Mazurka paper figures

DJM

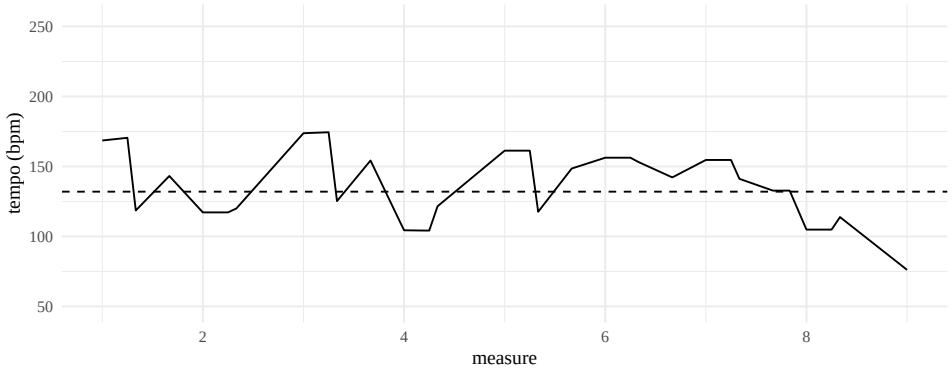
8/20/2018

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')
```

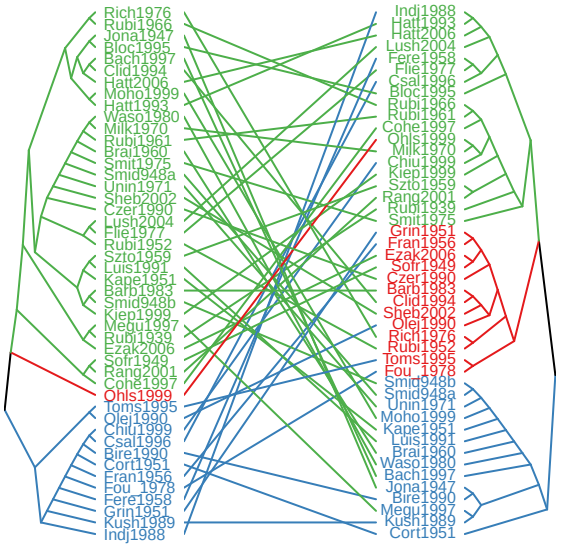


Comparing clusters

```
nclusts = 3
perfs = tempos[,-c(1:3)] %>% as.matrix %>% t
# bad_perf = grep('Block',rownames(pvec_ml))
hc_parm = pvec_ml %>% Dist %>% hclust
hc_perf = perfs %>% dist %>% percentize %>% hclust
short_labs = rownames(perfs)
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
```

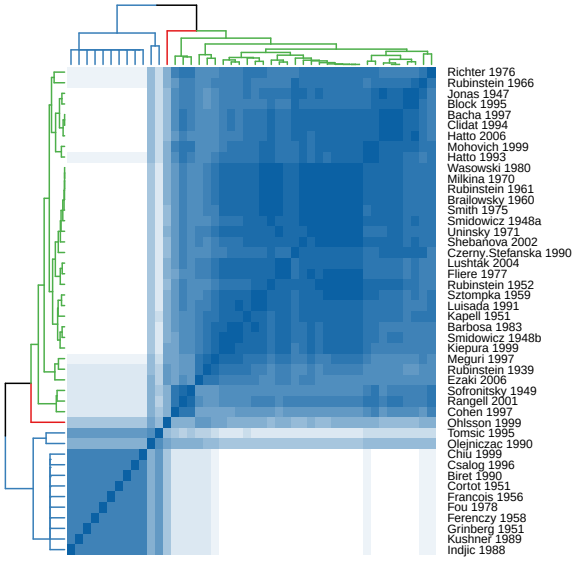
```
dend_parm = hc_parm %>% as.dendrogram
dend_perf = hc_perf %>% as.dendrogram

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

```
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)
```



