vzorce*

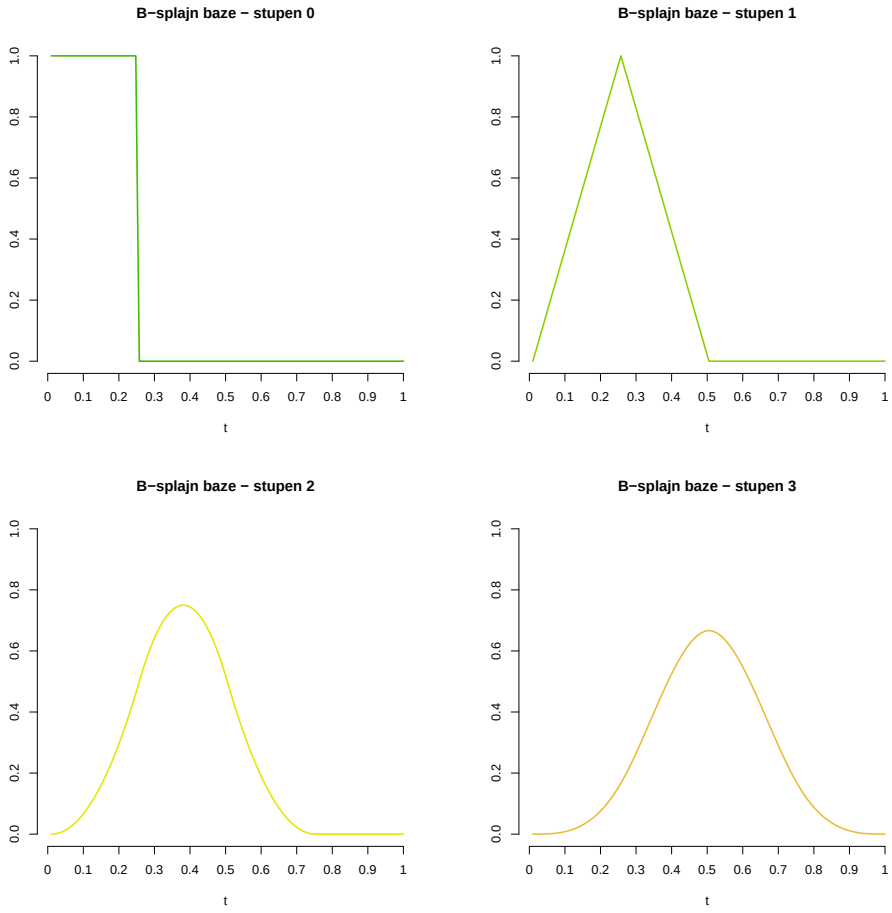
$$\begin{aligned} B_{j,1}(s) &= \begin{cases} 1 & \text{pro } s_j \leq s < s_{j+1}, \\ 0 & \text{jinak,} \end{cases} \\ B_{j,k}(s) &= \frac{s - s_j}{s_{j+k-1} - s_j} B_{j,k-1}(s) + \frac{s_{j+k} - s}{s_{j+k} - s_{j+1}} B_{j+1,k-1}(t)\,. \end{aligned}$$

V případě, že se ve sčítanci vyskytuje výraz $\frac{0}{0}$, položíme ho rovný 0. Na obrázku (2.4) jsou vykresleny B-splajny řádu 1, 2, 3 a 4. *Počet funkcí báze = řádu splajnu + počet vnitřních uzlů* +1. V následující větě jsou shrnuty základní vlastnosti B-splajnu.

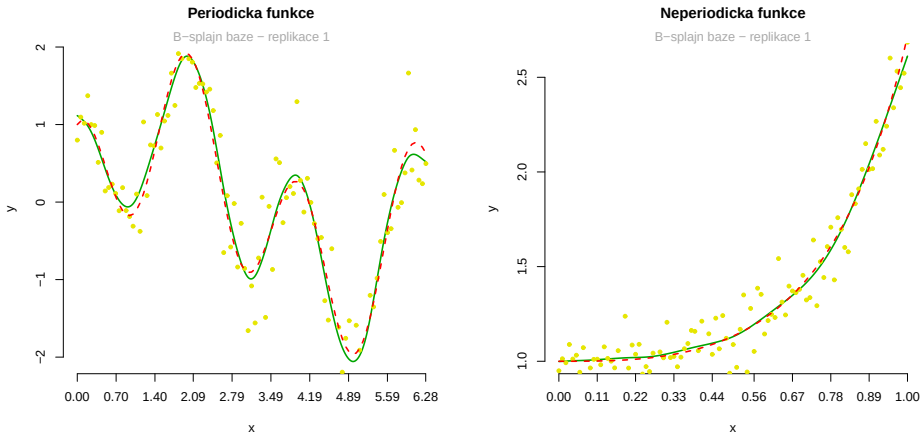
Věta 2.2.1 *Nechť* $B_{j,k}(s)$ *je B-splajn z definice 2.2.2 definovaný na intervalu* T, *potom pro všechna* $k = 1, \ldots, K$ *platí následující vlastnosti.*

- *B-splajn báze je složena z nezáporných funkcí, tj.*

$$B_{j,k}(s) = 0, \quad s \notin [s_j, s_{j+k}] \quad \text{a zároveň} \quad B_{j,k}(s) > 0, \quad s \in (s_j, s_{j+k})\,. \tag{2.13}$$



Obrázek 2.4: První čtyři funkce B-splajn báze.



Obrázek 2.5: Vhodnost B-splajn báze pro periodická a neperiodická data - žlutě jsou vyznačena generovaná data, červeně odhadovaná funkce a zeleně výsledný odhad.

- *Součet všech funkcí báze je v každém bodě intervalu T roven* 1*, tj.*

$$\sum_j B_{j,k}(s) = 1, \quad s \in T. \tag{2.14}$$

Důkaz: Viz [4]. △

Z definice B-splajnu je vidět, že uzly $(s_1, \ldots, s_m)$ nemusí být totožné s $(t_1, \ldots, t_n)$. Následující simulační studie ukazuje vliv volby uzlů na kvalitu výsledného odhadu $\widehat{y}(t)$.

2.2.4 Pokračování příkladu 1

Tento příklad navazuje na Příklad 1 (2.2.2). Výchozí data jsou shodná. Opět použijeme penalizaci a pro hledání vhodného parametru λ zobecněnou cross-validaci. Rozdíl je pouze v bázi. Tentokrát použijeme kubický B-splajn, jehož uzly se shodují s definičním oborem generovaných dat. Výsledky jsou na obrázku (2.5).

Je vidět, že B-splajn je vhodný pro oba typy dat. Abychom mohli porovnat metodu z příkladu 1 (2.2.2) a zde popsanou metodu, simulovali jsme 1000 periodických i neperiodických datových řad, provedli vyhlazení podle výše uvedených postupů a spočítali odhad střední čtvercové chyby.

Definice 2.2.3 *Střední čtvercová chyba* (MSE) *odhadu* $\widehat{y}(t)$ *funkce* $f(t)$ *definované na* $T \subset \mathbb{R}$ *je definovaná jako*

$$MSE(\widehat{y}) = E(\widehat{y}(t) - f(t))^2, \tag{2.15}$$

	Periodická data	Neperiodická data
B-splajn báze	$2,1224$	0.0007
Fourierova báze	$2,0296$	$0,0234$

Tabulka 2.1: Hodnoty odhadu MSE

pokud tato střední hodnota existuje.

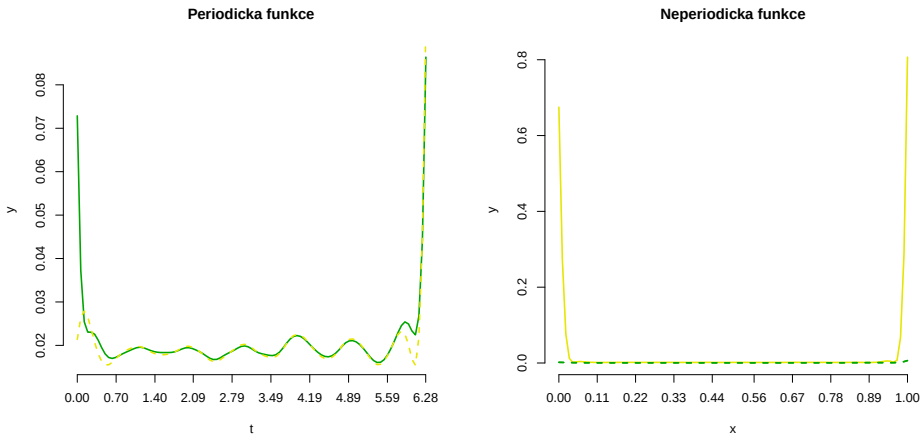
Takto definovaná střední čtvercová chyba funkcionálního odhadu je také funkce, a proto je možné kvalitu odhadu posuzovat v jednotlivých bodech definičního oboru. Pokud však chceme jednoznačně porovnat dva odhady, bude vhodnější použít průměrnou střední čtvercovou chybu (MMSE).

Definice 2.2.4 *Průměrná střední čtvercová chyba* (MSE) *odhadu* $\widehat{y}(t)$ *funkce* $f(t)$ *definované na* $T \subset \mathbb{R}$ *je definovaná jako*

$$MMSE(\widehat{y}) = \int_T E(\widehat{y}(t) - f(t))^2 \, dt, \tag{2.16}$$

pokud tato střední hodnota existuje.

