

## 1. Introducción

Hasta ahora hemos visto varios modelos lineales, los cuales se ajustan a distintas situaciones que se puedan presentar. Sin embargo, observamos que todos ellos sufren de algunos defectos comunes. Uno de ellos es que algunas variables explicativas de sobra pueden perjudicar el modelo, si es que están correlacionadas con variables explicativas importantes. Otros problemas se dan en el diseño del modelo, ya sea que subajustemos o sobreajustemos un modelo. En el caso de que contemos con muchas variables explicativas, también hay un problema de interpretación, ya que todas contribuyen a la predicción del modelo y puede dificultar el entendimiento de las interacciones de las cantidades que estudiamos.

En general, dada una variable dependiente  $Y$ , tendremos un conjunto de  $p$  variables explicativas  $X_1, \dots, X_p$  que podrían llegar a ser importantes para predecir  $Y$  -puede que algunas de estas variables sean una misma cantidad transformada-, pero no tenemos certezas de ello. Determinar adecuadamente las variables explicativas importantes es de suma importancia para atacar todos los problemas antes mencionados.

Hay varias estrategias que apuntan a solucionar estos problemas. Veremos algunas de ellas en este repartido.

## 2. Selección de modelos

La idea más intuitiva es, simplemente, probar todos los modelos posibles y quedarse con el mejor (luego de fijar algún criterio). Una idea posible es ajustar todos los posibles modelos lineales al tomar todos los posibles subconjuntos de variables predictivas (por ejemplo, minimizando el error cuadrático), y luego nos quedamos con el que minimice el error por validación cruzada. Lo más importante es que el criterio que usemos para comparar los modelos tenga en cuenta que dichos modelos usan cantidades de variables explicativas distintas (por eso usamos validación cruzada, y no el error cuadrático de la muestra).

El problema surge rápidamente cuando  $p$  es grande. Considerando de que el intercept siempre será considerado en todos los modelos, la cantidad total de modelos a evaluar es  $2^p$ . Para atacar este problema, se puede considerar una variante ambiciosa (*greedy*) de la selección de modelos. Se comienza desde el modelo que solo considera el intercept. Luego, se estudia cual es la variable explicativa que más contribuye a la predicción de  $Y$ . Y el siguiente paso es, dentro de las  $p - 1$  variables explicativas restantes, se estudia cual de ellas es la que mejora más el modelo que ya tenemos. Al final, llegaremos al modelo que contiene a todas las variables explicativas, pero nos quedaremos el modelo intermedio que nos de mejor error por validación cruzada.

Si bien la variante ambiciosa estudia y compara muchos menos modelos, no necesariamente va a llegar al modelo óptimo. Lo que sucede es que las variables explicativas que forman el mejor modelo con  $d$  variables explicativas, no necesariamente están contenidas en el conjunto de variables explicativas que forman el mejor modelo con  $d+1$  variables explicativas.

Otra variante ambiciosa es comenzar considerando a todas las variables predictivas, e ir eliminando las menos importantes en cada paso. La idea es análoga a la anterior, y se estudian y comparan la misma cantidad de modelos. Estas variantes ambiciosas no necesariamente llegan al mismo mejor modelo, o a la misma cantidad de variables importantes. Hay una observación importante a hacer, y es que si  $n > p$ , entonces no se puede comenzar considerando a todas las variables explicativas, y debemos comenzar necesariamente con ninguna variable explicativa.

Por último, también es posible combinar ambas variantes ambiciosas, fijando maneras de ir sucesivamente agregando o removiendo variables explicativas. Sin embargo, no se usan tan frecuentemente.

## 3. Criterios de selección

La estimación del error de modelo por validación cruzada no es el único criterio disponible para seleccionar un modelo. Para fijar notación, sea  $n$  el número de observaciones  $(Y_i, X_{1i}, \dots, X_{pi})$ , y  $0 \leq d \leq p$  el número de variables explicativas que usamos en cada modelo.

La estrategia de validación cruzada es intentar estimar el verdadero error de predicción del modelo. Sin embargo, otra estrategia posible es ajustar el error de predicción con la muestra de entrenamiento,  $SC_{Err}$ .

