Win_Probability

Cosa serve per vincere nel basket?

Per cominciare, carichiamo i dati

```r
data <- read.table("data/team_stats.txt", header = T)
head(data)
```

```
   W  L  FG3M.GP  FG3A.GP Possessions.GP      eFG.      TOV.      ORB.
1 36 46 13.71951 37.70732       116.2517 0.5394910 0.1164420 0.2795523
2 64 18 16.47561 42.46341       111.0093 0.5782180 0.1075498 0.2305870
3 32 50 13.28049 36.70732       111.4127 0.5307924 0.1177779 0.2596180
4 21 61 12.06098 34.00000       108.8693 0.5293004 0.1264663 0.2316076
5 39 43 11.47561 32.07317       111.0322 0.5339283 0.1102734 0.2549402
6 48 34 13.51220 36.78049       109.6980 0.5566592 0.1236209 0.2276056
        FTR
1 0.2513186
2 0.2236344
3 0.2341590
4 0.2119725
5 0.2358632
6 0.2337717
```

Come primo modello consideriamo come esplicative le componenti principali derivanti dai 4 fattori

```r
apply(data[, 6:9], 2, sd)
```

```
       eFG.        TOV.        ORB.         FTR
0.019345562 0.008797573 0.022746742 0.019644549
```

La percentuale di palle perse ha una varianza molto diversa dalle altre metriche. E' necessario standardizzare i dati

```
pca <- prcomp(data[, 6:9], scale. = TRUE)
```

Guardiamo la percentuale di varianza spiegata cumulata per capire quante componenti utilizzare

```
cumsum((pca$sdev ^ 2) / sum(pca$sdev ^ 2))
```

```
[1] 0.4586319 0.7454943 0.9205574 1.0000000
```

2 componenti principali spiegano una il 74% della varianza totale

```
X.0 <- pca$x[, c(1, 2)]
```

Ora che ho ricavato le esplicative, posso procedere con la costruzione del modello

```
win.prob.glm.pca <- glm(cbind(data[, 1], data[, 2]) ~ X.0[, 1] + X.0[, 2], family = binomial
summary(win.prob.glm.pca)
```

```
Call:
glm(formula = cbind(data[, 1], data[, 2]) ~ X.0[, 1] + X.0[,
    2], family = binomial)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.003983   0.041528  -0.096    0.924
X.0[, 1]     0.377461   0.033112  11.400   <2e-16 ***
X.0[, 2]     0.050898   0.039330   1.294    0.196
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 266.01  on 29  degrees of freedom
Residual deviance: 122.61  on 27  degrees of freedom
AIC: 271

Number of Fisher Scoring iterations: 4
```

Il coefficiente della seconda variabile canonica è non significativo. Anche l'intercetta risulta non significativa, con il relativo coefficiente sostanzialmente degenere in 0. Aggiorniamo il modello

```r
win.prob.glm.pca <- update(win.prob.glm.pca, . ~ . - X.0[, 2])
summary(win.prob.glm.pca)
```

```
Call:
glm(formula = cbind(data[, 1], data[, 2]) ~ X.0[, 1], family = binomial)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.003545   0.041512  -0.085    0.932
X.0[, 1]     0.377246   0.033098  11.398   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 266.01  on 29  degrees of freedom
Residual deviance: 124.28  on 28  degrees of freedom
AIC: 270.67

Number of Fisher Scoring iterations: 4
```

Guardando la devianza residua non siamo soddisfatti dell'adattamento del modello. Il modello corrente, con soli 2 parametri in più rispetto al modello nullo, spiega molto di più di quest'ultimo. Tuttavia viene rifiutata l'ipotesi di adattamento del modello corrente rispetto a quello nullo

H0: modello nullo H1: modello corrente

```
pchisq(win.prob.glm.pca$null.deviance - win.prob.glm.pca$deviance,
       win.prob.glm.pca$df.null - win.prob.glm.pca$df.residual, lower.tail = FALSE)
```

```
[1] 1.116071e-32
```

