Moduł 12
Przypadek użycia 2
Analiza reakcji biochemicznych

iBigWorld:
Innovations for Big Data in a Real World

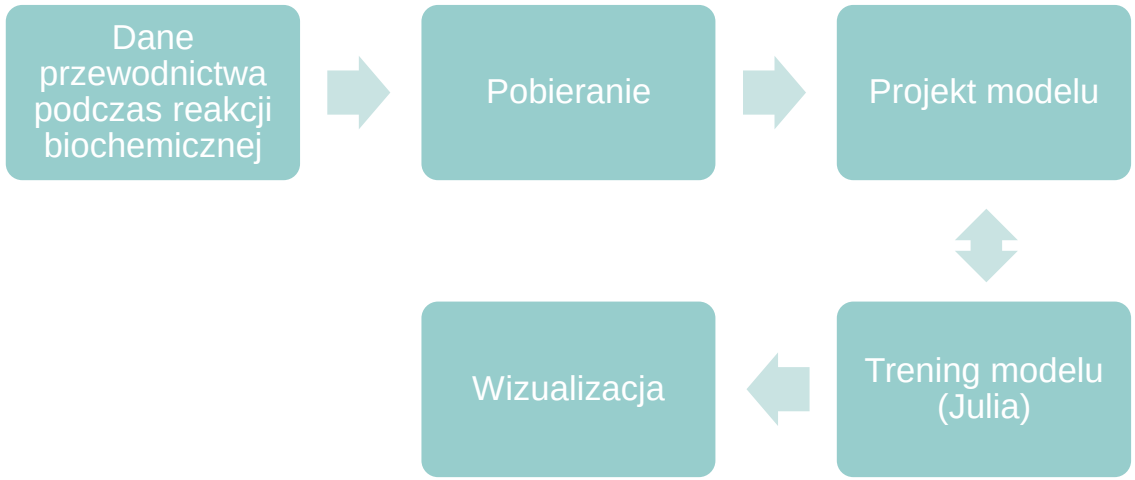
Zespół UBB

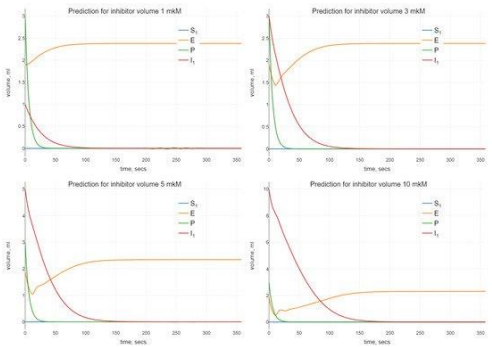
iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Identyfikacja parametrów reakcji biochemicznych

- Opracowano i przeanalizowano numerycznie i eksperymentalnie modele dynamiczne kompleksów enzym-substrat i enzym-substrat-inhibitor.
- Konstruujemy wielosubstratowy model wieloinhibitorowy.
- Opracowano algorytm identyfikacji parametrów, który został przetestowany na danych eksperymentalnych przewodnictwa roztworu i jest zgodny z Big Data.
- W Julii zaimplementowano obliczenia o wysokiej wydajności.





iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Pobieranie danych

- *Interakcja enzym-substrat*
- Badania przeprowadzono w roztworach wodnych z użyciem następujących substancji: białko – BSA; enzym – enzym acetylocholinoesteraza (AChE); oraz substrat — chlorek acetylocholiny (AChCl). Do kompleksu utworzonego z BSA 2 mg/ml (objętość 4 ml) i enzymem 2 mg/ml (objętość 0,1 ml) (kompleks jest znany jako usieciowany agregat enzymatyczny (CLEA)), substrat 2 mg /ml w różnych objętościach (0,1 ml, 0,3 ml, 0,9 ml, 1,5 ml, itd.). W efekcie otrzymano próbki:
  - Próbka 1. BSA + enzyme + 0.1 mL substratu;
  - Próbka 2. BSA + enzyme + 0.3 mL substratu;
  - Próbka 3. BSA + enzyme + 0.9 mL substratu;
  - Próbka 4. BSA + enzyme + 1.5 mL substratu;
  - itd...
- Przewodność kompleksu BSA + enzym + substrat badano za pomocą specjalnie skonstruowanego układu pomiarowego, w czasie 500 s z próbkowaniem sygnału co 5 s.

