

# Analysis of eGFR trajectories from Hong Kong Diabetes Registry

Guozhi Jiang

April 22, 2015

## Contents

<b>1</b>	<b>Description of data</b>	<b>2</b>
1.1	Data overview . . . . .	2
1.2	Outcomes . . . . .	5
<b>2</b>	<b>Analysis</b>	<b>10</b>

# 1 Description of data

## 1.1 Data overview

The data were comprised by two parts: the baseline data and the follow-up eGFR data. The baseline data, which were extracted from the Hong Kong Diabetes Registry(HKDR), included the information of clinical assessments and laboratory investigations at enrollment, and the well-defined complication outcomes censored to 31st, January 2009. As here we focused on the ESRD outcome, we only selected those Chinese patients with no history of ESRD which was defined according to the ICD-9 codes and eGFR <15. Therefore, we obtained a cohort consisted of 718 ESRD events and 8810 event-free patients at censoring date. The follow-up data included all the creatinine records from enrollment to 2014, and the eGFR were corresponding calculated using the Chinese-modified MDRD formula.

```
> base_dat <- read.table("../data/ESRD1_Prospect2014_CH-T2D_1218vs8336.csv",header=TRUE,sep=",");
> base_dat <- transform(base_dat, doin=cal.yr(date), dob=cal.yr(DOB, "%d/%m/%Y"),
+                        dox=cal.yr(ESRD1_DATE));
> base_dat <- transform(base_dat, dodm = pmin(YEAR_DIA + runif(length(YEAR_DIA)), doin));
> dob_na <- which(is.na(base_dat$dob));
> base_dat$dob[dob_na] <- (floor(base_dat$doin[dob_na]) - base_dat$AGE[dob_na]) + runif(length(dob_na));
> #write.table(base_dat, "trans_base.csv", sep=",", row.names=F, col.names=T);
> base_subdat <- subset(base_dat, select=c("Obs_id", "doin", "dob", "dodm", "dox",
+                                         "SEX", "ESRD1_END"))
> dim(base_subdat);

[1] 9554      7

> names(base_subdat);

[1] "Obs_id"      "doin"        "dob"         "dodm"        "dox"         "SEX"
[7] "ESRD1_END"

> head(base_subdat);

  Obs_id   doin    dob    dodm    dox SEX ESRD1_END
1      1 2002.805 1940.598 2002.493 2014.493  0         0
2      2 1996.757 1939.194 1995.153 2014.493  1         0
3      3 1996.585 1935.172 1983.217 2014.493  1         0
4      4 2001.203 1927.048 1980.747 2004.648  1         0
5      5 1997.381 1924.527 1993.908 2014.493  1         0
6      6 1999.221 1922.345 1991.221 2010.564  0         0

> table(base_subdat$ESRD1_END);

 0     1
8336 1218
```

The outcomes of interest were named as "ESRD1\_HIST" (0 represents no ESRD history), "ESRD1\_END"(endpoint censored to 2009) and "ESRD1\_TIME"(follow-up period).

```
> follow_dat <- read.table("../data/eGFR_19940714_20140630.csv", header=TRUE, sep=",");
> follow_dat <- transform(follow_dat, dolab=cal.yr(test_date));
> follow_dat <- subset(follow_dat, select=-c(test_date));
> dim(follow_dat);
```

```
[1] 391551      4
```

```
> head(follow_dat);
```

	Obs_id	F_eGFR0	creatinine	dolab
1	1	80.6474	84	2002.632
2	1	97.9131	71	2002.643
3	1	73.5245	91	2002.663
4	1	91.8750	75	2002.767
5	1	72.5693	92	2002.805
6	1	81.4311	83	2003.914

We merged the baseline data and the follow-up eGFR data according to the id of subject:

```
> merged_dat <- merge(base_subdat, follow_dat, by=intersect("Obs_id", "Obs_id"), sort=F);
> dim(merged_dat);
```

```
[1] 366122      10
```

```
> head(merged_dat);
```

	Obs_id	doin	dob	dodm	dox	SEX	ESRD1_END	F_eGFR0	creatinine
1	1	2002.805	1940.598	2002.493	2014.493	0	0	102.8468	66
2	1	2002.805	1940.598	2002.493	2014.493	0	0	91.8750	75
3	1	2002.805	1940.598	2002.493	2014.493	0	0	96.8863	71
4	1	2002.805	1940.598	2002.493	2014.493	0	0	80.6474	84
5	1	2002.805	1940.598	2002.493	2014.493	0	0	87.7935	77
6	1	2002.805	1940.598	2002.493	2014.493	0	0	103.2903	66

	dolab
1	2014.359
2	2002.767
3	2005.951
4	2002.632
5	2007.558
6	2012.812

```
> addmargins(table(table(merged_dat$Obs_id)));
```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
350	65	69	74	94	141	182	179	158	174	160	150	155	172	158	176	
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
172	222	214	200	220	160	174	196	178	164	177	159	159	164	153	119	
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	
130	129	132	125	112	105	117	126	123	118	113	99	88	122	76	93	
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	