En esa línea, definimos

$$C_p = \frac{1}{n}(SC_{Err} + 2d\hat{\sigma}^2)$$

donde  $\hat{\sigma}^2$  es una estimación de la varianza de los residuos del modelo, que se estima usando las  $p$  variables explicativas. Lo que se puede probar es que, si  $\hat{\sigma}^2$  es un estimador insesgado de la varianza, entonces  $C_p$  es un estimador insesgado del error del modelo usando una muestra de prueba. Por lo tanto, debemos conservar el modelo que minimice  $C_p$ .

El criterio de información de Akaike (o *AIC*) está basado en la verosimilitud obtenida a partir del modelo ajustado. Recordamos que un criterio equivalente a la minimización de  $SC_{Err}$  es la maximización de la verosimilitud. En general,

se define

$$AIC = 2d - 2\log(\hat{L})$$

donde  $\hat{L}$  es el valor de la verosimilitud obtenido. Se puede probar, bajo los supuestos de un modelo lineal multivariado, que

$$AIC = \frac{1}{n\hat{\sigma}^2}(SC_{Err} + 2d\hat{\sigma}^2)$$

y como  $\hat{\sigma}^2$  no depende de  $d$ , quiere decir que los criterios  $C_p$  y  $AIC$  son proporcionales, y por lo tanto equivalentes.

El criterio de información bayesiana (o  $BIC$ ) surge de un criterio similar a  $AIC$ . En general, se define

$$BIC = d\log(n) - 2\log(\hat{L})$$

y se estima la distribución de los datos a partir de un enfoque bayesiano. Se puede probar, bajo los supuestos de un modelo lineal multivariado, que

$$BIC = \frac{1}{n\hat{\sigma}^2}(SC_{Err} + \log(n)d\hat{\sigma}^2).$$

Observar que el criterio obtenido es similar a  $AIC$ , pero cambia la penalización a  $d$  de 2 a  $\log(n)$ . Como  $\log(n) > 2$  para todo  $n > 7$ , tenemos que casi siempre, la penalización a la cantidad de variables explicativas en el modelo es mayor en  $BIC$ , y por lo tanto obtendremos modelos con menos variables explicativas importantes.

Por último, tenemos el estadístico  $R^2$  ajustado. Recordamos que el estadístico  $R^2$  es

$$R^2 = 1 - \frac{SC_{Err}}{SCT}$$

Donde  $SCT = \sum_{i=1}^n (Y_i - \bar{Y})^2$ . Al igual que con  $SC_{Err}$ , el estadístico  $R^2$  mejora al aumentar la cantidad de variables explicativas. Por lo tanto, añadimos una penalización para ajustarlo

$$AdjustedR^2 = 1 - \frac{SC_{Err}/(n-d-1)}{SCT/(n-1)}.$$

Aquí, al igual que con el estadístico  $R^2$ , cuanto más grande sea el estadístico  $R^2$  ajustado, mejor. Una ventaja que tiene el estadístico  $R^2$  ajustado es su simplicidad. Sin embargo, a pesar de la popularidad del estadístico  $R^2$  ajustado, no está tan bien motivado -teóricamente hablando- si lo comparamos con los criterios antes mencionados.

## 4. Código en R

Veamos como aplicar estas ideas a una base de datos. Nos concentraremos en la base de datos `Hitters` del paquete `ISLR`. El primer paso es preparar la base de datos para que podamos trabajar en ella. Una observación importante es que esta base de datos tiene datos faltantes (indicados como `NA` en las bases de datos de R). Es posible que, para una observación dada, dispongamos de los valores de todas las variables (explicativas o dependientes), salvo para una o dos... y eso es suficiente para generar problemas. Por lo tanto, si la cantidad de datos faltantes es baja, una de las estrategias más usuales es ignorar las observaciones con algún dato faltante.

```
library(ISLR)
Hitters = Hitters
sum(is.na(Hitters)) # Cuenta cantidad de datos faltantes en la base de datos

## [1] 59

Hitters = na.omit(Hitters) # Removemos las observaciones con datos faltantes
sum(is.na(Hitters))

## [1] 0
```