Přehled výsledných odhadů MMSE je shrnut v tabulce (2.1). Na obrázku 2.6 jsou vykresleny odhady MSE pro jednotlivé typy dat a bází. Zeleně je vyznačená MSE odhadu zkonstruovaného pomocí B-splajn báze, žlutě pak MSE Fourierova odhadu. Z výsledků je zřejmě, že pro periodická data je lepší volit Fourierovu bázi. Odhad vytvořený pomocí B-splajn báze neodpovídá skutečnosti pro hodnoty na okrajích definičního oboru, ale celkový rozdíl není zdaleka tak výrazný jako u neperiodických dat, kde se jasně jako lepší ukázala B-splajn báze. Fourierova báze v tomto případě dává velmi špatné výsledky na okrajích definičního oboru.



Obrázek 2.6: Porovnání Fourierovy báze a B-splajn báze - zeleně je vyznačena MSE B-splajn odhadu, žlutě MSE Fourierova odhadu.

Kapitola 3

Analýza funkcionálních dat

3.1 Definice základních charakteristik

Pro funkcionální data, stejně jako pro bodová pozorování, definujme základní charakteristiky, které jsou nezbytné pro další analýzu. V celé této kapitole budeme předpokládat, že máme n fukcionálních pozorování $y_i(t)$, kde $t \in T$ a T je nějaký časový interval.

Průměrová funkce:

$$\bar{y}(t) = \frac{1}{n} \sum_{i=1}^{n} y_i(t), \quad t \in T. \tag{3.1}$$

Rozptylová funkce:

$$\mathrm{var}_y(t) = \frac{1}{n-1} \sum_{i=1}^{n} [y_i(t) - \bar{y}(t)]^2, \quad t \in T. \tag{3.2}$$

Kovarianční funkce:

$$\mathrm{cov}_y(t_1, t_2) = \frac{1}{n-1} \sum_{i=1}^{n} [y_i(t_1) - \bar{y}(t_1)] [y_i(t_2) - \bar{y}(t_2)], \quad t_1, t_2 \in T. \tag{3.3}$$

Korelační funkce:

$$\mathrm{cor}_y(t_1, t_2) = \frac{\mathrm{cov}_y(t_1, t_2)}{\sqrt{\mathrm{var}_y(t_1)\,\mathrm{var}_y(t_2)}}, \quad t_1, t_2 \in T. \tag{3.4}$$

K popisu závislosti dvou funkcionálních veličin slouží následující dvě funkce:

Cross-kovarianční funkce:

$$\mathrm{cov}_{x,y}(t_1,t_2)=\frac{1}{n-1}\sum_{i=1}^{n}\left[x_i(t_1)-\bar{x}(t_1)\right]\left[y_i(t_2)-\bar{y}(t_2)\right],\quad t_1,t_2\in T \tag{3.5}$$

Cross-korelační funkce:

$$\mathrm{cor}_{x,y}(t_1,t_2)=\frac{\mathrm{cov}_{x,y}(t_1,t_2)}{\sqrt{\mathrm{var}_x(t_1)\,\mathrm{var}_y(t_2)}},\quad t_1,t_2\in T. \tag{3.6}$$

3.1.1 Intervaly spolehlivosti

V této kapitole uvedeme obecný tvar intervalu spolehlivosti pro odhad $\widehat{\boldsymbol{y}}$, jak je uveden v článku [7], a porovnáme ho s intervalem spolehlivosti uvedeným v knize [5].

Budeme předpokládat, že máme pouze jeden funkcionální odhad $\widehat{\boldsymbol{y}}(t)$, kde $t\in T$, spočítaný na základě dat $y(t_1),\ldots,y(t_J)$, kde $t_1,\ldots,t_J\in T$. V článcích [13] a [8] je dokázáno, že splajny jsou Bayesovské odhady s Gaussovu apriorní hustotou, pro které platí

$$\widehat{\boldsymbol{y}}=\boldsymbol{S}_{\lambda,\Phi}\boldsymbol{y},$$
$$Var(\widehat{\boldsymbol{y}})=\sigma^2\boldsymbol{S}_{\lambda,\Phi}.$$

Parametr σ^2 se odhadne jako

$$\widehat{\sigma}^2=\frac{SSE}{n-tr(\boldsymbol{S}_{\lambda\Phi})}.$$

Potom $100(1-\alpha)\,\%$ interval spolehlivosti má mee

$$\widehat{y}(t_i)\pm z_{\alpha/2}\widehat{\sigma}\sqrt{(\boldsymbol{S}_{\lambda\Phi})_{ii}}, \tag{3.7}$$

kde $z_{\alpha/2}$ značí $\alpha/2$ kvantil normovaného normálního rozdělení.

Přístup z knihy [5] se liší odhadem rozptylu $\widehat{\boldsymbol{y}}(t)$. Tento přístup předpokládá existenci n funkcionálních odhadů téže funkce (n funkcionálních pozorování). Zajímá nás rozptyl odhadů $\widehat{\boldsymbol{y}}(t)$

$$\sigma_e^2(t_j)=\frac{1}{n-1}\sum_{i=1}^{n}\left[y_i(t_j)-\boldsymbol{c}_j'\phi(t)\right]^2.$$

Dále označíme

$$\Sigma_e^2 = \left(\widehat{\sigma}_e^2\left(t_1\right), \ldots, \widehat{\sigma}_e^2\left(t_m\right)\right)'. \tag{3.8}$$

Předpokládáme, že rozptyl funkcionálních dat je také funkcionální pozorování, proto ho vyhladíme pomocí metod popsaných va kapitole 2. Nechť $\{\psi_r(t), r = 1, \ldots, R\}$ je vhodná báze, pak

$$\widehat{\sigma}_e^2\left(t_j\right) = \sum_{r=1}^{R} b_r \psi_r\left(t_j\right). \tag{3.9}$$

$$\Sigma_e^2 = \boldsymbol{\Psi b}. \tag{3.10}$$

Nyní můžeme jako odhad $\widehat{\sigma}$ z výrazu (3.7) v bodě t_j použít

$$\widehat{\sigma}\left(t_j\right) = \Sigma_e\left(t_j\right)\sqrt{\left(\boldsymbol{S}_{\lambda\Phi}\right)_{ii}}. \tag{3.11}$$

Porovnání obou typů intervalů spolehlivosti pro data z příkladu 1 je na obrázku 3.1. Pro vytvoření odhadu byla použita B-splajn báze (pro Fourierovu báze jsou výsledky obdobné). Zeleně plně je znázorněn odhad založený na první replikaci, zelená přerušovaná je odhadovaná funkce, červeně je vyznačen 95% interval spolehlivosti z článku [7] a žlutě 95% interval z knihy [5]. Je vidět, že v obou případech je širší interval založený pouze na jedné replikaci, tudíž obsahující menší informaci.

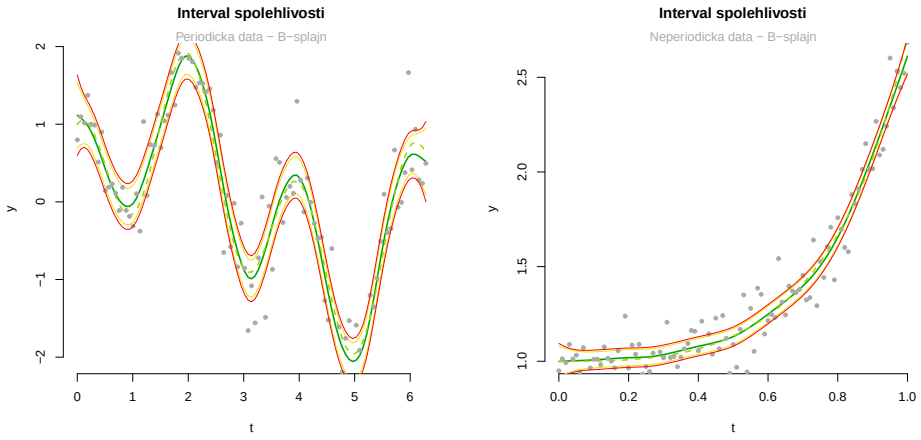
3.2 Concurrent model pro funkcionální data

Cílem je najít způsob, jak modelovat závislost dvou (popř. více) funkcionálních náhodných veličin. Použijeme model popsaný v [10] jako *concurrent model*.

3.2.1 Popis modelu

Jedná se o model, který je v podstatě funkcionálním rozšířením regresního modelu. Tento model vysvětluje funkcionální náhodnou veličinu y pomocí jiných funkcionálních náhodných veličin x_j, kde $j = 1, \ldots, q$. Proměnná q značí počet vysvětlujících proměnných. Obecný tvar modelu se dá zapsat jako

$$y_i(t) = \sum_{j=1}^{q} x_{ij}(t)\,\beta_j(t) + \epsilon_i(t), \tag{3.12}$$



Obrázek 3.1: Intervaly spolehlivosti - zeleně plně odhad založený na první replikaci, zeleně přerušovaně odhadovaná funkce, červeně 95% interval spolehlivosti z článku [7] a žlutě 95%f interval z knihy [5].

kde $i = 1, \ldots, n$ je počet pozorování a pro jednoduchost uvažujme $t \in [0, 1]$. Velmi často $x_{i0} = 1$, potom je ve sčítanci $\beta_0(t)$ obsažena variabilita, kterou nelze vyjádřit pomocí proměnných $x_j(t)$. Dále předpokládejme, že náhodné složky modelu (3.12) $\boldsymbol{\epsilon_i(t)}$ jsou nezávislé a stejně rozdělené. V doslovném překladu bychom mohli použít označení *souběžný model*. Tento název vystihuje to, že hodnoty náhodná veličina y v čase t jsou modelovány pomocí hodnot náhodných veličin x_j ve shodném čase. Maticový zápis výrazu 3.12:

$$\boldsymbol{y}(t) = \boldsymbol{Z}(t)\boldsymbol{\beta}(t) + \boldsymbol{\epsilon}(t). \tag{3.13}$$

Optimální parametr $\boldsymbol{\beta}$ budeme hledat obdobně jako v klasickém regresním modelu. Situace je však v tomto případě komplikovanější, protože jednotlivé β_j nejsou konstanty, ale funkce v čase t.