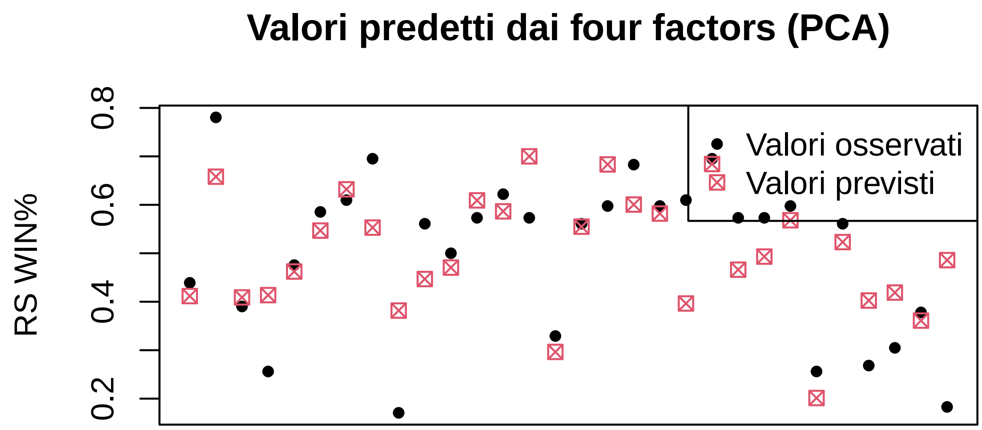
H0 : modello corrente H1: modello saturo

```
pchisq(win.prob.glm.pca$deviance, win.prob.glm.pca$df.residual, lower.tail = FALSE)
```

```
[1] 4.281288e-14
```

Compariamo i valori osservati con i valori previsti dal modello. L'asse x indica le squadre in ordine alfabetico

```
with(data,{
  plot(1:30, W / 82, pch = 20,
       xlab = "", xaxt = "n", ylab = "RS WIN%",
       main = "Valori predetti dai four factors (PCA)")
  points(1:30, fitted(win.prob.glm.pca), pch = 7, col = 2)
  legend("topright", legend = c("Valori osservati", "Valori previsti"),
         col = c(1, 2), pch = c(20, 7))
})
```



L'adattamento non è sicuramente dei migliori. Il modello, utilizzando solamente due esplicative, deve mediare le percentuali osservate e nei casi estremi (numero di vittorie molto alto o molto basso) dà risultati fuorvianti

Costruiamo un altro modello prendendo come esplicative i 4 fattori senza applicare la PCA. La PCA comporta infatti una riduzione della dimensionalità: il modello che ne deriva ha, oltre all'intercetta, 2 parametri (numero di PC considerate) invece che 4. Vediamo se, con più esplicative, il modello si adatta meglio ai dati

```r
win.prob.glm.0 <- glm(cbind(W, L) ~ eFG. + TOV. + ORB. + FTR,
                      family = binomial, data = data)
summary(win.prob.glm.0)
```

```
Call:
glm(formula = cbind(W, L) ~ eFG. + TOV. + ORB. + FTR, family = binomial,
    data = data)

Coefficients:
```

```
             Estimate Std. Error z value Pr(>|z|)    
(Intercept)  -16.529      2.464  -6.707 1.98e-11 ***
eFG.          27.151      3.110   8.730  < 2e-16 ***
TOV.         -22.573      5.395  -4.184 2.87e-05 ***
ORB.           6.779      2.444   2.773  0.00555 ** 
FTR           11.353      2.272   4.997 5.84e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 266.010  on 29  degrees of freedom
Residual deviance:  66.058  on 25  degrees of freedom
AIC: 218.45

Number of Fisher Scoring iterations: 4
```

Il modello appena costruito fornisce un adattamento migliore ai dati rispetto a quello visto in precedenza

