# Mazurka paper figures

DJM

8/20/2018

Suggested order

1. Parameter interpretation in Fliere
2. Using parameters to examine two different performances
3. Clustering performances (compare the clusters)
   a. what can we say about the parameters of each cluster? what is different about them?
4. Similar performances (Rubinstein)
5. Model issues

Short tempo

```
ggplot(tempos, aes(x=note_onset, y=Rubinstein_1961)) +
  geom_line() + ylab('tempo (bpm)') + xlab('measure') +
  scale_x_continuous(breaks=1:4*2, limits = c(1,9)) +
  theme_minimal(base_family = 'Times') +
  geom_hline(yintercept = 132, linetype='dashed')
```
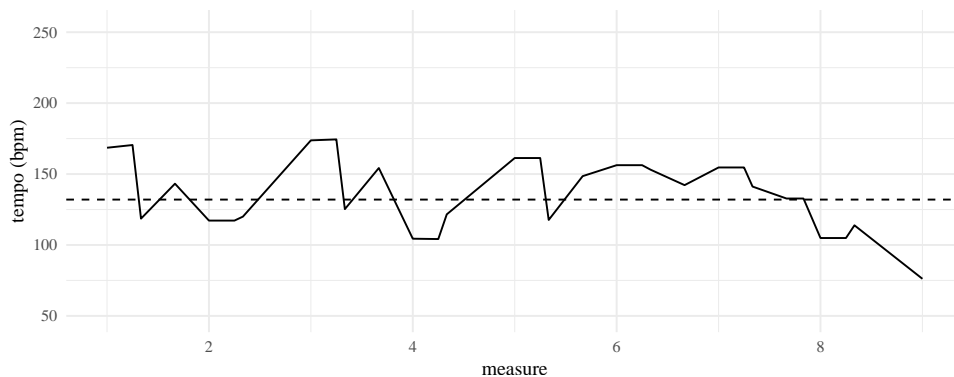

Comparing clusters

```
perfs = tempos[,-c(1:3)] %>% as.matrix %>% t
bad_perf = grep('Block',rownames(pvec_ml))
hc_parm = pvec_ml[-bad_perf,] %>% dist %>% percentize %>% hclust
hc_perf = perfs[-bad_perf,] %>% dist %>% percentize %>% hclust
short_labs = rownames(perfs)[-bad_perf]
lens = nchar(short_labs)
short_labs = paste0(substr(short_labs,1,4), substr(short_labs,lens-3,lens))
hc_parm$labels = short_labs
hc_perf$labels = short_labs
dend_parm = hc_parm %>% as.dendrogram
dend_perf = hc_perf %>% as.dendrogram
```

```r
dend_parm = dend_parm %>% set('labels_col', value=fivecolors[1:4], k=4) %>%
  set('branches_lty', 1) %>%
  set('branches_k_color', value=fivecolors[1:4], k=4)
dend_perf = dend_perf %>% set('labels_col', value=fivecolors[1:4], k=4) %>%
  set('branches_lty', 1) %>%
  set('branches_k_color', value=fivecolors[1:4], k=4)
col_lines_by_left_groups <- fivecolors[cutree(dend_parm, 4, order_clusters_as_data=FALSE)]
```

```r
tanglegram(dend_parm,dend_perf, color_lines = col_lines_by_left_groups,
           columns_width = c(1,1,1), axes=FALSE, rank_branches = TRUE, type='t',
           # left_dendo_mar = c(0,1,0,8), right_dendo_mar = c(0,8,0,1),
           margin_top = 0,
           margin_bottom = 0, margin_inner = 3.5,
           #remove_nodePar = TRUE,
           lab.cex=.75, lwd=1, edge.lwd=1)
```
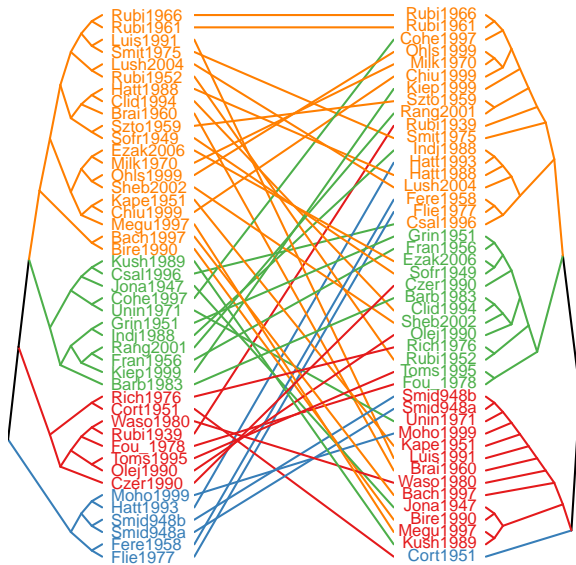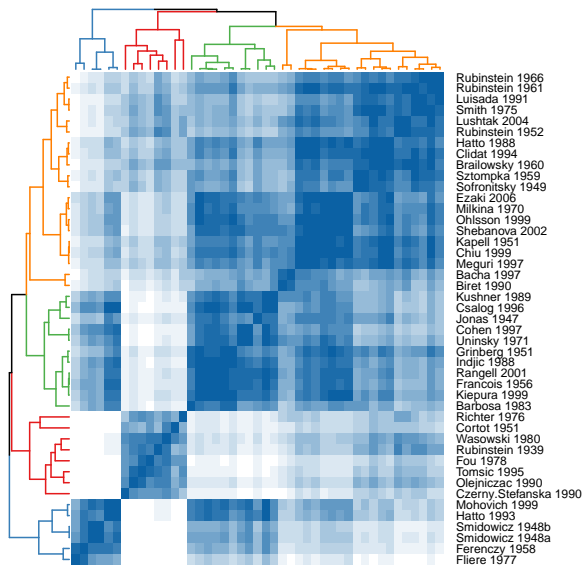


```r
heatmap.2(as.matrix(percentize(dist(pvec_ml[-bad_perf,]))),
          Rowv = dend_parm, Colv = dend_parm,
          symm=TRUE,
          density.info = 'none', trace='none',
          labRow = sub('_',' ',row.names(pvec_ml)[-bad_perf]),
          labCol = NA,
          key.title = NA,
          col=colorRampPalette(c('#0b61a4','white')),
          key.xlab = NA,
          margins = c(1,6),
          cexRow = .6,
          cexCol = .6,
          lhei=c(1,8),
          lwid=c(1,8),
          offsetCol = 0, offsetRow = 0,
          key=FALSE
)
```