3.2.2 Minimalizační kritérium

Nyní se budeme zabývat vhodným minimalizačním kritériem pro nalezení optimálního odhadu pro parametr $\boldsymbol{\beta}$. Pokud bychom zvolili prosté rozšíření regresního součtu nejmenších čtverců (2.3), výsledný model by byl zatížen příliš vysokým rozptylem.

$$SSE = \int_T (\boldsymbol{y}(t) - \boldsymbol{Z}(t)\boldsymbol{\beta}(t))'(\boldsymbol{y}(t) - \boldsymbol{Z}(t)\boldsymbol{\beta}(t))\,dt \tag{3.14}$$

Z tohoto důvodu rozšíříme výraz (3.14) o penalizační část, která nám zaručí určitou míru hladkosti výsledného modelu. Výsledné minimalizační kritérium má tvar

$$\begin{aligned} LMSSE\left(\boldsymbol{\beta}\right) &= \int_T \left(\boldsymbol{y}\left(t\right) - \boldsymbol{Z}\left(t\right)\boldsymbol{\beta}\left(t\right)\right)' \left(\boldsymbol{y}\left(t\right) - \boldsymbol{Z}\left(t\right)\boldsymbol{\beta}\left(t\right)\right) dt + \\ &+ \sum_{j=1}^{q} \lambda_j \int_T \left[L_j \beta_j\left(t\right)\right]^2 dt, \end{aligned} \tag{3.15}$$

kde L je označení diferenciálního operátoru. Optimální velikost parametru $\boldsymbol{\lambda}$ lze opět určit cross-validací. Dále budeme předpokládat, že pro regresní funkce β_j existuje rozvoj

$$\beta_j\left(t\right) = \sum_{k=1}^{K_j} b_{kj}\theta_{kj}\left(t\right) = \boldsymbol{\theta}_j\left(t\right)'\boldsymbol{b}_j, \tag{3.16}$$

kde K_j je počet funkcí báze $\{\theta_{kj}\}$. Dále definujme matici

$$\boldsymbol{b} = \left(\boldsymbol{b_1}', \boldsymbol{b_2}', \ldots, \boldsymbol{b_q}'\right)',$$

která má q řádků a $K_\beta = \sum_{j=1}^{q} K_j$ sloupců. Potom $\boldsymbol{\Theta}\left(t\right)$ je definována jako

$$\boldsymbol{\Theta}\left(t\right) = \begin{bmatrix} \boldsymbol{\theta}_1'\left(t\right) & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\theta}_2'\left(t\right) & \ldots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \boldsymbol{\theta}_q'\left(t\right) \end{bmatrix}. \tag{3.6}$$

Potom $\boldsymbol{\beta}\left(t\right) = \boldsymbol{\Theta}\left(t\right)\boldsymbol{b}$ a výraz (3.13) je možné zapsat jako

$$\boldsymbol{y}\left(t\right) = \boldsymbol{Z}\left(t\right)\boldsymbol{\Theta}\left(t\right)\boldsymbol{b} + \boldsymbol{\epsilon}\left(t\right).$$

Dále definujme blokovou diagonální matici $\boldsymbol{R}$ s j bloky

$$\lambda_j \int_T \left[L_j\boldsymbol{\theta}_j\left(t\right)\right]' \left[L_j\boldsymbol{\theta}_j\left(t\right)\right] dt.$$

Minimalizační kritérium LMSSE (3.15) je možné upravit následujícím způsobem:

$$\begin{aligned} LMSSE\left(\boldsymbol{\beta}\right) &= \int_T \left[\boldsymbol{y}\left(t\right)'\boldsymbol{y}\left(t\right) - 2\boldsymbol{b}'\boldsymbol{\Theta}\left(t\right)'\boldsymbol{Z}\left(t\right)'\boldsymbol{y}\left(t\right) + \right. \\ &\left. + \boldsymbol{b}'\boldsymbol{\Theta}\left(t\right)'\boldsymbol{Z}\left(t\right)'\boldsymbol{Z}\left(t\right)\boldsymbol{\Theta}\left(t\right)\boldsymbol{b}\right] dt + \boldsymbol{b}'\boldsymbol{R}\left(\lambda\right)\boldsymbol{b}. \end{aligned} \tag{3.18}$$

Odhad parametru $\boldsymbol{b}$ je řešením soustavy normálních rovnic. Tuto soustavu získáme, když položíme derivaci $LMSSE(\boldsymbol{\beta})$ rovnu nule. Pokud označíme

$$\boldsymbol{A} = \int_T \boldsymbol{\Theta}(t)' \boldsymbol{Z}(t)' \boldsymbol{Z}(t) \boldsymbol{\Theta}(t)\, dt + \boldsymbol{R}(\lambda), \tag{3.20}$$

$$\boldsymbol{d} = \int_T \boldsymbol{\Theta}(t)' \boldsymbol{Z}(t)' \boldsymbol{y}(t)\, dt, \tag{3.21}$$

pak soustavu normálních rovnic můžeme zapsat takto

$$\boldsymbol{A}\widehat{\boldsymbol{b}} = \boldsymbol{d}. \tag{3.22}$$

V některých případech je možné řešení soustavy (3.22) vyjádřit explicitně, ale obecně je vhodné tuto soustavu řešit numerickými metodami integrace.

3.2.3 Intervaly spolehlivosti pro regresní parametry

Pro konstrukci intervalů spolehlivosti pro regresní parametr $\boldsymbol{\beta}$, je zapotřebí nejprve spočítat rezidua modelu (3.12). Pro p-té pozorování i-té replikace bude mít příslušné reziduum tvar

$$r_{pi} = y_{pi} - \sum_{j=1}^{q} x_{ij}(t_p)\, \beta_j(t_p), \tag{3.23}$$

což se v maticovém zápisu dá přepsat jako

$$r_{pi} = y_{pi} - \boldsymbol{Z}_i(t_p)\, \boldsymbol{\beta}(t_p), \tag{3.24}$$

kde $\boldsymbol{Z}_i$ značí i-tý řádek matice $\boldsymbol{Z}$. Pokud existují y_{pi} pro všechna $i = 1, \ldots, n$ a $p = 1, \ldots, P$, je možné zkonstruovat odhad kovarianční matice reziduí. Tento odhad má následující tvar

$$\Sigma^* = \frac{1}{n}\boldsymbol{r}\boldsymbol{r}', \tag{3.25}$$

kde $\boldsymbol{r}$ je matice reziduí. Je nutné si uvědomit, že variabilita obsažená v matici Σ^* obsahuje také variabilitu z modelu, který byl použit pro vyhlazení dat (viz (2.2)). Je důležité si uvědomit, že y_{ip} jsou pozorovaná data a $y(t)$ je jejich vyhlazení. Z modelu (2.2) vyplývá, že

$$y_i(t) = \boldsymbol{c}_{\boldsymbol{i}}'\boldsymbol{\phi}(t),$$

kde $\boldsymbol{c}_i$ je vektor regresních koeficientů a $\boldsymbol{\phi}(t) = (\phi_1(t), \ldots, \phi_L(t))'$ je příslušná báze. Pokud dále označíme $\boldsymbol{C} = (\boldsymbol{c}_1', \ldots, \boldsymbol{c}_n')'$, platí

$$\boldsymbol{y}(t) = \boldsymbol{C}\phi(t). \tag{3.26}$$

Jestliže toto vyjádření $\boldsymbol{y}(t)$ dosadíme do soustavy normálních rovnic (3.22), dostaneme

$$\begin{aligned} \widehat{\boldsymbol{b}} &= \boldsymbol{A}^{-1} \int_T \boldsymbol{\Theta}'(t)\, \boldsymbol{Z}'(t)\, \boldsymbol{C}\boldsymbol{\phi}(t)\, dt = \\ &= \boldsymbol{A}^{-1} \left[\int_T \boldsymbol{\phi}'(t) \otimes (\boldsymbol{\Theta}'(t)\, \boldsymbol{Z}'(t)) dt \right] \mathrm{vec}(\boldsymbol{C}) \end{aligned} \tag{3.27}$$

kde $\mathrm{vec}(\cdot)$ je funkce, která převádí matici na vektor. Pro $\boldsymbol{A} \in \mathbb{R}^{n\times n}$ platí $\mathrm{vec}(\boldsymbol{A}) = (a_1, \ldots, a_n)'$, kde $a_1, \ldots, a_n$ jsou sloupce matice $\boldsymbol{A}$. Symbol $\otimes$ značí Kroneckerův součin. Uvedeme definici z knihy [6].

Definice 3.2.1 *Nechť $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{B} \in \mathbb{R}^{p\times q}$. Potom Kroneckerův součin matic $\boldsymbol{A}$ a $\boldsymbol{B}$ je definován jako matice*

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} a_{11}\boldsymbol{B} & \ldots & a_{1n}\boldsymbol{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\boldsymbol{B} & \ldots & a_{mn}\boldsymbol{B} \end{bmatrix} \in \mathbb{R}^{mp\times nq}. \tag{3.28}$$

Odvození vzorce (3.27) jsme provedli podle následující věty.