La base de datos `Hitters` tiene información de salario de beisbolistas y varias estadísticas que se les toman. El objetivo es predecir el salario de los deportistas basados en sus estadísticas. Sin embargo, no esperamos que todas las estadísticas sean relevantes. Comenzaremos con la selección del mejor subconjunto de variables aleatorias. Usaremos el paquete `leaps` y su función `regsubsets`.

```

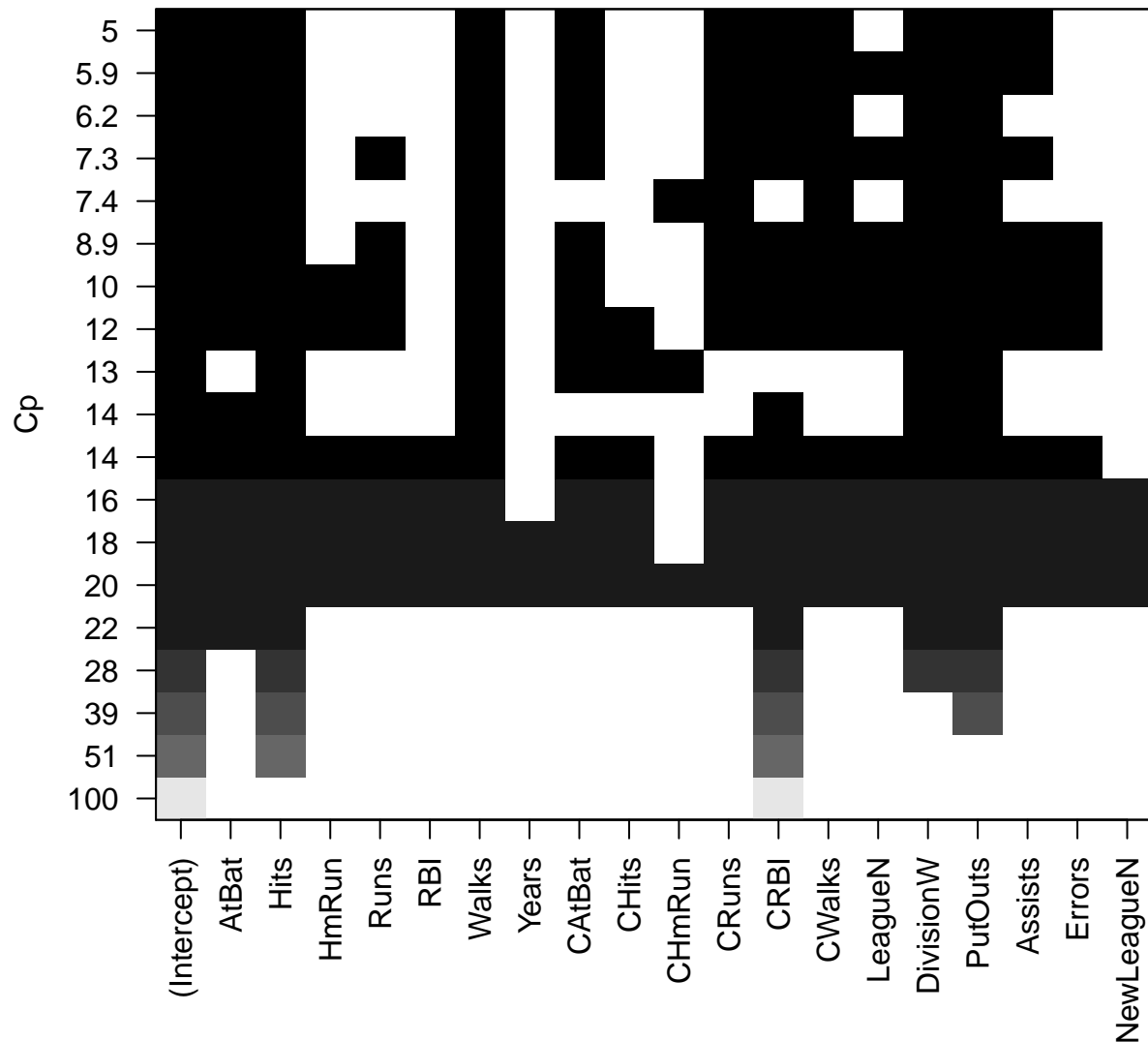
library(leaps)
reg_full = regsubsets(Salary ~ ., data = Hitters, nvmax = 19)
summary(reg_full)

## Subset selection object
## Call: regsubsets.formula(Salary ~ ., data = Hitters, nvmax = 19)
## 19 Variables (and intercept)
##           Forced in Forced out
## AtBat      FALSE      FALSE
## Hits       FALSE      FALSE
## HmRun      FALSE      FALSE
## Runs       FALSE      FALSE
## RBI        FALSE      FALSE
## Walks      FALSE      FALSE
## Years      FALSE      FALSE
## CAtBat     FALSE      FALSE
## CHits      FALSE      FALSE
## CHmRun     FALSE      FALSE
## CRuns      FALSE      FALSE
## CRBI       FALSE      FALSE
## CWalks     FALSE      FALSE
## LeagueN   FALSE      FALSE
## DivisionW  FALSE      FALSE
## PutOuts    FALSE      FALSE
## Assists    FALSE      FALSE
## Errors     FALSE      FALSE
## NewLeagueN FALSE      FALSE
## 1 subsets of each size up to 19
## Selection Algorithm: exhaustive
##           AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun CRuns
## 1 ( 1 ) " " " " " " " " " " " " " " " "