Rubinstein 1966
Rubinstein 1961
Luisada 1991
Smith 1975
Lushtak 2004
Rubinstein 1952
Hatto 1988
Clidat 1994
Brailowsky 1960
Sztompka 1959
Sofronitsky 1949
Ezaki 2006
Milkina 1970
Ohlsson 1999
Shebanova 2002
Kapell 1951
Chiu 1999
Meguri 1997
Bacha 1997
Biret 1990
Kushner 1989
Csalog 1996
Jonas 1947
Cohen 1997
Uninsky 1971
Grinberg 1951
Indjic 1988
Rangell 2001
Francois 1956
Kiepura 1999
Barbosa 1983
Richter 1976
Cortot 1951
Wasowski 1980
Rubinstein 1939
Fou 1978
Tomsic 1995
Olejniczac 1990
Czerny.Stefanska 1990
Mohovich 1999
Hatto 1993
Smidowicz 1948b
Smidowicz 1948a
Ferenczy 1958
Fliere 1977

```r
clusts = cutree(as.hclust(dend_parm), k = 4)
save(clusts, file = '../extras/ClusterLabels.Rdata')
```

```r
heatmap.2(as.matrix(percentize(dist(perfs[-bad_perf,]))),
        Rowv = dend_perf, Colv = dend_perf,
        symm=TRUE,
        density.info = 'none', trace='none',
        labRow = sub('_',' ',row.names(pvec_ml)[-bad_perf]),
        labCol = NA,
        key.title = NA,
        col=colorRampPalette(c('#0b61a4','white')),
        key.xlab = NA,
        margins = c(1,6),
