How Many Title Contenders? (draft)

NHR

Sunday, January 04, 2015

Introduction and The Model

I've frequently got the "winner 2014-15" odds from Betfair and will use them as one over (implied) probability here.

I use a c# code for downloading the data and storing them in a SQL database. The CSV file used here is extracted from the DB with a BCP QUERYOUT command.

```r
# Read Data
odds_all <- read.table("./data/PL_Odds_History.csv", header = FALSE,
                       sep = "\t", stringsAsFactor = FALSE);
colnames(odds_all) <- c("Market", "Team", "Back", "Lay", "Time_char");
odds_all$Time <- strptime(odds_all$Time_char, "%Y-%m-%d %H:%M:%S.000")

# I keep the character one because I'm gonna use it as the factor in split function later.

# Subset Data
odds_winner <- odds_all[odds_all$Market == "Winner",]
row.names(odds_winner) <- NULL

head(odds_winner); summary(odds_winner);
```

```
##   Market Team Back      Lay               Time_char                Time
## 1 Winner  ARS   11    11.50 2014-06-27 18:00:45.000 2014-06-27 18:00:45
## 2 Winner  AVL 1000 10000.00 2014-06-27 18:00:45.000 2014-06-27 18:00:45
## 3 Winner  BUR 1000 10000.00 2014-06-27 18:00:45.000 2014-06-27 18:00:45
## 4 Winner  CHE    3     3.05 2014-06-27 18:00:45.000 2014-06-27 18:00:45
## 5 Winner  CRY 1000 10000.00 2014-06-27 18:00:45.000 2014-06-27 18:00:45
## 6 Winner  EVE  180   200.00 2014-06-27 18:00:45.000 2014-06-27 18:00:45
```

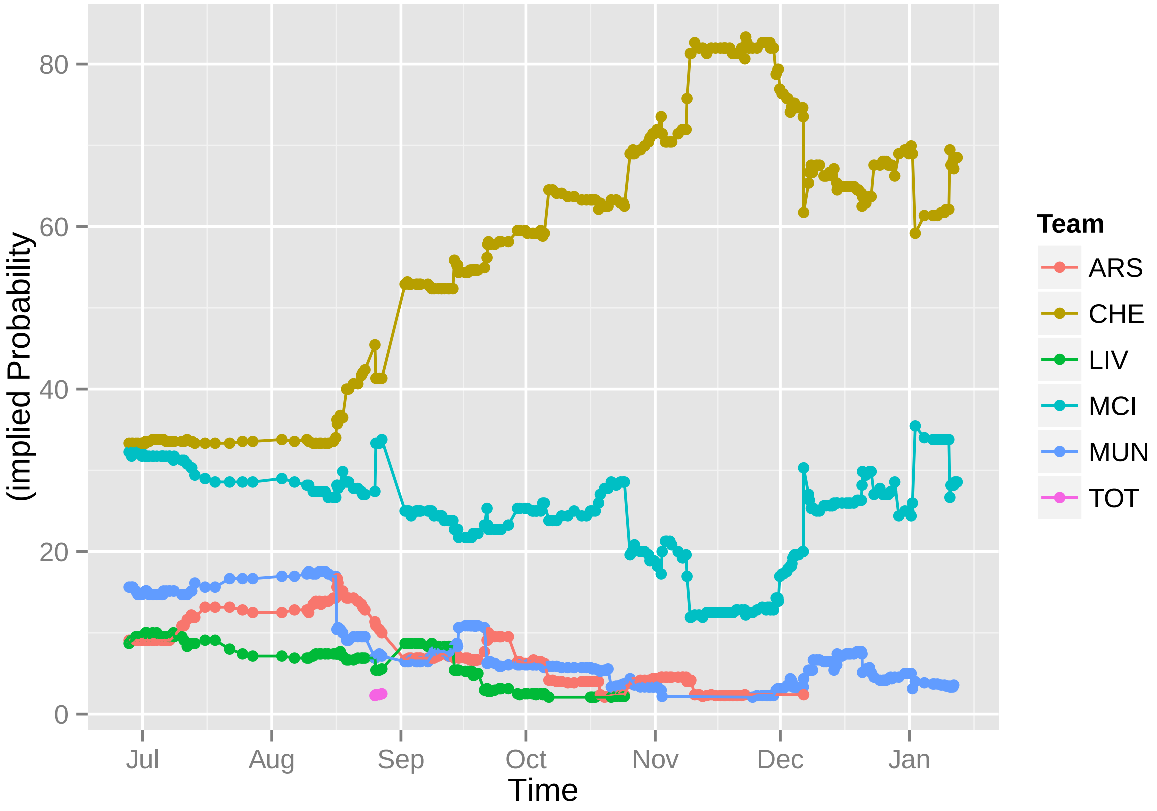
```
##     Market              Team                Back             Lay
##  Length:5680        Length:5680        Min.   :   1.2   Min.   :    1.22
##  Class :character   Class :character   1st Qu.: 100.0   1st Qu.:  110.00
##  Mode  :character   Mode  :character   Median :1000.0   Median :10000.00
##                                        Mean   : 682.2   Mean   : 6283.68
##                                        3rd Qu.:1000.0   3rd Qu.:10000.00
##                                        Max.   :1000.0   Max.   :10000.00
##   Time_char              Time
##  Length:5680        Min.   :2014-06-27 18:00:45
##  Class :character   1st Qu.:2014-09-07 00:48:55
##  Mode  :character   Median :2014-10-16 17:23:42
##                     Mean   :2014-10-13 07:27:28
##                     3rd Qu.:2014-12-01 12:00:21
##                     Max.   :2015-01-12 11:04:47
```