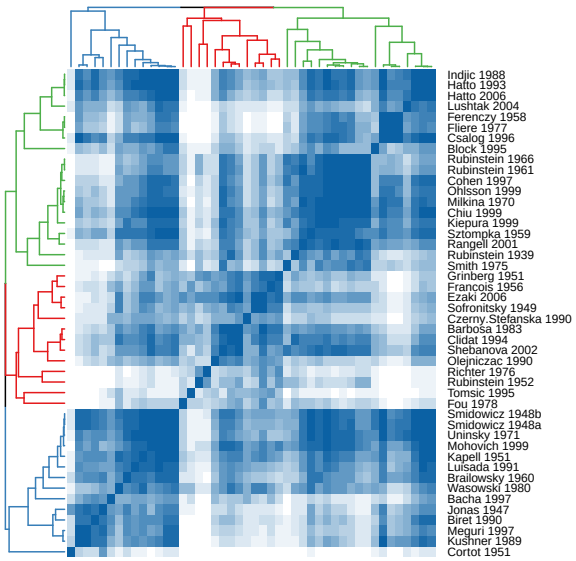

```
heatmap.2(as.matrix(Dist(pvec_ml)),
          Rowv = dend_parm, Colv = dend_parm,
          symm=TRUE,
          density.info = 'none', trace='none',
          labRow = sub('_',' ',row.names(pvec_ml)),
          labCol = NA,
          key.title = NA,
          col=colorRampPalette(c('#0b61a4','white')),
          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
```

```
)
```



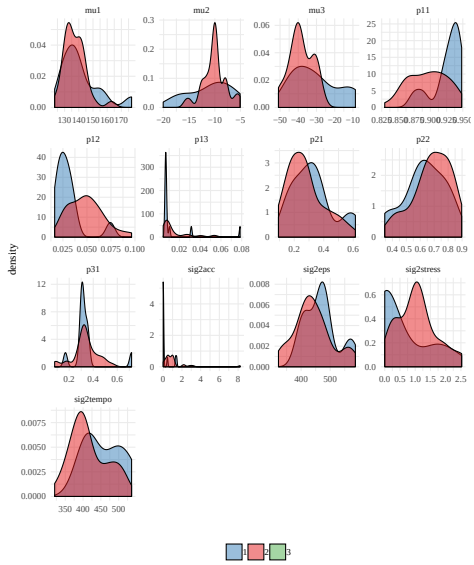

```
pvec_ml$clust = as.factor(cutree(as.hclust(dend_parm), k = nclusts))
```

```
heatmap.2(as.matrix(percentize(dist(perfs))),
          Rowv = dend_perf, Colv = dend_perf,
          symm=TRUE,
          density.info = 'none', trace='none',
          labRow = sub('_',' ',row.names(pvec_ml)),
          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
)
```



Cluster densities

```
pvec_ml %>% gather(key='parameter',value='value',-clust) %>%
  ggplot(aes(x=value,fill=clust)) + geom_density(alpha=.5) +
  facet_wrap(~parameter,scales='free') +
  scale_fill_manual(values=fivecolors[1:nclusts]) + xlab('') +
  theme(legend.title = element_blank(), legend.position = 'bottom')
```



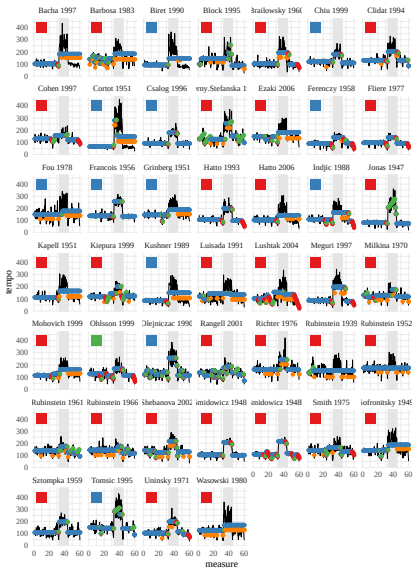
Interpreting parameters

```
convert10to4 <- function(path){
  t1 = c(1,2,4,2,3,1,3,1,3,1)
```