Věta 3.2.1 *Pro libovolné matice $\boldsymbol{A}, \boldsymbol{B}$ a $\boldsymbol{C}$, pro které je definován součin $\boldsymbol{ABC}$, platí*

$$vec(\boldsymbol{ABC}) = (\boldsymbol{C}' \otimes \boldsymbol{B})\, vec(\boldsymbol{A}). \tag{3.29}$$

Důkaz: Viz [6]. $\triangle$

Vzorec (3.27) popisuje závislost řešení soustavy normálních rovnic (3.22) na způsobu vyhlazení původních dat. Nyní přistoupíme k vyjádření rozptylu odhadu $\widehat{\boldsymbol{b}}$.

$$\begin{aligned} \mathrm{var}\left(\widehat{\boldsymbol{b}}\right) &= \mathrm{E}\left(\widehat{\boldsymbol{b}} - \mathrm{E}\widehat{\boldsymbol{b}}\right)\left(\widehat{\boldsymbol{b}} - \mathrm{E}\widehat{\boldsymbol{b}}\right)' = \\ &= \boldsymbol{A}^{-1} \left[\int_T \boldsymbol{\Phi}(t) \otimes (\boldsymbol{\Theta}'(t)\, \boldsymbol{Z}'(t)) dt \right] \mathrm{E}\left(\mathrm{vec}(\boldsymbol{C}) - \mathrm{E}(\mathrm{vec}(\boldsymbol{C}))\right) \cdot \\ &\cdot \quad \left(\mathrm{vec}(\boldsymbol{C}) - \mathrm{E}(\mathrm{vec}(\boldsymbol{C}))\right)' \left(\boldsymbol{A}^{-1} \left[\int_T \boldsymbol{\Phi}(t) \otimes (\boldsymbol{\Theta}'(t)\, \boldsymbol{Z}'(t)) dt \right] \right)' = \\ &= \boldsymbol{F}\boldsymbol{J}\boldsymbol{F}', \end{aligned} \tag{3.30}$$

kde $\boldsymbol{F} \in \mathbb{R}^{K_\beta \times nL}$ a

$$\boldsymbol{F} = \boldsymbol{A}^{-1}\left[\int_T \boldsymbol{\Phi}(t) \otimes (\boldsymbol{\Theta}'(t)\,\boldsymbol{Z}'(t))dt\right],$$

přičemž platí $\widehat{\boldsymbol{b}} = \boldsymbol{F}\mathrm{vec}(\boldsymbol{C})$. Dále $\boldsymbol{J} \in \mathbb{R}^{nL \times nL}$ je bloková diagonální matice s bloky $\boldsymbol{G}\Sigma^*\boldsymbol{G}'$, kde $\boldsymbol{G} \in \mathbb{R}^{L \times L}$, $\boldsymbol{G} = (\boldsymbol{\Phi'\Phi} + \lambda\boldsymbol{R})^{-1}\boldsymbol{\Phi'}$ (viz výraz (2.7)) a platí $\boldsymbol{C} = \boldsymbol{G}\boldsymbol{y}$. Meze $100(1-\alpha)\,\%$ intervalu spolehlivosti odhadu parametru $\boldsymbol{\beta}$ budou mít tvar

$$\widehat{\boldsymbol{\beta}} \pm z_{\alpha/2}\sqrt{\mathrm{var}\left(\widehat{\boldsymbol{b}}\right)\boldsymbol{\Theta}_{ii}}, \tag{3.31}$$

kde $\widehat{\boldsymbol{\beta}}(t) = \boldsymbol{\Theta}(t)\,\widehat{\boldsymbol{b}}$.

3.2.4 Příklad 2

V tomto příkladu budeme srovnávat spolehlivost modelu v závislosti na různém rozptylu modelovaných veličin. Nejprve budeme uvažovat pouze rozptyl závislé proměnné, poté přidáme rozptyl concurrent modelu a nakonec se zamyslíme nad možnou variabilitou nezávislé proměnné.

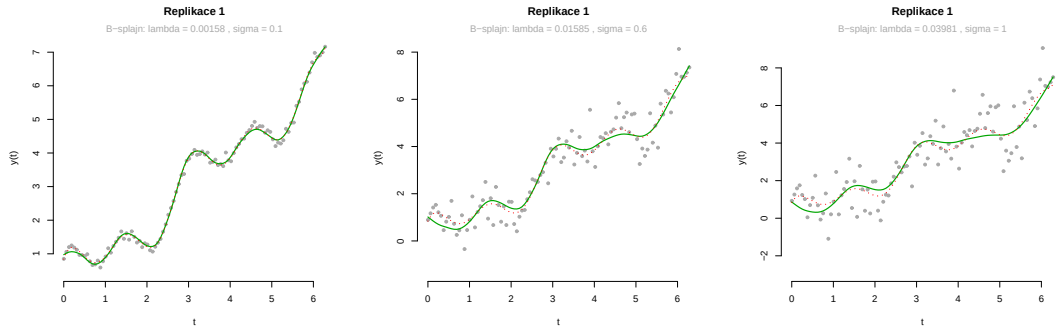
Pro výpočty použijeme interval $T = [0, 2\pi]$, počet replikací $n = 50$. Nezávislé proměnné budou tvaru $x_{i1}(t) = 1$ a $x_{i2}(t) = \sin(t) + \cos(\pi t)$ pro $t \in T$. Dále definujeme hodnoty regresních funkcí a následně dopočítáme hodnoty závislé proměnné v čase. Tento otočený postup nám umožní posoudit, jak dobře model pracuje. Jako regresní funkce zvolíme $\beta_0(t) = t$ a $\beta_1(t) = \cos(t)$. Zbývá určit tvar závislé proměnné $\boldsymbol{y}$. Platí

$$y_i(t_p) = t_p + \cos(t_p)\left[\sin(t_p) + \cos(\pi t_p)\right] + \epsilon(t_p),$$

přičemž $(t_1, \ldots, t_P)$ je ekvidistantní dělení intervalu T, $P = 101$ a rozdělení náhodné složky ϵ je $N(0, \sigma^2)$. Za σ budeme postupně brát 0,1, 0,6 a 1. Na obrázku (3.2) jsou zobrazeny hodnoty $\boldsymbol{y}_1(t_p)$ (šedé body) spolu s odhady $\widehat{\boldsymbol{y}}_1(t_p)$ (zelené křivky). K vyhlazení byl použit kubický B-splajn s penalizací druhého řádu, uzly v bodech t_p a parametrem λ, který byl určen zobecněnou cross-validací. Hodnoty parametru λ pro různé hodnoty σ jsou v tabulce (3.1). Tabulka dále obsahuje průměrnou $RMSE$, která se spočítá jako průměr $RMSE_i$ jednotlivých křivek. Definice $RMSE_i$ pro model (3.13) je

$$RMSE_i = \sqrt{\frac{1}{n}(\boldsymbol{y}_i - \boldsymbol{Z}_i\boldsymbol{\beta})'(\boldsymbol{y}_i - \boldsymbol{Z}_i\boldsymbol{\beta})} \tag{3.32}$$

Dále odhadneme parametr $\boldsymbol{\beta}$ modelu (3.13). Při výpočtu minimalizačního kritéria (3.15) položíme $\lambda_c = 0,00001$ (označení λ_c budeme i nadále používat pro parametr λ concurrent modelu). Kód pro výpočet v softwaru R je na přiloženém CD.



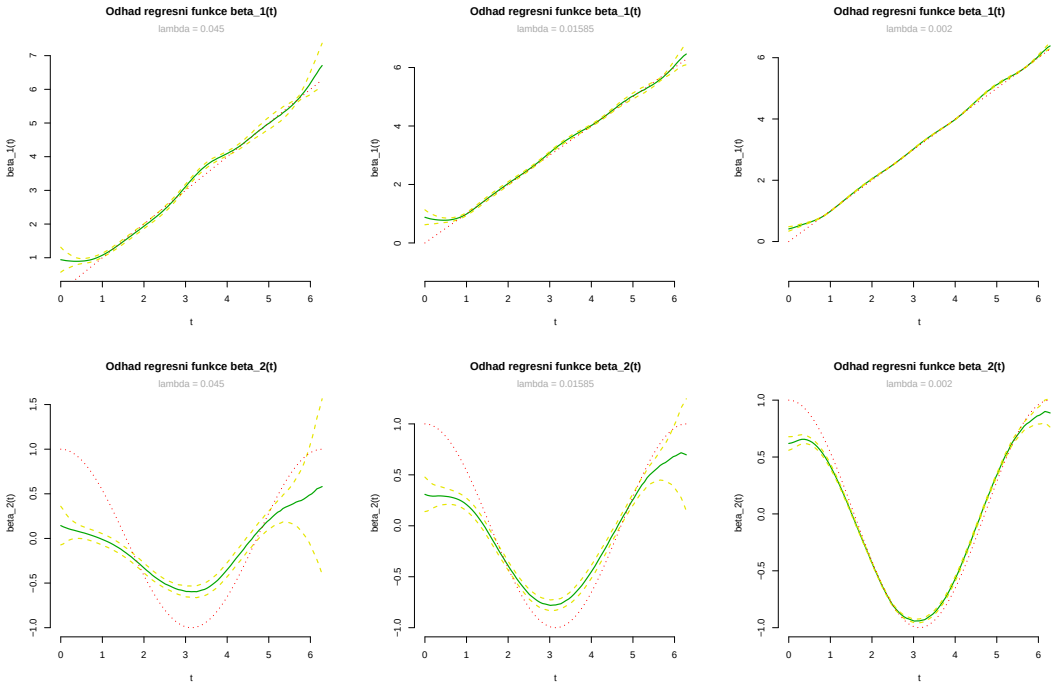
Obrázek 3.2: Zobrazení $y_1(t_p)$ (šedě), B-splajn odhad (zeleně), skutečná hodnota $y(t)$ (červěně tečkovaně).