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Pobranie danych

- df<-read.xlsx(„conductivity.xls",sheetName = "Sheet1",stringsAsFactors=F)
- df <- df[which(df$time_sec>=250),]
- tmpvec <- df$substrate_0dot1
- df$substrate_0dot1 <- df$substrate_0dot1 - c(rep(tmpvec[1],length(tmpvec)))
- tmpvec <- df$substrate_0dot3
- df$substrate_0dot3 <- df$substrate_0dot3 - c(rep(tmpvec
- tmpvec <- df$substrate_0dot9
- df$substrate_0dot9 <- df$substrate_0dot9 - c(rep(tmpvec
- tmpvec <- df$substrate1dot5
- df$substrate1dot5 <- df$substrate1dot5 - c(rep(tmpvec[1

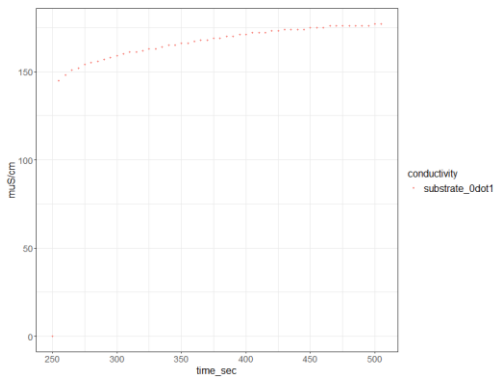
time_sec	substrate_0dot1	substrate_0dot3	substrate_0dot9	substrate1dot5
250	0	0.0	0.0	0.0
255	145	628.6	279.2	1291.4
260	148	383.6	719.2	1287.4
265	151	385.6	739.2	1301.4
270	152	387.6	753.2	1305.4
275	154	388.6	762.2	1308.4
280	155	389.6	770.2	1311.4
285	156	390.6	778.2	1314.4
290	157	391.6	784.2	1319.4
295	158	392.6	790.2	1324.4
300	159	393.6	795.2	1331.4
305	160	394.6	801.2	1337.4
310	161	395.6	805.2	1345.4
315	161	396.6	809.2	1351.4
320	162	397.6	813.2	1356.4
325	163	398.6	817.2	1362.4
330	163	399.6	820.2	1367.4
335	164	400.6	823.2	1373.4
340	165	400.6	825.2	1377.4
345	165	401.6	828.2	1381.4
350	166	402.6	830.2	1385.4
355	166	403.6	832.2	1388.4
360	167	404.6	834.2	1391.4

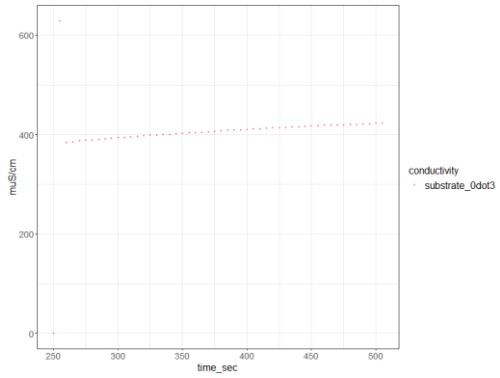
iBigWorld:2020-1-PL01-KA203-082197

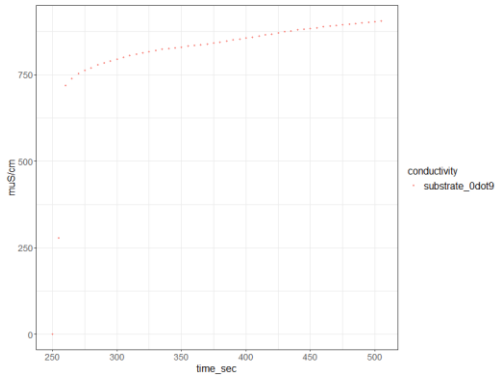
Co-funded by the Erasmus+ Programme of the European Union