        cexRow = .6,
        cexCol = .6,
        lhei=c(1,8),
        lwid=c(1,8),
        offsetCol = 0, offsetRow = 0,
        key=FALSE
)
```
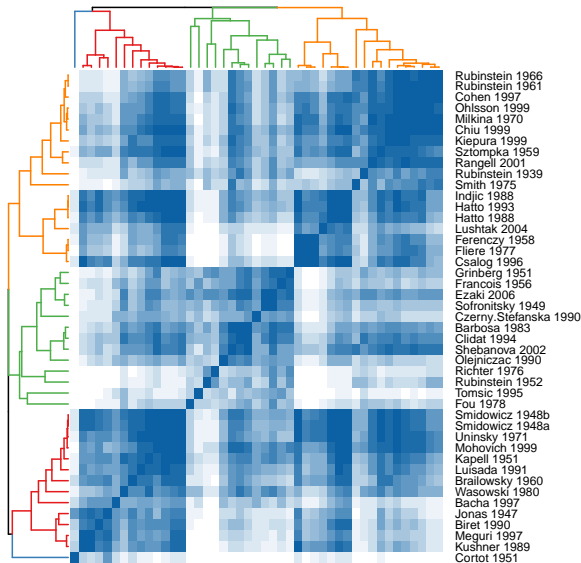
Interpreting parameters

```r
plotStates <- function(performers, pars, tempos,
                       noplot=FALSE,
                       particleNumber = 200,
                       initialMean = c(132,0),
                       initialVariance = c(400,10)){
  lt = diff(c(tempos$note_onset, 61))
  alldfs = NULL
  for(perf in performers){
    params = unlist(pars[row.names(pars)==perf,])
    y = matrix(tempos[[perf]], nrow = 1)
    mats = yupengMats(lt, params[1], params[2:4], params[5:8],
                      params[9:12], initialMean, initialVariance)
    bs = beamSearch(mats$a0, mats$P0, c(1,0,0,0,0,0,0,0), mats$dt,
                    mats$ct, mats$Tt, mats$Zt,
                    mats$Rt, mats$Qt, mats$GGt, y, mats$transMat, particleNumber)
    bestpath = bs$paths[which.max(bs$weights),]
    kal = kalman(mats, bestpath, y)
    df = data.frame(performer=perf, measure = tempos$note_onset, tempo = c(y),