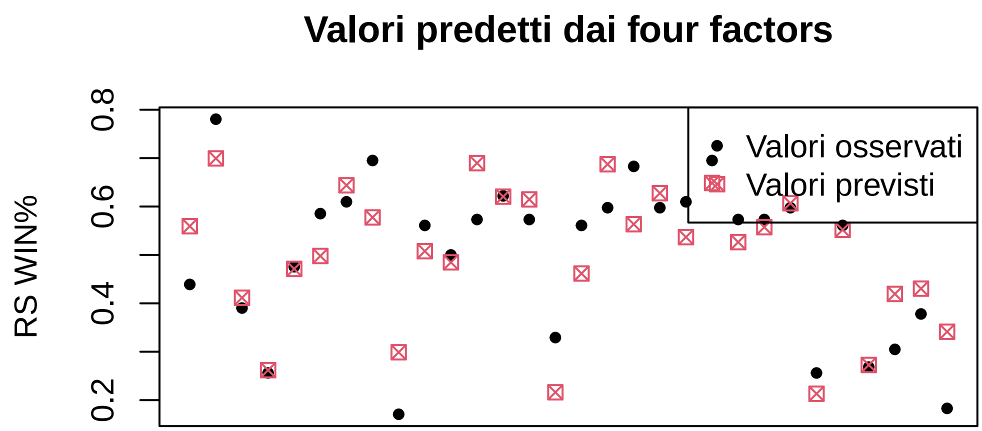
```r
compare.matrix <- matrix(data = c(win.prob.glm.pca$aic, win.prob.glm.0$aic,
      BIC(win.prob.glm.pca), BIC(win.prob.glm.0)), nrow = 2, byrow = TRUE)
colnames(compare.matrix) <- c("PCA", "4Factors")
row.names(compare.matrix) <- c("AIC", "BIC")
compare.matrix
```

```
         PCA 4Factors
AIC 270.6745 218.4485
BIC 273.4769 225.4545
```

Anche qui confrontiamo valori osservati e predetti

```r
with(data,{
  plot(1:30, W / 82, pch = 20,
       xlab = "", xaxt = "n", ylab = "RS WIN%",
       main = "Valori predetti dai four factors")
  points(1:30, fitted(win.prob.glm.0), pch = 7, col = 2)
  legend("topright", legend = c("Valori osservati", "Valori previsti"),
```

```r
        col = c(1, 2), pch = c(20, 7))
})
```



L'adattamento è visibilmente migliorato, tuttavia non siamo soddisfatti rispetto a ciò che otteniamo con il modello saturo

H0: modello corrente H1: modello saturo

```r
pchisq(win.prob.glm.0$deviance, win.prob.glm.0$df.residual, lower.tail = FALSE)
```

```
[1] 1.45632e-05
```

Considerata l'importanza che hanno il tiro da tre punti e il pace (ritmo di gioco) nel basket moderno costruiamo un nuovo modello aggiungendo 3 esplicative: numero di tiri da tre segnati, numero di tiri da tre tentati e possessi per partita

```r
win.prob.glm <- glm(cbind(W, L) ~ ., family = binomial, data = data)
summary(win.prob.glm)
```

```
Call:
glm(formula = cbind(W, L) ~ ., family = binomial, data = data)

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)     -2.59283    3.46643  -0.748 0.454471
FG3M.GP          0.44301    0.20727   2.137 0.032564 *
FG3A.GP         -0.13037    0.08036  -1.622 0.104734
Possessions.GP  -0.10708    0.02956  -3.623 0.000292 ***
eFG.            18.71868    5.76091   3.249 0.001157 **
TOV.           -14.78279    5.66268  -2.611 0.009039 **
ORB.            10.49181    2.94775   3.559 0.000372 ***
FTR             10.19580    2.44917   4.163 3.14e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 266.010  on 29  degrees of freedom
Residual deviance:  31.607  on 22  degrees of freedom
AIC: 190

Number of Fisher Scoring iterations: 4
```

Il coefficiente relativo ai tiri da tre punti tentati a partita è non significativo. Aggiorniamo il modello

```r
win.prob.glm <- update(win.prob.glm, . ~ . - FG3A.GP)
summary(win.prob.glm)
```

```
Call:
glm(formula = cbind(W, L) ~ FG3M.GP + Possessions.GP + eFG. +
    TOV. + ORB. + FTR, family = binomial, data = data)

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)     -4.57984    3.23859  -1.414  0.15732
FG3M.GP          0.11591    0.04763   2.434  0.01495 *
```

```
Possessions.GP  -0.13370    0.02456  -5.444 5.20e-08 ***
eFG.            26.07929    3.55737   7.331 2.28e-13 ***
TOV.           -14.84186    5.65338  -2.625  0.00866 **
ORB.            12.05612    2.78137   4.335 1.46e-05 ***
FTR             11.00650    2.39943   4.587 4.49e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 266.010  on 29  degrees of freedom
Residual deviance:  34.239  on 23  degrees of freedom
AIC: 190.63

Number of Fisher Scoring iterations: 4
```