Eksploracja danych

- tmp=melt(df,id.vars=c("time_sec"),variable.name="conductivity",value.name="muS/cm")
- png(paste(paste("substrate_0dot1"),"png", sep = "."), width = 800, height = 600)
- p<-ggplot(data=tmp[which(tmp$conductivity=="substrate_0dot1"),],aes(x=time_sec,y=`muS/cm`,color=conductivity))+geom_point(size=1)+my_theme
- plot(p)
- dev.off()
- png(paste(paste("substrate_0dot3"),"png", sep = "."), width = 800, height = 600)
- p<-ggplot(data=tmp[which(tmp$conductivity=="substrate_0dot3"),],aes(x=time_sec,y=`muS/cm`,color=conductivity))+geom_point(size=1)+my_theme
- plot(p)
- dev.off()
- png(paste(paste("substrate_0dot9"),"png", sep = "."), width = 800, height = 600)
- p<-ggplot(data=tmp[which(tmp$conductivity=="substrate_0dot9"),],aes(x=time_sec,y=`muS/cm`,color=conductivity))+geom_point(size=1)+my_theme
- plot(p)
- dev.off()
- png(paste(paste("substrate_1dot5"),"png", sep = "."), width = 800, height = 600)
- p<-ggplot(data=tmp[which(tmp$conductivity=="substrate_1dot5"),],aes(x=time_sec,y=`muS/cm`,color=conductivity))+geom_point(size=1)+my_theme
- plot(p)
- dev.off()







iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Przetwarzanie wysokowydajne w Julia

```
library(JuliaCall)
library(diffeqr)
de <- diffeqr::diffeq_setup()
f <- JuliaCall::julia_eval("function f(du, u, h, p, t)
  k_d=p[1]
  a=p[2]
  m=p[3]
  tau_m=p[4]
  n_S_0=p[5]
  n_E_0=p[6]
  n_P_0=p[7]
  n=500
  rmse=sqrt((m+1)/a^2)
  lags=range(1/n, length=n, stop=(tau_m+(m+1)/a+3*rmse)*(1-1/n))
  du[1]= -k_d*u[2]*u[1]
  du[2]=-k_d*u[2]*u[1] + k_d*(tau_m+(m+1)/a+3*rmse)*(1/n)*sum([psi(a,m,tau_m,tau)*h(p,t-tau)[2]*h(p,t-tau)[1] for tau in lags])
  du[3]=k_d*(tau_m+(m+1)/a+3*rmse)*(1/n)*sum([psi(a,m,tau_m,tau)*h(p,t-tau)[2]*h(p,t-tau)[1] for tau in lags])
end")
h <- JuliaCall::julia_eval("function h(p, t)
  [p[5],p[6],p[7]]
end")
psi <- JuliaCall::julia_eval("function psi(a, m, tau_m, tau)
  if tau <= tau_m
    res = 0
  else
    res = (a^(m+1)/gamma(m+1))*(tau-tau_m)^m * exp(-a*(tau-tau_m))
  end
  res
end")
```

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Przetwarzanie wysokowydajne w Julia (kontynuacja)

- u0 <- c(0.3,1,0)
- tspan <- c(max(0,min(df$time_sec)),max(0,max(df$time_sec)))
- abstol <- 1e-2
- reltol <- 1e-2
- saveat <- round(seq(from=min(df$time),to=max(df$time),length.out=500))
- saveat <- unique(saveat)
- p <- c(0.05,1,20,5,u0[1:3])
- constant_lags <- c(100.0)
- JuliaCall::julia_assign("u0", u0)
- JuliaCall::julia_assign("tspan", tspan)
- JuliaCall::julia_assign("constant_lags", constant_lags)
- JuliaCall::julia_assign("p", p)
- prob <- JuliaCall::julia_eval("using SpecialFunctions; DDEProblem(f, u0, h, tspan, p)")
- system.time(sol <- de$solve(prob,abstol=abstol,reltol=reltol, saveat=saveat, de$MethodOfSteps(de$Tsit5())))
- udf <- as.data.frame(cbind(sol$t,t(sapply(sol$u,identity))))
- colnames(udf) <- c("t","S","E","P")