                    inferred = c(kal$ests), state = convert8to4(bestpath))
    alldfs = rbind(alldfs, df)
  }
  if(noplot) return(alldfs)
  ggplot(alldfs, aes(x=measure, y=tempo)) + ylim(0, max(df$tempo)) +
    annotate('rect',xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf,
             alpha=.2) +
    theme_minimal(base_family = 'Times') +
    geom_line(aes(y=tempo), color='black')+
    geom_point(aes(y=inferred),color=fivecolors[alldfs$state])+
    facet_wrap(~performer, labeller = as_labeller(function(x) gsub('_',' ', x))) +
    theme(legend.position = 'none', legend.title = element_blank())+
    ylab('tempo (beats/minute)') + xlab('measure number') +
```
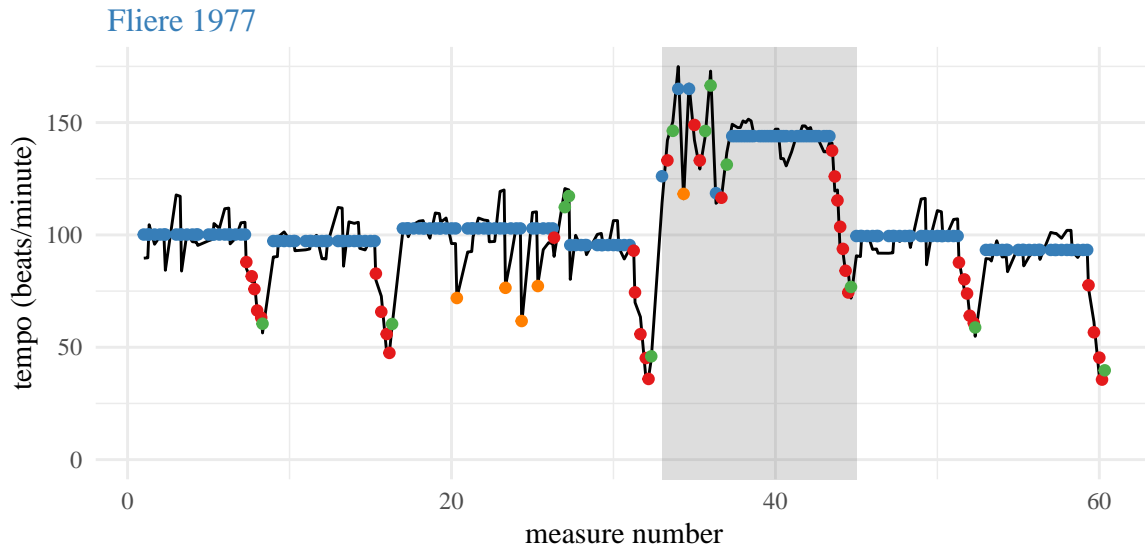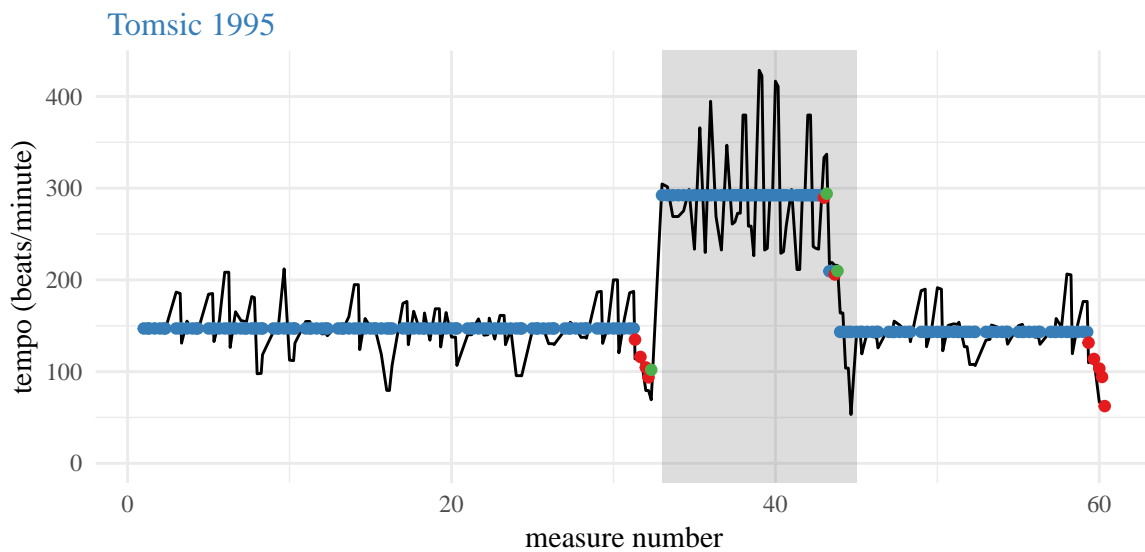
```
    theme(strip.text = element_text(size=12, hjust=0,color = 'black'))
        #strip.background = element_rect(fill='grey90',linetype = 'blank'))
}
```
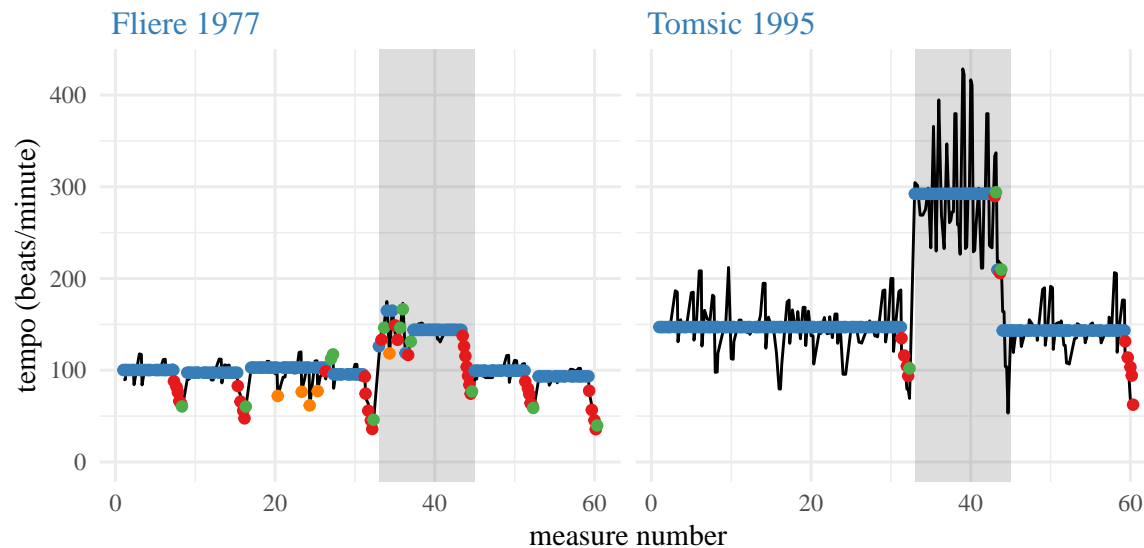
```
perfs = c('Fliere_1977','Tomsic_1995')
plotStates(perfs[1], pvec_ml, tempos)
```