93	98	73	75	69	62	54	56	68	59	62	55	54	48	43	34
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
32	38	36	43	38	36	27	34	26	29	40	30	28	28	30	21
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
25	22	22	23	25	29	23	23	25	24	19	16	21	16	17	14
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112
10	18	22	12	14	10	12	17	11	13	16	13	12	10	8	12
113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128
9	8	13	8	8	10	11	12	10	6	7	4	5	5	4	5
129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144
6	10	6	8	2	9	9	1	9	5	6	3	2	2	4	3
145	146	147	148	149	150	151	153	154	155	156	158	159	160	161	162
1	2	7	4	7	7	4	2	5	4	2	7	4	2	2	1
163	164	165	166	167	169	170	171	172	173	175	176	177	178	179	181
1	5	2	1	4	2	3	2	1	6	3	4	3	4	4	1
182	183	186	187	191	193	194	195	196	197	198	199	200	204	206	207
2	2	1	3	2	1	1	3	1	2	1	2	2	1	1	1
208	209	210	211	212	213	216	218	221	223	224	225	228	231	232	238
1	1	1	1	1	1	2	1	1	1	1	1	2	1	1	1
240	241	243	249	250	251	252	254	258	267	282	284	288	293	301	308
1	1	1	1	1	2	2	1	1	3	1	1	1	1	1	1
319	353	431	725	Sum											
1	1	1	1	1	9554										

We only selected those records between baseline and event/censoring dates,that said, those eGFR records before baseline or after event/censoring dates were removed. Moreover, we also calculated the follow-up age, duration of diabetes, and the backward time gap between event/censoring date and measurement date of eGFR, named "F\_AGE","F\_DMAGE" and "BW\_TIME", respectively.

```
> sub_merged <- subset(merged_dat, (1:nrow(merged_dat)) %in% intersect(which(dolab >= doin), which(dolab <= doex)))
> sub_merged <- transform(sub_merged, F_AGE=dolab-dob, F_DMAGE=dolab-dodm, BW_TIME=dolab-dox);
> dim(sub_merged);
```

```
[1] 282607      13
```

```
> head(sub_merged);
```

	Obs_id	doin	dob	dodm	dox	SEX	ESRD1_END	F_eGFR0	creatinine
1	1	2002.805	1940.598	2002.493	2014.493	0	0	102.8468	66
3	1	2002.805	1940.598	2002.493	2014.493	0	0	96.8863	71
5	1	2002.805	1940.598	2002.493	2014.493	0	0	87.7935	77
6	1	2002.805	1940.598	2002.493	2014.493	0	0	103.2903	66
7	1	2002.805	1940.598	2002.493	2014.493	0	0	81.2613	83
8	1	2002.805	1940.598	2002.493	2014.493	0	0	81.4311	83
	dolab	F_AGE	F_DMAGE	BW_TIME					
1	2014.359	73.76044	11.865461	-0.1341547					
3	2005.951	65.35250	3.457521	-8.5420945					
5	2007.558	66.95962	5.064640	-6.9349760					
6	2012.812	72.21355	10.318575	-1.6810404					

```

7 2004.568 63.96988 2.074906 -9.9247091
8 2003.914 63.31554 1.420560 -10.5790554

> str(sub_merged);

'data.frame':      282607 obs. of  13 variables:
 $ Obs_id      : int  1 1 1 1 1 1 1 1 1 1 ...
 $ doin        : num  2003 2003 2003 2003 2003 ...
 $ dob         : num  1941 1941 1941 1941 1941 ...
 $ dodm        : num  2002 2002 2002 2002 2002 ...
 $ dox         : num  2014 2014 2014 2014 2014 ...
 $ SEX         : int  0 0 0 0 0 0 0 0 0 0 ...
 $ ESRD1_END   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ F_eGFR0     : num  102.8 96.9 87.8 103.3 81.3 ...
 $ creatinine  : num  66 71 77 66 83 83 92 74 73 69 ...
 $ dolab       : num  2014 2006 2008 2013 2005 ...
 $ F_AGE       : num  73.8 65.4 67 72.2 64 ...
 $ F_DMAGE     : num  11.87 3.46 5.06 10.32 2.07 ...
 $ BW_TIME     : num  -0.134 -8.542 -6.935 -1.681 -9.925 ...

> range(sub_merged$BW_TIME);

[1] -21.03491  0.00000

> #write.table(sub_merged, "effective_subdat.csv", sep="," , row.names=F, col.names=T)