Typ báze	B-splajn		
σ	$\mathbf{0,1}$	$\mathbf{0,6}$	$\mathbf{1,0}$
λ	$0,002$	$0,016$	$0,040$
průměrná $RMSE$	$0,100$	$0,595$	$0,995$

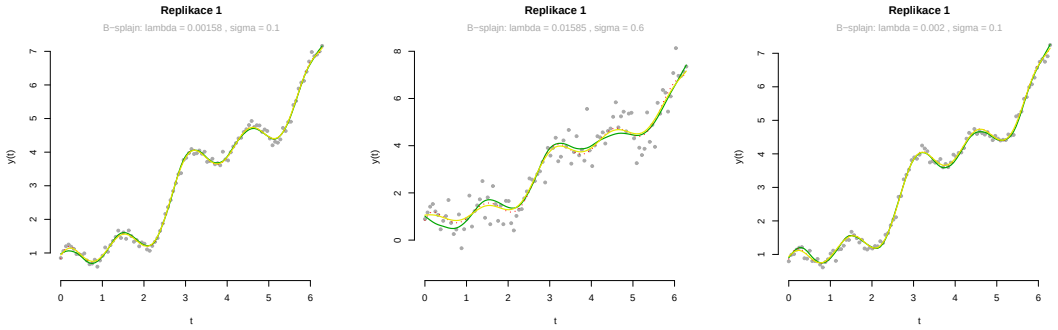
Tabulka 3.1: Hodnoty parametru λ pro různá σ

Na obrázku (3.3) jsou vykresleny odhady regresních funkcí $\boldsymbol{\beta}$ (zelená křivka) s 95% intervaly spolehlivosti (žlutá přerušovaná křivka) a pro srovnání je zobrazena i funkce, která byla použita při generování dat. Je vidět, že se původní funkce od odhadu liší, a to tím více, čím je vyšší rozptyl $\boldsymbol{\epsilon}$. Dále stojí za povšimnutí širší interval spolehlivosti na okrajích intervalu T. Vyplývá to z vlastnosti B-splajnu, který je na okrajích definičího intervalu méně stabilní, a proto zde mají odhady vyšší rozptyl. Na obrázku (3.4) je vidět srovnání původní funkce použité při generování dat (červená tečkovaná křivka), odhad této funkce pomocí B-splajnu (zelená křivka) a odhadu, který dává concurrent model (žlutá křivka). Je vidět, že se B-splajn odhad od simulační křivky lišil více než výsledný odhad. Je to způsobeno tím, že informace obsažená v concurrent modelu je větší než informace, se kterou pracuje B-splajn model. Dále je patrné, že výsledný odhad má méně ostré lokální extrémy, což kopíruje rozdíl mezi funkcí $\boldsymbol{\beta}$ a jejím odhadem. Tento rozdíl je patrnější u dat s vyšším rozptylem (stejně jako u regresní funkce).

Do této chvíle jsme pracovali s daty, kde k $\boldsymbol{x}_{i1}$ a $\boldsymbol{x}_{i2}$ byly shodné pro všechna $i = 1, \ldots, 50$. Nyní jako nezávislé proměnné použijeme funkce, které zkonstruujeme jako hladké odhady $\boldsymbol{x}_{i1}(t_p) + \psi(t_p)$ a $\boldsymbol{x}_{i2}(t_p) + \psi(t_p)$, přičemž ψ je náhodná, normálně rozdělená veličina s nulovou střední hodnotou a rozptylem $0,5^2$. Tvar regresní funkce zůstává stejný a pro konstrukci pozorování $\boldsymbol{y}$ použijeme $\sigma = 0,6$. Budeme testovat vliv



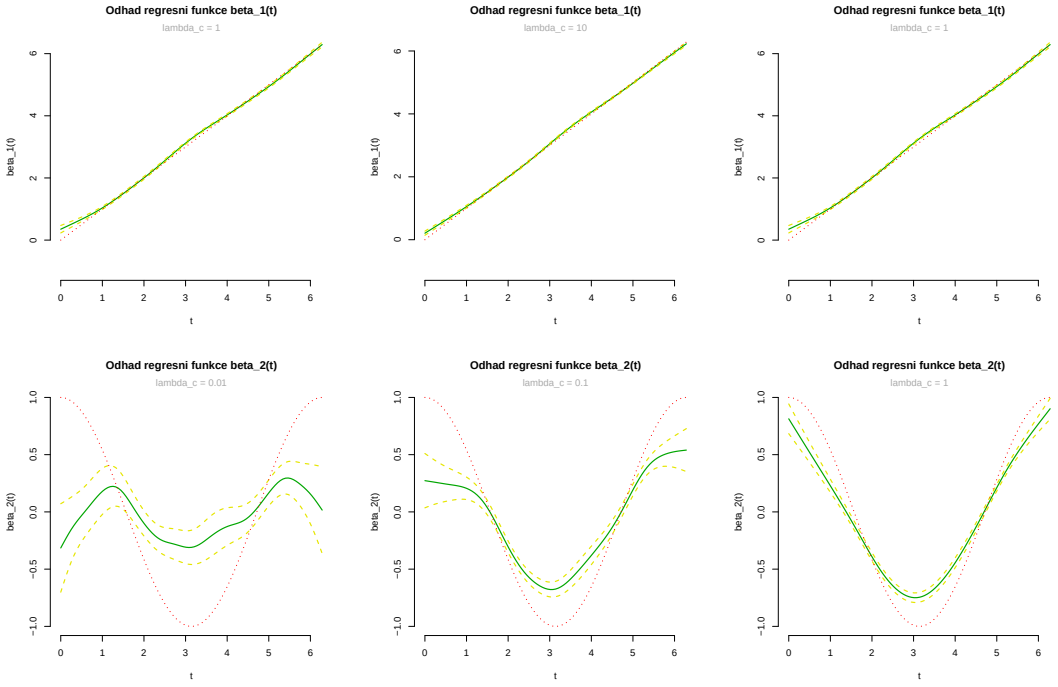
Obrázek 3.3: Zobrazení odhadů regresní funkce $\beta(t)$ (zeleně), 95% intervaly spolehlivosti (žlutě), hodnota $\beta(t)$ použitá při generování dat (červěně tečkovaně).



Obrázek 3.4: Zobrazení $y_1(t_p)$ (šedě), B-splajn odhad (zeleně), skutečná hodnota $y(t)$ (červěně tečkovaně) a výsledný odhad z concurrent modelu (žlutě).

λ_c	**0, 001**	**0, 1** (CV)	**1**
průměrná $RMSE$	0, 5958	0, 6002	0, 6050

Tabulka 3.2: Hodnoty parametru λ pro různá σ

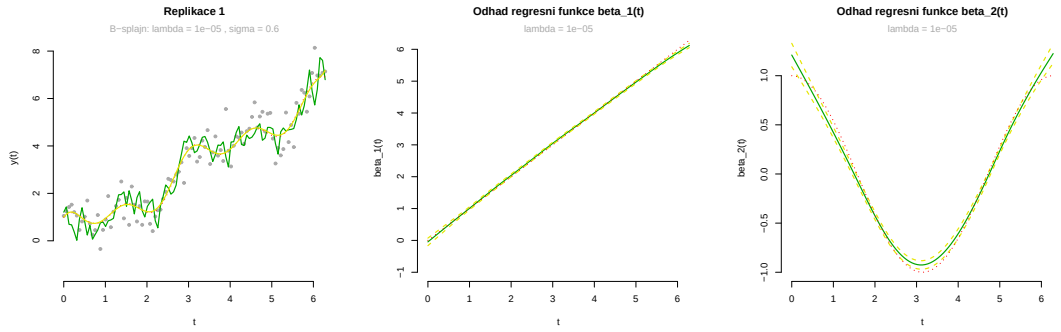


Obrázek 3.5: Zobrazení odhadů regresní funkce $\beta(t)$ (zeleně), 95% intervaly spolehlivosti (žlutě), hodnota $\beta(t)$ použitá při generování dat (červěně tečkovaně).

parametru λ_c jednak na tvar odhadu regresní funkce, ale také na celkovou přesnost modelu (velikost průměrné RMSE). V tabulce (3.2) jsou shrnuty hodnoty průměré RMSE pro různé hodnoty λ_c a na obrázku 3.5 jsou zobrazeny příslušné odhady regresních funkcí.

Z výsledku je patrné, že odhad regresní funkce silně závisí na volbě parametru λ_c. Je však zajímavé, že predikční schopnost modelu se příliš nemění. Ve všech případech je dobrá. Pokud by rozptyl regresních funkcí vysoký, s rostoucí hodnotou parametru λ_c by se kvalita odhadu začala postupně zhoršovat.

Nyní se podíváme, co se stane, když odhad závislé funkcionální proměnné bude málo penalizovaný. Data použijeme stejná jako v předchozí části, $lambda_c = 1$ a $\lambda = 0, 00001$. Odhady regresní funkce jsou na obrázku (3.6) a průměrná RMSE je 0,5907. Pokud



Obrázek 3.6: Zobrazení odhadů regresní funkce $\beta(t)$ (zeleně), 95% intervaly spolehlivosti (žlutě), hodnota $\beta(t)$ použitá při generování dat (červěně tečkovaně).

výsledky porovnáme s výsledky pro $\lambda = 0,016$ (výsledek zobecněné cross-validace), je vidět, že pro horší odhad proměnné $\mathbf{y}$ dává model lepší odhad regresní funkce a zároveň predikční schopnost modelu je stále velmi dobrá.

3.3 Aplikace na reálná data

Jak již bylo zmíněno v úvodu, jedním z cílů této práce je najít vhodný model, který by popisoval vzájemnou závislost objemové aktivity radonu (dále OAR) naměřené na různých místech testovaného objektu. K řešení této úlohy využijeme výše uvedené postupy. Začneme tím, že se podrobněji podíváme na strukturu dat, se kterými budeme dále pracovat.

3.3.1 Popis dat

Data byla naměřena ve dnech 3.10.2008 až 20.10.2008 v rodinném domě Lažný v rámci studie vlivu užívání stavby na výsledné hodnoty OAR. Měřila se objemová aktivita radonu (Rn·m^3), teplota (°C) a relativní vlhkost (%). Měření probíhalo na pěti různých místech, v různých časových okamžicích. Přehled časové struktury měření je v tabulce 3.3. V dalším textu budeme pro jednotlivá místa a příslušné datové řady používat označení M1 až M5 (viz 3.3).