Fliere 1977

```
plotStates(perfs[2], pvec_ml, tempos)
```


Tomsic 1995

```
plotStates(perfs, pvec_ml, tempos)
```

Different smoothing

Try splines, replicating knots, l1tf?

```
nsplines = 64 # 1 knot per bar plus boundary
B = bs(tempos$note_onset, df=nsplines, intercept = TRUE)
preds_smooth = fitted(lm(tempos[[perfs[1]]]~B-1))
single.knots = match(seq(4,56,by=4)+1,tempos$meas_num)
double.knots = match(c(16,24,32,44)+1, tempos$meas_num)
triple.knots = match(c(16,24,32,44)+1, tempos$meas_num)
quad.knots = match(c(16,24,32,44)+1, tempos$meas_num)
all.knots = tempos$note_onset[
  sort(c(single.knots,double.knots,triple.knots,quad.knots))]
B1 = bs(tempos$note_onset, knots = all.knots, intercept = TRUE,Boundary.knots = c(1,61))
preds_music = fitted(lm(tempos[[perfs[1]]]~B1-1))
extras=data.frame(x=tempos$note_onset,y1=preds_smooth,y2=preds_music)
perf1 = plotStates(perfs[1], pvec_ml, tempos, noplot = TRUE)
perf1$ss = preds_smooth
perf1$ms = preds_music
perf1 %>% select(measure, ss, ms, inferred) %>%
  gather(key='key',value='value',-measure) %>%
  ggplot(aes(x=measure)) + geom_line(aes(y=value, color=key)) +
  scale_color_manual(values = fivecolors[c(3,4,5)],