Let's have a look at the (implied) peobability of each of the 20 teams winning the title vs. time.
Note: add a paragraph about back/lay, errorbar, and normalising the sum of probabilities to one).

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.0.3
```

```
prob_min = 2

qplot(Time,
      100/Back,
      data = odds_winner[odds_winner$Back < 100/prob_min, ],
      colour = Team,
      ylab = "(implied Probability",
      ) +
    geom_line()
```



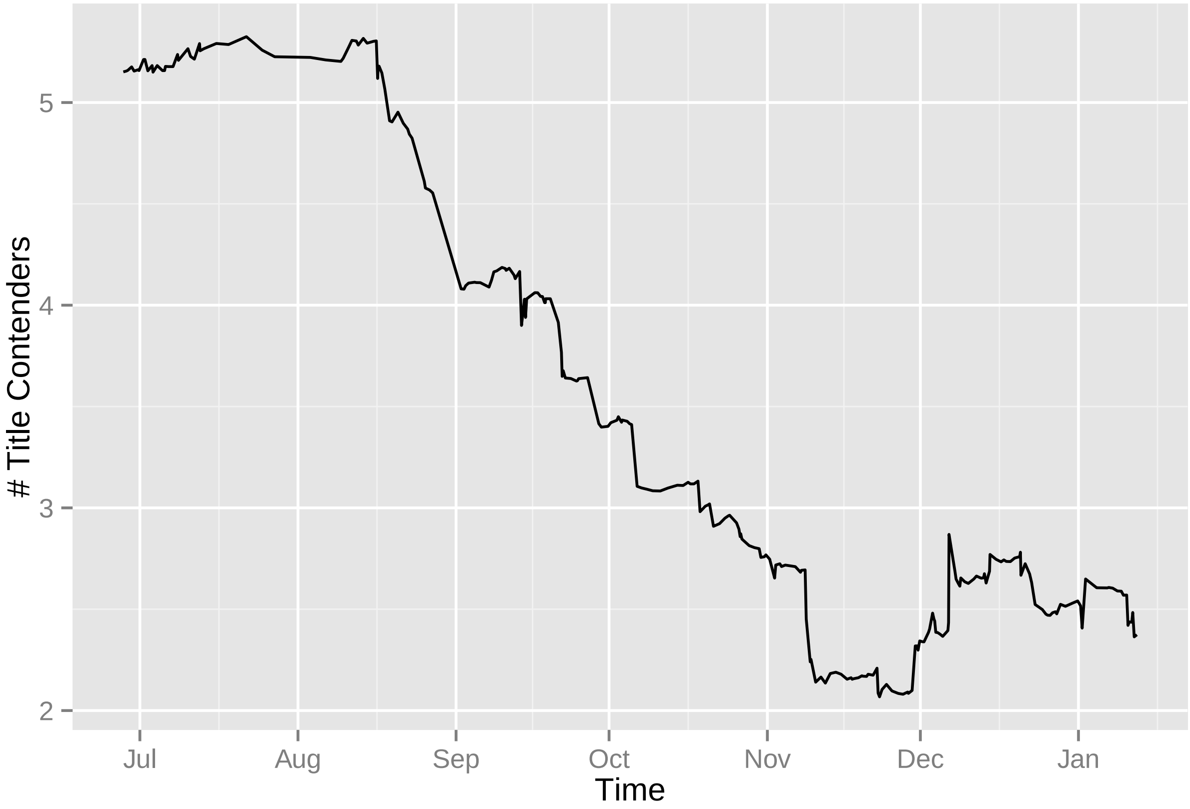
All the data points with (implied) probability smaller than 2% are removed.

Note: a few lines about Shannon Entropy and Information

Using Shannon Entropy, number of title contenders vs. time looks like:

```
inf_time <- data.frame(
    sapply(
        split(
            odds_winner,
            factor(odds_winner$Time_char)),
        function(x) sum(log(x$Back)/x$Back)))
```

```
colnames(inf_time) <- "inf"

inf_time$Time <- strptime(row.names(inf_time), "%Y-%m-%d %H:%M:%S.000")
row.names(inf_time) <- NULL

library(ggplot2)

plot_tc <- ggplot(inf_time, aes(x=Time, y=exp(inf))) +
    geom_line() +
    xlab("Time") +
    ylab("# Title Contenders")

plot_tc
```



TA DA!

The next question is “What real events do those sudden jumps correspond to?”

```
library(reshape2)
```

```
## Warning: package 'reshape2' was built under R version 3.0.3
```

```
data_back <- dcast(odds_winner, Time ~ Team , value.var = "Back")
data_back <- data_back[order(data_back$Time),]
```

```r
inf <- log(data_back[,2:21])/data_back[,2:21]
delta_inf <- inf[1:dim(inf)[1]-1,]- inf[2:dim(inf)[1],]
delta_inf <- data.frame(Time =data_back$Time[2:length(data_back$Time)], delta_inf)


# melt back
delta_inf_melt <- melt(delta_inf, id = 1)


delta_inf_melt[which.max(delta_inf_melt$value),]
```

```
##                      Time variable      value
## 2877 2014-08-16 14:56:23      MUN 0.06523906
```

```r
delta_inf_major <- delta_inf_melt[order(-abs(delta_inf_melt$value)),]

# Major changes in teams' inf contributions
head(delta_inf_major)
```

```
##                      Time variable       value
## 1074 2014-12-06 14:45:26      CHE -0.07170223
## 2877 2014-08-16 14:56:23      MUN  0.06523906
## 2943 2014-09-21 15:52:32      MUN  0.06508658
## 3112 2015-01-11 22:19:57      MUN  0.05831451
## 1118 2015-01-02 09:30:00      CHE -0.05423953
## 2327 2014-09-01 23:49:13      LIV -0.05180179
```

```r
# do the same for total inf.

delta_tot_inf <- inf_time[1:dim(inf_time)[1]-1,1]- inf_time[2:dim(inf_time)[1],1]
delta_tot_inf <- data.frame(Time = inf_time[2:dim(inf_time)[1],2],
                            delta_tot_inf)


delta_tot_inf_melt <- melt(delta_tot_inf, id = 1)
delta_tot_inf_melt[which.max(delta_tot_inf_melt$value),]
```

```
##                    Time      variable     value
## 63 2014-09-01 23:49:13 delta_tot_inf 0.1097697
```

```r
delta_tot_inf_major <- delta_tot_inf_melt[order(-abs(delta_tot_inf_melt$value)),]
head(delta_tot_inf_major)
```