L'intercetta risulta non significativa

```r
win.prob.glm <- update(win.prob.glm, . ~ . - 1)
summary(win.prob.glm)
```

```
Call:
glm(formula = cbind(W, L) ~ FG3M.GP + Possessions.GP + eFG. +
    TOV. + ORB. + FTR - 1, family = binomial, data = data)

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
FG3M.GP          0.12845    0.04677   2.747  0.00602 **
Possessions.GP  -0.15610    0.01879  -8.309  < 2e-16 ***
eFG.            23.14268    2.87738   8.043 8.77e-16 ***
TOV.           -16.80767    5.47489  -3.070  0.00214 **
ORB.            11.14120    2.70475   4.119 3.80e-05 ***
FTR             10.33303    2.34414   4.408 1.04e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 266.010  on 30  degrees of freedom
```

```
Residual deviance:  36.243  on 24  degrees of freedom
AIC: 190.63

Number of Fisher Scoring iterations: 4
```

L'adattamento rispetto ai modelli precedenti è migliorato

```
compare.matrix <- cbind(compare.matrix, c(win.prob.glm$aic, BIC(win.prob.glm)))
colnames(compare.matrix) <- c("PCA", "4Factors", "4Factors + 3PTShot")
compare.matrix
```

```
         PCA 4Factors 4Factors + 3PTShot
AIC 270.6745 218.4485           190.6340
BIC 273.4769 225.4545           199.0412
```

Accettiamo l'ipotesi nulla ad un livello di significatività del 5%

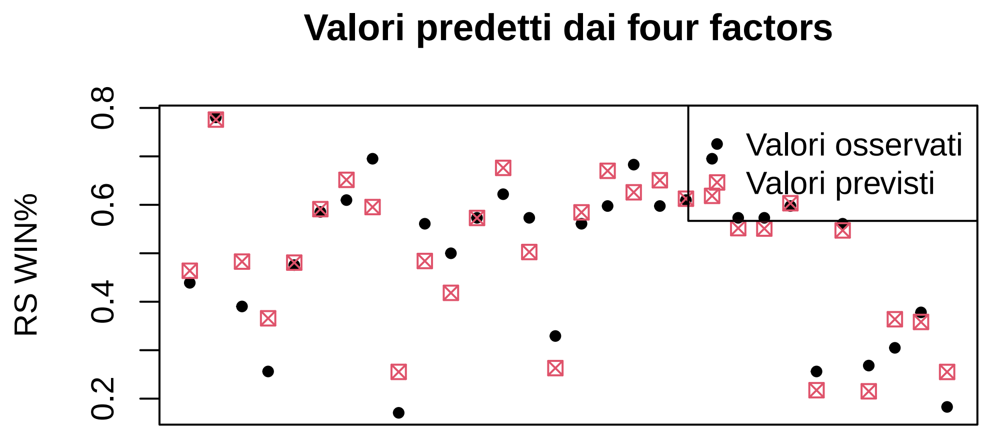
H0: modello corrente H1: modello saturo

```
pchisq(win.prob.glm$deviance, win.prob.glm$df.residual, lower.tail = FALSE)
```

```
[1] 0.05197609
```

Grafico valori previsti e osservati

```
with(data,{
  plot(1:30, W / 82, pch = 20,
       xlab = "", xaxt = "n", ylab = "RS WIN%",
       main = "Valori predetti dai four factors")
  points(1:30, fitted(win.prob.glm), pch = 7, col = 2)
  legend("topright", legend = c("Valori osservati", "Valori previsti"),
         col = c(1, 2), pch = c(20, 7))
})
```



Media della discrepanza tra il numero di vittorie effettivo e il numero di vittorie stimato

```
mean(abs(residuals(win.prob.glm, type = "response"))) * 82
```

```
[1] 3.85446
```

Metodo alternativo

```
mean(abs(data$W - fitted(win.prob.glm) * 82))
```

```
[1] 3.85446
```