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Strojenie modelu

- Implementacja algorytmu COBYLA do problemu

Input data: $X_{exp}, \Pi_{lower}, \Pi_{upper}, \Pi_{init}$
Result: Π_{opt}

form the initial simplex C_{init} with the vertices $\Pi_0^{init}, \Pi_1^{init}, \ldots, \Pi_6^{init}$;
repeat
  for the current simplex C calculate the values $\hat{\Phi}_C(\Pi_i), i = \overline{0,6}$;
  search the vertex Π_p determined by the equation $\hat{\Phi}_C(\Pi_p) = \min\{\hat{\Phi}_C(\Pi_i), i = \overline{0,6}\}$;
  calculate new vertex as $\Pi_{new} := -\theta\Pi_p + (1+\theta)\frac{1}{6}\sum_{i=\overline{0,6}, i\neq p}\Pi_i$, where reflection coefficient $\theta \in (0,1)$ being chosen as small as possible in order $\hat{\Phi}_C(\Pi_p)$ not were the least calculated function value so far;
  form modified simplex C_{new} replacing vertex Π_p with Π_{new};
  search Π_{opt} as a solution of the problem of linear optimization

$$\left.\begin{array}{ll} \texttt{minimize} & \hat{\Phi}_{C_{new}}(\Pi), \\ \texttt{subject to} & \Pi \in C_{new} \end{array}\right\} \tag{21}$$

until *stop condition*;
return Π_{opt};

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

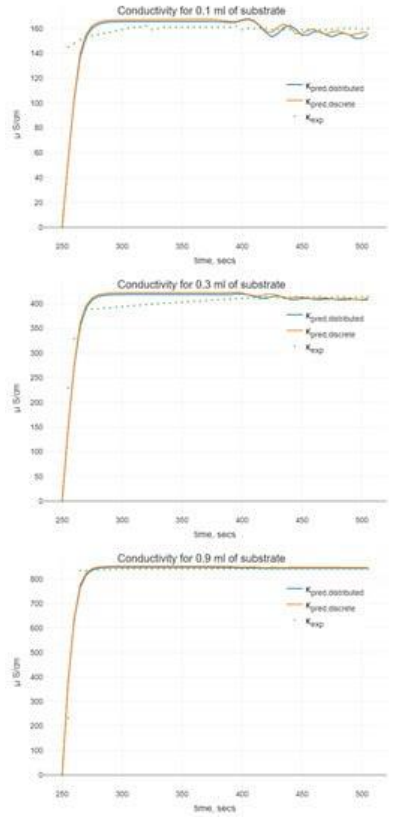
Strojenie modelu

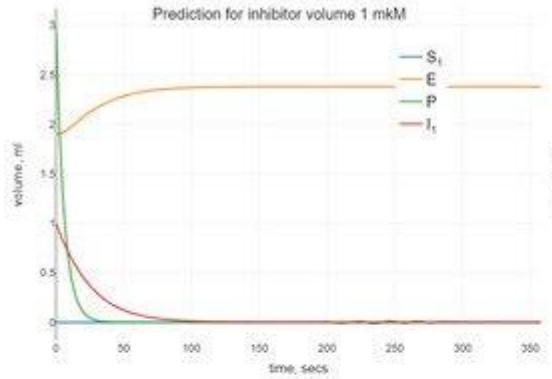
- Implementacja algorytmu COBYLA do problemu