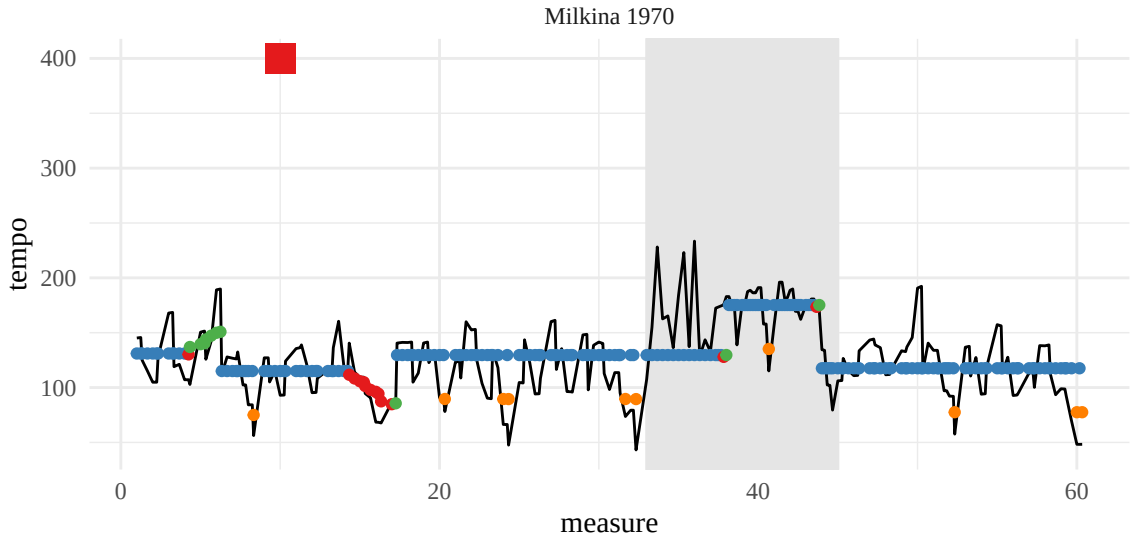
```
  path10 = t1[path+1]
  path10
}
```

Plotting performances

```
plots = vector("list", 4)
lt = diff(c(tempos$note_onset, 61))
for(i in 1:nrow(pvec_ml)){
  params = unlist(pvec_ml[i,])
  y = matrix(tempos[,i+3], nrow = 1)
  pmats = yupengMats(lt, params[1], params[2:4], params[5:7], params[8:13],
                     c(132,0), c(400,10))
  beam = beamSearch(pmats$a0, pmats$P0, c(1,0,0,0,0,0,0,0,0,0),
                     pmats$dt, pmats$ct, pmats$Tt, pmats$Zt,
                     pmats$HHt, pmats$GGt, y, pmats$transMat, 200)
  bestpath = beam$paths[which.max(beam$weights),]
  kal = kalman(pmats, bestpath, y)
  plots[[i]] = data.frame(measure = tempos$note_onset, tempo = c(y),
                  inferred = c(kal$ests), state = convert10to4(bestpath))
}
plots = bind_rows(plots)
plots$performer = rep(rownames(pvec_ml), each=length(y))
plots$clust = as.factor(rep(pvec_ml$clust, each=length(y)))
plots$pointx = 10
plots$pointy = 400
deunderscore = function(x) gsub('_',' ',x)
ggplot(plots) +
  geom_rect(data=data.frame(xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf),
              aes(xmin=xmin,xmax=xmax,ymin=ymin,ymax=ymax),
              fill = 'gray90', color = 'gray90') +
  geom_line(aes(x=measure, y=tempo), color='black') +
  geom_point(aes(x=measure, y=inferred, color=as.factor(state))) +
  scale_color_manual(values = c("blue","red","green","orange")) +
  theme(legend.position = 'none', legend.title = element_blank()) +
  facet_wrap(~performer, labeller = labeller(performer = deunderscore)) +
  geom_point(aes(x=pointx,y=pointy,color=clust),alpha=.5, size=5, shape=15) +
  scale_color_manual(values = fivecolors)
```