## 2 ( 1 ) " " "*" " " " " " " " " " " " "
## 3 ( 1 ) " " "*" " " " " " " " " " " " "
## 4 ( 1 ) " " "*" " " " " " " " " " " " "
## 5 ( 1 ) "*" "*" " " " " " " " " " " " "
## 6 ( 1 ) "*" "*" " " " " " " "*" " " " " "
## 7 ( 1 ) " " "*" " " " " " " "*" " " "*" "*" "
## 8 ( 1 ) "*" "*" " " " " " " "*" " " " " "*" "*"
## 9 ( 1 ) "*" "*" " " " " " " "*" " " " " "*" "*"
## 10 ( 1 ) "*" "*" " " " " " " "*" " " " " "*" "*"
## 11 ( 1 ) "*" "*" " " " " " " "*" " " " " "*" "*"
## 12 ( 1 ) "*" "*" " " "*" " " "*" " " " " " "*" "*"
## 13 ( 1 ) "*" "*" " " "*" " " "*" " " " " " "*" "*"
## 14 ( 1 ) "*" "*" "*" "*" " " "*" " " " " " "*" "*"
## 15 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" "*" " " "*"
## 16 ( 1 ) "*" "*" "*" "*" "*" "*" " " " " "*" "*" " " "*"
## 17 ( 1 ) "*" "*" "*" "*" "*" "*" " " " " "*" "*" " " "*"
## 18 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" " " "*"
## 19 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " " "*" "*" " " "*"
##           CRBI CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN
## 1 ( 1 ) "*" " " " " " " " " " " " "
## 2 ( 1 ) "*" " " " " " " " " " " " "
## 3 ( 1 ) "*" " " " " " " " " "*" " " " "
## 4 ( 1 ) "*" " " " " "*" " " "*" " " " " "
## 5 ( 1 ) "*" " " " " "*" " " "*" " " " " "
## 6 ( 1 ) "*" " " " " "*" " " "*" " " " " "
## 7 ( 1 ) " " " " " " "*" " " "*" " " " " "
## 8 ( 1 ) " " "*" " " " "*" " " "*" " " " " "
## 9 ( 1 ) "*" "*" " " " " "*" " " "*" " " " "