```

## 1.2 Outcomes

We removed those records with missing data. We first plotted the observed creatinine and eGFR values (Figure 1). From the figure, we can see there are some abnormal records, which may be due to measurement or typo errors.

```

> nomiss_dat <- sub_merged[complete.cases(sub_merged), ];
> dim(nomiss_dat);

[1] 279002      13

> with(nomiss_dat, table((F_eGFR0>300) + (F_eGFR0>1000)));

      0      1      2
278701  265   36

> with(nomiss_dat, plot(dolab, F_eGFR0, pch=16, cex=0.3,
+                        xlab="Date of measurement", ylab="eGFR"));

> with(nomiss_dat, plot(dolab, creatinine, pch=16, cex=0.3,
+                        xlab="Date of measurement", ylab="Creatinine"));

```

We removed those records with eGFR  $\geq 300$ , which were considered to be errors. The updated distributions were shown in Figure 2.

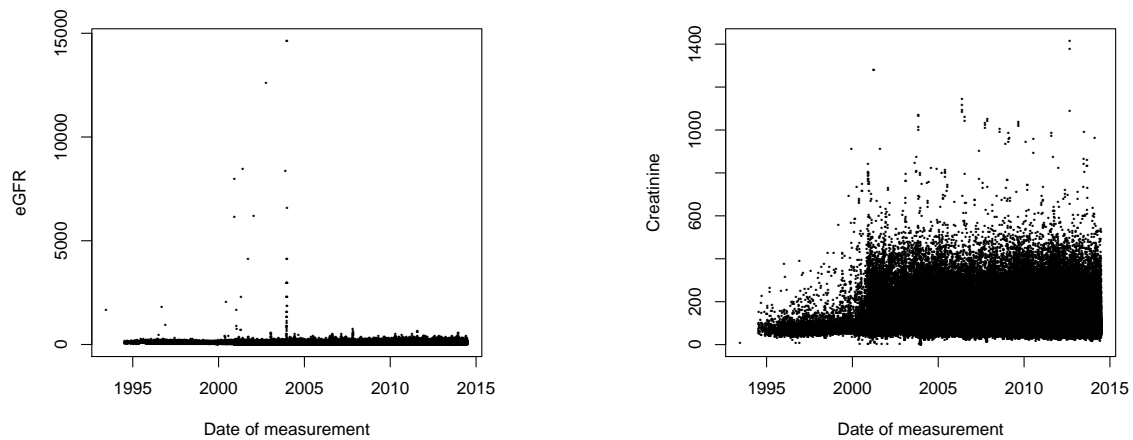


Figure 1: *Distribution of the raw data of eGFR and creatinine.*

```
> sub_nomiss <- subset(nomiss_dat, F_eGFR0<300);
> dim(sub_nomiss);

[1] 278701    13

> with(sub_nomiss[sample(1:nrow(sub_nomiss), 2000),],
+       plot(BW_TIME, F_eGFR0, pch=16, cex=0.3,
+           xlab="Time before ESRD", ylab="eGFR"));
> abline(h=15, col="red");

> with(sub_nomiss[sample(1:nrow(sub_nomiss), 2000),],
+       plot(BW_TIME, creatinine, pch=16, cex=0.3, #ylim=c(0,400),
+           xlab="Time before ESRD", ylab="Creatinine"))
```

As we here only focused on the patients developed ESRD, we extracted those subjects and plotted the distribution of eGFR. We then removed those patients with only one measurement, and summarized the number of measurement for each subject (Figure 3).

```
> event_dat <- subset(sub_nomiss, ESRD1_END==1);
> dim(event_dat);

[1] 41444    13

> length(unique(event_dat$Obs_id));

[1] 1199

> num_test <- table(event_dat$Obs_id);
> id_keep <- names(which(num_test>1));
> event_dat <- subset(event_dat, Obs_id %in% id_keep);
> (num_stat <- table(table(event_dat$Obs_id)));
```