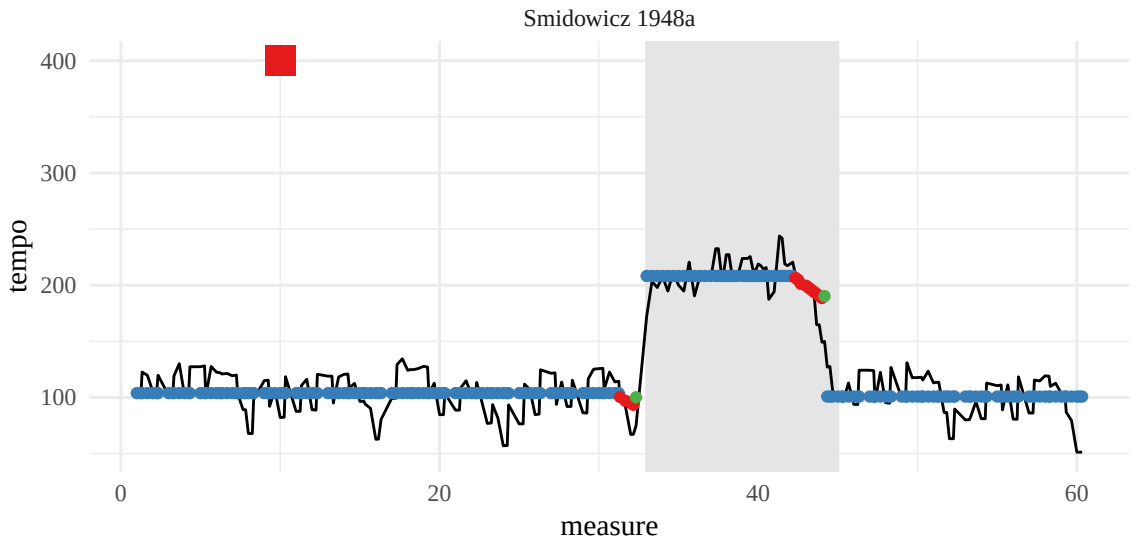



```
ggplot(filter(plots, performer == 'Milkina_1970')) +
  geom_rect(data=data.frame(xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf),
              aes(xmin=xmin,xmax=xmax,ymin=ymin,ymax=ymax),
              fill = 'gray90', color = 'gray90') +
  geom_line(aes(x=measure, y=tempo), color='black') +
  geom_point(aes(x=measure, y=inferred, color=as.factor(state))) +
  scale_color_manual(values = c("blue","red","green","orange")) +
  theme(legend.position = 'none', legend.title = element_blank()) +
  facet_wrap(~performer, labeller = labeller(performer = deunderscore)) +
  geom_point(aes(x=pointx,y=pointy,color=clust),alpha=.5, size=5, shape=15) +
  scale_color_manual(values = fivecolors)
```