Z údajů v tabulce (3.3) je vidět, že první tři datové řady nemohou být pravidelné. Skutečně zde došlo k situaci, že interval mezi dvěma měřeními je odlišný od všech ostatních. V prvním případě tato situace mezi měřeními z 14.10.2008. První z nich proběhlo v 16:10, a další následovalo až v 16:56. V druhém případě se jednalo opět o

Místo měření	Počátek měření	Konec měření	Časový interval měření
Dětský pokoj - M1	3.10.2008 16:10	20.10.2008 13:56	30 min
Chodba - M2	3.10.2008 16:04	20.10.2008 13:55	30 min
Kuchyň - M3	3.10.2008 15:58	20.10.2008 13:52	30 min
Kuchyň 1 - M4	3.10.2008 16:24	20.10.2008 15:00	2 min
Obývací pokoj -M5	3.10.2008 17:00	20.10.2008 15:00	60 min

Tabulka 3.3: Časová struktura měření

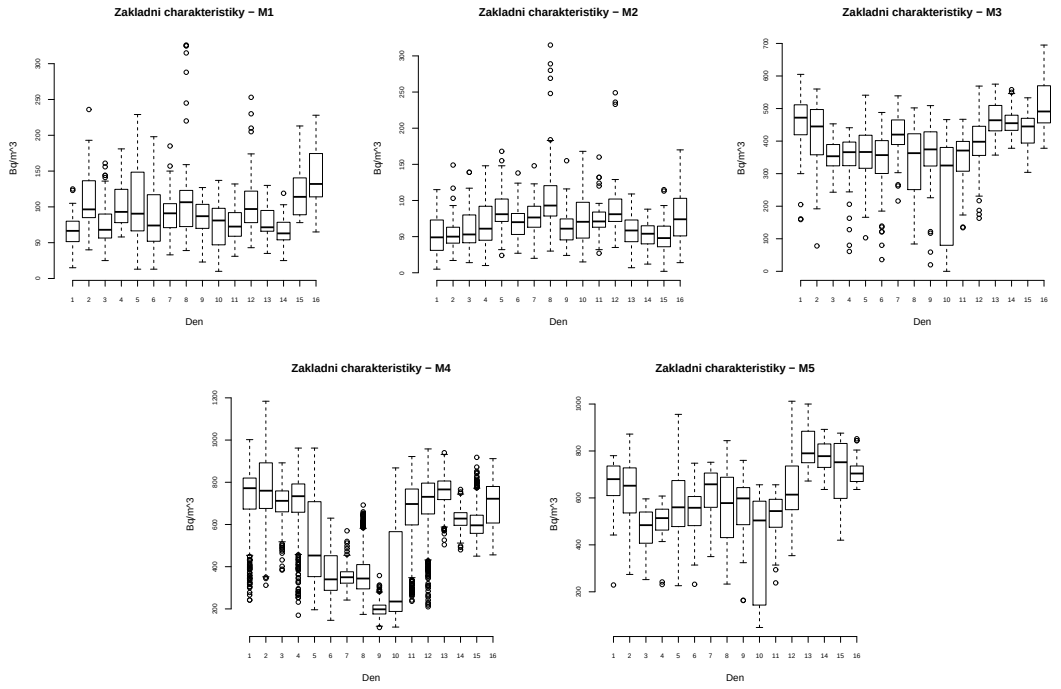
měření s datem 14.10.2008, ale tentokrát v časech 16:04 a 16:55. V posledním případě se jedná stále o 14.10.2008, ale časy jsou 15:58 a 16:52. Za funkcionální náhodnou veličinu budeme považovat vyhlazený průběh OAR v jednom dni.

Pro vytvoření funkcionálních dat použijeme postupy popsané v kapitole (2.1). Nejprve pozorování rozdělíme podle data měření. Pro první a poslední den není k dispozici dostatek dat, proto je z další analýzy vyřadíme. Získáme tedy 16 řad pro každé místo měření. Na obrázku (3.3.1) jsou graficky znázorněny základní charakteristiky pro jednotlivé řady naměřených hodnot (jedná se o klasický krabicový graf, kde střední tučná čára značí medián, spodní a horní okraje obdélníku značí první a třetí kvartil). V datech M3 a M4 chybí po jednom pozorování. V prvním případě chybí jak čas měření, tak naměřená hodnota. V datech M4 chybí pouze naměřená hodnota. Jednou z možností, jak tuto situaci řešit, by bylo pozorování úplně vynechat. Potom by se však naměřené hodnoty nedaly zapsat do jedné matice a optimální vyhlazení kratší řady by muselo být provedeno zvlášť. To je při výpočtech velmi nepraktické, proto chybějící hodnotu nahradíme lineární aproximací. Pro zjednodušení další práce normalizujeme časové okamžiky na interval $[0, 1]$

Z těchto grafů je patrná podobnost mezi M1, M2 a M3, M5. Budeme hledat model pro vztah mezi M1 a M2. Na první pohled se zdají hodnoty těchto veličin podobné. Dá se to předpokládat i proto, že se místa měření nacházela blízko sebe.

3.3.2 Vytvoření fukcionálních dat

Pro vyhlazení dat použijeme Fourierovu bázi, protože očekáváme, že průběh OAR bude vykazovat periodicitu. Vyplývá to z toho, že data byla měřena souvisle, a proto by konečná hodnota jednoho dne neměla být příliš rozdílná od počáteční hodnoty dne následujícího. Problém je v tom, že hodnoty nebyly měřeny pro všechny dny ve stejných časech, proto použijeme pouze prvních 11, u kterých je čas konzistentní.

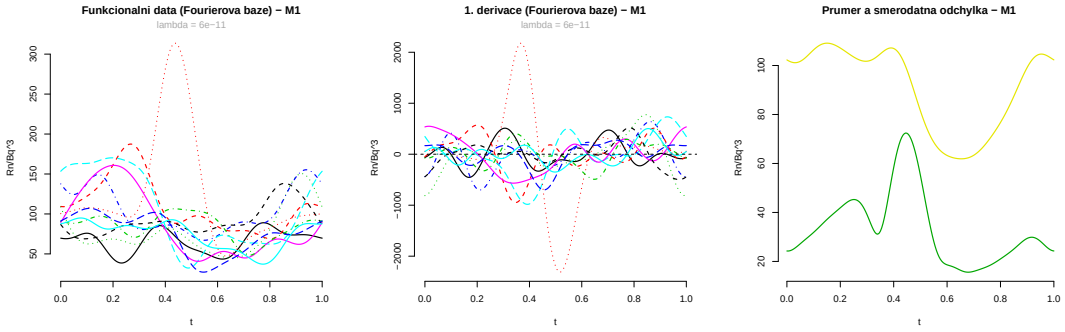


M1

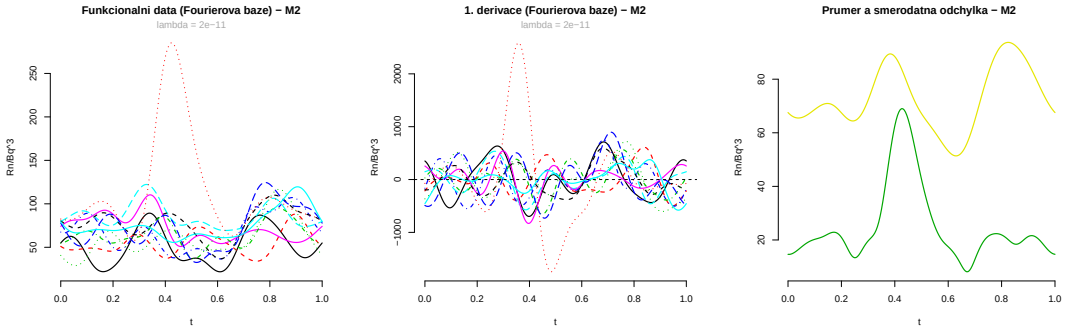
Dimenze báze je 49 (koeficienty modelu odhadneme na základě penalizovaného součtu čtverců (2.5) a PEN_4, proto může být dimenze báze tak vysoká). Parametr $\lambda = 6 \cdot 10^{-11}$ byl určen pomocí zobecněné cross-validace. Normalita reziduí se nepotvrdila pouze u 2. a 11. křivky. Z důvodu nedostatku dat tyto křivky z další analýzy nevyřadíme. Pro testování normality jsme použili W test (viz [12]). Na obrázku 3.8 jsou zobrazeny výsledné odhady, jejich první derivace a na posledním grafu průměr a směrodatná odchylka odhadů.

M2

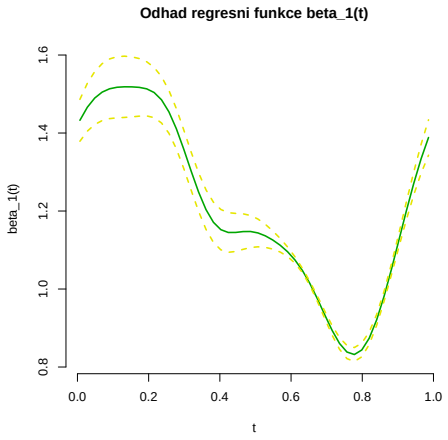
Dimenze báze a penalizační kritérium jsou shodné jako v předchozím případě. Parametr $\lambda = 2 \cdot 10 - 11$ byl opět určen zobecněnou cross-validací. Normalita reziduí se neprokázala u 2., 9. a 10. křivky. Na obrázku 3.8 jsou zobrazeny výsledné odhady, jejich první derivace a na posledním grafu průměr a směrodatná odchylka odhadů.