#removed those patients with only one measurement

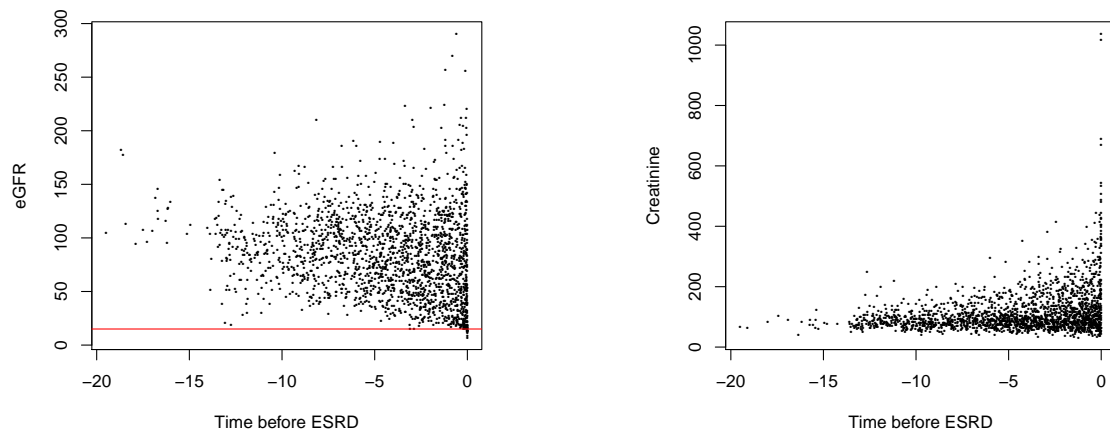


Figure 2: *Distribution of eGFR and creatinine with 2000 samples after removing  $eGFR \geq 300$ . The red line represents  $eGFR=15$ .*

```

2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21
51 21 23 18 17 19 23 18 21 21 22 17 21 23 20 18 18 23 14 16
22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41
23 17 20 18 15 17 17 32 17 13 11 24 18 13 23 14 20 24 15 16
42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61
14 16 10 14 11 11 10 10 8 6 11 10 7 8 7 7 7 7 7 8
62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81
10 11 4 8 10 5 5 8 4 2 4 2 3 3 4 8 2 3 3 1
82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 98 99 100 102 103
3 4 5 5 1 3 3 2 1 4 4 2 3 1 2 1 2 1 1 1
104 105 106 108 109 111 112 113 114 115 117 120 122 125 126 128 129 131 132 133
4 1 2 1 1 2 1 2 1 3 4 1 1 1 1 1 1 1 1 1
136 142 146 147 148 153 173
1 1 1 1 1 1 1

```

```

> dim(event_dat);

[1] 41412    13

> length(unique(event_dat$Obs_id));

[1] 1167

> event_dat <- transform(event_dat, log_eGFR=log(event_dat$F_eGFR0));

> par(mar=c(5,4,1,2));
> plot(as.numeric(names(num_stat)),num_stat,type="h",lwd=3,xaxs="i",xlim=c(0,140),
+       xlab="No. of measurements",ylab="No. of subjects",yaxt="n");
> axis(side=2);