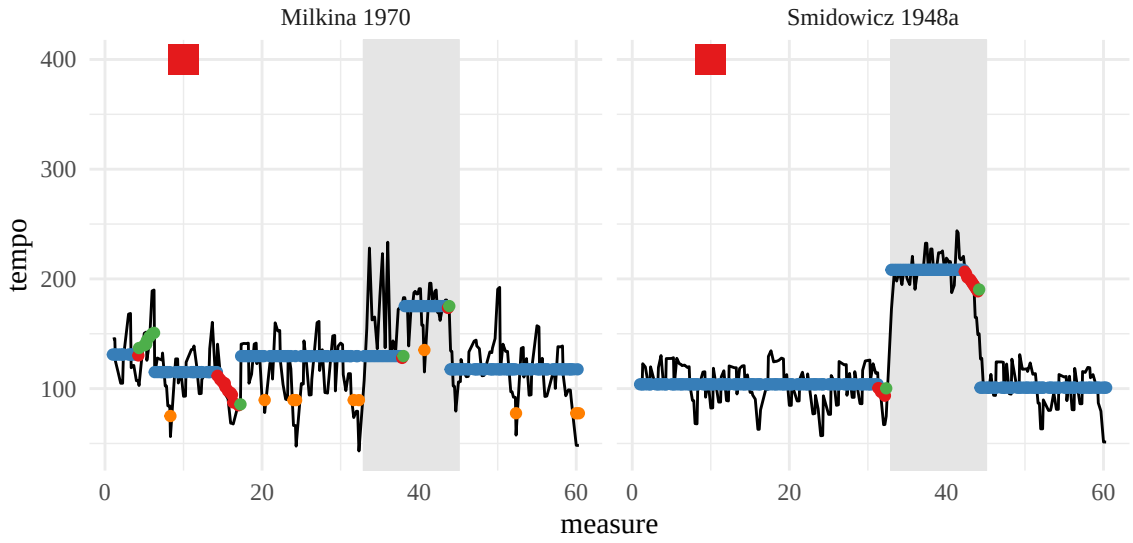



```
ggplot(filter(plots, performer == 'Smidowicz_1948a')) +
  geom_rect(data=data.frame(xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf),
              aes(xmin=xmin,xmax=xmax,ymin=ymin,ymax=ymax),
              fill = 'gray90', color = 'gray90') +
  geom_line(aes(x=measure, y=tempo), color='black') +
```

```
  geom_point(aes(x=measure, y=inferred, color=as.factor(state))) +
  scale_color_manual(values = c("blue","red","green","orange")) +
  theme(legend.position = 'none', legend.title = element_blank()) +
  facet_wrap(~performer, labeller = labeller(performer = deunderscore)) +
  geom_point(aes(x=pointx,y=pointy,color=clust),alpha=.5, size=5, shape=15) +
  scale_color_manual(values = fivecolors)
```




```
ggplot(filter(plots, performer %in% c('Milkina_1970','Smidowicz_1948a'))) +
  geom_rect(data=data.frame(xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf),
              aes(xmin=xmin,xmax=xmax,ymin=ymin,ymax=ymax),
              fill = 'gray90', color = 'gray90') +
  geom_line(aes(x=measure, y=tempo), color='black') +
  geom_point(aes(x=measure, y=inferred, color=as.factor(state))) +
  scale_color_manual(values = c("blue","red","green","orange")) +
  theme(legend.position = 'none', legend.title = element_blank()) +
  facet_wrap(~performer, labeller = labeller(performer = deunderscore)) +
  geom_point(aes(x=pointx,y=pointy,color=clust),alpha=.5, size=5, shape=15) +
  scale_color_manual(values = fivecolors)
```

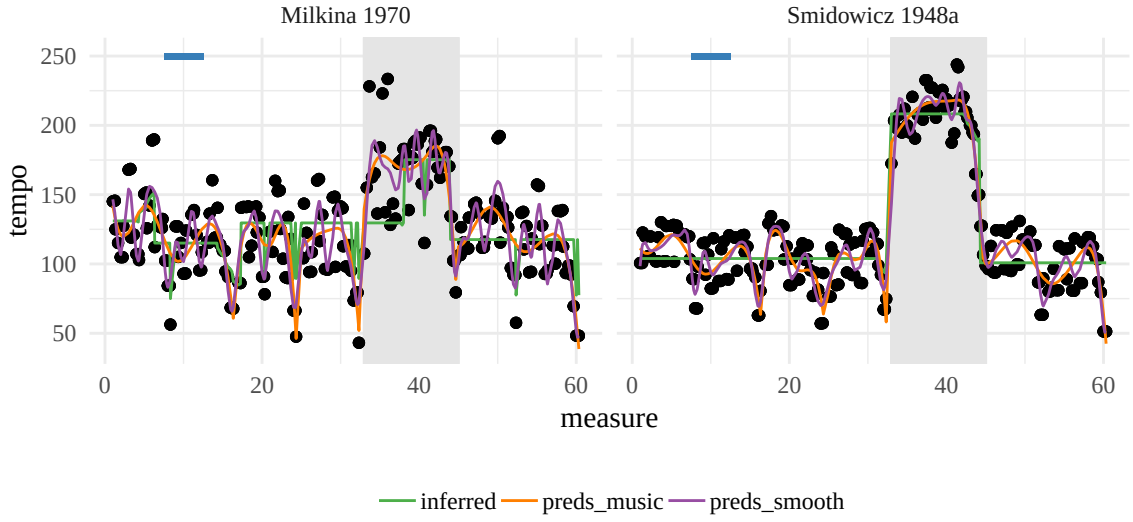


Different smoothing

Try splines, replicating knots, l1tf?