```

```
## 10 ( 1 ) "*" "*" " " "*" "*" "*" " " " "
## 11 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " "
## 12 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " "
## 13 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " "
## 14 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " "
## 15 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " "
## 16 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " " "
## 17 ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 18 ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*"
## 19 ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" *
```

`plot(reg_full, scale = "Cp")`



Podemos preguntar por los criterios de selección de modelos antes mencionados, entre otros.

```
summary(reg_full)$bic
## [1] -90.84637 -128.92622 -135.62693 -141.80892 -144.07143 -147.91690
## [7] -145.25594 -147.61525 -145.44316 -143.21651 -138.86077 -133.87283
## [13] -128.77759 -123.64420 -118.21832 -112.81768 -107.35339 -101.86391
## [19] -96.30412
```

Distintos criterios seleccionarán distintos modelos óptimos. Una vez que elegimos alguno de ellos basados en algún criterio, podemos pedir los coeficientes de dicho modelo. En el vector anterior observamos que el óptimo para el  $BIC$  se da para 6 variables explicativas, y en el gráfico observamos que el óptimo para  $C_p$  se da para 10 variables explicativas.

```
coef(reg_full, 6)
## (Intercept)      AtBat      Hits      Walks      CRBI
##  91.5117981 -1.8685892  7.6043976  3.6976468  0.6430169
##  DivisionW      PutOuts
## -122.9515338  0.2643076

coef(reg_full, 10)
## (Intercept)      AtBat      Hits      Walks      CAtBat
## 162.5354420 -2.1686501  6.9180175  5.7732246 -0.1300798
##      CRuns      CRBI      CWalks      DivisionW      PutOuts
##  1.4082490  0.7743122 -0.8308264 -112.3800575  0.2973726
##      Assists
##  0.2831680
```

También podemos usar `regsubsets` para las variantes ambiciosas. Observar que los modelos óptimos obtenidos para cada uno de los métodos es distinto.

```
reg_fwd = regsubsets(Salary ~ ., data = Hitters, nvmax = 19, method = "forward")
reg_bwd = regsubsets(Salary ~ ., data = Hitters, nvmax = 19, method = "backward")
summary(reg_fwd)

## Subset selection object
## Call: regsubsets.formula(Salary ~ ., data = Hitters, nvmax = 19, method = "forward")
## 19 Variables (and intercept)
##           Forced in Forced out
## AtBat      FALSE      FALSE
## Hits       FALSE      FALSE
## HmRun      FALSE      FALSE
## Runs       FALSE      FALSE
## RBI        FALSE      FALSE
## Walks      FALSE      FALSE
## Years      FALSE      FALSE
## CAtBat     FALSE      FALSE
## CHits      FALSE      FALSE
## CHmRun     FALSE      FALSE
## CRuns      FALSE      FALSE
## CRBI       FALSE      FALSE
## CWalks     FALSE      FALSE
## LeagueN   FALSE      FALSE
## DivisionW  FALSE      FALSE
## PutOuts    FALSE      FALSE
## Assists    FALSE      FALSE
## Errors     FALSE      FALSE
## NewLeagueN FALSE      FALSE
## 1 subsets of each size up to 19
## Selection Algorithm: forward
##           AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun CRuns
```

```

## 1 ( 1 ) " " " " " " " " " " " " " " " " " "
## 2 ( 1 ) " " "*" " " " " " " " " " " " " " "
## 3 ( 1 ) " " "*" " " " " " " " " " " " " " "
## 4 ( 1 ) " " "*" " " " " " " " " " " " " " "
## 5 ( 1 ) "*" "*" " " " " " " " " " " " " " "
## 6 ( 1 ) "*" "*" " " " " " "*" " " " " " " " "
## 7 ( 1 ) "*" "*" " " " " " "*" " " " " " " " "
## 8 ( 1 ) "*" "*" " " " " " "*" " " " " " " "*"
## 9 ( 1 ) "*" "*" " " " " " "*" " " "*" " " " " "*"
## 10 ( 1 ) "*" "*" " " " " " "*" " " "*" " " " " "*"
## 11 ( 1 ) "*" "*" " " " " " "*" " " "*" " " " " "*"
## 12 ( 1 ) "*" "*" " " "*" " " "*" " " "*" " " " " "*"
## 13 ( 1 ) "*" "*" " " "*" " " "*" " " "*" " " " " "*"
## 14 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" " " " " "*"
## 15 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" "*" " " " "*"
## 16 ( 1 ) "*" "*" "*" "*" "*" "*" " " "*" "*" " " " "*"
## 17 ( 1 ) "*" "*" "*" "*" "*" "*" " " "*" "*" " " " "*"
## 18 ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" " " " "*"
## 19 ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" " " "*"
##
##          CRBI CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN
## 1 ( 1 ) "*" " " " " " " " " " " " " " "
## 2 ( 1 ) "*" " " " " " " " " " " " " " "
## 3 ( 1 ) "*" " " " " " " " " "*" " " " " "
## 4 ( 1 ) "*" " " " " "*" " " "*" " " " " "
## 5 ( 1 ) "*" " " " " "*" " " "*" " " " " "
## 6 ( 1 ) "*" " " " " "*" " " "*" " " " " "
## 7 ( 1 ) "*" "*" " " "*" " " "*" " " " " "
## 8 ( 1 ) "*" "*" " " "*" " " "*" " " " " "
## 9 ( 1 ) "*" "*" " " "*" " " "*" " " " " "
## 10 ( 1 ) "*" "*" " " "*" " " "*" " " " " "
## 11 ( 1 ) "*" "*" "*" "*" " " "*" " " " " "
## 12 ( 1 ) "*" "*" "*" "*" " " "*" " " " " "
## 13 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" " "
## 14 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" " "
## 15 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" " "
## 16 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" " "
## 17 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" "*"
## 18 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" "*"
## 19 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" "*"