```

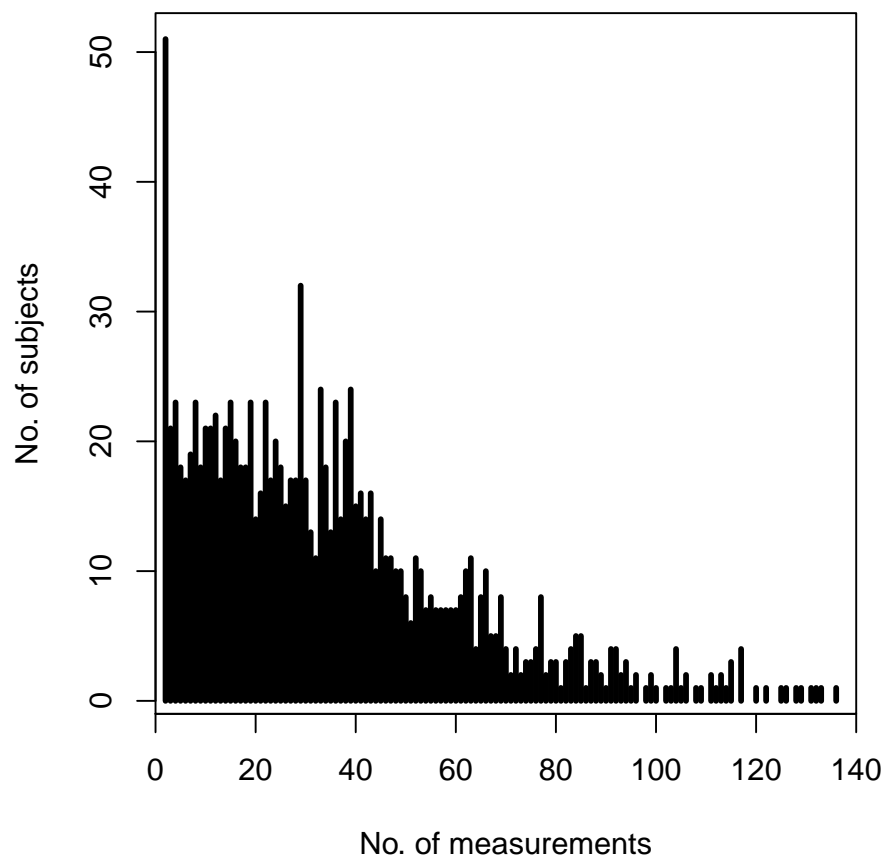


Figure 3: *Number of eGFR records for patients.*

```

> dest_eGFR <- density(event_dat$F_eGFR0);
> plot(dest_eGFR, xlim=c(0,150), xlab="eGFR", lwd=3, yaxs="i",
+       ylab="Density", main="ESRD subjects", bty="n");
> abline(v=quantile(event_dat$F_eGFR0, probs=c(50, 75, 90)/100),
+        col="red");

> dest_logeGFR <- density(log(event_dat$F_eGFR0));
> plot(dest_logeGFR, lwd=3, xlab="Ln(eGFR)",
+       ylab="Density", main="ESRD subjects");

```

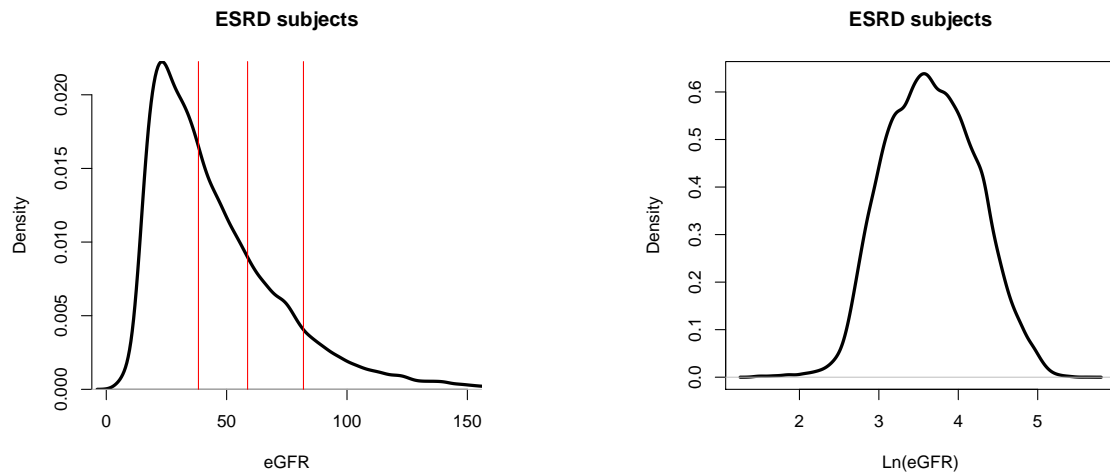


Figure 4: *Distribution of eGFR and  $\log(eGFR)$  for ESRD subjects. Red lines represents the 50, 75 and 90 percent of subjects, respectively*

As shown in Figure 4, the original distribution of eGFR is skewed, whilst it's close to be normal after log-transformed.

## 2 Analysis

We first fit the model using the "lcm" function, and specified "linear" as the link function. The "lcm" with linear link function is similar with the "hlme" function, but the reason for using "lcm" function is that the confidence interval of predict values can be obtained by using the corresponding prediction function.

```
> (kn <- seq(-12, 0, , 5));

[1] -12 -9 -6 -3 0

> MM <- Ns(event_dat$BW_TIME, knots=kn);
> dim(MM);

[1] 41412 4

> MM <- detrend(MM, event_dat$BW_TIME);
> dim(MM);

[1] 41412 3

> head(MM);

      1      2      3
[1,] 0.04689555 0.38709327 -0.10183513
[2,] -0.02867914 0.38150137 -0.07236472
[3,] -0.06988576 0.30174461 -0.04175845
[4,] -0.05881167 -0.13637325 0.03674734
[5,] -0.07504976 0.02845691 0.01176823
[6,] -0.34770860 -0.36093378 0.33529565

> (colnames(MM) <- paste("x", colnames(MM), sep=""));

[1] "x1" "x2" "x3"

> event_dat <- cbind(event_dat, MM);
> dim(event_dat);

[1] 41412 17

> event_model <- hlme(log_eGFR ~ BW_TIME + x1 + x2 + x3 + F_AGE + SEX + F_DMAGE,
+                        mixture =~ BW_TIME + x1 + x2 + x3,
+                        random =~ BW_TIME,
+                        subject = "Obs_id", ng=4, data=event_dat);

Be patient, hlme is running ...
The program took 10768.32 seconds

> event_model;
```