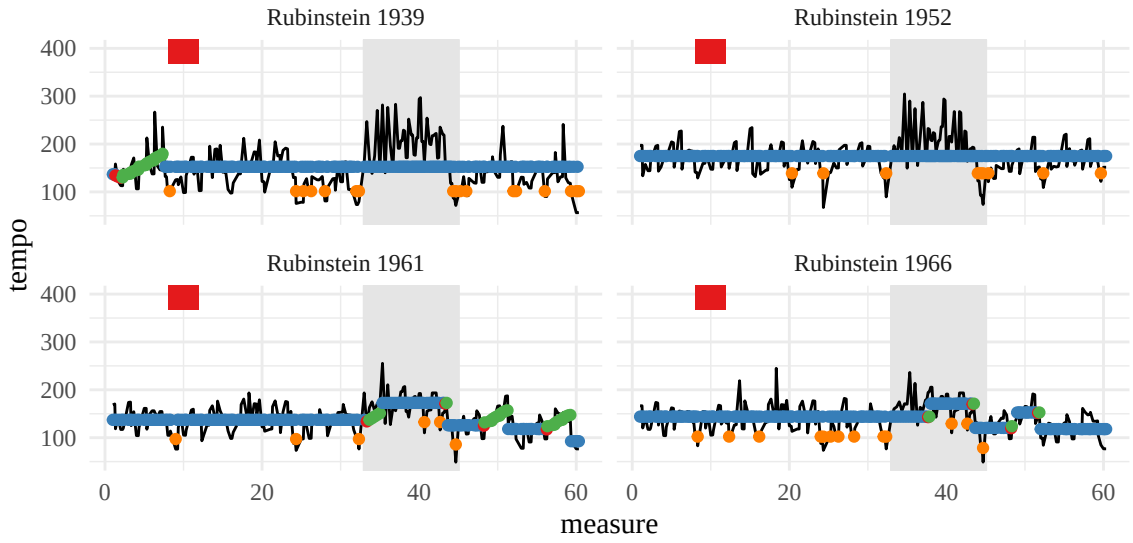
```
nsplines = 64 # 1 knot per bar plus boundary
B = bs(tempos$note_onset, df=nsplines, intercept = TRUE)
single.knots = match(seq(4,56,by=4)+1,tempos$meas_num)
double.knots = match(c(16,24,32,44)+1, tempos$meas_num)
triple.knots = match(c(16,24,32,44)+1, tempos$meas_num)
quad.knots = match(c(16,24,32,44)+1, 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))

spline_music = plots %>% group_by(performer) %>%
  mutate(preds_smooth = fitted(lm(tempo~B-1)),
         preds_music = fitted(lm(tempo~B1-1)),
         pointy = 250)
filter(spline_music , performer %in% c('Milkina_1970','Smidowicz_1948a')) %>%
  gather(key='key',value = 'value', -tempo,
         -measure, -state, -performer, -clust, -pointx, -pointy) %>%
  ggplot() +
  geom_rect(data=data.frame(xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf),
              aes(xmin=xmin,xmax=xmax,ymin=ymin,ymax=ymax),
              fill = 'gray90', color = 'gray90', show.legend = FALSE) +
  geom_point(aes(x=measure, y=tempo), color='black', show.legend = FALSE) +
  geom_line(aes(x=measure, y=value, color=key)) +
  scale_color_manual(values=fivecolors[3:5]) +
  theme(legend.position = 'bottom', legend.title = element_blank()) +
  facet_wrap(~performer, labeller = labeller(performer = deunderscore)) +
  geom_rect(aes(xmin=pointx-2.5, xmax=pointx+2.5,
                ymin=pointy-2.5,ymax=pointy+2.5,fill=clust),
            show.legend = FALSE,
             alpha=.5,size=5) +
```

```
  scale_fill_manual(values = fivecolors)
```