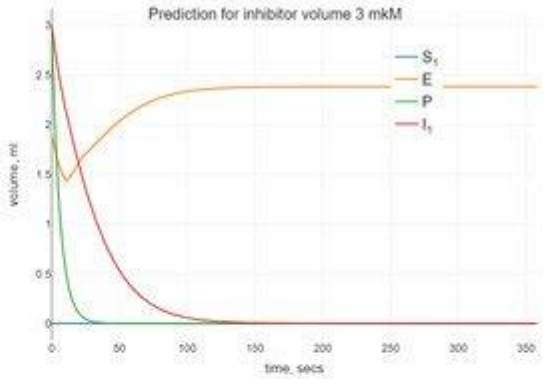
```
# M. J. D. Powell, ``A direct search optimization method that models the objective and constraint functions by linear interpolation,'' in Advances in Optimization # and Numerical Analysis, eds. S. Gomez and J.-P. Hennart
library(nloptr)
  fn.sir <- function(x) {
    sqrt(sum(ssq(x)^2))
  }
  hin.sir <- function(x) {
    h <- numeric(61)
    h[1:29] <- upper - x
    h[30:59] <- x - lower
    h[60] <- x[14]-x[15]-x[16]-x[17]-x[18]-x[19]-x[20]-x[21]
    h[61] <- -x[14]+x[15]+x[16]+x[17]+x[18]+x[19]+x[20]+x[21]
    #h[2] <- -x[14]+x[15]+x[16]+x[17]+x[18]+x[19]+x[20]+x[21]
    return(h)
  }
  fitval.cobyla <- cobyla(parms, fn.sir, hin = hin.sir,
                          nl.info = TRUE, control = list(xtol_rel = 1e-3,maxeval = 100))

  parest <- fitval.cobyla$par

  fitval.cobyla$value
```

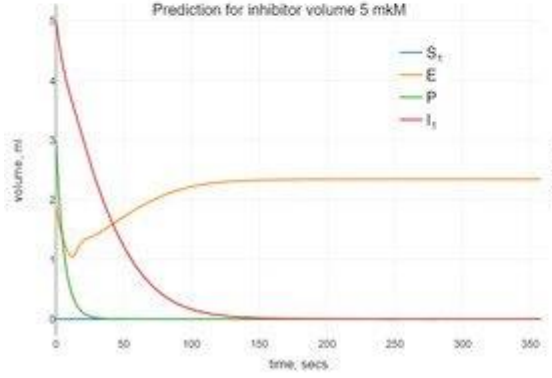

Co-funded by the Erasmus+ Programme of the European Union

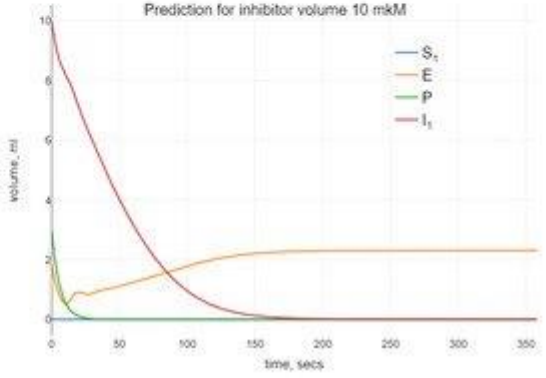
Wizualizacja









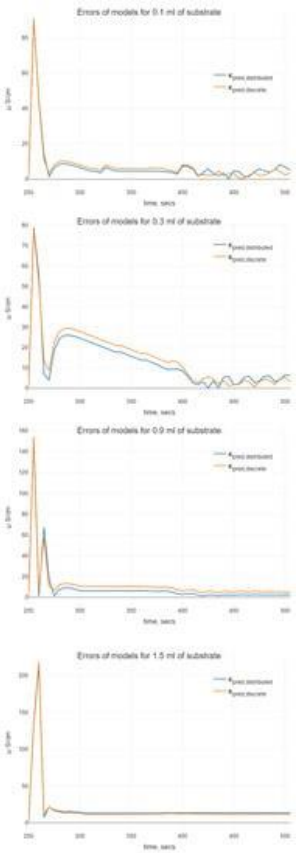


iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the
Erasmus+ Programme
of the European Union

Wizualizacja

```
library(plotly)
fig <- plotly::plot_ly(udf, x = sol$t, y = ~S, name="S",type =
'scatter', mode = 'lines') %>% plotly::add_trace(y = ~E,
name="E") %>% plotly::add_trace(y = ~P,name="P") %>%
  layout(
    title = list(text="Prediction for substrate volume 0.1
ml",y=0.95),
    xaxis = list(title = "time, secs"),
    yaxis = list(title = "volume, ml"),
    font = font
  )
orca(fig, file = paste(paste("Prediction_0dot1"),"png", sep =
"."))
```



iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the
Erasmus+ Programme
of the European Union