Heterogenous linear mixed model  
fitted by maximum likelihood method

```
hlme(fixed = log_eGFR ~ BW_TIME + x1 + x2 + x3 + F_AGE + SEX +  
      F_DMAGE, mixture = ~BW_TIME + x1 + x2 + x3, random = ~BW_TIME,  
      subject = "Obs_id", ng = 4, data = event_dat)
```

Statistical Model:  
Dataset: event\_dat  
Number of subjects: 1167  
Number of observations: 41412  
Number of latent classes: 4  
Number of parameters: 30

Iteration process:  
Convergence criteria satisfied  
Number of iterations: 24  
Convergence criteria: parameters= 9e-08  
                      : likelihood= 3.5e-06  
                      : second derivatives= 1.4e-12

Goodness-of-fit statistics:  
maximum log-likelihood: -4265.84  
AIC: 8591.69  
BIC: 8743.55

Now we can further investigate the posterior probabilities of subjects in each class (Figure 5 and 6).

```
> clr<- c("black","red","blue", "yellow");  
> postprob(event_model);
```

Posterior classification:

	class1	class2	class3	class4
N	273.00	304.00	436.00	154.0
%	23.39	26.05	37.36	13.2

Posterior classification table:  
--> mean of posterior probabilities in each class

	prob1	prob2	prob3	prob4
class1	0.8572	0.1229	0.0195	0.0004
class2	0.0972	0.7891	0.1110	0.0027
class3	0.0420	0.1330	0.7514	0.0736
class4	0.0044	0.0194	0.1151	0.8612

Posterior probabilities above a threshold (%):

	class1	class2	class3	class4
prob>0.7	77.29	65.79	58.03	79.22
prob>0.8	67.77	52.30	45.18	71.43
prob>0.9	59.71	40.46	35.55	59.74

```

> str(event_model$pprob);

'data.frame':      1167 obs. of  6 variables:
 $ Obs_id: num  16 21 22 36 37 42 44 54 56 71 ...
 $ class : int   3 1 4 2 2 1 1 3 3 3 ...
 $ prob1 : num  3.12e-03 7.50e-01 4.97e-09 4.06e-01 3.54e-05 ...
 $ prob2 : num  2.39e-01 2.50e-01 1.81e-06 5.94e-01 9.15e-01 ...
 $ prob3 : num  7.55e-01 8.15e-10 4.89e-03 2.42e-04 8.47e-02 ...
 $ prob4 : num  2.58e-03 5.46e-45 9.95e-01 1.04e-15 1.78e-11 ...

> plot_ppr

function(fit_model,clr)
{
  par(mfrow=c(1,3), mar=c(3,0,1,1), oma=c(0,3,0,0),
      las=1, bty="n", mgp=c(3,1,0)/1.6);
  ng <- fit_model$ng;
  ppr <- fit_model$pprob;
  num <- table(ppr$class);
  for(i in 1:ng) {
    hist(ppr[ppr$class==i,i+2],breaks=0:50/50,
         col=clr[i], border=clr[i], ylim=c(0,60),
         main="", xlab="", yaxt="n", yaxs="i");
    if(i==1) axis(side=2);
    text(0.4,30,paste("Class",i," n=",num[i]),
         font=2,col=clr[i], cex=1);
  }
}

> plot_ppr(event_model,clrs);

> ng <- event_model$ng;
> post_pr <- event_model$pprob;
> #par(bty="o");
> pairs(post_pr[, 2+1:ng], pch=16, col=clrs[post_pr$class],
+       cex=0.4, gap=0);

```

We built a data set for prediction to plot the estimated trajectories. Here the median age 65, median DM duration 12 and sex as male were used. The function "plot\_predictY" incorporated the prediction function "predictY" in LCMM package (Figure 7).

```

> (adjust_var <- event_model$Xnames[-c(1:5)]);

[1] "F_AGE" "SEX" "F_DMAGE"

> length(table(event_dat$BW_TIME))

[1] 6964

> x <- sort(unique(event_dat$BW_TIME));
> length(x);