Similar performances

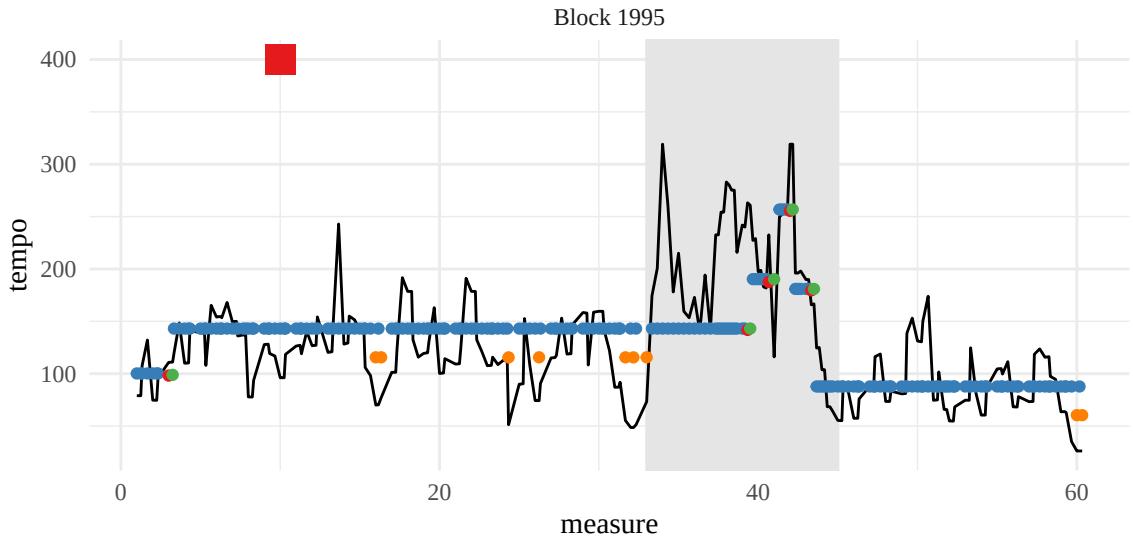
```
ggplot(filter(plots, substr(performer,1,5) == 'Rubin')) +
  geom_rect(data=data.frame(xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf),
              aes(xmin=xmin,xmax=xmax,ymin=ymin,ymax=ymax),
              fill = 'gray90', color = 'gray90') +
  geom_line(aes(x=measure, y=tempo), color='black') +
  geom_point(aes(x=measure, y=inferred, color=as.factor(state))) +
  scale_color_manual(values = c("blue","red","green","orange")) +
  theme(legend.position = 'none', legend.title = element_blank()) +
  facet_wrap(~performer, labeller = labeller(performer = deunderscore)) +
  geom_point(aes(x=pointx,y=pointy,color=clust),alpha=.5, size=5, shape=15) +
  scale_color_manual(values = fivecolors)
```




```
# note that the 1952 recording is the only one in a different cluster
```

Bad estimation

```
ggplot(filter(plots, performer == 'Block_1995')) +
  geom_rect(data=data.frame(xmin = 33, xmax = 45, ymin = -Inf, ymax = Inf),
              aes(xmin=xmin,xmax=xmax,ymin=ymin,ymax=ymax),
              fill = 'gray90', color = 'gray90') +
  geom_line(aes(x=measure, y=tempo), color='black') +
  geom_point(aes(x=measure, y=inferred, color=as.factor(state))) +
  scale_color_manual(values = c("blue","red","green","orange")) +
  theme(legend.position = 'none', legend.title = element_blank()) +
  facet_wrap(~performer, labeller = labeller(performer = deunderscore)) +
  geom_point(aes(x=pointx,y=pointy,color=clust),alpha=.5, size=5, shape=15) +
  scale_color_manual(values = fivecolors)
```



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?)