                     labels = c('model','splines','splines (informed knots)')) +
  geom_point(data=perf1, aes(x=measure, y=tempo), color=fivecolors[1]) +
  annotate('rect',xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf,
           alpha=.2) +
  theme_minimal(base_family = 'Times') +
  theme(legend.position = 'bottom',legend.title = element_blank())
```
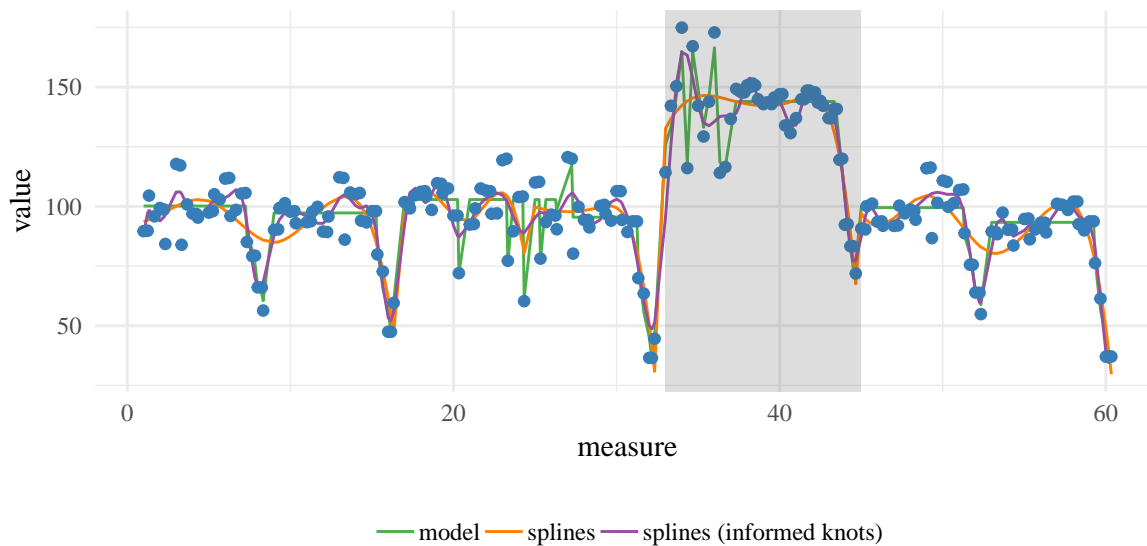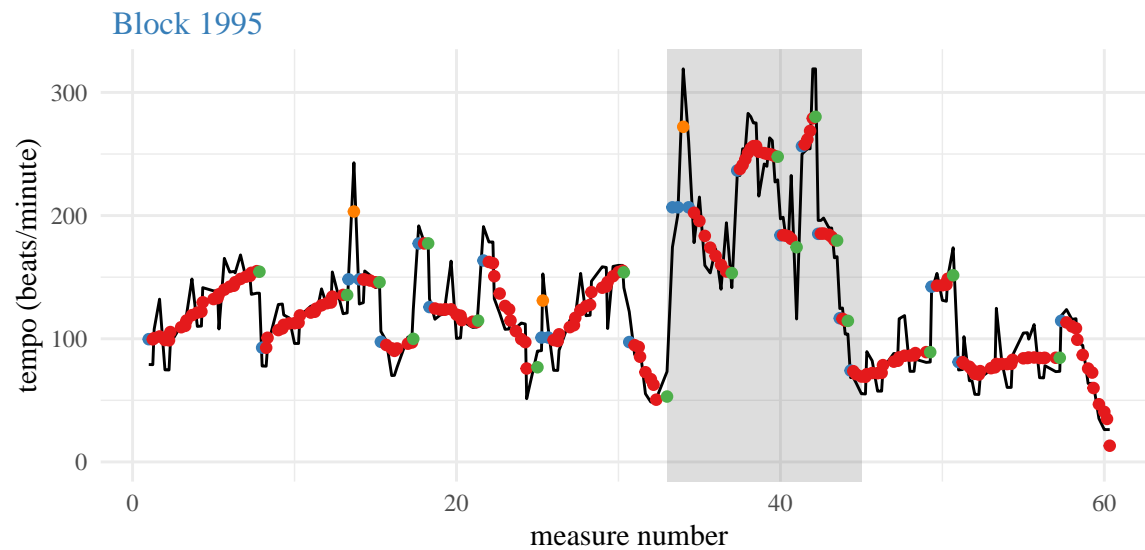
Similar performances

```r
rubins = row.names(pvec_ml)[grep('Rubinstein', row.names(pvec_ml))]
plotStates(rubins, pvec_ml, tempos)
```
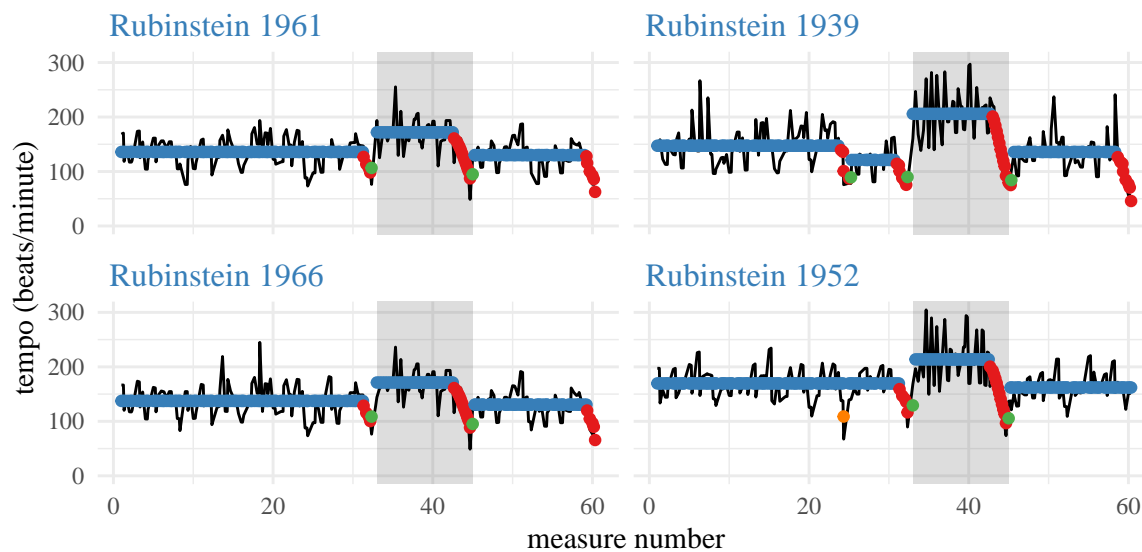


```r
# note that the 1939 recording is the only one in a different cluster
```

Bad estimation

```r
plotStates('Block_1995', pvec_ml, tempos)
```

Block 1995

Problems with the model

- Problem with retransitioning to state 1
- states 2 and 3 aren't constrained to always decrease/increase, only in mean
- state 4 may not always emphasize a slow down
- previous 2 have to do with Gaussian assumptions
- necessity for strong priors
- but priors are on parameters, not on path (how would we want this to change?)