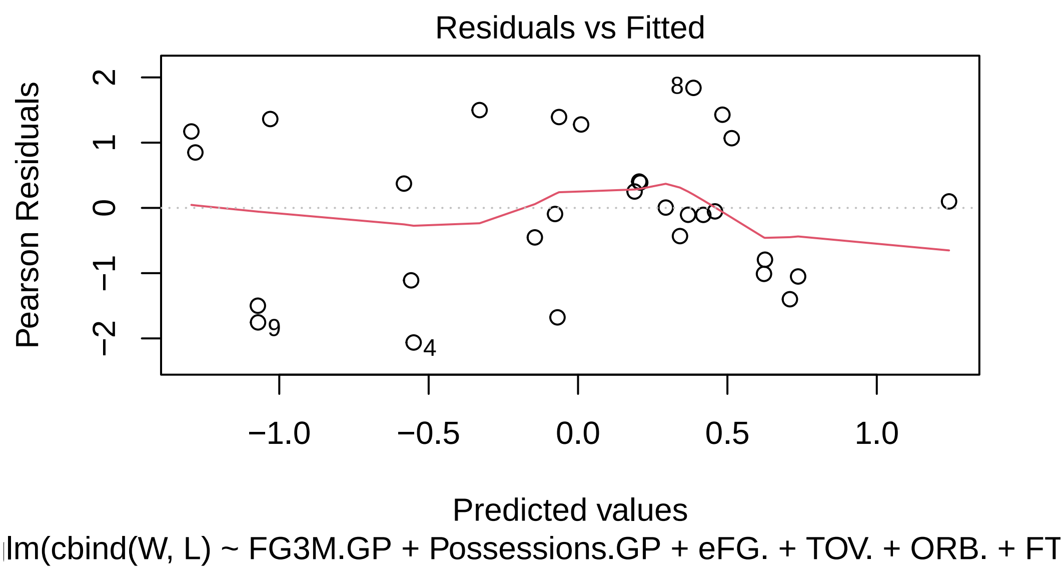
```
sd(abs(data$W - fitted(win.prob.glm) * 82))
```

```
[1] 2.685087
```

Ciò significa che in media sbaglio la previsione delle vittorie di 4 unità

Non vi è struttura di dipendenza tra i valori predetti e i residui

```r
plot(win.prob.glm, which = 1)
```



Le possibili interazioni aggiuntive non sono significative

```r
add1(win.prob.glm, . ~ . + (.)^2, test = "Chisq")
```

```
Single term additions

Model:
cbind(W, L) ~ FG3M.GP + Possessions.GP + eFG. + TOV. + ORB. +
    FTR - 1
                       Df Deviance    AIC    LRT Pr(>Chi)
<none>                      36.243 190.63
FG3M.GP:Possessions.GP  1   34.077 190.47 2.1661  0.14108
FG3M.GP:eFG.            1   34.787 191.18 1.4559  0.22758
```

```
FG3M.GP:TOV.          1   32.629 189.02 3.6141  0.05729 .
FG3M.GP:ORB.          1   34.230 190.62 2.0128  0.15598
FG3M.GP:FTR           1   36.057 192.45 0.1861  0.66621
Possessions.GP:eFG.   1   34.388 190.78 1.8551  0.17319
Possessions.GP:TOV.   1   34.092 190.48 2.1513  0.14245
Possessions.GP:ORB.   1   34.134 190.53 2.1087  0.14647
Possessions.GP:FTR    1   34.506 190.90 1.7368  0.18754
eFG.:TOV.             1   34.467 190.86 1.7764  0.18260
eFG.:ORB.             1   35.291 191.68 0.9525  0.32908
eFG.:FTR              1   34.807 191.20 1.4366  0.23070
TOV.:ORB.             1   34.068 190.46 2.1755  0.14022
TOV.:FTR              1   34.119 190.51 2.1240  0.14501
ORB.:FTR              1   35.778 192.17 0.4648  0.49538
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

In conclusione, il migliore dei modelli considerati è quello con le seguenti esplicative: tiri da tre punti segnati a partita, possessi a partita, percentuale effettiva, percentuale di palle perse, percentuale di rimbalzi offensivi e numero di tiri liberi rispetto ai tiri dal campo.