```

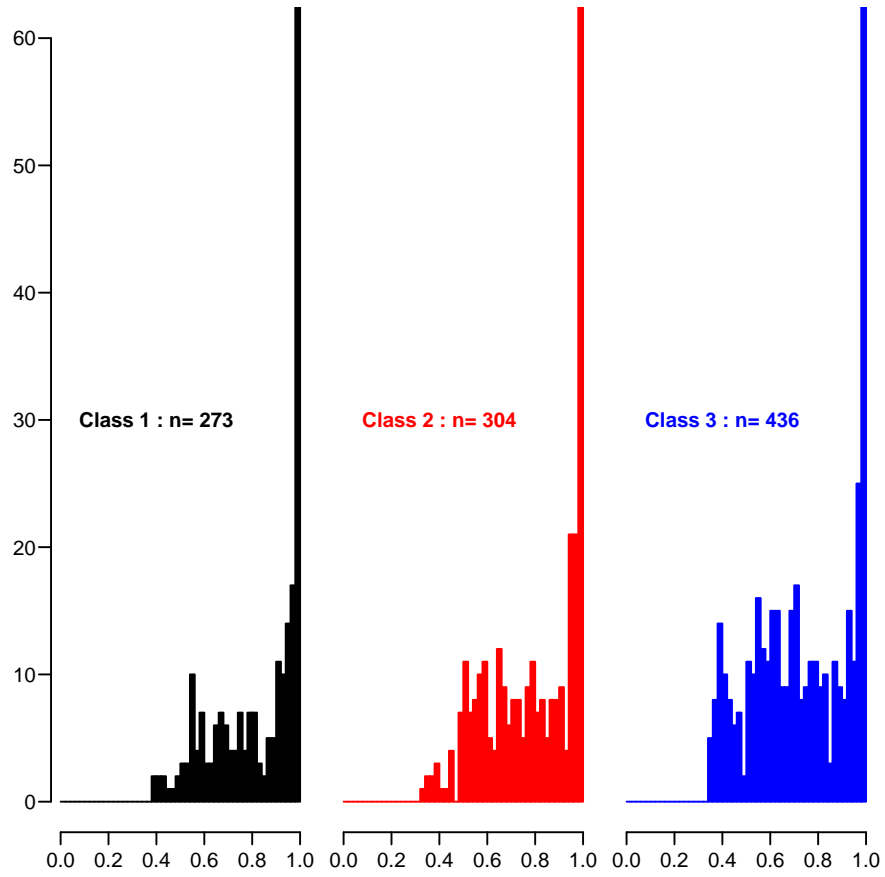


Figure 5: *Posterior probabilities of 3 classes for the ESRD model.*

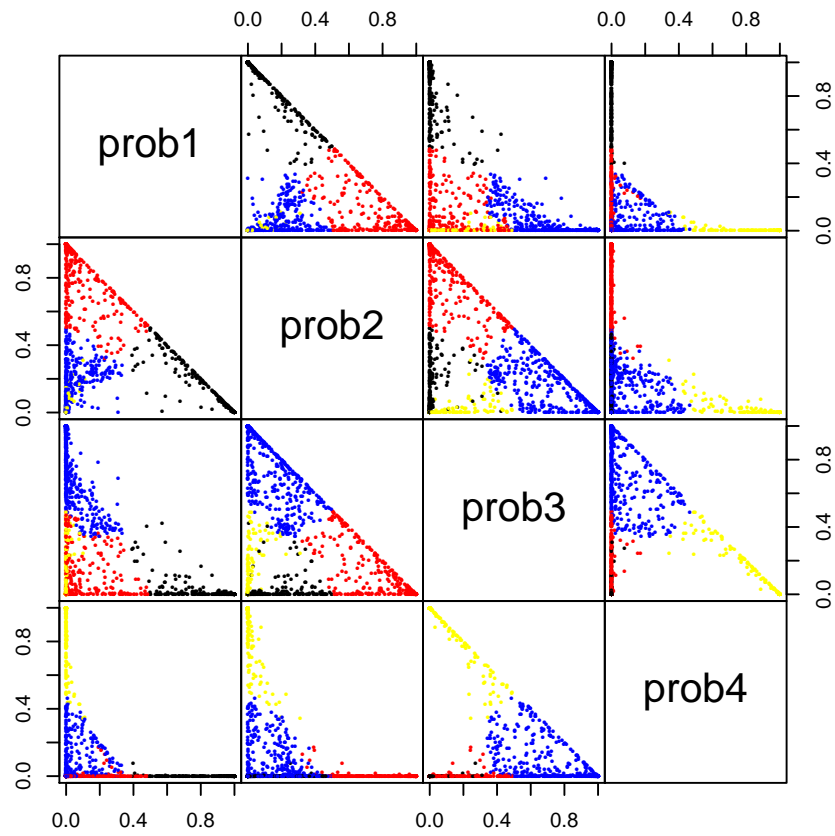


Figure 6: *Pairwise posterior probabilities from the fitted model using "lcm".*

```
[1] 6964
```

```
> wh <- match(x, event_dat$BW_TIME)[1:69*100];  
> length(wh);
```

```
[1] 69
```

```
> plotdata <- data.frame(1, event_dat$BW_TIME[wh], MM[wh,],  
+                         F_AGE = 65 + event_dat$BW_TIME[wh],  
+                         SEX = 1,  
+                         F_DMAGE = 15 + event_dat$BW_TIME[wh]);  
> names(plotdata) <- event_model$Xnames;  
> head(plotdata);
```