```
##                     Time      variable       value
## 225 2014-12-06 14:45:26 delta_tot_inf -0.16447834
## 63  2014-09-01 23:49:13 delta_tot_inf  0.10976972
## 208 2014-11-29 23:59:09 delta_tot_inf -0.09956278
## 269 2015-01-02 09:30:00 delta_tot_inf -0.09594705
## 177 2014-11-08 14:54:58 delta_tot_inf  0.09429239
## 133 2014-10-06 12:04:46 delta_tot_inf  0.09363281
```

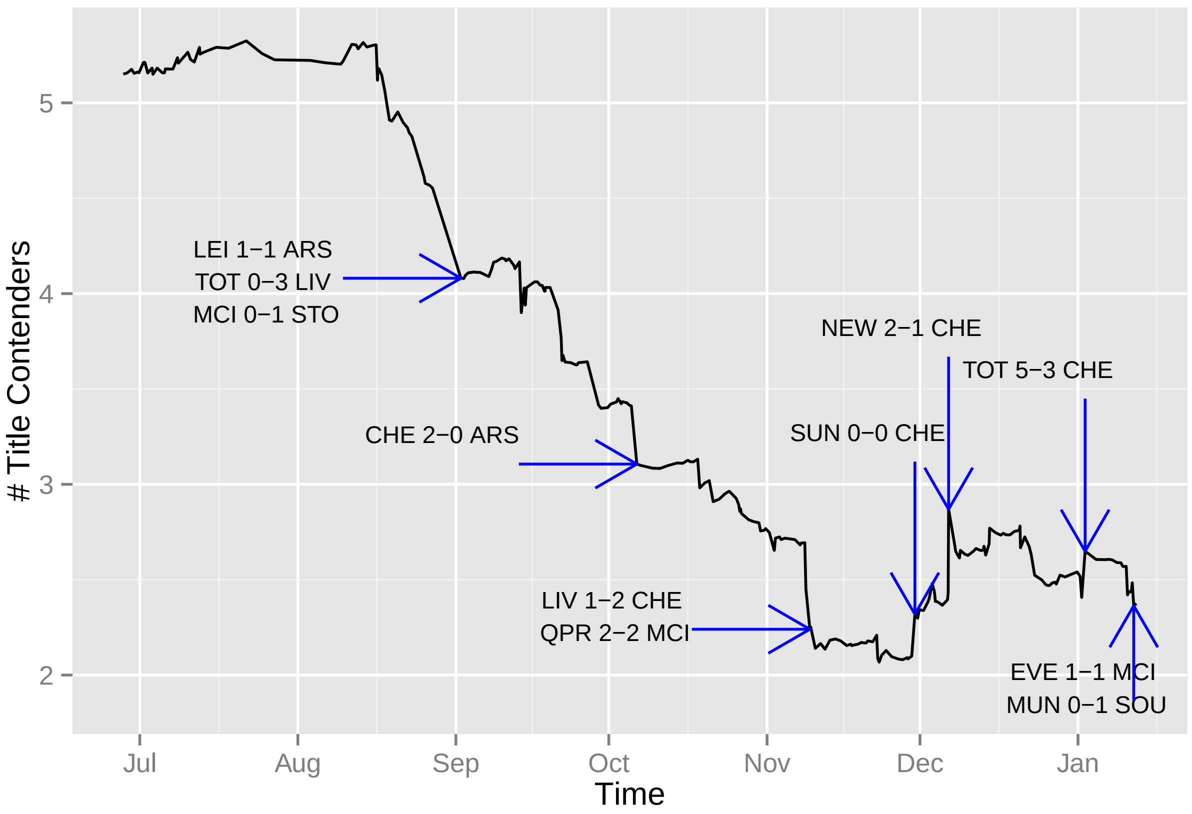
```
major_events_DF <-
    merge(
        dcast(delta_inf_major[1:100,], Time ~ variable , value.var = "value"),


        dcast(delta_tot_inf_major[1:7,], Time ~ variable , value.var = "value")
        , by = "Time")


major_events_DF <- major_events_DF[order(-abs(major_events_DF$delta_tot_inf)),]

major_events_DF
```

```
##                   Time         ARS         CHE EVE         LIV         MCI
## 6 2014-12-06 14:45:26          NA -0.07170223  NA          NA -0.03990710
## 1 2014-09-01 23:49:13  0.04972183  0.02838006  NA -0.05180179  0.02004441
## 5 2014-11-29 23:59:09 -0.01597259 -0.02520978  NA          NA -0.01463797
## 7 2015-01-02 09:30:00          NA -0.05423953  NA          NA -0.01748832
## 3 2014-11-08 14:54:58  0.01174690  0.02658210  NA  0.01621571  0.01861955
## 2 2014-10-06 12:04:46  0.04086789  0.02774514  NA  0.01157197          NA
## 4 2014-11-09 09:35:52          NA  0.04202287  NA          NA  0.04747846
##           MUN        SOU SWA        TOT delta_tot_inf
## 6 -0.02295259         NA  NA         NA   -0.16447834
## 1  0.01167571         NA  NA 0.04617028    0.10976972
## 5 -0.01771218         NA  NA         NA   -0.09956278
## 7 -0.02045079         NA  NA         NA   -0.09594705
## 3          NA 0.01275724  NA         NA    0.09429239
## 2          NA         NA  NA         NA    0.09363281
## 4          NA         NA  NA         NA    0.09021432
```

and the graph