Obrázek 3.7: Odhady funkcionálních dat, první derivace odhadu, průměr odhadů (žlutě) a směrodatná odchylka odhadů (zeleně)



Obrázek 3.8: Odhady funkcionálních dat, první derivace odhadu, průměr odhadů (žlutě) a směrodatná odchylka odhadů (zeleně)



Obrázek 3.9: Odhad regresní funkce modelu (3.33)

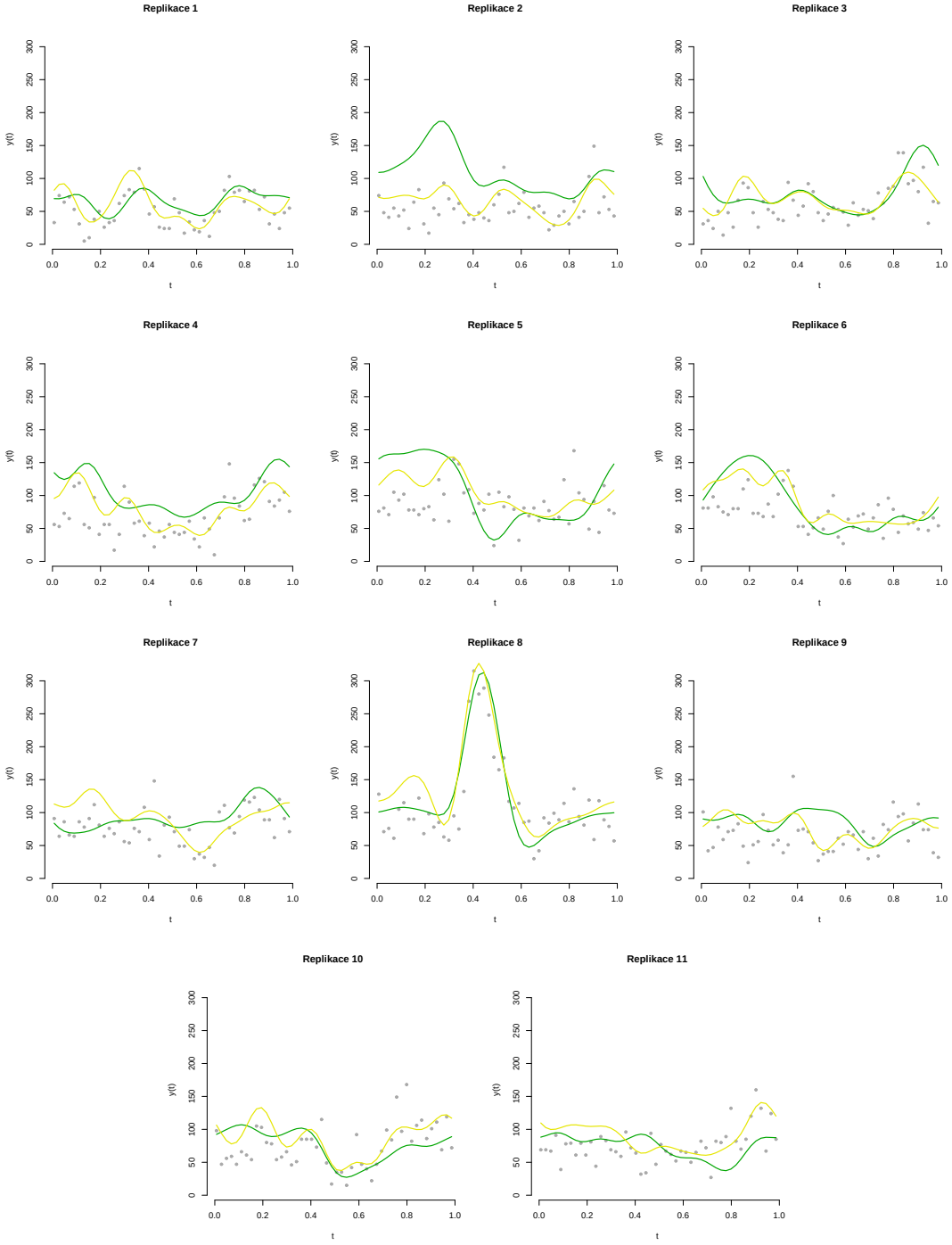
3.3.3 Concurrent model

Budeme odhadovat model

$$M1_i(t) = M2_i(t)\,\beta(t) + \epsilon_i(t), \tag{3.33}$$

kde $t \in \mathbb{R}$. Parametr λ_c jsme určili cross-validací a jeho hodnota je 0,3. Odhad regresní funkce $\beta(t)$ spolu s 95% intervalem spolehlivosti je na obrázku 3.9. Průměrná RMSE vychází 28,83. Na obrázku 3.10 jsou odhady M1 predikované modelem (3.33). Je vidět, že interval spolehlivosti pro odhad $\beta(t)$ je širší v první polovině definičního oboru. Je to způsobeno vyšším rozptylem M1. Tvar funkce odpovídá tomu, že průměr M1 je v první polovině intervalu T vyšší, než průměr M2, proto je regresní funkce větší než jedna. Je vidět, že i v druhé polovině odpovídá průběh regresní funkce skutečnosti.

Tento příklad dobře ukazuje, že metoda skutečně funguje. Pro konstrukci složitějších modelů je nutné provést podrobnější analýzu rozptylu, např. pomocí metody hlavních komponent (viz [11]).



Obrázek 3.10: Odhad regresní funkce modelu (3.33)

Kapitola 4

Výpočetní prostředí

Tato kapitola shrnuje možnosti praktické aplikace postupů popsaných v této práci. Pokud chceme využít již vytvořené procedury, je možné si vybrat ze dvou programovacích jazyků. Prvním z nich je MATLAB. Podrobnější popis k funkcím v tomto jazyce je možné najít v článku [9] nebo v knize [5]. Druhým jazykem, ve kterém existuje implementace funkcionální datové anylýzy je R, popř. S-Plus. Knihovna napsaná v tomto jazyce se jmenuje `fda`. Podrobný popis funkcí a jejich parametrů je možné najít v nápovědě k této knihovně nebo v knize [5], proto nebudeme u jednotlivých funkcí uvádět podrobné vysvětlení všech parametrů, ale pouze ty klíčové. Cílem této kapitoly je vysvětlit hlavní strukturu práce s funkcionálními daty a zároveň zmínit základní funkce této knihovny.

4.1 Základní funkce

Abychom mohli získat z vektoru hodnot funkcionální pozorování, je nutné nejprve vytvořit bázi. Knihovna `fda` umožňuje použít několik typů bází, ale uvedeme pouze výše uvedenou bázi odvozenou od Fourierových řad (viz (2.12)) a b-splajn (viz (2.13)) bázi. Jedná se o dva nejčastěji používané typy. Funkce, které vytváří tyto typy bází, se jmenují následovně:

```
   create.fourier.basis(rangeval=c(0,1), nbasis=3, period=diff(rangeval),
dropind=NULL, ...)
```

parametry:

`rangeval`: Vektor obsahující krajní body definičního oboru báze.
`nbasis`: Celé číslo určující počet funkcí báze.

`period`: Perioda Fourierových funkcí.

`dropind`: Tento parametr umožňuje vyloučit některé funkce z báze (např. nastavení `dropind=1` vytvoří bázi bez absolutního členu).

```
create.bspline.basis(rangeval=NULL,nbasis=NULL, norder=4, breaks=NULL,
dropind=NULL, ...)
```

parametry:

`rangeval`: Vektor obsahující krajní body definičního oboru báze.

`nbasis`: Celé číslo určující počet funkcí báze.

`norder`: Řád splajnu. Platí *řád splajnu = stupeň splajnu - 1.*

`breaks`: Dělící body intervalu, na kterém je b-splajn definovaný (krajní body spolu s vnitřními uzly). Pokud označíme `nbreaks = length(breaks)`, pak platí `nbasis = nbreaks + norder - 2`

`dropind`: Viz `create.fourier.basis`.

Další parametry slouží k přesnější konstrukci výsledné báze. Výsledkem těchto funkcí jsou objekty třídy `basisfd`, což je `list` obsahující informaci o typu báze (`type`), počtu funkcí báze (`nbasis`), definičním oboru (`rangeval`) a dalších parametrech (`params`). Pokud zadáme počet funkcí Fourierovy báze sudý, automaticky je doplněn o chybějící člen. Zobrazení funkcí báze lze získat pomocí funkce `plot()`.

K získání matice konkrétních hodnot báze v daných bodech definičního oboru (matice $\mathbf{\Phi}$), popřípadě jejich derivací, slouží funkce

```
getbasismatrix(evalarg, basisobj, nderiv=0)
```

parametry:

`evalarg`: Vektor bodů, ve kterých mají být spočítány hodnoty báze.

`basisobj`: Objekt třídy `basisfd`.

`nderiv`: Celé číslo určující stupeň derivace.

Nejjednodušší cestou, jak vytvořit z vektoru hodnot funkcionální pozorování, je použít funkci `data2fd`. Tato funkce je sice velmi jednoduchá (parametry jsou pouze vstupní data, definiční obor a objekt třídy `basisfd`), ale optimalizace se provádí pouze na základě SSE (viz (2.3)) a nelze použít penalizaci. Proto nebývají výsledky příliš kvalitní. Další funkcí, jejímž výstupem je funkcionální datový objekt, je

```
smooth.basis(argvals, y, fdParobj, wtvec=rep(1, length(argvals)),
```

`fdnames=NULL)`

parametry:

`argvals:` Definiční obor diskrétních pozorování.

`argvals:` Pole obsahující diskrétní pozorování. V případě více replikací se předpokládá, že počet sloupců je roven počtu replikací a počet řádků odpovídá počtu pozorování v jedné replikaci.