	intercept	BW_TIME	x1	x2	x3	F_AGE	SEX
1	1	-14.22040	-0.6919963	-0.001011389	-0.621725804	50.77960	1
2	1	-12.62149	-0.6178121	-0.138170718	-0.250384157	52.37851	1
3	1	-11.96715	-0.5874523	-0.194302593	-0.098414192	53.03285	1
4	1	-11.56468	-0.5682702	-0.228569177	-0.005717254	53.43532	1
5	1	-11.20602	-0.5490492	-0.258027568	0.073654734	53.79398	1
6	1	-10.87201	-0.5277820	-0.283754898	0.142452077	54.12799	1

	F_DMAGE
1	0.779603
2	2.378508
3	3.032854
4	3.435318
5	3.793977
6	4.127995

```
> range(plotdata$BW_TIME);
```

```
[1] -14.22039699 -0.08761123
```

```
> pred_event <- exp(predictY(event_model, plotdata, var.time="BW_TIME", draws=TRUE)$pred);  
> ylim <- range(pred_event);  
> lwd_main <- 4;  
> lwd_ci <- 1;  
> for (i in 1:ng)  
+ {  
+   plot(y = pred_event[, i], x = plotdata$BW_TIME, type = "l", col= clr[i],  
+       ylim = ylim, lwd = lwd_main, xlab = "Time before events(years)", ylab = "eGFR");  
+   points(y = pred_event[, i + ng], x = plotdata$BW_TIME, type = "l", lty = "dashed", col = clr[i],  
+         ylim = ylim, lwd = lwd_ci);  
+   points(y = pred_event[, i + 2*ng], x = plotdata$BW_TIME, type = "l", lty = "dashed", col = clr[i],  
+         ylim = ylim, lwd = lwd_ci);  
+   par(new = TRUE);  
+ }  
> abline(h = 15, lty = "dashed");  
> legend("topright", legend = paste("class", 1:ng), col=clr, lty=1, lwd=lwd_main, cex=0.8);  
> par(new = FALSE);
```

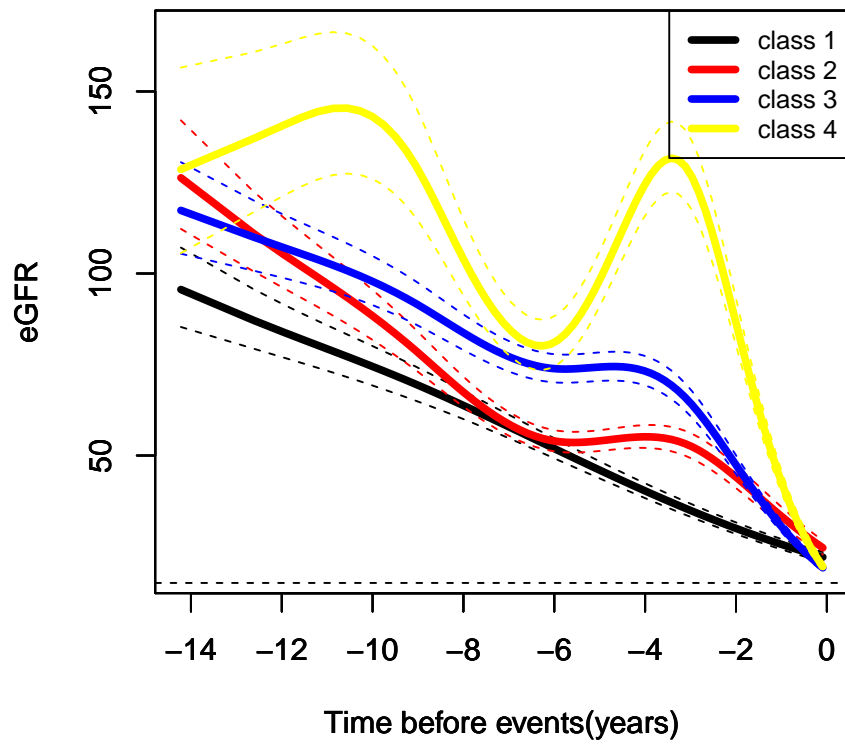


Figure 7: Mean trajectories of eGFR for the three latent classes of subjects developing ESRD. The horizontal dashed line represents eGFR is equal to 15