```

```
summary(reg_bwd)
```

```

## Subset selection object
## Call: regsubsets.formula(Salary ~ ., data = Hitters, nvmax = 19, method = "backward")
## 19 Variables (and intercept)
##          Forced in Forced out
## AtBat      FALSE      FALSE
## Hits       FALSE      FALSE
## HmRun       FALSE      FALSE
## Runs        FALSE      FALSE
## RBI         FALSE      FALSE
## Walks       FALSE      FALSE
## Years       FALSE      FALSE
## CAtBat      FALSE      FALSE
## CHits       FALSE      FALSE
## CHmRun      FALSE      FALSE
## CRuns       FALSE      FALSE
## CRBI        FALSE      FALSE
## CWalks      FALSE      FALSE

```

```

## LeagueN      FALSE      FALSE
## DivisionW    FALSE      FALSE
## PutOuts      FALSE      FALSE
## Assists      FALSE      FALSE
## Errors       FALSE      FALSE
## NewLeagueN   FALSE      FALSE
## 1 subsets of each size up to 19
## Selection Algorithm: backward
##           AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun CRuns
## 1 ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 2 ( 1 ) " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3 ( 1 ) " " "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 4 ( 1 ) "*" "*" " " " " " " " " " " " " " " " " " " " " " " " " " "
## 5 ( 1 ) "*" "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " "
## 6 ( 1 ) "*" "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " "
## 7 ( 1 ) "*" "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " "
## 8 ( 1 ) "*" "*" " " " " " " "*" " " " " " " " " " " " " " " " " " " "
## 9 ( 1 ) "*" "*" " " " " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 10 ( 1 ) "*" "*" " " " " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 11 ( 1 ) "*" "*" " " " " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 12 ( 1 ) "*" "*" " " "*" " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 13 ( 1 ) "*" "*" " " "*" " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 14 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" " " " " " " " " " " " " " " " "
## 15 ( 1 ) "*" "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 16 ( 1 ) "*" "*" "*" "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " "
## 17 ( 1 ) "*" "*" "*" "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " "
## 18 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " "*" " " "*" " " "*" " " " " " "
## 19 ( 1 ) "*" "*" "*" "*" "*" "*" "*" " " "*" " " "*" " " "*" " " " " " "
##           CRBI CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN
## 1 ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 2 ( 1 ) " " " " " " " " " " " " " " " " " " " " " " " " " " " " "
## 3 ( 1 ) " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " "
## 4 ( 1 ) " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " "
## 5 ( 1 ) " " " " " " " " " " " "*" " " " " " " " " " " " " " " " " "
## 6 ( 1 ) " " " " " " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 7 ( 1 ) " " "*" " " " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 8 ( 1 ) "*" "*" " " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 9 ( 1 ) "*" "*" " " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 10 ( 1 ) "*" "*" " " " " "*" " " "*" " " " " " " " " " " " " " " " "
## 11 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 12 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 13 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 14 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 15 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 16 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 17 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 18 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "
## 19 ( 1 ) "*" "*" "*" " " "*" " " "*" " " "*" " " " " " " " " " " " "

```