`wtvec:` Vektor obsahující váhy pro hodnoty `y`.

`fdnames:` Umožňuje zadání názvů výsledného objektu.

`fdParobj:` Pokud je třídy `basisfd`, potom se funkce `smooth.basis` shoduje s `data2fd`. To znamená, že parametr λ je nulový a nedochází k penalizaci. Další možnou třídou tohoto parametru je `fdPar`, potom je možné aplikovat optimalizaci pomocí penalizovaného součtu čtverců PSSE (viz 2.5).

Výstupem této funkce je `list` obsahující funkcionální datový objekt (`fd`), počet stupňů volnosti (`df`), hodnotu GCV (`gcv`), vektor (popř. matici) koeficientů $\boldsymbol{C}$ (viz model 2.2), `SSE`, penalizační matici $\boldsymbol{R}$ (`penmat`) a matici $\boldsymbol{G}$ (`y2cMap`). Třída tohoto objektu je `fdSmooth`.

Objekt třídy `fdPar` je výstupem funkce

`fdPar(fdobj=NULL, Lfdobj=NULL, lambda=0, penmat=NULL)`,

parametry:

`fdobj:` Funkcionální datový objekt. Jednou z možností je objekt třídy `basisfd` (další možnosti jsou popsány v nápovědě k této funkci).

`Lfdobj:` Lineární diferenciální operátor, který určuje penalizační kritérium. Může to být objekt třídy `Lfd` nebo celé číslo určující řád derivace.

`lambda:` Nezáporné číslo určující míru vyhlazení.

`penmat:` Tento parametr slouží k uložení tvaru penalizační matice a tím umožňuje ušetřit čas výpočtu při opakovaném přepočítávání.

Aby bylo možné určit hodnotu parametru lambda pomocí zobecněné cross-validace, je potřeba funkce

`lambda2gcv(log10lambda, argvals, y, fdParobj)`,

parametry:

`log10lambda:` $\log_{10}(\lambda)$.

`argvals:` Definiční obor `y`.

`y`: Diskrétní pozorování.
`fdParobj`: Objekt třídy `fdPar`.

Výstupem je hodnota zobecněné cross-validační funkce (viz (2.10)).

Nakonec uvedeme funkce, která umožňuje z funkcionálního datového objektu zkonstruovat konkrétní hodnoty výsledného odhadu (popř. jeho derivací) v daných bodech.

```
predict(object, newdata=NULL, Lfdobj=NULL, ...),
eval.fd(evalarg, fdobj, Lfdobj=0)
```

parametry:

`object`: Objekt třídy `fdPar` nebo `fdSmooth`.
`newdata`: Vektor hodnot, ve kterých chceme počítat hodnotu funkcionálního datového objektu
`fdobj`: Objekt třídy `fd`.
`Lfdobj`: Nezáporné celé číslo, které určuje stupeň derivování nebo objekt třídy `Lfd`.

4.2 Charakteristiky funkcionálních dat

Nyní uvedeme funkce, které počítají základní charakteristiky z kapitoly (3.1). V dalším textu budeme předpokládat, že `fdobj` je výstupem funkce `smooth.basis`.

Průměrová funkce:

```
mean.fd(fdobj).
```

Standardní odchylka:

```
sd.fd(fdobj).
```

Kovarianční funkce a cross-kovarianční funkce:

```
var.fd(fdobj1,fdobj2=fdobj1).
```

U poslední funkce záleží na vstupních datech. Pokud je vstupem pouze jedna funkcionální veličina, výstupem je kovarianční funkce. Pokud jsou však vstupem dvě funkcionální veličiny, je výstupem cross-kovarianční funkce. Korelační a cross-korelační funkce

se dopočítá pomocí vzorců z kapitoly 3.1 (viz (3.4) a (3.6)).

4.3 Concurrent model

Nyní se budeme zabývat počítáním concurrent modelu pomocí knihovny `fda`. Slouží k tomu funkce, které si nyní uvedeme. Jak napovídá název knihovny, obecně slouží k výpočtu regresních funkcionálních modelů, ale zde se budeme zabývat pouze variantou počítající již zmíněný concurrent model (viz (3.12)).

```
fRegress(y, xfdlist, betalist, ...)
```

parametry:

`y`: Závislá proměnná. Může být ve formátu `fd` nebo `fdPar`.

`xfdlist`: Objekt třídy `list` obsahující nezávislé proměnné (včetně konstatního členu). Prvkem tohoto seznamu může být buď vektor konstant (v případě, že je nezávislá proměnná skalární), nebo objekt třídy `fd`. V obou případech se počet replikací musí shodovat s počtem replikací nezávislé proměnné `y`.

`betalist`: Jedná se opět o seznam. Jeho délka musí být shodná s délkou `xfdlist`. Prvky jsou objekty třídy `fdPar`, které definují jakým způsobem budou odhadovány regresní funkce.

Výstupem této funkce je seznam obsahující `y`, `xfdlist`, `betalist`, funkcionální datový objekt pro odhady parametru β (`betaestlist`), predikované hodnoty (`yhatfdobj`), inverzní matici $\boldsymbol{C}^{-1}$, `y2cMap` (viz dříve), odhad standardní odchylky odhadu parametru β (`betastderrlist`), kovarianční matici `bvar` a matici $\boldsymbol{F}$ (`c2bMap`).

Následující funkce počítá hodnotu LMSSE (viz 3.15).

```
fRegress.CV(y, xfdlist, betalist, ...)
```

parametry:

`y`: Viz `fRegress()`.
`xfdlist`: Viz `fRegress()`.
`betalist`: Viz `fRegress()`.

Výstupem je hodnota minimalizačního kritéria (3.15) (`SSE.CV`) a cross-validační chyba (`errfd.cv`).

Poslední funkcí, kterou se budeme zabývat je

```
fRegress.stderr(y, xfdlist, betalist, ...)
```

parametry:

`y`: Viz `fRegress()`.

`y2cMap`: Matice, která transformuje závislou proměnnou $\boldsymbol{y}$ do vektoru (popř. matice) koeficientů $\boldsymbol{C}$. Tato matice je výstupem funkce `smooth.basis`.

`SigmaE`: Matice (popřípadě dvourozměrný funkcionální objekt), která odhaduje kovarianční strukturu reziduí modelu.

Výstupem je `list` následujících tří objektů. Prvním je `betastderrlist`. Tento objekt obsahuje funkcionální objekty odhadující standarní odchylky regresních funkcí. Jejich počet odpovídá počtu nezávislých proměnných. Další je `bvar` symetrická výběrová kovarianční matice regresních koeficientů. Poslední je matice `c2bMap`, která umožňuje transformaci koeficientů použitých pro vyhlazení závislé proměnné na regresní koeficienty.

Existuje mnoho dalších funkcí, které umožňují například pohodlné zobrazení výsledků nebo podrobnější analýzu funkcionálních dat. Cílem této kapitoly však nebylo uvést přehled všech funkcí, ale pouze těch, které jsou klíčové pro aplikaci výše uvedených postupů.

Kapitola 5

Shrnutí

Podařilo se nám teoreticky popsat metody pro získání funkcionálních dat na základě diskrétních pozorování. Na simulovaných datech jsme ukázali výhody a nevýhody jednotlivých přístupů. Dále jsme zformulovali model pro funkcionální data, který modeluje závislost funkcionálních náhodných veličin dynamicky v čase. Tento model je citlivý na rozptyl proměnných a také na velikost parametru λ_c. Model není příliš citlivý na velikost parametru λ, který určuje míru vyhlazení závislé proměnné. Na simulovaných datech bylo vidět, že i pro nevhodný parametr λ dává model dobré výsledky a že jeho predikční schopnost je lepší než predikční schopnost modelu (2.2).

Nakonec jsme tento model aplikovali na reálná data. Struktura modelu byla úmyslně volena jednoduše, aby výsledky byly jasně interpretovatelné a ověřitelné.

Literatura

[1] de Boor, C.: *A Practical Guide to Splines*. Springer, 2001.

[2] Craven, P.; Wahba, G.: Smoothing Noisy Data with Spline Functions: Estimating the Correct Degree of Smoothing by the Method of Generalized Cross-validation. *Numerische Mathematik*, ročník 31, 1998: s. 377–403.

[3] Diggle, P. J.; Hutchinson, M. F.: On Spline Smoothing with Autocerrelated Errors. *Australian Journal of Statistics*, ročník 31, 1989: s. 166–182.

[4] Efron, B.; Tibshirani, R. J.: *An Introduction to the Bootstrap*. CRC Press LLC, 1998.

[5] Hooker, G.; Graves, S.; Ramsay, J. O.: *Functional Data Analysis with Matlab and R*. Springer, 2009.

[6] Laub, A. J.: *Matrix Analysis for Scientists and Engineers*. SIAM, 2005.

[7] Mao, W.; Zhao, L. H.: Free-knot Polynomial Splines with Confidence Intervals. *J. R. Statist. Soc. B*, ročník 65, 2003: s. 901–919.

[8] Nychka, D.: Bayesian Confidence Intervals for Smoothing Splines. *JASA*, ročník 83, 1988: s. 1134–1143.

[9] Ramsay, J. O.: MATLAB, R and S-PLUS Functions for Functional Data Analysis, Říjen 2005.

[10] Ramsay, J. O.; Hooker, G.; Graves, S.: *Functional data analysis with R and MATLAB*. Springer, 2009, 202 s.

[11] Ramsay, J. O.; Silverman, B. W.: *Functional Data Analysis*. Springer, 1997.

[12] Royston, P.: Algorithm AS 181: The W test for Normality. *Applied Statistics*, ročník 31, 1982: s. 176–180.

[13] Wahba, G.: Bayesian 'Confidence Intervals' for Cross-validated Smoothing Spline. *JRSS B*, ročník 45, 1983: s. 133–150.