```
# Now investigate!
# 2014-12-06 14:45:26       NEW 2-1 CHE
# 2014-09-01 23:49:13       LEI 1-1 ARS TOT 0-3 LIV MCI 0-1 STO
# 2014-11-29 23:59:09       SUN 0-0 CHE
# 2015-01-02 09:30:00       TOT 5-3 CHE
# 2014-11-08 14:54:58       LIV 1-2 CHE
# 2014-10-06 12:04:46       CHE 2-0 ARS
# 2014-11-09 09:35:52       QPR 2-2 MCI


# Back to the original graph

event_time <- c('2014-12-06 14:45:26', '2014-09-01 23:49:13',
                '2014-11-29 23:59:09', '2015-01-02 09:30:00',
                '2014-10-06 12:04:46', '2014-11-09 09:35:52',
                '2015-01-11 22:19:57')

event_res <- c('NEW 2-1 CHE', 'LEI 1-1 ARS \nTOT 0-3 LIV \nMCI 0-1 STO',
               'SUN 0-0 CHE', 'TOT 5-3 CHE', 'CHE 2-0 ARS',
               'LIV 1-2 CHE \nQPR 2-2 MCI', 'EVE 1-1 MCI \nMUN 0-1 SOU')
```

```r
event <- data.frame(Time = event_time, result = event_res)

event$Time <- strptime(event$Time, "%Y-%m-%d %H:%M:%S")


event <- event[order(event$Time),]


lab_arr <- subset(inf_time, Time %in% event$Time)

lab_arr$x    <- lab_arr$Time
lab_arr$y    <- exp(lab_arr$inf) + 0.8
lab_arr$xend <- lab_arr$Time
lab_arr$yend <- exp(lab_arr$inf)
lab_arr$inf  <- log(exp(lab_arr$inf)+1)
lab_arr$Time <- lab_arr$Time -800000

lab_arr$y[1] <- lab_arr$y[1] - 0.8
lab_arr$inf[1] <- log(exp(lab_arr$inf[1])-0.8)
lab_arr$x[1] <- lab_arr$x[1] - 2000000
lab_arr$Time[1] <- lab_arr$Time[1] - 2500000

lab_arr$y[2] <- lab_arr$y[2] - 0.8
lab_arr$inf[2] <- log(exp(lab_arr$inf[2])-0.8)
lab_arr$x[2] <- lab_arr$x[2] - 2000000
lab_arr$Time[2] <- lab_arr$Time[2] - 2500000

lab_arr$y[3] <- lab_arr$y[3] - 0.8
lab_arr$inf[3] <- log(exp(lab_arr$inf[3])-0.8)
lab_arr$x[3] <- lab_arr$x[3] - 2000000
lab_arr$Time[3] <- lab_arr$Time[3] - 2500000

lab_arr$y[7] <- lab_arr$y[7] - 1.3
lab_arr$inf[7] <- log(exp(lab_arr$inf[7])-1.3)

#lab_arr$Time[1] <- lab_arr$Time[1] + 1000000
#lab_arr$inf[1]  <- log(exp(lab_arr$inf)-1)

library(grid)

plot_tc <- ggplot(inf_time, aes(x=Time, y=exp(inf))) +
    geom_line() +
    xlab("Time") +
    ylab("# Title Contenders")  +
    geom_text(data=lab_arr,
          label=event$res , vjust=1, size = 3) +
    geom_segment(data=lab_arr,
                 mapping=aes(x= x, y=y, xend=xend, yend=yend),
                 arrow=arrow(), size=0.5, color="blue")

plot_tc
```


LEI 1−1 ARS
TOT 0−3 LIV
MCI 0−1 STO
CHE 2−0 ARS
LIV 1−2 CHE
QPR 2−2 MCI
NEW 2−1 CHE
TOT 5−3 CHE
SUN 0−0 CHE
EVE 1−1 MCI
MUN 0−1 SOU
# Title Contenders
5
4
3
2
Jul
Aug
Sep
Oct
Nov
Dec
Jan
Time